

UNIVERSIDADE FEDERAL DE SANTA CATARINA

RBC para o Problema de Suporte ao Cliente nas Empresas
de Prestação de Serviço de Software: O Caso Datasul

Gerson Volney Lagemann



0.295.776-2



UFSC-BU

Florianópolis, setembro de 1998

UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE
PRODUÇÃO PPGE/UFSC

**RBC para o Problema de Suporte ao Cliente nas Empresas de
Prestação de Serviço de Software: O Caso Datasul**

Dissertação submetida à Universidade Federal de Santa Catarina
para obtenção do grau de Mestre em Engenharia

GERSON VOLNEY LAGEMANN

Orientadora: Profa. Rosina Weber Lee, Dra.

Florianópolis, setembro de 1998

RBC para o Problema de Suporte ao Cliente nas Empresas de Prestação de Serviço de Software: O Caso Datasul

GERSON VOLNEY LAGEMANN

Esta dissertação foi julgada adequada para a obtenção do título de "Mestre em Engenharia", especialidade Engenharia de Produção e aprovada em sua forma final pelo Programa de Pós graduação.



Prof. Ricardo Miranda Barcia, Ph.D

Coordenador do Programa de Pós-graduação em Engenharia de Produção

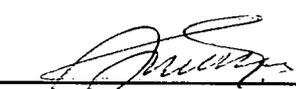
Banca Examinadora:



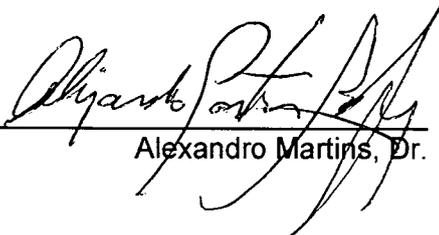
Rosina Weber Lee, Dra. – Orientadora



Vera Lúcia Duarte do Valle Pereira, Dra.



Fernando A. O. Gauthier, Dr.



Alexandro Martins, Dr.

Dedico este trabalho aos meus pais e a minha esposa.

AGRADECIMENTOS

A Profa. Rosina Weber pela orientação neste trabalho. Tenho certeza de que sem a sua colaboração e atenção os caminhos teriam sido muito mais árduos.

Ao Professor Fernando A. O. Gauthier pelo crédito depositado na minha pessoa.

Ao Departamento de Pós-Graduação em Engenharia de produção, na pessoa de seu coordenador, Prof. Ricardo Miranda Barcia, que acolheu minha proposta de mestrado e pela atenção e presteza sempre dedicada.

A Datasul S. A. na pessoa do seu diretor Sr. Adolfo Reimer pela disponibilização de toda a infra estrutura necessária para elaboração deste trabalho.

Aos colegas, amigos e professores que muito ajudaram com conhecimento, incentivo e amizade no decorrer do programa de mestrado.

À CAPES e a UDESC pelo auxílio financeiro.

SUMÁRIO

	Página
	vii
	viii
	ix
CAPÍTULO 1	1
Introdução	1
1.1 Generalidades.....	1
1.2 Objetivos.....	2
1.3 Importância do Trabalho.....	3
1.4 Organização do Trabalho.....	4
CAPÍTULO 2	6
Fundamentação Teórica	6
2.1 Inteligência Artificial.....	6
2.1.1 As Técnicas de IA.....	7
2.1.2 Aquisição do Conhecimento.....	8
2.1.3 Representação do Conhecimento.....	8
2.2 Raciocínio Baseado em Casos (RBC)	13
2.2.1 Representação do Conhecimento em RBC.....	16
2.2.2 Indexação.....	18
2.2.3 Recuperação.....	19
2.2.4 Adaptação.....	21
2.2.5 Aprendizagem.....	23
CAPÍTULO 3	25
Domínio	25
3.1 Suporte ao Cliente.....	25
3.1.1 O Problema.....	28
3.2 O Caso Datasul.....	30
3.2.1 A Empresa.....	30
3.2.2 Clientes.....	31
3.2.3 Empregados.....	31
3.2.4 Produtos.....	31
3.2.5 Serviços.....	32
3.3 O Problema de Suporte da Datasul.....	33
3.3.1 O Suporte.....	33
3.3.2 A Implementação.....	35
3.4 RBC e Suporte Telefônico.....	36
3.5 Base de Dados X Base de Casos.....	39
CAPÍTULO 4	43
Implementação	43
4.1 Introdução.....	43
4.2 Definição do Sistema.....	44
4.3 Representação e Indexação dos Casos.....	45
4.3.1 O Banco de Dados de FO's.....	45
4.3.2 A Base de Casos.....	47
4.3.3 A Transformação dos Casos.....	48
4.3.4 A Definição dos Índices.....	51
4.3.5 Avaliação da Similaridade e Recuperação.....	53
4.4 Exemplo.....	58
4.4.1 A Função de Recuperação.....	59
4.5 Resultados.....	62
CAPÍTULO 5	64
Conclusão e Desenvolvimentos Futuros	64
5.1 Desenvolvimentos Futuros.....	66
Bibliografia.....	68

Apêndice A.....	72
Apêndice B.....	73
Apêndice C.....	74
Apêndice D.....	75

LISTAS

LISTA DE FIGURAS

	Página
Figura 2-1 O Ciclo de RBC.....	16
Figura 2-2 O Espaço do Problema e da Solução.....	17
Figura 3-1 <i>Help Desk</i> com múltiplos pontos de contato.....	26
Figura 3-2 <i>Help Desk</i> com único ponto de contato.....	27
Figura 3-3 Relacionamentos e Tabelas.....	40
Figura 4-2 Modelo Entidade Relacionamento (MER) do SHD.....	51
Figura 4-3 Exemplo da estrutura da lista de sinônimos.....	57
Figura 4-4 Janela principal de apresentação do software SHD.....	58
Figura 4-5 Janela Help Desk.....	59
Figura 4-6 Uma possível consulta ao SHD.....	60
Figura 4-7 Caso recuperado (similaridade = 63.44%).....	61
Figura 4-8 Caso recuperado (similaridade = 51,99%).....	61

LISTA DE TABELAS

	Página
Tabela 4-1 Tipos da FO's.....	46
Tabela 4-2 Registros por produto na base de dados de FO's.....	47
Tabela 4-3 Estrutura da base de dados de FO's.....	48
Tabela 4-4 Atributos da base de Casos.....	48
Tabela 4-5 Estrutura de dados do SHD.....	50
Tabela 4-6 Estrutura do dicionário de sinônimos.....	56
Tabela 4-7 Resultados do SHD.....	62

GLOSSÁRIO

- CBEMD** : Case Based Equipment Malfunction Diagnosis (Base de Casos de Diagnósticos sobre o mau funcionamento de equipamentos).
- CI** : Centro de Informações
- e.g.** : “exempli gratia” (por exemplo)
- FO** : Ficha de Ocorrência
- i.e.** : “id est” (isto é)
- MER** : Modelo Entidade Relacionamento
- RBC** : Raciocínio Baseado em Casos
- SHD** : Sistema de Help Desk
- SQL** : Structured Query Language (Linguagem estruturada de consulta)

1. INTRODUÇÃO

1.1 Generalidades

Este trabalho é resultado da aplicação dos conhecimentos adquiridos no Programa de Pós-Graduação em Engenharia de Produção e Sistemas, área de concentração de Inteligência Artificial, da Universidade Federal de Santa Catarina.

Inteligência Artificial (IA) é a parte da Engenharia da Produção e da Ciência da Computação que se destina a desenvolver programas computacionais inteligentes. Um programa inteligente é aquele capaz de resolver um problema que exija uma manifestação tipicamente humana, como o raciocínio dedutivo, por exemplo .

Entre as principais pesquisas em IA estão: Sistemas Especialistas, Processamento da Linguagem Natural, Redes Conexionistas, Computação Evolutiva, Conjuntos Difusos, Agentes Inteligentes e Raciocínio Baseado em Casos. A técnica de Raciocínio Baseado em Casos (RBC) é usada neste trabalho para implementar uma solução ao problema de suporte ao cliente nas empresas de prestação de serviços de *software*.

O raciocínio baseado em casos (RBC) é uma técnica usada para encontrar soluções para um novo problema usando soluções adotadas para problemas similares já resolvidos [RIE89]. Em um sistema de RBC, primeiramente são pesquisados e encontrados problemas similares antigos já resolvidos, obtidos de uma base de casos. Depois, o sistema ajusta a solução considerando as diferenças entre o problema novo e o antigo. A idéia básica de RBC é que os humanos modificam a estratégia de solução de um novo problema baseados em

conhecimento e experiências com problemas passados similares, adaptando as soluções, usando-os para explicar novas situações, criticando novas soluções, ou inferindo sobre fatos para interpretar novas situações ou criar uma solução para o problema atual [KOL93].

1.2 Objetivos

O objetivo geral deste trabalho consiste em melhorar a prestação de serviço de suporte ao cliente através de um sistema chamado Sistema de Help Desk (SHD). O SHD possibilita ao usuário, a partir de uma base de casos já existente, realizar buscas e encontrar problemas muito similares aos seus, já resolvidos. Deve ser, portanto, capaz de aumentar o nível de satisfação dos clientes do suporte, diminuindo os preços dos seus produtos e serviços, possibilitando entrega efetiva e rápida com conseqüente aumento da qualidade.

A principal característica da aplicação de suporte ao cliente, que justifica este trabalho, é que o produto das empresas de *software* é grande e complexo, distribuído a muitos clientes e com inúmeros especialistas revezando-se no suporte. O suporte ao cliente exige especialistas com profundo conhecimento do negócio (funções implementadas) e do próprio *software*. O suporte ao cliente exige também um pronto atendimento ao cliente. Observa-se que os melhores técnicos não estão em atividades produtivas, como, por exemplo, o desenvolvimento de novos produtos. O grande número de especialistas exige a administração do conhecimento. Enfim, estes são alguns dos problemas encontrados atualmente em empresas prestadoras de serviço de *software* no que se refere ao atendimento aos clientes e que justifica este trabalho.

Com a implementação do SHD busca-se reduzir os custos de manter especialistas no suporte, considerando que funcionários não especialistas usando o sistema terão plenas condições de atender os clientes satisfatoriamente. Ao substituir os especialistas do suporte, estes estarão disponíveis para atividades produtivas e mais criativas, reduzindo conseqüentemente a insatisfação destes por trabalharem no suporte. Na busca para atingir os objetivos propostos procurar-se-á utilizar informações já disponíveis em bancos de dados convencionais e toda a infra-estrutura computacional (linguagem de programação e gerenciador de banco de dados) já existentes na empresa, característica inovadora e diferencial deste trabalho.

1.3 Importância do Trabalho

Melhorar o atendimento ao cliente é, no momento atual, a maior preocupação das empresas. Para atingirem este objetivo, as empresas colocam à disposição dos seus clientes ferramentas que agilizam esta interação. Entre as ferramentas mais usadas estão: suporte telefônico, telefones gratuitos e Internet. O SHD passa a integrar o rol destas ferramentas. Com o SHD, o suporte telefônico, que entre as ferramentas disponibilizadas é a que representa o maior custo, pode ser reduzido em até 80%.

Uma importante vantagem da implementação deste sistema está na associação de novas tecnologias, IA e RBC, com o aproveitamento de bases de dados e ferramentas de *softwares* (sistema operacional, gerenciador de banco de dados e linguagens de programação) já existentes na empresa. Outra vantagem está na melhora da prestação de serviço ao cliente. Porém, a principal vantagem

está na redução dos custos com o suporte, tanto para o fornecedor, pelo alto custo despendido em manter canais de comunicações com especialistas para atender aos clientes, quanto para os clientes, pelo custo em fazer as chamadas e o tempo gasto na espera de um retorno.

1.4 Organização do Trabalho

Este trabalho está estruturado em cinco capítulos: Introdução, Fundamentação Teórica, Domínio, Implementação e Conclusão.

A introdução, apresentada no capítulo um, consiste das seções Generalidades, Objetivos, Importância do Trabalho e Organização do Trabalho.

O segundo capítulo consiste na fundamentação dos princípios da IA, que ajudaram na definição da técnica de RBC para resolver o problema e nos conceitos da técnica de RBC.

O terceiro capítulo descreve o domínio da aplicação onde será utilizada a técnica de RBC. É revisado o atual funcionamento do suporte ao cliente, é caracterizado o ambiente do caso escolhido, e são relacionados os casos de sucesso de sistemas de RBC para suporte e a utilização de bancos de dados com bases de casos.

O quarto capítulo apresenta a implementação da solução ao caso específico da Datasul. É detalhado, passo a passo, como cada uma das etapas da técnica de RBC foi implementada. É apresentado o produto SHD resultante deste trabalho e exemplificada uma situação fictícia detalhando o funcionamento do *software*.

O quinto capítulo conclui este trabalho ressaltando as contribuições e recomendações para futuras implementações, garantindo a continuidade do mesmo.

2. FUNDAMENTAÇÃO TEÓRICA

O objetivo principal deste capítulo é apresentar os conceitos da Inteligência Artificial (IA) e do Raciocínio Baseado em Casos (RBC) aplicados na solução do problema deste trabalho. São introduzidos alguns tópicos de IA e RBC considerados importantes para a compreensão de suas utilidades no domínio de suporte ao cliente.

2.1 Inteligência Artificial (IA)

A Inteligência Artificial (IA) é um ramo da Ciência da Computação e oficialmente nasceu em 1956, porém ela é muito mais antiga, pois foi construída a partir de idéias filosóficas, científicas e tecnológicas herdadas de outras ciências, algumas tão antigas quanto a lógica [BIT96].

O objetivo principal da IA é a criação de modelos para a inteligência e a construção de sistemas computacionais baseados nestes modelos. Para alcançar este objetivo, a IA dividiu-se em três atividades[BIT96]:

- Desenvolvimento de modelos formais para a inteligência humana, tema da ciência cognitiva, também chamada Psicologia Computacional.
- Desenvolvimento de aplicações educacionais, comerciais ou industriais utilizando técnicas de IA.
- Exploração e experimentação de técnicas computacionais que representem potencial para a simulação do comportamento inteligente, a chamada IA Básica.

A ênfase atual da IA está na passagem da pesquisa para a implementação [BIT96]. Nesta passagem, as técnicas de IA desenvolvidas em laboratório usando linguagens como Prolog e Lisp precisam ser também implementadas com linguagens comerciais, tipo C e C++, por exemplo. Este trabalho se insere neste contexto, pois desenvolve uma aplicação utilizando técnicas de IA e utiliza uma linguagem comercial para o seu desenvolvimento.

2.1.1 As técnicas da IA

Os pesquisadores de IA nos últimos anos têm se preocupado mais com a aplicação dos conceitos da IA no mundo real do que com o desenvolvimento de novos teoremas [BIT96]. Estas novas pesquisas têm dividido a IA em muitos ramos [RUS95]. Entre eles estão, Sistemas Especialistas, Processamento da Linguagem Natural, Redes Conexionistas, Algoritmos Genéticos, Conjuntos Difusos, Agentes Inteligentes e Raciocínio Baseado em Casos.

O Raciocínio Baseado em Casos (RBC), detalhado na seção 2.2, técnica que é objeto deste trabalho, é um sistema que soluciona novos problemas através da adaptação de soluções que foram usadas para resolver problemas passados [RIE89].

Escolheu-se RBC, entre as técnicas de IA, para criar uma alternativa de solução ao problema de suporte. A razão que levou a utilização de RBC para o problema de suporte está nos vários casos de sucesso [WAT97] já desenvolvidos em grandes empresas internacionais.

Nas seções 2.1.2 e 2.1.3 são apresentados os conceitos sobre Aquisição e a Representação do Conhecimento, pontos importantes que conduziram à escolha de RBC para implementar a solução ao problema de suporte.

2.1.2 Aquisição do Conhecimento

A aquisição do conhecimento é a tarefa executada por engenheiros de conhecimento com o objetivo de extrair conhecimento especialista para ser representado em sistemas inteligentes [WAT86]. Os Sistemas Especialistas (SE) compreendem os sistemas baseados em conhecimento que executam tarefas especialistas [DUR94]. As necessidades de aquisição de conhecimento em RBC são reduzidas, se comparadas com os Sistemas Especialistas (SE), isto devido a pouca necessidade de representação do conhecimento [LEA96].

A aquisição do conhecimento em um sistema RBC pode ser tanto extensa quanto praticamente inexistente, variando de acordo com a tarefa e o domínio da aplicação. No presente trabalho, utiliza-se o conhecimento já armazenado em bancos de dados, restando apenas definir a representação do conhecimento.

2.1.3 Representação do Conhecimento

Para resolver seus problemas os humanos vão acumulando conhecimento desde o nascimento, que lhes permite agir de modo inteligente. Geralmente pensa-se que a capacidade dos humanos em resolver seus problemas é simplesmente pela acumulação de conhecimento. Porém, este processo não é tão simples, a armazenagem deste conhecimento envolve um ciclo completo de

processamento da informação, que vai desde a coleta do conhecimento pelos sentidos, até seu armazenamento definitivo no cérebro [RAB95].

Tratar conhecimento com mecanismos artificiais, como os computadores, não é tarefa fácil. Para se manipular o conhecimento é primordial que se consiga formas de representá-lo. A representação do conhecimento é uma das áreas mais ativas da IA, envolvendo os maiores desafios [RUS95].

Os pesquisadores de IA têm usado os resultados alcançados pela psicologia cognitiva, sobre as teorias de como os humanos resolvem seus problemas. O objetivo destas pesquisas é desenvolver técnicas que melhor representem os diferentes tipos de conhecimento humano no computador. Assim, como não existe uma única teoria para explicar a organização do conhecimento humano ou a melhor técnica para representar dados em um sistema computacional convencional, assim também não há uma estrutura de representação do conhecimento ideal [RIC93].

Uma boa representação do conhecimento em um determinado domínio deve possuir as seguintes propriedades [RIC93]:

- Adequação Representacional: capacidade de representar todos os tipos de conhecimento necessários naquele domínio.
- Adequação Inferencial: capacidade de manipular as estruturas representacionais de modo a derivar novas estruturas que correspondam a novos conhecimentos, inferidos a partir de conhecimentos antigos.
- Eficácia Inferencial: capacidade de incorporar à estrutura de conhecimento informações adicionais que podem ser usadas para focalizar a atenção dos mecanismos de inferência nas direções mais promissoras.

- Eficácia aquisitiva: capacidade de adquirir novas informações facilmente. O caso mais simples envolve a inserção direta, por meio de uma pessoa, de novos conhecimentos na base de dados. O ideal seria se o próprio programa fosse capaz de controlar a aquisição de conhecimento.

Até o momento os pesquisadores não conseguiram reunir em uma única técnica de representação do conhecimento as quatro propriedades citadas acima. Como resultado existem várias técnicas para representação do conhecimento [RIC93]:

- Conhecimento Relacional Simples, a maneira mais simples de representar conhecimento declarativo é através de um conjunto de relações idêntico ao usado por sistemas de banco de dados relacionais. Este tipo de representação do conhecimento é simples pois devido a sua independência fornece capacidades inferenciais muito fracas. Talvez um fator importante deste tipo de representação seja a vantagem de usá-lo como entrada para mecanismos de inferência mais poderosos.
- Conhecimento Herdável, uma das formas mais úteis de inferência é a herança de atributos, onde os elementos de classes específicas herdam atributos e valores de classes mais genéricas em que estão incluídos. Os objetos são definidos a partir de um conjunto de atributos e valores que associados descrevem a base de conhecimento. Para suportar a herança os objetos precisam organizar-se em classes que precisam ser organizadas em uma hierarquia de generalizações.
- Conhecimento Inferencial, a herança é uma forma poderosa de inferência. Porém, algumas vezes é necessário a força da lógica tradicional para

descrever as inferências necessárias. Qualquer conhecimento para ser útil necessita um procedimento de inferência que consiga explorá-lo. O procedimento de inferência deve implementar as regras de inferência lógica padrões. Existe vários procedimentos deste tipo, alguns que raciocinam a partir de fatos para chegar as conclusões e outros que partem da conclusão desejada para os fatos fornecidos (comprovação de hipótese). Um dos procedimentos mais comumente usados é a resolução, que explora uma prova através da estratégia da contradição.

- Conhecimento Procedimental, o conhecimento procedimental ou operacional especifica o que fazer e quando. Este tipo de conhecimento pode ser representado em programas de diversas maneiras, sendo mais comum simplesmente como código sobre como fazer algo. A máquina usa o conhecimento quando executa o código para realizar uma tarefa. Esta forma de representação do conhecimento deixa muito a desejar em relação às propriedades de adequação inferencial e em relação a eficiência aquisitiva.

Para definir alguma técnica de representação do conhecimento, primeiramente é necessário preocupar-se com o problema a tratar [RIC93]. Entre os aspectos importantes presentes no problema devem ser analisados:

- Há ocorrência de atributos nos objetos do domínio do problema? Se houver, quais são e há tratamento adequado com o mecanismo que se esta propondo?
- Existem relações importantes entre os atributos dos objetos?
- Qual o nível necessário para representação do conhecimento?
- Existe um bom conjunto de primitivas das quais pode ser derivado o conhecimento?

- Como devem ser representados os conjuntos de objetos?
- Existindo uma grande base de dados com conhecimento armazenado, como acessá-la quando for necessário?

Para representar conhecimento em RBC pode-se utilizar redes semânticas, *scripts* e *frames*, entre outras [WEB98]. Regras, que representam conhecimento *procedural* em SE's, também podem ser usadas para representar conhecimento em sistemas RBC [LEE96] [LEN96].

Pacotes de organização de memória (MOP's – Memory Organization Packets) é um conceito de representação de conhecimento desenvolvido para representar eventos estereotípicos [SCH82]. O conceito de MOP's é uma evolução no conceito de *scrips* [KOL93]. Os MOP's são estruturas de conhecimento que representam experiências, onde os eventos são representados por cenas incluindo situações representadas por informação narrativa e descritiva.

Redes semânticas são grafos direcionados ligados por nós para representar objetos e conexões que representam a relação entre objetos [QUI68] [BRA79] [DEA95]. Podem ser usadas para representar uma classe, suas instâncias e suas características. Os arcos podem representar relações e atributos.

Frames é uma estrutura de dados que representa uma entidade através de suas características e potencialidades [WEB98]. As características são representadas por pares atributo-valor, e as potencialidades por métodos.

Regras são seqüências lógicas de fatos compostas por antecedentes (premissas) e conseqüentes (conclusões). Uma conclusão é disparada toda vez que um antecedente comprova que um fato é verdadeiro. Fatos podem ser conectados pelos operadores lógicos E, OU e NÃO [WEB98]. As conclusões

normalmente atualizam atributos de objetos, disparam métodos ou novas regras [WAT86] [DUR94].

A representação formulário é composta por um conjunto de campos valorados, lembrando linhas e colunas das tabelas dos bancos de dados relacionais. Com este tipo, pode-se representar casos em estruturas organizacionais planas [KOL93].

Concluindo, somando-se a intenção do autor em trabalhar numa área da IA onde há carência de trabalhos, mais os casos de sucesso de RBC no domínio de suporte [WAT97] [LEE96] [LEN96] e mais o tratamento que RBC dá a Aquisição e a Representação do Conhecimento, justifica-se RBC para elaboração deste trabalho.

2.2 Raciocínio Baseado em Casos (RBC)

A origem da técnica RBC é atribuída a uma pesquisa na área da ciência cognitiva desenvolvida por Schank e Abelson em 1977. Neste trabalho eles afirmaram que todo o conhecimento é armazenado no cérebro como um *script*. Estes *scripts* compõem uma memória conceitual, descrevendo informações sobre eventos típicos como um almoço em um restaurante ou uma consulta médica [SCH77]. Porém, experimentos com fatos mostraram que somente os *scripts* não representavam o conhecimento armazenado na memória. Era comum as pessoas misturarem fatos parecidos como consulta médica e consulta dentária no momento da lembrança.

Roger Schank continuou pesquisando sobre o papel que a lembrança de fatos passados (i.e., casos) e situações padrões ou memória organizada

desempenham na solução e na aprendizagem dos problemas [SCH77]. Uma vez entendido o que foi colocado por Schank, talvez seja possível encontrar referências significantes para RBC nas observações de Wittgenstein sobre conceitos naturais de que, tais como mesas e cadeiras são de fato polimórficos e não podem ser classificados por um único conjunto de características, mas podem ser definidos por um conjunto de instâncias (i.e., casos) que tenham semelhanças [WIT53].

Um forma bastante clara de explicar raciocínio baseado em casos é encontrada no exemplo:

“Um turista francês em visita ao Rio de Janeiro estava planejando visitar amigos em Petrópolis, cidade serrana daquele estado. Havia alugado um carro já há alguns dias, mas ainda não estava familiarizado com o trânsito. Solicitou informações no hotel onde estava hospedado sobre como chegar a Petrópolis. Explicaram-lhe detalhadamente como chegar a Av. Brasil. Não conseguiram, entretanto, precisar exatamente em que altura da Av. Brasil deveria pegar a BR316, rodovia que o levaria a região serrana. Ele lembrou porém, que certa vez havia tido dificuldades para sair de uma rodovia expressa na França. Naquela vez, trafegava em alta velocidade pela pista central da via e quando avistou a saída não teve mais tempo suficiente para pegá-la. Lembrou também, que na segunda vez que pretendia pegar aquela mesma saída, andou muitos quilômetros lentamente na faixa da direita para não perdê-la. Perdeu tempo precioso, mas notou que a uns 500 metros antes da saída havia um enorme “out door” da Coca-cola. Na terceira vez, andou pela faixa central até o momento que avistou o “out door” e então tomou a faixa da direita a tempo de pegar a saída. Esta lembrança,

Ihe sugere que pergunte aos seus informantes a respeito de algum ponto de referência importante nas proximidades da saída. É informado de que há um posto de combustíveis da Texaco aproximadamente a 1 quilômetro antes da saída. Perfeito, dirigiu rapidamente pela faixa da esquerda até avistar o posto, tomou então a faixa da direita e pegou a BR316 no local exato.”

Neste caso hipotético, o turista está usando raciocínio baseado em casos para planejar o seu passeio a Petrópolis. No raciocínio baseado em casos, o indivíduo lembra de situações passadas semelhantes a atual e reutiliza-as adaptando a solução para resolver o problema atual.

O mecanismo de inferência de RBC soluciona novos problemas através da adaptação de soluções que foram usadas para resolver problemas passados [RIE89].

É possível simplificar o processo mental de RBC como sendo cíclico e composto por quatro “R” [AAM94]: (ver figura 2-1)

- Recuperar o(s) caso(s) mais similar(es);
- Reusar o(s) caso(s) para resolver o problema;
- Revisar a solução proposta, se necessário, e;
- Reter a nova solução incluindo-a na base de casos.

Outros autores [LEA96] referem-se ao ciclo de RBC através das tarefas de recuperar, propor, criticar, justificar, avaliar, adaptar e armazenar. Este ciclo raramente ocorre sem a participação de pelo menos uma pessoa [WAT97]. Muitas ferramentas e aplicações que usam RBC automatizam a primeira etapa do ciclo, ou seja, a recuperação e a reutilização. Já a revisão (i.e., adaptação) e a decisão

sobre retenção do novo caso (i.e., aprendizagem) é feita pelo próprio usuário [WAT97].

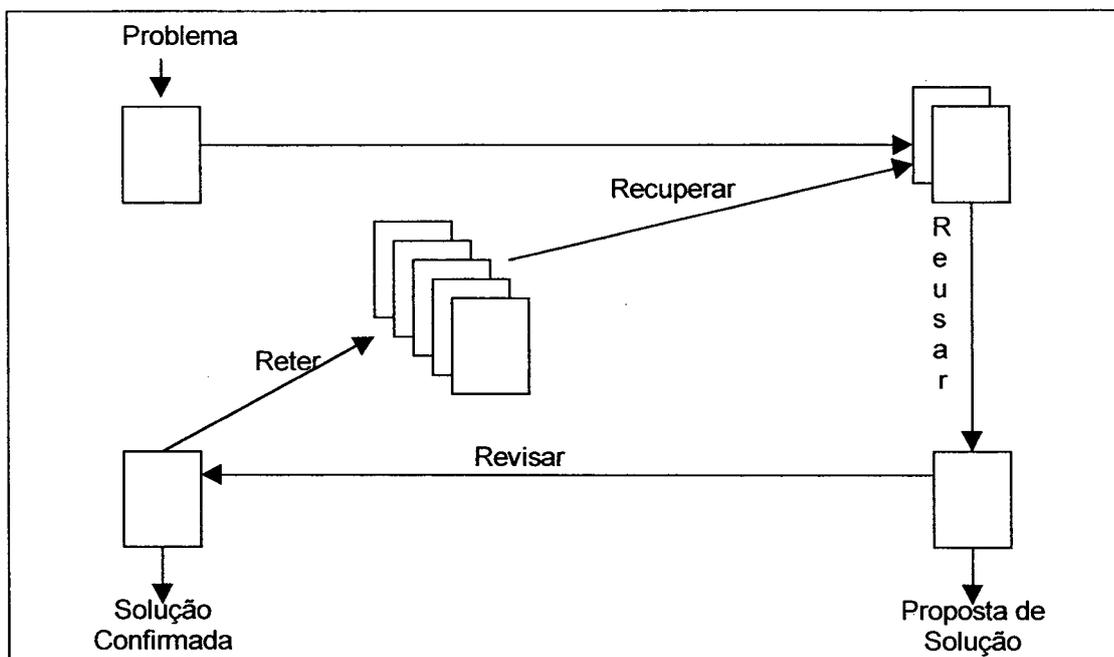


Figura 2-1 O ciclo do RBC [WAT97]

2.2.1 Representação do Conhecimento em RBC

Nesta seção será definido o que é um caso para RBC e serão enumerados seus componentes.

Um caso é definido pela representação do conhecimento contido em uma experiência vivida que dirige o indivíduo a alcançar seus objetivos [LEA96]. Todo caso é composto por: *Problema*: descreve o estado do mundo real onde o caso ocorre; e, *Solução*: contém o estado das soluções derivadas para o problema [WAT97]. Pode-se ainda, visualizar o caso pela representação do espaço do problema e pela representação do espaço da solução, veja figura 2-2.

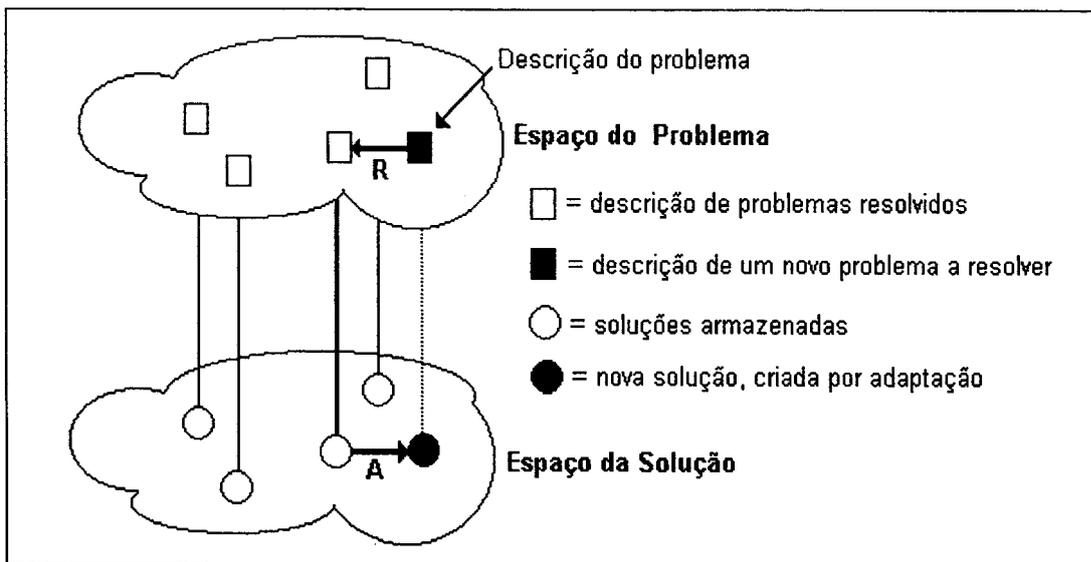


Figura 2-2 – O Espaço do Problema e da Solução [WAT97]

Os casos têm por origem a derivação de um problema e da respectiva solução, e suas partes são representadas pelos seus espaços. Quando ocorrer a recuperação, o caso poderá ainda ser adaptado dando origem a um novo caso que poderá ou não ser incorporado a base de casos para solucionar novos problemas. Pode-se ainda afirmar que o modelo RBC assume um relacionamento $1 \rightarrow 1$ no que se refere a Problema e Solução [LEA96].

Para representar os casos pode-se armazenar dados utilizando os diferentes tipos usados pelos computadores. Arquivos e Bancos de Dados podem ser usados para armazenar informações sobre os casos, tais como: nomes, códigos de produtos, valores etc, em arquiteturas de representação plana.

Existe ainda uma indefinição entre os pesquisadores de IA sobre que informações exatamente devem fazer parte de um caso [WAT97]. Entretanto, é muito importante no momento de decidir o que irá definir um caso, levar em consideração a funcionalidade da informação e a sua facilidade de aquisição.

2.2.2 Indexação

As informações são indexadas para que possam mais fácil e rapidamente serem recuperadas. Em livros, por exemplo, há a preocupação com a correta identificação das necessidades futuras de recuperação para melhor definir quais informações farão parte dos índices. Devido a esta preocupação tem-se em livros, normalmente, um índice por assunto (capítulos) e no final o índice remissivo por palavra.

A indexação é a essência do RBC, pois orienta a avaliação da similaridade [WEB98]. A indexação determina o que comparar entre os casos para determinar sua similaridade. Assim, RBC além de usar índices com os objetivos de facilidade e rapidez na recuperação [WAT97], também usa-os para realizar eficientemente a atribuição de similaridade entre os casos.

Em Leake [LEA96], pode-se verificar que a abordagem dada à indexação para criação de índices em RBC é diferente da usada em bancos de dados. Nestes, o principal objetivo da indexação é balancear a estrutura organizacional, dividindo o conjunto de dados em partições com tamanhos aproximados. Já em RBC, a abordagem principal é criar índices para diferenciar um caso do outro atendendo aos vários objetivos das recuperações futuras.

As informações contidas em um caso podem ser de dois tipos [WAT97]:

- Informações indexadas para uso na recuperação, e;
- Informações não indexadas que fornecem dados não necessários para recuperação.

Por exemplo, em um sistema médico tem-se informações tipo: idade, sexo, peso e altura, que são fortes candidatas a futuras necessidades de recuperação. Entretanto, há outras do tipo: fotografia do paciente, que jamais serão usadas na recuperação em um sistema médico que use RBC.

Bons índices devem satisfazer as seguintes propriedades [KOL93]:

- Devem ser suficientemente abstratos a fim de recuperar casos relevantes numa grande variedade de situações futuras, e;
- Devem ser suficientemente concretos para serem facilmente identificados em situações futuras.

Uma difícil tarefa na definição dos índices é prever que tipos de necessidade irão surgir e que tipos de informações serão necessárias para recuperar casos em situações futuras. Muitos esforços foram feitos para estabelecer regras gerais de vocabulário de índices em classes particulares nas tarefas de RBC [DOM92] [LEA92] [SCH90], mas esta tarefa acaba ainda sendo desenvolvida para atender os objetivos específicos da recuperação de cada aplicativo que use RBC.

2.2.3 Recuperação

Uma característica importante dos sistemas de RBC é possuir alternativas para identificar os casos a fim de conseguir representá-los e indexá-los, garantindo que sejam recuperados os mais úteis para resolver o problema do usuário. Somente consegue-se alternativas para identificar os casos através de procedimentos de comparação e medição de similaridades [LEA96]. A determinação da medida de similaridade é um importante componente para

identificar a utilidade do caso. Deve-se considerar também, que o grau de utilidade de um caso depende dos propósitos a que ele se destina e quais dos seus aspectos foram relevantes no passado. Estas considerações habilitam os procedimentos de comparação a determinar em que dimensão é importante para um caso focá-lo no julgamento da similaridade [LEA96].

Considerando que a definição dos índices retrata todos os aspectos a serem considerados na recuperação, os algoritmos de comparação poderão usá-los para se orientarem na busca, determinando quais características devem ser focadas no julgamento da similaridade. Qualquer caso pode ser indexado de várias formas, entretanto, o importante é que o algoritmo de comparação esteja apto a distinguir entre os aspectos relevantes em cada caso específico.

No processo de recuperação não se pode falar apenas em comparação e medição da similaridade, precisa-se falar também sobre as estruturas que serão usadas pelos algoritmos para dirigir a busca. Na recuperação são usadas duas técnicas de busca: Vizinho-mais-próximo e Indutiva [RIC93].

No tipo de busca do vizinho-mais-próximo os aspectos de definição e identificação dos índices é fator fundamental para uma recuperação de sucesso. Garantidos estes aspectos a técnica de busca indica em qual região do espaço de busca dos problemas o problema em questão está inserido. O próximo passo é por comparação e valorização das similaridades encontrar aqueles mais parecidos [WAT97].

No tipo de busca indutiva constroem-se árvores de decisão baseadas em dados de problemas passados. Em sistemas RBC a base de casos é analisada por um algoritmo de indução, que cria a árvore de decisão classificando ou

indexando os casos. O algoritmo mais usado pelos sistemas de RBC é o ID3 [QUI86]. Para que o ID3 construa a árvore de decisão a partir dos casos da base de casos, é necessário passar-lhe os atributos que melhor identificam os casos. Encontrado o primeiro atributo o ID3 monta o 1º nó da árvore. O passo seguinte é encontrar dois novos atributos que formem os próximos nós e assim por diante. Montada a árvore a partir da base de casos, o próximo passo é percorrer a árvore com o caso em questão. Quando chegar no último nó da árvore teremos os casos mais similares.

Como é possível verificar as técnicas vizinho-mais-próximo e indutiva são muito boas. Porém, cada um delas têm características que mais se apropriam para determinados tipos de problemas. A técnica do vizinho mais próximo é mais indicada para problemas com bases de casos pequenas e com poucos atributos indexados, devido ao volume de cálculos necessários para determinar cada um dos atributos indexados e cada um dos casos. A técnica indutiva, por sua vez, é bem mais rápida, somente ficando lenta para bases de casos muito grandes. O principal problema desta técnica é que casos inéditos não recuperarão nada [WAT97].

2.2.4 Adaptação

Pelo fato de nenhum problema passado ser exatamente igual a um problema atual, soluções passadas usualmente são adaptadas para solucionar novos problemas [KOL93]. A Adaptação poder ser uma simples substituição de um atributo da solução por outro ou uma complexa e total modificação na estrutura da solução.

A adaptação pode ser feita de várias formas [WAT97]:

- Pela inclusão de um novo comportamento à solução recuperada;
- Pela eliminação de um comportamento da solução recuperada, ou;
- Pela substituição de parte de um comportamento.

Podem ocorrer situações onde há a presença de mais de uma das formas.

Os métodos para proceder a adaptação podem ser classificados da seguinte forma [LEA96] [KOL93]: Métodos de Substituição, Método da Transformação, Objetivos Especiais de Adaptação e Modificação e Repetição Derivacional.

Apesar da adaptação poder ser usada de várias formas e em várias situações, ela não é essencial [WAT97]. Muitos dos sistemas comerciais de RBC não implementam a adaptação. Eles simplesmente recuperam o caso mais similar e disponibilizam a solução para o usuário, deixando-o livre para proceder a adaptação. Isto acontece devido a grande complexidade da adaptação. Na maioria das vezes, para implementar a adaptação é necessário representar um volume muito grande de conhecimento.

Uma alternativa para a adaptação é o sistema RBC perguntar ao usuário se ele deseja que o sistema faça adaptação. Caso ele responda que sim, então que a adaptação seja feita, mas que após seja dada ao usuário a possibilidade de interagir com as modificações, decidindo por sua aplicação total, parcial ou ainda descartando-a.

2.2.5 Aprendizagem

A aprendizagem em um sistema de RBC acontece principalmente pela acumulação de novas experiências em sua memória e pela correta indexação dos problemas [KOL93]. Um sistema de RBC somente se tornará mais eficiente quando estiver preparado para, a partir das experiências passadas e da correta indexação dos problemas, aprender.

A implementação da aprendizagem em um sistema RBC está baseada no fato de que a partir das experiências passadas o sistema esteja apto a analisar os efeitos da sua solução e armazenar informações sobre o que deu certo, o que não deu certo e porque. Com estes procedimentos o programa poderá montar melhores respostas. Isto torna o sistema RBC muito mais confiável, a medida que antecipa erros cometidos no passado.

Na IA, quando fala-se de aprendizagem, é comum pensar no aprendizado por generalização, tanto pela indução quanto baseado em explicações. Enquanto a memória de RBC informa sobre similaridades entre casos e também informa sobre quando generalizações podem ser formadas. A formação de generalizações indutivas é responsável apenas por algum aprendizado. O RBC obtém a maioria do seu aprendizado de duas maneiras [LEA96]:

- Pela acumulação de novos casos, e;
- Pela identificação de quais índices utilizar ou não na recuperação.

Quanto mais casos houverem na memória do RBC maior o número de casos que, agrupados por famílias (conjuntos de atributos que definem a similaridade), poderão contribuir para solução de um novo caso. Casos que

tiveram sucesso em recuperações anteriores e casos que não tiveram sucesso dão maior amplitude de cobertura ao problema. Novos índices dão a sintonia necessária para recuperar os casos em situações diversas mas apropriadas [LEA96].

O que foi discutido nos dois últimos parágrafos de forma alguma quer dizer que a generalização não é importante. É claro que sempre que RBC puder encontrar casos indexados da mesma forma e todos sugerindo a mesma solução ou que possam ser classificados da mesma forma, a generalização poderá ser muito útil. Existem programas de RBC que usam redes de discriminação redundante e hierárquica das características compartilhadas (por exemplo, *MEDIATOR* [SIM85] [KOL88], *PERSUADER* [SYK87] [SYK88], *JULIA* [HIN88] [HIN89] [HIN92]), procedendo generalização no momento de adicionar estes casos as suas bibliotecas de casos [KOL93].

Concluindo, RBC pode aprender a partir das suas experiências, o que poderá ser muito útil em situações onde a aprendizagem é fundamental à solução. Aproveitando soluções passadas para resolver novos problemas e antecipando e evitando erros cometidos vai melhorar em muito a qualidade da solução. O aprendizado acontece, na sua maior parte, como um subproduto do raciocínio. Armazenando novas experiências na memória de casos e disponibilizando-as para futuros casos. Informando e analisando falhas, e armazenando-as na memória para evitar novas falhas no futuro. Sem dúvida, a indexação deve ser feita considerando a aprendizagem. Os casos devem ser indexados de forma que facilitem o aproveitamento deste conhecimento.

3. DOMÍNIO

Este capítulo aborda a realidade atual do suporte ao cliente, descreve sua origem e seus principais problemas. Apresenta a empresa, caso deste trabalho, e todos os detalhes importantes para a compreensão da amplitude do seu problema. São listadas pesquisas e casos de utilização de RBC para o problema de suporte telefônico e aproveitamento de bases de dados como bases de casos.

3.1 Suporte ao Cliente

Empresas prestadoras de serviço, assistência técnica e provedores de serviços em geral, têm no aumento da qualidade de seus serviços um dos poucos diferenciais que proporcionam aumento de competitividade no mercado. Muitas empresas estão preocupadas cada vez mais em melhorarem o atendimento ao cliente (DAV93). Para atingirem este objetivo, colocam a disposição dos seus clientes ferramentas que agilizam esta interação. São muitos os procedimentos, com ou sem o suporte de *hardware/software*, que interagem neste processo: suporte telefônico (*help desk*), telefones gratuitos e Internet, por exemplo.

O termo *help desk* surgiu com o aparecimento dos computadores pessoais nos anos 80, quando cresceu a necessidade de suporte aos novos usuários de computadores, entre eles gerentes, técnicos e secretarias. Assim, muitas empresas criaram os Centros de Informação (CI's) para auxiliar no uso dos computadores pessoais na empresa. Os primeiros sistemas usados pelos CI's foram os sistemas gerenciadores de bancos de dados com informações sobre os clientes de *software* e *hardware*. Com o advento dos sistemas especialistas,

importantes funções de auxílio a clientes puderam ser desenvolvidas pelos CI's. Assim, os teóricos de sistemas especialistas chamaram estes sistemas de *Sistemas Especialistas Baseados em Diagnósticos* [KAM93] e o pessoal do CI's chamou o conjunto, sistema mais funções de auxílio de *help desk*.

Inicialmente usados para problemas relacionados com os computadores, os *help desk* atualmente são usados para auxiliar clientes a qualquer tipo de assistência por telefone. Os clientes fazem um contato com um operador de *help desk* e numa operação bem simples os operadores tomam conhecimento do problema e baseados na sua experiência e conhecimento prestam uma informação ou recomendam uma determinada ação para resolver o problema.

Os *help desk* são mais encontrados de duas formas:

- Múltiplos pontos de contato: os clientes podem escolher entre os grupos de *help desk*, especialistas em um determinado produto ou tipo de problema, ver figura 3-1.

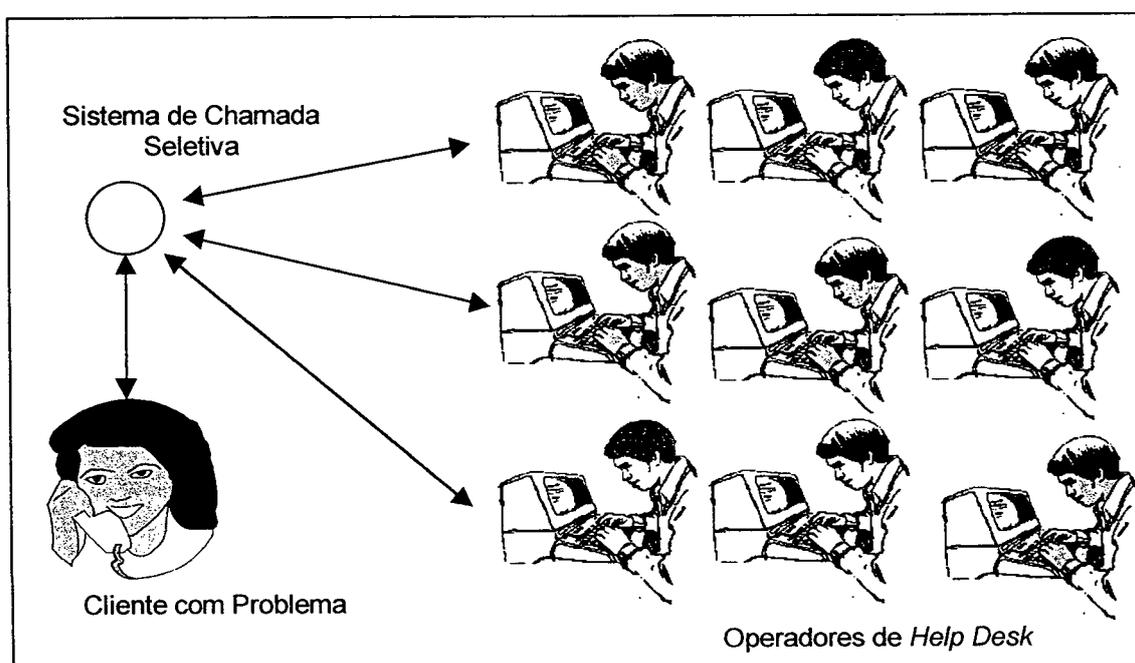


Figura 3-1 – *Help Desk* com múltiplos pontos de contato [WAT97]

- Ponto de contato único: todos os contatos telefônicos dos clientes são primeiramente feitos por um grupo de operadores. Se estes operadores não conseguirem resolver o problema então o contato é passado para um segundo grupo que é composto por especialistas com mais conhecimento e experiência. Este processo é também conhecido por hierarquia de contato [WAT97], ver figura 3-2.

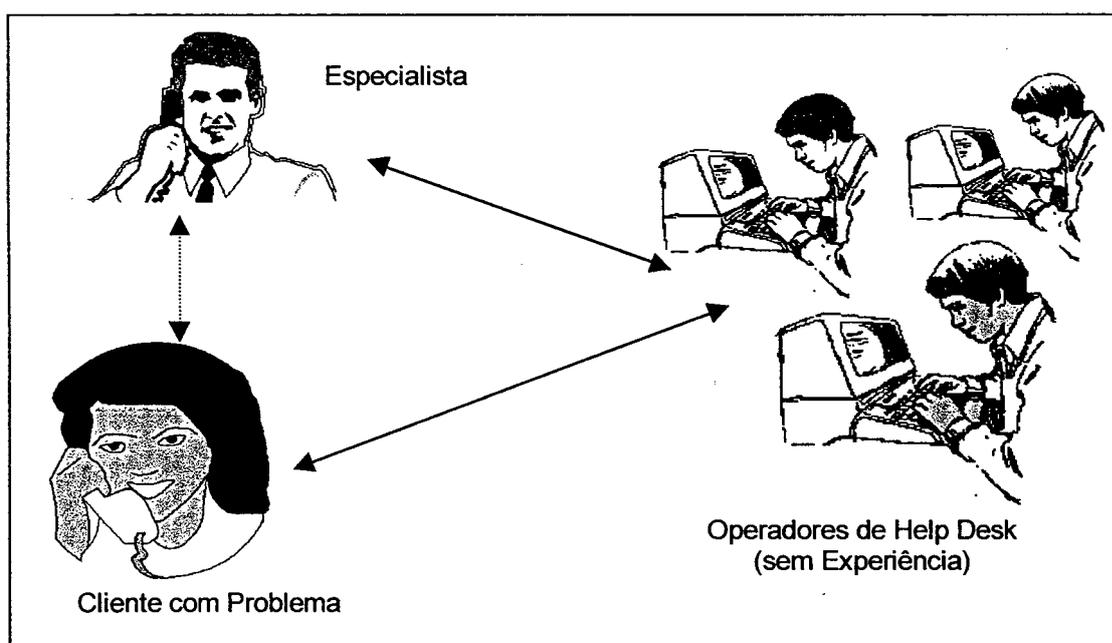


Figura 3-2 – *Help Desk* com único ponto de contato [WAT97]

Entre os tipos citados existe uma preferência, por parte dos clientes, pelo ponto de contato único, visto que eles não gostam que sua chamada seja passada de uma pessoa para outra e outra, o que é característico no tipo múltiplos pontos de contato.

O serviço de suporte telefônico continuou evoluindo juntamente com a tecnologia das telecomunicações e informática. Há empresas que mantêm com seus clientes um *help desk* através de telefone gratuito (também conhecido pelo

termo em inglês *toll free*). A tecnologia agregada neste tipo de serviço pode monitorar o volume, a velocidade e a eficiência dos atendimentos e também conduzir a consulta do cliente através de um sistema eletrônico que está preparado para resolver as questões mais simples e corriqueiras, sem a necessidade de intervenção do especialista. A tecnologia computacional agrega facilidades pela rapidez e disponibilidade de tempo integral. Sempre existe a possibilidade do sistema eletrônico não atender as necessidades do cliente e então este ser atendido pelo especialista.

A Internet também é usada para auxiliar na tarefa de atendimento ao cliente. Através dela são disponibilizadas páginas (*sites*) onde os clientes podem encontrar procedimentos para resolver os problemas mais comuns. É possível também abrir chamados por mensagens usando o correio eletrônico.

3.1.1 O Problema

Por mais facilidades que colocam à disposição dos clientes, as empresas prestadoras de serviços ainda têm problemas que acabam afetando o nível de atendimento ao cliente. Entre estes problemas pode-se citar:

- Alta rotatividade de pessoal especialista;
- Alto custo em manter o suporte telefônico;
- Insatisfação dos clientes com o serviço.

A alta rotatividade do pessoal especialista que trabalha no suporte telefônico, auxiliado ou não por tecnologia computacional, acontece em parte devido ao estresse, característico da função, a repetibilidade e a falta de

criatividade. Estes fatores acabam contribuindo para a insatisfação destes profissionais ocasionando transferência de setor ou o desligamento da empresa.

O alto custo do suporte telefônico está relacionado com o valor do serviço cobrado pelas companhias telefônicas e pelos altos salários pagos aos especialistas que trabalham no setor. Quanto ao custo do serviço de telecomunicação nada pode ser feito. Já os custos com os especialistas são passíveis de redução. Pelo que representam para a empresa, estes profissionais são bem remunerados e podem trazer um retorno muito maior em um setor produtivo da empresa, por exemplo no desenvolvimento de novos produtos. Por pressão dos próprios especialistas e dos altos custos, estes profissionais são transferidos para outros setores e em seu lugar assumem profissionais menos qualificados a um custo menor.

A insatisfação dos clientes é gerada pelo constante ingresso de pessoas novas no setor, com menos experiência e conhecimento, seja pela alta rotatividade do setor ou por ações de redução de custos.

Para as empresas tornarem-se mais competitivas, através do atendimento, é necessário atacar os problemas da alta rotatividade e dos altos custos do suporte telefônico já discutidos. A proposta é automatizar o máximo possível o serviço, objetivando uma redução em torno de 80% no envolvimento do especialista. Isto reduziria o problema do estresse da função, compartilharia o tempo do profissional com outros setores produtivos da empresa e conseqüentemente reduziria o custo do serviço. Por exemplo, das oito horas diárias de trabalho do especialista, seis poderiam ser dedicadas a atividades produtivas e duas a atividades de suporte telefônico.

3.2 O Caso Datasul

Para um melhor entendimento do problema de suporte telefônico, será construída uma solução com acompanhamento dos resultados. Escolheu-se a Datasul, empresa do ramo de *software*, para implementação e aplicação deste trabalho.

3.2.1 A Empresa

A Datasul é uma empresa com matriz em Joinville (SC) e filiais em São Paulo, Rio de Janeiro, Minas Gerais, Rio Grande do Sul, franquias no Paraná, Ceará, Santa Catarina, Campinas e Bahia e subsidiárias nos Estados Unidos, África do Sul e México. A Datasul é pioneira no Brasil no desenvolvimento de um sistema integrado para gestão empresarial em banco de dados e linguagem de quarta geração. É líder no mercado nacional em aplicativos SGBD, e terceira na América Latina, segundo pesquisa do IDC (*Institute Data Corporation*). É a 34ª colocada no Top 50 em sistemas de manufatura, segundo a revista *Manufacturing System*; 41ª empresa de informática do *ranking* nacional, conforme pesquisa 500 Maiores, do jornal Folha de S. Paulo, publicada em junho passado. Segundo o Balanço Anual da Gazeta Mercantil, a Datasul é a segunda colocada no segmento de *software* no Brasil.

Desde 1994, a Datasul vem desenvolvendo o programa da Qualidade Total, coordenado por um conselho que é presidido pelo próprio presidente da empresa, com o objetivo da busca contínua de qualidade dos seus produtos e serviços. Além disto, a Datasul recebeu o certificado ISO 9001, conferido pelo órgão certificador brasileiro, o Inmetro, no mês de dezembro de 1996. Em fevereiro de

1997, recebeu o certificado ISO 9001 TÜVCert, concedido pelo organismo certificador alemão RWTÜV.

3.2.2 Clientes

Possui uma base instalada de mais de 10 mil módulos em mais de mil locais, totalizando cerca de 900 clientes. Seu faturamento em 1997 foi de setenta e quatro milhões de dólares.

3.2.3 Empregados

A Datasul tem cerca de 650 empregados diretos distribuídos entre matriz e filiais. Estima-se que além dos empregados diretos, existam ainda em torno de 1000 empregados indiretos representados por profissionais que trabalham em empresas franqueadas, em revendas e empresas de consultoria.

3.2.4 Produtos

O produto mais importante da Datasul é o Magnus. O Magnus se apresenta como um sistema completo, reunindo recursos capazes de otimizar e integrar todos os processos administrativos e produtivos existentes dentro de qualquer empresa, independentemente do seu porte e da sua estrutura. Ele atende satisfatoriamente a um amplo leque de aplicações, essencialmente no que se refere a Suprimentos, Manufatura, Planejamento, Logística, Expedição, Atendimento a Clientes, Recursos Humanos e aos controles contábil, financeiro e fiscal destes processos.

Outro produto importante da empresa é o Datasul-EMS (*Enterprise Management System* - Sistema de Gerenciamento Empresarial). O EMS é um

software flexível, com características e padrões internacionais, totalmente gráfico escrito em *Progress*¹ 8 orientado a objetos. Conceitualmente, o produto foi definido com a participação de analistas de negócios, clientes, consultores internacionais e pesquisas de mercado. Incorpora, entre outros, recursos técnicos como: Multiplanta, EIS (*Executive Information System*), EDI (*Electronic Data Interchange*) e Bancos Históricos. Possui interfaces com produtos de terceiros, como: editores de texto, planilhas eletrônicas, gerenciadores de correio eletrônico, coletores de dados, modem e fax. Cálculos de FASB (*Financial Accounting Standards Board*), CMCAC (Contabilidade em Moeda de Capacidade Aquisitiva Constante) e Multimoeda . Utiliza, também, novas tecnologias como a filosofia cliente/servidor e conectividade; e introduz novas técnicas como custeio ABC (*Activity Based Cost*) e padrão, controle por unidade de negócio e ERP (*Enterprise Resource Planning*).

3.2.5 Serviços

A DATASUL oferece aos seus clientes uma estrutura de Suporte e Manutenção adequada às suas necessidades, com a utilização de procedimentos e ferramentas padronizadas.

O suporte telefônico é uma das modalidades oferecidas, tem por objetivo registrar, encaminhar e solucionar problemas relacionados a instalação e utilização dos produtos que possam eventualmente ocorrer. A comunicação pode ser feita por fax, modem ou telefone. Através do suporte telefônico são sanadas

¹ Gerenciador de banco de dados e linguagem de 4ª geração. Atualmente na versão 8.2A .

dúvidas. Já por fax ou modem, o usuário pode não apenas registrar dúvidas, mas também encaminhar sugestões e solicitações diversas.

A Datasul disponibilizou recentemente aos seus clientes o suporte via Internet, através do qual o cliente pode abrir uma Ficha de Ocorrência (FO), atualizar dados cadastrais e senhas e ainda consultar versões de programas. Além destas possibilidades, o cliente, ao acessar a *home page* (página na Internet) da Datasul encontra informações sobre o produto através do Informativo Técnico. Outra facilidade à disposição é a busca de programas pela Internet

O Treinamento ao cliente é oferecido através de uma programação de cursos que são ministrados em várias cidades. Há, também, a possibilidade de serem realizados cursos fechados, no próprio cliente.

A Datasul oferece ainda assistência e consultoria aos seus clientes com o objetivo de auxiliá-los na correta implantação dos produtos e fornecer orientações padronizadas em eventuais casos de alteração de procedimentos internos para que haja uma perfeita harmonia entre a utilização dos produtos e as diretrizes da empresa usuária.

3.3 O Problema de Suporte da Datasul

3.3.1 O Suporte

A Datasul entende por suporte todo o envolvimento necessário para suportar e manter as funções existentes no produto, permitindo aos usuários o seu pleno uso. Atualmente, os clientes da Datasul têm as seguintes formas de acesso ao serviço de Suporte da Datasul:

- **Ficha de Ocorrência:** através de fax, correio eletrônico ou digitação na *homepage* do suporte e utilizando o formulário adequado (FO), as dúvidas, problemas e sugestões podem ser reportados e serão encaminhados à área responsável. Todos os registros referentes a uma FO são feitos no Sistema de Controle de Fichas de Ocorrência da Datasul, permitindo acompanhamento mais preciso e consulta aos históricos de qualquer situação reportada. Após uma análise, a FO pode percorrer os seguintes fluxos:
 - Se for uma dúvida, será enviada resposta através de fax ou correio eletrônico, com todos os esclarecimentos que forem necessários.
 - Se for um problema (erro), a FO será encaminhada à área de Manutenção de Sistemas, onde serão feitas as devidas correções.
 - Se for um problema que já tenha sido anteriormente detectado e resolvido, será solicitado ao cliente que atualize a versão do programa e, se for necessário, será também disponibilizado um programa especial para acerto da base de dados.
 - Se for uma solicitação de implementação de novas características ou funções, será encaminhada para ser analisada pela Comissão Técnica, que fará a aprovação para desenvolvimento, ou fará a sugestão de outras alternativas.
- **Suporte Telefônico:** Utilizado para esclarecer pequenas dúvidas que possam ser rapidamente respondidas. Todo contato mantido com o Suporte Telefônico inclui um registro no banco de dados de FO's. Atualmente são registrados em torno de 10.000 contatos/mês. Caso o especialista consiga resolver o problema o processo está encerrado e as informações armazenadas a respeito do

contato serão mantidas apenas para efeitos estatísticos. Porém, caso o especialista não consiga resolver o problema ele solicita ao cliente que encaminhe uma FO a Datasul. Mensalmente são cadastradas em torno de 4.000 FO's. A Datasul dispõe de 90 troncos telefônicos, onde 10 ramais atendem o suporte telefônico do Magnus.

Em resumo, um *software* grande e complexo, 900 clientes e inúmeros empregados atendendo a clientes e revezando-se no Suporte Telefônico. O processo exige que no suporte telefônico estejam especialistas com profundo conhecimento de negócio e do *software*. O processo exige um pronto atendimento ao cliente. Registram-se reclamações de clientes que não conseguem ter acesso ao suporte telefônico, pois os ramais estão sempre ocupados. Observa-se que os melhores técnicos não estão em atividades produtivas, como por exemplo o desenvolvimento de novos produtos.

3.3.2 A Implementação

É fundamental para a Datasul que uma proposta de solução gere o menor custo possível, aproveitando a base de dados de FO's já cadastradas e toda a tecnologia (sistemas operacionais, linguagens de programação e gerenciador de banco de dados) já disponível na empresa. Isto implica em considerar o Progress como gerenciador de banco de dados e linguagem de programação para a implementação da solução.

A solução proposta é a implantação de um sistema computacional com inferência inteligente, para automatizar o atendimento dos clientes no suporte telefônico. A escolha deste sistema fundamenta-se no fato de que a partir da

descrição do problema pelo cliente, é possível identificar problemas similares já resolvidos na base de FO's existente.

3.4 RBC e Suporte Telefônico

Existem pesquisas e experiências a partir de 1991 usando RBC para resolver problemas de suporte telefônico. O objetivo desta seção é resumidamente apresentar as mais importantes.

Em 1991 a Compaq, fabricante mundial de computadores, em colaboração com a Inference Corp., desenvolveu um sistema de gerenciamento de suporte ao cliente usando RBC, ao qual deram o nome SMART [ACR92]. O Smart iria se integrar a uma rotina de atendimento ao cliente já existente. Nesta nova proposta o engenheiro de suporte da Compaq solicitaria ao cliente as informações cadastrais (nome, número de contrato etc) e uma pequena descrição do problema. Esta descrição seria submetida ao SMART que procuraria inicialmente na base de casos aqueles mais similares ao problema em questão. O SMART devolveria ao engenheiro de suporte uma lista de casos com algumas perguntas associadas. O engenheiro usaria estas perguntas para solicitar mais detalhes ao cliente. Após, estes dados seriam alimentados no SMART e uma nova execução seria feita. Então, uma lista mais elaborada dos casos relevantes seria apresentada ao engenheiro que verificaria se uma delas resolveria o problema do cliente. Como resultado, a Compaq verificou que 87% dos casos eram resolvidos pelo SMART e em média com 2 minutos de processamento. O SMART recebeu o prêmio "AAAI (Innovative Application in Artificial Intelligence Award)" em 1992.

No projeto CASCADE da Digital em 1992, foi usada recuperação baseada em casos no suporte técnico aos clientes que tinham problemas com o sistema operacional VMS [SIM90]. O CASCADE sugeria aos engenheiros da Digital, fabricante mundial de computadores, soluções a problemas relatados pelos clientes usando raciocínio baseado em casos e recuperação validada. A **recuperação validada** baseia-se na recuperação descendente de todos os casos relevantes baseada nas características iniciais e na recuperação ascendente onde modelos tratam somente os casos recuperados obtendo características mais discriminantes do problema. O resultado foi que sessenta por cento dos problemas submetidos foram resolvidos com sucesso a um custo estimado de vinte mil dólares.

Em, 1993 a L&G, maior instituição financeira do Reino Unido, resolveu implantar um serviço de atendimento a clientes internos para padronizar a plataforma de computadores da empresa [WAT97]. A partir das necessidades especificadas pelos funcionários da empresa a respeito do tipo de serviço onde estaria sendo necessária a aquisição de um computador, o sistema deveria sugerir a configuração ideal do equipamento, considerando regras de padronização da plataforma, agilizando a aquisição e, conseqüentemente, o recebimento. A solução foi o desenvolvimento de um sistema de RBC chamado SWIFT. O SWIFT foi desenvolvido através da integração entre *CBR Express* com *CasePoint*, *Asymetrix ToolBook* e *Microsoft Access*. O SWIFT integrava-se com a base de dados corporativa da L&G. O desenvolvimento foi rápido e usou uma metodologia chamada *Client-Centered Approach (CCA)*. Com a utilização do SWIFT a média

de tempo necessária para receber o equipamento passou de 14 para 5 dias, o que foi considerado um ótimo resultado.

Em 1995 a *Broderbund Software*, maior fornecedor americano de *softwares* educacionais, implementou o sistema de RBC *Gizmo Tapper* [WAT97]. O sistema tinha 3.000 casos registrados, foi desenvolvido com a ferramenta *CasePoint WebServer* integrante do CBR3 da *Inference Corp*. O objetivo principal era resolver, via Internet, o grande número de chamadas que havia após o natal de cada ano, onde as crianças, principalmente, tinham dúvidas sobre o funcionamento dos jogos e *softwares* educacionais que haviam ganho. Após o natal de 1995 foi constatado que o sistema conseguiu atender a 2,5 vezes mais chamadas que o suporte ao cliente no ano anterior

Após uma análise do serviço ao cliente, foram identificadas falhas no uso dos dados mantidos em bases de dados [LEE96]. Foi sugerido o uso destas bases de dados associadas a RBC para melhorar o serviço ao cliente. A proposta passava pela centralização das chamadas de clientes, integração da base de dados, um sistema de acompanhamento de serviços da assistência técnica, desenvolvimento de um sistema de diagnóstico em computador, construção de um sistema de suporte ao *help desk*, uso de *notebooks* pelo pessoal da assistência técnica, introdução de um sistema de informações logísticas e um ambiente de informática mais flexível. No desenvolvimento do sistema de diagnóstico de mau funcionamento dos equipamentos baseado em casos (CBEMD), foram usados comandos SQL² para recuperar os casos. A recuperação foi feita pela

² Linguagem estruturada para acesso a Bancos de Dados

comparação direta dos atributos **código do equipamento**, **código da máquina** e **sintoma**. Após recuperados os casos, foi calculada a similaridade considerando os atributos dos casos e um peso para cada coincidência encontrada neles. O CBEMD conseguiu sucesso em trinta por cento dos problemas, considerando a primeira sugestão. Considerando a segunda e terceira sugestões o sucesso foi de quarenta e seis por cento. Este resultado foi considerado muito bom, visto que no sistema manual a assistência técnica conseguia somente seis por cento. O custo total de desenvolvimento e implantação foi de trezentos e sessenta e cinco mil dólares.

As aplicações de *help desk* e suporte ao cliente são as áreas de maior sucesso comercial de sistemas RBC [WAT97]. Entre as razões para este sucesso está o fato do RBC ser ideal para aplicações de relativa superficialidade e com grande abrangência. Outra característica importante é o fato do RBC também disponibilizar uma metodologia para captura de novos problemas/soluções de forma simples. Muitas das bases de casos usadas em aplicações help-desk e de suporte ao cliente são mantidas pelos próprios usuários finais, sem a necessidade de engenheiros de conhecimento.

3.5 Base de Dados X Base de Casos

Dados ou informações são a base para as instituições tomarem decisões, desde o nível estratégico até o operacional [LEE96].

Atualmente os dados são armazenados pelas instituições em bases de dados corporativos ou não. Corporativos são as grandes bases de dados que contém informações que são usadas por todos os departamentos. Bases de dados

não corporativas ou proprietárias são aquelas que contêm somente dados a respeito do seu negócio específico, não integrando com outras bases.

Base de Dados é o depósito de um conjunto de arquivos de dados relacionados com o objetivo de proporcionar ao usuário facilidades de manuseio, controle da redundância, consistência, integridade, segurança, compartilhamento e padronização [DAT91]. O significado dos dados está contido no relacionamento entre os itens de dados. Veja na figura 3-3 que os dados apresentados sem as relações com os itens de dados, nada significam.

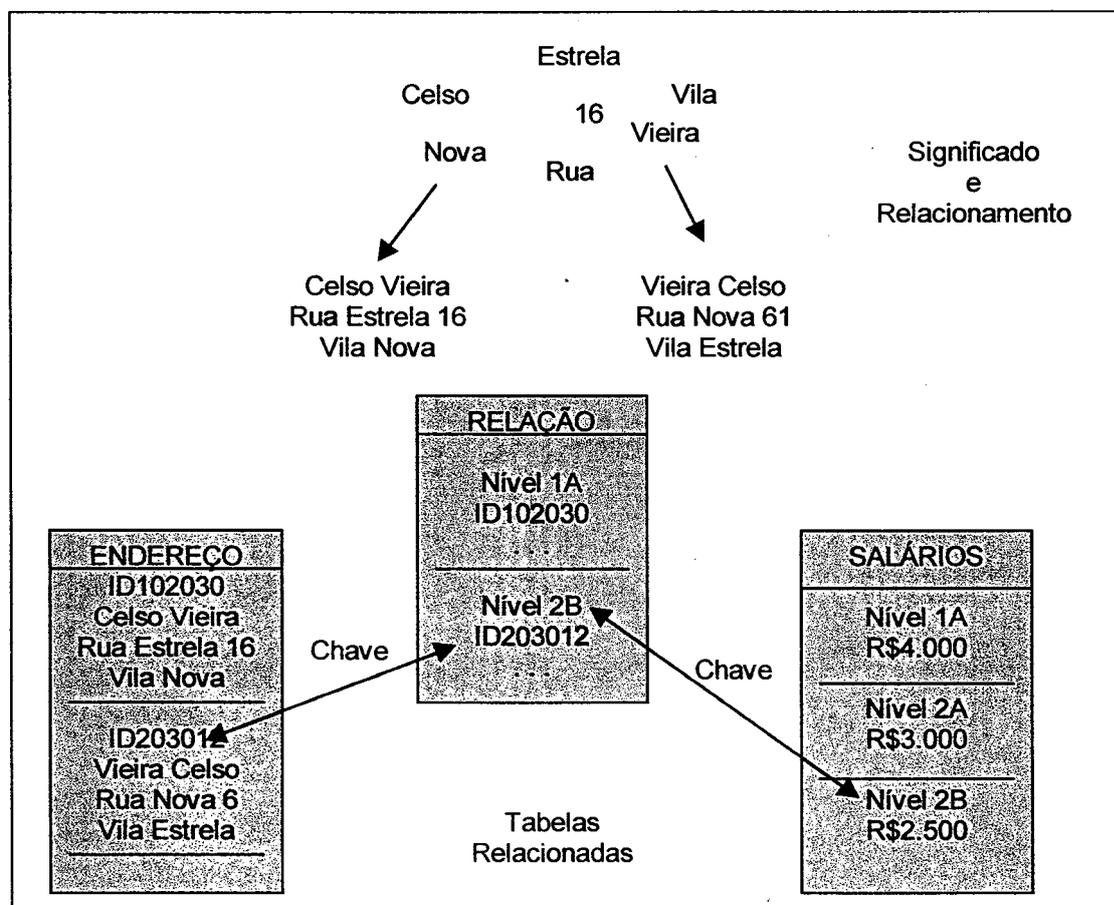


Figura 3-3 Relacionamentos e Tabelas [WAT97]

Com a utilização das bases de dados, reduz-se parcialmente o espaço necessário para armazenar os dados e facilita-se o processo de manutenção, visto que os itens de dados são únicos. Os relacionamentos entre os itens são definidos pelas tabelas relacionadas e identificados por códigos chamados “chaves” (figura 3-3).

Uma outra forma de representar os dados em base de dados é a técnica de orientação a objeto. Em uma base de dados orientada a objetos, os registros individuais (por exemplo, Celso Vieira, Vieira Celso) são armazenados como instâncias de uma classe chamada funcionário. Nesta classe está definido tudo que é comum a todos os empregados. A classe empregado pode ser subdividida em subclasses (por exemplo, funcionários horistas, mensalistas). Um processo chamado herança atribui as instâncias os itens de dados definidos por suas classes e super classes. A classificação do mundo real em hierarquia de objetos (classes) com o uso de propriedades como a herança implementam grande poder de representação as bases de dados em relação aos bancos de dados relacionais.

Base de casos é a estrutura usada para armazenar os casos em um sistema RBC. Como visto no capítulo 2, o caso é composto de duas partes principais:

- a lição ou lições que ele ensina, e;
- o contexto no qual esta lição pode ser ensinada (os índices).

Logo, a estrutura de uma base de casos é que suporta o registro destas informações, possibilitando que sejam mais tarde correta e rapidamente recuperadas. O *CBR3 (CBR Express, CasePoint, Generator, Tester e CasePoint WebServer)* produzido pela Inference Corp., produto de RBC para *help desk* com

maior sucesso, 500.000 usuários finais em 22 países, organiza sua estrutura de caso em bases de dados relacional (*Raima por default*, viabilizando também *Microsoft SQL Server, Sybase SQL Server, Oracle e Informix*).

A utilização de bases de dados relacional, para organização dos casos, proporciona maior flexibilidade para realizar buscas, compartilhar e gerenciar dados [LEE96].

Concluindo, as bases de dados sozinhas não conseguem resolver adequadamente todos os problemas do mundo real, principalmente quando há a necessidade de representar conhecimento e experiência [LEE96]. As bases de casos, por sua vez, podem usufruir das vantagens de flexibilidade de pesquisa, compartilhamento e gerenciamento de dados usando as bases de dados (relacional ou objeto) como forma de representar o conhecimento e a experiência contidos nos casos. Outro aspecto da maior relevância é o fato de que hoje ter-se muitos dados armazenados em bases de dados que podem vir a ser aproveitados para criar os casos, como neste trabalho.

4. IMPLEMENTAÇÃO

4.1 Introdução

O Sistema de Help Desk (SHD) busca melhorar a prestação de serviço de suporte ao cliente da Datasul, que é um problema similar ao de muitas empresas de desenvolvimento de *software* que também prestam assistência a seus clientes. A solução adotada usa a tecnologia de RBC e através da transformação de um banco de dados de problemas já existente em base de casos de RBC, visa dar ao usuário a possibilidade de realizar buscas e encontrar problemas muito similares aos seus, já resolvidos.

Uma das alternativas buscadas foi procurar no mercado uma *shell* (*software* para desenvolvimento) de RBC que pudesse atender aos requisitos do problema. Entre estas foram analisadas: INRECA da Technno versão 1.0, ESTEEM da Esteem *Software* na versão 1.4 e CBR3 EXPRESS da Inference Corp. Nenhuma oferece interface com banco de dados Progress o que elimina a possibilidade de aproveitá-las para mapear os dados.

Apesar da ausência de interface para banco de dados Progress, as *shell's* foram testadas e avaliadas. Entre elas, a *shell* CBR3 EXPRESS da Inference Corp. foi aquela que apresentou os melhores resultados, principalmente na facilidade de migração dos dados do Progress para o banco nativo da ferramenta, pela boa performance com grande volume de dados e transações e pela sua modularidade.

A decisão foi então pelo desenvolvimento do SHD baseado na fundamentação teórica do capítulo 2 IA e RBC, e também nas características do domínio da aplicação, conforme abordado no capítulo 3.

O presente capítulo apresenta a definição do escopo abrangido pelo presente sistema e as etapas da implementação, descrevendo os problemas encontrados e as soluções adotadas.

4.2 Definição do Sistema

O sistema desenvolvido aproveita o banco de dados de FO existente (capítulo 3 seção 3) transformando-o em uma base de casos. Na sua função de recuperação, o sistema faz inferências sobre os problemas armazenados para tentar resolver novos problemas de clientes e dos próprios especialistas de suporte da Datasul, também denominados Operadores de Help Desk (OHD).

O sistema abrange todas as funções necessárias para possibilitar ao usuário (cliente ou OHD) facilmente informar seu problema atual e obter como resposta os problemas já resolvidos mais similares encontrados na base de casos. A utilização do SHD não altera o processo de funcionamento de suporte ao cliente da Datasul, conforme visto no capítulo 3. O SHD somente agrega um novo procedimento opcional ao cliente ou ao OHD. O cliente antes de comunicar-se com o suporte da Datasul pode consultar o SHD via Internet acessando a página da Datasul. Os OHD's antes de repassarem a chamada para o nível hierárquico superior na estrutura de *help desk* podem tentar resolver o problema usando o SHD numa rede local.

4.3 Representação e Indexação dos Casos

Os atributos valorados são usados para representarem os casos. Para representar os casos é necessário conhecimento sobre o conteúdo e o contexto das experiências que são representadas nestes casos. O conhecimento do SHD é originário do banco de dados FO's, justificando a ausência da fase de aquisição do conhecimento. Os atributos valorados podem assumir diferentes funções, tais como descrever o caso, indexar o caso para orientar a recuperação, descrever soluções e armazenar informações sobre o resultado do uso do caso [KOL93].

4.3.1 O Banco de Dados de FO's

O conhecimento necessário para representação dos casos origina-se da transformação do banco de dados de FO numa base de casos. Parte-se do princípio que as experiências descritas nas FO's, que contêm o conhecimento, possam ser reutilizadas para resolver problemas atuais. A transformação do banco de dados de FO's em casos iniciou-se pela avaliação do volume de dados contido no banco de dados de FO's. A Datasul mantém em seu banco de FO's todos os registros de chamadas feitos ao suporte nos últimos 3 anos. A cada início de ano, os registros do terceiro ano mais antigo são eliminados. Considerando que semanalmente são inseridos no banco de dados de FO's aproximadamente 2.500 novos registros, isto corresponde a 10.000/mês, 120.000/ano e ao final do terceiro ano 360.000 registros.

Analisando detalhadamente o banco de dados de FO's foi constatado que nem todos os registros deveriam ser transformados em casos na base de casos. Verificou-se que 90% dos problemas são relativos a implementações no *software*

nos últimos 2 ou 3 meses, e apenas 10 % dos problemas repetem-se nos clientes com defasagem superior a 3 meses. Considerando este fato, resolveu-se que somente seriam transformados em casos os registros dos últimos 12 meses. Outra constatação foi que 60% dos registros são simples chamadas ao suporte, que somente são armazenadas para consultas estatísticas. Os restantes 40% dividem-se por tipos de FO, ver tabela 4-1 e por produto, ver tabela 4-2. Somente serão casos as FO's já solucionadas. Os produtos a serem considerados serão aqueles diretamente relacionados com os clientes externos, priorizados pela empresa, e limitados ao idioma português, já que para a avaliação da similaridade é fundamental a identificação do idioma.

Tabela 4-1: Tipos das FO's

Situação da FO	Percentual	Resolvida?
Digitada	1	Não
Transferida	1	Não
Lista de Espera	1	Não
Em Processo	3	Não
Aguardando Retorno	2	Não
Retorno do Cliente	1	Não
Reaberta	2	Não
Aguardando Análise	3	Não
Aprovado para desenvolvimento	1	Não
Aprovado para futura versão	6	Não
Inconsistente	1	Não
Aguardando Comissão Técnica	2	Não
Consistida	1	Não
Envio de Fax/E-mail	1	Não
Repriorizada	1	Não
Fax/E-mail – FO Baixada	6	Sim
Comentários para Histórico	2	Não
Cancelada	2	Não
Solucionada	63	Sim

Após esta primeira etapa, o banco de dados de FO's a ser transformado em base de casos conta com 9.872 registros. A próxima etapa é determinar a

estrutura organizacional da base de dados de casos e a forma de representação destes.

Tabela 4-2: Registros por produto no Banco de Dados de FO's

Produto	Descrição do Produto	Percentual	Considera?
CIG00	Sistemas Internos	4	Não
EMB00	Magnus Argentina	7	Não
EMH00	Magnus Versão Internacional	19	Não
FOG00	Sistema de Ficha de Ocorrência	1	Não
MF00	Magnus Versão F.00	2	Sim
MF50	Magnus Versão F+	0	Sim
MG00	Magnus Versão G.00	12	Sim
MG97	Magnus 97 (em desenvolvimento)	9	Não
MI00	Magnus Versão I.00	8	Sim
MI01	Magnus Versão I.01	0	Sim
MJ00	Magnus Versão J.00	0	Sim
MKA00	Marketing	1	Não
MN00	Magnus Grade N.00	1	Sim
PROGR	Progress	19	Não
RHC00	Folha de Pagamento C.00	1	Sim
RHD00	Folha de Pagamento D.00	4	Sim
RHI00	Magnus-RH I.00	1	Sim
RTB	Roundtable	3	Não
SIA00	Sistema Interno Administrativo A.00	8	Não

4.3.2 A Base de Casos

A estrutura do banco de dados de FO, além dos dados contidos no documento "Ficha de Ocorrência", preenchido pelo cliente da Datasul (apêndice A) contém também dados para estatísticas e acompanhamento de problemas. Para este trabalho somente são importantes os dados dos problemas informados pelos clientes. Na tabela 4-3 estão relacionados os dados do documento "Ficha de Ocorrência" e seu significado. Para transformar as experiências descritas no banco de dados de FO's em casos é necessário identificar quais atributos destas experiências, comparados com outros de um novo problema, podem de alguma forma auxiliar na medição da similaridade. Os valores contidos nos atributos

determinam uma medida de similaridade entre os casos e conduzem a solução do problema em questão.

Tabela 4-3: Estrutura do banco de dados de FO's

Atributo	Significado
Nº	Número da FO
CLIENTE	Nome abreviado do cliente
TIPO FO	Código do Tipo de FO
DATA DE EMISSÃO	Data de emissão da FO
PRODUTO	Produto onde ocorreu o problema
MÓDULO	Módulo do Produto onde ocorreu o problema
CONTRATO	Código do contrato do cliente com a Datasul
CONTATO	Nome da pessoa de contato no cliente
VER. PROGRESS	Versão da linguagem de programação onde aconteceu o problema
PROGRAMA	Código do programa no módulo/produto onde aconteceu o problema
VERSÃO	Versão do programa
MÍDIA	Mídia na Qual será enviada a solução do problema, caso haja a necessidade de enviar versões de programas.
SIS. OP.	Sistema operacional do cliente onde aconteceu o problema
OCORRÊNCIA	Descrição detalhada do problema ocorrido
SOLUÇÃO	Descrição da solução do problema

4.3.3 A Transformação dos Casos

Os atributos que representam os casos estão na tabela 4-4. Entre estes atributos temos os atributos que apresentam características de índice, seção 2.2.2. Atributos com características de índice são aqueles que podem vir a ser usados na avaliação da similaridade.

Tabela 4-4: Atributos da base de casos

Atributo	Significado	Características de Índice
Produto	Produto onde ocorreu o problema	Sim
Módulo	Módulo onde ocorreu o problema	Sim
Programa	Programa onde ocorreu o problema	Sim
Problema Detalhado	Descrição detalhada do problema ocorrido	Não
Problema Resumido	Descrição do problema somente com as palavras técnicas do Problema Detalhado	Sim
Solução do Problema	Descrição da solução do problema	Não

Os casos são obtidos do banco de dados de FO's por um programa de importação. Com exceção do atributo **problema resumido**, todos os demais são

copiados do banco de dados de FO's para a base de casos. Este processo de importação deve acontecer mensalmente.

O atributo **problema resumido** foi obtido a partir do **problema detalhado**, sendo um subconjunto deste com a única função de agilizar o processo de busca durante a recuperação. Nem todo o texto é significativo para a solução do problema, para a frase somente são relevantes as figuras de linguagem verbo e substantivo [TEA91]. Para compor o atributo **problema resumido** definiu-se que somente seriam selecionadas as palavras do atributo **problema detalhado** que existissem no **dicionário de palavras técnicas**.

O dicionário de palavras técnicas consiste de uma relação de palavras comuns ao domínio de *software* e do negócio no qual o *software* se aplica (e. g., *software* Magnus – Negócio: Contabilidade). A forma de determinação destas palavras foi feita obedecendo três critérios:

- Todas as palavras do manual de documentação do produto que tivessem mais de dois caracteres;
- Todas as siglas de Produto, Módulo e Programa;
- A partir do atributo **problema** da base de dados de FO's construiu-se um algoritmo que conta cada ocorrência de todas as palavras com mais de dois caracteres. No final elabora uma curva de Pareto [FOG92] , onde 20% das palavras que correspondem a 80% do total de ocorrências são armazenadas no dicionário de palavras técnicas.

A definição de quais atributos do banco de dados de FO's seriam descritores de caso, seção 2.1.3, deve-se ou a sua importância no contexto: todo e qualquer problema acontece quando o cliente está executando algum programa

de algum módulo de um determinado produto; ou pela compreensão de que sua presença pode melhorar significativamente o resultado da recuperação. Na tabela 4-5 é listada a atual estrutura da base de casos do SHD.

Tabela 4-5: Estrutura de Dados do SHD

Tabela/Campo	Formato	Descrição
Produto		Tabela de produtos
- Cd_prod	X(05)	Código do produto
- Desc_prod	X(50)	Descrição do produto
Módulo		Tabela de módulos dos produtos
- Cd_prod	X(05)	Código do produto
- Cd_mod	X(03)	Código do módulo
- Desc_mod	X(50)	Descrição do módulo
Programa		Tabela de programas dos módulos dos produtos
- Cd_prod	X(05)	Código do produto
- Cd_mod	X(03)	Código do módulo
- Cd_prog	X(06)	Código do programa
- Desc_prog	X(50)	Descrição do programa
Palavra		Tabela Dicionário de Palavras Técnicas
- Nome	X(15)	Palavra técnica
- Qtd	9999999	Quantidade de ocorrências na curva de Pareto
- Val_fon	X(30)	Representação fonética da palavra
Sinônimo		Tabela de sinônimos
- Palavra	X(15)	Palavra
- Sinônimo	X(15)	Sinônimo de palavra
- Pai	X(15)	Palavra considerada principal na lista de sinônimos
Problema		Tabela de casos
- Cd_prod	X(05)	Código do produto
- Cd_mod	X(03)	Código do módulo
- Cd_prog	X(06)	Código do programa
- Desc_prob	X(800)	Descrição do caso
- Desc_prob_o	X(800)	Descrição otimizada do caso
- Desc_solu	X(800)	Descrição da solução do caso
- Qtd_pal	9999999	Quantidade de palavras técnicas da descrição otimizada

As tabelas de **produto**, **módulo** e **programa** foram criadas para facilitar a interface usuário-SHD, permitindo que no momento da descrição de um problema estes atributos sejam informados através de um lista de seus possíveis valores. A tabela **palavra** foi criada para armazenar o dicionário de palavras técnicas. A tabela **sinônimo**, foi criada para armazenar as palavras técnicas sinônimas detalhada na seção 4.3.5. E, a tabela **problema** (a base de casos propriamente)

que armazena os casos. Na figura 4-2 utiliza-se os conceitos da análise estruturada de sistemas [MAR91] para apresentar o modelo entidade relacionamento do SHD.

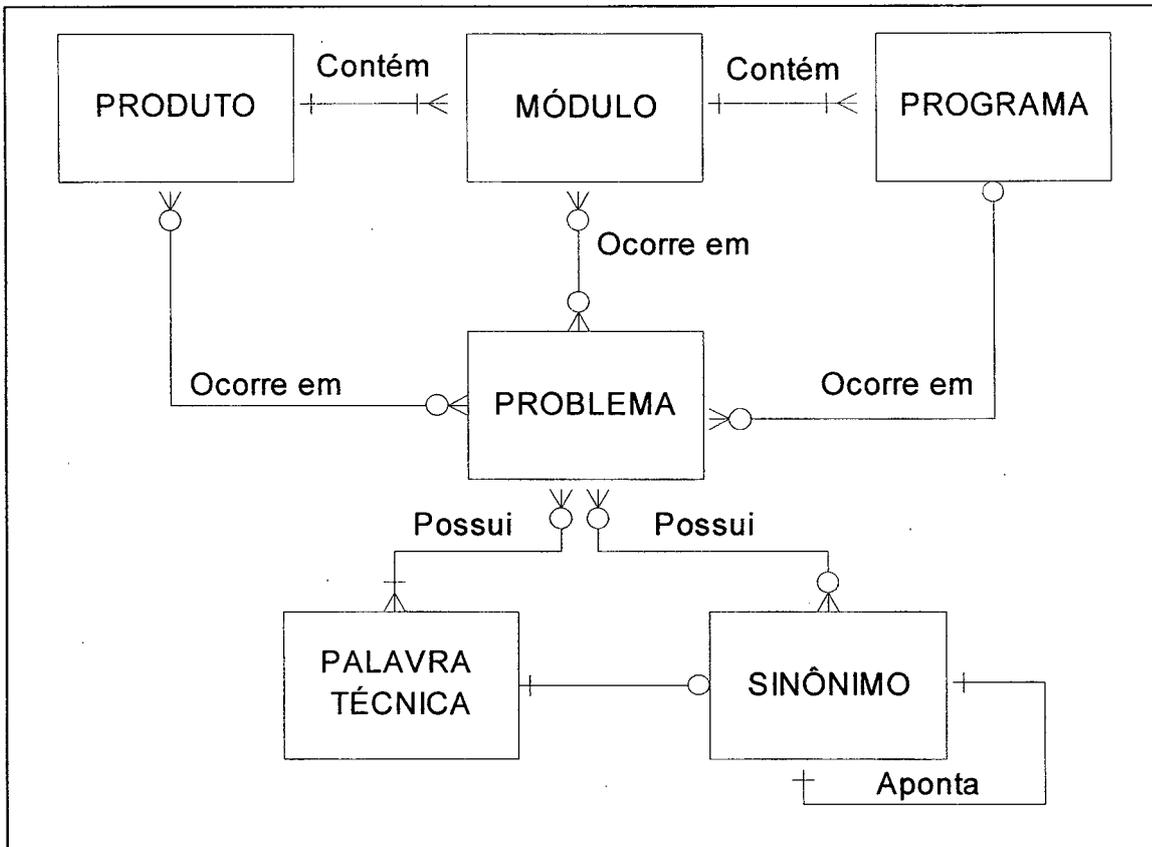


Figura 4-2 – Modelo Entidade Relacionamento (MER) do SHD

4.3.4 A Definição dos Índices

Após termos definido o conteúdo e o contexto dos casos, seção 4.3.3, definimos os atributos que melhor representam as experiências descritas nas fichas de ocorrência (FO's) da Datasul a serem recuperadas. A definição dos índices é um etapa importante da representação do caso.

Os índices são: **programa**, **módulo**, **produto** e **problema resumido** (tabela 4-4). É possível representar a função de comparação dos índices por:

$f(a_i, c_j) = [0, 1]$ onde: f – função de similaridade
 a_i – atributo do caso de entrada
 c_j – atributo do caso candidato

Índice produto: Certos problemas são comuns a determinados produtos, por exemplo, um problema relativo a interface gráfica somente se verifica em produtos gráficos. Portanto a comparação deve ser feita entre produtos iguais ou com alguma característica em comum. Outro fator importante é a facilidade, que tem o cliente, de obter a informação sobre o produto onde verificou-se o problema. Apesar da Datasul comercializar atualmente vários produtos (ver tabela 4-2 na seção 4.3.2), o cliente possui um único produto da Datasul. Ressalvas devem ser feitas aos casos de atualizações de versão de produto, quanto temporariamente o cliente fica com dois produtos simultaneamente durante um curto período de tempo. Portanto, para o cliente é fácil informar o valor do atributo **produto**. Na avaliação da similaridade, determinados problemas são comuns a determinados produtos e considerando o volume de casos que estão distribuídos entre vários produtos o atributo **produto** possui características de índice. Produtos onde $f(a_i, c_j) = 0$, podem ainda serem comparados com um conjunto de regras (seção 4.3.5).

Índice módulo: as mesmas razões que levaram à definição do índice produto se aplicam em parte ao índice módulo. Diferenciam-se apenas pelo fato do cliente sempre possuir n módulos de um determinado produto. Pode-se afirmar também que na maioria dos problemas é fácil de identificar o atributo **módulo** quando verifica-se o problema (e.g., Módulo de Contabilidade, Módulo de Custos). E também, certos problemas são comuns a determinados módulos, por exemplo, o problema “Como proceder para cadastrar título” pode tanto se referir ao módulo

de contas a pagar quanto ao módulo contas a receber. Módulos onde $f(a_i, c_i) = 0$, podem ainda serem comparados com um conjunto de regras (seção 4.3.5).

Índice programa: uma importante característica dos problemas, que justifica a definição do índice programa, é que eles são comuns as mesmas funções (i, e., programas), por exemplo, o problema “Nota Fiscal Impressa com Data Errada” é específico dos problemas existentes no programa que executa a função de imprimir notas fiscais. Na maioria das vezes, o atributo **programa** é facilmente identificado pelo cliente quando verifica-se algum problema.

Índice problema resumido: está baseado na técnica de processamento de texto [LEA96], que permite comparar palavra por palavra. Através da comparação das palavras do atributo **problema resumido** do caso corrente com o mesmo atributo dos casos da base, pode-se encontrar boas respostas. Por exemplo, qualquer problema que contenha a palavra “imprimir” dever ser avaliado para encontrar resposta ao problema “Impossível imprimir envelope de pagamento”.

4.3.5 Avaliação da Similaridade e a Recuperação

A principal atividade desta etapa é a comparação e a medição da similaridade entre o caso de entrada e os casos da base de casos. Para comparar e medir deve-se determinar o que comparar e como determinar a importância de cada comparação. No SHD, o que comparar e o que medir é definido por índices, funções e regras.

A medida de similaridade foi obtida por média ponderada. Veja nas equações: (1) como é calculada a similaridade para os atributos **produto**, **módulo** e **programa**; (2) como é calculada a similaridade do atributo **problema resumido**;

e, (3) como é calculada a medida de similaridade total de cada um dos casos da base de casos.

$$Sgud_k = (SVg_k * Pg) + (SVu_k * Pu) + (SVd_k * Pd) \quad (1)$$

$$Sr_k = \sum_{i \in A} (L_i r_k / m * Pr) \quad (2)$$

$$S_k = Sgud_k + Sr_k \quad (3)$$

Onde:

S_k : Similaridade do Caso

$Sgud_k$: Similaridade dos atributos programa, módulo e produto

Sr_k : Similaridade do atributo problema resumido

k : Caso

g : Atributo Programa

u : Atributo Módulo

d : Atributo Produto

r : Atributo Problema Resumido

SV : Valor da Similaridade

P : Peso

A Conjunto das palavras recuperadas

L_i Palavra i do atributo problema resumido

n : Número de palavras do atributo problema resumido do caso de entrada

n' : Número de palavras do atributo problema resumido de cada um dos casos recuperados da memória

m : $\text{Max}(n, n')$

O peso dos atributos **produto**, **módulo** e **programa** é 10 para cada um deles. O peso do atributo **problema resumido** é 70. A determinação dos pesos inicialmente foi aleatória e após várias simulações com validações dos casos recuperados pelos especialistas da Datasul chegou-se a estes valores.

A equação (1), avalia a similaridade dos atributos programa, módulo e produto. Para medir a similaridade de um caso candidato da base de casos em

relação ao caso de entrada, o SHD compara o atributo programa do caso candidato com o mesmo atributo do caso de entrada. Existindo a igualdade, a medida de similaridade para o atributo programa daquele caso candidato é 1, considerando o peso do atributo. Se os atributos tiverem valores diferentes então a medida de similaridade para aquele caso é nula (0). Da mesma forma, é feita a comparação para os atributos módulo e produto acumulativamente. A diferença entre estes atributos é que para **módulo** e **produto** quando a comparação falhar (<>), o SHD recorre a um conjunto de 21 regras, ver apêndice B, que serão usadas para determinar o valor da medida de similaridade (apêndice C). As regras foram definidas para que fossem considerados, na comparação e na avaliação da similaridade, produtos e módulos que possuam alguma semelhança, na natureza do negócio que atendem, (e.g., Contas a Pagar e Contas a Receber) e na construção do próprio produto e módulo (e.g. produtos com interface gráfica). Através de vários testes com recuperação validada pelos especialistas da Datasul definiu-se os valores para cada relação de módulos e produtos, ver apêndice C tabela-1.

A equação (2) avalia a similaridade do atributo **problema resumido** do caso candidato com o mesmo atributo do caso de entrada. Para medir a similaridade o SHD compara cada uma das palavras do atributo **problema resumido** do caso candidato com o conjunto de palavras do mesmo atributo do caso de entrada. Existindo a igualdade, a medida de similaridade para o atributo é ponderada na razão de 1 para o número máximo de palavras entre o caso candidato e o caso de entrada, considerando o peso do atributo. Para cada uma

das palavras do atributo **problema resumido** do caso candidato é calculada a medida de similaridade (apêndice C – tabela-2).

A equação (3) representa a medida de similaridade do caso candidato, pois é o resultado da soma das medidas de similaridade individuais dos atributos.

A recuperação é feita pelo SHD considerando a medida de similaridade dos casos candidatos. O SHD através da sua função de recuperação possibilita ao usuário informar um limiar para a medida de similaridade dos casos que serão recuperados. Isso significa dizer que o usuário pode determinar que somente serão recuperados os casos cuja medida de similaridade for igual ou superior ao limiar por ele determinado. Os casos recuperados são listados possibilitando ao usuário acesso aos **atributos problema detalhado e solução do problema**.

Durante os testes do sistema verificou-se que existiam palavras sinônimas que influenciavam na avaliação de similaridade dos casos, pois falhavam na comparação do atributo **problema resumido** (e. g., vídeo e monitor). Solucionou-se o problema com a criação da tabela **sinônimo** no banco de dados do SHD (ver estrutura na tabela 4-6).

Tabela 4-6: Estrutura do dicionário de sinônimos

Atributo	Significado
Principal	Identifica se o atributo "palavra" é principal ou sinônima
Palavra	Palavra principal ou sinônima
Sinônimo	Palavra sinônima

A estrutura para representação dos sinônimos é baseada na teoria de listas circulares duplamente encadeadas [TER91]. Entre os sinônimos, um é escolhido como palavra principal. Durante a aquisição do conhecimento e durante a recuperação, cada uma das palavras é verificada no dicionário de sinônimos.

Quando encontrada, a palavra é convertida para a palavra principal da lista de sinônimos e então armazenada na memória. Um exemplo de uma lista de sinônimos pode ser vista na figura 4-3. O conteúdo da tabela **sinônimo** foi obtido dos especialistas da Datasul e incluído manualmente no banco de dados.

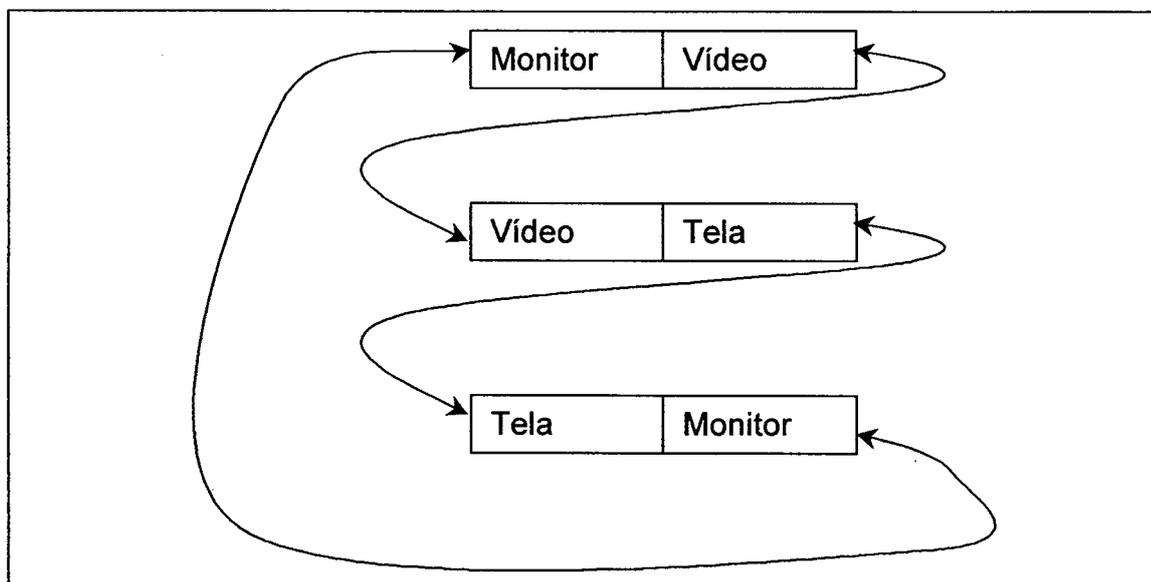


Figura 4-3: Exemplo da estrutura de lista dos sinônimos

A função de similaridade definida nesta seção foi obtida a partir de vários testes com validação da recuperação por parte dos especialistas da Datasul. As similaridades dos atributos e seus pesos foram testadas com um grande número de combinações possíveis e com diferentes números. Aqueles números que representaram casos recuperados com maior sucesso são aqueles apresentados nesta seção. Certamente nem todas as necessidades estão contempladas na indexação, mas com certeza as mais relevantes para o problema. Futuramente é possível que, em virtude de novas versões do *software* da Datasul, novas necessidades mais relevantes que estas aqui apresentadas surjam. Neste caso, será necessário uma reavaliação da indexação.

4.4 Exemplo

A forma escolhida para apresentar o SHD foi através de um exemplo real de um cliente da Datasul. O exemplo será apresentado por figuras copiadas durante a execução do *software*, seguida de comentários necessários ao completo entendimento da situação.

A figura 4-4 abaixo mostra a janela inicial e principal do SHD. Existe nesta janela, os itens de menu: *Cadastros*, *Recuperação*, *Ajuda* e *Sair*. Em *Cadastros* estão as opções de execução dos procedimentos: *Cadastrar Palavras Técnicas*, *Cadastrar Sinônimos*, *Cadastrar Produtos*, *Cadastrar Módulos* e *Cadastrar Programas*.

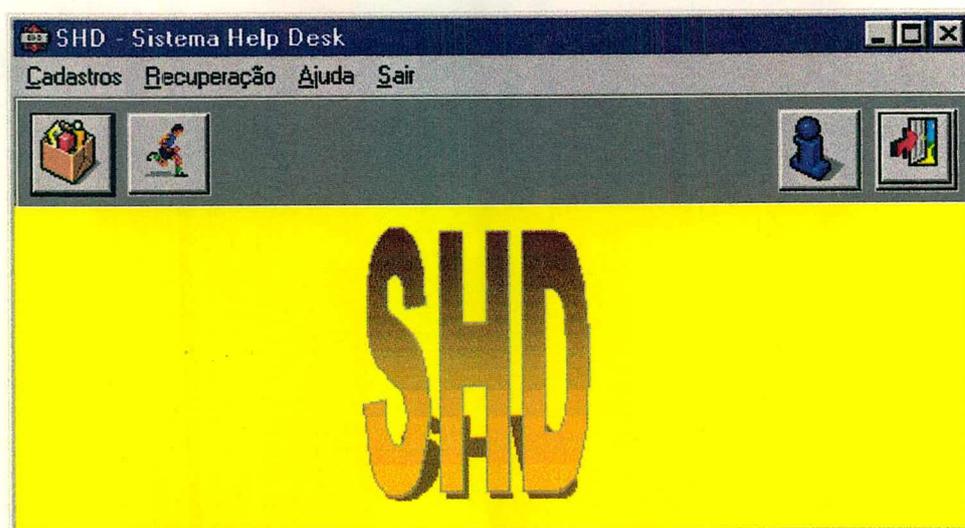


Figura 4-4: Janela principal de apresentação do *software* SHD.

No item de menu *Recuperação* está o processo de informar o problema e executar a busca na base de casos. No item de menu *Ajuda* está o processo de auxílio ao usuário para utilização do *software*. No item de menu *Sair* está o encerramento de utilização do *software*. Abaixo do menu estão alguns botões

(figura de uma caixa de objetos etc) que possibilitam executar respectivamente as mesmas funções dos itens de menu apresentados .

4.4.1 A Função de Recuperação

Ao optar pelo item de menu *Recuperação* ou escolher o botão correspondente a esta função, o sistema irá apresentar a janela *Help Desk* conforme figura 4-5.

Produto	Módulo	Programa	Caso	Sim

Figura 4-5: Janela *Help Desk*

Na janela *Help Desk* o usuário deverá informar os atributos: *Produto*, *Módulo*, *Programa* e *Descrição Detalhada do Problema*. No que se refere aos primeiros três atributos, o sistema já apresenta uma lista das possibilidades, facultando ao usuário informar somente aqueles que forem conhecidos. Entretanto, o quarto atributo, *descrição detalhada do problema*, é indispensável,

sem o qual o sistema acusará erro. Neste atributo o usuário poderá detalhar o seu problema, limitado apenas a 800 caracteres.

O usuário poderá ainda informar um limite de similaridade, assumida pelo sistema inicialmente como 50%. Durante o processo de recuperação o sistema, através da função de similaridade, calcula o percentual de semelhança entre o caso origem e os casos da base de casos. Logo, esta função possibilita ao usuário definir um percentual mínimo a partir do qual lhe interessa recuperar os casos. Somente serão apresentados os casos que atingirem no mínimo este percentual.

Após informados os dados o usuário deve escolher o botão *RECUPERAR* e o sistema irá realizar a busca e apresentar no quadro *CASOS SIMILARES* aqueles encontrados. Veja na figura 4-6 os campos preenchidos com dados de uma possível consulta de um cliente da Datasul. Observe que somente foram relacionados os casos com percentual de similaridade maior ou igual a 50%.

The screenshot shows a window titled "SHD - Help Desk". At the top, there are three dropdown menus: "Produto: MG00", "Módulo: MAP", and "Programa: AP0508". Below these is a text area labeled "PROBLEMA" containing the text "Em que ponto deste programa existe chamada para a função X". To the right of this area is a button labeled "RECUPERAR". Below the "PROBLEMA" area is a table titled "CASOS SIMILARES". The table has five columns: "Produto", "Módulo", "Programa", "Caso", and "Sim". The first two rows of data are highlighted in blue. To the right of the table is a vertical slider control labeled "Limite de Similaridade" with a value of "50" at the bottom.

Produto	Módulo	Programa	Caso	Sim
MG00	MAP	AP0508	Caro Fornecedor Solicitamos progra	63.44
MG00	MAP	AP0508	Solicitacao de chamada 'X'.Objetiv	51.99

Figura 4-6: Uma possível consulta ao SHD

Veja na figura 4-7 e 4-8 a apresentação dos atributos problema detalhado e solução detalhada do primeiro e segundo caso recuperados respectivamente.

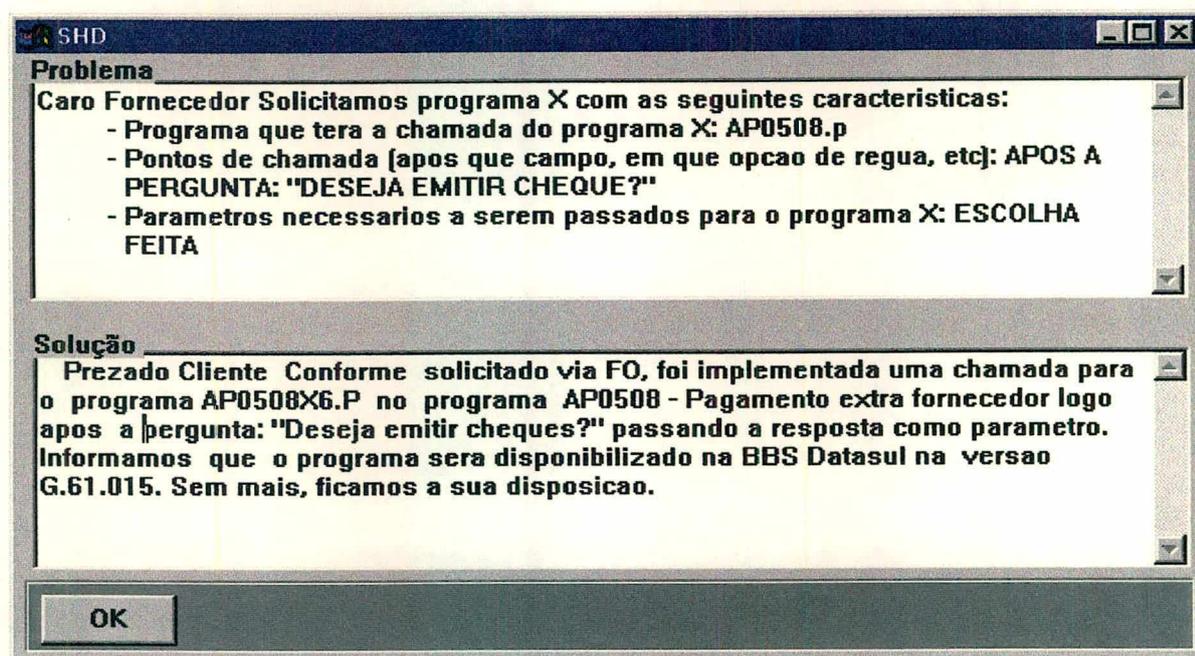


Figura 4-7: Caso recuperado (similaridade = 63.44%)

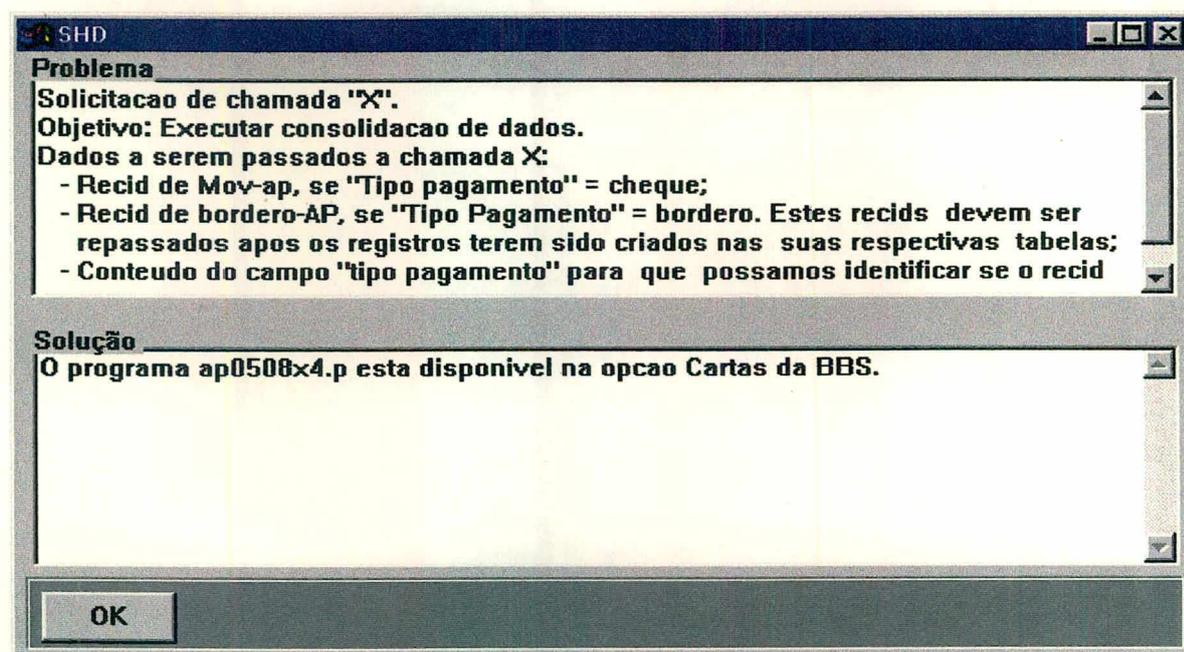


Figura 4-8: Caso recuperado (Similaridade = 51.99%)

A diferença de similaridade (63.44 → 51.99) entre os dois caso deve-se a presença de uma palavra técnica a mais na descrição do primeiro caso e a menor quantidade de palavras também no primeiro caso, o que influi proporcionalmente no cálculo.

O usuário poderá então verificar entre os casos listados aquele que julgar mais apropriado (menor esforço para adaptação da solução). A tarefa de escolha do caso mais apropriado é feita através da constatação das diferenças entre o caso recuperado e o problema atual. Após identificado o caso, a tarefa é adaptar a solução, manualmente, de acordo com as diferenças constatadas.

4.5 Resultados

Para efeito de análise de performance do SHD foram elaborados vários testes dos quais três são detalhados abaixo (tabela 4-7 e apêndice D). O computador usado nestes testes foi um Pentium II 300Mhz com 64Mb de RAM. A base de casos continha 9.872 casos como já descrito (seção 4.3.1). A etapa de conversão da base de dados de FO's em base de casos consumiu 108 horas de máquina. Na função de Recuperação do SHD obteve-se em média 2 segundos como tempo de resposta. O menor tempo de resposta obtido foi 0,6 segundos e o maior tempo foi de 5 segundos (ver tabela 4-7).

Tabela 4-7: Resultados do SHD

Problemas ³	Limite de Similaridade	Tempo de Resposta (segundos)	Casos Recuperados	Sucesso
Teste 1	50	0,6	2	Sim
Teste 2	60	1,5	4	Sim
Teste 3	70	5	2	Não

³ Os problemas são detalhados no apêndice D.

Concluindo, no exemplo apresentado (seção 4.4) o usuário conseguiu obter uma boa resposta ao seu problema. Talvez não seja a melhor resposta, mas pelas informações fornecidas é uma boa resposta. O usuário poderá melhorar a resposta através de um detalhamento da descrição do problema. Poderá ser mais específico, dirigindo a descrição para palavras chaves e mais técnicas, retirando o ruído (palavras que não contribuem para determinação do problema). O usuário poderá ainda trabalhar com o limite do percentual de similaridade, diminuindo-o ou aumentando-o. Enfim, neste exemplo típico, o SHD atendeu aos seus objetivos, já que forneceu uma boa resposta ao usuário, disponibilizando funções para que ele não necessite fazer uma chamada telefônica ao suporte, proporcionando ganhos aos dois lados envolvidos no problema, o cliente e o fornecedor. Também durante os teste do SHD (três deles detalhados na seção 4-5) o resultado foi muito bom. Em torno de 66% dos testes obteve-se sucesso (uma solução passada serviu para resolver um novo problema). Constatou-se que é possível aumentar o percentual de sucesso instruindo o usuário do SHD a dirigir a descrição do problema para aproveitar melhor as potencialidades do mecanismo de inferência, pelo uso de palavras técnicas e pela variação do limite do percentual de similaridade.

5. CONCLUSÃO E DESENVOLVIMENTOS FUTUROS

As empresas buscam constantemente aumentar o nível de satisfação dos seus clientes. Este aumento pode ser pela diminuição dos preços dos seus produtos e serviços, pela rapidez e efetiva entrega, e pela qualidade. O sistema proposto neste projeto busca atender aos três objetivos.

Com o SHD é possível reduzir preços de serviços aos clientes, pois no momento que automatiza o processo de suporte, há uma conseqüente redução com custos de pessoal (menos pessoas no suporte e pessoas menos especializadas) e tarifas de comunicações (menos linhas alugadas).

O SHD proporciona também, soluções mais rápidas aos problemas dos clientes. Em alguns casos os clientes resolverão seus problemas sem dependerem de uma linha disponível, na sua empresa ou no fornecedor, para realizar uma chamada ao suporte (via Internet e com tarifa local). Assim, os benefícios acima se transformam em qualidade de produto e serviço aumentando o nível de satisfação dos clientes.

A tecnologia de RBC proporcionou o fundamento tecnológico no desenvolvimento do SHD sendo essencial e responsável pelo seu sucesso. Somente com uma metodologia de IA como RBC é possível associar heurísticas, lógica e regras de produção. Com RBC foi possível realizar uma busca num espaço onde as semelhanças não estão apenas no conteúdo dos atributos dos casos, mas também em características que somente são identificadas no momento da busca.

Uma importante contribuição que este projeto deixa para a comunidade científica é a comprovação do sucesso obtido na utilização de linguagens de programação comercial e gerenciadores de bancos de dados (neste caso Progress) no armazenamento e manipulação de informações em arquiteturas de representação plana.

Outra contribuição relevante que este projeto deixa é o aproveitamento, por uma nova tecnologia, de informações já armazenadas. A utilização da base de dados de FO's da Datasul, através de um processo de transformá-la em uma base de casos de RBC, atingiu completamente os objetivos. Pelos motivos já abordados na seção 4.4.2 temos clientes mais satisfeitos, o fornecedor do serviço de suporte pôde colocar no setor profissionais com menor conhecimento e experiência de negócio e do *software*, os especialistas estarão envolvidos, na maior parte do tempo de trabalho, em atividades produtivas (bom para o profissional, bom para a empresa) e pelo aproveitamento da base de dados de FO's da Datasul, transformada em base de casos.

A atividade de aquisição do conhecimento é muito intensa e complexa exigindo grande esforço para sua realização. Devido ao aproveitamento da base de dados de FO's a necessidade de aquisição do conhecimento foi bastante reduzida e em pouco tempo se conseguiu visualizar os primeiros resultados do trabalho. A aquisição de conhecimento somente foi necessária para a calibragem da similaridade seção 4.3.5. Para a tarefa de calibragem usou-se o conhecimento do autor e dos especialistas de suporte da Datasul.

A utilização da plataforma de *hardware* e *software* já existente na Datasul também foi muito importante. Outras alternativas de implementação necessitavam

de investimento, principalmente em *software*. Já o SHD aproveitou a linguagem de programação e o gerenciador de banco de dados *Progress 8.2* para programar os processos e armazenar a base de casos, respectivamente.

Este trabalho vem ao encontro das recentes publicações da IA, segundo as quais o atual estágio da IA é aplicar os conceitos até agora estudados. O SHD apresenta os conceitos de IA e RBC aplicados com sucesso no domínio de suporte exemplificado pelo caso Datasul.

5.1 Desenvolvimentos Futuros

A tecnologia de RBC apresenta as etapas Aprendizagem, Representação do Conhecimento, Indexação da Base de Casos, Recuperação e Adaptação. A aprendizagem e a adaptação, por não serem fundamentais e por serem complexas no SHD, desafiam a um novo projeto.

A aprendizagem atual é feita pela inclusão de novos casos na base de dados de FO's e conseqüente inclusão também na base de casos do SHD. O ideal é integrar o SHD com o *software* de FO's atual, reduzindo a duplicação dos dados. Uma alternativa de implementação seria transferir para o SHD todo o controle sobre os problemas (cadastramento, acompanhamento, solucionado ou não, etc) deixando no sistema de FO's somente a parte de registro dos contatos. Nesta proposta, somente seriam novos casos no SHD problemas com similaridade de recuperação inferior a 50%, por exemplo.

A adaptação no SHD é manual, ou seja, o próprio usuário adapta o caso a ser reaproveitado para a situação atual. Um novo projeto poderia aprofundar-se no processo de adaptação de RBC e implementá-lo, ainda que sempre dando ao

usuário a possibilidade de adaptá-lo manualmente. Poderiam ser incluídos novos atributos ou identificados entre os existentes, aqueles que tipicamente podem ser alterados para resolver problemas similares. Por exemplo, quanto ao atributo módulo, a adaptação automática poderia sugerir alterar na solução do problema sua presença pela do módulo do problema atual.

A criação de mais índices para recuperação, após um tempo de utilização do SHD, poderá ser avaliada. No momento não há subsídios para criá-los, mas acredita-se que, com a utilização, esta necessidade poderá se verificar.

6. BIBLIOGRAFIA

- [AAM94] Aamodt, A. & Plaza, E., Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches, *AI Communications*, 7(i), 39-59, 1994.
- [ACR92] Acron, T. & Walden, S., *SMART: Support management automated reasoning technology for Compaq customer service*, In Proceedings of the 4th Innovative Application of AI Conference, 1992.
- [BAR97] Barreto, J. M., *Inteligência Artificial no Limiar do século XXI*, Florianópolis - SC, 1997.
- [BIT96] Bittencourt, G., *Inteligência Artificial Ferramentas e Teorias*, Instituto de Computação, UNICAMP, Campinas – SP, 1996.
- [BRA79] Brachman, R. *On the epistemological status of semantic networks*. in N. Findler (ed.) *Associative Networks, Representation and Use of Knowledge by Computer*, Academic Press, 1979.
- [CAR86] Carbonell, J. G. , Michalski, R. S. e Mitchell T., *An Overview of Machine Learning*. In *Machine Learning: An Artificial Intelligence Approach*. páginas 3-23. Springer-Verlag, R. S. Michalski, J. C. Carbonell, T. Mitchel, 1984.
- [DAT91] Date, C. J., *Introdução ao Sistema de Banco de Dados*, Ed. Campus Ltda, Rio de Janeiro – RJ, 1991.
- [DEA95] Dean, T., Allen, J. & Aloimonos. Y., *Artificial Intelligence Theory and Practice*. Addison-Wesley Publishing Company, Menlo-Park, CA, 1995.
- [DOM92] Domeshek, E., *Do the Right Thing: A Component Theory for Indexing Stories as Social Advice*, Ph.D. diss, Dept. of Computer Science, Yale University, Also Technical Report, 26, Institute for the Learning Sciences, Northwestern University, 1992.
- [DUR94] Durkin, J., *Expert Systems: design and development*, Prentice Hall Inc., Englewood Cliffs NJ, 1994.
- [FOG92] Fogarty, D. W., Blackstone, J. H. & Hoffman T. R., *Production & Inventory Management*, APICS, New York – NY, 1992
- [HAM86] Hammond, K. J., *CHEF: A Model of Case-Based Planning*. In Proceedings od AAI-86, Cambridge, MA:AAI Press / MIT Press, 1986.

- [HIN88] Hinrichs, T. R., *Towards and architecture for open world problem solving*, in Proceedings: Workshop on Case Based Reasoning (DARPA), Clearwater, Florida, San Mateo – CA: Morgan Kaufmann, 1988.
- [HIN89] Hinrichs, T. R., *Strategies for adaptation and recovery in a design problem solver*, in Proceedings: Workshop on Case Based Reasoning (DARPA), Pensacola Beach, Florida, San Mateo – CA: Morgan Kaufmann, 1989.
- [HIN92] Hinrichs, T. R., *Problem solving in open worlds: A case study in design*. Northvale NJ: Erlbaum, 1992.
- [KAM93] Kamp, G., *Integrating Semantic Structure and Technical Documentation in Case-Based Service Support*, University of Hamburg, 1993
- [KIT94] Kitano, H. & Shimazu H., *The Experience-Sharing Architecture: A Case Study in Corporate-Wide Case-Based Software Quality Control*, 1994
- [KOL93] Kolodner, J. *Case-Based Reasoning*, Morgan Kaufmann Pub. Inc., San Francisco, CA, 1996
- [KOT89] Koton, P., *Using Experience in Learning and Problem Solving*. PhD. Thesis, Laboratory of Computer Science, Massachusetts Institute of Technology. MIT/LCS/TR-441, 1989.
- [LEA92] Leake, D. B., *Evaluating Explanations: A Content Theory*, Hillsdale, N.J.: Lawrence Erlbaum, 1992.
- [LEA96] Leake, D. B., *Case-Based Reasoning Experiences, Lessons & Future Directions*, AAAI Press/The MIT Press, Menlo Park, CA 94025, 1996.
- [LEE96] Lee, J. S. & Xon, Y. X., *A Customer Service Process Innovation Using the Integration of Data Base and Case Base*, Expert Systems with Applications, Vol. 11, No. 4, pp. 543-552, 1996
- [LEN96] Lenz, M., Burkhard H-D & Pirk, P., *CBR for diagnosis and decision support*, AI Communications 9, 138-146, 1996.
- [MAR91] Martin, J., *Engenharia da Informação: introdução*, Ed. Campus Ltda, Rio de Janeiro – RJ, 1991
- [QUI68] Quillian, M.R. *Semantic Memory, Semantic Information Processing*, M. L. Minsky (ed.), MIT Press, Cambridge, Mass, 1968.
- [QUI86] Quinlan, J. R., *Induction of Decision Trees in Machine Learning*, Morgan Kaufmann Pub. Inc., San Francisco, CA, 1996.

- [RAB90] Rabuske, R. A., *Linguagem Natural*, Trabalho final do XVII curso avançado em sistemas de computação – CASC 17, IBM do Brasil, Rio de Janeiro – RJ, 1990.
- [RAB95] Rabuske, R. A., *Inteligência Artificial*, Universidade Federal de Santa Catarina, Florianópolis, 1995.
- [RIC93] Rich, E. & Knight, K., *Inteligência Artificial*, Ed. Makron Books, São Paulo, 1993.
- [RIE89] Riesbeck, C. K. & Schank, R. L., *Inside Case-based Reasoning*. Hillsdale, NJ: Erlbaum, 1989.
- [RUS95] Russel S. & Norvig P., *Artificial Intelligence: A Modern Approach*, Prentice Hall Inc, 1995.
- [SCH77] Schank, R. & Abelson R., *Scripts, Plans, Goals and Understanding*. Hillsdale, NJ: Lawrence Erlbaum Associates, 1977.
- [SCH82] Schank, R., *Dynamic Memory: A theory for learning in computers and people*. New York: Cambridge Univ. Press. 1992.
- [SCH90] Schank, R. & Osgood, R., *A Content Theory of Memory Indexing*, Technical Report, 2, Institute for the Learning Sciences, Northwestern University, 1990.
- [SIM85] Simpson, R. L., *A computer model of case-based reasoning in problem solving: An investigation in the domain of dispute mediation*, Georgia Institute of Technology, Scholl of Information and Computer Science Technical Report No. GIT-ICS-85/18, Atlanta, 1985.
- [SIM90] Simoudis, E. & Miller, J. S., *Validade Retrieval in Case-Based Reasoning*, Digital Equipment Corporation, Cambridge Research Lab. 1990 .
- [SIM91] Simoudis, E. & Miller, J. S., *The Application of CBR to Help Desk Applications*. In *Proceedings of the DARPA Case-based Reasoning Workshop*, (1991) pp. 25-50.
- [SIM92] Simoudis, E., *Using Case-Based Retrieval For Customer Technical Support*. In *IEEE EXPERT*, OCTOBER 1992.
- [SIM93] Simoudis, E. & Shutt, T. S., *COPRA: Computer Operations Problem Resolution Assistant*. IEEE, 1993.

- [SYK87] Sykara, E. P., *Resolving adversarial conflicts: An approach to integrating case-base and analytic methods*. Georgia Institute of Technology, School of Information and Computer Science Technical Report No. GIT-ICS-87/26. Atlanta GA, 1987.
- [SYK88] Sykara, E. P., *Using Case Based Reasoning for plan adaptation and repair*, in Proceedings Workshop on Case Based Reasoning (DARPA), Clearwater, Florida, San Mateo – CA: Morgan Kaufmann, 1988.
- [TEA91] Terra, E., *Curso Prático de Gramática*, Ed. Scipione Ltda, São Paulo – SP, 1991.
- [TER91] Terada, R., *Desenvolvimento de Algoritmos e Estruturas de Dados*, McGraw-Hill, Makron, São Paulo – SP, 1991.
- [WAT86] Waterman, D.A., *A guide to expert systems*. Addison-Wesley Publishing Company, 1986.
- [WAT97] Watson, I., *Applying Case-Based Reasoning: Techniques for Enterprise Systems*, Morgan Kaufmann Pub. Inc., San Francisco, CA, 1997.
- [WEB98] Weber, R., *Pesquisa Jurisprudencial Inteligente*, Tese de Doutorado, Curso de Pós-graduação em Engenharia de Produção, UFSC, Florianópolis – SC, 1998.
- [WIT53] Wittgenstein, L., *Philosophical Investigations*. United Kingdom:Blackwell, 1953
- [ZAD65] Zadeh, L. A., Fuzzy Sets. *Information and Control*, 8:338-353, 1965.

APÊNDICE B

Regras para Comparação e Avaliação da Similaridade

Regra	Descrição
1	Se (Mce = "AP" e Mbc = "CR") ou (Mce = "CR" e Mbc = "AP") então "recuperar"
2	Se (Mce = "CT" e Mbc = "FB") ou (Mce = "FB" e Mbc = "CT") então "recuperar"
3	Se (Pce = "EMB00" e Pbc = "EMH00") ou (Pce = "EMH00" e Pbc = "EMB00") então Recuperar.
4	Se (Pce = "MG00" e Pbc = "MF00") ou (Pce = "MF00" e Pbc = "MG00") então "recuperar"
5	Se (Pce = "MG00" e Pbc = "MF50") ou (Pce = "MF50" e Pbc = "MG00") então "recuperar"
6	Se (Pce = "MG00" e Pbc = "MI00") ou (Pce = "MI00" e Pbc = "MG00") então "recuperar"
7	Se (Pce = "MG00" e Pbc = "MI01") ou (Pce = "MI01" e Pbc = "MG00") então "recuperar"
8	Se (Pce = "MG00" e Pbc = "MJ00") ou (Pce = "MJ00" e Pbc = "MG00") então "recuperar"
9	Se (Pce = "MF00" e Pbc = "MF50") ou (Pce = "MF50" e Pbc = "MF00") então "recuperar"
10	Se (Pce = "MF00" e Pbc = "MI00") ou (Pce = "MI00" e Pbc = "MF00") então Recuperar
11	Se (Pce = "MF00" e Pbc = "MI01") ou (Pce = "MI01" e Pbc = "MF00") então "recuperar"
12	Se (Pce = "MF00" e Pbc = "MJ00") ou (Pce = "MJ00" e Pbc = "MF00") então "recuperar"
13	Se (Pce = "MF50" e Pbc = "MI00") ou (Pce = "MI00" e Pbc = "MF50") então "recuperar"
14	Se (Pce = "MF50" e Pbc = "MI01") ou (Pce = "MI01" e Pbc = "MF50") então "recuperar"
15	Se (Pce = "MF50" e Pbc = "MJ00") ou (Pce = "MJ00" e Pbc = "MF50") então "recuperar"
16	Se (Pce = "MI00" e Pbc = "MI01") ou (Pce = "MI01" e Pbc = "MI00") então "recuperar"
17	Se (Pce = "MI00" e Pbc = "MJ00") ou (Pce = "MJ00" e Pbc = "MI00") então "recuperar"
18	Se (Pce = "MI01" e Pbc = "MJ00") ou (Pce = "MJ00" e Pbc = "MI01") então "recuperar"
19	Se (Pce = "RHC00" e Pbc = "RHD00") ou (Pce = "RHD00" e Pbc = "RHC00") então "recuperar"
20	Se (Pce = "RHC00" e Pbc = "RHI00") ou (Pce = "RHI00" e Pbc = "RHC00") então "recuperar"
21	Se (Pce = "RHD00" e Pbc = "RHI00") ou (Pce = "RHI00" e Pbc = "RHD00") então "recuperar"
Mce = Atributo Módulo do Caso de Entrada Mbc = Atributo Módulo do Caso da Base de Casos Pce = Atributo Produto do Caso de Entrada Pbc = Atributo Produto do Caso da Base de Casos	

APENDICE C

Medidas de Similaridade usadas no SHD

		Atributos		
		Produto [0 , 1] Peso 10	Módulo [0 , 1] Peso 10	Programa 0 / 1 Peso 10
Comparação	=	1,0	1,0	1,0
	<>	0,0	0,0	0,0
R E G R A S	1	-	0,8	
	2	-	0,5	
	3	0,7		
	4	0,8		
	5	0,7		
	6	0,6		
	7	0,5		
	8	0,4		
	9	0,9		
	10	0,8		
	11	0,7		
	12	0,6		
	13	0,8		
	14	0,7		
	15	0,6		
	16	0,9		
	17	0,8		
	18	0,8		
	19	0,8		
	20	0,6		
	21	0,7		

Tabela-1

		Caso de Entrada – Atributo Problema Resumido Peso 70			
		Palavra 1 [0 , 1]	Palavra 2 [0 , 1]	Palavra "n" [0 , 1]
Caso Recuperado da Base de Casos	=	1/m	1/m	1/m
	<>	0	0	0

n = número de palavras do caso de entrada resumido
 n' = número de palavras do caso recuperado da base de casos
 m = Max(n, n')

Tabela-2

APENDICE D

Teste 1				
Problema Novo				
Produto	Módulo	Programa	Similaridade	
MG00	MAP	AP0508		
Problema:	Em que ponto deste programa existe chamada para a função X			
Problemas Recuperados				
MG00	MAP	AP0508	63,44	
Caso:	Caro Fornecedor Solicitamos programa X com as seguintes características: - Programa que terá a chamada do programa X: AP0508.p; - Pontos de chamada (após que campo, em que opção de régua, etc): APÓS A PERGUNTA: "DESEJA EMITIR EQUÊ?"; - Parâmetros necessários a serem passados para o programa X: ESCOLHA FEITA			
Solução:	Prezado Cliente Conforme solicitado via FO, foi implementada uma chamada para o programa AP0508X6.P no programa AP0508 - Pagamento extra fornecedor logo após a pergunta: "Deseja emitir cheques?" passando a resposta como parâmetro. Informamos que o programa será disponibilizado na BBS Datasul na versão G.61.015. Sem mais, ficamos a sua disposição.			
MG00	MAP	AP0508	51,99	
Caso:	Solicitação de chamada "X". Objetivo: Executar consolidação de dados. Dados a serem passados a chamada X: Recid de Mov-ap, se "Tipo pagamento" for cheque; Recid de bordero-AP, se "Tipo Pagamento" for bordero, sendo que estes recids devem ser repassados após os registros terem sido criados nas suas respectivas tabelas; conteúdo do campo "tipo pagamento" para que possamos identificar se o recid passado e de mov-ap ou bordero-AP.			
Solução:	O programa ap0508x4.p esta disponível na opção Cartas da BBS			
Teste 2				
Problema Novo				
MG00	MCE	CE0201		
Problema:	Como entrar com notas do módulo de estoque que possuam e contabilizem os seguintes impostos: de renda, I.N.S.S, I.S.S.			
Problemas Recuperados				
MG00	MCE	CE0201	81,80	
Caso:	Necessito entrar com notas do módulo de estoque que possua :imposto de renda, I.N.S.S, I.S.S. Estas notas não integram com O.F., mais sim com AP. Como devo proceder, inclusive para que contabilize estes Impostos.			
Solução:	O Recebimento não trata especificamente estes impostos, assim sugerimos que seja utilizado a tela de digitação de Duplicatas, onde devem ser incluídos documentos com espécie "IR", "IS" e "IN", e o respectivo valor. Após, no módulo de Contas a Pagar, devem ser acertados os títulos com informações específicas, como por exemplo o código de retenção, para o IR. A contabilização será normal no estoque, lançando os valores na conta Transitória de Fornecedores, apenas no módulo AP é que as contas dos impostos serão utilizadas.			
MG00	MCE	CE0201	63,00	
Caso:	Para notas fiscais de devolução, inclusive com impostos, onde existam itens com naturezas de operações diferentes, o programa leva para o movimento de estoque os itens com natureza de operação do primeiro item? Como são contabilizados os Impostos?			
Solução:	Sim, o programa leva para o movimento de estoque os itens com natureza de operação do primeiro item. Os impostos são contabilizados da mesma forma de uma nota fiscal de devolução normal.			

MG00	MCE	CE0201	61,00	
Caso:	Solicitamos Programa de acerto para corrigir as Notas Fiscais de Entrada digitados a partir de 01/01/97 com a classificação Fiscal com 10 dígitos.			
Solução:	Estamos disponibilizando na BBS opção "Cartas" e no suporte self-service opção B/C o programa especial SPCE5004, cuja função é alterar o formato da classificação fiscal de 10 para 8 dígitos.			
MG00	MCE	CE0201	60,33	
Caso:	Em virtude do decreto 2092(D.O.U) que trata a classificação fiscal de 10 p/ 8 caracteres, e ao grande número de itens em nosso estoque, a alteração dos mesmos demanda um certo prazo e em janeiro/97 para não parar o processo, recebemos varias notas fiscais com a classificação de 10 posições. Como poderemos resolver os itens que entraram com a classificação antiga??			
Solução:	Prezado Cliente: Conforme solicitação, estamos disponibilizando na opção "Cartas" do BBS, o programa especial SPCE5004, cuja função e alterar o formato da classificação fiscal de 10 posições para 8 posições nos documentos gerados pela rotina de recebimento de materiais (CE0201), conforme o parametrizado na rotina de atualização de classificações fiscais (CD0603).			
Teste 3				
Problema Novo				
MG00	MCD			
Problema:	O programa cd0924 esta dando mensagem de erro de compilação			
Problemas Recuperados				
MG00	MCD	CD0904	76,00	
Caso:	Compilação de programa esta dando erro, cuja mensagem é: "nome de campo desconhecido - perc-red-sub", no sub-programa cd0904a.p			
Solução:	Caro Cliente, favor buscar nova versão (G.17.003) do programa cd0904 no self-service			
MG00	MCD	CD0205	70,62	
Problema:	Problemas na compilação: Conforme resposta a FO numero 128, o arquivo INC.ZIP foi descompactado para atualização do diretório inc; no entanto, o erro persistiu somente durante a compilação do programa mgen105a.p, que se encontra juntamente com o programa CD0205-G.29.000.			
Solução:	Devido ao fato de termos feito uma liberação manual para sua empresa, enviamos junto com seus programas um sub-programa do multi-planta. Verificamos que sua empresa nao possui o modulo do multi-planta, portanto realmente qualquer programa (ou sub-programa) deste modulo acusaria erro de compilação. Pedimos que este programa seja eliminado de seu diretório. OBS: este sub-programa somente eh chamado pelo cd0205 quando o módulo de multi-planta estiver implantado, portanto nao há problema algum em eliminar este sub-programa.			