


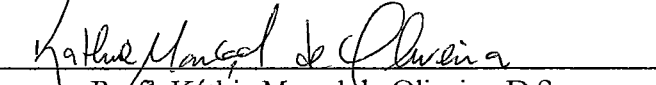
REPRESENTAÇÃO DA DISTRIBUIÇÃO DO CONHECIMENTO,  
HABILIDADES E EXPERIÊNCIAS ATRAVÉS DA ESTRUTURA  
ORGANIZACIONAL

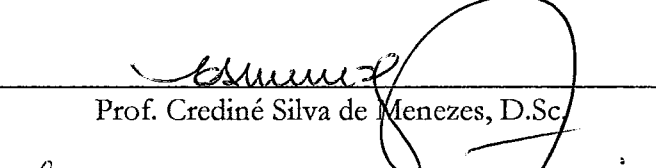
Gleison dos Santos Souza

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS  
PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE  
FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS  
NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM  
ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Aprovada por:

  
Prof.<sup>a</sup> Ana Regina Cavalcanti da Rocha, D.Sc.

  
Prof.<sup>a</sup> Kátia Marçal de Oliveira, D.Sc.

  
Prof. Crediné Silva de Menezes, D.Sc.

  
Márcio de Oliveira Barros, D.Sc.

RIO DE JANEIRO, RJ - BRASIL  
ABRIL DE 2003

SOUZA, GLEISON DOS SANTOS

Representação da Distribuição do  
Conhecimento, Habilidades e Experiências  
através da Estrutura Organizacional [Rio de  
Janeiro] 2003

IX, 96 p. 29,7 cm (COPPE/UFRJ,  
M.Sc., Engenharia de Sistemas e  
Computação, 2003)

Tese - Universidade Federal do Rio de  
Janeiro, COPPE

1. Gerência de Conhecimento
  
2. Ambientes de Desenvolvimento de  
Software Orientados à Organização

I. COPPE/UFRJ II. Título (série)

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

REPRESENTAÇÃO DA DISTRIBUIÇÃO DO CONHECIMENTO,  
HABILIDADES E EXPERIÊNCIAS ATRAVÉS DA ESTRUTURA  
ORGANIZACIONAL

Gleison dos Santos Souza

Abril/2003

Orientadora: Ana Regina Cavalcanti da Rocha

Programa: Engenharia de Sistemas e Computação

Esta dissertação descreve as características desejadas de uma ferramenta para representação da distribuição de conhecimento, habilidades e experiências através da estrutura da organização. Seu objetivo principal é possibilitar que desenvolvedores de software rapidamente encontrem, dentro da estrutura organizacional, os profissionais mais adequados à realização de uma atividade ou à solução de um problema. Para isto, permite representar a estrutura organizacional e descrever as suas unidades organizacionais e posições desempenhadas dentro destas unidades, além de definir que conhecimentos, habilidades e experiências são necessários para o bom desempenho de cada uma dessas posições. A ferramenta é genérica, ou seja, independente de organização ou domínio específicos, fazendo uso de ontologias de organização, do domínio e de Engenharia de Software para descrever as organizações que desenvolvem e mantêm software. Dispõe de mecanismos para busca e visualização das informações registradas, podendo tanto ser utilizada no planejamento quanto no decorrer de projetos com o intuito de obter informações sobre a estrutura organizacional e sobre os profissionais da organização. O trabalho fundamenta-se nos conceitos de Gestão do Conhecimento e Ambientes de Desenvolvimento de Software Orientados à Organização.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

REPRESENTATION OF THE DISTRIBUTION OF KNOWLEDGE, SKILLS  
AND EXPERIENCES THROUGHOUT THE ORGANIZATIONAL STRUCTURE

Gleison dos Santos Souza

Abril/2003

Advisor: Ana Regina Cavalcanti da Rocha

Department: System and Computing Engineering

This dissertation describes the characteristics of a tool that aims to represent the distribution of knowledge, skills and experiences throughout the organizational structure. Its goal is to make possible for developers to find inside the organization's structure the most capable professionals for an activity or the solution of a problem related to the development of a software's project. The organizational structure is described into organizational units and each organizational unit is described into positions. The description of the structure includes a set of important characteristics to the different positions inside the organization. The presented concepts are generic and allow the description of any type of organization independently of its core business. It uses organization, domain and software engineering ontologies to describe organizations that develop and maintain software. It has search and visualization mechanisms and can be used before or during software projects to gather information about organizational structure and organizational professionals. The work is based on Knowledge Management and Enterprise-Oriented Software Development Environments concepts.

*À Vanessa*

## AGRADECIMENTOS

Gostaria de agradecer aqui a todos os que me ajudaram direta ou indiretamente na realização desta tese. O primeiro agradecimento vai para minha família, sempre presente apesar, às vezes, da distância. Também é preciso agradecer aos amigos Alexandre Novello, Bruno Ratton, Carla Delgado, Dario, Fábio Zlot, Glauco e José Antônio. Às, várias, amizades adquiridas durante o trabalho nos projetos da COPPETEC, Carmen, Carla Valle, Gustavo, Leonardo, Marcelo, Vanessa Almeida. Aos amigos do mestrado Augusto, Catarina, Cátia, Fábio, Lílian, Luciana, Luís Filipe, Mariano, Mariella, Monalessa, Patrícia, Roberta, Rodrigo, Viviane. Aos amigos do laboratório, Sávio, David, Rômulo e em especial ao Sômulo pela ajuda sempre que necessária. Ao pessoal administrativo da COPPE/Sistemas, em especial à Ana Paula Prata, e aos professores da linha de Engenharia Software, Guilherme e Cláudia. À CAPES pelo apoio financeiro. À Káthia pelo apoio e amizade desde o início do meu projeto final de curso. À Karina por seus valiosos comentários, por nossas conversas e por estar sempre disposta a colaborar na realização desta tese. À Ana Regina, minha orientadora, pela amizade, orientação e incentivo, por acreditar em mim e no meu trabalho. Ao Márcio e ao Crediné por participarem da minha banca. Por fim, mas não menos importante, um agradecimento especial à minha namorada, Vanessa, pelo carinho, companheirismo, paciência e compreensão durante os meus anos de mestrado.

*Todos os dias quando acordo,  
Não tenho mais o tempo que passou  
Mas tenho muito tempo:  
Temos todo o tempo do mundo.*

Legião Urbana em *Tempo Perdido*

# ÍNDICE

<b><u>CAPÍTULO I - INTRODUÇÃO .....</u></b>	<b><u>1</u></b>
1.1 Motivação.....	1
1.2 Objetivo da Tese.....	3
1.3 Estrutura do Texto.....	4
<b><u>CAPÍTULO II - GESTÃO DE CONHECIMENTO .....</u></b>	<b><u>6</u></b>
2.1 Introdução.....	6
2.2 Definindo Conhecimento.....	7
2.3 Adquirindo Conhecimento .....	11
2.4 Gestão de Conhecimento.....	14
2.5 Memória Organizacional.....	20
2.6 Ontologias e Gestão de Conhecimento.....	21
2.7 Ferramentas para Gestão de Conhecimento.....	23
2.8 Conclusão.....	26
<b><u>CAPÍTULO III - AMBIENTES DE DESENVOLVIMENTO DE SOFTWARE E A ESTAÇÃO</u></b>	
<b><u>TABA .....</u></b>	<b><u>27</u></b>
3.1 Introdução .....	27
3.2 Ambientes de Desenvolvimento de <i>Software</i> .....	28
3.3 A Estação Taba.....	29
3.4 Ambientes de Desenvolvimento de <i>Software</i> Orientados a Domínio.....	30
3.5 Ambientes de Desenvolvimento de <i>Software</i> Orientados a Organização .....	32
3.6 Implementação Atual.....	34
3.7 Conclusão.....	35
<b><u>CAPÍTULO IV - REPRESENTAÇÃO DA DISTRIBUIÇÃO DO CONHECIMENTO,</u></b>	
<b><u>HABILIDADES E EXPERIÊNCIAS ATRAVÉS DA ESTRUTURA ORGANIZACIONAL.....</u></b>	<b><u>37</u></b>
4.1 Introdução.....	37
4.2 Abordagens Relacionadas na Literatura.....	38
4.3 Fundamentação / Características Desejadas.....	42
4.4 Características da Ferramenta.....	45
4.4.1 Estrutura Organizacional.....	47
4.4.2 Tipos de ontologias utilizadas.....	49
4.4.3 Conhecimentos, Habilidades e Experiências .....	52
4.4.4 Busca .....	56
4.5 Conclusão.....	58
<b><u>CAPÍTULO V - A FERRAMENTA SAPIENS.....</u></b>	<b><u>59</u></b>
5.1 Introdução.....	59
5.2 Cordis.....	60
5.3 Cordis-FBC .....	62



5.4 Exemplo de Uso / Detalhes da Implementação..... 65

5.5 Conclusão..... 80

**CAPÍTULO VI - CONSIDERAÇÕES FINAIS..... 81**

6.1 Introdução..... 81

6.2 Conclusão e Contribuições ..... 81

6.3 Perspectivas Futuras..... 83

**REFERÊNCIAS BIBLIOGRÁFICAS ..... 85**

**ANEXO I - MODELO DE CLASSES UTILIZADO PARA DESCREVER A ESTRUTURA ORGANIZACIONAL..... 94**

**ANEXO II - NOTAÇÃO UTILIZADA PARA REPRESENTAR ONTOLOGIAS UTILIZANDO UML..... 96**

# CAPÍTULO I - INTRODUÇÃO

## 1.1 Motivação

O desenvolvimento de um *software* envolve muitas variáveis, o que torna a sua realização crítica em algumas situações. Muitas vezes, decisões sobre alocação de pessoal precisam ser tomadas com extrema agilidade, assim como é necessário descobrir com rapidez quem consultar sobre determinado assunto referente à atividade que está sendo realizada no momento. Ter uma base de dados atualizada e sempre à disposição sobre a organização é um fator determinante para que seja possível tomar decisões desse tipo da maneira mais acertada.

A motivação para este trabalho surgiu da necessidade de tornar possível a desenvolvedores de *software* encontrarem rapidamente, dentro da estrutura organizacional, os profissionais mais adequados à realização de uma atividade ou à solução de um problema relativo ao desenvolvimento de um projeto de *software*.

Conhecimento é produzido e utilizado durante o desenvolvimento de *software*. Por diversos motivos, o conhecimento importante neste contexto encontra-se disperso na mente de várias pessoas e/ou em documentos armazenados em vários lugares e mídias. Conseqüentemente, desenvolvedores de *software*, muitas vezes, falham em atender rápida e adequadamente as solicitações de seus clientes, porque o conhecimento de que necessitam foi perdido pela organização ou está em lugares desconhecidos ou inacessíveis. Sempre que o conhecimento requerido não está disponível, o desenvolvedor de *software* tem que partir do zero para a solução do problema, experimentar e construir novamente o conhecimento, o que tem sérias implicações em termos da qualidade da solução, do tempo e do custo necessários para obtê-la. Recursos anteriormente gastos para obter o conhecimento serão novamente requeridos e, pior, erros cometidos anteriormente podem ser novamente cometidos (VILLELA *et al.*, 2001b).

Quando um novo sistema envolvendo conhecimento sobre o modelo organizacional precisa ser desenvolvido seria bom saber quais são as pessoas na organização que podem fornecer informações sobre o sistema. Da mesma forma, quando

um desenvolvedor de *software* precisa de um especialista em algum assunto para a realização de uma tarefa deveria haver um meio de encontrar dentro da organização quem possui este conhecimento ou onde é utilizado.

Um sistema que dê suporte à gestão de capacidades baseado numa ontologia de capacidades e outras especificações bem definidas pode ajudar uma organização a alinhar as habilidades atuais e futuras dos empregados com os objetivos estratégicos de negócio (Stader e Macintosh, 1999). A especificação das capacidades humanas permite a realização de processos adequados à capacidade dos recursos humanos da organização (ACUÑA *et al.*, 2000).

Além disso, é fundamental destacar que conhecimento é indiscutivelmente um importante ativo de uma empresa e, por isso, sua formalização, captura e reutilização devem ser sempre incentivadas. Entretanto, o conhecimento de uma organização é muito amplo e formalizá-lo pode ser uma tarefa extremamente difícil. Por isso, a abordagem inicial, visando montar uma infra-estrutura básica de aquisição e reutilização de conhecimento, é uma tentativa de formalização do conhecimento que a Organização detém de si mesma (SANTOS, 2001).

Para poder atingir o objetivo almejado é necessário identificar quais as informações prioritárias sobre a organização se deseja, em um primeiro momento, armazenar. Uma parte importante do conhecimento organizacional é o auto-conhecimento, ou seja, tudo o que a organização sabe sobre si mesma. Este conhecimento engloba, mas não é restrito a, a estrutura de departamentos, cargos e posições da empresa, aos processos organizacionais realizados e informações sobre os funcionários, tais como que posições exercem ou exerceram e conhecimentos, habilidades e experiências possuídos por eles.

Seguindo esse raciocínio, podemos dividir esse conhecimento em três partes distintas, mas que se relacionam: Estrutura da Organização, Processos Organizacionais e Informações sobre Pessoal. Os elementos de ligação entre essas três partes são as habilidades, conhecimentos e experiências necessários a um funcionário para a realização de etapas do processo organizacional ou requeridos para sua alocação a uma unidade organizacional.

## 1.2 Objetivo da Tese

O objetivo desta tese é descrever a representação da distribuição das habilidades, conhecimentos e experiências detidos por cada indivíduo através da estrutura de uma organização. A proposta se baseará em idéias propostas no âmbito de Gestão de Conhecimento, da Estação TABA e dos Ambientes de Desenvolvimento de *Software* Orientados a Organização (ADSOrg).

Como decorrência dessa proposta, as características de uma ferramenta para tal é apresentada. Ela é genérica o suficiente para que seja independente de uma organização ou domínios específicos. Como está inserida no contexto da Estação TABA, beneficia-se de alguns elementos propostos para compor a infra-estrutura básica definida para os ADSOrgs (VILLELA *et al.*, 2000 e 2001b): faz uso de ontologias de organização, do domínio e de Engenharia de *Software* para descrever as organizações que desenvolvem e mantêm *software*.

Os usuários da ferramenta são, a princípio, todos os funcionários da organização a quem ela se destina, uma vez que através de sua utilização serão coletadas informações importantes e vitais para a empresa como um todo. Entretanto, nem todos os dados coletados são de interesse coletivo. A descrição da organização (unidades organizacionais e posições existentes), além da informação sobre os funcionários atuais, podem servir de fonte de consulta para uma grande gama de profissionais. Por exemplo, os dados sobre as habilidades, conhecimentos e experiências detidos por cada pessoa além da alocação dessas pessoas às posições serão mais úteis ao engenheiro de *software* responsável por definir a alocação de pessoal em um projeto. Ao departamento de Recursos Humanos, por outro lado, será prioritária a informação sobre alocação de pessoal às posições dentro das unidades organizacionais.

Uma vez que os dados estejam cadastrados, é necessário permitir que estes sejam acessíveis sempre que necessário e desejado. Para isso são descritas formas de visualização da estrutura organizacional e navegação através dela, bem como formas de pesquisas sobre o conteúdo da base de dados através de mecanismo que permita identificar de forma direta ou indireta pessoas detentoras de determinado conhecimento, habilidade ou experiência.

Desta forma, serão possíveis a representação da estrutura organizacional (unidades que a compõe e como estas se relacionam quanto à distribuição de autoridade e

responsabilidade), descrição das unidades organizacionais (que posições são desempenhadas e que conhecimentos, habilidades e experiências são requeridos) e a associação dos profissionais da organização às posições que desempenham (incluindo os conhecimentos, habilidades e experiências possuídos). A descrição dos processos organizacionais foge ao escopo desta tese. O objetivo principal do trabalho é a descrição de uma organização que desenvolve *software*, entretanto, os conceitos apresentados são genéricos e espera-se que seja possível, a descrição de qualquer tipo de organização independente de sua área de atuação.

### 1.3 Estrutura do Texto

A estrutura dessa tese foi definida com o intuito de guiar o leitor ao seu melhor entendimento. Após a introdução são apresentados conceitos de gestão de conhecimento utilizados no trabalho, seguidos pela apresentação de um histórico da Estação TABA. A descrição da tese propriamente dita ocupa dois capítulos, um com a fundamentação teórica e outro com um exemplo de utilização da ferramenta construída. Após as considerações finais são apresentadas as referências bibliográficas utilizadas.

Esta tese está dividida nos seguintes capítulos:

- **Capítulo I - Introdução:** capítulo introdutório, descrevendo sucintamente a motivação e objetivos do trabalho, além da organização da tese.
- **Capítulo II – Gestão de Conhecimento:** apresenta uma revisão da literatura sobre Gestão de Conhecimento;
- **Capítulo III – Ambientes de Desenvolvimento de *Software* e a Estação TABA:** apresenta um histórico da Estação TABA ao longo do tempo, incluindo sua definição e o surgimento dos Ambientes de Desenvolvimento Orientados a Domínio e a Organização.
- **Capítulo IV – Representação da Distribuição do Conhecimento, Habilidades e Experiências através da Estrutura Organizacional:** neste capítulo há a motivação teórica deste trabalho e situa-o no contexto de gestão de conhecimento e no contexto da Estação TABA. Comenta-se, ainda, sobre a implementação realizada da ferramenta SAPIENS.

- **Capítulo V – A Ferramenta SAPIENS:** comenta mais detalhadamente a utilização da ferramenta gerada através de um exemplo de uso dentro do contexto do ambiente *Cordis-FBC*.
- **Capítulo VI – Considerações Finais:** apresenta as considerações finais e trabalhos futuros que podem ser realizados.
- **Referências Bibliográficas:** contém a listagem das referências bibliográficas utilizadas.
- **Anexo I:** apresenta as alterações realizadas no modelo de classes da Estação TABA para que fosse possível descrever a estrutura organizacional.
- **Anexo II:** apresenta a utilizada para representar ontologias utilizando a notação para modelos de classes na UML.

# CAPÍTULO II - GESTÃO DE CONHECIMENTO

## 2.1 Introdução

Conhecimento é indiscutivelmente um importante ativo de uma empresa e, por isso, sua formalização, captura e reutilização devem ser sempre incentivadas. Desenvolvedores de *software* precisam utilizá-lo intensivamente durante o processo de desenvolvimento de *software*. Este conhecimento é acumulado pelos funcionários da companhia e pode ser útil no suporte a projetos de *software* e ao aprendizado organizacional.

Antes que se possa falar sobre os benefícios da gestão de conhecimento para uma organização, é necessário definir o que é considerado como sendo conhecimento. Com o intuito de guiar e auxiliar no entendimento dessa definição, algumas questões podem ser levantadas:

- O que é conhecimento?
- Qual a diferença entre dado, informação e conhecimento?
- Quais os diferentes tipos de conhecimento existentes?
- Qual a diferença entre o conhecimento possuído por uma pessoa e por uma organização?

Após as questões iniciais, outras questões importantes poderão, então, ser respondidas:

- Onde o conhecimento pode ser encontrado?
- Que processos podem ser utilizados para a captura de conhecimento?
- Como o conhecimento pode ser adquirido?
- O que é gestão de conhecimento?
- Como os sistemas de gestão de conhecimento interagem com os usuários?
- Como os fatores culturais são relevantes?
- Que vantagens competitivas o conhecimento pode trazer a uma organização?
- Qual o papel das ontologias nos sistemas de gestão de conhecimento?

Para complementar a discussão, é importante comentar sobre alguns sistemas de gestão de conhecimento, ver que tipos de abordagens são sugeridos na literatura e que tipo de ferramentas têm sido construídas.

O objetivo deste capítulo não é realizar uma discussão completa sobre os tópicos relacionados à gestão de conhecimento, mas apenas fazer uma introdução ao assunto, comentando seus principais aspectos. O presente capítulo está dividido em oito seções:

- **Introdução:** contém a apresentação deste capítulo.
- **Definição de conhecimento:** contém algumas definições e conceitos básicos para o entendimento do assunto;
- **Obtenção do conhecimento:** nesta seção comenta-se sobre como ocorre o processo de aquisição de conhecimento e cita-se algumas fontes de conhecimento dentro de uma organização.
- **Gestão de conhecimento:** define-se o termo gestão de conhecimento e comenta-se sobre alguns aspectos de sua criação e desenvolvimento que devem ser considerados. Discutem-se, também, aspectos culturais que devem ser analisados e levados em consideração.
- **Memória Organizacional:** nessa seção é definido o conceito de memória organizacional e são descritos alguns requisitos desejáveis para sua implementação e manutenção.
- **Ontologias e gestão de conhecimento:** contextualiza o uso de ontologias em sistemas de gestão de conhecimento.
- **Ferramentas para gestão de conhecimento:** apresenta alguns sistemas de gestão de conhecimento, descreve abordagens sugeridas na literatura e algumas ferramentas construídas.
- **Conclusão:** contém um resumo dos tópicos apresentados e sua relevância para o trabalho proposto.

## 2.2 Definindo Conhecimento

A definição é o primeiro passo para que se possa entender algo. Encontrar uma definição amplamente aceita para o termo conhecimento não é uma tarefa fácil. A seguir



tentaremos defini-lo da forma como será tratado nesse trabalho. Convém alertar, no entanto, que não há uma definição amplamente aceita e imune a possíveis críticas (MENDONÇA *et al.*, 2001).

Segundo o dicionário Aurélio (FERREIRA, 1975), “conhecimento é ato ou efeito de conhecer; idéia, noção; informação, notícia, ciência; prática de vida, experiência; discernimento, critério, apreciação; consciência de si mesmo, acordo”. Talvez, no âmbito desse trabalho, as melhores definições apresentadas por esse mesmo dicionário na sua tentativa de definir o que é conhecimento venham da filosofia: “processo pelo qual se determina a relação entre sujeito e objeto; a posição, pelo pensamento, de um objeto como objeto, variando o grau de passividade ou de atividade que se admitam nessa posição; a apropriação do objeto pelo pensamento, como quer que se conceba essa apropriação: como definição, como percepção clara, apreensão completa, análise etc”.

Conhecimento não é algo tangível, concreto. É um conceito abstrato, sua existência, ou a forma como se dá essa existência, depende da pessoa que o detém e/ou o identifica. Não é um objeto que encerra em si mesmo a sua definição. Um conhecimento pode não ser considerado como tal por uma pessoa apesar de sê-lo por outra. Seu significado, função e aplicação dependem da opinião de um observador.

BIGGAM (2001), lista três critérios para que algo seja considerado conhecimento (note que a natureza de suas proposições é altamente subjetiva):

- Deve ser verdadeiro;
- O observador deve acreditar que é verdadeiro;
- O observador deve ser capaz de saber que é verdadeiro.

Algumas vezes, conceitos como conhecimento, dado e informação se confundem e é necessário diferenciá-los para evitar possíveis mal entendidos. As definições abaixo foram retiradas, principalmente, de (CARLINER, 1999) citando (WURMAN, 1989) e (MARKKULA, 1999).

- Dados são fatos que não tem significado próprio, formam um conjunto discreto, objetivo de fatos sobre determinados eventos. Não dizem nada sobre sua própria importância ou irrelevância, mas constituem um material importante para a criação de informação.

- Informação é um dado ao qual se atribuiu ou adicionou um significado. Pode ser descrita como uma mensagem, usualmente na forma de um documento ou alguma forma audível ou visível de comunicação. Informação tem o poder de mudar a forma como se percebe algo, de interferir no seu julgamento e comportamento.
- Conhecimento, por outro lado, é um dado que as pessoas podem aplicar em suas vidas, é informação combinada com experiência, contexto, interpretação e reflexão. É uma forma altamente valiosa de informação e que está pronta para ser aplicada em decisões e ações.

Imagine um conjunto de características que não tenha significado algum sem uma sintaxe apropriada. Ao se incluir uma sintaxe, as características se transformam em dados. Mas em diferentes situações elas poderiam ser vistas como diferentes tipos de informação. Portanto, num determinado contexto os dados se transformam em informação. Agora, se essas informações são (re)unidas e relacionadas uma com as outras, elas se transformam em conhecimento (SNOEK, 1999).

Há, basicamente, dois tipos de conhecimento: tácito e explícito. Conhecimento explícito é um conhecimento codificado em, por exemplo, patentes, diagramas e documentos. Pode ser expresso em palavras e números, é facilmente comunicado e compartilhado na forma de dado físico, fórmulas científicas, procedimentos codificados ou princípios universais. Conhecimento tácito, ao contrário, é altamente pessoal e difícil de ser formulado e compartilhado. É armazenado pela mente, embutido em experiências individuais e envolve fatores intangíveis e subjetivos como valores, perspectivas e considerações pessoais (MARKKULA, 1999) (WINCH, 1999).

Algumas propriedades do conhecimento explícito (WINCH, 1999):

- Pode ser formalizado;
- É possivelmente específico a um determinado contexto, mas não específico a determinados indivíduos;
- Pode ser armazenado, navegado e transferido;
- Pode ser facilmente compartilhado e comercializado;

Algumas propriedades do conhecimento tácito (WINCH, 1999):

- Específico a indivíduos ou grupos;
- Sua disponibilidade depende da existência de indivíduos e grupos que o detém;
- Apenas pode ser comercializado com os indivíduos ou através de contato direto.

Descrever o conhecimento como tácito ou explícito não explica o que constitui conhecimento: apenas ilustra que conhecimento pode ser explicitado ou não. Isso não significa dizer que divisão do conhecimento em tácito e explícito não é prestativa na exploração do conhecimento: pelo contrário, reconhecer que muito do que ocorre numa organização permanece tácito é um estágio importante no desejo de explorar o conhecimento para se tirar vantagem competitiva dele (BIGGAM, 2001).

Distinguir entre conhecimento tácito e explícito é mais que um exercício. É um desafio colocar uma descrição operacional no valor que o conhecimento traz a uma organização (CARLINER, 1999). O conhecimento organizacional e o conhecimento pessoal se confundem na medida que uma organização é composta por pessoas e que o conhecimento de todas essas pessoas é parte fundamental do que se considera conhecimento organizacional.

BIGGAM (2001) nos diz que “o objetivo de muitos negócios é tornar explícitos as habilidades e conhecimentos que permanecem pessoais aos empregados. Conhecimento organizacional é visto como um conhecimento coletivo que ajuda a distinguir uma organização da outra. O conhecimento pessoal envolverá uma combinação de conhecimentos tácitos e explícitos; portanto, o conhecimento organizacional, similarmente, também consistirá de uma coleção do conhecimento tácito e explícito”.

BASILI *et al.* (2001) alertam para uma série de problemas decorrentes de falhas no gerenciamento e relacionados ao conhecimento detido por uma organização. Dentre eles pode-se destacar:

- A saída repentina de um funcionário ocasiona grandes perdas em certas áreas, mas às vezes é até difícil definir que conhecimento foi perdido.

- Um funcionário pode ter aprendido muito durante um projeto, mas não tendo tempo para formalizar e disseminar esse aprendizado, o conhecimento adquirido não poderá ser transferido a outros.
- Uma contratação pode ser considerada um problema e não uma ajuda quando for necessário gastar muito tempo dando suporte ao novo funcionário, prejudicando, assim os demais membros da equipe.
- Um gerente de projeto pode subestimar o prazo de desenvolvimento de algum produto porque não há dados coletados a partir de projetos anteriores.

É preciso, de alguma forma, transformar o conhecimento tácito dos funcionários em explícito, este é um dos grandes desafios das organizações atualmente. Somente dessa forma será possível manter esse conhecimento presente na organização, mesmo após mudanças no quadro funcional. Somente dessa forma será, também, possível difundir determinados conhecimentos por vários outros membros da organização, aumentando, de um modo geral, o conhecimento detido pelas pessoas e, conseqüentemente, pela própria organização. Segundo QINGRUI *et al.* (1999), aprendizado organizacional é a integração entre o conhecimento individual e o processo de 'pensamento sistemático'.

## 2.3 Adquirindo Conhecimento

Antes de comentar o que é e como se dá a gestão de conhecimento é preciso saber como obter este conhecimento, como identificar mecanismos que permitam sua captura, aquisição e reutilização. Paralelamente a isto, é importante identificar as áreas e situações em que pode ser encontrado e áreas e situações onde sua aplicação seja útil e/ou relevante.

Segundo SIVAN (2000), qualquer organização tem um conjunto de objetivos aos quais se atém. Não importam quais sejam esses objetivos eles serão atingidos, ou, pelo menos, perseguidos, através de processos organizacionais. Os processos organizacionais são parecidos entre si, são conjuntos de atividades dentro da organização que terminam com um resultado mensurável e estão lá para guiar as organizações e prover meios com os quais possam atingir seus objetivos.

As principais fontes de conhecimento detidas por uma organização são seus processos internos ou podem ser identificadas ao longo desses processos. Estes processos

são, muitas vezes, intrínsecos à organização e dela não podem ser desassociados. É natural pensar que ao longo da execução desses processos possam ser identificadas as maiores fontes de conhecimento, assim como seus maiores beneficiários. Parece lógico, então, que ao se criar processos relativos à aquisição e disseminação de conhecimento estes sejam inseridos e adaptados aos processos organizacionais já existentes.

Geralmente, há um tipo de conhecimento envolvido nos processos organizacionais que, apesar de possuir pequenas variações, é padrão, existindo em todas as organizações. Ao mesmo tempo, há um tipo de conhecimento que é específico à organização e tem um papel primordial em sua habilidade de manter-se competitiva. Para melhorar os processos e explorar ao máximo seus potenciais, é necessário explorar a transferência de conhecimento que ocorre dentro de cada um deles (SIVAN, 2000).

Os processos organizacionais, apesar de serem uma importante fonte de conhecimento, não são a única. A rigor, tudo o que é produzido, utilizado, manipulado por uma organização, é uma fonte potencial de conhecimento. MURCH (2000) enumera uma série dessas fontes:

- Todos os produtos e serviços da organização;
- Todos os processos, *templates* e planos usados para criar produtos e serviços;
- Dados corporativos em banco de dados, documentos eletrônicos, material arquivado em papel, repositórios, *warehouses*, arquiteturas e arquivos eletrônicos criados pelos empregados;
- Informações sobre vendedores, fornecedores e concorrentes;
- Informações sobre consumidores e clientes;
- Internet e *newsgroups*;
- Alianças, relações estratégicas e dados criados a partir delas;
- Documentos de pesquisa, relatórios, arquivos, patentes e segredos de mercado;
- Todas as arquiteturas e tecnologias corporativas;
- Todos os tipos de dados digitais e não digitais.

Note que todas essas informações têm um ciclo de vida, não surgem espontaneamente: são criadas, atualizadas, consultadas e, posteriormente, arquivadas. Elas geralmente não podem ser consideradas diretamente como conhecimento. Para que se

transformem em conhecimento é necessário um trabalho de identificação, formalização e valoração dos itens relevantes. É natural supor, então, que se possa, ou se deva, definir um processo para gerenciar esse conhecimento.

Os processos para gestão de conhecimento encontrados na literatura têm pequenas variações entre si, mas mantêm uma estrutura básica composta de três fases distintas: aquisição, formalização e reutilização. Abaixo, descrevemos três propostas para um processo de gestão de conhecimento. Note que, apesar da diferente divisão de etapas, todos seguem um princípio comum.

Segundo PROBST *et al.* (1999), citado em (BUERGEL e SAEUBERT, 1999), atividades de gestão de conhecimento podem ser categorizadas em: identificação, aquisição, desenvolvimento, compartilhamento, distribuição, uso e conservação de conhecimento. Ainda segundo BUERGEL e SAEUBERT (1999), estas atividades obviamente representam um potencial de criação de valor.

QINGRUI *et al.* (1999) dividem um processo para gestão de conhecimento em cinco atividades básicas: produção, aprendizado, acumulação, difusão e reativação. Segundo os autores:

- “Aprendizado individual e organizacional são as fontes da acumulação de conhecimento. Aprendizado individual consiste de aprendizado formal e informal, enquanto aprendizado organizacional é executado durante atividades de pesquisa e desenvolvimento e conhecimento ‘extramuro’.”
- “Para o gerenciamento da difusão do conhecimento, os canais e políticas para difundir o conhecimento preservado pela empresa incluem a construção de *intranets*, sistemas de comunicação informal e um sistema de gerenciamento de rede.”
- “A reativação do conhecimento se refere à integração do conhecimento acumulado para emergir novo conhecimento. A interação dos processos inclui a síntese, reestruturação e divulgação do conhecimento.”

STAAB *et al.* (2001), propõem um processo composto por 4 fases:

- **Criação ou importação:** o conteúdo precisa ser criado ou convertido para que possa se adequar às convenções da empresa. Criação de conhecimento acessível eletronicamente se situa, geralmente, entre os extremos da formalidade e da

informalidade. Por vários motivos, a importação de itens de conhecimento possui tão ou mais importância que sua criação.

- **Captura:** itens de conhecimento devem ser capturados para que se possa determinar sua importância e como devem ser incorporados à convenção de vocabulário da empresa.
- **Recuperação e acesso:** este passo satisfaz as necessidades de pesquisas e buscas por conhecimento;
- **Utilização:** conhecimento não será somente criado, mas também reutilizado no futuro. Muito comumente, não é só o conhecimento em si que é de extremo interesse, mas as derivações que podem ser feitas a partir dele.

Segundo QINGRUI *et al.* (1999), o aprendizado e a acumulação de conhecimento são o ponto inicial de um programa de gestão de conhecimento. Eles alertam para o rápido ritmo de envelhecimento e modificação do conhecimento, e que, por isso, o principal enfoque da gestão de conhecimento está em como obter conhecimento que seja valioso e como acumular o conhecimento específico da empresa. Os aprendizados pessoais e organizacionais são importantes fontes de acumulação de conhecimento e, por isso, ainda segundo QINGRUI *et al.* (1999), é dada muito mais ênfase às formas de gerenciamento do aprendizado e conhecimento organizacional e individual.

## 2.4 Gestão de Conhecimento

MARKKULA (1999) alerta que, apesar de haver muitas definições para gestão de conhecimento, ainda não existe uma definição amplamente aceita. MURCH (2000), de certa forma, também concorda ao afirmar que “fala-se muito em gestão de conhecimento mas pouco se entende; há muitas definições e muito alarde sobre o assunto”.

De um modo geral, gestão de conhecimento pode ser definida como uma disciplina que promove uma abordagem integrada visando identificar, capturar, recuperar, compartilhar e avaliar os bens de conhecimento de uma empresa. Estes bens de conhecimento podem incluir base de dados, documentos, políticas e procedimentos além de experiências e conhecimentos tácitos e não capturados detidos pelos trabalhadores (Gartner Group, citado em (MARKKULA, 1999)).

Como é de se esperar, outras definições podem ser encontradas na literatura: umas complementares, outras equivalentes. Dentre as mais importantes estão:

- Para MURCH (2000), gestão de conhecimento é um conjunto de processos, estruturas organizacionais, aplicações e tecnologias que tem por objetivo disponibilizar as habilidades dos empregados, equipes de projeto e unidades de negócio a agir de maneira rápida e eficiente. A gestão de conhecimento alcança esse objetivo provendo acesso imediato ao conhecimento armazenado num repositório ou *knowledge-warehouse*, incluindo muita coisa que é conhecida, porém não documentada. Requer uma abordagem integrada para identificar, gerenciar e compartilhar bens de informação da empresa.
- Daniel E. O'LEARY (1998a), diz que gestão de conhecimento é um processo de conversão de conhecimento de fontes acessíveis a uma organização e conexão desse conhecimento às pessoas. É o gerenciamento formal de conhecimento com o objetivo de facilitar a criação, acesso e reuso do conhecimento, normalmente utilizando tecnologia avançada (O'LEARY, 1998b).
- Segundo WHITEHALL (1998), gestão de conhecimento está preocupada em ativamente e sistematicamente criar, coletar, gerenciar e disponibilizar conhecimento e informação a uma organização. Este conhecimento geralmente está desestruturado, disperso, inconsistente e incompleto. Entretanto, se disponibilizado no contexto apropriado, é relevante para a resolução de problemas referentes ao negócio. Gestão de conhecimento envolve pessoas, conteúdo, processo e tecnologia.

Gestão de conhecimento no sentido de 'manipulação otimizada de conhecimento' é capaz de dar suporte a atividades de criação de valor e, por isso, contribuir indiretamente à criação de valor pela empresa (BUERGEL e SAEUBERT, 1999).

DAVENPORT (1996) lista dez princípios sobre gestão de conhecimento. Segundo ele, quando uma organização decide com quais destes princípios concordar é possível criar planos e abordagens detalhadas a partir deles. A lista é composta por:

- **Gestão de conhecimento é caro** - Conhecimento é um bem, mas gerenciamento efetivo requer investimento de outros bens. Há várias atividades que requerem dinheiro e trabalho, entre as quais pode-se citar captura de



conhecimento, desenvolvimento de infra-estrutura tecnológica e treinamento (DAVENPORT, 1996). O'LEARY e STUDER (2001) alertam que as organizações devem especificar claramente seus objetivos antes de iniciar o desenvolvimento de soluções para gestão de conhecimento.

- **Gerenciamento efetivo de conhecimento requer soluções híbridas envolvendo pessoas e tecnologia** – Quando é necessário entender um conhecimento, interpretá-lo num contexto amplo, combiná-lo com outros tipos de informação ou sintetizar várias formas não estruturadas de conhecimento, pessoas são as ferramentas recomendadas. Por outro lado, para a captura, transformação e distribuição de conhecimento altamente estruturado que muda rapidamente, computadores são mais eficientes que pessoas (DAVENPORT, 1996). O'LEARY e STUDER (2001) reforçam essa idéia ao afirmar que a tecnologia é um fator crucial de sucesso, mas não é o mais importante: devem ser analisados os aspectos culturais e organizacionais para que a solução apresentada seja um sucesso. Alertam, ainda, que aplicações de gestão de conhecimento devem estar inseridas na estrutura da organização de forma que permitam o aprendizado organizacional e devem se basear no envolvimento de todos os funcionários. Para ABECKER *et al.* (1998), as informações devem ser coletadas e organizadas a partir de diversas fontes, os usuários devem ser sempre ouvidos durante a manutenção e evolução dos sistemas e os sistemas devem ser integrados ao ambiente de trabalho já existente.
- **Gestão de conhecimento requer gerentes de conhecimento** – conhecimento não será bem gerenciado até que algum grupo dentro da empresa tenha uma responsabilidade clara pelo trabalho. Dentro das tarefas que deverão ser realizadas estão: a coleta e categorização do conhecimento, estabelecimento de uma infra-estrutura tecnológica orientada a conhecimento e monitoração do uso do conhecimento (DAVENPORT, 1996).
- **Gestão de conhecimento envolve muita política** – Gerentes de conhecimento astutos tenderão a reconhecer e cultivar uma política favorecimento à gestão de conhecimento, farão *lobby* para o seu uso e valoração. Cultivarão formadores de opinião influentes como usuários iniciais de abordagem de gestão de conhecimento, tentarão fazer com que o conhecimento seja utilizado da melhor maneira possível dentro da organização (DAVENPORT, 1996).

- **Gestão de conhecimento se beneficia mais de mapas do que de modelos, mais dos mercados do que de hierarquias** – A dispersão de conhecimento como descrita em mapas de conhecimento pode ser ilógica, mas ainda será mais útil a um usuário que um modelo hipotético de conhecimento que é melhor entendido por seus criadores e raramente completamente implementado. (DAVENPORT, 1996).
- **Compartilhar e utilizar conhecimento não são atos naturais** – entrar com um conhecimento num sistema e procurar por conhecimento de outros não é apenas ameaçador, mas também requer esforço, por isso, as pessoas devem ser altamente motivadas a fazê-lo (DAVENPORT, 1996). ABECKER *et al.* (1998) dizem que deve haver uma distribuição ativa das informações relevantes aos usuários que possam dela se beneficiar. WHITEHALL (1998), de forma semelhante, alerta para a necessidade de mesclar a distribuição passiva e ativa de conhecimento.
- **Gestão de conhecimento significa melhorar processos de trabalho que envolvem conhecimento** – A estrutura básica dos processos varia de acordo com a empresa e tipo de indústria, mas geralmente inclui pesquisa de mercado, projeto e desenvolvimento de produto. Se melhoramentos reais são feitos à gestão de conhecimento, também devem ser feitos aos processos de negócio (DAVENPORT, 1996). O'LEARY e STUDER (2001) sugerem que a gestão de conhecimento deve ser inserida nos processos organizacionais, mesmo que para isso estes processos devam passar por uma reengenharia para se adequar à nova realidade.
- **Acesso ao conhecimento é apenas o começo** – acesso é importante, mas gestão de conhecimento também requer atenção e engajamento. Para que consumidores de conhecimento prestem atenção ao conhecimento, eles precisam se tornar mais que receptores passivos (DAVENPORT, 1996). Mais uma vez, para esse aspecto é necessário cobrar empenho das pessoas envolvidas e criar mecanismos de distribuição ativa do conhecimento.
- **A gestão de conhecimento nunca tem fim** – Uma razão para que gestão de conhecimento nunca termine é que as categorias de conhecimento requeridas mudam constantemente. Novas tecnologias, abordagens, opiniões de consumidores surgem constantemente. Companhias mudam suas estratégias,

estruturas organizacionais e ênfase em serviços e produtos. Novos gerentes e profissionais tem novas necessidades para o conhecimento (DAVENPORT, 1996). Outros motivos para isso são apresentados por O'LEARY e STUDER (2001): mercados altamente voláteis requerem visões sempre atualizadas do comportamento dos consumidores e atividades estratégicas de competidores; estruturas de gestão são responsáveis por tomadas de decisões em todos os níveis organizacionais; prazos curtos para desenvolvimento de produtos requerem alta integração e gerenciamento de conhecimento de diferentes departamentos.

- **Gestão de conhecimento requer um contrato de conhecimento** – Não é claro na maioria das organizações quem possui ou tem direitos de uso ao conhecimento do funcionário, se é o próprio funcionário, a organização, se somente pode ser utilizado pela organização durante o período em que a pessoa estiver empregada etc. Se conhecimento está realmente se tornando um recurso cada vez mais valioso, pode-se esperar uma maior definição dos aspectos legais relacionados (DAVENPORT, 1996).

No aspecto tecnológico, é bom salientar a importância de se utilizar técnicas de engenharia do conhecimento quando uma organização emprega várias abordagens para gestão de conhecimento. Esta tática está se tornando comum em grandes organizações, que já utilizam uma vasta quantidade de sistemas de informação interligados de alguma forma e onde sistemas multifacetados de conhecimento são habituais. Um esforço adicional deve ser feito para que haja uma integração, uma ligação entre as várias fontes de conhecimento, unindo seus conteúdos (PREECE *et al.*, 2001).

Convém lembrar que gestão de conhecimento é multidisciplinar. Envolve recursos humanos, cultura e organização da empresa, assim como tecnologia da informação e ferramentas de apoio. A gestão de conhecimento pode melhorar a competitividade de uma empresa apenas por integrar completamente aspectos tecnológicos com os aspectos organizacionais e humanos (O'LEARY e STUDER, 2001).

A interação entre os sistemas de gestão de conhecimento e seus usuários é um aspecto de extrema importância. Forçar os responsáveis pela criação do conhecimento a usarem uma estrutura formal melhora a usabilidade do conhecimento, mas cerceia o processo de criação. (WHITEHALL, 1998). Sistemas de gestão de conhecimento devem ser

úteis e valiosos tanto para os seus criadores quanto para os utilizadores, mas fazer isso não é trivial (WHITEHALL, 1998). Subordinada ao conceito de gestão de conhecimento está a suposição que pessoas compartilharão espontaneamente de informações com outros. Para iniciar a cadeia de gestão de conhecimento, organizações precisam mudar sua cultura para recompensar as pessoas por proverem conhecimento e desmotivá-las a escondê-lo. É preciso desenvolver o conhecimento tácito entre funcionários, clientes e fornecedores, através de cursos de treinamento e outros tipos de esforços (CARLINER, 1999).

Muitos grupos dentro de uma organização têm algum papel na captura, armazenamento, transformação e disseminação do conhecimento. No centro da questão estão os grupos responsáveis pelos sistemas de informação, que desenvolvem e gerenciam a infra-estrutura tecnológica. Estes grupos estão, geralmente, na vanguarda (CARLINER, 1999).

Segundo HENDRIKS e VRIENS (1999), apesar de ‘conhecimento’ ser a primeira palavra em sistemas baseados em conhecimento, estes dificilmente são considerados a partir de uma perspectiva baseada em conhecimento. Neste mesmo trabalho são apresentados oito ‘domínios de avaliação de valor de conhecimento’ que apontam para classes de atividades de conhecimento que são essenciais quando do projeto do plano estratégico de conhecimento para uma organização:

- Avaliação do conhecimento atual, ou seja, identificação de vários elementos no *portfolio* do conhecimento e determinação de seus *status*;
- Estabelecimento de valor estratégico de elementos do *portfolio*;
- Comparação do *portfolio* com os dos competidores;
- Determinação do conhecimento requerido;
- Criação de novo conhecimento;
- Distribuição e compartilhamento de conhecimento;
- Aplicação de conhecimento;
- Avaliação do conhecimento.

## 2.5 Memória Organizacional

Memória organizacional é uma representação persistente e explícita de informações e conhecimento numa organização. Tem como objetivo facilitar o acesso, compartilhamento e reuso dessas informações e conhecimento pelos membros da organização para atividades individuais ou coletivas (van HEIJST *et al.*, 1996) (DIENG *et al.*, 1999). Uma memória organizacional almeja distribuir o conhecimento correto à pessoa certa, na hora certa e no formato certo para que possibilite a ação correta (DIENG, 2000).

Segundo ACKERMAN e HALVERSON (2000), memória é tanto um artefato que guarda seu estado quanto um artefato que está simultaneamente inserido em muitos processos individuais e organizacionais. Além disso, para a memória ser útil deve carregar consigo alguma marca de autenticidade. Para servir corretamente a seus usuários, o seu criador deve projetar corretamente as consequências de seu uso futuro.

De acordo com ABECKER *et al.* (1998), a principal função de uma memória organizacional é aumentar a competitividade da organização melhorando o modo como ela gerencia o seu conhecimento. Deve dar apoio à criação de conhecimento e aprendizado organizacional. Deve ser mais que um sistema de informação: deve, também, ser capaz de transformar informação em ação.

ABECKER *et al.* (1998) citam cinco requisitos que consideram cruciais para o sucesso de uma prática industrial de memória organizacional:

- Coleta e organização sistemática de informação de várias fontes;
- Minimização de soluções imediatistas;
- Exploração do *feedback* dos usuários visando manutenção e evolução;
- Integração com o ambiente de trabalho; e
- Apresentação ativa de informação relevante.

Para uma memória organizacional ser efetiva, usuários devem receber informação relevante no tempo certo sem serem sobrecarregados com uma grande quantidade de dados irrelevantes. Informação é relevante apenas se os usuários podem fazer suas tarefas melhor com ela do que sem ela. Por isso, a relevância da informação é sempre definida com respeito a esse uso (ABECKER *et al.*, 1998).

Empresas podem ter sucesso no desenvolvimento e manutenção de uma memória organizacional se seguirem os seguintes princípios (ABECKER *et al.*, 1998):

- Explorar a facilidade de fontes de informação disponíveis;
- Não se prender a uma tentativa de formalizar completamente o conhecimento;
- Usar ferramentas para aquisição automatizada de conhecimento;
- Encorajar *feedback* e sugestões dos usuários para melhorias; e
- Certificar-se da consistência de novos conhecimentos sugeridos e/ou adicionados.

## 2.6 Ontologias e Gestão de Conhecimento

Ontologia é uma representação de vocabulário, geralmente especializada para algum domínio ou assunto. Não é o vocabulário que se qualifica como uma ontologia, mas a conceitualização que seus termos no vocabulário tentam capturar. O termo ontologia algumas vezes é utilizado para se referir a um corpo de conhecimento, tipicamente um senso comum sobre um determinado domínio, utilizando um vocabulário como representação. Em outras palavras, a representação textual provê um conjunto de termos com os quais são descritos os fatos em algum domínio, enquanto o corpo do conhecimento utilizando aquele vocabulário é uma coleção de fatos sobre um domínio (CHANDRASEKARAN *et al.*, 1999).

Ontologias e bases de conhecimento estão intimamente relacionadas em gestão de conhecimento. Ontologias definem as características e visões da base de conhecimento e também empregam modelos que são úteis na definição e acesso a bases de conhecimento (O'LEARY, 1998b).

Uma ontologia provê a estrutura básica ou esqueleto sobre a qual uma base de conhecimento pode ser construída. Uma ontologia provê um conjunto de conceitos e termos para descrever algum domínio, enquanto uma base de conhecimento utiliza aqueles termos para representar o que é verdade sobre algum mundo real ou hipotético (SWARTOUT, 1999). Bases de conhecimento utilizam ontologias para especificação de visões e estruturas não ambíguas (O'LEARY, 1998a). Atualmente, um dos grandes impedimentos para o compartilhamento do conhecimento é a utilização de diferentes

termos e conceitos para descrever domínios por diferentes sistemas. Se for possível desenvolver ontologias que possam ser utilizadas por múltiplos sistemas, eles compartilhariam uma terminologia comum que facilitaria o compartilhamento e reuso. O'LEARY (2000) alerta para o fato que diferentes empresas terão diferentes ontologias para suas bases de conhecimento, mas se o custo de construir sua própria ontologia for maior que o benefício de utilizar uma já existente, estas empresas tenderão a utilizar uma ontologia padrão.

Ontologias são particularmente úteis para recuperação e acesso a conhecimento. STAAB *et al.* (2001) diz que ontologias podem ser utilizadas para derivar futuras visões do conhecimento. Em particular, pode-se explorar a ontologia para auxiliar a navegação. Pode-se explorar o que está presente na memória coletiva da organização sem necessitar fazer uma pergunta específica – o que pode ser uma tarefa complicada para iniciantes. A utilização de uma ontologia permite a derivação de relações entre os itens desta memória e permite, também, a derivação descrições adicionais.

O'LEARY (1998b) cita cinco exemplos de utilização de ontologia em sistemas de gestão de conhecimento:

- Quando sistemas de gestão de conhecimento utilizam grupos de discussão, os usuários devem ser capazes de identificar quais grupos são de interesse. Ontologias servem para definir o escopo desses grupos, sendo utilizadas para criar tópicos que os distinguirão.
- Quando sistemas de gestão de conhecimento dispõem de mecanismos de busca, para prover um nível apropriado de precisão, sistemas baseados em conhecimento precisam determinar de forma não ambígua que tópicos podem ser encontrados em cada base de conhecimento.
- Para utilizar um sistema de filtragem de informação, usuários geralmente devem especificar palavras-chave ou conceitos que capturem a natureza do conhecimento desejado.
- Ontologias facilitam a reutilização de artefatos arquivados em sistemas de gestão de conhecimento, pois podem ser utilizadas para categorizá-los, facilitando sua posterior identificação.

- Sem um apropriado conjunto de ontologias, a falta de uma linguagem comum pode causar confusão em sistemas que provêem oportunidades de colaboração e utilização de conhecimento detido por especialistas.

## 2.7 Ferramentas para Gestão de Conhecimento

Gestão de conhecimento é, certamente, uma disciplina muito discutida, estudada e valorizada atualmente. Com isto, é possível encontrar grande número de relatos sobre vários trabalhos relacionados a gestão de conhecimento e aprendizado organizacional nas publicações especializadas. Uma revisão da literatura sobre o assunto não estaria completa se não houvesse uma seção com alguns exemplos de sistemas para gestão de conhecimento, abordagens e ferramentas propostas e/ou construídas.

As abordagens são as mais variadas possíveis. Pode-se citar, dentre outras, ontologias (van ELST e ABECKER, 2001) (MAMANI-MACEDO e LEITE, 2001), simulação de jogos (OH e van der HOEK, 2001) e gerenciamento de experiências (MENDONÇA *et al.*, 2001). Também fazem parte das experiências relatadas a criação de sistemas de gerenciamento de memória organizacional com integração de vários tipos de informação e conhecimento, proporcionando mecanismos inteligentes para acesso ao conhecimento (von WANGENHEIM *et al.*, 2001).

A seguir, faz-se um breve resumo de trabalhos encontrados na literatura.

Fábrica de Experiências (BASILI *et al.*, 2001) é um conceito proposto por Basili na década de 80 e desde então têm sido aplicado e aprimorado através de vários trabalhos por diferentes pesquisadores, por exemplo (HOUDEK e SCHNEIDER, 1999) e (RUHE, 1999). Fábrica de Experiências (MENDONÇA *et al.*, 2001) é uma infra-estrutura organizacional que tem por objetivo produzir, armazenar e reutilizar experiências adquiridas em organizações que desenvolvem *software*. Uma empresa de desenvolvimento de *software* é organizada na forma de duas organizações distintas, uma com foco na produção do produto de *software* e outra com foco no aprendizado a partir de experiências e na melhoria do desenvolvimento de *software*. Estas divisões interagem entre si para dar suporte a seus objetivos. Fábricas de Experiências reconhecem que a melhora dos processos e produtos de *software* requerem (MENDONÇA *et al.*, 2001):



- uma contínua acumulação de experiências avaliadas e sintetizadas em pacotes de experiências;
- armazenamento de pacotes de experiências numa base integrada e acessível por diferentes partes da organização; e
- criação de perspectivas pelas quais diferentes partes da organização podem olhar na mesma base de experiências de diferentes modos.

Von WANGENHEIM, *et al.* (2001) descrevem uma abordagem híbrida para uma infra-estrutura técnica em gestão de conhecimento para organizações de pesquisa e desenvolvimento. A abordagem integra vários tipos de informação e conhecimento e prevê mecanismos inteligentes para acesso ao conhecimento, assim como mecanismo para a contínua evolução e melhoria do sistema de gerência de memória organizacional. O sistema integra técnicas de diversas áreas, como raciocínio baseado em casos (*Case-Based Reasoning*), recuperação de informação e processamento de linguagem natural numa única infra-estrutura criando um assistente inteligente para o desenvolvimento e pesquisa em *software*. A abordagem também integra tecnologia de diferentes áreas, incluindo bases de dados, hipertexto, sistemas de e-mail, recuperação de informação e raciocínio baseado em casos. Técnicas de raciocínio baseado em casos são utilizadas para representação do conhecimento, recuperação baseada em similaridade e aprendizado incremental. Técnicas de recuperação de informação e processamento de linguagem natural são a base para a extração de informação a partir de consultas em linguagem natural e documentos, assim como a evolução do conhecimento geral do domínio. Hipertextos são utilizados para permitir a exploração interativa entre os itens. Técnicas de filtragem de informação são utilizadas para recomendar itens a potenciais usuários interessados.

ABECKER *et al.* (1999) descrevem três projetos que serviram de base para o projeto FRODO:

- **KnowMore:** a idéia principal é prover múltiplas fontes de conhecimento heterogêneo através de uma descrição compreensível de conhecimento baseada numa ontologia informal de informação. Esta ontologia de informação contém noções das ontologias de organização e domínio de aplicação da empresa. O KnowMore faz distribuição pró-ativa de informação integrada a seu respectivo processo de negócio através de uma máquina de *workflow*.

- **ENRICH:** o foco principal deste projeto está na gestão de conhecimento relacionada a aspectos de *groupware*. Está baseado na idéia de que conhecimento pode ser considerado como informação ligada a algum modelo formal.
- **Know-Net:** tem por objetivo desenvolver uma ferramenta de *software*, integrada à *intranet*, que integre funcionalidades de *groupware* com métodos de inteligência artificial que permitam a manipulação de objetos de conhecimento.

Van ELST e ABECKER (2001) discutem três dimensões básicas da informação que têm impacto fundamental na utilidade de ontologias para a memória organizacional: formalismo, estabilidade e compartilhamento da informação. Contém, também, uma breve discussão de técnicas que são utilizadas para encontrar um equilíbrio no qual cada uma dessas dimensões leva à caracterização de papéis dos atores relacionados a ontologias num cenário de memória organizacional. São descritos vários papéis com respeito a seus objetivos, conhecimento, competências, direitos e obrigações. Estas idéias são utilizadas no decorrer da realização do projeto FRODO, citado no parágrafo anterior.

OH e van der HOCK (2001) descrevem uma abordagem de simulação para educação de desenvolvimento de *software* baseada em tecnologia de jogos de simulação (nestes jogos os jogadores vivem suas vidas num mundo virtual e tem de trabalhar almejando atingir certos objetivos, algumas vezes conflitantes). Segundo os autores, esta abordagem permite a indivíduos desenvolver um entendimento dos processos de *software* utilizados em suas organizações, enquanto as organizações, como um todo, se beneficiam da habilidade de explorar diferentes abordagens nos seus processos de desenvolvimento de *software*. Acrescentam, ainda, que ambientes de simulação para desenvolvimento de *software* baseados em jogos podem efetivamente facilitar o aprendizado do processo de desenvolvimento de *software* em ambos os níveis. Indivíduos podem usar tal ambiente para criar um entendimento de várias dinâmicas que envolvem o processo de desenvolvimento de *software* próprio de sua organização e o desenvolvimento de *software* em geral. Por outro lado, organizações podem usar este ambiente para prover uma visão geral de seus processos, permitindo exames críticos para descobrir possíveis áreas para melhoria e caminhos específicos através dos quais tais melhorias podem ser feitas.

Mais alguns exemplos de sistemas de gestão de conhecimento serão apresentados na seção 4 do capítulo IV.

## 2.8 Conclusão

Neste capítulo foi apresentada uma revisão da literatura sobre gestão de conhecimento. Termos e conceitos a serem utilizados nos próximos capítulos foram definidos e inter-relacionados. Estes conceitos incluem, mas não são limitados a, *conhecimento*, *gestão de conhecimento*, *memória organizacional* e *ontologias*. Discutiu-se formas de obtenção de conhecimento e o papel das ontologias nos sistemas de gestão de conhecimento. Por fim, alguns sistemas de gerência de conhecimento descritos na literatura foram apresentados.

Estes termos e conceitos são particularmente importantes para situar o trabalho aqui apresentado dentro da realidade dos sistemas de gerência de conhecimento e para o perfeito entendimento dos capítulos seguintes.

No próximo capítulo serão descritos os conceitos relacionados a Ambientes de Desenvolvimento de *Software*, à sua evolução para Ambientes de Desenvolvimento de *Software* Orientados a Domínio e, posteriormente, para Ambientes de Desenvolvimento de *Software* Orientados a Organização. Por fim, discute-se a Estação TABA, ambiente de desenvolvimento de *software* no qual este trabalho está inserido.

# CAPÍTULO III - AMBIENTES DE DESENVOLVIMENTO DE *SOFTWARE* E A ESTAÇÃO TABA

## 3.1 Introdução

Ambientes de Desenvolvimento de *Software* (ADS) são importantes ferramentas no auxílio e ao longo de todo o processo de desenvolvimento de *software*. Provêem suporte automatizado a diversas atividades como codificação, gerenciamento, controle, reparo, melhorias etc.

O conceito de ADS surgiu na década de 70 e tem evoluído desde então. A COPPE possui um grupo de trabalho em ambientes de desenvolvimento de *software* responsável pela definição e criação da Estação TABA (ROCHA *et al.*, 1990). Este projeto surgiu no início da década de 90 e consiste de um meta-ambiente de desenvolvimento de *software* capaz de gerar, através de instanciação, outros ADS. Ao longo desses anos de trabalho o conceito de ADS evoluiu para a definição de ADS com suporte à utilização de informações sobre o conhecimento do domínio de aplicação durante o desenvolvimento e, mais recentemente, para a utilização de conhecimento organizacional. Para acompanhar essa evolução, a Estação TABA passou a instanciar, também, ambientes orientados a domínio e ambientes orientados a organização.

Para o perfeito entendimento do assunto é preciso responder algumas perguntas:

- O que são e como surgiram os ambientes de desenvolvimento de *software*?
- Como e por que os ADS evoluíram para incluir informações sobre o conhecimento do domínio?
- Como e por que os ADS orientados a domínio evoluíram para incluir informações sobre o conhecimento organizacional?
- Como a estrutura e requisitos da Estação TABA foram definidos inicialmente?
- Como foi a evolução da Estação TABA para gerar ADS orientados a domínio e orientados a organização?

Por fim, discutem-se brevemente detalhes da implementação da Estação TABA e algumas das ferramentas que a compõem.

O objetivo deste capítulo é situar o leitor nos conceitos envolvidos no desenvolvimento da Estação TABA, desde a sua definição até o momento atual, e comentar brevemente sobre sua estrutura. O presente capítulo está dividido em sete seções:

- **Introdução:** contém a apresentação deste capítulo.
- **Ambientes de Desenvolvimento de *Software*:** comenta sobre ambientes de desenvolvimento de *software*, origens e objetivos;
- **A Estação TABA:** apresenta o objetivo, as características e os requisitos da Estação TABA;
- **Ambientes de Desenvolvimento de *Software* orientados a domínio:** comenta sobre a evolução dos ambientes de desenvolvimento de *software* com a utilização de conhecimento sobre o domínio durante o processo de desenvolvimento;
- **Ambientes de Desenvolvimento de *Software* orientados a organização:** apresenta a evolução dos ambientes de desenvolvimento de *software* com a inclusão de conhecimento organizacional;
- **Modelo Atual e Implementação:** comenta a implementação da Estação TABA e algumas de suas ferramentas já implementadas por ocasião da definição dos ADSOD e dos ADSOrg.
- **Conclusão:** contém um pequeno resumo dos tópicos apresentados e sua relevância para o trabalho proposto. Contém, também, a descrição dos tópicos a serem discutidos no próximo capítulo.

### 3.2 Ambientes de Desenvolvimento de *Software*

Um Ambiente de Desenvolvimento de *Software* é definido como sendo um sistema computacional que provê suporte para o desenvolvimento, reparo e melhorias em *software* e para o gerenciamento e controle dessas atividades (MOURA e ROCHA, 1992). Para tal, o ADS contém um repositório com todas as informações relacionadas com o projeto ao longo do seu ciclo de vida. Além disso, possui ferramentas que oferecem apoio às várias

atividades técnicas e gerenciais passíveis de automação que devem ser realizadas no projeto. TRAVASSOS (1994) enfatiza que o ADS deve se preocupar com o apoio às atividades individuais e ao trabalho em grupo, o gerenciamento de projeto, o aumento da qualidade geral dos produtos e o aumento da produtividade, permitindo ao engenheiro de *software* acompanhar o projeto e medir a evolução dos trabalhos através de informações obtidas ao longo do desenvolvimento.

Com a definição de ADS e a preocupação de que *software* não deve ser desenvolvido de forma *ad hoc*, surgiram pesquisas em ambientes centrados em processo (GARG e JAZAYERI, 1995), onde se defende a idéia de ferramentas integradas e incorporadas a um processo de desenvolvimento de *software* específico. Um processo de desenvolvimento de *software* é um conjunto bem definido e ordenado de atividades, somado aos recursos utilizados e produzidos e ao conjunto de ferramentas e técnicas para apoio à realização destas atividades (PFLEEGER, 2001).

ADS diferem um dos outros de acordo com a natureza de seu repositório, escopo das ferramentas fornecidas e/ou da tecnologia adotada. Pesquisas na área têm explorado diferentes aspectos em relação às ferramentas de apoio. Atualmente, o maior desafio está na construção e integração de ferramentas que possam ser adaptadas para uso em novos contextos e de ferramentas para dar suporte ao processo de adaptação e integração.

### 3.3 A Estação Taba

A Estação TABA (ROCHA *et al.*, 1990) é um meta-ambiente capaz de gerar, através de instanciação, ambientes de desenvolvimento de *software* adequados às particularidades de processos de desenvolvimento e de projetos específicos. ROCHA *et al.* (1990) definem meta-ambiente como um ambiente que abriga um conjunto de programas que interagem com os usuários para definir interfaces, selecionar ferramentas e estabelecer os tipos de objetos que irão compor o ambiente de desenvolvimento específico.

O projeto TABA foi criado no início da década de 90, a partir da constatação de que domínios de aplicação diferentes possuem características distintas e que estas devem incidir nos ambientes de desenvolvimento através dos quais os engenheiros de *software* desenvolvem aplicações. Desta forma, a Estação TABA tem por objetivo auxiliar na definição, implementação e execução de ADS adequados a contextos específicos.

Com o intuito de atender a este objetivo, quatro funções foram definidas originalmente para a Estação TABA (TRAVASSOS, 1994):

- Auxiliar o engenheiro de *software* na especificação e instanciação do ambiente mais adequado ao desenvolvimento de um produto específico a partir do processo de *software* e/ou de uma definição do domínio de aplicação;
- Auxiliar o engenheiro de *software* na implementação das ferramentas necessárias ao ambiente definido;
- Permitir aos desenvolvedores do produto de *software* a utilização da estação através do ambiente desenvolvido;
- Permitir a execução do *software* na estação configurada para o seu desenvolvimento.

Na Estação TABA, um ADS é instanciado a partir da definição de um processo de desenvolvimento. Dois ambientes foram definidos e implementados de acordo com essa concepção: o ambiente Orixás (WERNECK, 1995), criado para apoiar o desenvolvimento de Sistemas Baseados em Conhecimento e o ambiente Memphis (WERNER *et al.*, 1997), criado para apoiar o desenvolvimento de *software* baseado em reutilização.

### **3.4 Ambientes de Desenvolvimento de *Software* Orientados a Domínio**

Um grande transtorno no desenvolvimento de *software* é o entendimento errôneo ou incompleto do problema que o *software* pretende resolver. Isto pode levar a produtos de *software* implementados corretamente, mas que resolvem os problemas errados. Um agravante desta situação, segundo FISCHER (1996), é o fato de problemas complexos exigirem mais conhecimento do que uma só pessoa pode possuir, o que faz com que a comunicação e a colaboração tornem-se fundamentais. A idéia central passou a ser, então, a de um entendimento compartilhado: os especialistas entendem a prática e os projetistas de sistemas conhecem a tecnologia. Tal compartilhamento precisa ocorrer durante todo o processo de desenvolvimento, pois novos requisitos surgem durante o desenvolvimento porque não podem ser identificados até que porções do sistema tenham sido projetadas e implementadas.

Com a intenção de facilitar a comunicação e, conseqüentemente, o entendimento do problema, surgiram, no contexto da Estação TABA, os Ambientes de Desenvolvimento de *Software* Orientados a Domínio (ADSOD), que buscam integrar o conhecimento do domínio aos Ambientes de Desenvolvimento de *Software* (OLIVEIRA *et al.*, 1999a).

ADSOD podem ser vistos como uma evolução dos ambientes de desenvolvimento de *software* tradicionais. Os ADSOD apóiam o desenvolvimento de *software* em domínios específicos através do uso do conhecimento deste domínio durante todo o processo de desenvolvimento para auxiliar o desenvolvedor no entendimento do problema (OLIVEIRA *et al.*, 1999b e 1999c). O domínio é uma área de aplicação na qual vários produtos de *software* serão desenvolvidos.

Desta forma, o aspecto central dos ADSOD é a introdução e uso do conhecimento do domínio no ADS, provendo apoio aos desenvolvedores na identificação correta dos requisitos do sistema e sua descrição (OLIVEIRA *et al.*, 1999b). Como conseqüência, espera-se obter um aumento de produtividade no desenvolvimento. O uso do conhecimento do domínio durante o desenvolvimento de *software* tende a tornar o processo de entendimento do problema mais fácil e agradável tanto para os desenvolvedores quanto para os especialistas do domínio que, muitas vezes, não são profissionais da área de informática (FISCHER, 1994).

Três ADSOD já foram instanciados utilizando essa infra-estrutura: **CORDIS** (OLIVEIRA *et al.*, 2000), **NETUNO** (GALOTTA, 2000) e **INSECTA** (FOURO, 2002). **CORDIS**<sup>1</sup> é um ADSOD para o domínio de Cardiologia e foi desenvolvido para ser utilizado na Unidade de Cardiologia e Cirurgia Cardiovascular/Fundação Bahiana de Cardiologia. **NETUNO** é um ADSOD para Acústica Submarina, desenvolvido de acordo com as particularidades do GAS/IPqM (Grupo de Acústica Submarina do Instituto de Pesquisas da Marinha). **INSECTA** é um ADSOD para o domínio de entomologia, desenvolvido para a EMBRAPA.

---

<sup>1</sup> Mais informações sobre o **CORDIS** podem ser encontradas na seção 5.2.



### 3.5 Ambientes de Desenvolvimento de *Software* Orientados a Organização

Para melhorar a maneira como as organizações desenvolvem e mantêm *software*, é fundamental melhorar a maneira como elas administram o conhecimento requerido para a realização desta atividade. Desenvolvedores de *software* precisam que todo o conhecimento relevante para a realização de suas atividades esteja facilmente disponível, o que inclui conhecimento sobre o domínio, diretrizes e melhores práticas organizacionais, técnicas e métodos de desenvolvimento de *software*, experiências anteriores com o uso destas técnicas e métodos e também com o processo de *software* (VILLELA *et al.*, 2000).

ADSOD (OLIVEIRA, 1999) (OLIVEIRA *et al.*, 2000) buscam assegurar a disponibilidade e o entendimento de conhecimento sobre domínios de aplicação. No entanto, o conhecimento necessário, para um esforço de engenharia de *software* eficiente, extrapola o conhecimento sobre o domínio, envolvendo normas e diretrizes organizacionais, melhores práticas, relatos de experiências, entre outros. Reconhecendo que as organizações que desenvolvem e mantêm *software* lidam de forma intensa com diversos tipos de conhecimento, os Ambientes de Desenvolvimento de Software Orientados a Organização (ADSOrg) representam uma evolução dos ADSOD. Têm por objetivo apoiar o gerenciamento completo do conhecimento requerido em uma atividade de engenharia de *software*, evitando que este conhecimento fique disperso ao longo da estrutura organizacional e, conseqüentemente, sujeito a dificuldades de acesso e, mesmo, a perdas (VILLELA *et al.*, 2000).

A motivação para a construção de ADSOrg surgiu de duas constatações (VILLELA *et al.*, 2000):

- duas ou mais organizações podem desenvolver *software* para um mesmo domínio com processos, interesses e características muito distintas, e,
- o conhecimento do domínio não é o único conhecimento importante para apoiar desenvolvedores de *software* em suas atividades. Outros conhecimentos também são extremamente importantes e úteis para os desenvolvedores, como: diretrizes e melhores práticas organizacionais, lições aprendidas com o uso de processos, métodos e técnicas de *software* etc.

ADSOrg possuem os seguintes objetivos (VILLELA *et al.*, 2000):

- prover para os desenvolvedores de *software* o conhecimento acumulado pela organização e relevante no contexto do desenvolvimento de *software*; e
- apoiar o aprendizado organizacional neste contexto.

As atividades realizadas dentro de uma organização podem ser classificadas em atividades de negócio e atividades de suporte ao negócio. As atividades de negócio são as diretamente relacionadas à finalidade organizacional, enquanto as outras buscam fornecer as condições necessárias para a realização das atividades de negócio (VILLELA *et al.*, 2000). Como o ADSOrg pretende apoiar a atividade de engenharia de *software* em uma organização e esta atividade possui diferentes características a depender do tipo de organização em que é realizada, foi verificada a necessidade de definir dois tipos de ADSOrg: um para Organizações de *Software* (por exemplo, empresas de serviços e consultoria em Informática) e outro para Organizações com outro tipo de negócio (por exemplo, empresas de telecomunicações, hospitais, bancos, seguradoras etc) (VILLELA *et al.*, 2000).

Baseado nestas constatações, as funções originais da Estação TABA foram revistas e ampliadas<sup>2</sup>. Suas funções atuais são:

- (i) Auxiliar o engenheiro de *software* na configuração do ambiente mais adequado para apoiar o desenvolvimento e a manutenção de *software* em uma organização específica (Ambiente Configurado), considerando seu processo de *software* e a gerência do conhecimento organizacional relevante neste contexto;
- (ii) Auxiliar o engenheiro de *software* na instanciação de ambientes de desenvolvimento de *software* para projetos específicos (caso a configuração de um ambiente para organização não seja possível ou considerada adequada);
- (iii) Auxiliar os gerentes de projeto na instanciação de ambientes de desenvolvimento de *software* para projetos específicos a partir de um Ambiente Configurado;

---

<sup>2</sup> Novos requisitos também foram definidos devido à definição dos ADSOrg. Os diretamente relacionados a esta tese podem ser encontrados na seção 4 do capítulo III.

- (iv) Auxiliar o engenheiro de *software* de empresas cujo negócio é o desenvolvimento e a manutenção de *software* para diversos clientes na especialização de processos da sua empresa de acordo com as particularidades de um cliente específico;
- (v) Auxiliar o engenheiro de *software* a implementar ferramentas necessárias aos ambientes;
- (vi) Apoiar, através dos ambientes instanciados, o desenvolvimento e a manutenção de *software*, bem como a gerência destas atividades;
- (vii) Permitir a execução do *software* na própria Estação, pelo menos para fins de teste.

OLIVEIRA (1999) descreve o **CORDIS**, um ADSOD para o domínio de cardiologia, instanciado a partir da Estação TABA, para uso na Fundação Bahiana de Cardiologia. Este ambiente já incorpora algumas características específicas da organização, por conter o processo padrão definido para ela. Como um ADSOrg envolve mais conhecimento sobre a organização do que o processo padrão, foi criado o **CORDIS-FBC**, um ambiente orientado à organização evoluído a partir do **CORDIS** (VILLELA *et al.*, 2000)<sup>3</sup>.

### 3.6 Implementação Atual

A implementação da Estação TABA foi iniciada em 1994 a partir do trabalho de TRAVASSOS (1994). No início, foi desenvolvida utilizando a linguagem Eiffel numa estação de trabalho da Sun Microsystems<sup>®</sup>. Com o passar do tempo, percebeu-se que o poder e a robustez da plataforma, apesar de adequadas ao ambiente de pesquisas, dificultava a experimentação das idéias em ambientes reais devido à falta de portabilidade do código para plataformas mais acessíveis e comumente utilizadas. Com o intuito de solucionar este problema, foi realizada uma re-implementação onde optou-se pela plataforma de microcomputadores e pelo uso da linguagem C++ (OLIVEIRA, 1999) (SANTOS e ZLOT, 1999).

---

<sup>3</sup> Mais informações sobre o **CORDIS** e o **Cordis-FBC** podem ser encontradas, respectivamente, nas seções 5.2 e 5.3.

Com a re-implementação, o modelo de dados foi revisto e avaliado de forma a limitar o trabalho de implementação a funcionalidades básicas, suficientes para a especificação, instanciação e execução de ambientes de desenvolvimento de *software*. A esse subconjunto foram acrescentadas novas características necessárias à definição e instanciação de ADSOD (OLIVEIRA, 1999). Algumas das ferramentas já definidas e construídas na versão anterior foram reimplementadas, como, por exemplo, **AssistPro** (FALBO, 1998) (FALBO *et al.*, 1999), outras ainda dependem de um novo esforço de implementação, como, por exemplo, a máquina de processos (ARAÚJO, 1998) e metagerador de ferramentas (FATEEV, 1998). Com a criação dos ADSOrg, novamente o modelo foi revisto para adequá-lo ao novo esquema de instanciação de ambientes e para permitir a definição de novas ferramentas.

Atualmente a Estação TABA possui ferramentas para: definição e instanciação de ADS e ADSOD (OLIVEIRA, 1999) (SANTOS e ZLOT, 1999); uma ferramenta para atender a configuração de ambientes para organizações (**Config**); definição, especialização e instanciação de processos (**AssistPro** (FALBO *et al.*, 1999) (FALBO, 1998), **EditPro** (SANTOS e ZLOT, 1999), **DefPro** (MACHADO, 2000) (MACHADO *et al.*, 2000a e 2000b)); definição de teorias de domínio e tarefas (**EDITED** (OLIVEIRA *et al.*, 2000), **EDITAR** (OLIVEIRA, 1999) (ZLOT, 2002) (ZLOT *et al.*, 2002)); apoio à atividade de Investigação do Domínio (**GENESIS** (GALOTTA, 2000), **NAVEGUE** (GALOTTA, 2000) e **REGCON** (GALOTTA, 2000)); avaliação de artefatos e de processos de *software* (GOMES, 2001); apoio ao planejamento do projeto (**RiscPlan** (FARIAS *et al.*, 2001) (FARIAS, 2002), **CustPlan** (BARCELLOS *et al.*, 2001) e **RHPlan**); descrição do conhecimento organizacional (VILLELA *et al.*, 2001a e 2001c) (SANTOS *et al.*, 2002); aquisição de conhecimento (MONTONI, 2002).

### 3.7 Conclusão

Neste capítulo foram apresentados os conceitos de Ambientes de Desenvolvimento de *Software*, Ambientes de Desenvolvimento de *Software* Orientados a Domínio e Ambientes de Desenvolvimento de *Software* Orientados a Organização. Foram descritos, também, a infra-estrutura, características e requisitos da Estação TABA e sua evolução para a instanciação de ambientes orientados a domínio e orientados a organização. Com isso,

encerra-se a revisão de literatura e a contextualização com os tópicos e conceitos que serão utilizados no decorrer dos capítulos seguintes.

No próximo capítulo, será descrito o trabalho realizado nesta tese: uma ferramenta para a representação da distribuição de conhecimento, habilidades e experiências através da estrutura organizacional. Este trabalho utiliza conceitos de gestão de conhecimento (discutidos no capítulo II) e está inserido no contexto de criação de Ambientes de Desenvolvimento de *Software* Orientados a Organização (discutido neste capítulo).

# CAPÍTULO IV - REPRESENTAÇÃO DA DISTRIBUIÇÃO DO CONHECIMENTO, HABILIDADES E EXPERIÊNCIAS ATRAVÉS DA ESTRUTURA ORGANIZACIONAL

## 4.1 Introdução

O desenvolvimento de um *software* envolve muitas variáveis, o que torna a sua realização crítica em algumas situações. Muitas vezes, decisões sobre alocação de pessoal precisam ser tomadas com extrema agilidade, assim como é necessário descobrir com rapidez quem consultar sobre determinado assunto referente à atividade que está sendo realizada no momento. Ter uma base de conhecimento atualizada e sempre à disposição sobre a organização é um fator determinante para que seja possível tomar decisões desse tipo da maneira mais correta possível.

Discutiu-se nos capítulos anteriores sobre gestão de conhecimento, ambientes de desenvolvimento de *software* e Estação TABA. Agora que parte da fundamentação teórica envolvida nesta tese já foi apresentada, é possível discutir as idéias propostas neste trabalho. Para o seu perfeito entendimento, cinco perguntas serão respondidas:

- Qual a motivação do trabalho apresentado?
- Qual a fundamentação teórica específica, ou seja, os aspectos mais relevantes da gestão de conhecimento e dos ADSOrg, que foi utilizada?
- Que abordagens semelhantes foram apresentadas na literatura?
- Quais dos requisitos definidos para a criação dos ADSOrgs foram relevantes para a definição desta proposta?
- Quais características uma ferramenta implementada a partir dessa proposta deveria possuir?

Detalhes técnicos da implementação não serão discutidos neste capítulo. No próximo capítulo, esta discussão poderá ser encontrada, juntamente com a descrição

detalhada de um exemplo de uso da ferramenta desenvolvida no contexto do ambiente Cordis-FBC.

Este capítulo apresenta a motivação teórica deste trabalho, situa-o no contexto de gestão de conhecimento e da Estação TABA e descreve as características desejadas para uma ferramenta que pretenda cumprir as idéias propostas. O presente capítulo está dividido em cinco seções:

- **Introdução:** contém a apresentação deste capítulo.
- **Abordagens Relacionadas na Literatura:** Apresenta ferramentas e soluções semelhantes descritas na literatura.
- **Fundamentação:** apresenta a fundamentação teórica por trás da abordagem proposta no contexto da Estação TABA, Ambientes de Desenvolvimento de *Software* Orientados a Organização e de Gestão de Conhecimento;
- **Características da Ferramenta:** descreve as principais características da ferramenta que foi construída em decorrência dessa tese. Contextualiza essas características com os requisitos desejados para os Ambientes de Desenvolvimento de *Software* Orientados a Organização e com algumas características desejadas a sistemas que lidem com Gestão de Conhecimento;
- **Conclusão:** contém uma breve síntese do capítulo e a descrição dos tópicos a serem discutidos no próximo capítulo.

## 4.2 Abordagens Relacionadas na Literatura

Vários trabalhos relacionados à gestão de conhecimento citam a necessidade de ter uma forma fácil de encontrar as pessoas envolvidas no desenvolvimento de *software*, similar a um sistema de páginas amarelas.

- PREECE *et al.* (2001) citam como exemplo de sistemas para suporte a atividades de gestão de conhecimento sistemas para gerenciamento de capacidades. Estes sistemas definem um meta-conhecimento que servirá como uma diretriz para capturar as capacidades dos trabalhadores e será utilizado para povoar uma base de dados de currículos. Segundo STADER e MACINTOSH (1999), estes sistemas permitem à organização ‘saber quem sabe o quê’.

- TIWANA e RAMESH (2001), listam como exemplos de sistemas de gestão de conhecimento diretórios de habilidades dos empregados.
- Para O'LEARY (1998a), a maior fonte de conhecimento são as pessoas e há a necessidade de conectá-las. Cita como exemplo uma ferramenta, o *ContactFinder*, que, em vez de tentar resolver os problemas dos usuários, oferece assistência para recomendar outras que podem ser capazes de resolver o problema.
- Em outro artigo, O'LEARY (1998c) cita como uma aplicação corporativa comumente utilizada, uma base de conhecimento de recursos humanos sobre as capacidades e habilidades dos empregados. Diz, ainda, que estas informações coletadas sobre os empregados podem incluir dados educacionais, especialidades, experiências passadas, dentre outras.
- MILTON *et al.* (1999) descrevem uma ferramenta utilizada principalmente por engenheiros de conhecimento experientes quando produzindo o documento que descreve o conhecimento para um sistema de gestão de conhecimento. Um dos produtos relacionados a esta ferramenta são “páginas amarelas” contendo uma lista de ‘quem-sabe-o-quê’ relacionadas a informações para contato das pessoas listadas.
- BASILI *et al.* (2001) alertam que, quando a base de dados da fábrica de experiências (discutida na seção 7 do capítulo II) está vazia ou não se tem o conhecimento diretamente, deve ser possível indicar onde ou com quem conseguir o conhecimento desejado.
- ACUÑA *et al.*, (2000) descrevem um modelo de processo de *software* aplicável na construção de sistemas convencionais e baseado em conhecimento centrado em capacidade. O autor descreve um processo de *software* que representa “o que é feito” e “quem está fazendo”.
- STADER e MACINTOSH (1999) descrevem a implementação de uma ferramenta para gestão de capacidades baseada na idéia que tais sistemas devem ser capazes de proporcionar suporte operacional para o projeto e planejamento de informação estratégico atacando quatro requisitos fundamentais: análise de lacunas de habilidades, análise de construção de equipe de projetos, planejamento de recrutamento e análise de treinamento.



Os trabalhos de ACUÑA *et al.* (2000) e STADER e MACINTOSH (1999), por terem algumas interseções com esta tese merecem uma atenção especial e, por isso, serão mais comentados a seguir.

As capacidades identificadas por ACUÑA *et al.* (2000) são características ou habilidades de comportamento geral de uma pessoa em relação ao seu papel no processo de *software* em detrimento de aspectos relativos a conhecimento tecnológico ou outro conhecimento específico. São definições de capacidades que afetam o grupo de profissionais de uma organização sem considerar uma posição ou uma função dentro da organização. Exemplos das capacidades identificadas pelos autores podem ser encontrados na tabela 4.1.

**Tabela 4.1 – Categorias de capacidades (ACUÑA *et al.*, 2000)**

<b>Categorias</b>	<b>Descrição</b>	<b>Capacidades</b>
<b>Meta habilidades</b>	Relacionadas a indivíduos. Servem como preparação para um desempenho eficiente como um futuro desenvolvimento profissional.	Análise de problemas; decisão; independência; tolerância ao stress; julgamento.
<b>Beta habilidades</b>	Necessárias para a adaptação e para a evolução profissional dentro da estrutura organizacional.	Conhecimento do ambiente; inovação; criatividade; aversão a risco; perseverança.
<b>Habilidades operativas</b>	Relacionadas ao desempenho eficiente. São habilidades operativas eficientes quando o profissional trabalha em tarefas pessoais ou individuais.	Auto-organização; disciplina; abordagem ambiental.
<b>Habilidades interpessoais</b>	Relacionadas ao sucesso em tarefas que assumem um contato interpessoal com outras pessoas. Geralmente estes tipos de habilidades estão intimamente relacionadas à eficiência em posições de contato social.	Negociação; sociabilidade; trabalho em equipe; cooperação; serviço e orientação ao consumidor.
<b>Habilidades gerenciais</b>	Necessárias para a gerência de outras pessoas dentro da organização, orientando o desempenho em diferentes níveis ou estágios de supervisão e com diferentes níveis de responsabilidades.	Avaliação de colaboradores; planejamento e organização; liderança de grupo.

Segundo os autores, com a identificação de 'quem' está fazendo 'o quê' será possível, a longo prazo, a identificação e homogeneização de valores básicos ou realmente importantes do desenvolvimento de *software* para a organização assim como uma identificação precisa das pessoas envolvidas no gerenciamento de qualidade de produtos, do processo de *software* e da própria organização (ACUÑA *et al.*, 2000).

Segundo STADER e MACINTOSH (1999), há quatro tipos principais de capacidades: técnicas, físicas, de projeto e de negócio. Capacidades técnicas estão relacionadas a técnicas, métodos, ferramentas e plataformas específicas de engenharia e sistemas de informação. Capacidades de projeto e negócio estão relacionadas a habilidades de gerenciar equipes e projetos e prover serviços de negócio. Capacidades técnicas e físicas tendem a ser demonstráveis onde capacidades de negócio e projeto são mais subjetivas ou menos fácil de confidencialmente qualificar competência. Capacidades podem ser bem gerais, por exemplo 'Projeto de Dispositivos', ou muito específicas, por exemplo 'Consertar um dispositivo específico'. A troca de informações é um grande detalhe em tomadas de decisões num registro de capacidades grande e difícil de manter. O sistema precisa definir capacidades num nível necessário para prover a organização com facilidades de suporte à decisão estratégica e tática.

As tabelas 4.2a e 4.2b contêm parte da lista de capacidades definidas hierarquicamente por STADER e MACINTOSH (1999).

**Tabela 4.2a – Hierarquia de capacidades de alto nível (STADER e MACINTOSH, 1999)**

<b>Capacidade Técnica</b>	Análise	Análise de requisitos
		Análise de riscos
		Análise de qualidade
		Teste
		Revisão
	Modelagem	-
	Engenharia	Projeto
		Integração
		Manutenção
		Uso
	Capacidade cognitiva	Entendimento
		Raciocínio
		Criatividade
Conhecimento		
Criação de documentos		
Gerência		
<b>Capacidade Física</b>	Capacidade manual	Desenvolvimento
		Reparo
		Remoção
		Filtragem
	Substituição	
Capacidade sensorial	-	

**Tabela 4.2b – Hierarquia de capacidades de alto nível (STADER e MACINTOSH, 1999)**

<b>Capacidade de Projeto</b>	Gerência	Planejamento
		Organização
		Controle
	Comunicação	Requisição
		Resposta
		Informar
<b>Capacidade de Negócio</b>	Marketing	Contato com o consumidor
	Comercial	Contratos
		Negociação
	Pessoal	-
Financeira	-	

### 4.3 Fundamentação / Características Desejadas

Gestão de capacidades é a prática de entender as capacidades que uma organização requer para cumprir seus objetivos de negócio (STADER e MACINTOSH, 1999). Consiste em identificar que habilidades individuais existem dentro da organização e ser capaz de confrontar o conhecimento de especialistas requerido com o conhecimento disponível para permitir preencher qualquer lacuna dentro da organização. É o alinhamento entre o conhecimento atual e futuro dos empregados com os objetivos estratégicos de negócio. Organizações precisam de funcionários que possuam habilidades muito específicas, não importando que sejam técnicas, operacionais ou gerenciais. Conforme as organizações se desenvolvem para atingir as necessidades estratégicas futuras, estas habilidades precisam ser rastreadas e as eventuais lacunas identificadas. Especificamente, organizações precisam ser capazes de (STADER e MACINTOSH, 1999):

- Apreciar as principais habilidades dos indivíduos necessárias para os projetos atuais e futuros;
- Reconhecer explicitamente as habilidades promissoras dos indivíduos que podem ser utilizadas em negócios atuais e futuros;
- Reconhecer as habilidades importantes para a sobrevivência do negócio a curto e longo prazo;
- Identificar indivíduos que tenham habilidades específicas;
- Alocar pessoal com habilidades específicas a projetos de forma eficiente e efetiva.

Os possíveis requisitos para a construção de um sistema que facilite a localização de uma pessoa com características específicas dentro de uma organização podem variar enormemente de acordo com as necessidades. E num dado momento, inevitavelmente, o problema inicial se dividirá em dois. Se o objetivo final é encontrar alguém pode ser desejável identificar onde dentro da estrutura da organização este alguém poderá ser encontrado. Para isso será necessário descrever a organização em seus detalhes, as unidades organizacionais que a compõe e posições (também denominadas funções ou cargos) relacionadas. Ou seja, o sistema construído alimentará uma base de dados sobre a estrutura da organização e sobre características possuídas por seus funcionários.

Dos oito 'domínios de avaliação de valor de conhecimento' citados por HENDRIKS e VRIENS (1999) para a construção de sistemas baseados em conhecimento (ver seção 4 do capítulo II) cinco são comentados no âmbito do sistema proposto:

- **Estabelecimento de valor estratégico de elementos do *portfolio*:** O conhecimento armazenado/coletado será útil na alocação de tarefas para a equipe de desenvolvimento de *software* no decorrer de projetos. Também pode ser útil para a avaliação dos funcionários e identificar possíveis lacunas existentes entre as características desejadas para um cargo e as características realmente possuídas pela pessoa que o ocupa.
- **Determinação do conhecimento requerido:** Deve-se ser capaz de identificar as habilidades, o conhecimento e a experiência possuídos por cada funcionário. Deve-se, também, ser capaz de armazenar a estrutura da organização e as habilidades, conhecimento e experiências requeridos pelas diversas partes desta estrutura.
- **Distribuição e compartilhamento de conhecimento:** o conhecimento armazenado ficará disponível para toda a organização. Dessa forma, os funcionários poderão identificar suas possíveis deficiências em relação ao cargo ocupado e os responsáveis pela alocação de tarefas no decorrer de um projeto de *software* poderão encontrar mais facilmente as informações e as pessoas de que necessitam;
- **Aplicação de conhecimento:** o conhecimento adquirido tem por função principal auxiliar a alocação de pessoal a atividades do processo de *software* e permitir a localização de um especialista dentro da organização em questão;

- **Avaliação do conhecimento:** com o uso freqüente da ferramenta será possível avaliar se os dados coletados são realmente suficientes para as finalidades para os quais foram definidos. Outra forma de avaliação pode ser feita quando da realização de pesquisas sobre o conteúdo da base de conhecimento, com o usuário atribuindo valores qualitativos aos dados retornados.

Como o trabalho proposto está inserido no contexto da Estação TABA (capítulo III), mais especificamente em Ambientes de Desenvolvimento de *Software* Orientados à Organização (seção 5 do capítulo III), alguns de seus requisitos serão supridos.

Os requisitos gerais dos Ambientes Configurados a serem supridos são:

- (i) **Permitir a descrição da estrutura organizacional:** um Ambiente Configurado deve permitir a descrição da estrutura organizacional e a definição dos conhecimentos, habilidades e experiências desejadas em cada posição desta estrutura.
- (ii) **Permitir a descrição dos profissionais da organização:** um Ambiente Configurado deve permitir a descrição do perfil dos profissionais da organização e a alocação dos mesmos à estrutura organizacional.

Os requisitos adicionais dos Ambientes Configurados para Organizações de *Software* a serem supridos são:

- (iii) **Permitir a descrição da estrutura organizacional dos clientes:** um Ambiente Configurado deve permitir a descrição da estrutura organizacional dos clientes da organização.
- (iv) **Permitir a descrição dos profissionais das organizações clientes:** um Ambiente Configurado deve permitir a descrição do perfil de profissionais das organizações clientes e a alocação dos mesmos a suas respectivas estruturas organizacionais.

Os requisitos gerais dos ADSOrg a serem supridos são:

- (v) **Apoiar a localização de profissionais:** Um ADSOrg deve apoiar a localização dos profissionais da organização mais adequados para auxiliar na execução de uma atividade ou na solução de um problema;

Os requisitos adicionais dos ADSOrg Instanciados em Organizações de *Software* a serem supridos são:

- (vi) **Permitir a evolução do modelo da organização cliente:** um ADSOrg deve permitir que o modelo da organização cliente, já disponível no ambiente configurado e composto da estrutura organizacional, dos profissionais alocados a esta estrutura e dos processos organizacionais, seja atualizado e expandido de acordo com as informações obtidas no projeto corrente.
- (vii) **Apoiar a localização de profissionais da organização cliente:** um ADSOrg deve apoiar a localização dos profissionais da organização cliente que sejam mais adequados para auxiliar na execução de uma atividade ou na solução de um problema.

Os requisitos apresentados estão fortemente relacionados, sendo que (i), (ii), (iii), (iv) e (vi) referem-se ao armazenamento de conteúdo e (v) e (vii) referem-se à utilização deste conteúdo. Uma equipe de projeto que tem acesso à infra-estrutura organizacional na qual está inserida é capaz de localizar, mais facilmente, os especialistas que podem trazer contribuições ao projeto. Isto é muito importante quando o conhecimento necessário não está disponível no ADSOrg ou quando, apesar de disponível, a sua aplicação requer entendimento mais profundo (VILLELA *et al.*, 2000).

O uso de ontologias permite a unificação do vocabulário utilizado, facilitando a associação entre as múltiplas bases de conhecimento e a comunicação e o entendimento compartilhado entre pessoas com diferentes necessidades e pontos de vista (USCHOLD e GRUNINGER, 1996) (O'LEARY, 1998c). O'LEARY (1998a) acredita que as organizações devem ser capazes de gerar ontologias que permitam aos usuários estabelecer os recursos que precisam e desejam. Sendo assim, ao descrever as relações entre os conceitos mapeados (ex.: organização, unidades organizacionais, posições, habilidades, conhecimento, experiências etc) é de grande importância que utilize uma ontologia para, dentre outras coisas, garantir a pertinência e corretude das relações definidas.

#### 4.4 Características da Ferramenta

Um dos produtos decorrentes deste trabalho é a descrição das características de uma ferramenta que possibilite a desenvolvedores de *software* rapidamente encontrarem,

dentro da estrutura organizacional, os profissionais mais adequados à realização de uma atividade ou à solução de um problema. Para que este objetivo seja alcançado foi definida uma lista de 8 requisitos a serem cumpridos (VILLELA *et al.*, 2001c):

1. permitir a representação da estrutura organizacional, isto é, a identificação de quais são as unidades (diretorias, setores etc.) que compõem a organização e como as unidades organizacionais estão relacionadas quanto à distribuição de autoridade e responsabilidade;
2. possibilitar a descrição das unidades organizacionais, ou seja, a especificação de quais são as posições desempenhadas dentro da unidade organizacional;
3. fazer a associação dos profissionais da organização às posições que desempenham;
4. permitir a determinação de quais conhecimentos, habilidades e experiências são necessários para o bom desempenho de cada posição;
5. fazer a associação, para cada profissional da organização, dos conhecimentos, habilidades e experiências adquiridas ao longo de sua carreira profissional;
6. permitir a visualização da estrutura organizacional e a navegação através da mesma, fornecendo, sob solicitação, detalhes sobre as unidades organizacionais e os profissionais alocados a elas;
7. possibilitar uma pesquisa direta que permita descobrir quem na organização possui determinado conhecimento, habilidade ou experiência;
8. possibilitar uma pesquisa indireta que permita ao usuário navegar entre os conhecimentos, habilidades e experiências disponíveis na organização em busca de quem possui a habilidade, o conhecimento ou a experiência mais próximo do desejado.

Os três primeiros itens se referem à descrição da estrutura da organização e alocação de pessoal. Os itens 4 e 5 se relacionam à determinação de conhecimentos, habilidades e experiências desejáveis para cada posição e possuídas por cada indivíduo. Os três últimos itens são referentes a diferentes formas de visualização e pesquisa sobre o conteúdo da base de dados construída.

Para melhorar o entendimento desta seção ela foi dividida em 4 subseções. A primeira contém a descrição da estrutura da organização e a alocação de pessoal nesta estrutura (referentes aos requisitos 1, 2 e 3). A segunda seção descreve os três tipos de

ontologias (organização, domínio e engenharia de *software*) utilizadas para ou durante a definição dessa ferramenta. A seção três está relacionada à determinação de conhecimentos, habilidades e experiências desejáveis para cada posição e possuídas por cada indivíduo (referentes aos requisitos 4 e 5). A última seção se refere às diferentes formas de visualização e pesquisa possíveis (referentes aos requisitos de 6 a 8).

#### 4.5.1 Estrutura Organizacional

A ferramenta deve ser genérica, ou seja, deve ser capaz de descrever a estrutura, completa ou em parte<sup>4</sup>, de qualquer organização, seja uma organização cuja principal área de atuação seja a construção de produtos de *software* para terceiros ou que desenvolva *software* para uso próprio.

Uma organização pode ser decomposta em unidades organizacionais. A estrutura organizacional caracteriza-se por uma cadeia de comando, ou seja, por uma linha de autoridade que interliga as posições da organização e especifica quem se subordina a quem (CHIAVENATO, 1998). A representação da estrutura organizacional consiste da identificação das unidades organizacionais e de como estas unidades estão relacionadas quanto à distribuição de autoridade e responsabilidade. A partir da descrição dessas relações entre unidades organizacionais e de sua organização em uma estrutura hierárquica será possível ter uma representação do seu organograma (referente ao requisito 1).

A nomenclatura de uma unidade organizacional pode variar muito dependendo de sua função ou apenas do vocabulário adotado pela organização em questão. Possíveis exemplos de unidades organizacionais são: diretoria, setor, departamento, divisão etc. Cada unidade organizacional por sua vez pode ser subdividida em partes menores, ou seja, a especificação de quais são as funções desempenhadas dentro da unidade organizacional (referente ao requisito 2).

---

<sup>4</sup> Por exemplo, quando uma empresa que desenvolve *software* para terceiros é contratada nem sempre é capaz de ter uma visão completa da organização cliente, às vezes tem acesso apenas a um determinado setor desta organização que é o patrocinador do produto e, por isso, é o único que precisa ou consegue ser descrito.



Devido a isto, CHIAVENATO (1998) lista dois tipos de autoridade entre unidades organizacionais e posições: em uma os gerentes têm o poder formal de dirigir e controlar os subordinados imediatos; em outra define o direito de aconselhar, recomendar e orientar sendo uma relação de comunicação: os especialistas aconselham os gerentes em suas áreas de especialidade.

Diferentes ambientes levam as empresas a adotar novas estratégias e as novas estratégias exigem diferentes estruturas organizacionais (CHIAVENATO, 1998). O desenho organizacional retrata a configuração estrutural da organização e implica no arranjo dos órgãos dentro da estrutura no sentido de aumentar a eficiência e a eficácia organizacional (CHIAVENATO, 1998). Todavia, como as organizações vivem em um mundo de mudança, a sua estrutura deve caracterizar-se pela flexibilidade e pela adaptabilidade ao ambiente e à tecnologia (CHIAVENATO, 1998).

Para ser eficiente (CHIAVENATO, 1998), a produção deve basear-se na divisão do trabalho, que nada mais é do que a maneira pela qual um processo complexo pode ser decomposto em uma série de pequenas tarefas que o constituem. Como consequência do princípio da divisão do trabalho surge a especialização: cada órgão ou cargo passa a ter funções e tarefas específicas e especializadas. A pluralidade de funções imposta pela especialização exhibe inevitavelmente o desdobramento da função de comando, cuja missão é dirigir todas as atividades para que estas cumpram harmoniosamente as suas respectivas missões. Isto significa que, além de uma estrutura de funções especializadas, a organização precisa também de uma estrutura hierárquica para dirigir as operações dos níveis que lhe estão subordinados.

A departamentalização (CHIAVENATO, 1998) é um meio para se obter homogeneidade de tarefas em cada órgão. No fundo, a departamentalização consiste em escolher modalidades de homogeneização de atividades, agrupando os componentes da organização em departamentos ou divisões. Os principais tipos de departamentalização são (CHIAVENATO, 1998): por funções, por produtos ou serviços, por localização geográfica, por clientes, por fases do processo produtivo e por projetos.

Difícilmente se encontra, na prática, a aplicação pura de um único tipo de departamentalização em toda a organização. É comum encontrar-se uma conjugação de

diversos tipos de departamentalização, seja no mesmo nível seja nos diferentes níveis hierárquicos da organização (CHIAVENATO, 1998).

Segundo BARNARD (1971), as pessoas não atuam isoladamente, mas através de interações com outras pessoas, para poderem alcançar da melhor maneira seus objetivos. Graças às diferenças individuais, cada pessoa tem as suas próprias características pessoais, suas capacidades e limitações. Para poderem sobrepujar suas limitações e ampliar suas capacidades, as pessoas precisam cooperar entre si para melhor alcançarem seus objetivos. É através da participação pessoal e da cooperação entre as pessoas que surgem as organizações.

Uma vez descrita a organização em termos de unidades organizacionais e posições é necessário fazer a alocação de pessoal a cada uma das posições. Uma pessoa pode ser alocada a várias posições dentro da organização em diferentes momentos, por isso, é necessário armazenar o período em que essa pessoa ficou alocada (referente ao requisito 3).

#### **4.4.2 Tipos de ontologias utilizadas**

A utilização das ontologias descritas (de Domínio, de Engenharia de *Software* e de Organização) visa minimizar ambigüidades na comunicação entre os agentes computacionais e humanos envolvidos no contexto dos Ambientes de Desenvolvimento de *Software* Orientados à Organização. Nenhuma dessas ontologias é contribuição deste trabalho: as ontologias de organização e de engenharia de software (VILLELA *et al.*, 2002) são contribuições de um trabalho de doutorado em andamento, as ontologias de domínio utilizadas no momento são descritas em (OLIVEIRA, 1999), (GALOTTA, 2000) e (FOURO, 2002).

##### **Ontologia de Organização**

A ontologia de organização (VILLELA *et al.*, 2002) tem como propósito fornecer um vocabulário comum, no qual o conhecimento sobre as organizações, útil para desenvolvedores de *software*, possa ser expresso. Este conhecimento inclui conhecimento sobre a estrutura, o comportamento e os profissionais da organização.

A ontologia de organização fornece os conceitos e atributos relacionados à estrutura e aos processos organizacionais, possibilitando a descrição da organização e

permitindo especificar, por exemplo, o contexto em que um item de conhecimento, habilidade ou experiência é necessário. Para esta ferramenta apenas será utilizada a parte da ontologia da organização referente à estrutura da organização. Os processos organizacionais, assim como os processos de *software*, utilizados fogem de seu escopo e, por isso, não serão levados em consideração.

Na figura 4.1 pode ser vista a sub-ontologia de estrutura da organização ontologia de organização. A notação utilizada nesta representação utiliza a notação de modelo de classes da UML é descrita no Anexo II. A versão final e completa dessa ontologia é uma das contribuições de um trabalho de doutorado em andamento. O mapeamento dos conceitos aqui apresentados para o modelo de classes correspondente (ver Anexo I) é contribuição deste trabalho.

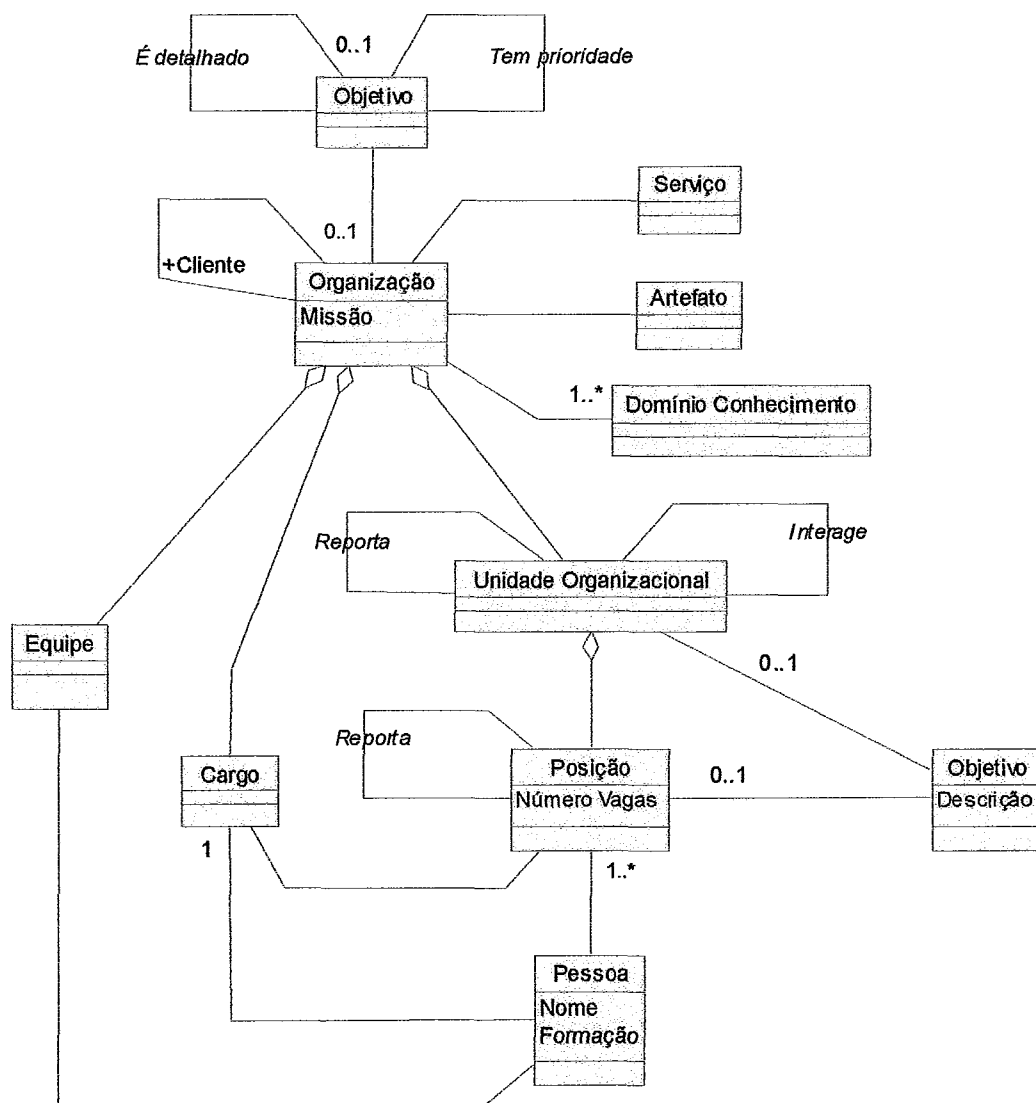


Figura 4.1 – Sub-ontologia de Estrutura da Organização

Os principais conceitos listados na figura 4.1 são:

- **Organização:** representa a organização que pode ser definida como sendo um grupo organizado de pessoas trabalhando em conjunto para cumprimento de uma missão. As pessoas são distribuídas nas diversas unidades organizacionais que compõem a estrutura da organização.
- **Unidade Organizacional:** é um agrupamento de componentes da organização (atividades, pessoas etc.) para que a organização possa ser econômica e eficiente. Cada unidade organizacional pertence a uma e somente uma organização e é composta por várias posições. As unidades organizacionais se relacionam entre si através de relações de subordinação, compondo uma estrutura hierárquica, ou de cooperação.
- **Posição:** Localização de uma pessoa na estrutura organizacional, estabelecendo a que unidade organizacional ela pertence, a que pessoas deve reportar e a quais deve comandar, considerando as posições que ocupam. Cada posição pertence a uma única unidade organizacional e se relaciona com outras posições através de relações de subordinação.
- **Pessoa:** Componente fundamental da organização, personificação de um recurso humano da organização. Representa as pessoas pertencentes à estrutura da organização e que estão alocadas a alguma posição.
- **Serviço:** É um produto (ou seja, algo oferecido para satisfazer a uma necessidade ou desejo de um determinado cliente ou mercado) não tangível, abstrato.
- **Artefato:** Qualquer coisa produzida pelo homem, e não por causas naturais, que é objeto de transformação de uma atividade, atuando como insumo ou produto dela.
- **Objetivo:** Enunciado escrito sobre os resultados a serem alcançados num período de tempo determinado.
- **Equipe:** Conjunto ou grupo de pessoas que se aplicam a uma tarefa.
- **Cargo:** A definição de cargo envolve a descrição das tarefas e responsabilidades atribuídas ao seu ocupante.
- **Domínio de Conhecimento:** Escopo de atuação de uma organização, delimitando o conjunto de atividades que ela pretende executar.

Esta ontologia foi utilizada como base para a definição das classes (que podem ser vistas no Anexo I) utilizadas pela ferramenta e, posteriormente, para auxiliar na busca de informações sobre a base de dados construída.

### **Ontologia do domínio**

A ontologia do domínio define o vocabulário para representar o conhecimento de um domínio de aplicação, expressando conceitos e fatos deste domínio, que pode ser, por exemplo, cardiologia, telecomunicação, acústica submarina etc. A ferramenta utiliza a teoria do domínio como uma forma de descrição do conhecimento do domínio possuído por um especialista. Esta informação será particularmente útil quando houver necessidade de pesquisa por um especialista em alguma área de conhecimento.

Atualmente na Estação Taba estão disponíveis três diferentes teorias de domínio: cardiologia (OLIVEIRA, 1999) (OLIVEIRA *et al.*, 2000), acústica submarina (GALOTTA, 2000) e entomologia (FOURO, 2002). Outras teorias de domínio podem ser definidas através da ferramenta EdiTeD (OLIVEIRA, 1999) (OLIVEIRA *et al.*, 2000).

### **Ontologia de Engenharia de *Software***

Por fim, a ontologia de Engenharia de *Software* define o vocabulário que orienta o registro e a distribuição de conhecimento sobre engenharia de *software* no ADSOrg. FALBO (1998) definiu uma ontologia de processo de *software* que fará parte da ontologia de engenharia de *software* definida por VILLELA *et al.* (2000).

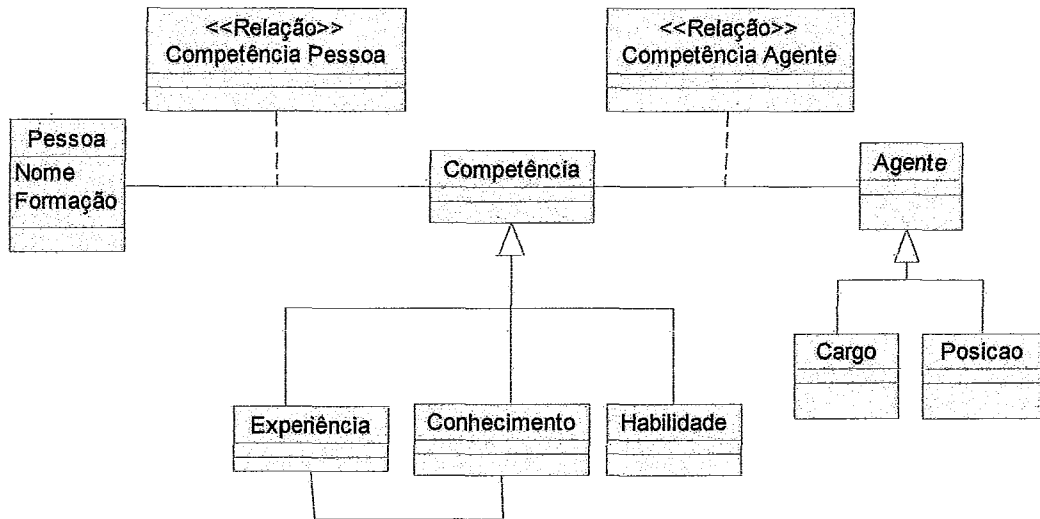
Engenharia de *software* também é um domínio de aplicação e como tal também pode ser associado a um determinado conhecimento. Tanto a ontologia do domínio quanto a de Engenharia de *Software* orientam o registro do mapa de distribuição de conhecimento, habilidades e experiências dentro da organização.

#### **4.4.3 Conhecimentos, Habilidades e Experiências**

Os funcionários da organização adquirem conhecimentos, habilidades e experiências com o passar do tempo. Muitas vezes, uma alocação a uma determinada posição é realizada baseando-se nessas características: cada posição requer um conjunto

mínimo de habilidades, conhecimentos e experiências (referente aos requisitos 4 e 5) para que suas atribuições possam ser cumpridas.

Além dos conceitos exibidos na figura 4.1, a ontologia também define os conceitos de competência, habilidade, conhecimento e experiência. Uma pessoa, assim como uma posição, pode ter associadas a ela várias competências, como pode ser visto na figura 4.2 (uma descrição da notação utilizada pode ser vista no Anexo II).



**Figura 4.2 – Ontologia de Organização – Relacionamentos relativos a Competências**

Os conceitos de conhecimento, habilidade e experiências estão intimamente relacionados, apesar de terem significados distintos. Na verdade, pode-se generalizá-los como sendo competência, ou seja, o elemento que torna uma pessoa capaz de executar uma determinada atividade. Neste trabalho, são assumidas as seguintes definições:

- **Competência:** elemento que torna uma pessoa capaz de executar uma determinada atividade.
- **Conhecimento:** competência adquirida que representa a apropriação de um objeto pelo pensamento através de definição, percepção clara, apreensão completa, análise etc. Por exemplo, podem ser considerados conhecimentos sobre a especialidade de cirurgia cardíaca ou sobre manipulação de instrumentos aeronáuticos.
- **Habilidade:** competência que representa aptidão nata ou adquirida não associada a uma atividade ou área de competência específica. Por exemplo, capacidade de liderança, disciplina ou tolerância a situações de estresse.

- **Experiência:** competência adquirida através da execução de atividades. Por exemplo, experiência em gerência de projetos, experiência na realização de análise de requisitos etc.

Devido a decisões tomadas durante a elaboração da ferramenta, a ser apresentada no próximo capítulo, a caracterização de uma competência é feita, basicamente, através de um nome (para fácil e rápida referência) e uma descrição (que pode ser utilizada para detalhar o seu verdadeiro significado). Um conhecimento pode ser decomposto, se necessário, em conhecimentos mais específicos objetivando aumentar o nível de detalhe e especialização da definição original.

Conhecimentos e experiências se relacionam com conceitos de um domínio de aplicação (representados por uma ontologia de domínio) e alguns de suas possíveis instâncias. Por exemplo, pode se relacionar um conhecimento “Realização de Cirurgia” ao conceito “Cirurgia” da ontologia de Cardiologia. Entretanto, caso se queira representar um outro conhecimento denominado “Realização de Cirurgia Cardíaca”, além de fazer a associação com o conceito “Cirurgia” pode-se associá-lo com uma de suas instâncias: “Cirurgia Cardíaca”. Essa associação dupla é feita porque uma associação apenas “Cirurgia” teria uma abrangência muito ampla e poderia não corresponder exatamente ao desejado, no caso “Cirurgia Cardíaca”. De forma semelhante, “Experiência em Análise de Requisitos” estaria associada ao conceito “Atividade” da ontologia de Processo de *Software* e à sua instância “Análise de Requisitos”.

A associação entre pessoas e posições com as competências não é sempre igual, geralmente levam em consideração um certo grau de variação. Por isso, também por decisões de implementação, foram criadas escalas de valores que podem ser associadas a uma determinada competência. Estas escalas (FENTON e PFLEEGER, 1997) podem ser ordinais (ou seja, o usuário poderá associar qualquer valor numérico) ou nominais (ou seja, são compostas por itens numerados). Alguns exemplos de escalas são listados a seguir:

- **Escala Básica de Habilidades:** 1 – “Não possui a habilidade nem participou de treinamento”; 2 – “Participou de treinamento”; 3 – “Possui habilidade”; 4 – “Possui grande habilidade”.
- **Escala Básica de Conhecimento:** 1 – “Nenhum”; 2 – “Conhece e sabe executar com supervisão”; 3 – “Conhece e executa sem supervisão”; 4 – “Possui expertise e mentoriza”.

- **Escala Básica de Experiência em número de projetos:** 1 – “Nenhum”; 2 – “Em 1 projeto”; 3 – “Em 2 projetos”; 4 – “Entre 3 e 5 projetos”; 5 – “Em mais de 5 projetos”.
- **Escala Básica de Experiências em tempo:** 1 – “Nenhum”; 2 – “Entre 2 e 6 meses”; 3 – “Entre 6 meses e 1 ano”; 4 – “Entre 1 e 3 anos”; 5 – “Acima de 3 anos”.

Abaixo há uma lista de perguntas que devem poder ser respondidas através da utilização da ferramenta. Os itens estão agrupados em quatro diferentes categorias.

- Descrição da organização em linhas gerais:
  - Quem é a organização?
  - Quais são seus objetivos e metas?
  - Como ocorre a distribuição de autoridade e a delegação de poder?
  - Qual é sua estrutura?
  - Como é decomposta em unidades organizacionais?
- Descrição das unidades organizacionais:
  - Qual a estrutura de uma unidade organizacional?
  - Como as unidades se relacionam entre si?
  - Dada uma unidade organizacional, com quem ela se relaciona?
  - Como são decompostas em posições?
  - Quais são os membros de uma determinada unidade?
- Descrição das posições que compõem uma unidade organizacional:
  - Como e com que finalidade as posições interagem entre si?
  - A quem e com que finalidade uma determinada posição se reporta?
  - Que itens de conhecimento são requeridos por uma determinada posição?
  - Que habilidades são requeridas por uma determinada posição?
  - Que experiências são requeridas por uma determinada posição?
  - Que pessoa na organização ocupa uma determinada posição?
- Descrição das pessoas associadas a uma determinada posição:
  - Que posição na organização ocupa uma determinada pessoa?



- Como o conhecimento, as habilidades e as experiências encontram-se distribuídos entre as pessoas que compõem a organização?
- Quem na organização possui um determinado conhecimento?
- Quem na organização possui uma determinada habilidade?
- Quem na organização possui uma determinada experiência?
- Que conhecimentos uma determinada pessoa possui?
- Que habilidades uma determinada pessoa possui?
- Que experiências uma determinada pessoa possui?

#### 4.4.4 Busca

Uma base de dados perderá seu sentido de existir se dela não puderem ser obtidos dados úteis. TIWANNA e RAMESH (2001) alertam que sem ferramentas de recuperação e procura inteligentes e poderosas para dar suporte à criação de meta-informação, um sistema de gestão de conhecimento é tão limitado quanto um sistema de informação tradicional.

Ontologias definem conceitos e relações entre eles que, às vezes, servem como base para a construção de classes no ‘modelo lógico’ de dados. Entretanto, algumas vezes, por motivos de implementação, nem todas as relações são mapeadas para o ‘modelo físico’ de dados. Ao realizar buscas estas relações originais tornam-se importantes por permitirem identificar os conceitos relacionados às classes existentes no modelo e que fazem parte da infra-estrutura do modelo de dados. Por isso, ao realizar uma busca a ontologia deve ser uma aliada, disponibilizando a descrição das relações entre os conceitos e, também, a descrição da estrutura de cada conceito.

Toda organização possui uma representação hierárquica de sua estrutura denominada organograma. Através do organograma pode-se ter uma visão geral de como a organização está estruturada, como cada unidade organizacional se relaciona com as outras além de quais posições compõem cada unidade organizacional (referente ao requisito 6). Deve ser possível, através dele, ver detalhes de cada unidade organizacional e de cada posição em específico. Também é útil para que se possa encontrar uma pessoa através da posição que ocupa.

O organograma mostra a estrutura da organização, mas não é o modo mais indicado para se encontrar um profissional: se a estrutura da organização for muito extensa o trabalho será árduo e nem sempre recompensador. Para isso é necessário permitir a busca por elementos mais específicos como, por exemplo, quem detém um determinado conhecimento, habilidade e experiência (referente ao requisito 7). Possivelmente este seja o modo mais eficiente para encontrar um especialista sobre algum assunto.

Quando a base de dados possuída é muito grande e extensa a escolha de uma boa estrutura de visualização de seu conteúdo é fundamental. Todas as informações dentro da base de dados se relacionam de alguma forma, formando uma grande rede de hipertexto derivada a partir das relações entre os conceitos. Assim, deve ser possível explorar essa característica, navegando através dessa base de dados percorrendo todos os seus itens através desses relacionamentos (referente ao requisito 8).

Uma estrutura denominada *Árvore Hiperbólica* é indicada para manipulação e visualização de grandes quantidades de dados organizados hierarquicamente (LAMPING *et al.*, 1995). Em tarefas de pesquisa esta estrutura obtém melhor desempenho em tempo e mais aprendizado do que árvore de dados (PIROLI *et al.*, 2000) (PIROLI *et al.*, 2001).

### **Árvore Hiperbólica**

O *browser* hiperbólico inicialmente exhibe uma árvore com sua raiz no centro, mas a exibição pode ser transformada suavemente para trazer outros nós ao foco. Devido ao fato de o espaço disponível para a exibição ser uma função contínua de sua distância na árvore para o ponto no centro, este contexto sempre inclui várias gerações de nós pais, primos e filhos, tornando fácil para o usuário explorar a hierarquia sem se perder<sup>5</sup> (LAMPING *et al.*, 1995).

Muitas estruturas hierárquicas, como organogramas ou estruturas de diretórios, são muito grandes para serem exibidas por completo na tela do computador. A exibição convencional mapeia toda a hierarquia numa região que é maior que a área disponível para visualização e então usa barras de rolagem para se mover pela região. O problema dessa

---

<sup>5</sup> Um exemplo de uma árvore hiperbólica, no contexto da ferramenta construída, pode ser vista no próximo capítulo, seção 4.

abordagem é não permitir que o usuário veja o relacionamento da parte visível da árvore com o resto da estrutura sem visões auxiliares. Seria muito útil ver toda a hierarquia enquanto se tem o foco em uma parte particular. Desse modo, as relações entre as partes que compõem o todo podem ser vistas em sua totalidade e o foco pode ser alterado de forma suave e contínua (LAMPING *et al.*, 1995) facilitando sua visualização.

## 4.5 Conclusão

Nos capítulos anteriores foram discutidas as definições envolvidas em gestão de conhecimento e ambientes de desenvolvimento de *software*. O histórico da Estação TABA, suas características e requisitos e sua evolução para a instanciação de Ambientes de Desenvolvimento Orientados a Organização também foram explicadas.

Este capítulo apresentou a motivação teórica deste trabalho e situa-o no contexto de gestão de conhecimento e da Estação TABA.

No próximo capítulo serão explicados os detalhes técnicos da implementação realizada. Para exemplificar o uso da ferramenta proposta, será descrito em detalhes um exemplo de uso da ferramenta no contexto do ambiente **Cordis-FBC**.

# CAPÍTULO V - A FERRAMENTA SAPIENS

## 5.1 Introdução

No capítulo anterior, foram discutidas as fundamentações teóricas que culminaram na construção da ferramenta **SAPIENS**, sua relação com aspectos da gestão de conhecimento e com os conceitos da Estação TABA e dos Ambientes de Desenvolvimento de *Software* Orientados a Organização, ao qual esta ferramenta está intimamente ligada.

Neste capítulo, são descritos detalhes técnicos da implementação de **SAPIENS** com a disponibilização, passo a passo, de um exemplo de uso. Um ambiente, denominado **Cordis-FBC**, está sendo construído para a validação das idéias propostas para a definição de ADSOrg (VILLELA *et al.*, 2000). Por isso, este ambiente será utilizado para a descrição das características da ferramenta.

Assim como o conceito de ADSOrg pode ser considerado uma evolução do conceito de ADS, o ambiente orientado à organização **Cordis-FBC** é uma evolução do ambiente orientado a domínio **Cordis**. Ambos são descritos neste capítulo e a utilização da ferramenta dentro do contexto do **Cordis-FBC** é destacada.

Este capítulo é constituído por cinco seções:

- **Introdução:** contém a apresentação deste capítulo;
- **Cordis:** descreve o ambiente de desenvolvimento de *software* orientado a domínio **Cordis**, desenvolvido para a Fundação Bahiana de Cardiologia e instanciado pela Estação TABA utilizando a Teoria de Domínio de cardiologia e um processo especializado;
- **Cordis-FBC:** descreve o ambiente de desenvolvimento de *software* orientado a organização **Cordis-FBC**. Este ambiente, desenvolvido para a Fundação Bahiana de Cardiologia, é um importante laboratório de pesquisa e validação dos conceitos envolvidos na construção de ADSOrg e é uma evolução do ambiente **Cordis**;

- **Exemplo de Uso / Detalhes da Implementação:** apresenta um exemplo completo de utilização da ferramenta e discute alguns aspectos de sua implementação;
- **Conclusão:** contém uma breve resumo do capítulo e a descrição dos tópicos a serem discutidos no próximo capítulo.

## 5.2 Cordis

Acreditando na importância de organizar o conhecimento do domínio para apoiar o desenvolvimento de sistemas Oliveira (OLIVEIRA *et al.*, 2000) definiu ambientes de desenvolvimento de *software* orientados a domínio (ADSOD). Esses ambientes possuem o conhecimento organizado através de ontologias do domínio (GRUBER, 1995) e consideram uma atividade de investigação do domínio para estabelecer o uso e o apoio no desenvolvimento de sistemas utilizando esse conhecimento.

O desenvolvimento de sistemas requer um contínuo aprendizado de aspectos relativos ao domínio pelos desenvolvedores de *software*, o que implica em muitas interações com os especialistas. Durante o trabalho realizado no desenvolvimento de diferentes sistemas na Unidade de Cardiologia e Cirurgia Cardiovascular/Fundação Bahiana de Cardiologia da Universidade Federal da Bahia (UCCV/FBC) percebeu-se que a atividade de estudo do domínio, embora não presente neste processo, era constante em todas as fases do desenvolvimento (OLIVEIRA, 1999). O trabalho realizado na UCCV/FBC, não só serviu de motivação para o estudo e definição de ADSOD, como, também, permitiu observar que o uso do conhecimento do domínio poderia atuar como diferencial importante durante o desenvolvimento de produtos de *software* (OLIVEIRA, 1999).

Um dos problemas cruciais no desenvolvimento de *software*, e que afeta diretamente a qualidade do produto, é a identificação correta do que o sistema a ser desenvolvido deve atender, isto é, sua especificação de requisitos. Esta questão é ainda mais crítica quando os desenvolvedores de *software* trabalham num domínio que não conhecem. Neste caso, a produtividade no desenvolvimento também é prejudicada. Esta situação ficou evidenciada durante o desenvolvimento de diferentes tipos de *software* na área de cardiologia pela equipe da COPPE em colaboração com a UCCV/FBC. Durante esse período, a equipe de desenvolvimento sofreu várias alterações de pessoal. Percebeu-se que cada novo integrante

da equipe tinha necessidade de entender os conceitos básicos de cardiologia antes de começar as suas atividades nos projetos. Ao se engajarem em um projeto concreto, os desenvolvedores tinham, ainda, que conhecer conceitos e tarefas relacionadas às especificidades do projeto. Baseado nessa premissa foram definidos Ambientes de Desenvolvimento de *Software* Orientados a Domínio (ADSOD) (OLIVEIRA *et al.*, 1999b) (OLIVEIRA, 1999) como uma ferramenta para apoiar desenvolvedores não familiares com o domínio na atividade de construção de sistemas.

A partir do estudo realizado sobre ADSOD foi definido e construído **CORDIS**, um ADSOD para cardiologia para ser utilizado no desenvolvimento de diferentes tipos de *software* na UCCV/FBC. Para construção do ambiente, se fez necessário a definição de uma teoria do domínio para cardiologia e a definição de um processo padrão de desenvolvimento de *software* (contendo a atividade de investigação do domínio) a ser utilizado na UCCV/FBC. Como o processo padrão descreve os elementos fundamentais que devem ser incorporados em qualquer processo de *software* definido em uma organização (EMAN *et al.*, 1998), podemos definir um processo para cada tipo de sistema a ser desenvolvido na UCCV/FBC.

A Teoria do Domínio para cardiologia foi definida por OLIVEIRA (1999) através da literatura e do conhecimento obtido com cardiologistas da UCCV/FBC. Para identificação das sub-teorias considerou-se dois aspectos principais: definir como está, realmente, estruturado o objeto principal da cardiologia, ou seja, o coração, e quais os conceitos relacionados a como descobrir, estudar e nomear eventos que ocorrem dentro do coração. A partir desses aspectos gerais, foram definidas cinco sub-teorias: anatomia do coração, que possui os conceitos sobre a estrutura do coração e sua fisiologia; patologia, que representam estados do coração, mas cuja classificação é importante para o propósito da Teoria do Domínio no ADSOD; avaliação clínica e exames complementares, que representam os conceitos utilizados pelos cardiologistas no processo de investigação médica; diagnóstico, com as classificações dos tipos de diagnóstico, e, terapia, que contém os conceitos sobre terapia empregadas em cardiologia. Todos os conceitos referentes a cada sub-teoria foram descritos e caracterizados por um conjunto de propriedades e exemplos de instâncias. Após definidas as sub-teorias e construídas suas ontologias correspondentes foram mapeadas as sub-teorias com as potenciais tarefas do domínio: diagnóstico, planejamento terapêutico, simulação e monitoração de pacientes. Esse mapeamento indica o que deve ser considerado quando se for desenvolver uma aplicação.

A Teoria do Domínio é disponibilizada, então, em um repositório que pode ser acessado nas diferentes atividades do desenvolvimento de sistemas. Existe, também, um repositório com descrição das diferentes tarefas e referências bibliográficas sobre a mesma (OLIVEIRA *et al.*, 2000). Além disso, foi definida a interface geral do ambiente e uma ferramenta que permite o aprendizado do domínio pela investigação dos conceitos (GALOTTA, 2000).

**CORDIS** é um ambiente de desenvolvimento de *software* orientado ao domínio de cardiologia que tem como objetivo apoiar o desenvolvimento de sistemas a partir do estabelecimento de um processo de *software* bem definido e o uso do conhecimento sobre cardiologia disponibilizado para apoiar o entendimento do domínio pelos desenvolvedores e estabelecer uma linguagem comum entre os diferentes participantes do desenvolvimento (usuários e desenvolvedores). A validação de cada parte do **CORDIS** (ontologia, processo e ferramentas) foi realizada considerando a opinião dos cardiologistas envolvidos no projeto (OLIVEIRA *et al.*, 2000).

### 5.3 Cordis-FBC

O ambiente **CORDIS** já incorpora algumas características específicas da organização, já que é instanciado a partir do processo padrão definido para ela. Como um ADSOrg envolve mais conhecimento sobre a organização do que o processo padrão, o experimento a ser realizado na Fundação Bahiana de Cardiologia, o **Cordis-FBC**, será a evolução do ADS **CORDIS** com o objetivo de torná-lo orientado à organização (VILLELA *et al.*, 2001b).

Este trabalho está situado no contexto do ADSOrg **CORDIS-FBC** que, como todo Ambiente de Desenvolvimento de Software Orientado a Organização, deve atender os seguintes requisitos (VILLELA *et al.*, 2001b):

- (i) possuir a representação da estrutura organizacional e facilitar a localização de especialistas cujo conhecimento e experiência podem ser úteis para um determinado propósito;

- (ii) armazenar conhecimento especializado sobre desenvolvimento e manutenção de *software*, e fornecer este conhecimento para as equipes de projeto quando necessário;
- (iii) possuir conhecimento sobre o domínio de conhecimento utilizado pela organização (neste caso, o de cardiologia) e fornecer ferramentas de acesso a esse conhecimento; e
- (iv) apoiar a contínua evolução do conhecimento armazenado no ambiente.

Para atender a esses requisitos, o **CORDIS-FBC** possui uma infra-estrutura de conhecimento na qual cada componente tem suas próprias metas e contém conhecimento importante como descritos a seguir (VILLELA *et al.*, 2001b):

- **Teoria do domínio:** composta de um conjunto de conceitos e relações do domínio, organizadas em subteorias, e do mapeamento desses conceitos com tarefas identificadas para o domínio considerado. Cada subteoria corresponde a uma ontologia do domínio de cardiologia (OLIVEIRA *et al.*, 2000).
- **Ontologias do domínio de cardiologia:** constituída por conceitos sobre anatomia do coração, achados e exames complementares e tipos de patologias, diagnóstico e terapias.
- **Descrição de Tarefas:** este componente contém a descrição de tarefas genéricas, independentes de organização e de domínio (OLIVEIRA *et al.*, 2000).
- **Ontologia de Tarefa:** constituída por conceitos e atributos relativos às tarefas genéricas de diagnóstico, predição, prescrição, planejamento, monitoração, simulação etc;
- **Teoria de ES (Engenharia de Software):** formada apenas pela Ontologia de ES;
- **Ontologia de ES:** define um vocabulário comum que orienta o registro e a distribuição de conhecimento sobre engenharia de *software* no ADSOrg e o projeto do mapa de distribuição de conhecimento da organização.
- **Descrição da Organização:** componente que descreve a organização para a qual o ADSOrg será construído, identificando, dentro do contexto da estrutura e dos modelos de processos organizacionais, quais as tarefas genéricas que são realizadas e o conhecimento de engenharia de *software* que é requerido;



- **Ontologia da Organização:** conceitos e atributos relacionados à estrutura e aos modelos de processo organizacionais que permitem especificar o contexto em que um item de conhecimento foi criado e o contexto de aplicação pretendido para um item de conhecimento;
- **Bases de Conhecimento/Dados:** registram o conhecimento e os dados obtido ao longo dos projetos de desenvolvimento de *software* e registram o mapa de distribuição do conhecimento, das habilidades e das experiências na organização;
- **Serviços de Gerência de Conhecimento:** serviços que irão permitir armazenar conhecimento na memória organizacional e disseminar e atualizar conhecimento já armazenado.

Os ADSOrg (e, por consequência, o Cordis-FBC) possuem ferramentas para definição, especialização e instanciação de processos (**AssistPro** (FALBO *et al.*, 1999) (FALBO, 1998), **EditPro** (SANTOS e ZLOT, 1999), **DefPro** (MACHADO, 2000) (MACHADO *et al.*, 2000a e 2000b)); definição de teorias de domínio e tarefas (**EDITED** (OLIVEIRA *et al.*, 2000), **EDITAR** (OLIVEIRA, 1999) (ZLOT, 2002) (ZLOT *et al.*, 2002)); apoio à atividade de Investigação do Domínio (**GENESIS** (GALOTTA, 2000), **NAVEGUE** (GALOTTA, 2000) e **REGCON** (GALOTTA, 2000)); apoio ao planejamento do projeto (**RiscPlan** (FARIAS *et al.*, 2001) (FARIAS, 2002), **CustPlan** (BARCELLOS *et al.*, 2001) e **RHPlan**); descrição do conhecimento organizacional (VILLELA *et al.*, 2001c) (SANTOS *et al.*, 2002) (VILLELA *et al.*, 2001a); aquisição de conhecimento (MONTONI, 2002).

A ferramenta **SAPIENS** é uma das ferramentas utilizadas para descrição do conhecimento organizacional. Permite a representação da estrutura organizacional com as habilidades, conhecimentos e experiências requeridos ao longo dessa estrutura. Além de permitir a alocação de pessoal, incluindo os conhecimentos, habilidades e experiências que cada profissional possui, contém, ainda, mecanismos para busca e navegação. Dessa forma, possibilita-se a criação de uma cultura de identificação, aquisição e disseminação do conhecimento existente que levará a organização a se conhecer melhor e tirar maior proveito de seu potencial. Mais detalhes sobre suas funcionalidades podem ser encontrados na seção a seguir.

## 5.4 Exemplo de Uso / Detalhes da Implementação

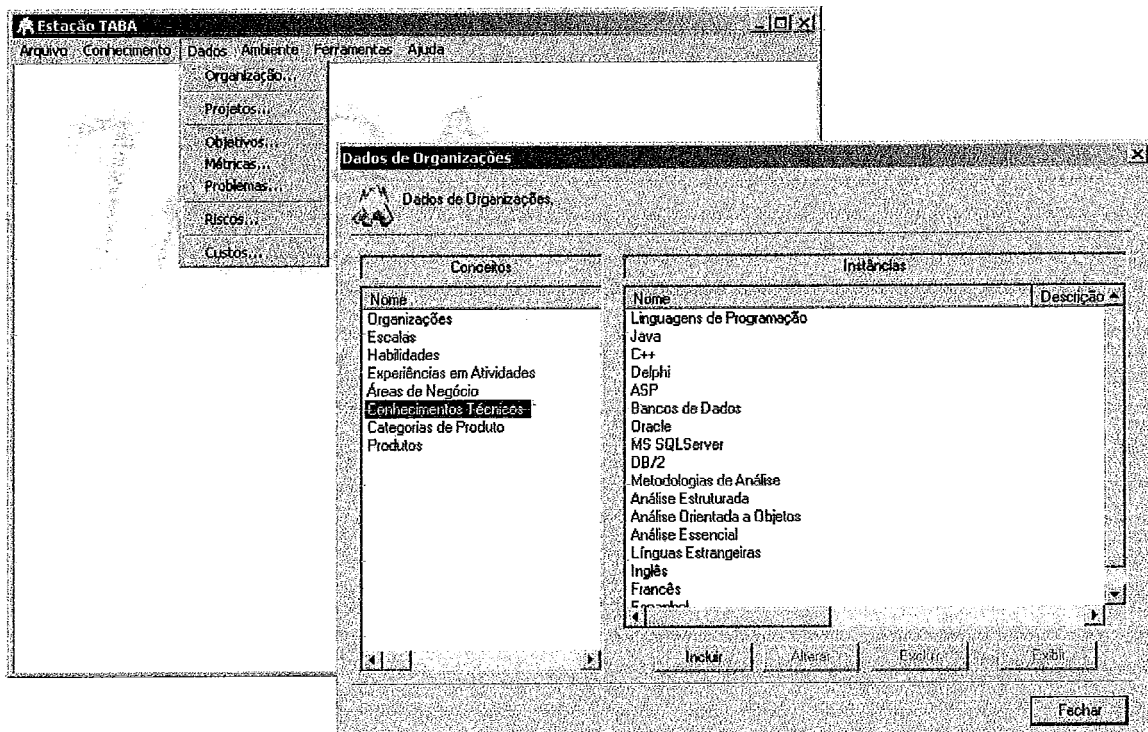
**SAPIENS** é uma ferramenta genérica, ou seja, independente de uma organização ou domínio específicos, e está baseada na infra-estrutura definida para ADSOrg, fazendo uso de ontologias de organização, do domínio e de Engenharia de *Software* para descrever as organizações que desenvolvem e mantêm *software*.

As funcionalidades disponíveis na ferramenta são descritas nessa seção. Primeiro são listadas as atividades custodiais: definição das habilidades, dos conhecimentos e das experiências. Depois são descritas a representação da estrutura organizacional e a associação dos profissionais às posições desempenhadas. Por fim, a visualização da estrutura organizacional, pesquisa e navegação sobre a base de dados.

Dentro do contexto de ADSOrg, os possíveis usuários da ferramenta são os desenvolvedores de *software* que podem utilizar a ferramenta para encontrar a pessoa mais adequada à solução de um problema e como auxílio no contexto da organização que está sendo modelada. A ferramenta auxilia o conhecimento da organização e a comunicação entre os funcionários de forma a agilizar a solução de problemas na etapa inicial de alocação de pessoal a um projeto de *software*. Além disso, pode servir de apoio a atividades do departamento de Recursos Humanos.

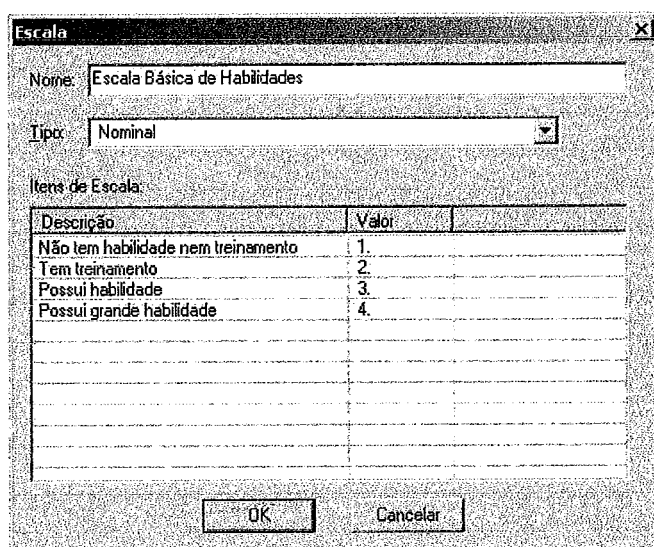
### **Dados necessários à utilização da ferramenta**

Para o correto funcionamento da ferramenta, alguns itens precisam ser especificados antes de se iniciar o seu uso: habilidades, conhecimentos, experiências e escalas. Uma vez cadastrados, poderão ser associados a posições e pessoas no decorrer da utilização da ferramenta **SAPIENS**. Este cadastro é feito fora da ferramenta porque podem ser (re)utilizados para várias organizações ou, até mesmo, várias vezes dentro de uma mesma organização. A figura 5.1 exibe o menu “Dados” da Estação TABA e a tela com os itens a serem cadastrados.



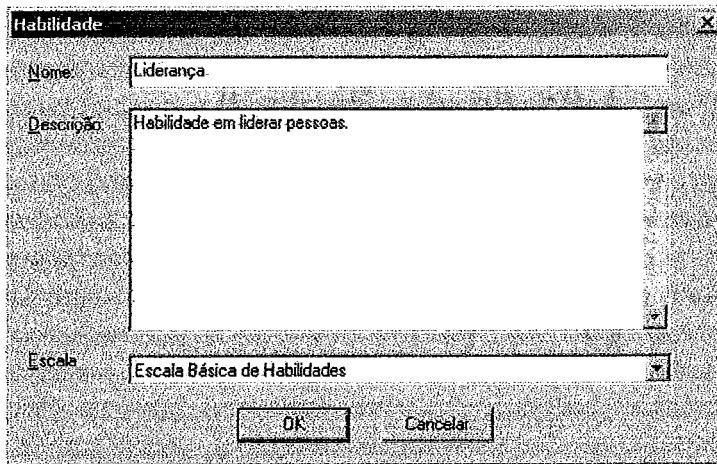
**Figura 5.1 – Menu Dados da Estação TABA**

O cadastro das escalas que serão utilizadas para determinar o nível de habilidades, conhecimentos e experiências possuídas por uma pessoa ou requeridas por uma posição é feito através da tela mostrada na figura 5.2. Para cada escala devem ser fornecidos um nome, um tipo, se ordinal ou nominal (FENTON e PFLEEGER, 1997), e os itens que a compoirão, no caso das escalas nominais. Cada item, por sua vez, deve conter um valor e uma descrição associados.



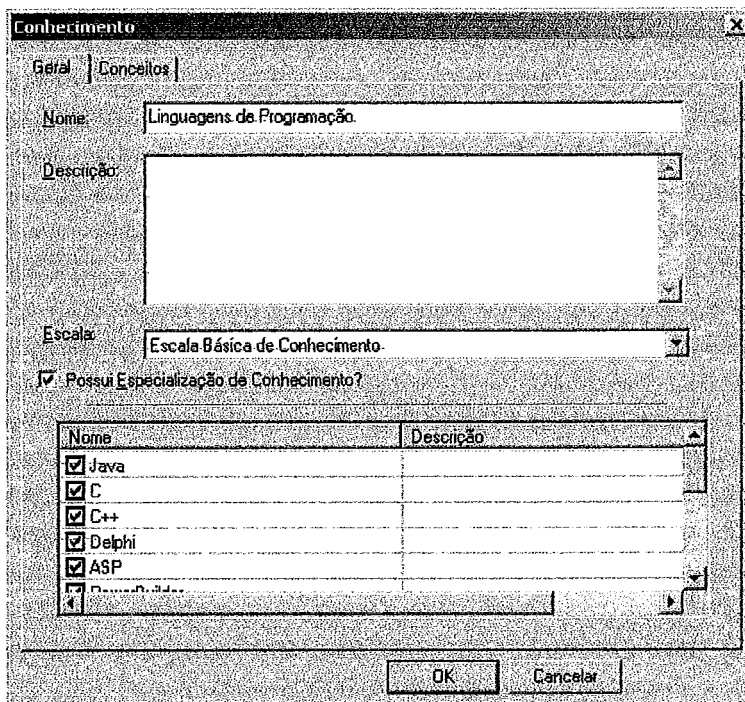
**Figura 5.2 – Cadastro de Escalas**

Os dados necessários para a inclusão de uma nova habilidade são: nome, descrição e a escala à qual está associada, como pode ser visto através da figura 5.3.



**Figura 5.3 – Cadastro de Habilidades**

Para o cadastro de conhecimento, os dados necessários são: nome, descrição e a escala à qual está associado. Caso o conhecimento sendo cadastrado possua sub-itens, ele será considerado uma categoria de conhecimento. Uma categoria de conhecimento não pode ser associada a uma pessoa ou uma posição, apenas os seus itens podem. Este cadastro pode ser visto através da figura 5.4a.

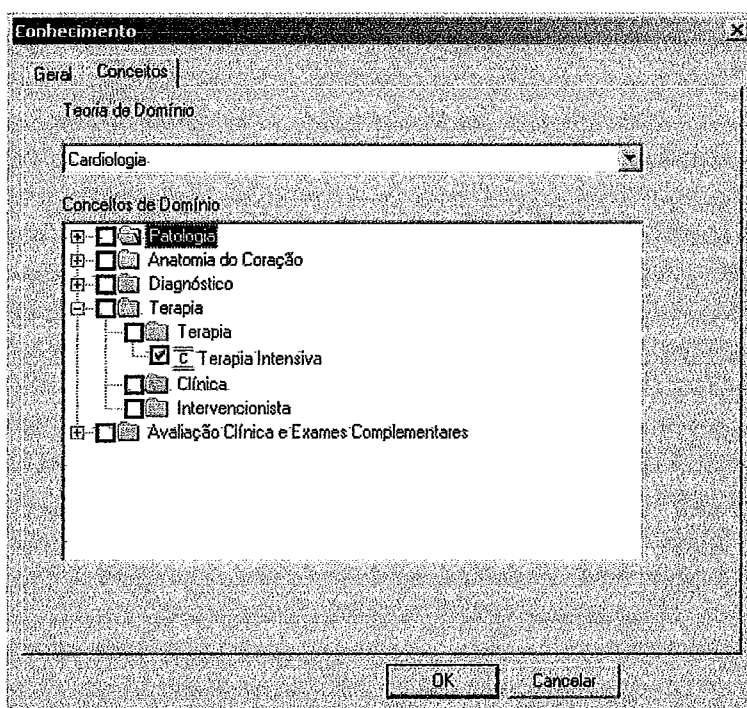


**Figura 5.4a – Cadastro de um novo conhecimento**

Quando o conhecimento estiver relacionado a um conceito de um domínio de aplicação (representado por uma ontologia de domínio), a associação ao conceito e a suas instâncias será feita conforme mostra a figura 5.4b. Os conceitos exibidos estão presentes

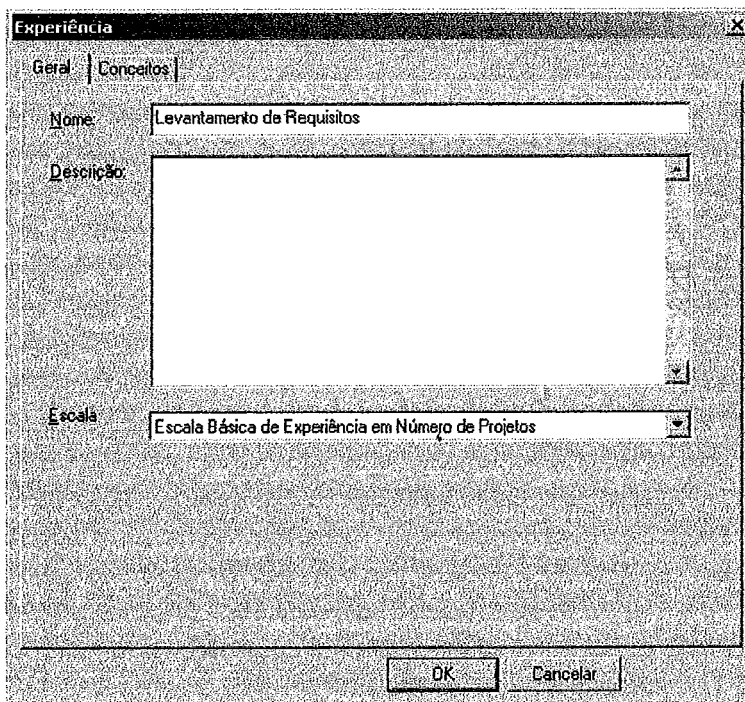
nas teorias de domínio cadastradas na Estação TABA através da ferramenta **EdiTeD** (OLIVEIRA, 1999). Para cada conceito é possível escolher quais exemplos são mais característicos daquele conhecimento. Estes exemplos podem ser cadastrados pela ferramenta **Genesis** (GALOTTA, 2000) ou podem ser, ainda, instâncias de classes que fazem parte do modelo de dados da Estação TABA e foram originadas de conceitos de alguma teoria de domínio (como, por exemplo, atividades, processos, áreas de negócio etc).

Por exemplo, pode se relacionar um conhecimento “Realização de Cirurgia Cardíaca” ao conceito “Cirurgia” da ontologia de Cardiologia e a uma de suas instâncias: “Cirurgia Cardíaca”. Essa associação dupla é feita porque uma associação apenas com o conceito (no caso “Cirurgia”) teria uma abrangência muito ampla e poderia não corresponder exatamente ao desejado (no caso “Cirurgia Cardíaca”).



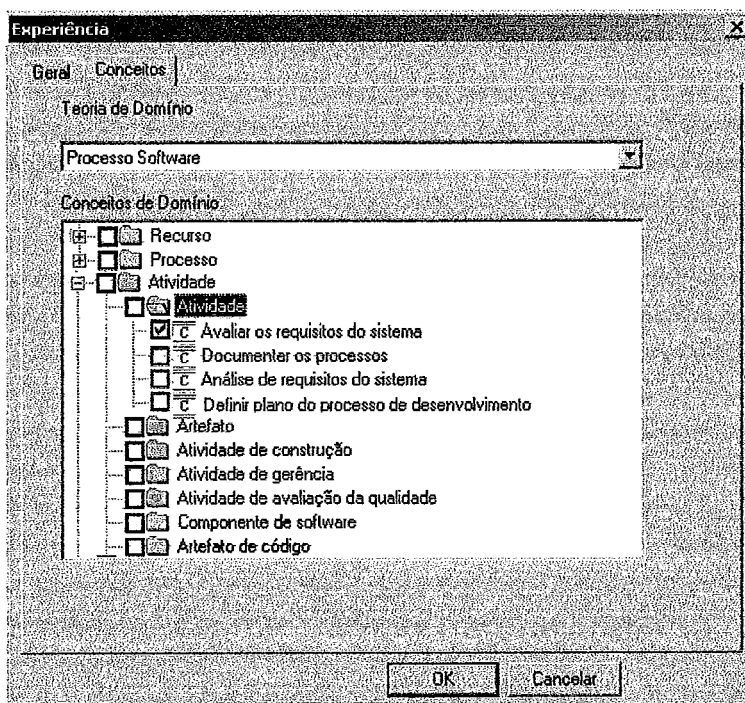
**Figura 5.4b – Cadastro de um novo conhecimento**

O cadastro de experiências é feito de forma semelhante ao cadastro de conhecimento, como pode ser visto pela figura 5.5a. Os dados necessários são: nome, descrição e a escala à qual está associada.



**Figura 5.5a – Cadastro de Experiência**

Quando a experiência estiver relacionada a um conceito de um domínio de aplicação (representado por uma ontologia de domínio), a associação ao conceito e a suas instâncias será feita de forma semelhante à associação realizada com um conhecimento (veja figura 5.5b).

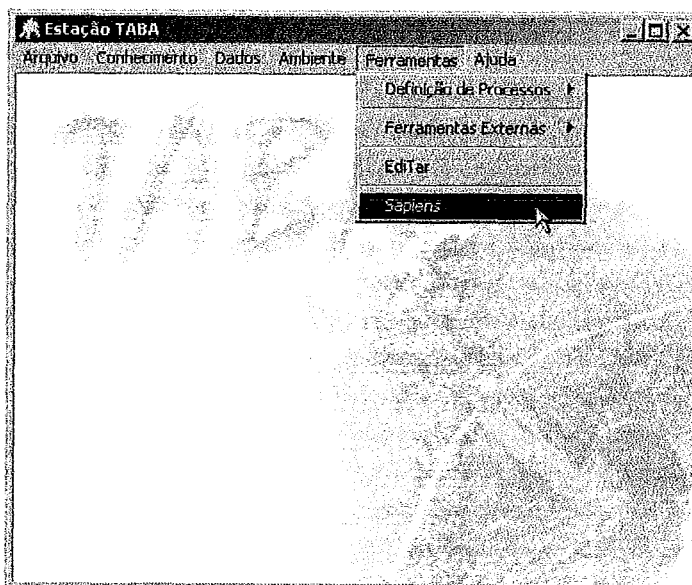


**Figura 5.5b – Cadastro de Experiência**

Os conceitos exibidos (ver figura 5.5b) estão presentes nas teorias de domínio cadastradas na Estação TABA através da ferramenta **EdiTeD** (OLIVEIRA, 1999). Para cada conceito é possível escolher quais exemplos são mais característicos daquele conhecimento. Estes exemplos podem ser cadastrados pela ferramenta **Genesis** (GALOTTA, 2000) ou podem ser, ainda, instâncias de classes que fazem parte do modelo de dados da Estação TABA e foram originadas de conceitos de alguma teoria de domínio (como, por exemplo, atividades, processos etc). Por exemplo, pode se relacionar uma experiência “Experiência em Análise de Requisitos” estaria associada ao conceito “Atividade” da ontologia de Processo de *Software* e à sua instância “Análise de Requisitos”.

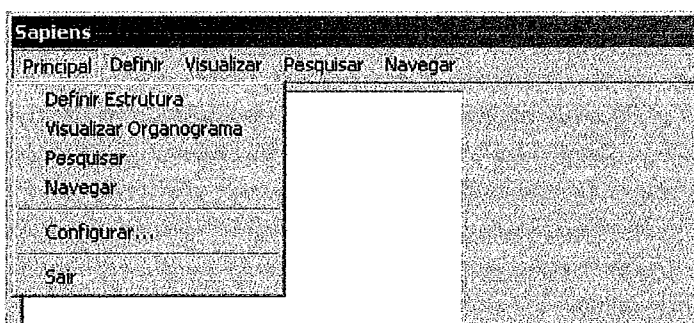
## Executando SAPIENS

Uma vez que as habilidades, conhecimentos e experiências já tenham sido descritos é possível utilizar a ferramenta **SAPIENS** para definir a estrutura da organização. A ferramenta pode ser executada através do menu Ferramentas do meta-ambiente da Estação TABA (ver figura 5.6).



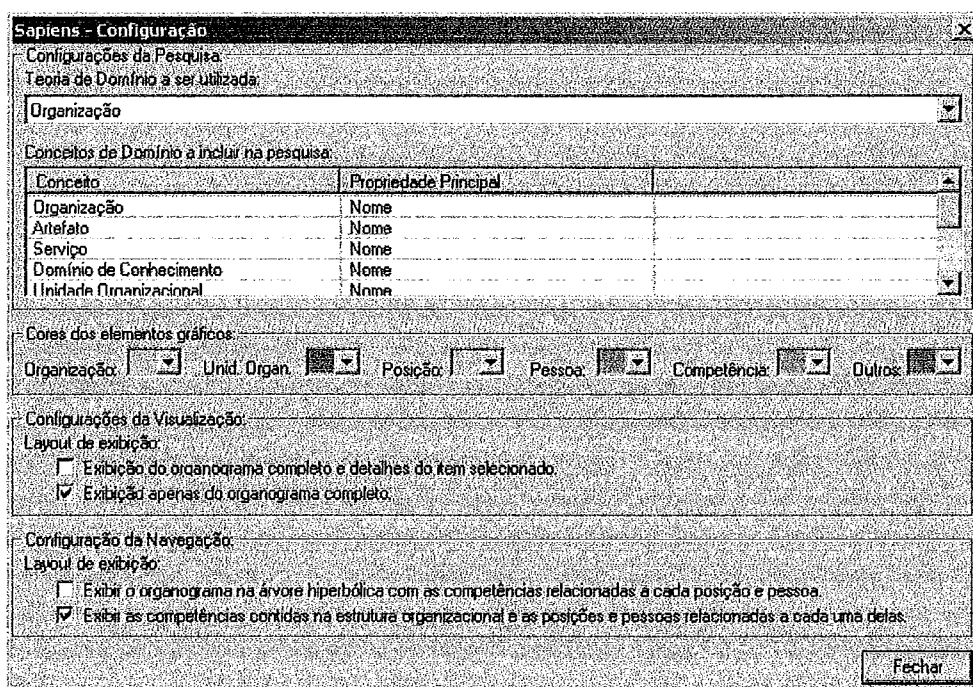
**Figura 5.6 – Menu Ferramentas e acesso a SAPIENS**

**SAPIENS** possui quatro módulos principais e um auxiliar que podem ser acessados a partir do menu da sua tela principal (figura 5.6): definição da estrutura organizacional, visualização da estrutura organizacional, pesquisa, navegação sobre a base de dados e configuração.



**Figura 5.6 – Detalhe da Tela Principal da ferramenta SAPIENS**

Para o correto funcionamento da ferramenta é preciso que alguns dados sejam previamente informados, como pode ser visto na figura 5.7. Deve ser informado o nome da teoria de domínio, inserida através da ferramenta **EdiTeD** (OLIVEIRA, 1999), cujos conceitos foram utilizados como base para a definição do modelo de dados (neste caso, “Organização”) e qual a propriedade principal de cada conceito desta teoria. Estas informações serão utilizadas durante a realização da pesquisa e da exibição de seu resultado. Outras configurações incluem as cores a serem utilizadas pelos elementos gráficos a serem exibidos pela ferramenta e a forma de exibição preferencial das telas de visualização da estrutura organizacional e navegação.



**Figura 5.7 – Tela de Configuração da ferramenta SAPIENS**



## Definição da Estrutura da Organização

A definição da estrutura da organização é o primeiro módulo da ferramenta. Para acessá-lo basta selecionar o item “Definir Estrutura” do menu “Principal” (ver Figura 5.6).

A interface com o usuário é constituída de dois painéis. Do lado esquerdo da tela é apresentada uma estrutura de árvore que representa a configuração atual da estrutura da organização e é atualizada de acordo com a inclusão de novos itens. No painel existente no lado direito é possível editar o item selecionado na árvore do lado esquerdo (ver Figura 5.8).

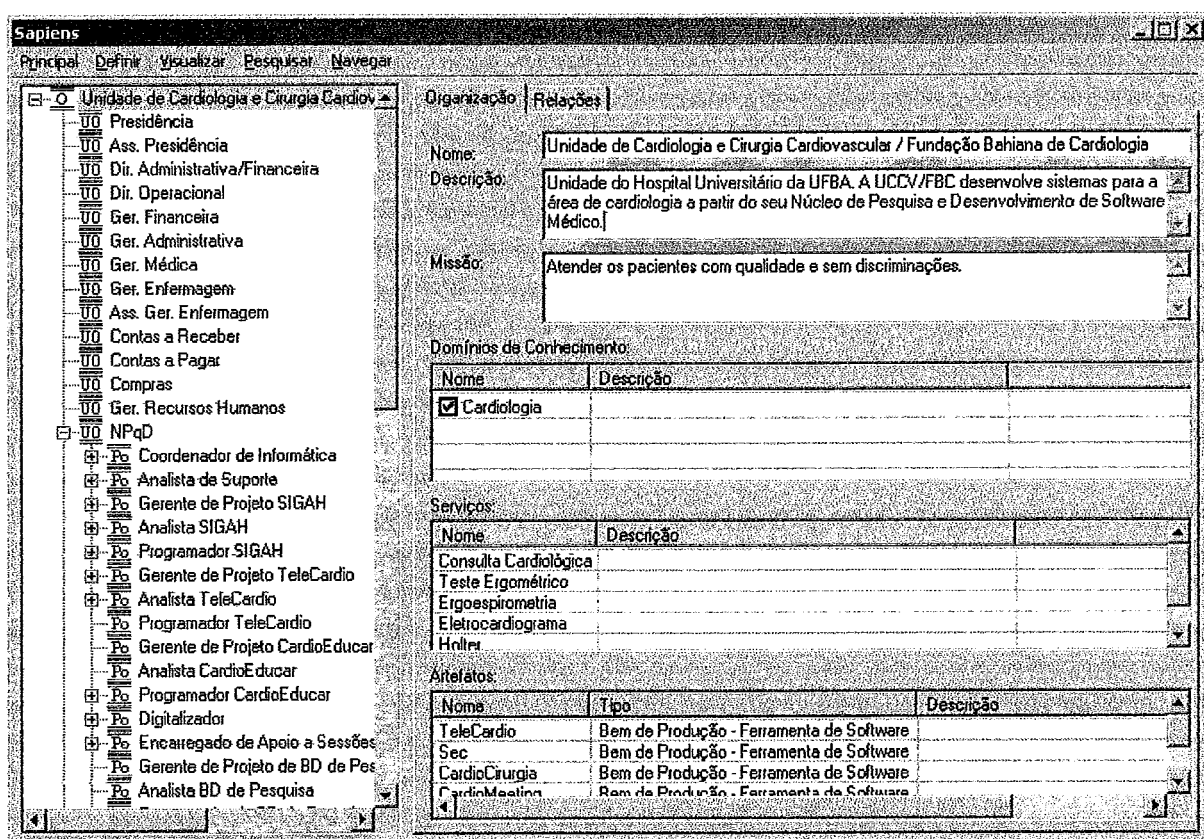


Figura 5.8 – Definição da Estrutura da Organização

A classe referente aos dados da Organização no modelo da Estação Taba servia única e exclusivamente à definição de processos padrão de uma organização através da ferramenta DEFPRO (MACHADO, 2000). Continha apenas um nome, uma descrição e uma lista de processos padrões associados. Com a ferramenta SAPIENS, novos atributos foram acrescentados: uma descrição da missão, potenciais e deficiências da organização, além de uma lista de produtos, cargos e áreas de negócio associadas. Quando o usuário

clique sobre o nome de uma organização é possível alterar suas características, como pode ser visto na Figura 5.8 que exibe a descrição da estrutura da UCCV/FBC.

Há ainda a possibilidade de definir como são as relações entre as unidades organizacionais e posições que compõem a estrutura organizacional (ver Figura 5.9). Estas relações serão utilizadas, posteriormente, para a exibição do organograma da organização em questão. Para cada relação é necessário especificar o tipo (que pode ser, “Reporta” e “Interage” para unidades organizacionais e “Reporta” para posições). Para cada uma das relações definidas é, ainda, necessário especificar uma descrição.

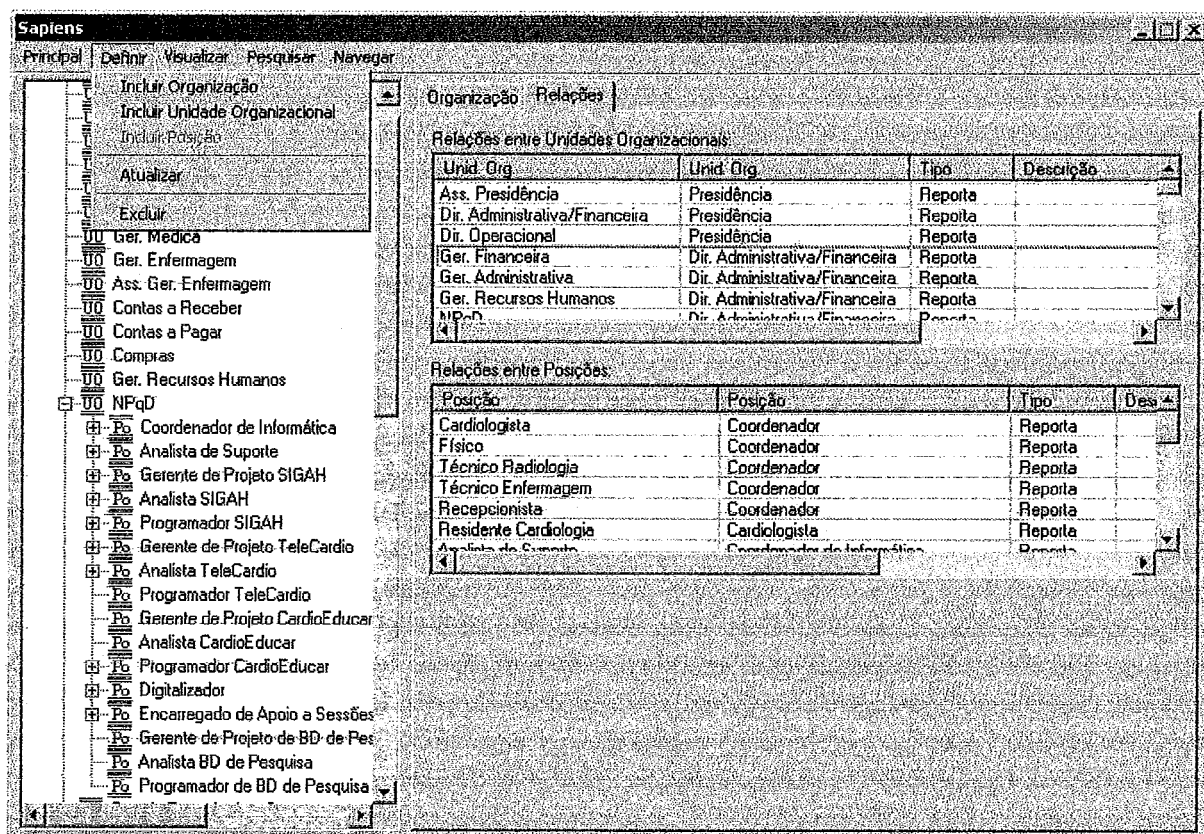
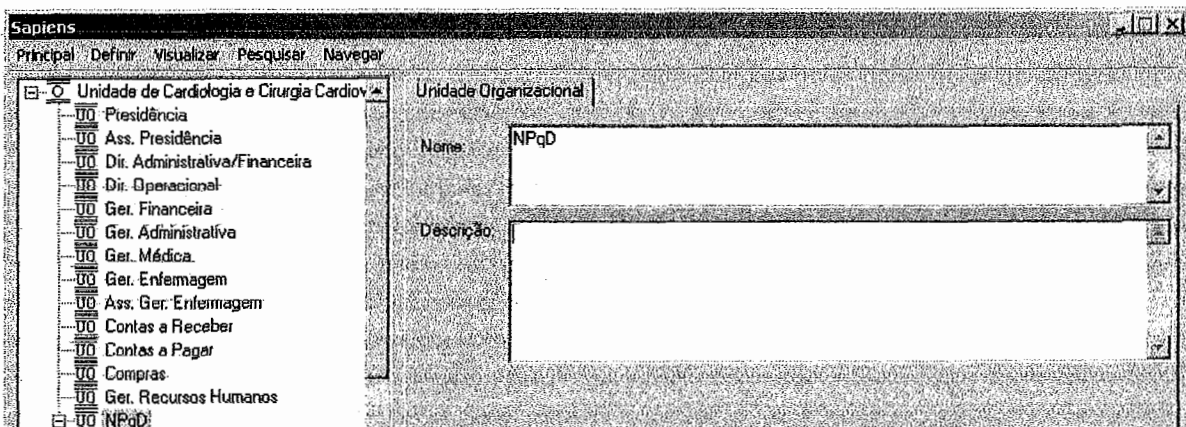


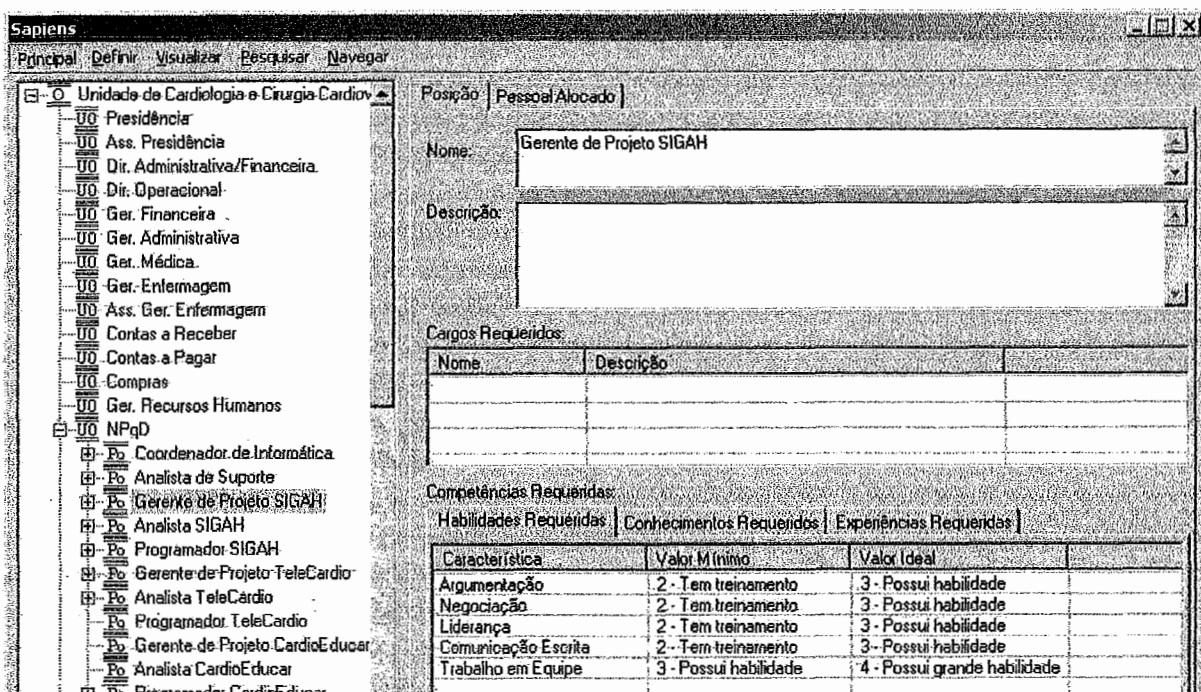
Figura 5.9 – Menu Definir e Definição de Relações

A inclusão de uma nova unidade organizacional pode ser feita através do item “Incluir Unidade Organizacional” do menu “Definir” (ver Figura 5.9). Através deste menu é possível, também, excluir o item selecionado através do item “Excluir” ou forçar uma atualização através do item “Atualizar”.



**Figura 5.10 – Detalhe da definição de Unidade Organizacional**

Os dados requeridos para descrever uma unidade organizacional são nome e descrição (ver Figura 5.10). Para cada unidade organizacional pode-se definir as posições que a compõe. Para isso, após selecionar uma unidade organizacional no painel do lado esquerdo da tela deve-se clicar no item 'Incluir Posição' do menu "Definir" (ver Figura 5.9).



**Figura 5.11 – Detalhe da definição de Posição**

Os dados requeridos para posições são nome e descrição (ver Figura 5.11). Deve-se escolher, dentre os conhecimentos, habilidades e experiências existentes, quais os necessários ou relevantes para o desempenho daquela posição. Para cada habilidade, e cada conhecimento e experiência, associada a uma determinada posição deve-se escolher um

nível ideal e um nível mínimo necessário para a sua realização. Este nível varia de acordo com a escala associada à habilidade.

Também se deve escolher o pessoal alocado àquela posição (ver Figura 5.12). Para cada pessoa é preciso definir o período em que ela ficou alocada àquela posição. Múltiplas alocações para uma mesma pessoa são permitidas. Para cada pessoa, também devem ser especificados um nome, e-mail e cargo que ela ocupa na organização.

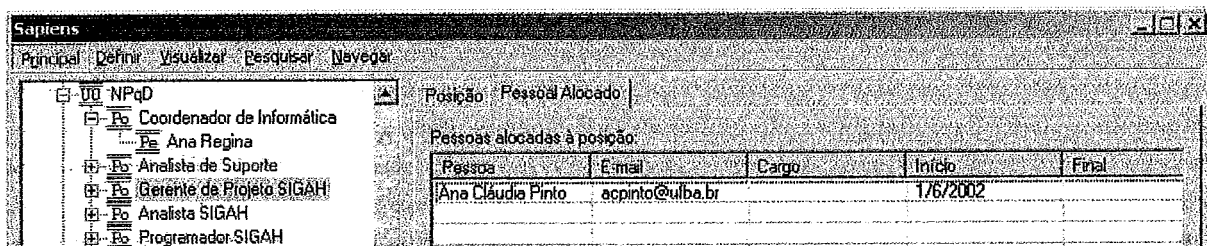


Figura 5.12 – Detalhe da alocação de pessoal a uma Posição

Uma vez que alguém foi associado a uma posição, um ícone referente à ela será exibido na árvore do lado esquerdo da tela e será possível, então, ao clicar sobre ele, especificar quais habilidades, conhecimentos e experiências ela possui. A cada associação deve ser atribuído um nível que varia de acordo com a escala associada à habilidade, conhecimento ou experiência em questão (ver Figura 5.13).

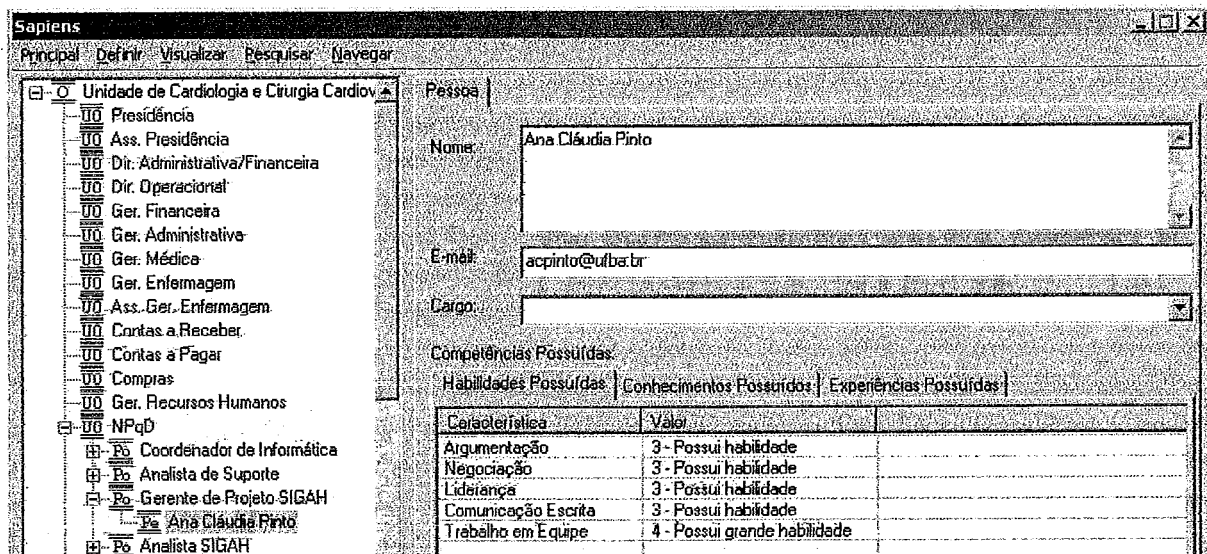


Figura 5.13 – Detalhe da definição de Pessoa

## Organograma

Uma vez que a descrição da organização tenha sido fornecida é possível visualizar a sua estrutura através de um organograma. Para isso o usuário deve selecionar o item 'Visualizar Organograma' conforme indica a figura 5.6 e indicar qual a organização da qual se deseja ver a estrutura. Há três formas de visualização do organograma disponível: exibição de estrutura de unidades organizacionais e posições de forma independente e exibição de detalhes (ver figura 5.14), exibição de estrutura de unidades organizacionais e posições sem opção de detalhes e uma mais simplificada que exhibe apenas o organograma completo (ou seja, a estrutura de unidades organizacionais).

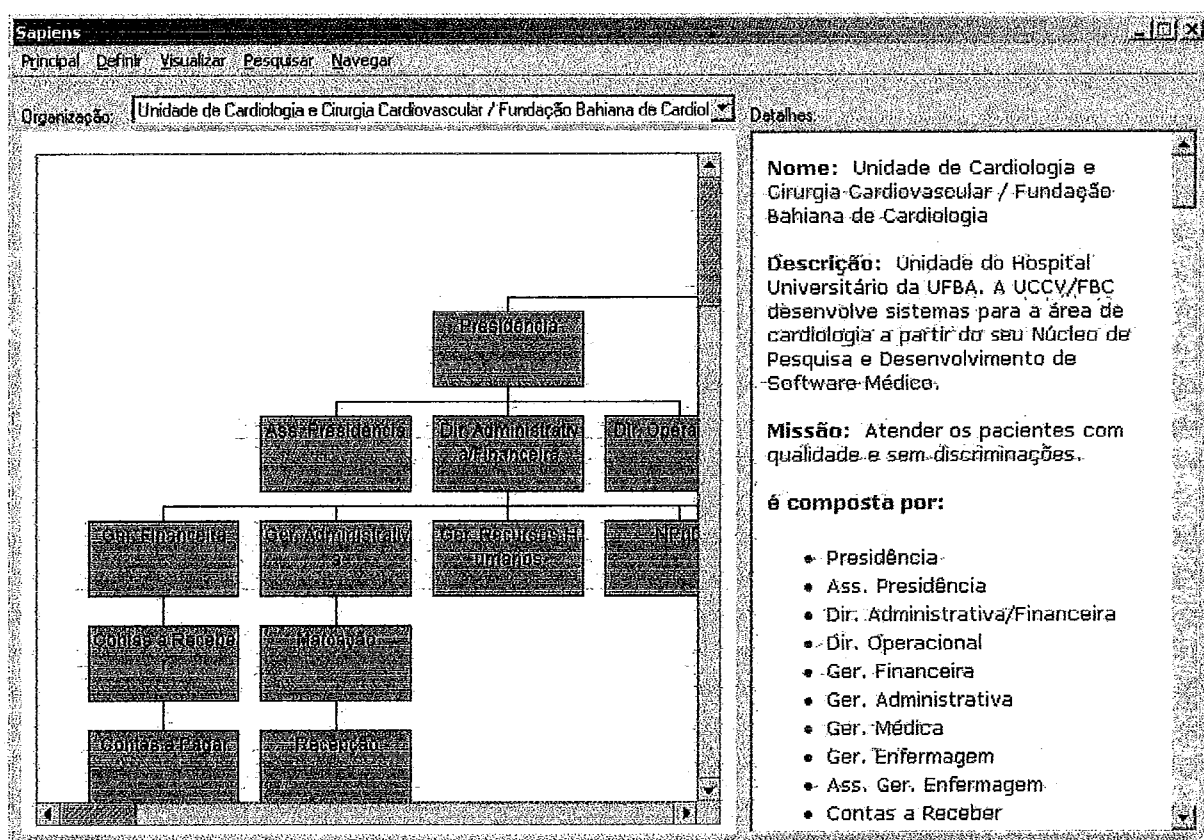


Figura 5.14 – Organograma e detalhe de uma unidade organizacional

No organograma estão graficamente representadas as unidades organizacionais e seus relacionamentos. Quando se seleciona uma unidade organizacional é possível ver sua estrutura, com as posições que possui e os relacionamentos entre elas.

Um painel na parte lateral esquerda da tela é utilizado para exibir as informações sobre o item selecionado, quer seja uma organização, unidade organizacional ou posição. As informações sobre a organização incluem nome, descrição, potenciais, deficiências,

missão, produtos, áreas de negócio e cargos. Também são listadas as relações entre unidades organizacionais existentes. Para cada unidade organizacional são listados nome, descrição, posições associadas, além das relações entre estas posições. Para as posições são listados nome, descrição, pessoal alocado (junto com o período de alocação de cada pessoa e as relações existentes entre elas), as habilidades, conhecimentos e experiências requeridos. Para cada pessoa são listados nome, e-mail, cargo e habilidades, conhecimentos e experiências possuídas.

Todas as informações listadas são baseadas nas relações definidas na ontologia de Organização, de onde as classes que formam a infra-estrutura da ferramenta **SAPIENS** são derivadas. Cada classe está associada a um conceito desta ontologia o qual possui propriedades e relações com outros conceitos. A partir destas informações é montado um código HTML com as informações pertinentes e, então, este código HTML é exibido na janela.

## **Pesquisa**

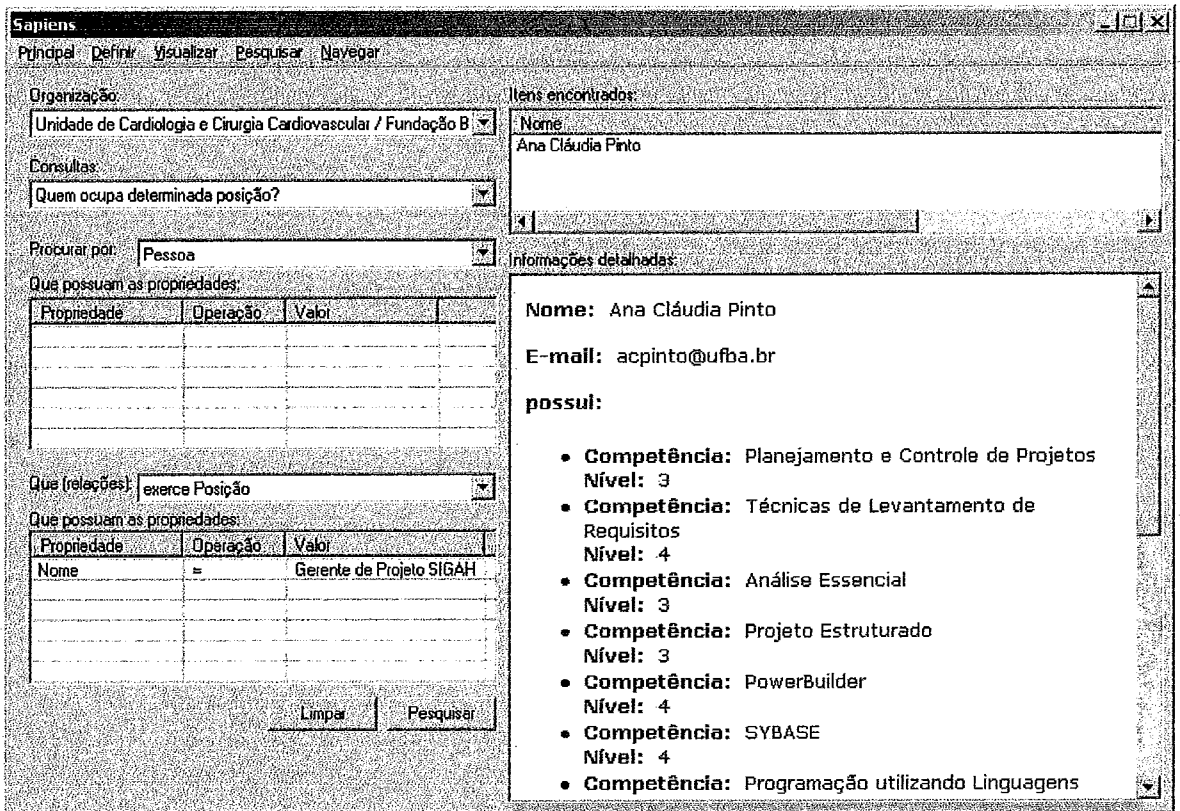
É possível fazer pesquisas sobre a base de dados da organização escolhendo a opção 'Pesquisar' do menu "Principal" da ferramenta **SAPIENS** (figura 5.6).

Há uma série de consultas previamente cadastradas. Estas consultas se baseiam nas relações existentes entre os conceitos que compõem a ontologia de organização. Cada consulta pré-cadastrada contém uma descrição (exibida na caixa indicada por "Consultas", como indica a figura 5.15), um item a ser procurado (exibido na caixa indicada por "Procurar por") e um item relacionado a este (exibido na caixa indicada por "E (relações)"). Caso o usuário não deseje realizar uma das consultas listadas, são exibidos todos os conceitos existentes na ontologia de organização. Ao escolher um dos conceitos, são listadas todas as relações envolvendo-o. A vantagem da consulta pré-cadastrada é o fato de o usuário não precisar conhecer a estrutura da ontologia de organização, já que cada consulta tem uma descrição que indica o seu objetivo.

As consultas atualmente pré-cadastradas são:

- Quem possui uma determinada competência?
- Quem ocupa determinado cargo?
- Quem ocupa determinada posição?
- Quais as competências de uma determinada pessoa?

- Quais as competências requeridas por uma determinada posição?
- Em que posições uma determinada competência é requerida?
- Que cargos existem na organização?



**Figura 5.15 – Pesquisa sobre os dados da base de dados da organização**

O usuário filtrar os dados a serem retornados pela pesquisa especificando critérios tanto para o item procurado quanto para o relacionado. A consulta exibida na figura 5.15, por exemplo, retorna todas as pessoas que estão alocadas à posição de “Gerente de Projeto SIGAH”. A lista de critérios apresentados ao usuário está baseada nas propriedades dos conceitos envolvidos na busca (no caso, a propriedade “Nome” do conceito “Posição”). As operações disponíveis variam de acordo com o tipo da propriedade em questão. Para propriedades do tipo booleanos as operações são “Igual” ou “Diferente”. Para propriedades dos tipos numéricos (números inteiros ou reais) as operações são “Igual”, “Diferente”, “Menor”, “Menor ou igual”, “Maior” ou “Maior ou igual”. Para as propriedades do tipo *string* as operações são “Igual”, “Diferente”, “Menor”, “Menor ou igual”, “Maior”, “Maior ou igual”, “Contém” ou “Não contém”.

Uma lista com os itens encontrados será exibida. Quando o usuário clicar sobre algum destes itens seus detalhes serão exibidos. A exibição dos detalhes dos dados (como

pode ser visto no painel do lado direito da tela) foi feita seguindo os mesmos moldes da realizada com os detalhes dos organogramas.

O algoritmo abaixo descreve como a pesquisa é realizada:

**Consulta:** Quem (que pessoa) exerce determinada posição?

**Conceito procurado:** < pessoa >

**Conceito relacionado:** < posição >

**Relação entre conceitos:** < pessoa > exerce < posição >

**Algoritmo:**

1. Encontre os objetos relacionados à organização atual que representam o conceito procurado (neste caso, < pessoas >)
2. Procure dentre objetos encontrados no passo anterior os que tenham as características desejadas
3. Encontre os objetos relacionados à organização atual que representam o conceito relacionado (neste caso, < posições >)
4. Encontre as posições relacionadas à organização atual
5. Procure dentre objetos encontrados no passo anterior os que tenham as características desejadas
6. Verifique se os objetos encontrados nos passos 2 e 4 estão relacionados de acordo com a relação escolhida (no caso "< pessoa > exerce < posição >")
7. Gere o resultado final da pesquisa com base nos objetos encontrados (no caso uma lista de < pessoas >)

## Navegação

É possível navegar sobre o conteúdo da base de dados da organização através da funcionalidade 'Navegar' do menu "Principal" da ferramenta **SAPIENS** (figura 5.6). Para que isso possa ser realizado é necessário que a estrutura da organização já tenha sido definida pelo usuário. Esta navegação explora a característica de hipertexto dos dados coletados. As relações entre os diversos itens que compõem a base de dados são exploradas para exibir ao usuário todas as informações relacionadas.

É exibida uma estrutura de árvore hiperbólica<sup>6</sup> com os dados coletados sobre a organização (figura 5.16). O nó raiz inicial é a própria organização. A partir dela o usuário pode navegar sobre suas relações com os outros itens da base de dados. Ao clicar sobre algum desses itens, seus dados são exibidos na parte lateral direita da tela e, então, o item se torna o novo nó raiz da árvore e fica em destaque. Esta representação da árvore explora as características das árvores hiperbólicas descritas na seção 4.4.4 do capítulo IV. Há duas possibilidades de exibição: a primeira exibe o organograma na árvore hiperbólica com as

---

<sup>6</sup> A implementação da árvore hiperbólica utilizada pelo **SAPIENS** foi feita em Java™ e é baseada no código fonte que pode ser encontrado no *site* <http://sourceforge.net/projects/hypertree/>.





# CAPÍTULO VI - CONSIDERAÇÕES FINAIS

## 6.1 Introdução

Após apresentar uma revisão da literatura sobre Gestão de Conhecimento e sobre a Estação TABA e os Ambientes de Desenvolvimento de Software Orientados a Organização, foi apresentada a fundamentação teórica para a construção de um sistema que permitisse representar a distribuição do conhecimento, habilidades e experiências através da estrutura organizacional. Por fim, comentou-se a implementação da ferramenta SAPIENS, junto com um exemplo de uso, realizada dentro do contexto do ambiente Cordis-FBC. Para finalizar, este último capítulo apresenta as considerações finais e trabalhos futuros que podem ser realizados.

Este capítulo é composto das três seções a seguir:

- **Introdução:** contém a apresentação deste capítulo;
- **Conclusão e contribuições:** apresenta as conclusões e as contribuições obtidas no decorrer e ao final do desenvolvimento dessa tese e da ferramenta desenvolvida.
- **Perspectivas Futuras:** nesta última seção, são apresentadas as perspectivas futuras deste trabalho, aspectos que podem ser mais bem aprofundados e possíveis melhorias na ferramenta construída.

## 6.2 Conclusão e Contribuições

O objetivo desta tese é descrever a representação da distribuição das habilidades, conhecimentos e experiências possuídos por cada indivíduo através da estrutura de uma organização. A proposta se baseia em idéias propostas no âmbito de Gestão de Conhecimento, da Estação TABA e dos Ambientes de Desenvolvimento de *Software Orientados a Organização*.

Espera-se que essa abordagem possibilite a criação de uma cultura de identificação, aquisição e disseminação do conhecimento existente na Organização que a levará a se conhecer melhor e tirar maior proveito de seu potencial.

Como decorrência dessa proposta, as características de uma ferramenta para tal foi apresentada. A ferramenta apresentada parece ser genérica o suficiente para que seja independente de uma organização ou domínios específicos. Como está inserida no contexto da Estação TABA, beneficiou-se de alguns elementos propostos para compor a infra-estrutura básica definida para os ADSOrgs (VILLELA *et al.*, 2000) (VILLELA *et al.*, 2001b): faz uso de ontologias de organização, do domínio e de Engenharia de *Software* para descrever as organizações que desenvolvem e mantêm *software*.

Foram descritas: representação da estrutura organizacional (unidades que a compõe e como estas se relacionam quanto à distribuição de autoridade e responsabilidade), descrição das unidades organizacionais (que posições são desempenhadas e que conhecimentos, habilidades e experiências são requeridos) e a associação dos profissionais da organização às posições que desempenham (incluindo os conhecimentos, habilidades e experiências possuídos).

A descrição da organização (unidades organizacionais e posições existentes), além da informação sobre os funcionários atuais, podem servir de fonte de consulta para uma grande gama de profissionais. Por exemplo, os dados sobre as habilidades, conhecimentos e experiências possuídos por cada pessoa além da alocação dessas pessoas às posições serão mais úteis ao engenheiro de *software* responsável por definir a alocação de pessoal em um projeto. Ao departamento de Recursos Humanos, por outro lado, será prioritária a informação sobre alocação de pessoal às posições dentro das unidades organizacionais.

Foram descritas formas de visualização da estrutura organizacional e navegação através dela, bem como formas de pesquisas sobre o conteúdo da base de dados através de mecanismo que permitam identificar de forma direta ou indireta pessoas detentoras de determinado conhecimento, habilidade ou experiência.

Dentre as contribuições deste trabalho podemos destacar:

- A construção da ferramenta **SAPIENS** que permite a descrição da estrutura de qualquer tipo de organização e a distribuição de conhecimentos, habilidades e experiências ao longo desta estrutura;
- A utilização da árvore hiperbólica foi proposta como forma de visualização de estruturas hierárquicas com grande massa de dados, cujo uso pode ser estendido para outros tipos de dados além do aqui utilizado;
- A utilização do organograma foi proposta como forma de visualização de estruturas hierárquicas simples, cujo uso pode ser estendido para outros tipos de dados além dos aqui utilizados;
- A realização de pesquisas baseadas nas descrições dos conceitos e relações definidos pelas ontologias utilizadas como base para a definição de um modelo de classes. A infra-estrutura construída, por ser genérica, pode ser estendida para utilização com outros tipos de ontologia como, por exemplo, a de Processo;
- A extensão do modelo de dados da Estação TABA para que fosse possível representar:
  - A descrição da organização em unidades organizacionais e posições;
  - A definição das competências (ou seja, habilidades, conhecimentos e experiências) que são possuídas pelas pessoas ou requeridas para a correta realização das atribuições de uma posição;
  - A distribuição de habilidades, conhecimentos e experiências ao longo da estrutura organizacional.

### 6.3 Perspectivas Futuras

A validação de uma ferramenta como esta implica em sua utilização em várias organizações e projetos e excede em muito o tempo esperado para uma tese de mestrado. A validação será feita no contexto global do projeto ADSOrg. No âmbito desta tese foi apresentado um exemplo de uso, no ADS **Cordis-FBC** (VILLELA *et al.*, 2001b) em desenvolvimento na COPPE e UCCV/FBC/UFBA.

Algumas melhorias que poderão ser acrescentadas à ferramenta são listadas abaixo:

- Controle de acesso e modificação de informações: dados sensíveis da empresa não podem estar disponíveis a todos os usuários, impedir a modificação dos dados por pessoal não autorizado a fazê-lo;
- Estabelecer critérios e mecanismos para classificação da informação contida na base de dados (por exemplo, dados mais consultados seriam mostrados em primeiro lugar nas consultas realizadas);
- Criar um mecanismo para guardar um histórico da evolução do conhecimento adquirido por uma pessoa ao longo de um determinado tempo;
- Integrar **SAPIENS** a outras ferramentas da Estação TABA que necessitem das informações supridas por ela;
- Criar uma nova forma de definição da estrutura organizacional através de uma interface gráfica;
- Extensão da representação de organograma para refletir as diferentes formas de organização, por exemplo, organizações matriciais;
- Substituir a árvore hiperbólica por outra estrutura com as mesmas características, mas que permita a exibição de ciclos entre os itens;
- Implementar melhorias na funcionalidade de pesquisa permitindo a escolha dos dados a serem exibidos como resultado;
- Implementar melhorias na funcionalidade de pesquisa permitindo a alteração das pesquisas pré-definidas e a criação de novas;
- Possibilitar a pesquisa indireta por outro critério (por exemplo, similaridade de competências);
- Integrar o módulo de pesquisa às ferramentas de apoio à atividade de Investigação do Domínio (atualmente, **GENESIS** (GALOTTA, 2000) e **NAVEGUE** (GALOTTA, 2000)).

## REFERÊNCIAS BIBLIOGRÁFICAS

- ABECKER, A., BERNARDI, A., HINKELMANN, K. *et al.*, 1998, "Toward a Technology for Organizational Memories", *IEEE Intelligent Systems*, v. 13, n. 3 (May/June), pp. 40-48.
- ABECKER, A. BERNARDI, A., SINTEK, M, 1999, "Developing a Knowledge Management Technology: An encompassing view on know more, know-net and enrich", In: *Proceedings of the 8th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*, pp. 216-222.
- ACKERMAN, M., HALVERSON, C., 2000, "Reexamining Organizational Memory", *Communications of the ACM*, v. 43, n. 1 (Jan), pp. 59-64.
- ACUÑA, S. T., *et. al*, 2000, "*Software Engineering and Knowledge Engineering Software Process: Formalizing the Who's Who*", In: *The 12th International Conference on Software Engineering & Knowledge Engineering*, Chicago, Estados Unidos, Junho 2000, pp. 221-230.
- ARAÚJO, M. A. P., 1998, *Automatização do Processo de Desenvolvimento de Software nos Ambientes Instanciados pela Estação TABA*, Tesc de M. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- BARCELLOS, M.P., ROCHA, A.R., TRAVASSOS, G.H., 2001, "Planejamento de Custos em Ambientes de Desenvolvimento de *Software* Orientados à Organização", *Workshop de Teses – XV Simpósio Brasileiro de Engenharia de Software*, Rio de Janeiro – RJ, Outubro.
- BARNARD, C. I. 1971. *As Funções do Executivo*. São Paulo, Ed. Atlas. in ref (Chiavenato, 1998).
- BASIL, V., LINDVALL, M., COSTA, P., 2001, "Implementing the Experience Factory concepts as a set of Experience Bases", *SEKE'2001*, pp. 102-109, Buenos Aires, Argentina, Jun.
- BIGGAM, J., 2001, "Defining Knowledge: an Epistemological Foundation for Knowledge Management", In: *Proceedings of the 34th Hawaii International Conference on System Sciences*, Maui, Hawaii

- BUERGEL, H.D. and SAEUBERT, H., 1999, "The generation of control measures to determine the success of Knowledge Management", *Proceedings of the International Conference on Technology and Innovation Management, PICMET*, Vol. 1, pp. 126-127.
- CARLINER, S., 1999, "Knowledge Management, Intellectual Capital, and Technical Communication". *Communication Jazzy: Improvising The New International Communication Culture*. Proceedings 1999 IEEE International Professional Communication Conference. New Orleans, Setembro/1999, pp. 85-91.
- CHANDRASEKARAN, B., JOSEPHSON, J. R., BENJAMINS V. R., 1999, "What Are Ontologies, and Why Do We Need Them?", *IEEE Intelligent Systems & their applications*, v. 14, n. 1 (Jan/Feb), pp. 20-26.
- CHIAVENATO, I., 1998. *Introdução à Teoria Geral da Administração*, 5a edição, Editora Campus.
- DAVENPORTI, 'I'. 1996. *Some Principles of Knowledge Management*, Online: <http://www.bus.utexas.edu/kman/kmprin.htm> (verificado em março/2003).
- DIENG, R. *et al.*, 1999, "Methods and Tools for Corporate Knowledge Management", *Int'l J. Human-Computer Studies*, Vol. 51, No. 3, September, pp. 567-598. In ref: (Dieng, 2000).
- DIENG, R., 2000, "Knowledge Management and the Internet", *IEEE Intelligent Systems*, vol. 15, n.3 (May/June), pp. 14-17
- FALBO, R. A., 1998, *Integração de Conhecimento em um Ambiente de Desenvolvimento*, Tese de D. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- FALBO, R., MENEZES, C., ROCHA, A. R., 1999, "Assist-Pro: Um Assistente Inteligente para Apoiar a Definição de Processos de Software", In: *Anais do XIII Simposio Brasileiro de Engenharia de Software*, pp. 147-162, Florianópolis, Brasil, Out.
- FARIAS, L., 2002, *Planejamento de Riscos em Ambientes de Desenvolvimento de Software Orientados a Organização*, Tese de M. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- FARIAS, L., ROCHA, A.R., TRAVASSOS, G.H., 2001, "Producing Project Risk Plans in Enterprise-Oriented Software Development Environments", *14th International Conference Software & Systems Engineering and their Applications*, Paris, December.

- FATEEV, M., 1998, *O Meta-gerador de Ferramentas da Estação TABA*, Tese de M. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- FENTON, N. e PFLEEGER, S., 1997, *Software Metrics: A Rigorous and Practical Approach*. PWS Publishing Company, Boston.
- FERREIRA, A. B. H. 1975. *Novo Dicionário Aurélio da Língua Portuguesa*. 1ª ed. Rio de Janeiro, RJ, Editora Nova Fronteira S. A.
- FISCHER, G., 1994; “Domain-Oriented Design Environments”, *Automated Software Engineering - The International Journal of Automated Reasoning and Artificial Intelligence in Software Engineering*, Vol 1, N 2 (Jun), pp 177-203.
- FISCHER, G., 1996, “Seeding, Evolutionary and Reseeding: capturing and evolving knowledge in domain-oriented design environments”, In: Sutcliffe, A., Benyon, B. van Assche (eds) - IFIP 8. 1/13. *Joint Working Conference – Domain-Knowledge for Interactive System Design*, pp 1-16, Geneva, Switzerland, May.
- FOURO, A. M., 2002, *Apoio à Construção de Base de Dados de Pesquisa em Ambientes de Desenvolvimento de Software Orientados a Domínio*, Tese de M. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- GALOTTA, C., 2000, *Netuno: Um Ambiente de Desenvolvimento de Software Orientado ao Domínio de Acústica Submarinha*, Tese de M. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- GARG, P., JAZAYERI, M., 1995, “Introduction”. In: Garg, P., Jazayeri, M. (eds), *Process-Centered Software Engineering Environments*, chapter 1, IEEE Computer Society Press.
- GOMES, A.G.J., 2001, *Avaliação de Processos de Software Baseada em Medições*, Tese de M. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- GRUBER, T. R., 1995; “Toward Principles for the Design of Ontologies used for Knowledge Sharing”, *International Journal Human-Computer Studies*, No 43, pp 907-928.
- HENDRIKS, P., VRIENS, D., 1999, “Knowledge-based systems and knowledge management: Friends or foes?”, *Information & Management*, v. 35, n. 2 (Feb), pp. 113-125.
- HOUDEK, F., SCHNEIDER, K., 1999, “Software Experience Center: The Evolution of the Experience Factory Concept”. In: *Proceedings of the 24<sup>th</sup> Annual Software Engineering*



*Workshop*, SEL-Software Engineering Laboratory, Goddard Space Flight Center, NASA, Greenbelt, USA.

LAMPING, J., RAO, R., PIROLI, P., 1995, "A Focus+Context Technique Based on Hyperbolic Geometry for Visualizing Large Hierarchies", *Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems*, ACM Press.

MACHADO, L.F.C., 2000, *Modelo para Definição, Especialização e Instanciação de Processos de software na Estação TABA*, Tese de M. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.

MACHADO, L.F., SANTOS, G., OLIVEIRA, K.M., ROCHA, A.R., 2000a, "Def-Pro: uma Ferramenta para Apoiar a Definição de Processo Padrão", *XI Conferencia Internacional de Tecnologia de Software*, pp. 165-179, Curitiba, PR, junho.

MACHADO, L.F., SANTOS, G., OLIVEIRA, K.M., ROCHA, A.R., 2000b, "Def-Pro: Apoio automatizado para Definição de Processos de Software", *Caderno de Ferramentas - XIV Simpósio Brasileiro de Engenharia de Software*, pp. 359-362, João Pessoa, PB, outubro.

MAMANI-MACEDO, N.; LEITE, J., 2001, "Criando uma Memória Organizacional sob uma Abordagem de Ontologias", In: *Quarto Workshop Ibero-americano de Engenharia de Requisitos e Ambientes de Software (IDEAS'01)*, San José, Costa Rica, Abr, pp. 385-397.

MARKKULA, M. 1999. "The Impact of Intranet-based Knowledge Management on Software Development", *Federation of European Software Measurement Associations - FESMA99*, Amsterdam, Holanda, pp. 151-160.

MENDONÇA, M. G., SEAMAN, C.B., BASILI, V., KIM, Y., 2001, "A Prototype Experience Management System for a Software Consulting Organization", *Software Engineering and Knowledge Engineering - SEKE 2001*; Buenos Aires, Argentina, June.

MILTON, N., SHADBOLT, N., COTTAM, H., HAMMERSLEY, M., 1999, "Towards a knowledge technology for knowledge management", *Int'l J. Human-Computer Studies*, n. 51, pp. 615-641.

- MONTONI, M. A., ROCHA, A.R., 2002, “Aquisição de Conhecimento em Processos de Negócios”, *Workshop de Teses – XVI Simpósio Brasileiro de Engenharia de Software*, Gramado – RS, Outubro.
- MOURA, L. M. V., ROCHA, A. R. C., 1992, *Ambientes de Desenvolvimento de Software*, Publicações Técnicas COPPE/UFRJ, ES-271/92, Rio de Janeiro, Brasil.
- MURCH, R., 2000, *Project Management – Best Practices for it professionals*, Prentice Hall, pp. 203-210.
- O’LEARY, D. E., 1998a, “Knowledge Management Systems: Converting and Connecting”, *IEEE Intelligent Systems*, v. 13, n. 3 (May/Jun), pp. 30-33.
- O’LEARY, D. E., 1998b, “Using AI in Knowledge Management: Knowledge Bases and Ontologies”, *IEEE Intelligent Systems*, v. 13, n. 3 (May/Jun), pp. 34-39.
- O’LEARY, D. E., 1998c, “Enterprise Knowledge Management”, *IEEE Computer*, vol. 31 n.3 (March), pp.54-61
- O’LEARY, D. E., 2000, “Different Firms, Different Ontologies, and No One Best Ontology”, *IEEE Intelligent Systems*, v. 15, n. 5 (Set/Out), pp. 72-78.
- O’LEARY, D. E., STUDER, R., 2001, “Knowledge Management: An Interdisciplinary Approach”, *IEEE Intelligent Systems*, vol.16, n.1 (January/February), pp. 24-25.
- OH, E., van der HOEK, A., 2001, “Adapting Game Technology to Support Individual and organizational Learning”, In: *The 13th International Conference on Software Engineering & Knowledge Engineering - SEKE*, Buenos Aires, Argentina, Junho 2001, pp. 347-354.
- OLIVEIRA, K. M., 1999, *Modelo para Construção de Ambientes de Desenvolvimento de Software Orientados a Domínio*, Tese de D. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- OLIVEIRA, K. M., GALOTTA, C., ROCHA, A. R., *et al.*, 2000, “Construção de Ambientes de Desenvolvimento de Software Orientados a Domínio”, In: *Jornadas Iberoamericanas de Ingeniería de Requisitos y Ambientes de Software IDEAS 2000 – Cancún*, México, Abr.

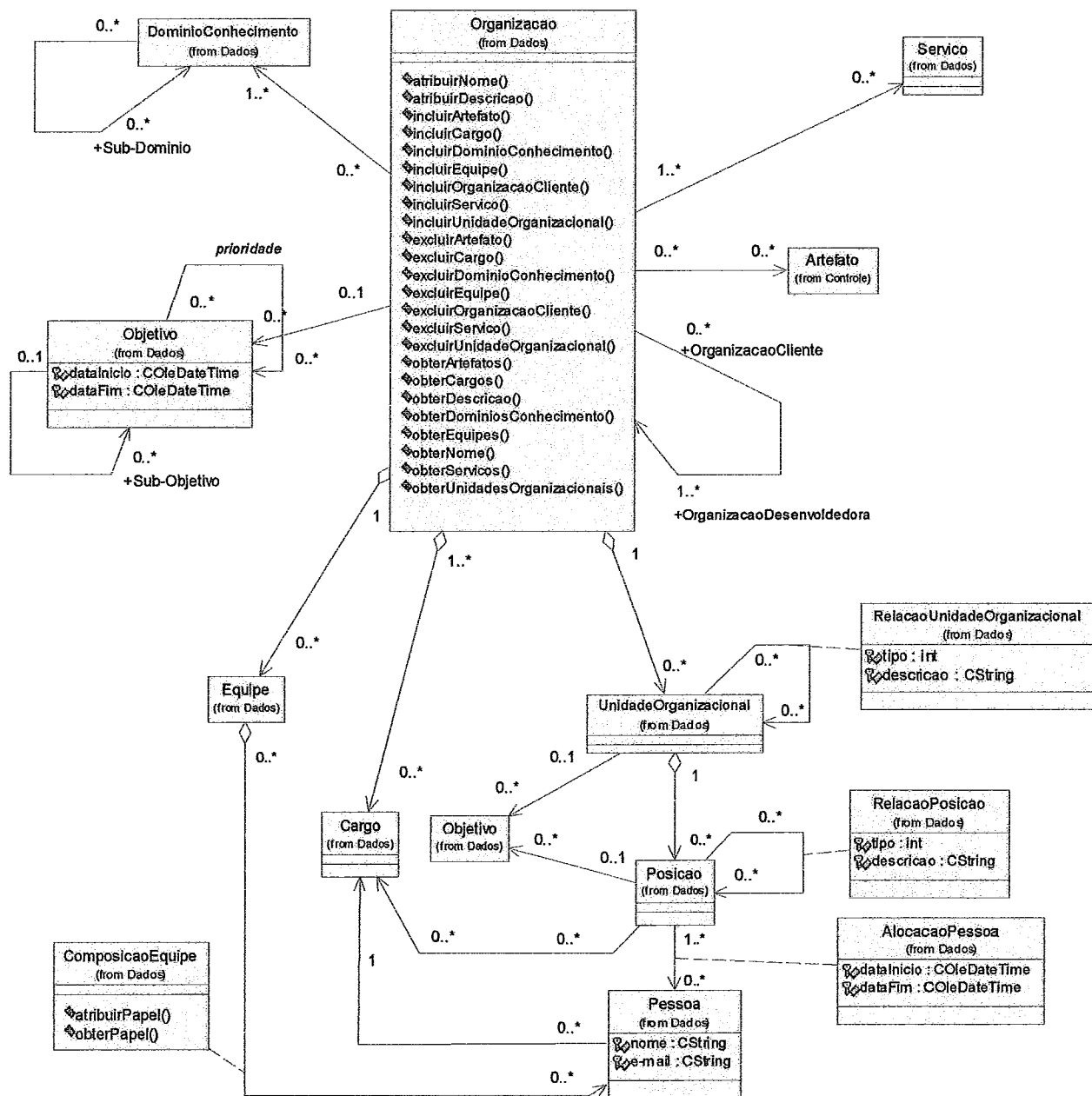
- OLIVEIRA, K. M., ROCHA, A. R., TRAVASSOS, G. H., 1999a, “A Domain-Oriented Software Development Environment for Cardiology”, In: *Proceedings of America Medical Informatics Association conference – AMLA*, Washington, D.C., Nov. (In press)
- OLIVEIRA, K. M., ROCHA, A. R., TRAVASSOS, G. H. *et al.*, 1999b, “Using Domain-Knowledge in Software Development Environments”, In: *Proceedings of the 11th International Conference on Software Engineering and Knowledge Engineering*, pp. 180-187, Kaiserlauter, Alemanha, Jun.
- OLIVEIRA, K. M., ROCHA, A. R., TRAVASSOS, G. H. *et al.*, 1999c, “O uso da Teoria do Domínio no Processo de Desenvolvimento de Software”, In: *Anais da X Conferencia Internacional de Tecnologia de Software*, pp 223-235, Curitiba, Brasil, Mai.
- OLIVEIRA, K. M., ROCHA, A. R., TRAVASSOS, G. H. *et al.*, 1998, *Ambientes de Desenvolvimento de Software Orientado a Domínio*, Publicações Técnicas, COPPE/UFRJ, ES 483/98, Rio de Janeiro, RJ, Brasil.
- PFLIEGER, S. L., 2001, *Software Engineering: Theory and Practice*, 2<sup>nd</sup> ed. Upper Saddle River, NJ: Prentice Hall.
- PIROLI, P., CARD, S.K. e WEGE, M.M.V.D, 2000, “The effect of information scent on searching information visualizations of large tree structures”, in: *Advanced Visual Interfaces*, AVI 2000, Palermo, Itália.
- PIROLI, P., CARD, S.K. e WEGE, M.M.V.D, 2001, “Visual information foraging in a Focus+Context visualization”, In: *Proceedings of CHI 2001*, v. 3, n. 1, Seattle, Washington, pp. 506-513.
- PREECE, A., FLETCHER, A., SLEEMAN, D., CURRY, D., MEANY, N., PERRY, P., 2001, “Better Knowledge Management through Knowledge Engineering”, *IEEE Intelligent Systems*, v. 16, n. 1, jan/fev, pp. 36-43.
- PROBST, G. J. B., RAUB, S., ROMHARDT, K., 1999, “Managing Knowledge: Building Blocks for Success”, *Wiley*.
- QINGRUI, X., JIANG, W., BAOHUA, X., 1999, “Knowledge: accumulation, diffusion and reactivation within organization”, *Proceedings of the Portland International Conference on Management of Engineering and Technology*, July, Vol. 1, pp. 486-487.

- ROCHA, A. R. C., AGUIAR, T. C., SOUZA, J. M., 1990, "TABA: A Heuristic Workstation for *Software* development", In: *Proceedings of COMPEURO 90*, Tel Aviv, Israel, Maio.
- RUHE, G., 1999, "Experience factory-based professional education and training", *Proceedings of the 12th Conference on Software Engineering Education and Training*, pp. 62-72.
- SANTOS, G., ROCHA, A.R., 2001, "Representação da Distribuição de Conhecimento e Habilidades através da Estrutura Organizacional", *Workshop de Teses – XV Simpósio Brasileiro de Engenharia de Software*, Rio de Janeiro – RJ, Outubro.
- SANTOS, G., ZLOT, F., 1999, *Definição e Instanciação e Ambientes na Estação TABA*, Projeto Final de Curso, UFRJ, Rio de Janeiro, RJ, Brasil.
- SANTOS, G., VILLELA, K, TRAVASSOS, G.H., ROCHA, A.R., 2002, "SAPIENS – Tool for Representing and Visualizing the Distribution of Knowledge and Skills throughout the Organizational Structure", *14th International Conference Software & Systems Engineering and their Applications*, Paris, December.
- SIVAN, Y. 2000. Knowledge Processes: "Driving Organizational Practices through Knowledge", *WebNet Journal*, v. 2, n. 4, out/dez, pp. 12-15, 44.
- SNOEK, B. 1999, *Knowledge Management and Organizational Learning*, Diploma Thesis, Fraunhofer/IESE, Kaiserslautern, Alemanha.
- STAAB, S., STUDER, R., SCHNURR, H., SURE, Y., 2001, "Knowledge Processes and Ontologies", *IEEE Intelligent Systems*, vol.16, n.1 (January/February), pp. 26-34.
- STADER, J. e MACINTOSH, A., 1999 "Capability modelling and knowledge Management". In *Applications and Innovations in Intelligent Systems VII*, Springer-Verlag, pp 33-50.
- SWARTOUT, W., TATE, A., 1999, "Ontologies – Guest Editors Introduction". *IEEE Intelligent Systems & their applications*, vol. 14, n. 1 (Jan/Fev), pp. 18-19.
- TIWANA, A., RAMESH, B., 2001, "Integrating Knowledge on the Web", *IEEE Internet Computing*, v. 5, n. 3 (May/June), pp. 32-39.

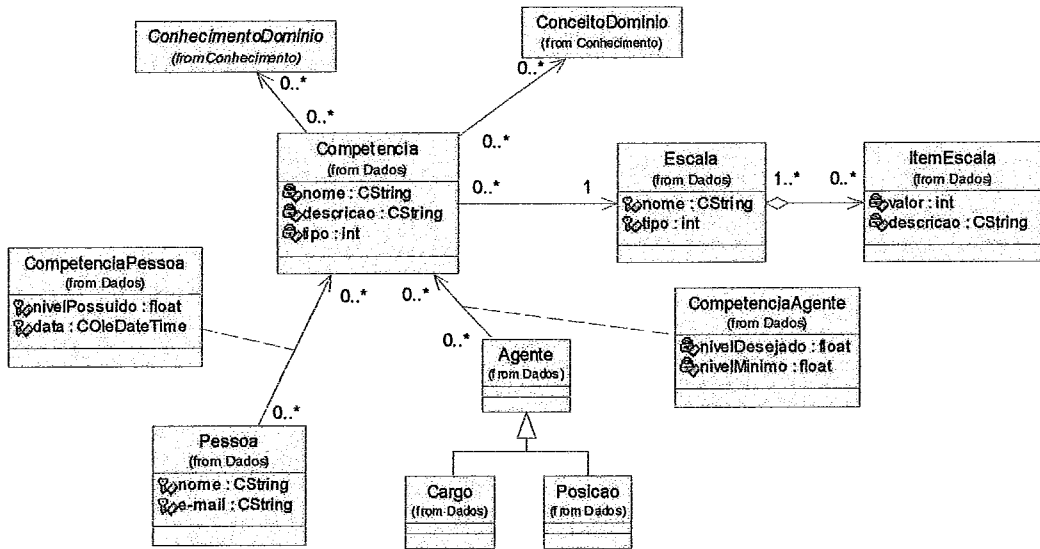
- TRAVASSOS, G. H., 1994, *O Modelo de Integração de Ferramentas da Estação TABA*, Tese de D. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- USCHOLD, M., GRUNINGER, M., 1996, "Ontologies: principles, methods and applications", *The Knowledge Engineering Review*, Vol 11:2, pp 93-136.
- van ELST, L., ABECKER, A., 2001, "Ontology-Related Services in Agent-Based Distributed Information Infrastructures", *Software Engineering and Knowledge Engineering - SEKE 2001*; Buenos Aires, Argentina, June, pp. 79-86.
- van HEIJST, G., van der SPEK, R., KRUIZINGA, E., 1996. "Organizing Corporate Memories", In: *Proc.10th Banff Workshop on Knowledge Acquisition for Knowledge-Based Systems (KAW' 96)*, Calgary, Canadá, ed. B. Gaines and M. Musen, SRDG Publications. in ref. (Dieng, 2000).
- VILLELA, K., ROCHA, A. R., TRAVASSOS, G.H., 2000, *Ambientes de Desenvolvimento de Software Orientados à Organização*, Publicação Técnica COPPE/UFRJ - ES530/00 Rio de Janeiro, RJ, Abril.
- VILLELA, K., SANTOS, G., GALOTTA, C., 2001a, "Estendendo a Estação TABA para a criação de Ambientes de Desenvolvimento de Software Orientados a Organização", *Caderno de Ferramentas - XV Simpósio Brasileiro de Engenharia de Software*, pp. 359-362, Rio de Janeiro, RJ, outubro.
- VILLELA, K., SANTOS, G., GALOTTA, C., 2001b, "Cordis-FBC: um Ambiente de Desenvolvimento de Software para Cardiologia", *I Workshop de Informática Médica*, Rio de Janeiro, RJ, outubro.
- VILLELA, K., SANTOS, G., BONFIM, C., *et al.*, 2001c, "Knowledge Management in Software Development Environments", *14th International Conference Software & Systems Engineering and their Applications*, Paris, dezembro.
- VILLELA, K., SANTOS, G., TRAVASSOS, G. H., ROCHA, A. R., 2002, "Gestão de Conhecimento em Ambientes de Desenvolvimento de Software", *2ª Jornada Ibero-Americana de Engenharia de Software e Engenharia de Conhecimento*, Salvador, Brasil, Outubro

- von WANGENHEIM, C.G., LICHTNOW, D., von WANGENHEIM, A, 2001, "A Hybrid Approach for Corporate Memory Management Systems in *Software R & D Organizations*", *Software Engineering and Knowledge Engineering - SEKE 2001*; Buenos Aires, Argentina, June.
- WERNECK, V. M., 1995, *Ambiente para Desenvolvimento de Software Baseado em Conhecimento*, Tese de D. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- WERNER, C. M. L., TRAVASSOS, G. H., DA ROCHA, A. C. *et al.*, 1997, "Memphis: A Reuse Based O. O. *Software Development Environment*", In: *Proceedings of TOOLS*, Beijing, China, Sep.
- WHITEHALL, B., 1998, "AAAI-97 Workshop on AI and Knowledge Management", *AI Magazine*, v. 19, n. 1 (Spring), pp. 130.
- WINCH, G., 1999, "Knowledge Management", *Manufacturing Engineer*, August, pp.178 – 180.
- WURMAN, R. S. 1989. *Information Anxiety: What to Do When Information Doesn't Tell You What You Need to Know*, New York: Doubleday. in ref. (Carliner, 1999).
- ZLOT, F., 2002, *Conhecimento de Tarefa em Ambientes de Desenvolvimento de Software Orientados a Domínio*, Tese de M. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- ZLOT, F. OLIVEIRA,K., ROCHA, A.R., 2002. "Modelling Task Knowledge to Support *Software Development*", *Software Engineering and Knowledge Engineering - SEKE'2002*, Ischia, Itália, Jul.

# ANEXO I - MODELO DE CLASSES UTILIZADO PARA DESCREVER A ESTRUTURA ORGANIZACIONAL



Classes utilizadas para descrever a estrutura da organização

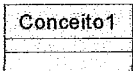
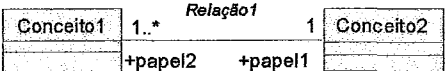
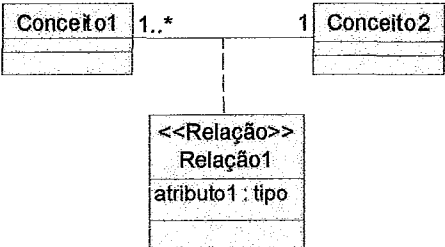
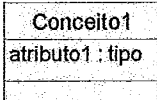
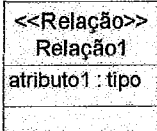
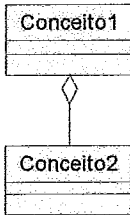


**Classes utilizadas para descrever a distribuição de competências através da estrutura organizacional**



## ANEXO II - NOTAÇÃO UTILIZADA PARA REPRESENTAR ONTOLOGIAS UTILIZANDO UML

A notação a seguir foi utilizada para representar a sub-ontologia de estrutura de Organização apresentada na seção 4 do capítulo IV utilizando a notação para modelos de classes na UML.

Ontologia	UML	Notação
Um conceito da ontologia é representado como uma...	... classe de modelo UML.	
Relações entre conceitos são representadas como...	... uma associação entre classes de um modelo UML.	
Uma relação entre conceitos que contenha propriedades é representada por...	... uma classe associativa de um modelo UML na qual é incluído o estereótipo <<Relação>>.	
Propriedades de conceitos são representadas como...	... atributos de classes.	
Propriedades de relações são representadas como...	... atributos de classes com o estereótipo <<Relação>>.	
Uma agregação é representada como...	... uma agregação de um modelo UML.	
Relações de hierarquia são representadas como...	... generalizações em um modelo UML.	