

Arquitetura para Gerenciamento de
Conhecimentos Explícitos sobre o Processo de
Desenvolvimento de Produto

Daniel Capaldo Amaral

Tese apresentada à Escola de Engenharia de
São Carlos, da Universidade de São Paulo,
como parte dos requisitos para obtenção do
título de Doutor em Engenharia Mecânica.


ORIENTADOR: Prof. Tit. Henrique Rozenfeld

São Carlos
2002

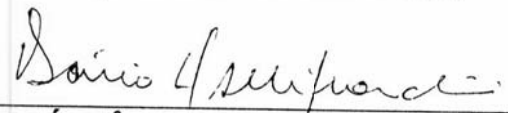
FOLHA DE JULGAMENTO

Candidato: Engenheiro **DANIEL CAPALDO AMARAL**

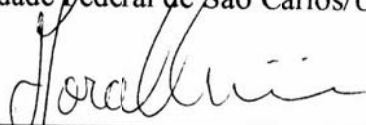
Tese defendida e julgada em 20-06-2002 perante a Comissão Julgadora:



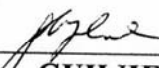
Prof. Tit. **HENRIQUE ROZENFELD (Orientador)**
(Escola de Engenharia de São Carlos/USP) emendada



Prof. Dr. **DÁRIO HENRIQUE ALLIPRANDINI**
(Universidade Federal de São Carlos/UFSCar) aprovada



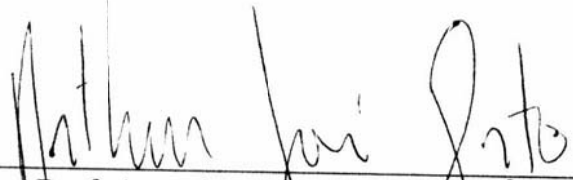
Prof. Dr. **FERNANDO ANTONIO FORCELLINI**
(Universidade Federal de Santa Catarina/UFSC) APROVADO



Prof. Assoc. **GUILHERME ARY PLONSKI**
(Faculdade de Economia, Administração e Contabilidade/USP) aprovado



Prof. Dr. **JOSÉ CLÁUDIO CYRINEU TERRA**
(Balance Consulting) aprovado



Prof. Associado **ARTHUR JOSÉ VIEIRA PORTO**
Coordenador do Programa de Pós-Graduação
em Engenharia Mecânica



Profa. Assoc. **MARIA DO CARMO CALIJURI**
Presidente da Comissão de Pós-Graduação

*À minha esposa Sayuri e
meus pais Sônia e Luis
Carlos (in memoram).*

AGRADECIMENTOS

Ao Professor Henrique Rozenfeld, pela oportunidade, apoio, incentivo, amizade e pela excepcional contribuição à minha formação como pesquisador.

A Sayuri, meu grande amor e companheira de todas as horas; sem dúvida, a maior incentivadora e “razão de ser” deste trabalho.

A toda minha família, em especial a meus pais (Sônia e Luis), irmãos (Carlos, André, Marina, Lê, Lu e Carlinhos) e avós (Elza e Américo), que estiveram sempre ao meu lado e souberam compreender as minhas ausências quando necessário.

À Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), pela bolsa e auxílios concedidos, que viabilizaram o desenvolvimento da pesquisa.

À empresa IDS-Scheer do Brasil pelo apoio na obtenção da cópia acadêmica do ARIS Toolset.

Aos amigos do Núcleo de Manufatura Avançada (NUMA), que direta ou indiretamente contribuíram muito para este trabalho. Em especial, ao Sérgio, Vander, Lucas e Xup e aos alunos de iniciação científica que me auxiliaram diretamente neste trabalho.

Aos funcionários do Núcleo de Manufatura Avançada (NUMA), pela disposição com que sempre me auxiliaram neste e em outros projetos de pesquisa: Fernandinho, André, Cristiane(s) e Francis.

Aos funcionários e docentes do Departamento de Engenharia de Produção da EESC, que me acolheram com grande atenção e carinho e me proporcionaram a tranqüilidade necessária para a conclusão desta pesquisa.

A todos os professores que encontrei durante minha vida acadêmica. Em especial, aos professores Toledo e Dário que, de maneira indireta e não menos importante, foram também responsáveis por este trabalho.

A Deus, pela oportunidade de realizar esta pesquisa que tanto me deu prazer, em um ambiente estimulante e desafiador como o que encontrei no Grupo de Engenharia Integrada (EI) do NUMA.

SUMÁRIO

<u>LISTA DE FIGURAS</u>	I
<u>LISTA DE TABELAS</u>	III
<u>LISTA DE QUADROS</u>	IV
<u>LISTA DE ABREVIATURAS E SIGLAS</u>	V
<u>RESUMO</u>	VI
<u>ABSTRACT</u>	VII
1. INTRODUÇÃO	1
<u>OBJETIVO</u>	3
<u>MÉTODO</u>	3
<u>ESTRUTURA DO TRABALHO</u>	4
<u>RESULTADOS</u>	4
2. REVISÃO BIBLIOGRÁFICA	5
<u>2.1 DESENVOLVIMENTO DE PRODUTO E INTEGRAÇÃO</u>	5
<u>2.1.1 O Processo de Desenvolvimento de Produto</u>	5
<u>2.2.2 Desafios para a Visão Integrada do Processo de Desenvolvimento de Produto</u>	8
<u>2.2. MODELAGEM DE EMPRESAS</u>	16
<u>2.2.1 Definição, Aplicações e Tipos de Modelos de empresa</u>	16
<u>2.2.2 Conceitos Fundamentais de Modelagem de Empresas</u>	19
<u>2.2.3 Métodos para a Modelagem de Empresas</u>	24
<u>2.2.4 ARIS (ARchitecture of Integrated Information Systems)</u>	28
<u>2.3 GESTÃO DO CONHECIMENTO</u>	34
<u>2.3.1 Conceitos Básicos de Gestão do Conhecimento</u>	34
<u>2.3.3 Tecnologia de informação aplicada à gestão do conhecimento</u>	41
<u>2.3.4 Formas de representação dos conhecimentos explícitos</u>	44
<u>2.3.5 Tecnologias e Ferramentas para Registro de Conhecimentos Explícitos</u>	47
3. MÉTODO	65
<u>3.1 PROBLEMA DE PESQUISA</u>	65
<u>3.2 JUSTIFICATIVA</u>	67
<u>3.3 HIPÓTESES</u>	68
<u>3.4 OBJETIVOS</u>	69
<u>3.5 MÉTODO</u>	71
<u>3.5.1 Referencial Teórico</u>	72
<u>3.5.2 Abordagem da Pesquisa</u>	73
<u>3.5.3 Método de Pesquisa</u>	74
<u>3.6 ETAPAS DO TRABALHO</u>	75
4. DESCRIÇÃO DA AROUTETURA PARA GERENCIAMENTO DE CONHECIMENTOS EXPLÍCITOS SOBRE O PROCESSO DE DESENVOLVIMENTO DE PRODUTO	79
<u>4.1 DEFINIÇÃO, PRINCÍPIO BÁSICO E REQUISITOS PARA A CONSTRUÇÃO DA AROUTETURA</u>	79
<u>4.2 VISÃO GERAL DA AROUTETURA</u>	82
<u>4.3 REPOSITÓRIO E OS TIPOS DE CONHECIMENTOS EXPLÍCITOS</u>	85
<u>4.3.1 Registros</u>	86
<u>4.3.2 Sentenças</u>	89

<u>4.3.3 Modelos de Processo:</u>	93
<u>4.4 FUNCIONALIDADES</u>	95
<u>4.5 ELEMENTOS DE APOIO</u>	98
<u>4.5.1 Modelo de validação e sistematização dos conhecimentos explícitos</u>	98
<u>4.5.2 Tipologia de processos de desenvolvimento de produto</u>	104
<u>4.5.3 Framework de modelagem</u>	106
<u>4.5.4 Modelo de Perfis de Usuário</u>	113
<u>4.6 UTILIZANDO A ARQUITETURA</u>	119
<u>5. DESENVOLVIMENTO DE UMA SOLUÇÃO BASEADA NA ARQUITETURA</u>	124
<u>5.1 DESCRIÇÃO GERAL DA SOLUÇÃO</u>	124
<u>5.2 ESPECIFICAÇÃO E DESCRIÇÃO DA INTRANET</u>	128
<u>5.2.1 Descrição do funcionamento da Intranet</u>	129
<u>5.2.2 Descrição do processo de desenvolvimento da intranet</u>	152
<u>5.2.3 Modelo da Base de Dados</u>	155
<u>5.2.4 Mapas de Telas</u>	159
<u>5.3 ADAPTAÇÃO DA FERRAMENTA DE MODELAGEM ARIS TOOLSET</u>	162
<u>5.3.1 Filtro de Modelagem</u>	164
<u>5.3.2 Estrutura de Diretórios Padrão na Ferramenta ARIS</u>	166
<u>5.3.3 Macro para a geração de Relatório</u>	169
<u>5.3.4 Modelos de Página WEB</u>	172
<u>5.3.5 Integração entre a ferramenta de modelagem e a intranet</u>	181
<u>6. EXEMPLO DE APLICAÇÃO DA SOLUÇÃO</u>	186
<u>6.1 CONTEXTO DA APLICAÇÃO</u>	186
<u>6.1.1 Objetivos e Critérios de Escolha</u>	186
<u>6.1.2 Descrição do Grupo de Engenharia Integrada</u>	188
<u>6.2 ETAPAS PARA A IMPLANTAÇÃO DO SISTEMA E REALIZAÇÃO DO EXEMPLO</u>	190
<u>6.2.1 Personalização (Customização) da Solução</u>	190
<u>6.2.2 “Alimentação” inicial do sistema</u>	192
<u>6.2.3 Treinamento e comunicação aos usuários</u>	193
<u>6.2.4 Avaliação do sistema</u>	193
<u>6.3 RESULTADOS DA APLICAÇÃO DA FERRAMENTA NO EXEMPLO</u>	194
<u>6.4 AVALIAÇÕES ADICIONAIS E EM ANDAMENTO DA SOLUÇÃO</u>	195
<u>Exemplo teórico da aplicação da solução em uma pequena empresa de alta tecnologia</u>	195
<u>Aplicação da solução numa comunidade de interesse em desenvolvimento de produto</u>	197
<u>7. CONCLUSÕES E PERSPECTIVAS FUTURAS</u>	199
<u>7.1 CONSIDERAÇÕES RELACIONADAS À PROPOSIÇÃO DA ARQUITETURA</u>	199
<u>7.2 CONSIDERAÇÕES RELACIONADAS À PROPOSIÇÃO DE UMA SOLUÇÃO</u>	202
<u>7.3 CONSIDERAÇÕES RELACIONADAS AO EXEMPLO DE APLICAÇÃO</u>	205
<u>7.4 CONCLUSÃO FINAL E PERSPECTIVAS FUTURAS</u>	206
<u>8. REFERÊNCIAS BIBLIOGRÁFICAS</u>	209

ANEXOS..... ERRO! INDICADOR NÃO DEFINIDO.

ANEXO 1 : **RELATÓRIO DO FILTRO ESPECIFICADO PARA A FERRAMENTA ARIS**ERRO!

INDICADOR NÃO DEFINIDO.

ANEXO 2 : **RELATÓRIO PADRÃO DO MODELO DE REFERÊNCIA DA METODOLOGIA**ERRO!

INDICADOR NÃO DEFINIDO.

LISTA DE FIGURAS

FIGURA 2. 1 VISÕES DA ARQUITETURA ARIS	28
FIGURA 2. 2 NÍVEIS DA ARQUITETURA ARIS	29
FIGURA 2. 3 FORMALISMOS NA ARQUITETURA ARIS	30
FIGURA 2. 4: EXEMPLO DE INTEGRAÇÃO DAS VISÕES DA ARQUITETURA ARIS	33
FIGURA 2. 5 : TRANSFORMAÇÃO ENTRE CONHECIMENTOS TÁCITOS E EXPLÍCITOS	37
FIGURA 3. 1 REFERENCIAL TEÓRICO E MÉTODOS	72
FIGURA 4. 1 ARQUITETURA PARA REGISTRO DE CONHECIMENTOS EXPLÍCITOS SOBRE O PROCESSO DE DESENVOLVIMENTO DE PRODUTO (ARCE-PDP)	83
FIGURA 4. 2 : REPOSITÓRIO DE CONHECIMENTOS EXPLÍCITOS DA ARCE-PDP	86
FIGURA 4. 3 : CLASSIFICAÇÃO DE REGISTROS SEGUNDO A ARQUITETURA	89
FIGURA 4. 4 : EXEMPLO DE CLASSIFICAÇÃO DE SENTENÇAS	91
FIGURA 4. 5: EXEMPLO DE CLASSIFICAÇÃO DE SENTENÇAS CONFORME O GRAU DE VALIDAÇÃO	92
FIGURA 4. 6: FUNCIONALIDADES BÁSICAS PREVISTAS NA ARQUITETURA	95
FIGURA 4. 7: EXEMPLO DO GRAU DE VALIDAÇÃO DE REGISTROS	99
FIGURA 4. 8 : EXEMPLO DE ÍNDICE DE VALIDAÇÃO DE SENTENÇAS	101
FIGURA 4. 9 : EXEMPLO ESQUEMÁTICO DE SENTENÇAS AVALIADAS	102
FIGURA 4. 10 ELEMENTOS DO FRAMEWORK DE MODELAGEM	108
FIGURA 4. 11 FORMALISMOS DE MODELAGEM UTILIZADOS EM CADA UMA DAS VISÕES	112
FIGURA 4. 12 : NÍVEIS DA VISÃO POR PROCESSO	113
FIGURA 4. 13 : REPRESENTAÇÃO ESQUEMÁTICA DO MODELO DE PERFIS DE USUÁRIO	115
FIGURA 4. 14 : REPRESENTAÇÃO DE DIFERENTES IMPLEMENTAÇÕES EMPREGANDO-SE A ARQUITETURA	121
FIGURA 5. 1 OBJETIVO DA SOLUÇÃO DESENVOLVIDA	125
FIGURA 5. 2: COMPONENTES PRINCIPAIS DA SOLUÇÃO DESENVOLVIDA	127
FIGURA 5. 3 : TELA PRINCIPAL DO SISTEMA	132
FIGURA 5. 4: TELA INICIAL DO MÓDULO PRÁTICAS SISTEMATIZADAS	134
FIGURA 5. 5: EXEMPLO DE UMA LISTA DE REGISTROS NO MÓDULO PRÁTICAS SISTEMATIZADAS	136
FIGURA 5. 6: PÁGINA DE COMENTÁRIOS DE REGISTROS	137
FIGURA 5. 7: EXEMPLO DE PÁGINA DE DETALHES DE REGISTRO	138
FIGURA 5. 8: EXEMPLO DE LISTA DE SENTENÇA	139
FIGURA 5. 9: EXEMPLO DE TELA DE DETALHES DA SENTENÇA	140
FIGURA 5. 10: EXEMPLO DE TELA PARA O CADASTRO DAS SENTENÇAS	141
FIGURA 5. 11 : PÁGINA INICIAL DO MÓDULO MODELOS DE REFERÊNCIA	143
FIGURA 5. 12: EXEMPLO DO RESULTADO DA CONSULTA DE UM LINK “DOC” PARA UM MODELO DE REFERÊNCIA ARMAZENADO EM HTML	144
FIGURA 5. 13: EXEMPLO DE TELA CONTENDO UMA FICHA DE MODELO DE REFERÊNCIA	145
FIGURA 5. 14: EXEMPLO DE CLASSIFICAÇÃO DE UM MODELO SEGUNDO A TIPOLOGIA DE PROCESSOS DE DESENVOLVIMENTO DE PRODUTO	146
FIGURA 5. 15: TELA INICIAL DO MÓDULO PÁGINAS DO CONHECIMENTO	147
FIGURA 5. 16: PÁGINA DO CONHECIMENTO	148
FIGURA 5. 17: TELA INICIAL DO MÓDULO PESSOAS & CONHECIMENTO	149
FIGURA 5. 18: EXEMPLO DE UMA FICHA COM INFORMAÇÕES DETALHADAS SOBRE UM USUÁRIO	150
FIGURA 5. 19: EXEMPLO DE TELA PARA ATUALIZAÇÃO DO DOMÍNIO DO USUÁRIO	151
FIGURA 5. 20: EXEMPLO DE LISTA DE GRUPOS DE DISCUSSÃO	152
FIGURA 5. 21 : PROCESSO DE DESENVOLVIMENTO DA INTRANET	154
FIGURA 5. 22: MODELO ENTIDADE-RELACIONAMENTO DA BASE DE DADOS	156
FIGURA 5. 23: MAPAS DE TELAS DO MÓDULO DE PRÁTICAS SISTEMATIZADAS	161
FIGURA 5. 24 : A ASSISTENTE PARA A CRIAÇÃO DO FILTRO DE MODELAGEM NA FERRAMENTA ARIS	166
FIGURA 5. 25: VISÕES DE UM MODELO ARMAZENADO NA FERRAMENTA DE MODELAGEM ARIS	167
FIGURA 5. 26: ESTRUTURA DE DIRETÓRIOS PADRÃO NA FERRAMENTA ARIS	169

<u>FIGURA 5. 27: TELA DE INÍCIO DA GERAÇÃO DO RELATÓRIO</u>	170
<u>FIGURA 5. 28: MACRO PARA A GERAÇÃO DE RELATÓRIOS DE MODELOS</u>	171
<u>FIGURA 5. 29 : MENU DA FERRAMENTA ARIS EMPREGADO NA GERAÇÃO DE MODELOS EM HTML</u>	173
<u>FIGURA 5. 30: EXEMPLO DE ARQUIVO-PADRÃO EM HTML EMPREGADO PELO ARIS NA GERAÇÃO DE PÁGINAS WEB DE MODELOS DE PROCESSO</u>	176
<u>FIGURA 5. 31: PÁGINA INICIAL</u>	179
<u>FIGURA 5. 32: EXEMPLO DA PÁGINA DO TIPO MODELO GERADA DEPOIS DA CUSTOMIZAÇÃO</u>	180
<u>FIGURA 5. 33: EXEMPLO DE PÁGINA DE UM OBJETO GERADA DEPOIS DA CUSTOMIZAÇÃO</u>	181
<u>FIGURA 5. 34: PÁGINA DE BUSCA - RESPONSÁVEL PELA INTEGRAÇÃO ENTRE AS PÁGINAS DE MODELO E OS CONHECIMENTOS EXPLÍCITOS ARMAZENADOS NA INTRANET</u>	182
<u>FIGURA 5. 35: INTEGRAÇÃO DOS MODELOS COM A INTRANET</u>	184

LISTA DE TABELAS

<u>TABELA 4. 1: CÁLCULO DO ÍNDICE DE VALIDAÇÃO PARA O EXEMPLO</u>	103
---	-----

LISTA DE QUADROS

<u>QUADRO 2. 1: FATORES QUE AFETAM O GERENCIAMENTO DO PDP E SUAS EVIDÊNCIAS NA LITERATURA</u>	14
<u>QUADRO 3. 1 SÍNTESE DAS PRINCIPAIS ETAPAS DO TRABALHO</u>	78
<u>QUADRO 4. 1 : CRITÉRIO PARA AVALIAÇÃO DO GRAU DE VALIDAÇÃO ENTRE REGISTRO E SENTENÇA</u>	100
<u>QUADRO 4. 2 : EXEMPLO DE AVALIAÇÃO DE SENTENÇAS</u>	104
<u>QUADRO 4. 3 DIMENSÕES DA TIPOLOGIA DE PROCESSOS DE DESENVOLVIMENTO DE PRODUTO</u>	106
<u>QUADRO 4. 4: FORMALISMOS E CONSTRUTOS EMPREGADOS EM CADA VISÃO</u>	110
<u>QUADRO 4. 5 : SITUAÇÕES PREVISTAS NA METODOLOGIA DE MODELAGEM</u>	111
<u>QUADRO 4. 6 SÍNTESE DOS DIFERENTES TIPOS DE PERFIS</u>	118
<u>QUADRO 5. 1: ESTRUTURA DE DIRETÓRIOS PADRÃO NA FERRAMENTA DE MODELAGEM</u>	168
<u>QUADRO 6. 1 : CLASSIFICAÇÃO DE CLASSE E SUBCLASSE DOS CONHECIMENTOS EXPLÍCITOS DO GRUPO EI</u>	191
<u>QUADRO 6. 2: ESTRUTURA DE CLASSES E SUBCLASSES PARA UMA PEQUENA EMPRESA</u>	196
<u>QUADRO 6. 3: CONTINUAÇÃO DO QUADRO 6.2</u>	197

LISTA DE ABREVIATURAS E SIGLAS

ARCE-PDP	- Arquitetura para Registro de Conhecimentos Explícitos sobre PDP
ARIS	- ARchitecture for Integrated Systems
ASP	- Active Server Pages
cPM	- Collaborative Product Management
EDM	- Eletronic Data Management
ERP	- Enterprise Resource Planning
GC	- Gestão do Conhecimento
GED	- Gerenciamento Eletrônico de Documentos
HTML	- Hypertext Markup Language
KM	- Knowledge Management
PDM	- Product Data Management
PDP	- Processo de Desenvolvimento de Produto
PM	- Project Management
TI	- Tecnologia de Informática
www	- World Wide Web
XML	- Extended Markup Language
TCP	- Transfer Control Protocol
IP	- Internet Protocol

RESUMO

AMARAL, D.C. (2001). *Arquitetura para gerenciamento de conhecimentos explícitos sobre o processo desenvolvimento de produto*. São Carlos, 2001. 190 p. Tese (Doutorado) – Escola de Engenharia de São Carlos, Universidade de São Paulo.

O processo de desenvolvimento de produto é uma atividade caracterizada por um ciclo de projetar-construir-testar que faz uso intensivo do conhecimento existente na empresa. Uma parte dele está armazenada nos documentos gerados no decorrer deste processo, são os conhecimentos explícitos. Este trabalho analisa as soluções computacionais dedicadas a auxiliar na gestão deste tipo de conhecimento e demonstra vários problemas: falta de um modelo conceitual apropriado; limitações referentes aos tipos de conhecimentos armazenados; e falta de uma sistemática de validação e sistematização. Propõe-se um conjunto de elementos, arquitetura, para superá-los, o qual pode ser utilizado como referência para o desenvolvimento de sistemas de gestão de conhecimentos explícitos. A arquitetura foi empregada no desenvolvimento de uma solução computacional específica, utilizando uma ferramenta de modelagem e uma intranet, e esta foi posteriormente aplicada para a gestão dos conhecimentos dentro de um grupo de pesquisa. Os resultados demonstram que a arquitetura é viável e que o sistema pode e deve ser aplicado em empresas que desenvolvam produtos. Encerra-se o trabalho apresentando as propostas de melhoria e de novas avaliações para ambas, a arquitetura e a solução desenvolvidas.

Palavras-Chave: processo de desenvolvimento de produto, modelagem de empresa, gestão do conhecimento, aprendizagem organizacional e ferramentas.

ABSTRACT

AMARAL, D.C. (2001). *Architecture for explicit knowledge management on product development processes*. São Carlos, 2001. 190 p. Tese (Doutorado) – Escola de Engenharia de São Carlos, Universidade de São Paulo.

The product development process is characterized by a continuing design-build-test cycle, marked by experimentation and the intensive use of enterprise knowledge. Part of them is tangible and is stored in the documents created during the product development process, named explicit knowledge. This work analyzes the computational solutions for support the explicit knowledge management. As result present a set of common limitations of commercial tools applied in this area. Three important aspects are: a lack of conceptual model to link the information systems at knowledge management effort; small knowledge representation types manipulated by these systems; and a lack of knowledge validation model, sufficient to “filter” the explicit knowledge created. For encompass these challenges, this work presents an architecture (a set of elements) designed to build explicit knowledge management systems. A solution aided by the architecture was created, which integrates a commercial modeling tool and an intranet especially designed. Finally, this solution was applied at specific case: to manage the explicit knowledge of a research group. The results show that the architecture is viable and sufficient to create explicit knowledge management systems. Finalizing, the work presents a set of future findings to architecture’s development research.

Keywords: product development processes, knowledge management, learning organization, enterprise modeling.

1. Introdução

O processo de desenvolvimento pode ser beneficiado enormemente pela gestão do conhecimento. As atividades relacionadas com este processo têm um caráter essencialmente criativo e, portanto, dependem das habilidades e conhecimentos das pessoas envolvidas. Além disso, a produtividade é obtida pelo uso intensivo de técnicas e métodos em constante evolução, os quais têm a experiência como fator decisivo para o sucesso durante aplicações. Neste contexto, uma eficiente gestão do conhecimento deve permitir que as experiências na aplicação das ferramentas e as soluções adotadas em cada projeto possam ser difundidas através da organização, obtendo a melhoria contínua neste processo de negócio.

Parte do conhecimento sobre desenvolvimento de produto se apresenta na forma de documentos (relatórios, desenhos, resultados de testes, atas, livros, etc...), e informações (endereços de sites, contatos, nomes de empresas fornecedoras, etc...) que podem ser armazenados e disponibilizados para as diversas pessoas ligadas a esta atividade, são os denominados conhecimentos explícitos. Um caminho para se aprimorar a Gestão do Conhecimento (GC) é criar ferramentas que auxiliem no gerenciamento de tais tipos de conhecimento. São ferramentas computacionais que possibilitam o acesso aos conhecimentos explícitos facilitando o seu armazenamento e recuperação.

Durante os últimos anos, com a ênfase no rótulo Gestão de Conhecimento e o crescimento desta área de pesquisa, tais ferramentas receberam grande atenção por parte de profissionais e pesquisadores. Muitas vezes, equivocadamente referenciadas como a solução para o gerenciamento do conhecimento. Evidentemente, tais sistemas contribuem, mas não são suficientes. A efetiva Gestão do Conhecimento depende de muitos outros aspectos tais como: cultura, motivação e capacitação de pessoal, entre outros.

No caso específico do processo de desenvolvimento de produto, porém, o papel destas ferramentas é particularmente importante. Cada projeto de desenvolvimento significa uma ocorrência única do processo de negócio, contendo

características e problemas particulares. O enfrentamento destes problemas é uma rica fonte de conhecimento para os membros das equipes de projeto, que, na maioria das empresas, não têm como transmitir sistematicamente este aprendizado para outros membros da organização. Parte significativa deste conhecimento está armazenado no formato explícito, inserida no conjunto geralmente volumoso de documentos criados durante o projeto, tais como relatórios de testes, desenhos e pareceres de discussões, entre tantos outros. Devido à quantidade de documentos e à natureza diversificada destes, é especialmente difícil recuperar e fazer uso dos conhecimentos ali armazenados. Assim, as ferramentas de gestão do conhecimento podem auxiliar o processo de desenvolvimento de produto permitindo um maior aproveitamento e eficiência na transmissão do conhecimento armazenado nesta parcela explícita. E, ainda, esta melhoria pode servir como um “catalisador” na medida que inclua referências às pessoas que foram protagonistas das soluções destes problemas, as quais “carregam em si” todo o conhecimento tácito que não pôde ser registrado.

Existem várias tecnologias e ferramentas de informática que podem auxiliar na gestão dos conhecimentos explícitos presentes no processo de desenvolvimento de produto. Uma análise destas ferramentas, apresentada neste trabalho, mostra que ainda existem vários problemas que precisam ser superados para que elas contribuam de maneira mais eficiente para a gestão do conhecimento. Três aspectos são mais relevantes. Em primeiro lugar elas não têm sido desenvolvidas dentro do contexto de um modelo conceitual mais amplo sobre gestão do conhecimento, prejudicando o processo de especificação das funcionalidades, avaliação, escolha e, principalmente implantação destas ferramentas; mais além, impedem que elas sejam inseridas harmoniosamente com as demais iniciativas de GC em curso dentro da organização, incluindo os fatores comportamentais e humanos, fundamentais dentro deste tipo de esforço. Um segundo aspecto é que elas carecem de sistemáticas mais elaboradas para validação e sistematização do conhecimento armazenado e, por fim, tais soluções precisam ser desenvolvidas de maneira aberta para que possam ser implementadas em partes, aproveitando a base de conhecimento e os investimentos depositados nos diversos sistemas já existentes na organização.

Objetivo

O trabalho aborda as soluções para gerenciamento de conhecimentos explícitos, dentro de uma área específica, que é o processo de desenvolvimento de produto (PDP). Foram empregados conceitos das áreas de estudo sobre o processo de desenvolvimento de produto, modelagem de empresa e gestão do conhecimento, com o intuito de avaliar as soluções existentes e propor um modelo teórico que especifica como desenvolver sistemas para gestão do conhecimento para o PDP, denominado Arquitetura para Gerenciamento de Conhecimentos Explícitos no Processo de Desenvolvimento de Produto (ARCE-PDP). Espera-se que, a partir deste modelo, diferentes profissionais possam especificar e implantar sistemas para gerenciamento de conhecimentos explícitos capazes de contribuir para a aprendizagem organizacional no desenvolvimento de produto, e, em última instância, melhorar o desempenho da empresa neste processo de negócio.

Neste trabalho, propõe-se a Arquitetura e, para verificar sua viabilidade, foi desenvolvida uma solução baseada nas suas especificações, formada a partir do desenvolvimento de uma intranet e sua integração com uma ferramenta de modelagem. Por fim, conduziu-se uma aplicação desta solução em um caso exemplo.

Método

A metodologia de pesquisa empregada pode ser classificada como pesquisa de campo do tipo participante-observador, também muitas vezes conhecido como pesquisa-ação. Dentro deste método, o pesquisador foi o responsável pelo desenvolvimento tanto da arquitetura quanto da solução, e, também, conduziu a aplicação da solução no caso específico. A adoção deste método foi motivada, principalmente, pela novidade da proposta. A avaliação da arquitetura exigia uma análise profunda e detalhada, incorporando desde os aspectos conceituais de gestão do conhecimento até os detalhes mais técnicos relacionados com a tecnologia de informação. O emprego deste método foi fundamental para se verificar a viabilidade e a importância de um modelo como o expresso na arquitetura, e também as vantagens e a viabilidade de utilizá-lo como referência para a criação de sistemas para gestão do conhecimento no PDP.

Estrutura do Trabalho

O trabalho é dividido em sete capítulos. O capítulo 2 apresenta uma breve revisão bibliográfica com os conceitos principais sobre gestão do conhecimento, modelagem de empresas e desenvolvimento de produto. Em seguida, no capítulo 3, descreve-se, em detalhes, a metodologia empregada. O capítulo 4 apresenta a arquitetura, contendo uma visão geral e detalhes importantes de cada um de seus elementos. O capítulo 5 apresenta a solução desenvolvida com base na arquitetura apresentada no capítulo anterior. O capítulo 6 descreve um caso de aplicação da solução e, no capítulo final, de número 7, são apresentadas as conclusões e considerações finais desenvolvidas a partir de todo o trabalho.

Resultados

Como primeiro resultado concreto deste trabalho, tem-se uma primeira comprovação da importância e a viabilidade de se empregar um modelo teórico para auxiliar o desenvolvimento de sistemas para a gestão do conhecimento no Processo de Desenvolvimento de Produto. As observações obtidas no desenvolvimento e na aplicação tanto da arquitetura como da solução, demonstraram que a arquitetura é, além de viável, útil para o desenvolvimento de sistemas de gerenciamento de conhecimentos explícitos. Além disso, são demonstrados vários pontos críticos, vantagens e desvantagens relacionadas com a implementação computacional da arquitetura e da solução específica. Por fim, o caso de aplicação da solução analisada serviu como mais uma confirmação da importância da arquitetura e permitiu a identificação de vários pontos para a melhoria da solução específica. Deve-se ressaltar, ainda, a geração de uma sólida base para a continuidade do desenvolvimento da arquitetura proposta, outra contribuição importante deste projeto de pesquisa. Este fato pode ser constatado na lista de temas para pesquisas futuras, apresentada ao final do trabalho.

2. Revisão Bibliográfica

Este capítulo apresenta os conceitos fundamentais sobre as 3 grandes áreas de conhecimento relacionadas com este trabalho: desenvolvimento de produto (2.1), modelagem de empresas (2.2) e gestão do conhecimento (2.3).

2.1 Desenvolvimento de Produto e Integração

A revisão de desenvolvimento de produto apresentada tem como objetivo principal demonstrar a definição de Processo de Desenvolvimento de Produto adotada, item 2.1.1, que é baseada numa abordagem interdisciplinar e integrada. Adicionalmente, procura-se demonstrar a importância e os desafios para se estudar o desenvolvimento de produto segundo esta abordagem, e de que maneira os trabalhos que unam conceitos de modelagem de empresa e gestão do conhecimento no contexto do desenvolvimento de produto podem auxiliar na superação destes desafios.

2.1.1 O Processo de Desenvolvimento de Produto

Na área de pesquisa sobre o processo de desenvolvimento de produto já é tradicional o emprego da abordagem por processos, na qual o desenvolvimento de produto é visto como algo além de um conjunto específico de atividades de engenharia, tais como cálculos, desenhos e prototipagem. Na visão de processos, o desenvolvimento de produtos engloba o conjunto de atividades realizadas pelos diversos setores funcionais da empresa, que permitem a transformação de informações sobre necessidades de mercado em informações e recursos para a produção de um produto específico.

Seguindo esta linha, adota-se neste trabalho uma das definições mais clássicas para desenvolvimento de produto, que é também uma das primeiras a adotar esta abordagem, definindo-o como: "...o processo pelo qual uma organização transforma dados sobre oportunidades de mercado e possibilidades técnicas em bens e informações para a fabricação de um produto comercial" (CLARK & FUJIMOTO, 1991).

Essa é uma definição clássica continua válida, bastando que fiquem claros os contornos, início e fim, deste processo; ou melhor, o que se entende de cada uma dessas informações: oportunidades de mercado, possibilidades técnicas e informações para a fabricação. Conforme apontado por ROZENFELD et al (2000), cada vez mais tem sido ampliado o escopo do processo de desenvolvimento de produto. O seu início, por exemplo, antes classicamente entendido como a definição das especificações do produto, hoje é tido como o planejamento estratégico e gerenciamento de *portfólio* de produtos. Segundo estes autores, várias empresas assumem que o desenvolvimento de produto é um processo essencial (“core”). As decisões relacionadas com a definição das especificações devem ser coerentes com o planejamento da linha completa de produtos e com os objetivos e estratégias pré-definidos pela organização. Portanto, estas empresas definem seu processo de desenvolvimento de produto como tendo por início o planejamento estratégico da empresa. Na outra ponta, ampliando também o escopo, atividades como a retirada do produto do mercado e o descarte dos produtos fora de uso estão cada vez mais sendo consideradas como parte do desenvolvimento de produto. Em algumas indústrias de ponta como a de eletro-eletrônicos (em especial celulares, multifuncionais, etc...), a diminuição drástica do ciclo de vida do produto tem feito com que a própria manufatura passe a fazer parte deste processo, sendo o produto em si visto como um grande projeto, que passa pelo processo de planejamento, venda, fabricação e retirada do mercado.

Neste trabalho, adota-se esse escopo maior. Assume-se que o processo de desenvolvimento de produto inicia com o desenvolvimento do plano estratégico e do *portfólio* de projetos; passa pela definição do produto, conceito e detalhamento, até a preparação dos recursos para a fabricação e termina com algumas das atividades de fabricação (controle de mudanças de engenharia), completa retirada do produto do mercado e descarte.

Em termos práticos, a visão por processos significa analisar o desenvolvimento de produto como o conjunto de atividades ordenadas no tempo e com entradas e saídas claramente definidas. As vantagens da aplicação dessa visão são:

- tornar claras as relações dentro da organização e entre a organização e o mercado (CLARK, 1991);
- permitir a identificação de características que influenciam o desempenho deste processo que muitas vezes não estão relacionados às clássicas atividades de engenharia (CLARK, 1991);
- ser uma visão comum a diferentes abordagens tais como reengenharia e sistemas de garantia de qualidade, entre outras; facilitando, portanto, a visão interdisciplinar (ROZENFELD, 2000);
- permitir a discussão de problemas específicos, sem perder de vista o contexto e as diferentes visões do processo. Isto permitiria melhores decisões para os especialistas, permitindo considerar fatores além daqueles relacionados com a área de conhecimento da sua especialização (ROZENFELD, 2000);
- fazer com que todas as pessoas possam visualizar o que elas fazem juntas e o que realmente agrega valor, ao invés do enfoque nas responsabilidades, hierarquias e funções individuais (DECHAMPS & NAYA, 1997) e
- realçar o aprendizado, pois salienta claramente as penalidades decorrentes da falta de coordenação funcional (DECHAMPS & NAYA, 1997).

Embora pareça simples e bastante razoável a aplicação da visão por processos na análise do desenvolvimento de produto, deve-se alertar que ela tem sido um grande desafio para pesquisadores e profissionais de empresa. Em primeiro lugar porque o processo de desenvolvimento de produto envolve, em grande medida, um esforço criativo a ser perpetrado num prazo longo de tempo, muitas vezes anos. Cada projeto de desenvolvimento se torna, então, uma experiência única porque envolve a solução de um conjunto de problemas específicos. Isso faz dele um processo que poderia ser classificado como “pouco estruturado”, isto é, onde conteúdo, tempo de duração e seqüência das atividades são diferentes, conforme as especificidades do projeto, tornando o processo difícil de ser descrito e visualizado de maneira integrada. Ao contrário de um processo como cobrança ou aprovação de seguro, por

exemplo, no qual um procedimento pode ser burocraticamente repetido diversas vezes, com alterações insignificantes entre elas, num prazo curto de tempo e permitindo que seja facilmente monitorado, chamado também de “estruturado”.

Outra característica diferencial e importante do PDP é o emprego de pessoas com formação e visões muito distintas, que fazem uso de diferentes áreas do conhecimento e, portanto, geralmente mais afeitas a dar importância a um conjunto específico de fatores: aqueles mais próximos da sua área de formação. São engenheiros, físicos, *designers*, economistas, administradores e até mesmo pessoas com formação em recursos humanos que avaliam, por exemplo, competências para os times. Assim, um grande desafio é obter e transmitir uma visão integrada e comum que considere os aspectos destas diferentes disciplinas.

A busca da visão integrada é fundamental para aprimorar a colaboração e o compartilhamento dos conhecimentos entre estes diferentes especialistas. Assim, no próximo item, apresentamos, em maiores detalhes, a importância e a evolução dos conceitos de integração na literatura de desenvolvimento de produto.

2.2.2 Desafios para a Visão Integrada do Processo de Desenvolvimento de Produto

A literatura abordando o processo de desenvolvimento de produto sempre manteve grande extensão e pulverização. Antes da década de noventa, porém, tais características eram ainda mais fortes. Os trabalhos, publicações e periódicos sobre este assunto eram restritos a domínios específicos do conhecimento e, com raras exceções, totalmente ligados a um único deles (engenharia, administração, marketing, direito e desenho industrial, entre outros).

Aos poucos, e, principalmente, depois dos anos 90, profissionais e pesquisadores de alguns campos específicos romperam os domínios do conhecimento da sua área de origem e formaram a base dos trabalhos e abordagens que, atualmente, buscam entender e intervir nesse processo como um todo. Na área de engenharia, PUGH(1978, 1996) foi um dos primeiros pesquisadores a estudar o processo de desenvolvimento como algo além da questão técnica ligada a uma disciplina, desenvolvendo para contrapor-la, a abordagem do *Total Design* (depois bastante desenvolvida por CLAUSING, 1994). Na área de administração, esta literatura sobre o processo de desenvolvimento de produto tem raízes nos estudos sobre inovação

tecnológica, a partir da qual, segundo BROWN & EISENDHART(1995), podem ser classificadas em três linhas de pesquisa definidas como Resolução de Problemas (IMAI, NONAKA & TAKEUCHI, 1985; CLARK, 1991); Rational Plan (COOPER, 1996); e Communication Web (Katz & Allen (1985).

Existem vários indícios da busca de uma visão integrada nesta nova e emergente literatura. Vem se tornando comum a soma de esforços de pesquisadores oriundos das diferentes tradições anteriormente citadas (administração, engenharia, etc.), o maior número de periódicos e livros interdisciplinares que divulgam trabalhos com diferentes enfoques e, talvez mais importante, o surgimento de um corpo de conceitos e técnicas típicos e comuns para estas diferentes áreas (tais como os conceitos de processo de desenvolvimento de produto, fases, definição de produtos, etc.); já delineando, portanto, um corpo de conhecimentos próprios.

Analisando esses trabalhos, parece evidente que houve um enfoque mais preocupado com o que é comum à todo o processo de desenvolvimento de produto, do que os aspectos que os torna diferente. Note que não se está afirmando que estes autores descuidaram ou negaram as contingências, simplesmente eram elas deixadas a um segundo plano, enquanto procuravam mostrar aspectos válidos em todos os casos, ou o que hoje comumente denomina-se *best practices*. Por exemplo, CLARK (1991) já apresentou indícios sobre importantes diferenças na gestão, como no caso do time de projeto e grau de integração, dependendo do grau de inovação do projeto, assim como PUGH o fez ao propor as categorias “dinâmico/estático”, porém, ambos, foram reconhecidos e deram especial atenção nos resultados aos aspectos que permitiam generalizações.

Um outro aspecto também nesta pesquisa é o fato da grande maioria dos trabalhos experimentais e de campo lidarem com casos de desenvolvimento totalmente novos. Isto, em parte, origina-se da crença, muito difundida nos trabalhos pioneiros, de que soluções identificadas para o caso mais complexo seriam mais facilmente transpostas para os casos mais simples. É assim que CLARK(1991) e muitos, depois dele, justificaram estudos enfocando os processos de grande inovação.

Pesquisas atuais acerca deste tema têm mostrado, porém, que as contingências podem levar a modificações muito grandes na forma de se desenvolver produtos, reivindicando um maior cuidado sobre o assunto tal como GRIFFIN

(1997), VERIZER(1998a e b), SONG & MONTOYA (1998), MULLINS & SUTHERLAND(1998), SWINK et al(1998), FUNK(1997), SOUNDER & SONG(1998). Principalmente em SWINK et al (1996), GRIFFIN (1998) e FUNK (1998) em que os autores são enfáticos na necessidade de mais pesquisas empíricas sobre as diferenças necessárias na intervenção do processo de desenvolvimento de produto. E de tal ordem são estas diferenças que, muitas assertivas retiradas de um tipo específico de processo de desenvolvimento de produto podem ter efeitos completamente diferentes, conforme se alteram esses aspectos ou fatores fundamentais. Um exemplo é a multidisciplinaridade e alta integração dos times, característica que, na primeira metade dos anos 90, era unanimemente vista como positiva na área. Pesquisas recentes têm demonstrado que seus efeitos podem ou não ser positivos conforme o tipo de projeto e, de tal intensidade que diferentes integrações funcionais seriam mais ou menos eficazes conforme o contexto (SWINK et al, 1996; e KAHN & MACDONOUGH, E.F. 1997).

Estas contingências, porém, na forma como são apresentadas atualmente tendem a “romper” a visão geral do processo de desenvolvimento de produto. Isto é, as pesquisas parecem segmentar-se, sendo dirigidas a tipos de processos de desenvolvimento bem particulares, como os trabalhos que vêm surgindo na universidade da Flórida sobre produtos altamente inovadores (SONG & MONTOYA, 1998; VERIZER, 1998 a e b; MULLINS et al, 1998) e àqueles sobre produtos plataforma (ROBERTSON & ULRICH, 1998; LUNDQVIST et al, 1996, MEYER et al, 1997).

Para que se caminhe no sentido de uma visão integrada, portanto, é preciso romper dois grandes desafios: desenvolver maneiras de representar os processos e armazenar os conhecimentos sobre este processo de maneira integrada. O primeiro desses dois desafios é o tema do item 2.2 desta revisão bibliográfica, isto é, trata da aplicação das ferramentas e conceitos sobre modelagem de empresas. Já o segundo deles pode ser auxiliado pelo desenvolvimento das abordagens e soluções computacionais para a gestão do conhecimento, item 2.3.

Deve-se ainda notar que um aspecto “pontual”, porém, muito importante para se obter a visão integrada do processo de desenvolvimento de produto é a identificação dos fatores (características) relacionados com o processo de negócio

(produtos, organizações envolvidas, atividades, contexto, etc...) que influenciam na forma como deve ser conduzido o desenvolvimento de produto. Eles são fundamentais para a contextualização e aplicação adequada das ferramentas, tecnologias e conhecimentos, de origens em diferentes áreas, dentro de um único processo de desenvolvimento. Pensando nisto acrescentou-se à esta revisão uma compilação inicial desses principais fatores, a qual auxiliou o desenvolvimento de um dos elementos da arquitetura apresentado mais adiante. Este levantamento foi baseado em uma pesquisa bibliográfica e entrevistas com especialistas (ROZENFELD & AMARAL, 1999).

- **Grau de inovação do produto/Produtos Plataforma:** este é o fator com maior evidência na literatura sobre influências no gerenciamento do processo de desenvolvimento de produto e também o mais amplamente citado. Desde os primeiros autores que analisaram o processo de desenvolvimento de produto de forma mais ampla e integrada, já era reconhecida a importância de se considerar o grau de inovação na determinação da forma de gerenciamento deste processo. É o caso de PUGH (1978;1996), CLARK & FUJIMOTO(1991) e WHEELWRIGHT & CLARK (1993). Quanto maior a inovação no produto deve-se aumentar a sobreposição entre o desenvolvimento da tecnologia e as etapas de detalhamento do projeto (FUNK, 1997), aplicar medidas de análise de viabilidade econômica que considerem adequadamente os maiores riscos envolvidos nestes projetos (SWINK et al, 1996, VERYZER, 1998a; VERYZER, 1998b), direcionar os esforços de desenvolvimento para qualidade, enquanto em projetos incrementais e de baixa inovação deve-se focar o tempo e custo (SWINK et al, 1996); seguir um processo menos estruturado (VERYZER 1996 a); aumentar o grau de compartilhamento de informações (SWINK et al, 1996); diminuir a influência da manufatura nas decisões do início do processo de desenvolvimento (SWINK, 1996); aumentar o esforço de coordenação (SONG &

MONTOYA, 1998; ROBERTSON & ULRICH, 1998; LUNDQVIST et al 1996; e MEYER et al, 1997).

- **Estratégia inter-projetos:** outro fator importante é o tempo que decorre entre o projeto que está sendo desenvolvido e o fim do projeto que está servindo de base para o desenvolvimento deste, isto é, do tempo de vida da plataforma que está sendo reaproveitada. Esse aproveitamento pode ter origem em um produto plataforma muito antigo, que acabou de ser desenvolvido ou, então, que está sendo desenvolvido paralelamente ao projeto de desenvolvimento analisado. A este elemento denomina-se estratégia inter-projetos. CUSUMANO & NOBEOKA (1998) demonstraram que esta estratégia é fundamental no desempenho do processo de desenvolvimento de produto. Em uma estratégia “simultânea”, aquela em que diferentes produtos desenvolvidos irão comungar a mesma solução ou subsistema, há a necessidade de intensa interação entre as equipes de ambos os projetos e de uma fase de proposição de especificações e conceitos mais cuidadosa. O conceito deve ser suficientemente robusto para suportar as diretrizes de ambos os projetos. Por outro lado, no caso onde adota-se a estratégia “posterior”, onde há um aproveitamento efetivo de um projeto finalizado e no qual a equipe ainda não se dissolveu, a fase conceitual é, ao contrário, mais simplificada e rápida pois existe um conceito e um time que está armazenando estas informações.
- **Estrutura do Mercado:** Com relação ao mercado, MULLINS & SUTHERLAND (1998) mostram evidências de que é preciso envolver mais os usuários no desenvolvimento da idéia quanto maior o grau de mudanças e o nível de concorrência. Iniciar protótipos e testes no início do desenvolvimento, dar ênfase à dimensão velocidade (dando capacidade de introduzir produtos rapidamente); limitar grandes pesquisas de mercado; envolver mais a alta direção no projeto; desenvolver uma cultura de desenvolvimento de produto dentro da empresa (todos devem se sentir responsáveis por este processo); e

aplicar estratégias de marketing direto são aspectos que precisam, portanto, ser considerados no modelo do processo de desenvolvimento de produto dentro destes ambientes.

- **Direcionamento Estratégico:** projetos direcionados à velocidade devem possuir grande interação entre pessoas de nível técnico, internas e externas à empresa, times menores e menos formalizados e grande sobreposição de fases (SWINK et al, 1996); projetos com alta prioridade em qualidade devem possuir grande interação entre clientes, projetistas e profissionais de marketing e menor sobreposição de fases (SWINK et al 1996) e alta prioridade em custos devem focalizar a interação entre fornecedores, manufatura e projetistas, muita prototipagem e uma dedicação maior na fase de projeto/processo (SWINK et al, 1996).
- **Dimensões da Qualidade do Produto:** GARVIN (1992) mostrou que empresas com melhor desempenho dirigiam os esforços dentro do seu processo de desenvolvimento no sentido de balancear as características de qualidade mais importantes para o consumidor. Em um estudo sobre a indústria de ar-condicionado, GARVIN(1992) identificou que as empresas de melhor desempenho nessa indústria priorizavam, durante o processo de desenvolvimento de produto, o parâmetro confiabilidade(com ensaios, testes, protótipos, ferramentas e atividades específicas para este campo), o mesmo que nas medições sobre os requisitos dos consumidores nesta indústria estava entre os de maior importância.
- **Tecnologia:** o próprio senso comum mostra que existem diferenças referentes à tecnologia principal do produto. Assim, em um projeto de um produto cuja tecnologia “core” é mecânica, este deverá ser gerenciado diferentemente do que um produto eletrônico ou químico, e assim por diante. Enfatizam esta questão PUGH(1978;1996) e SWINK et al (1996). E, dentro de uma mesma tecnologia podem existir diferentes graus de complexidade. Por exemplo, um projeto de desenvolvimento de produto qualquer, contendo dezenas de peças,

deve ser gerenciado diferentemente daquele com milhares ou milhões (CLARK & FUJIMOTO, 1991; SWINK et al 1996).

- **Interface com o Usuário:** um produto que tenha uma interface com o consumidor em que há maior quantidade de elementos que envolvam, eminentemente, características emocionais e cognitivas tem uma interface mais complexa (por exemplo, um carro) do que um produto cujo consumidor possua um conhecimento técnico profundo e os parâmetros sejam quantitativos, objetivos e bem conhecidos (por exemplo, um redutor). Quanto maior a complexidade da interface, maior a necessidade de se segmentar os mercados e analisar os *trade-offs* entre características, maior o cuidado na identificação dos requisitos, formulação do conceito e antecipação de testes (FUNK, 1997); maior a interação de pessoal de marketing e consumidores no desenvolvimento (CLARK & FUJIMOTO, 1991; FUNK, 1997; SWINK et al, 1996); e um sistema de informações compartilhado (SWINK et al, 1996).
- **Cultura Organizacional:** SOUNDER & SONG (1998) enfatizam também a cultura da organização, que afetaria significativamente a maneira de gerenciar o processo de desenvolvimento de produto, sem muitos detalhes sobre como isto afeta a forma de gerenciar o desenvolvimento de produto.

O Quadro 2. 1 resume os fatores, correlacionando-os aos trabalhos que os evidenciam mais fortemente.

Quadro 2. 1: Fatores que afetam o gerenciamento do PDP e suas evidências na literatura

Fator	Evidências na Literatura
Grau de inovação no produto / Produto Plataformas	PUGH (1978; 1996); CLARK & FUJIMOTO (1991); WHEELWRIGHT & CLARK (1993); FUNK(1997); VERYZER(1999 a e b); (SWINK et al, 1996); MEYER et al (1997); LUNDQVIST et al (1996); SONG & MONTOYA (1998)
Parâmetros de qualidade do produto	GARVIN(1988);
Interface do Produto com o usuário	CLARK & FUJIMOTO (1991); FUNK(1997)
Tecnologia do Produto	CLARK & FUJIMOTO (1991)
Complexidade da Estrutura interna	CLARK & FUJIMOTO (1991); FUNK(1997); SWINK et al (1996)

Cultura	SOUNDER & SONG (1998)
Direcionamento estratégico	SWINK et al (1996)
Estratégia Inter-Projetos	CUSUMANO
Mercado	MULLINS & SUTHERLAND (1998)
Estratégia de Negócio da Empresa	SOUNDER & SONG(1998);
Tecnologia	PUGH(1996);CLARK & FUJIMOTO (1991)

2.2. Modelagem de Empresas

Neste item, procurou-se definir e apresentar os conceitos principais sobre modelagem de empresas e fornecer uma visão geral das ferramentas disponíveis. Especial atenção foi conferida ao *framework* ARIS, que é utilizado como base para o desenvolvimento do trabalho.

2.2.1 Definição, Aplicações e Tipos de Modelos de empresa

Facilitar a troca de informações e o relacionamento harmonioso entre os diversos elementos de uma organização é uma questão fundamental para o gerenciamento, desde os primórdios da divisão do trabalho. A Integração de Empresa pode auxiliar neste objetivo, sendo definida como “harmonizar os fluxos de informação, controle e material dentro da organização, para aprimorar a comunicação, cooperação e coordenação dentro da empresa tal que seja atingido uma maior produtividade, flexibilidade, capacidade de reação e um melhor gerenciamento da mudança” (VERNADAT, 1996b, p.).

Para permitir a integração de empresas é preciso que os sistemas que a compõem, sejam eles sistemas computacionais (aplicações) ou mesmo sistemas mais complexos formados por homens, máquinas e também *softwares*, sejam capazes de trocar informações entre si. Para esse fim, são necessárias mais do que conexões físicas. Cada um dos atores, neste processo de comunicação, precisa ser capaz de entender as informações transmitidas. É neste ponto que entra o modelo de empresa, cuja construção pode proporcionar um referencial comum a toda organização sobre as definições de seus elementos e de como eles trabalham conjuntamente para gerar as atividades fim da organização.

2.2.1.1 Definição de modelo de Empresa

Um modelo qualquer pode ser definido como "uma representação (com maior ou menor grau de formalidade) da abstração de uma realidade expressa em algum tipo específico de formalismo" (VERNADAT, 1996, p. 24). Um modelo de empresa é um tipo específico de modelo formado por um conjunto de modelos que procuram representar as diferentes visões da empresa. Conforme VERNADAT (1996, p. 71), "um modelo de empresa é um conjunto consistente e complementar de modelos descrevendo vários aspectos de uma organização e que tem por objetivo auxiliar um ou mais usuários de uma empresa em algum propósito".

Deve-se notar dois aspectos nesta definição, as ênfases do autor sobre a existência de mais de um modelo destinado a diferentes aspectos ou visões, e o da necessidade de um objetivo específico. Ambos estão ligados à complexidade inerente a uma empresa ou organização. Como uma empresa é um sistema complexo constituído por elementos de características diversas (pessoas, equipamentos, computadores, documentos, *softwares*, entre outros) e em constante interação entre si, dificulta-se muito, ou mesmo impede-se, a construção de um formalismo que represente de forma útil todo esse sistema, sendo necessário trabalhar com diversas visões sobre aspectos específicos que, complementares e devidamente consistentes, possam, em conjunto, prover uma abstração da empresa como um todo. O segundo aspecto, do objetivo, é importante pois esclarece o uso que se fará desse modelo. Permite a diminuição da complexidade do modelo, enfatizando que ele deve ater-se aos aspectos necessários para a obtenção do objetivo especificado.

VERNADAT(1996) define também o conceito de **modelagem de empresa** como sendo o " ...conjunto de atividades, métodos e ferramentas relacionadas ao desenvolvimento de modelos de empresa". Existe, porém, uma visão mais ampla, que não será a adotada neste trabalho, que assume este termo como uma das diferentes abordagens que buscam a integração de empresa, sendo o modelo de empresa propriamente dito uma entre as ferramentas empregadas neste tipo de intervenção. Para a definição de modelagem dentro desta abordagem mais ampla, ver, por exemplo, LIM, JUSTER, & PENNINGTON, 1997.

2.2.2.2 Funções de um Modelo de Empresa

A função fundamental do modelo de empresa é a de permitir "um referencial semântico de unificação, um mecanismo de compartilhamento do conhecimento dentro da empresa, construído pela aplicação de princípios e ferramentas específicas de modelagem" (VERNADAT, 1996, p.18).

Esta visão compartilhada é importante para diversas atividades e processos de negócio da empresa. Portanto, a aplicação da modelagem de empresas pode ter várias aplicações. As principais foram compiladas por VERNADAT (1996) e SCHEER(1998 a):

- Obtenção de uma maior compreensão da empresa;

- Aquisição e registro de conhecimentos para uso posterior;
- racionalização e garantia do fluxo de informações;
- projeto e especificação de uma parte da empresa (funções, informação, comunicação, entre outros);
- base para análises de partes ou aspectos da empresa;
- base para a simulação do funcionamento da empresa;
- base para tomada de decisões sobre operações e a organização da empresa; e
- base para o desenvolvimento e implantação de *softwares* de forma integrada;
- documentação de sistemas de garantia da qualidade;
- implementação de sistemas de medição para o processo de negócio;

2.2.1.2 Tipos de Modelo de Empresa

Há uma distinção comum na área de estudo sobre modelagem de empresa relacionada com o conceito de modelo de referência, modelo particular ou específico, sendo, este último, também chamado simplesmente de modelo ou modelo de empresa.

Um modelo de referência é um modelo de aplicação mais ampla e geral que pode ser utilizado como referência para o desenvolvimento de modelos específicos. Os modelos específicos são modelos que representam e/ou são utilizados por uma empresa específica, numa situação específica.

KELLER & TEUFEL (1998), numa abordagem preocupada com a implantação de sistemas de gestão, classificam os modelos de referência em:

- **modelos de referência de setor:** quando descrevem um processo de negócio aplicável a um setor industrial ou conjunto de empresas específicas;
- **modelos de sistemas de informação:** que descrevem um processo de negócio relacionado com o uso de um sistema de informação específico;
- **modelos de empresa:** aqueles que servem de referência para as atividades dentro de uma empresa. Os modelos de empresa podem ser

classificados entre modelos “As-IS” e “To-Be”. Os primeiros representam a situação vigente e os segundos uma situação ideal que se espera atingir.

2.2.2 Conceitos Fundamentais de Modelagem de Empresas

Neste item, são apresentados dois conceitos básicos fundamentais sobre modelagem de empresa. O primeiro deles é o conceito de processo de negócio sobre o qual se sedimenta toda esta área do conhecimento. O segundo são os princípios que devem ser seguidos dentro da modelagem de empresa.

Processo de Negócio

O conceito de processo de negócio é a base fundamental de todos os modelos de empresa. Ele substitui a clássica visão funcional por uma visão horizontal onde a unidade de análise passa a ser a cadeia de atividades/eventos existente dentro da empresa. Este conceito tem sido adotado nas mais diversas áreas da administração de empresas (GARVIN, 1998) e é a base para diversas abordagens como Qualidade Total, custeio ABC, etc... (veja ROZENFELD, 2000 e DAVENPORT, 1994, p.167).

Uma definição clássica e simples de processo de negócio é a apresentada por DAVENPORT (1994, p.7): “uma ordenação específica das atividades de trabalho no tempo e no espaço, com um começo, um fim, e *inputs* e *outputs* claramente identificados”.

Os processos são classificados por DAVENPORT (1994) segundo os tipos Operacional e de Gerenciamento. O primeiro tipo seria formado pelos processos clássicos, aqueles que atravessam a organização transformando seus *inputs* nos *outputs* esperados pelo mercado: desenvolvimento de produto, atendimento de pedido e fabricação por exemplo. Os processos de gerenciamento são aqueles que envolvem planejamento, fixação de metas, monitoramento, tomadas de decisões e comunicação relacionados com os demais processos da organização, tais como formulação de estratégia, planejamento e orçamento, avaliação de desempenho e relatórios.

É comum também encontrar uma distinção dos processos quanto ao seu grau de estruturação, ou seja, quanto à clareza da seqüência de suas atividades. Uma definição clara a este respeito é oferecida por VERNADAT (1996b), segundo o qual:

processos estruturados são aqueles cuja seqüência de atividades é completamente definida e processos semi-estruturados são aqueles cuja seqüência de atividades é parcialmente definida.

Existem outros enfoques para o conceito de processo na literatura de administração, que foram habilmente compilados e são apresentados em GARVIN (1998). Esse autor gerou um modelo conceitual que organiza diferentes visões sobre processos. Embora esses novos enfoques não afetem diretamente a área de modelagem de empresas, são citados aqui, no intuito de mostrar as implicações possíveis para essa área. GARVIN(1988) identifica três abordagens sobre processos na literatura.

- **Processos de Trabalho (*work processes*):** abordagens em que se define processos tal qual apresentado no início da revisão, ou seja, como uma seqüência de atividades. As raízes dessa abordagem, segundo esse autor, estão nas áreas ligadas à engenharia industrial e de organizações e métodos.
- **Processos Comportamentais (*behavioral processes*):** abordagens que enfocam como processos os modelos de comportamento e a interação entre as pessoas. Tem origens nas áreas de teoria organizacional e dinâmica de grupos. O processo de comportamento é a seqüência de passos a ser seguida pelos aspectos cognitivos e comportamentais do trabalho.
- **Processos de Mudança (*Change Processes*):** enfoca os processos como uma seqüência de eventos ao longo do tempo e tem sido utilizado por pesquisadores de diversas áreas como teoria das organizações, psicologia social, especialistas de manufatura e pesquisadores relacionados com a área de aprendizagem organizacional. Estes pesquisadores descrevem como os processos a dinâmica onde indivíduos, grupos e organizações se adaptam, se desenvolvem ou se fragmentam. Nesta linha, o autor relaciona, como uma das categorias, os processos de aprendizagem organizacional.

As implicações disto para a área de modelagem residem nas críticas feitas com relação à abordagem de Processo de Trabalho, na qual se enquadram as

abordagens existentes na área de modelagem de empresa. Para esse autor, esse tipo de abordagem enfatiza sobremaneira o redesenho de processos, relegando pouca atenção à operação e gerenciamento dos mesmos. Além disso, segundo ele, esta visão de processos traria riscos de não serem considerados aspectos importantes para a organização e controle destes mesmos processos.

Como alternativa, o autor propõe um modelo conceitual (GARVIN, 1998) que agrupa todas essas abordagens numa visão única. Ela considera o fluxo de transformação das atividades (*Work processes*), o fluxo de transformações interpessoais (*Behavioral Processes*) e a transformação dessas entidades, ao longo do tempo (*Change Process*). Os últimos dois aspectos iriam exigir possivelmente formas de representações inovadoras, tais como linguagens multimídia e emprego de conceitos de aprendizagem organizacional.

Princípios de Modelagem de Empresas

Para que as técnicas de modelagem possam suportar os objetivos ao qual se destinam é preciso que elas respeitem, segundo ROSS (1977) apud VERNADAT (1996), quatro princípios básicos: que esteja bastante claro o **propósito** do modelo ou modelagem de empresas; que a **abrangência** do modelo "cubra" toda a amplitude do domínio do problema que será tratado com sua ajuda; que os **pontos de vista** adotados pelo modelo sejam coerentes, não ignorando nenhum aspecto importante do problema e que o **nível de detalhe** do modelo seja suficiente, isto é, que haja precisão e "granularidade" suficientes para o objetivo a ser alcançado com a modelagem.

Além destes princípios gerais, VERNADAT (1996) considera que uma técnica de modelagem precisa estar de acordo com um outro conjunto de princípios que guiam a construção de representações úteis. Estes princípios são apresentados à seguir e muito se parecem com princípios de modelagem de requisitos de *software*, adicionando, entretanto, alguns aspectos específicos devido à particular complexidade enfrentada pelos modelos de empresa.

Separação de Conceitos. Este é o princípio segundo o qual a técnica de modelagem precisa abordar a empresa a partir de um conjunto de elementos menores e distintos, ou seja, abordá-la "aos pedaços", formando um conjunto de elementos onde cada um representa uma única e funcional parte do domínio do problema, como

produto, processo e planta. Segundo esse princípio, não há a possibilidade de abordagens que pretendam modelar "a empresa como um todo" em um único conceito.

Decomposição Funcional. Segundo esse princípio, a técnica de modelagem de empresa precisaria permitir o mapeamento hierárquico de todas as funções da empresa, começando pela definição de funções mais macro, decompondo-as num conjunto de subfunções, até atingir a descrição das funções mais específicas. "Analisar mercado" é uma função numa empresa que poderia ser decomposta em "identificar consumidores", "desenvolver perfil de consumidores", "homologar", "produzir lote piloto", etc., que, por sua vez, poderiam ser decompostas em outras subfunções, e assim por diante, conforme o objetivo e uso do modelo.

Modularidade. Para diminuir a complexidade, facilitando o gerenciamento das alterações e a manipulação do modelo, é preciso que a técnica de modelagem permita a definição de módulos que possam ser combinados ou reutilizados num projeto de modelagem posterior.

Generalização. Este conceito é similar ao utilizado em modelagem de objetos e de dados na área de informática. Segundo este princípio a técnica de modelagem de empresas deve possibilitar a criação de classes que agrupem os objetos segundo propriedades idênticas. Assim, uma propriedade definida para uma classe é transferida para todos os objetos que compõem esta classe. O artigo de MERTINS, JOKEL e JÄKEL (1997) apresenta exemplos da utilização de técnicas de modelagem de empresas orientada a objetos).

Reuso. O método de modelagem deve permitir que o modelo possa ser construído com base em blocos predefinidos ou modelos parciais, agrupados por propriedades semelhantes, de forma tal que se garanta, tanto quanto possível, a reutilização de partes do modelo em novos modelos;

Separação entre Comportamento e Funcionalidade. Uma técnica de modelagem de empresa deve garantir a separação entre funcionalidades, ou seja, o que a empresa faz, do comportamento, como ela o faz. Deve, também, descrever os estados assumidos durante a descrição da funcionalidade proposta. Esta separação proporciona flexibilidade ao modelo, diminuindo o efeito da modificação e facilitando o entendimento;

Separação entre Processos e Recursos. Similar ao item anterior, a técnica de modelagem deve permitir a separação entre processos, no sentido de "coisas" que devem ser feitas, dos recursos, que são bens materiais e informações necessárias para fazê-lo;

Conformidade. Esse princípio diz respeito à precisão da representação. Os *constructs* utilizados pelo método de modelagem devem permitir uma sintaxe e semântica claras, consistentes, não redundantes e capazes de descrever qualquer aspecto do domínio de aplicação da modelagem de empresa;

Visualização: a técnica de modelagem deve proporcionar, também, uma linguagem de representação de fácil comunicação (fácil leitura e representação), suportada por uma representação gráfica clara e não ambígua;

Simplicidade versus Adequação: esse princípio, de forma resumida, exige "que a técnica de modelagem deve ser rica o suficiente para expressar o que precisa ser expresso" (VERNADAT, 1996, p. 82), ou seja, que ela seja a mais simples possível sem que haja perda na adequação ao propósito do modelo. Deve otimizar o compromisso (*trade off*) entre simplicidade e adequação.

Gerenciamento da Complexidade: qualquer técnica de modelagem de empresa deve ser capaz de lidar com sistemas de alto grau de complexidade, pois, os sistemas organizacionais são complexos e dinâmicos. Caso o domínio da aplicação seja pouco complexo não há necessidade de aplicar modelagem de empresa. Nestes casos o próprio bom senso e linguagens visuais comuns tornam-se suficientes para a solução do problema. Portanto, se a técnica de modelagem não for capaz de descrever um sistema complexo, sua validade é nula;

Rigor da Representação: o modelo não deve ser ambíguo e muito menos redundante, permitindo a verificação de propriedades, análises e simulações do comportamento do sistema, da empresa;

Separação entre Dados e Controles: particularmente, para os modelos de empresa é preciso que haja uma distinção entre dados, informações necessárias para a realização de um processo, dos controles, informações utilizadas pelo sistema para tomar decisões de como e quando operar (neste aspecto o modelo de empresa apresenta algumas características semelhantes aos sistemas computacionais de tempo real);

Finalmente, é preciso atentar para o fato de que todos estes princípios formam uma lista de caráter ideal. É questionável se eles são válidos para todas as aplicações de modelagem de empresa e se não há diferentes graus de importância entre cada um deles. Os autores citados não comentam esta questão.

2.2.3 Métodos para a Modelagem de Empresas

Vários métodos para auxiliar o desenvolvimento de modelos de empresa já foram criados, ou então adaptados de outras áreas, como a de sistemas de informação. Há, portanto, diversas propostas na literatura, com diferentes níveis de complexidade. Algumas se restringem a um formalismo gráfico, enquanto outras são estruturas complexas formadas por um conjunto de diferentes métodos e formalismos (gráficos, matemáticos ou lingüísticos) que suportam toda a intervenção.

Adotando o termo método como genérico, isto é, incluindo todos os meios de suporte à modelagem de empresa, verifica-se que é difícil classificar os métodos existentes, não só pela sua variação em termos de grau de complexidade e sofisticação, como também pela própria falta de consenso sobre o significado de termos como método, metodologia, arquitetura, *framework* e formalismo de modelagem.

DOUMEINGTS, VALLESPER & CHEN (1995) classificam os métodos de modelagem em três tipos de elementos: Modelos de Referência ou Frameworks de Modelagem, formados por um conjunto de modelos contemplando uma coleção fixa de *building blocks* (elementos fundamentais); Formalismos de modelagem formados por um conjunto de elementos específicos capazes de representar uma parte da realidade, relativa a um subconjunto do domínio do problema e Abordagens Estruturadas: abordagens que cobrem o desenvolvimento da arquitetura de desenvolvimento de Integração de Empresas, detalhando, passo-a-passo todo o seu desenvolvimento.

Esta classificação se torna um pouco confusa quando se verifica que os métodos da categoria Abordagens Estruturadas poderiam ser também enquadrados nas categorias formalismo e *framework*. Além disso, é estranho considerar, como fazem os autores, modelo de referência “e” *framework* juntos, numa mesma

categoria. Sua intenção é compreensível no contexto do trabalho que ele desenvolve, porém torna-se dúbio quando se pretende enquadrar os métodos de modelagem de empresa de maneira mais geral.

Outro autor importante a abordar o assunto é VERNADAT (1996), que define *Framework*, Arquitetura e Metodologias de modelagem. Segundo este autor, um Framework de modelagem é entendido como "uma coleção de princípios de modelagem, métodos, ou ferramentas relevantes para um dado domínio de aplicação"(VERNADAT, 1996, p.25). De outro modo, sutilmente diferente, ele define um *framework* para modelagem de empresa como "o conjunto formado pelo escopo, conceitos e métodos necessários para a modelagem de empresas" (VERNADAT, 1996, p.32). Para VERNADAT(1996), um *framework* é algo mais geral que Arquitetura, que ele define como um conjunto finito de componentes inter-relacionados e agrupados para formar um todo consistente funcionalmente". Assim, "diferentes arquiteturas podem ser propostas dentro de um dado framework" (VERNADAT, 1996, p.32). Por fim, VERNADAT(1996, p.32) define Metodologia de Modelagem como "um conjunto de métodos, modelos e ferramentas a serem utilizadas de uma maneira estruturada para resolver um dado problema".

Um primeiro problema com esta classificação de VERNADAT (1996) é que a definição de metodologia é ampla e se confunde com a de *framework*. Além disso, não há distinção entre os conceitos de método e modelo, presentes nesta definição de metodologia, tornando dúbio o entendimento, tanto do que é metodologia como a sua distinção em relação ao *framework*, método e modelos. Outra questão importante, com respeito a estas definições, é a diferença apresentada entre *framework* e arquitetura. Apesar de clara a intenção do autor, que vê na arquitetura uma forma específica de "instanciar" o *framework*, quando se analisa um método específico de modelagem, é difícil identificar a distinção entre *framework* de modelagem e arquitetura. Um *framework* não deixa de ter um conjunto de elementos específicos com o fim também específico e aplicável em um dado domínio. Portanto, essa distinção realizada pelo autor, apesar de mais clara que a anterior, parece pouco útil ao propósito de se classificar e entender as ferramentas e metodologias de modelagem.

A grande maioria dos autores da área de modelagem de empresas adota um ou mais destes termos anteriores como rótulo comum para todo esse conjunto (não diferenciando-os), sendo a palavra *framework* a designação mais comum. Diante, da insuficiência destas definições, tais termos foram redefinidos pelo autor e essas definições passaram a fazer parte da arquitetura de modelagem proposta, de acordo com o item 4.5.3.

Independentemente do nome, os principais métodos de modelagem são:

- ISO : este *framework* tem sido desenvolvido pelo *International Standard Organization* (ISO) no Comitê Técnico TC 184, Subcomitê 5 e grupo de trabalho 1 (ISO TC184/SC5/WG1). Esse comitê existe desde 1986 com o intuito de estudar padronizações na área de automação industrial e integração. O framework, denominado ISO Reference Model consiste de três submodelos (VERNADAT, 1996): um modelo de contexto; um modelo do chão de fábrica (Shop Floor Production Model – SFPM) e o modelo de atividade genérica (GAM – Generic Activity Model), que descreve as atividades e fluxos (de materiais, informação e recursos);
- CEN ENV 40003: este framework foi desenvolvido pelo Comitê Europeu de Padronização (European Committee for Standardization) dentro do grupo de trabalho em Arquitetura de Sistemas CIM, que faz parte do Comitê Técnico TC 310 e Grupo de Trabalho 1. Sua proposta visa padronizar às atividades na área de modelagem de empresas dando suporte a Manufatura Integrada por Computador. Este framework, que teve muita influência do CIMOSA que será descrito posteriormente, define três dimensões e procura padronizar os modelos existentes dentro delas. As dimensões são: 1. dimensão da generalização; 2. dimensão dos modelos; 3. dimensão das visões;
- **CIMOSA:** o projeto CIMOSA, European Open System Architecture for CIM, foi desenvolvido de 1986 até 1994 pelo Consórcio AMICE (Arquitetura Européia para CIM) dentro do projeto ESPRIT (EP688,

5288 e 7110) desenvolvido por um grupo de fornecedores de sistemas CIM, grandes usuários e acadêmicos (VERNADAT, 1996; KOSANKE, 1995; ZELM, VERNADAT & KOPSANKE, 1995). É um framework completo compreendendo tanto a modelagem propriamente dita como a metodologia de implantação CIM. É composto de uma definição geral do escopo e implantação de CIM, guias para a implantação e um framework que define termos e padrões para modelagem e implantação;

- **SADT/IDEF:** o SADT “nasceu pelas mãos” de Douglas T. Ross, num projeto de desenvolvimento de uma linguagem estruturada para programação de máquinas-ferramenta, no MIT, no fim da década de 60 (MARCA & MACGOWAN, 1988). Este formalismo trazia algumas características revolucionárias, que auxiliaram sobremaneira a descrição e desenvolvimento dos sistemas de software complexos que começavam a aparecer. Já o IDEF foi desenvolvido durante o projeto ICAM (Integrated Computer Aided Manufacturing), realizado pela Força Aérea Americana na década de 1980. Esta arquitetura foi elaborada compondo três diferentes modelos. O primeiro, denominado IDEF0, compõe-se da modelagem funcional do sistema e foi baseado (e, portanto, é idêntico) ao formalismo utilizado pelo SADT. Atualmente, vários outros níveis e modelos foram adicionados ao IDEFX. O primeiro nível de modelagem do IDEF, o IDEF0 que é igual ao SADT, é a ferramenta de modelagem de empresas mais utilizada na prática, graças à sua enorme simplicidade, chegando mesmo a tornar-se um padrão de fato para esse fim (CANTAMESSA & PAOLUCCI, 1998).

Destes, o método SADT e ARIS são os mais conhecidos e amplamente utilizados.

No próximo item, apresenta-se o *framework* de modelagem ARIS, por ser empregado no desenvolvimento da arquitetura. Para se obter uma discussão mais detalhada sobre cada um dos principais deles, deve-se consultar principalmente VERNADAT (1996) e RENTES (1995); adicionalmente DOUMEINGTS,

VALLESPER & CHEN (1995); SAVOLAINEN et al (1995) e REITHOFER & NAEGER (1997).

2.2.4 ARIS (ARchitecture of Integrated Information Systems)

ARIS é a sigla para *ARchitecture of Integrated Information Systems*. Este *framework* de modelagem foi desenvolvido pelo Prof. Scheer, na Alemanha, entre 1992 e 1994, e enfatiza os aspectos de engenharia de *software* e organizacionais da empresa (VERNADAT, 1996; ARIS, 1995; SCHEER, 1992; SCHEER, 1998a; SCHEER, 1998b). No ARIS são definidos quatro visões e três níveis de modelagem que podem ser visualizados nas figuras 2.1 e 2.2. Dentro de cada visão e nível, existem vários métodos (*methods*), cada um deles significando um diferente formalismo gráfico, que representa um aspecto da empresa. Um método é um conjunto de construtos gráficos, com uma determinada regra de sintaxe para relacioná-los. Os construtos são denominados objetos (*objects*) dentro desse *framework*, e, portanto, um método contém um conjunto finito de objetos e relações entre eles.

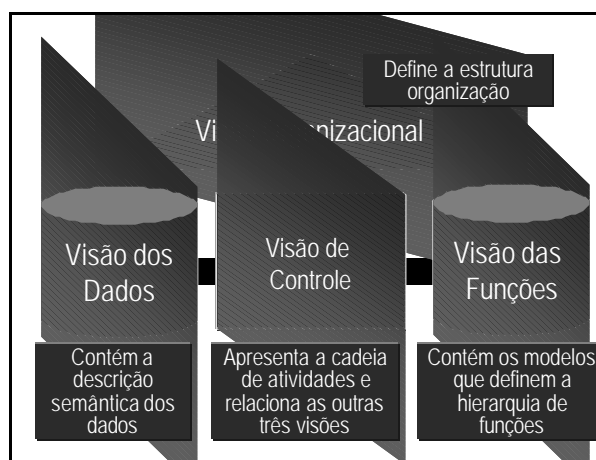


Figura 2. 1 Visões da arquitetura ARIS

Fonte: adaptado de Vernadat (1992)

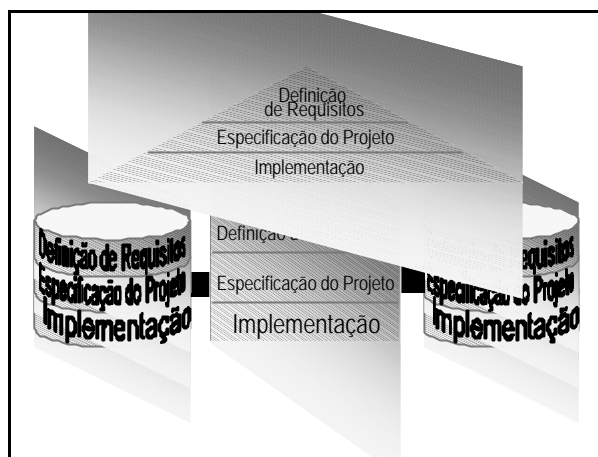


Figura 2. 2 Níveis da arquitetura ARIS

Fonte: adaptado de Vernadat (1992)

As visões presentes na arquitetura ARIS são (Fig. 2.1):

Visão Funcional: permite construir modelos que definem, de maneira hierárquica, todas as funções da empresa, começando pelas mais abrangentes (definição dos objetivos e metas dos processos de negócio), passando pelas atividades mais gerais de cada um dos processos, e decompondo-as até o maior nível de detalhe necessário. É possível chegar até mesmo ao nível de detalhe de funções de programação específicas dos aplicativos de *software*;

Visão dos Dados: é utilizada para definir os modelos de dados, partindo das definições das informações mais complexas (relatórios ou conjunto de informações), passando pelo modelo de dados e seus relacionamentos, a definição de esquemas e definindo a própria base de dados, isto é, tabelas e campos;

Visão Organizacional: permite especificar e detalhar a estrutura organizacional da empresa. Inclui desde a definição das divisões e unidades de negócios, a estrutura de cargos e seus ocupantes, até a estrutura física com os equipamentos. Ênfase especial é dada para a estrutura de informática na medida que há métodos específicos para a modelagem da rede de computadores da empresa;

Visão de Processo: é a visão que permite relacionar as três visões anteriores. Nesta visão, há métodos de modelagem específicos para definir a relação entre funções e dados, funções e organização, organização e dados e, principalmente, capazes de integrar as três funções, utilizando-se, principalmente, o conceito de evento.

Cada uma das visões é dividida em três níveis, são eles: **Requisitos**, onde são realizadas as definições mais gerais dos termos do sistema; o nível de **Projeto**, em que se define uma das formas de encontrar os requisitos e nível de **Implementação**, em que se define, detalhadamente, como o determinado elemento ocorre dentro da organização (SCHEER, 1992; e VERNADAT, 1996).

Os principais métodos da arquitetura ARIS são listados na figura 2.3. Tratam-se dos métodos mais conhecidos. O objetivo desta figura é possibilitar uma idéia mais concreta dos conteúdos de cada visão e nível. Para uma descrição completa dos métodos da arquitetura ARIS deve-se consultar SCHEER (1998b).

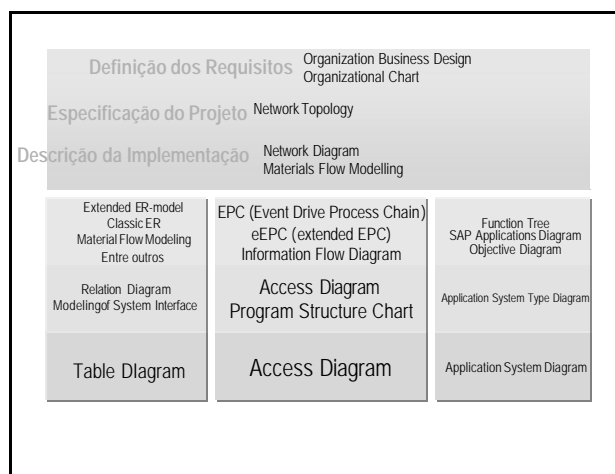


Figura 2. 3 Formalismos na arquitetura ARIS

No nível da especificação dos requisitos e visão da **organização** temos, por exemplo, um método que descreve as divisões do negócio da empresa (unidades, mini-fábricas, setores, etc.), o *Organization Business Design*, e outro, que descreve a hierarquia organizacional com cargos, responsabilidades e nomes de pessoas, entitulado *Organizational Chart*. Ainda na visão organização, porém no nível de projeto, tem-se, entre outros, um método onde se pode especificar a topologia da rede (*Network Topology*) e, no nível de implementação, a possibilidade de descrever a rede, modelando cada dispositivo de *hardware* presente em cada ponto (*Network Diagram*). Possui também um método para descrever os materiais do chão de fábrica como máquinas e equipamentos de movimentação (*Materials Row Modeling*). Portanto, na visão de funções, pode-se modelar desde os objetivos, passando pela estrutura organizacional, e seguir o detalhamento até à descrição física da alocação das pessoas, recursos de *hardware* e produtos da empresa.

Na visão de **funções** e nível de requisitos, há disponível, por exemplo, o método Árvore de Funções (*Function Tree*), que permite descrever a hierarquia de funções da empresa até o nível de detalhe que se desejar, por exemplo, projetar, movimentar, etc.. . Dentro desse mesmo nível, o *Objective Diagram* permite modelar os objetivos estratégicos da empresa, em termos gerais. Há, ainda, métodos específicos para algumas aplicações, como é o caso do *software* de integração R/3 da SAP, cuja implantação pode ser modelada mais facilmente com o emprego dos construtos do método *SAP Application System Diagram*. No nível de projeto desta visão são descritos os sistemas que realizam funções. Por exemplo, empregando o *Application System Type Diagram*, pode-se descrever cada tipo de aplicação empregada na empresa, por exemplo CAD, planilha e processador de texto. No nível de implementação pode-se especificar, empregando-se o *Application System Diagram*, cada licença de cada aplicação da empresa. Existem ainda outros diagramas onde é possível especificar funções de aplicações desenvolvidas especialmente para a organização (*in house*). Novamente, portanto, atravessando os diferentes níveis desta visão, dos requisitos até a implementação, vão-se descrevendo os objetivos e metas estratégicas, as atividades necessárias para atingi-las e os sistemas que a suportam, até atingir a descrição de funções de programação dos *softwares* necessários.

Os principais métodos da visão de **dados** são o MER (Modelo Entidade-Relacionamento) clássico e o MER estendido, o qual contém variações comumente empregadas do MER original. Esse último método inclui ainda um objeto especial denominado *Cluster*. Este objeto, proposto originalmente por Scheer dentro da arquitetura ARIS, permite especificar um conjunto complexo de dados conforme as variações empregadas pelos usuários dentro das atividades, por exemplo um formulário, um documento, etc.. Representa, portanto, um conjunto de dados que pode ter origem em uma única ou várias fontes de dados (no jargão de informática, um MER inteiro ou um conjunto deles). Este objeto tem um papel bastante importante na integração entre a visão de dados e as demais. Pois cada *cluster* mostra a informação como utilizada pelo usuário dentro do processo de negócio e, com o emprego dos métodos das visões de dados nos níveis de projeto e implementação,

são relacionados com as fontes de dados informatizadas, guardando, portanto, a relação da origem dos dados que compõem a informação utilizada pelo usuário.

Na visão de **dados**, existe, ainda, uma grande quantidade de diagramas e notações permitidas. No nível de requisitos, pode-se descrever a estrutura das informações descrevendo os *clusters* (no método ER estendido) e modelos das bases de dados empregando o método Entidade-Relacionamento (ER). Ainda neste nível é possível descrever a estrutura dos demais objetos informação, que são os recursos consumidos e transformados dentro do processo, tais como descrição das matérias-primas empregadas (*Material Row Modeling*). No nível de projeto, estes modelos lógicos da informação, descritos na camada anterior, podem ser detalhados para a implementação das bases de dados. Por exemplo, pode-se especificar, empregando o *Relation Diagram*, o modelo relacional, sobre o qual a base de dados será construída. No *Modelling of Systems Interface*, pode-se compor, hierarquicamente, entidades, eventos e termos técnicos, conforme o seu processamento no sistema. No nível mais baixo, o de implementação, é possível especificar as tabelas de dados da base relacional com a descrição dos seus respectivos campos.

A visão de **processo** descreve o processo como um todo, empregando-se os objetos descritos em cada uma das demais visões, conforme exemplificado na figura 2.4. Para tanto, possui uma quantidade muito grande de modelos específicos capazes de descrever o relacionamento entre as 3 outras visões de forma combinatória, isto é, cada conjunto de duas delas ou todas de uma única vez. No nível de requisitos, o diagrama mais importante é o EPC (*Event Drive Process Chain*) que descreve o fluxo do processo, introduzindo o conceito de evento como uma alteração significativa do *status* do sistema, no caso a empresa. Com esse conceito e método, pode-se especificar o fluxo de todas as funções, intercaladas por eventos que resultam de sua ação e que as disparam. Este diagrama é a “espinha dorsal”, a partir da qual é possível relacionar todos os elementos, sua ocorrência (especificadas pelos eventos) e permite agregar quem é o responsável por fazê-la (visão organizacional), somando-se ainda as informações e dados necessários (visão dos dados). Basta, para isto, empregar o eEPC (*extended Event Process Chain*), um método que, além dos objetos do EPC, possui os objetos mais importantes de cada uma das demais visões, todos eles podendo ser relacionados com o objeto função conforme a figura 2.4.

Apesar deste ser o método mais importante, existem vários outros dentro desta visão e nível, tal como o *Information Flow Diagram* que permite especificar o fluxo de dados. No nível de projeto podem-se representar os detalhes de como se dão as interações entre objetos das diferentes visões, dentro do processo de negócio. É composto por métodos como o *Access Diagram*, que mostra o fluxo de dados entre aplicações no nível de projeto, mostrando como os diferentes tipos de aplicações se comunicam. Outro exemplo é o método de *Program Structure Chart*, que permite especificar a estrutura da aplicação e a utilização de cada parte dela dentro do processo. No nível de implementação, o *Access Diagram* pode ser utilizado para especificar a ligação entre dados, funções e organização no nível de implementação (ligando, respectivamente campos de tabelas com funções de programas e a alocação, ou sua localização física na rede).

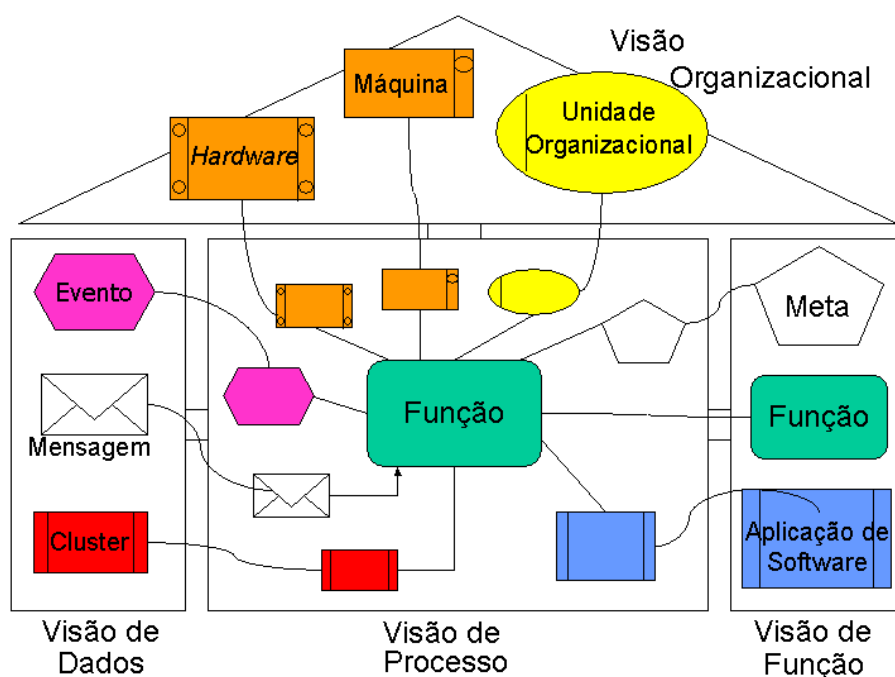


Figura 2. 4: Exemplo de integração das visões da arquitetura ARIS

Fonte: Adaptado de SCHEER (1998 b, p.37)

Novamente, alerta-se para o fato de que os métodos apresentados estão entre os mais conhecidos deste *framework*, mas representam apenas parte dele, havendo, portanto, um conjunto grande de outros métodos não apresentados.

Para o auxílio na manipulação de todos estes métodos e visões foi desenvolvida uma ferramenta computacional pela empresa IDS-SCHEER, denominada *ARIS Toolset*. Ela possui uma interface gráfica *for windows* e

disponibiliza uma grande quantidade de métodos para cada visão e nível do *framework* ARIS. Ela permite o desenvolvimento e gerenciamento de todos os modelos de maneira integrada numa meta-base de dados. Desta forma um objeto tem uma existência única, mesmo com ocorrência em vários modelos ao mesmo tempo. Isto facilita o gerenciamento e aumenta a consistência, de tal forma que a edição dos objetos em um modelo específico se reflete em todos os demais. Esta ferramenta está hoje na versão 5.0 e a empresa pretende lançar em breve a versão 6.0.

2.3 Gestão do Conhecimento

Apresentam-se, a seguir, os conceitos principais de gestão do conhecimento empregados no trabalho. É importante notar que a área de gestão do conhecimento é bastante ampla e possui atualmente diversas abordagens e linhas de pensamento bastante distintas entre si. Portanto, esta é uma revisão bibliográfica parcial, que se limita aos conceitos referentes à abordagem empregada; embora, ao fim da apresentação dos conceitos, seja feita uma pequena contextualização das diferenças frente às demais linhas teóricas presentes na área. Outra característica desta revisão é a intencional ênfase na aplicação da tecnologia de informática, por ser esse o tema central do trabalho. O primeiro item, 2.3.1, define o que é conhecimento e gestão do conhecimento, para, em seguida, definir conhecimento explícito, item 2.3.2, e uma breve apresentação do papel da Tecnologia de Informação dentro da gestão do conhecimento (2.3.3). Uma vez transmitidos estes conceitos básicos, apresenta-se uma compilação das diferentes formas de registro dos conhecimentos explícitos (2.3.4) e as tecnologias e ferramentas (2.3.5) desenvolvidas para este fim.

2.3.1 Conceitos Básicos de Gestão do Conhecimento

Definição

As empresas modernas cada vez mais enfrentam um ambiente dinâmico, convivendo com altas taxas de inovação tecnológica e um elevado nível de competitividade. Para enfrentar esses desafios, é necessário que elas se mantenham em permanente mudança e aprimoramento, desde a forma como realizam seus negócios (modelos de negócios) até como realizam suas atividades (processo de negócio) e, no nível da tarefa, garantindo que seus colaboradores estejam continuamente desenvolvendo suas habilidades e competências. Deve-se garantir que

elas sejam dinâmicas o suficiente para aprimorar-se e reinventar-se continuamente, conforme a evolução tecnológica e cultural no seu ramo de atividade.

Desta constatação, surgiu o conceito de empresa que aprende ou, como mais difundido, Learning Organization. Uma das primeiras definições e que se mantém ainda atual é a fornecida por GARVIN(1988): “uma organização que aprende” é uma organização habilitada a criar, adquirir e transferir conhecimento e, à partir deles, modificar seu comportamento e criar novos conhecimentos”. De modo semelhante, SENGE (1999) define “organizações que aprendem” como aquelas em que é despertado o comprometimento e são mobilizadas as capacidades das pessoas no sentido de se manter um aprendizado contínuo em todas as esferas da organização.

Para promover e atingir este estado, é preciso criar um conjunto de esforços dedicados a garantir e incentivar a criação, o registro e o compartilhamento do conhecimento, a denominada Gestão do Conhecimento. Terra(2000) apresenta um modelo conceitual em que organiza estas atividades em 3 planos: estratégico (visão estratégica da organização); organizacional (contendo as políticas de recursos humanos, a cultura organizacional e a estrutura organizacional); e a infra-estrutura (Sistemas de Informação e de Mensuração de Resultados);

Este tema está atualmente na pauta de discussão dos mais diversos tipos de organizações em todo o mundo (empresas, hospitais, escolas, universidade, etc..) e, sem dúvida, está entre as áreas de pesquisa em maior evolução. É uma área essencialmente multidisciplinar, que envolve conhecimentos dos campos da teoria das organizações, filosofia, psicologia cognitiva e ciência da informação, entre outros (BARROSO & GOMES, 1999).

Para complementar a definição de gestão do conhecimento é preciso definir mais propriamente o termo conhecimento. DAVENPORT & PRUSAK(1998) diferenciam três grandes classes de elementos relacionados com conhecimento dentro das empresas. São eles:

- **dado:** conjunto discreto e objetivo de fatos sobre um determinado evento ou objetos. É, portanto, a parcela quantificável e objetiva do estoque de informação e conhecimento de uma empresa, e está armazenado em bancos de dados ou documentos da empresa (exemplo: temperatura atmosférica é de 32 graus Celsius);

- **informação:** mensagem contendo um emissor e um receptor e cujo significado envolve uma nova interpretação de algo, baseado em um conjunto de dados. Exemplo: conforme um valor de temperatura e pressão atmosféricas, podemos inferir que deverá chover amanhã, portanto, tem-se uma informação a respeito do clima. Dentro de qualquer empresa, há um complexo e contínuo fluxo de informações seja por meios tecnológicos como sistemas computacionais, seja simplesmente através da interação entre as pessoas e
- **conhecimento:** é uma mistura fluida de experiências, valores, informação contextual e intuição, formando um *framework* (um “painel”) na mente de uma pessoa que a habilita a interpretar, avaliar e tomar decisões acerca de casos, experiências e/ou informações.

Conhecimento é definido, portanto, como algo que não pode ser totalmente estruturado (“não-palpável”), impossível de ser totalmente capturado e ter sua lógica “dissecada” (desestruturada), e que, por tudo isso, está presente somente dentro das pessoas. Mais ainda, o conhecimento só se manifesta quando é utilizado. Deste conjunto, origina-se uma característica importante desta definição de conhecimento: sua orientação para a ação. O conhecimento só se materializa quando empregado e, quanto maior o conhecimento de uma pessoa, mais rica e “frutífera” será sua apreciação e análise dos dados e informações disponíveis. Conseqüentemente, maior a qualidade das decisões tomadas durante sua rotina diária nas empresas.

Tipos de Conhecimento

NONAKA & TAKEUCHI (1997) realizaram uma importante distinção entre 2 tipos de conhecimento, que tem sido amplamente empregada e que permite uma maneira prática de se abordar o assunto. Os autores dividiram o conhecimento em dois tipos:

conhecimentos explícitos: são os conhecimentos estruturados e capazes de ser verbalizados. Portanto é a parte estruturada e objetiva do conhecimento. Pode ser armazenada, transportada e compartilhada por meio de documentos e sistemas computacionais. Fazem parte do conhecimento explícito, por exemplo, normas, registros de bibliográficos, livros, procedimentos de trabalho, e

conhecimento tácito: são os conhecimentos inerentes às pessoas, isto é, as habilidades que esta possui. Trata-se, portanto, da parcela não estruturada do conhecimento, a qual não pode ser registrada e/ou facilmente transmitida a outra pessoa.

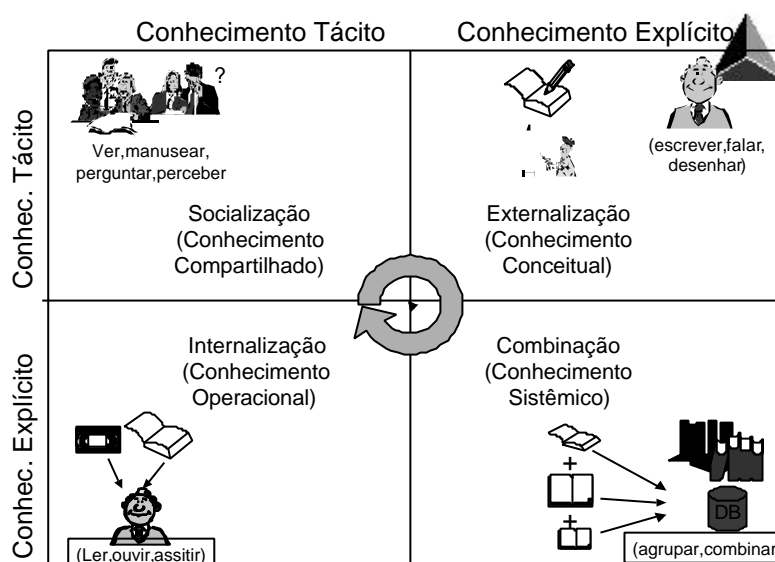


Figura 2.5 : Transformação entre Conhecimentos Tácitos e Explícitos

Fonte: SILVA & ALLIPRANDINI (2000)

Segundo os mesmos autores, existe um ciclo contínuo dentro das empresas onde conhecimento explícito esteja se transformando em tácito e vice-versa. Este ciclo é representado por ele como uma “espiral do conhecimento”, que é apresentada, simbolicamente, na figura 2.5, elaborada por SILVA & ALLIPRANDINI (2000). Cada quadrante representa um tipo específico de transformação do conhecimento. O quadrante superior esquerdo representa a Socialização, a transformação do conhecimento Tácito em Tácito. Isto ocorre quando as pessoas conversam ou discutem entre si. O quadrante ao lado, superior direito, representa a transformação do conhecimento tácito em explícito por meio da Externalização, presente no ato de escrever, registrar ou dissertar sobre um assunto. Abaixo, quadrante inferior direito, vê-se o processo de combinação que permite a transformação do conhecimento explícito em explícito, gerando o conhecimento sistêmico. Por fim, no quadrante inferior esquerdo tem-se o processo de Internalização, a partir do qual conhecimentos explícitos são colocados em prática pela pessoa, gerando novos conhecimentos tácitos. A transformação, ou a espiral, é simbolizada pela seta circular central, que

representa os conhecimentos tácitos compartilhados, sendo sistematizados em explícitos; e estes, por sua vez, sendo novamente compartilhados, sintetizados e postos em prática gerando novos conhecimentos tácitos.

Segundo essa visão, o papel da Gestão do Conhecimento é garantir e facilitar a “fluidez” destes processos em toda a organização (a manutenção e crescimento deste fluxo de transformação). A Gestão do Conhecimento, portanto, assume o papel de cuidar não apenas da infra-estrutura para registro e organização de conhecimentos explícitos, mas, também, de garantir as condições ambientais, de formação básica e motivação que faça com que as pessoas reforcem este ciclo de transformações.

Ainda outra consideração que pode ser feita a partir destes conceitos é que diferentes organizações, conforme sua estratégia de mudança e situação, deveriam enfatizar diferentes dimensões, tácita ou explícita, em seu esforço de gestão do conhecimento. É o que propõem BLOODGOOD & SALISBURY (2001). Eles apresentam um referencial teórico em que diferentes estratégias de mudança (reconfiguração de recursos ou de aquisição de recursos) exigiriam esforços diferenciados de gestão do conhecimento, mais voltados para o conhecimento tácito ou explícito. É evidente, também, que isso deve acontecer dependendo do ambiente organizacional atual da empresa. Empresas onde há pouca tradição em trabalhos com times e de interação entre as pessoas poderiam enfatizar a dimensão tácita. Acontece o contrário naquelas empresas onde há um pobre compartilhamento dos conhecimentos explícitos.

Linhas de Pesquisa e Abordagens para a Gestão do Conhecimento

As definições de aprendizagem organizacional, gestão do conhecimento e conhecimento apresentadas no decorrer deste trabalho, juntas, formam o referencial teórico que o suporta. Porém, a área de gestão do conhecimento é bastante nova, multidisciplinar e vem recebendo uma grande atenção por parte de diversos pesquisadores. Existe, portanto, uma grande quantidade de definições de conhecimento e abordagens de gestão de conhecimento, bastante distintas entre si.

SHIN, HOLDEN & SCHIMIDT (2001) realizaram uma compilação das definições de conhecimento e identificaram três “escolas” ou linhas de pesquisa. Os autores as classificam em definição de conhecimento baseado no objeto, processo e mente (“mind”). A última delas, a de conhecimento como um objeto, é formada

pelos autores que consideram conhecimento organizacional como o conjunto de artefatos que podem ser armazenados e manipulados dentro da organização. A linha de processo analisa, como conhecimento, os procedimentos detalhados para a realização das tarefas, os quais poderiam ser assimilados dependentemente do nível de capacitação dos usuários. Por último, a da mente, dentro da qual se enquadra a definição de gestão do conhecimento adotada neste trabalho, que define conhecimento como sendo algo que está dentro das pessoas e é formado pelo conjunto de habilidades para resolver os problemas.

RUBENSTEIN-MONTANO (2001) analisaram uma amostra de 32 abordagens (por eles denominadas *frameworks*), classificando-as entre 3 tipos: prescritivas, descritivas e a combinação de ambas. As prescritivas são abordagens que ditam métodos de como agir para realizar a gestão do conhecimento. Por outro lado, as puramente descritivas são abordagens que descrevem os elementos que compõem a gestão do conhecimento e as suas relações, fornecendo subsídios para as análises e tomadas de decisão sobre ação. Por fim, há algumas que misturam ambas as características anteriores. No estudo que foi conduzido por estes autores, tendo como referencial teórico principal o pensamento sistêmico (ver SENGE, 1999), foram obtidas quatro conclusões principais:

1. os *frameworks* prescritivos por natureza se atêm principalmente a um conjunto de tarefas da gestão do conhecimento, não mostrando o contexto e os objetivos de mudança na organização como um todo;
2. os *frameworks* não consideram a noção de laço duplo (*double-loop*) de aprendizagem proposta por Argyris;
3. não há ainda uma definição única do que é uma abordagem para gestão do conhecimento e
4. existem muitos conceitos comuns entre os diferentes *frameworks* (abordagens), mas ordenados e estruturados de forma muito distintas;

Para RUBENSTEIN & MONTANO (2001), as duas primeiras conclusões reforçam a idéia de que estas abordagens não são consistentes com o pensamento sistêmico e, portanto, na interpretação do autor deste trabalho, dificilmente contribuiriam para a aprendizagem organizacional na acepção de SENGE (1999) e aquela adotada neste trabalho. As duas conclusões finais, 3 e 4, indicam, para os

autores do artigo, que existe uma grande distância entre as abordagens, impedindo a comparação, relacionamento e coesão entre elas. Como recomendações, RUBENSTEIN & MONTANO (2001) propõem que novas abordagens em gestão do conhecimento devam: (1) ser prescritivas e descritivas; (2) ser consistentes com o pensamento sistêmico, isto é, (2a) as metas e estratégias da empresa devem estar relacionadas com a gestão do conhecimento, (2b) o planejamento deve ocorrer antes do início das atividades de gestão do conhecimento, (2c) aspectos culturais da organização devem ser reconhecidos e as atividades de gestão do conhecimento devem considerá-los, (2d) a gestão do conhecimento deve ser vista como um processo evolucionário; (3) as tarefas de gestão do conhecimento devem incluir encontrar, verificar, armazenar, organizar, compartilhar e utilizar o conhecimento; (4) deve existir uma distinção entre conhecimentos explícitos e tácitos e cada um deles deve ser manipulado apropriadamente e (5) as aprendizagens do tipo *single-loop* e do tipo *double-loop* devem ambas serem consideradas dentro da abordagem.

Existe, ainda, uma outra distinção bastante comum nos textos sobre gestão de conhecimento, que é a divisão dos modelos teóricos e iniciativas em duas linhas, conforme a perspectiva e o foco de ação. Haveria, assim, abordagens centradas no armazenamento ou codificação do conhecimento e outras centradas nas pessoas. A primeira é, geralmente, chamada de codificação ou sistemas. Seria formada pelas iniciativas nas quais a principal preocupação é com a implantação de sistemas de informação, arquivos e organização do conhecimento explícito. A segunda é chamada de foco nas pessoas ou abordagem de criação de redes. São abordagens cuja iniciativa principal está na identificação de competências, no incentivo ao trabalho em equipe e compartilhamento de conhecimentos tácitos. Exemplos de trabalhos que fazem esta distinção são BLODGOOD & SALISBURY (2001) e CARAYANNIS (1998). Essa distinção é sinal de que ainda persiste na área de gestão do conhecimento uma polarização entre uma tradição de pesquisa e intervenção, originária das áreas de ciência e tecnologia de informação (em especial o gerenciamento de documentos), mais preocupada com os repositórios de conhecimentos em si, e uma tradição da área de recursos humanos e administração, voltada para os aspectos culturais e o compartilhamento do conhecimento tácito.

Dessas três análises, pode-se concluir que o grande desafio ao se desenvolver uma abordagem para analisar e intervir no campo da gestão do conhecimento é unir as tradições das linhas de administração de recursos-humanos e organizacional, mais preocupadas com a dimensão organizacional, o mapeamento e desenvolvimento das competências, com a preocupação com os aspectos da tecnologia da informação. De acordo com SHIN, HOLDEN & SCHIMIDT (2001), o papel dos sistemas de gerenciamento de conhecimento explícitos deve ir além de meros repositórios. Eles devem apoiar, também, o contacto entre as pessoas, para que a parcela de conhecimento tácito que sempre há de estar “ao redor” de um conhecimento explícito possa ser também transmitida, auxiliar nas demais tarefas de diagnóstico e tomada de decisões sobre a gestão do conhecimento e proporcionar, além de informações, oportunidades de melhoria e aprendizado para os seus usuários.

Para tanto, é preciso que a Tecnologia de Informação seja pensada dentro deste nível mais amplo de transformação do conhecimento. Isto significa especificar e escolher tais tecnologias levando em consideração o seu papel dentro do esforço maior de suportar a gestão do conhecimento, aprimorando a aprendizagem organizacional.

2.3.3 Tecnologia de informação aplicada à gestão do conhecimento

A tecnologia de informação (TI), sabidamente, tem sido o grande habilitador do processo de melhoria das empresas nas últimas 3 décadas. Este novo paradigma da gestão do conhecimento não altera esta situação, mantendo o *status* desta tecnologia como sendo de importância fundamental para as empresas. Neste sentido ele até mesmo destaca e impulsiona ainda mais a adoção da TI. Porém se não há mudanças quanto à importância, há uma modificação fundamental na forma como esta tecnologia deve ser incorporada dentro das organizações. Até pouco tempo, os sistemas de informação eram elaborados de maneira estática, visando auxiliar as atividades da empresa, conforme o desenho do processo de negócio atual. Pode-se considerar que se assumia a existência de um processo de negócio ideal ou ótimo, que deveria ser encontrado e para o qual a TI seria o grande habilitador. Isto fica patente nas iniciativas da reengenharia, em conceitos como *best-practice* e na abordagem do *benchmarking*.

Seguindo o paradigma da gestão do conhecimento adotado neste trabalho, a forma de estruturação dos sistemas de informação e seus objetivos básicos tornam-se significativamente diferentes. Nesse novo paradigma, abandona-se a idéia de um processo de negócio e forma de operação ótimo ou ideal. Em seu lugar, fica a visão de uma empresa em constante mutação, capaz de alterar a sua forma de operação, gradual e rapidamente, conforme o aprendizado adquirido com os resultados alcançados, sejam eles positivos ou negativos. Como decorrência, os sistemas de informação devem ser mais do que um depósito de informações estáticas e necessárias para a tomada de decisões dentro do processo atual. Devem permitir, além disso, que elas sejam modificadas de diferentes maneiras, e que sejam adicionados nestes repositórios de dados informações nunca antes previstas pelos desenvolvedores do sistema e em múltiplas formas de representação. Precisam, ainda, servir de ferramentas de comunicação e troca de experiência entre as pessoas presentes na organização, auxiliando na tomada de decisões e documentando-as. Tudo isto para ir além do auxílio às atividades e passar a induzir o aprendizado contínuo das pessoas e a modificação nos processos. Portanto a mudança fundamental é que os sistemas de informação precisam ser elaborados para mudar e, mais ainda, ser indutores deste processo de mudanças. (THANNHUBER, TSENG & BULLINGER, 2001; e BLOODGOOD & SALISBURY, 2001). Somente se inserida dentro desta filosofia, a TI poderá colaborar com a aprendizagem organizacional e, conseqüentemente, com a melhoria, também contínua, dos processos de negócio da organização.

Em decorrência desta nova forma de inserção dentro das organizações, os sistemas de informação devem conter, além do tradicional gerenciamento de informações e documentos (como relatórios e formulários), todo um conjunto de ferramentas que suportem o trabalho em grupo (e também, no futuro, módulos, componentes ou agentes inteligentes que auxiliem nas tomadas de decisões MALHOTRA, 2001).

Desenvolver propostas de sistemas para gestão de conhecimentos envolve reunir tecnologias que suportem estes diferentes recursos, contextualizados no paradigma da aprendizagem organizacional. Abordando-se a questão dessa maneira, uma infinidade de propostas, modelos e conceitos relacionados à tecnologia de

informação precisam ser considerados. A maneira escolhida para estudar e analisar estas propostas foi divide-as em três conjuntos distintos: formas de representação de conhecimentos explícitos, tecnologias de informática e soluções de gestão do conhecimento existentes.

Para se capturar ou codificar um conhecimento explícito, é preciso colocá-lo dentro de um formato e de um suporte específicos. Por formato, entende-se a estrutura da codificação e por suporte (ou mídia, que também é uma palavra cabível) entende-se o meio físico empregado para armazenar esta estrutura. Por exemplo, um relato de um caso qualquer é um conhecimento explícito. Se ele é apresentado na forma de um texto descritivo do caso, pode-se dizer que a estrutura da codificação é uma descrição. Esse mesmo conhecimento poderia ser representado por outras estruturas como, por exemplo: um conjunto de regras, um gráfico ou uma história, entre tantos outros. Cada um desses formatos poderia ser materializado em diferentes “suportes”: uma folha de papel, um documento eletrônico de um editor de texto e outros. A este primeiro aspecto da estrutura da codificação, convencionou-se chamar neste trabalho de forma de representação do conhecimento explícito. Portanto um primeiro passo para entender os sistemas de gestão do conhecimento explícito é identificar as diferentes formas de representação de conhecimentos explícitos que eles são capazes de manipular.

Geralmente, as soluções para gestão do conhecimento empregam uma ou mais dessas diferentes formas de representação. Mas há, também, uma grande diferença na natureza dessas soluções. Elas podem ser construídas empregando-se tecnologias básicas de informática, sistemas gerenciadores de bases de dados relacionais, linguagens de programação puras como C++, linguagens de programação para internet como HTML e XML, entre outras. Nesse caso a organização pode escolher as formas de representação mais convenientes e construir sistemas *sob medida*, contendo os diferentes tipos de conhecimento explícito que desejar armazenar e utilizando as formas de representação que mais lhe convier. Estas ferramentas básicas empregadas na construção de sistemas de GC são agrupadas no trabalho como Tecnologias de Informática para gestão do conhecimento. Outra maneira é a empresa desenvolver soluções empregando sistemas comerciais específicos para a gestão do conhecimento, os quais são aqui

classificados como Ferramentas para a gestão do conhecimento. As ferramentas são construídas com base em uma ou mais tecnologia de TI e disponibilizam um conjunto específico de funcionalidades e formas de representação dos conhecimentos explícitos.

Uma Ferramenta para a gestão do conhecimento é, portanto, entendida como a aplicação de um ou mais tecnologias e ferramentas de gestão do conhecimento, as quais armazenam conhecimentos explícitos empregando um ou mais formas de representação. No item 2.3.4, apresentam-se as formas de representação e, no item 2.3.5, as tecnologias e ferramentas para gestão do conhecimento explícito. Os termos formas de representação, tecnologias, ferramentas e soluções devem ser assim entendidos, durante todo o texto, em especial no decorrer dos próximos dois itens.

2.3.4 Formas de representação dos conhecimentos explícitos

DAVENPORT & PRUSAK (1998) definem codificação como sendo a tarefa de disponibilizar o conhecimento organizacional em um formato que seja acessível. Para essa tarefa, é essencial identificar as fontes de conhecimento. Em seguida, é preciso criar meios para que os conhecimentos dessas fontes sejam registrados de maneira a ser rapidamente codificados, facilmente encontrados e eficientemente aproveitados pelos usuários (DAVENPORT & PRUSAK, 1998). Um aspecto fundamental para o sucesso na busca destes objetivos é a forma como esse conhecimento explícito é representado.

A primeira forma de representação, mais tradicional e, também, mais conhecida já foi citada anteriormente. Tratam-se dos livros, documentos internos, relatórios, listas de verificação, procedimentos, anotações, etc... A essas formas de representação denomina-se de não padronizadas, as quais foram criadas na rotina das empresas com diferentes intuítos e que, num projeto de gerenciamento de conhecimentos explícitos, precisam ser consideradas. Elas existem em grande quantidade em qualquer organização e é um grande desafio organizá-las apropriadamente. Grandes empresas mais do que nunca têm investido em suas antigas bibliotecas, centros de documentação, manuais de qualidade e outras fontes deste tipo de informação, objetivando que partilhem do esforço de gestão do conhecimento, disponibilizando o conhecimento explícito armazenado nesta forma

não padronizada de representação. Deve-se alertar ainda que este esforço não deve ser restrito à esfera da empresa. ANAND, MANZ & GLICK (1998) afirmam que não se pode descuidar dos conhecimentos que tem origem externa à organização, sendo necessário incluí-los nesta sistemática de identificação e registro.

Um meio mais específico de armazenar conhecimentos explícitos, e um dos primeiros a ficarem mais conhecidos dentro da área de gestão do conhecimento, é o formato denominado aqui de Mapas do Conhecimento. São os conhecimentos explícitos que apontam onde está o conhecimento, isto é, pessoas, documentos e bases de dados que contêm os diferentes conhecimentos explícitos (DAVENPORT & PRUSAK, 1998). Um mapa específico é, *per si*, um conhecimento explícito que facilita enormemente o processo de transformação e difusão do conhecimento. Eles permitem que as pessoas identifiquem fontes de conhecimentos explícitos e, além disso, quais são os indivíduos que “carregam” algum conhecimento tácito sobre algum assunto. Um mapa eficiente deve ser construído conforme os requisitos das pessoas dentro da organização e deve apontar tanto para conhecimentos que estejam internos na empresa como externos (DAVENPORT & PRUSAK, 1998). Deve-se atentar ainda que a palavra mapa não significa necessariamente uma representação gráfica. Considera-se por um conhecimento na forma de mapa aquele que sintetiza a localização de outros conhecimentos explícitos, independente se este mapa está na forma de uma lista, uma imagem, uma base de dados, entre outros.

Outra forma de representação é a baseada em Narrativas, ou seja, histórias sobre acontecimentos em um determinado projeto ou atividade. Segundo DAVENPORT & PRUSAK (1998), esta é uma maneira interessante de se armazenar conhecimentos mais próximos ao tácito, isto é, mais desestruturados. Isto porque narrar, ou seja, contar histórias é uma forma bastante natural e fácil para se expor e demonstrar um conhecimento complexo, ainda não sistematizado. Porém os autores não mencionam que, apesar de facilitar o registro, este formato tem como inconveniente a dificuldade de busca (como encontrar um conhecimento que esteja implícito no relato ?) e a necessidade de um esforço maior de interpretação. Geralmente, para contornar este problema, ao menos em parte, emprega-se uma estrutura de tópicos padronizada, que deve ser seguida pelo narrador, contendo, por exemplo, contexto, problema, solução adotada e resultados encontrados.

Uma terceira forma é a linguagem estruturada, na qual o conhecimento é expresso numa linguagem formada a partir do idioma corrente, adicionando-se restrições e padrões que tornam o texto menos ambíguo. É o caso do *English-Like*, utilizado na área de análise de sistemas. Uma forma semelhante de representação, e que tem origem próxima, é o registro por meio de regras, conforme as abordagens de regras de produção da área de Inteligência Artificial e Sistemas Especialistas (veja por exemplo RODGERS et al, 1999 e SU, HWANG & LIU, 2000). Nesse meio de representação, o conhecimento é registrado como um conjunto de regras inter-relacionadas que, por serem estruturadas, tornam-se menos ambíguas e de mais fácil localização. Entretanto realizar o registro torna-se cada vez mais difícil dependendo da complexidade do conhecimento. Uma forma de representação mais flexível, que, de certa forma, segue o princípio das regras, são os denominados Mapas Cognitivos. Existem muitas variações, mas, em geral, eles são formados por dois construtos: sentenças e flechas. As sentenças são frases próximas às regras, que são ligadas por flechas que podem indicar consequência, antecedência, deduções ou induções; ou mais de uma delas, conforme a abordagem (consulte NOVAK, 1998, para uma descrição mais detalhada de uma abordagem de mapa e seus fundamentos teóricos e RAMESH & TIRWANA, 1999, para um exemplo de ferramenta de gestão do conhecimento baseada neste tipo de representação). Esses mapas são hoje aplicados em diferentes áreas do conhecimento, desde a Pesquisa Operacional (PO *soft*) até a análise de discursos em ciências humanas. A origem deles está no movimento da psicologia cognitiva.

Pode-se representar conhecimentos explícitos por meio das ontologias, que são especificações explícitas dos conceitos relativos a um determinado domínio (veja STUDER, BENJAMIN & FENSEL 1998; O'LEARY, 1998; DIENG). As ontologias também são utilizadas como apoio à construção de modelos de referência, apresentados no item 2.2 deste trabalho, os quais, por sua vez, também são uma forma de representação de conhecimentos explícitos (ABECKER et al, 1998; DIENG *et al*, 1999).

Portanto poderíamos resumir as formas de representação de conhecimentos explícitos em:

1. não padronizadas;

2. mapas do conhecimento;
3. narrativas;
4. linguagens Estruturadas;
5. regras;
6. mapas Cognitivos;
7. ontologias e
8. modelos de Processo de Negócio.

Essa é uma primeira revisão que poderia ser aprofundada por meio de um levantamento bibliográfico especialmente projetado para esse fim, com maior abrangência e maior cuidado, em comparação ao apresentado neste trabalho.

2.3.5 Tecnologias e Ferramentas para Registro de Conhecimentos Explícitos

Apresentadas no item 2.3.3, as soluções de tecnologias de informação disponíveis para auxiliar o gerenciamento do conhecimento foram estudadas em dois grupos, separadamente, denominados tecnologias para a gestão do conhecimento e ferramentas para a gestão do conhecimento.

Descrição das Tecnologias

As tecnologias são as soluções básicas de TI, mais genéricas, que podem ser empregadas para a construção de ferramentas dos mais diversos tipos. São, por exemplo, as linguagens de programação básicas e sistemas gerenciadores de bases de dados, descritas conforme o texto abaixo.

- **Linguagens de Programação** - a mais direta e óbvia categoria de tecnologias básicas para a construção de sistemas de gestão do conhecimento é formada pelas linguagens de programação. Em termos de construção de sistemas para *desktop* ou redes locais, as linguagens mais clássicas são o C++, Visual Basic e Pascal. Houve uma grande revolução com o desenvolvimento da tecnologia da internet, que popularizou a linguagem Hypertext Markup Language, popularmente conhecida como HTML (TANENBAUM, 1996). Mais recentemente, uma linguagem, em especial, tornou-se bastante importante na construção de sistemas de gestão do conhecimento, o JAVA. Ela foi desenvolvida pela SUN Microsystems para automação de eletrodomésticos. Projetada com a

sintaxe e comandos próximos ao C++, ela possui como principais vantagens ser totalmente orientada a objeto e a simplicidade. Contém uma arquitetura baseada em uma JAVA Virtual Machine (JANDL, 2002), uma camada de *software* que “roda” acima do sistema operacional e permite que os objetos dessa linguagem funcionem independentemente do sistema operacional utilizado. Ela se disseminou bastante porque, combinada com HTML, permite construir páginas dinâmicas, uma vez que muitos *browsers* possuem uma Java Virtual Machine. Foram desenvolvidas também linguagens de *script* especialmente projetadas para esse ambiente *web*, tais como o PERL e *JavaScript*, atualmente bastante utilizadas. Está em andamento uma grande transformação na indústria de *software* com o surgimento do sucessor do HTML, a XML, ou *Extended Markup Language*, que permite a construção de páginas dinâmicas e, mais importante, numa forma de programação que separa estrutura de dados, a programação visual das páginas e o conteúdo (PIMENTEL & TEIXEIRA, 2001). O padrão de programação XML está revolucionando a indústria de *software* por possuir um potencial enorme de vir a se tornar um padrão mundialmente aceito para construção de componentes e para troca de dados entre aplicações baseadas na *web*.

- **Internet/Intranet/Extranet/Portais** - a internet é a rede mundial que se formou a partir da interconexão de diversas redes e computadores empregando-se os protocolos TCP e IP (TANENBAUM, 1996). No início da década de 90, essa rede começou a se ampliar enormemente com o surgimento da tecnologia WWW ou World Wide Web. Essa tecnologia (constituída de um tipo de sistema, denominado *browser*, e um protocolo de rede, o *www*) permitia uma fácil e agradável busca de informações dentro da rede, empregando um formato de hipertexto (TANENBAUM, 1996). Em um curto espaço de tempo, a rede mundial, antes um nicho de super especialistas em TI (capazes de dominar os diferentes protocolos, aplicativos e sistemas operacionais), tornou-se um acessível instrumento de trabalho do grande público. A internet está sendo hoje amplamente utilizada dentro das empresas no desenvolvimento de diversos tipos de

sistemas de informação. A principal vantagem é que ela permite agrupar, num mesmo local, informações oriundas de diferentes fontes ou diferentes bases de dados, cada qual empregando diferentes plataformas. Além disso, a interface com o usuário, via *browsers*, tem se tornado um padrão efetivo na indústria de informática e permite a inclusão de imagens, sons e vídeo, facilitando o aprendizado e uso dos sistemas. Com relação à Gestão do Conhecimento, as principais vantagens que esta tecnologia traz são: a possibilidade de interligar sistemas heterogêneos (permitindo unir num mesmo local conhecimentos explícitos registrados em diferentes sistemas espalhados pela empresa); a possibilidade de conectar informações internas e externas (onde a externa pode ser a própria rede da internet); alta flexibilidade em termos de tipo de registro (num sistema baseado em web pode-se disponibilizar, via *download*, arquivos de *software* independentemente do padrão) e a facilidade de uso e aprendizado. Os sistemas de gerenciamento do conhecimento tendem a utilizar amplamente essa tecnologia. Empregando-a, pode-se construir as chamadas *intranets/extranets*, que são sistemas (um *site* na linguagem da Web) englobando, por meio da internet, a empresa e principais clientes ou unidades externas (CARAYANNIS, 1998). Outro tipo de tecnologia são os portais do conhecimento que são *sites* cuja característica é a de ser o ponto principal de partida para a busca de registros de conhecimento explícito. DIENG et al (1998), KLOOSTER et al (1997), AGOSTA (1999); MAURER (1998) e SCOTT(1998) abordam o emprego da tecnologia *web* na gestão do conhecimento.

- **Tecnologias *Server-Side* de programação para WEB** - uma categoria de difícil delimitação é a formada pelas tecnologias denominadas e agrupadas como *server-side*, isto é, a programação dinâmica das páginas é baseada nos servidores e não no *browser* como as linguagens de programação tradicionais para *web*, como JAVA e JAVAScript. As duas principais: o *Active Server Pages* (ASP) da Microsoft (BATTISTI, 2002) e o *Java Server Pages* (JavaSP) da SUN Microsystems. Esta tecnologia se baseia na lógica de construir tradutores dentro do servidor *web*, que

transformam páginas HTML especiais em páginas HTML padrão, as quais são por sua vez enviadas para o *browser*. O que as páginas que residem no servidor têm de especial é a característica de, em meio à programação padrão escrita em HTML, existirem “pedaços” de código, identificados por *tags* especiais, que instruem o servidor sobre como completar a página, por exemplo, adicionando uma lista dinâmica de nomes e endereços. Este código especial pode ser escrito em diferentes linguagens, conforme a necessidade e o desejo do usuário. Por exemplo, no ASP pode-se escrever o código em VBScript, C++ ou Java, bastando que se identifique anteriormente, no início do próprio “pedaço” de código, qual delas está sendo empregada. A mais nova destas tecnologias é o PHP, especialmente projetado para o ambiente Linux, que emprega uma linguagem específica próxima ao JAVA (CONVERSE & PARK, 2001). Por tratar-se de uma plataforma de *software*-aberto (gratuita e de fácil acesso), tem sido amplamente difundida e utilizada. Essa categoria de tecnologia fica num meio termo entre as duas primeiras apresentadas, linguagem de programação e tecnologia *Web*, pois são soluções compostas por linguagens de programação, interpretadores e sistemas operacionais, entre outros componentes, que precisam funcionar conjuntamente para gerar a aplicação. As vantagens são segurança, rapidez e controle das páginas *web* geradas, pois, o *browser* do usuário “visualiza” apenas código HTML padrão.

- **Gerenciadores de Bancos de Dados** - a tecnologia de Gestão de Banco de Dados é uma das mais básicas da atual indústria de *software*, pois, é aplicada conjuntamente com a maioria das demais tecnologias e suporta grande parte das ferramentas de gestão do conhecimento, principalmente as de emprego corporativo. São empregadas para o armazenamento e controle dos dados. Também estão bastante difundidas, de tal modo que grande parte das empresas já utilizam sistemas gerenciadores de bases de dados para suas diversas aplicações, sejam elas comerciais ou desenvolvidas dentro da própria empresa (vide DATE, 1999). Assim, conforme o grau de utilização e da estrutura de informática presente na

organização, pode-se criar, de maneira rápida e fácil, um sistema simples para registro e manutenção de Mapas de Conhecimento, Narrativas ou mesmo registros de regras empregando-se esta tecnologia. A vantagem em relação ao registro manual (em papel) é a facilidade no momento da busca e registro. Ela permite a existência de campos e listas de palavras padronizadas facilitando o cadastro e, ao mesmo tempo, a busca, pois evita redundâncias e erros de digitação. É vantajosa frente à mistura dos dados com os aplicativos porque novas versões dos sistemas podem ser criadas, preservando-se a base de dados e, portanto, os dados históricos da empresa.

- **Agentes Inteligentes** - são *softwares* que realizam uma determinada tarefa e são capazes de se adaptar, isto é, de agir de maneira diferente, conforme informações obtidas do ambiente (CARAYANNIS, 1998). É uma tecnologia nova e está estreitamente relacionada com a WEB, na medida que há grandes possibilidades de criação deste tipo de *software* nesse ambiente. Uma vez desenvolvidos, esses agentes serão de extrema importância nos sistemas de gestão do conhecimento, pois devem permitir buscas inteligentes, maior flexibilidade na “customização” da informação, oferecer auxílio nas tomadas de decisões (com cálculos e análises de dados), entre outras vantagens que potencializarão a tecnologia WEB (AGOSTA, 1999; MAURER, 1998).
- **DataMining e Datawarehouse** - são grandes depósitos de dados formados por meta-bases de dados que interligam informações de diversas bases, muitas vezes com plataformas heterogêneas e empregando estrutura de dados multidimensionais. *DataMining* é a tecnologia, baseada em técnicas de inteligência artificial e inferência, cujo intuito é cruzar e obter informação da análise das grandes massas de dados armazenadas em *Datawarehouse*. Essas tecnologias podem ser empregadas conjuntamente, permitindo a construção de memórias organizacionais e a “mineração” ou descoberta de novos conhecimentos, nas massas de dados que vêm sendo criadas dentro das empresas. Ela permite buscas complexas e a descoberta de novos dados (DIENG et al, 1999). Porém analisando-se os modelos

propostos e os artigos que retratam a tecnologia de informática na área de gestão do conhecimento, percebe-se que esta ferramenta parece pouco empregada. Uma das razões pode ser os altos custos e a complexidade dos projetos de *Datawarehouse* e *Datamining*.

Descrição das Ferramentas

Há uma grande quantidade de ferramentas sendo comercializadas, e propostas por pesquisadores, com o rótulo de Gestão do Conhecimento. Elas abrangem desde soluções mais pragmáticas, simples e principalmente baseadas em WEB, até sistemas mais complexos, oriundos das áreas de Inteligência Artificial e Sistemas Especialistas. A principal distinção entre estes dois “polos” é quanto à geração de inferências. Enquanto estes últimos empregam sofisticados recursos da tecnologia da informática para gerar automaticamente inferências sobre os registros, as primeiras ferramentas limitam-se ao registro e armazenamento, cabendo ao usuário humano a análise e a inferência.

MATTA, CORBY & PRASAD (1998) propõem uma biblioteca genérica para o desenvolvimento de sistemas para registros de conhecimento sobre conflitos em atividades de engenharia simultânea. Nesta proposta, o conhecimento explícito é registrado na forma de Narrativas, contendo textos estruturados por um conjunto finito de atributos.

Na linha de sistemas especialistas, RODGERS et al (1999) mostram um exemplo de desenvolvimento de um sistema baseado em regras, denominado WEBCADET, para registro de conhecimentos sobre a fase de conceito dentro do processo de desenvolvimento de produto. Já REIMER (1997) apresenta o EULE2, um sistema destinado a registrar e gerenciar o conhecimento explícito em empresas de escritório, que utiliza uma linguagem estruturada, mas que, indo além, faz uma integração com regras e o que ele chama de memória organizacional, isto é, conhecimentos explícitos no formato não-padronizado, como documentos diversos e textos.

Em RAMESH & TIRWANA (1999) e TIRWANA & RAMESH (2001) são descritos o desenvolvimento e a aplicação de um sistema para gerenciamento de conhecimento em times de desenvolvimento de produto. Nele, são aplicadas

representações do tipo Narrativas, somadas com representações do tipo Mapas Cognitivos. Ambos são empregados para representar e armazenar as discussões travadas pelo grupo e as decisões tomadas, para que possam ser recuperadas no futuro, o que é uma solução bastante interessante para o mapeamento de decisões. A aplicação (TIRWANA & RAMESH, 2001) foi realizada numa empresa do ramo de imprensa, em equipes de desenvolvimento de produtos de informação como CD's, jornais e treinamentos, entre outros.

DIENG et al (1999) apresentam uma revisão sobre técnicas e métodos para registro de conhecimentos explícitos, por eles denominados memórias organizacionais, em que citam três outros sistemas não apresentados aqui: CYGMA, REX e MKSM. Em sua análise, eles concluem que a tendência a ser seguida para os sistemas de memória organizacional é o emprego de mais de uma das formas de representação na construção dos conceitos.

Um exemplo de emprego de múltiplas representações é a proposta de ABECKER et al (1998). Os autores propõem um modelo de três camadas representadas por uma pirâmide. A inferior, na base da pirâmide, é denominada **nível de objeto** e contém conhecimentos explícitos não padronizados e casos na forma de narrativas. Na segunda camada, superior ao nível de objeto, se encontra o **nível descritivo**, contendo ontologias que resumem, contextualizam e guiam os usuários frente ao conhecimento registrado na camada inferior. Esse nível tem por objetivo realizar uma precisa seleção e controle de acesso à informação e aumentar a compreensão do usuário na interpretação dos conhecimentos explícitos no nível de objeto. Na ponta da pirâmide há a **camada de aplicação** que contém conhecimentos explícitos na forma de modelos de referência cuja finalidade é facilitar a busca do conhecimento e fornecer ao usuário o contexto, em termos de etapa e atividade do processo de negócio em que o conhecimento explícito deve ser utilizado.

Um modo diferente e interessante de unir as formas de representação de conhecimentos explícitos é apresentado por THANNHUBER, TSENG & BULLINGER (2000), numa abordagem mais operacional, em que a tecnologia de informática é apresentada dentro de um modelo conceitual de gestão do conhecimento baseado no conceito de "autopoiesis". Os autores propõem 2 (dois) elementos principais: um ambiente de processamento de declarações e um

framework. O conhecimento é armazenado no primeiro destes elementos, de acordo com um conjunto de blocos de processo de negócio (*process building blocks*). Os blocos são posteriormente empregados na montagem de um processo de negócio específico, empregando uma ontologia e um conjunto de objetos específicos presentes no *framework*. O objetivo dos autores é criar uma organização capaz de se auto-regular e reagir rapidamente às mudanças do ambiente, semelhantemente à auto-regulação de sistemas vivos, expressa nesse conceito de autopoiesis. Se os autores foram particularmente felizes, contextualizando a ferramenta dentro de um modelo teórico para a gestão do conhecimento, parecem reducionistas frente às demais ferramentas, ao considerar, praticamente, um único tipo de representação de conhecimento explícito que são os modelos de processo.

Existe também um conjunto de ferramentas comerciais, vendidas como solução para a gestão do conhecimento. Analisando-as, foi possível agrupá-las em oito (8) categorias, conforme a semelhança nos tipos de funcionalidades oferecidas e sua origem.

- **Módulos de Recursos Humanos de Sistemas Integrados** - atualmente grandes sistemas corporativos, os chamados ERP (Enterprise Resource Planning), como SAP e PeopleSoft, estão incluindo, em seus módulos de recursos humanos, funcionalidades para registro e armazenamento de conhecimentos (DANVENPORT & PRUSAK, 1998). Por enquanto, as funcionalidades desses sistemas se limitam, principalmente, aos dados sobre as pessoas, sua formação, carreira, experiências e conhecimentos dominados, aproveitando-se da estrutura de dados necessária para as funcionalidades principais destes sistemas, tais como folha de pagamento e controle financeiro (DAVENPORT & PRUSAK, 1998). Comparando com os meios de registro, portanto, limitam-se, principalmente, à criação dos Mapas de Conhecimento. Uma forma de escapar dessa limitação pode ser a utilização de sistemas complementares de registro do conhecimento. São sistemas com funcionalidades mais sofisticadas de gestão do conhecimento explícito (registros de Narrativas, Regras e outros), que trabalham integrados com esses grandes sistemas corporativos (DAVENPORT & PRUSAK, 1998). Independentemente da forma,

empregando ou não sistemas complementares, parece haver uma tendência dos fornecedores desses grandes sistemas integrados, em incorporar conceitos de Gestão do Conhecimento. Isto pode ser motivado pela própria mudança que vem ocorrendo nos setores de Recursos Humanos das grandes empresas, que, cada vez mais, se vêem como gestores de competências e conhecimentos. Mais recentemente, dois “*players*” (concorrentes) principais neste setor, a SAP e a ORACLE, têm adotado o termo *Business Intelligence* como padrão, sendo as funcionalidades ligadas à gestão do conhecimento, incorporadas abaixo deste novo rótulo (SAP 2002; ORACLE 2002).

- **Ferramentas de Modelagem** - ferramentas sofisticadas de modelagem estão também incorporando módulos para registro de conhecimentos explícitos. Por serem utilizadas na atividade de descrição dos processos de negócio e geração de documentações para a padronização, tornam-se propícias para a geração de mapas de conhecimento. É o caso da ferramenta ARIS da IDS-SCHEER que está comercializando uma configuração específica de sua *suite* de ferramentas de modelagem especialmente para projetos de gerenciamento do conhecimento (SCHEER, 1998b). Seu emprego nesses projetos tem sido no levantamento das competências e padrões de trabalho empregados na empresa.
- **Ferramentas de Workgroup Computing** - trata-se de uma classe de soluções de suporte ao trabalho colaborativo. Essas soluções compreendem funcionalidades integradas de Vídeo-Conferência, Agenda Compartilhada, Email, Editores de Texto, Editores de Figura, Visualizadores, Diários, entre outras. Elas auxiliam a gestão do conhecimento facilitando a organização e armazenamento dos conhecimentos explícitos não-padronizados e a comunicação entre os membros da organização. As duas soluções comerciais mais difundidas caminham fortemente nessa direção; são elas o Microsoft Office e o Lotus Notes. O primeiro é uma extensamente difundida *suite* de programas da Microsoft, que hoje é comercializada como uma ferramenta para obtenção

do *Knowledge Management* (MICROSOFT, 1999). Com relação ao seu desenvolvimento, a tendência é que, além desse foco em conhecimento, a integração entre os diversos usuários seja centralizada numa *intranet* gerada por uma das soluções incluídas nesse pacote (MICROSOFT, 2000). Já o Lotus Notes, certamente, é a ferramenta mais amplamente utilizada no desenvolvimento de sistemas de registro de conhecimentos explícitos. As grandes consultorias internacionais, maiores e mais antigas usuárias de registros de conhecimentos explícitos, têm construído e utilizado soluções baseadas nessa ferramenta há vários anos (DAVENPORT & PRUSAK, 1998). O *site* “Collaborative Tools and Software” traz uma lista de produtos dessa categoria, veja PROSCI (2001).

- **Sistemas de gerenciamento de documentos (GED, PDM, EDM ou cPDM):** existe uma classe de soluções que hoje são comercializadas como sistemas para gestão de conhecimento, que têm origem ou se restringem às funcionalidades próximas aos sistemas de gerenciamento de documentos, também conhecidos como Gerenciamento Eletrônico de Documentos (GED), *Electronic Data Management* (EDM), Product Data Management (PDM) e, mais recentemente, como *collaborative Product Management*, sigla cPDM (CIMDATA, 2001). Essa classe de sistemas suporta a organização e controle dos documentos provendo: armazenamento de documentos permitindo sua classificação por atributos *customizados* e poderosas ferramentas de busca, controle de fluxo do processo de trabalho (*workflow*), funcionalidades de segurança incluindo *check in* e *check out* de documentos e *vault* e visualização de documentos, entre outras, que permitem o armazenamento seguro de documentos eletrônicos. A diferença do PDM e cPDM em relação aos demais é que estes são mais sofisticados e voltados para a engenharia. Eles contêm funcionalidades específicas, como controle da lista de materiais (Bill of Material, BOM) e algumas funcionalidades de gerenciamento de projetos. Existem sistemas para a gestão do conhecimento que vieram dessa tradição, ou seja, eram antigos sistemas

GED ou PDM, que hoje são comercializados com o rótulo de gestão do conhecimento (veja, por exemplo, o caso do Docushare, XEROX, 2001). Ou então, são sistemas desenvolvidos para GC, mas se limitam a este tipo de funcionalidade, servindo principalmente ao armazenamento de registros no formato não-padronizado. Algumas delas são bastante sofisticadas e permitem “customizações” via programação e, por isso, poderiam ser utilizados como base para a construção de soluções mais completas de GC que considerassem, também, outras formas de representação.

- **Portais de Gestão do Conhecimento:** essa classe é a que parece estar em maior ascensão, havendo muitas pequenas empresas e uma infinidade de produtos. Reúne produtos que permitem a criação de portais corporativos e/ou intranets ou extranets. As principais funcionalidades são o armazenamento de documentos via *upload*, unida com funcionalidades simples de trabalho colaborativo tais como agenda compartilhada na rede. Um exemplo é o produto “intranet.com” (INTRANETS.COM, 2001). Essas ferramentas trabalham com conhecimentos explícitos, principalmente nos formatos não-estruturado e mapas.
- **Sistemas de gerenciamento de projeto:** a classe de sistemas para gerenciamento de projeto é mais tradicional. Foi iniciada há muitos anos com soluções que auxiliavam o controle do cronograma e com otimizações do tipo do cálculo de caminho-crítico. Com o passar dos anos e o desenvolvimento da tecnologia de informática, eles têm também seguido o conceito de colaboração. Atualmente, esses sistemas têm caminhado no sentido de criação de portais (*workspaces* ou *intranets*) colaborativos para os times de projeto. A idéia é criar um ambiente virtual onde todo o time de projeto tenha acesso às informações sobre o andamento, principais documentos, reuniões, ferramentas de comunicação e agenda compartilhada. Portanto se aproximam bastante do conceito dos novos PDM’s e cPDM’s; e são bastante aptos para suportar o gerenciamento de conhecimento. Veja, por exemplo, a *suite* de soluções para gerenciamento de projeto da empresa Primavera (PRIMAVERA,

2001). Para uma lista de sistemas de gerenciamento de projetos veja PROJECT MANAGEMENT CENTER (2001).

- **Sistemas *Peer-to-Peer* (GROOVE)** - a classe de sistemas *peer-to-peer* é bastante nova e, pode-se até mesmo dizer, revolucionária em relação às apresentadas anteriormente, pois se baseia em um novo conceito de organização dos dados. Enquanto as soluções anteriores caminham para o paradigma *web* com servidores centrais de dados, os sistemas *peer-to-peer* buscam a descentralização. O princípio fundamental, similar ao do *Napster*, é o de criar um sistema que permita a sincronização e compartilhamento dos documentos entre cada computador pessoal espalhado na rede, eliminando a necessidade de um servidor de dados centralizado. Há quatro anos, em 1998, Ray Ozzie, famoso por ter criado o Lotus Notes, abriu a empresa GROOVE Networks (GROOVE Networks, 2001) para desenvolver e comercializar um *software* para a gestão do conhecimento que fosse baseado nessa idéia, também denominado GROOVE. Este sistema, que atualmente está na sua primeira versão, permite criar espaços virtuais específicos para times de projeto lotados na mesma rede (local ou internet). Os documentos e informações armazenados por um dos membros podem ser visualizados pelos demais. Caso desperte, em algum desses outros usuários, o interesse pelo documento, basta que ele o transfira para o seu computador pessoal, empregando um simples processo de *download*. Possui ainda mecanismos de sincronização e ligação com bases de dados e conjunto grande de ferramentas de comunicação, como agenda, grupos de discussão e outros. É um *software* capaz de ser bastante modificado, quase uma *shell*, permitindo personalizar as funcionalidades e as características deste *workspace*. Este sistema, pelo que se saiba, é um dos únicos voltados para a gestão do conhecimento empregando essa tecnologia, que possui histórico de sucesso apenas no intercâmbio de arquivos de música, livros e outras áreas, fora do âmbito corporativo. Além disso, as implantações comerciais são recentes, principalmente no ano de 2001. Portanto é uma incógnita o futuro das soluções baseada nessa tecnologia que, pelo seu

caráter inovador, deve ainda enfrentar uma grande resistência dentro do ambiente empresarial (DUTTON, 2001). Seu criador e a empresa *Groove Networks* apostam fortemente neste caminho, alardeando as vantagens de menor custo de manutenção dos dados, maior flexibilidade, maior aproximação dos membros do grupo de trabalho, entre outras vantagens (GROOVE, 2002; e TECHNOLOGY REVIEW, 2001). Porém não se pode esquecer o custo do aumento do tráfego na rede e problemas relativos à segurança dos dados (SELTZER, 2002).

- **Ferramentas para desenvolvimento de sistemas especialistas, engenharia baseada no conhecimento e ontologias** - existe um grupo de ferramentas para gestão do conhecimento que tem origem na área de sistemas especialistas. É formado por soluções comerciais que permitem o armazenamento de regras ou criação de ontologias, formando uma base de conhecimento. Sob esta base, é possível construir sistemas de apoio à tomada de decisões relativas ao domínio do conhecimento armazenado. Portanto esta classe de ferramenta pode ser aplicada para a gestão do conhecimento quando se deseja armazenar o conhecimento no formato de regras ou como ontologias. Um exemplo é o ExpertRules (ATLAR, 2002).

Análise das ferramentas para gestão do conhecimento

Como foi possível demonstrar, no decorrer deste item, existem muitas tecnologias e ferramentas que podem ser empregadas para auxiliar a gestão do conhecimento. Elas estão atravessando um período de mudanças em que padrões de fato, como as páginas HTML dinâmicas e bases de dados relacionais, convivem com novas promessas como as linguagens XML e a possibilidade de manutenção de dados distribuídos na rede.

Estas diferentes ferramentas e tecnologias, se bem exploradas pelos profissionais da área da gestão do conhecimento, podem gerar soluções que auxiliem grandemente a cooperação, a criação e compartilhamento de conhecimentos. Mas, esta revisão demonstra, também, vários desafios e problemas que estes profissionais podem enfrentar ao caminhar nesta direção.

O primeiro deles, que parece ser bastante evidente a partir da revisão realizada, é o fato das diferentes categorias de ferramentas apresentadas se especializarem em formas diferentes de representação de conhecimentos explícitos. Os sistemas de gerenciamento de documentos e portais são interessantes para o controle dos conhecimentos explícitos não padronizados; as de modelagem de empresas para a criação de mapas, as de trabalho colaborativo (*workgroup computing*) para suportar o contato com as pessoas que detêm o conhecimento tácito relacionado ao conhecimento explícito e as de sistemas especialistas para armazenar regras, assim por diante. Cada uma das diferentes categorias de ferramentas comerciais enfatiza e permite capturar conhecimentos explícitos em um conjunto específico de formas de representação. Como consequência, cada uma delas trabalha com uma parcela restrita do universo de conhecimentos explícitos da organização. A primeira grande implicação, para pesquisadores e profissionais, é a necessidade de que as soluções de gerenciamento do conhecimento explícito permitam o armazenamento de conhecimentos em diferentes formas de representação e permitam que o usuário estabeleça relacionamentos entre eles. Isto serve de alerta para os profissionais envolvidos com o desenvolvimento de soluções nesta área, pois, torna-se evidente que, na grande maioria dos casos, uma única ferramenta comercial não será suficiente para suportar um processo eficiente de gestão do conhecimento. Deve-se buscar o melhor aproveitamento das ferramentas em uso na empresa, e complementá-las por meio de integrações com novas ferramentas ou desenvolvimentos próprios. Ao final, obter-se soluções híbridas que possam armazenar, de maneira eficiente e integrada, conhecimentos explícitos de diferentes formatos, ampliando a parcela de conhecimento explícito gerenciado pela organização.

Outro aspecto é que estas ferramentas (comerciais ou não) não são apresentadas e contextualizadas dentro de um modelo conceitual teórico sobre gestão do conhecimento. Exceção principal deve ser feita para a proposta de THANNHUBER, TSENG & BULLINGER (2000). Corrobora-se, assim, a afirmação de RUBENSTEIN & MONTANO (2001) de haver necessidade de se localizar o papel das soluções de TI, dentro de um modelo conceitual para a gestão do conhecimento. Este fato é muito significativo, pois, faz com que tais ferramentas se

limitem a uma visão restrita de gestão do conhecimento, baseada no conceito de repositório, com um enfoque, primordialmente, na sua dimensão explícita; sem considerar o relacionamento que tal dimensão deve ter em relação à parcela tácita do conhecimento e, conseqüentemente, o estímulo ao ciclo de transformação.

A multiplicidade de tecnologias e classes de ferramenta, aliada à falta de localização dentro de um modelo conceitual de gestão do conhecimento, dificulta e, até mesmo impede, a realização de iniciativas efetivas na área de gestão do conhecimento. Nestas condições, é difícil identificar em que ferramentas e tecnologias investir e tomar a decisão sobre como personalizá-las, de tal modo que, realmente, contribuam com o esforço de gestão do conhecimento geral dentro da organização.

Um outro aspecto evidente na análise realizada é a pouca ênfase nos aspectos de validação dos conhecimentos armazenados. A inexistência de uma sistemática de validação expõe o sistema ao risco de se tornar obsoleto em um curto espaço de tempo, por acúmulo de conhecimentos irrelevantes para a maioria dos usuários. Conhecimentos irrelevantes dificultam a busca e, se incorretos, geram insegurança quanto à legitimidade dos demais conhecimentos armazenados no sistema. Ao mesmo tempo, caso seja empregada uma sistemática de validação muito complexa (“burocratizada”) para eliminar este problema, pode-se desestimular as pessoas a registrar seus conhecimentos explícitos. SILVA & ROZENFELD (1998), relatam uma experiência interessante de uma empresa multinacional do ramo de autopeças. Esta empresa implantou um sofisticado sistema de registro de conhecimento explícito no desenvolvimento de produto que (denominados de Best/Bad Practice) estava sendo muito pouco utilizado. Mostrou-se que a causa principal da não utilização era a dificuldade de se propor e validar um novo conhecimento. O usuário precisava reunir uma grande quantidade de documentos para submeter a proposta de nova Best/Bad Practice. Essa proposta era analisada por uma comissão internacional e, se aprovada, transformava-se em um registro no sistema. Conseqüentemente, muitas pessoas dentro da organização, possuidoras de conhecimentos explícitos, há muito por elas utilizados, não o submetiam à apreciação, tanto pelo tempo necessário para documentar a proposta como pela probabilidade de ver sua contribuição negada.

Há, ainda, a questão da facilidade de uso, um ponto importante para incentivar as pessoas a compartilharem seus conhecimentos explícitos. As atividades de recuperação e armazenamento de conhecimentos são sempre marginais às atribuições principais dos diversos profissionais. Portanto, caso tome um tempo elevado, dificilmente esses indivíduos deverão sacrificar o objetivo principal de tomar uma decisão ou executar uma tarefa, em favor de atividades deste tipo. Apesar de toda a revolução da tecnologia de informação com a internet e sistemas colaborativos, ainda é possível notar que esses sistemas precisam de *interfaces* mais aprimoradas.

Considerando estas barreiras apresentadas em conjunto, pode-se deduzir que a questão fundamental, na criação de soluções comerciais úteis para a gestão do conhecimento, é a própria inexistência de uma abordagem ou modelo conceitual, que incorpore uma definição clara do papel da tecnologia da informação, dentro desse esforço. Isso fica evidente na análise realizada, reforçando o trabalho de RUBENSTEIN-MONTANO (2001). Sem este modelo fica difícil especificar como se deve integrar todos estes diferentes tipos de conhecimentos explícitos, de forma que seja útil e auxilie na gestão do conhecimento.

Pode-se, portanto, resumir as constatações principais sobre as soluções para a gestão do conhecimento nos seguintes pontos:

- cada classe de ferramenta enfatiza um conjunto específico de formas de representação, trabalhando com uma parcela dos conhecimentos explícitos.
- falta um modelo teórico/abordagem (*framework*) para a gestão do conhecimento que oriente a construção de sistemas para este fim. Em especial, um modelo que incorpore aspectos descritivos (contendo um referencial para a Gestão do Conhecimento) e prescritivos (mostrando como se deve organizar estes conhecimentos e o papel das ferramentas, indo além de modelos atuais baseados em um conjunto de atividades ligados ao armazenamento e recuperação);
- falta de sistemáticas para a validação do conhecimento explícito armazenado;

- essas soluções devem ser criadas de maneira aberta, permitindo que as empresas adaptem-nas às suas necessidades e também utilizem-nas de forma combinada com sistemas legados, permitindo também que aproveitem, ao máximo, as tecnologias e ferramentas comerciais disponíveis no mercado, que, conforme pode ser observado nesta análise, são em quantidade cada vez maior.

Conclui-se a análise abordando as tendências em relação às ferramentas de gestão do conhecimento. A primeira a ser sentida, de maneira clara e forte, é a da adoção da plataforma *web* como sendo o ambiente ideal de desenvolvimento de tais classes de sistemas. Suas vantagens de acesso, simplicidade e armazenamento de diferentes mídias fazem com que seja difícil imaginar uma solução para gestão do conhecimento, sem essa plataforma. O cenário mais provável é o de um usuário acessando um portal de conhecimento, uma *intranet* ou um *site* específico do projeto em que ele trabalha e “navegando”, por meio de um único *front-end* WEB, nos conhecimentos explícitos que podem estar sendo acessados de diferentes origens (podendo cada qual estar baseada em uma das diferentes tecnologias acima mencionadas). Por exemplo, pode-se imaginar um usuário acessando um site na web (que ele reconhece como único) e realizando buscas em diferentes Bases de Dados, obtendo informações, realizando transações e treinamentos nos Sistemas ERP's, recorrendo a agentes inteligentes para atividades e decisões mais sofisticadas, verificando o cronograma do projeto e mesmo obtendo novos conhecimentos gerados por um *Datamining*; tudo isto integrado com ferramentas de trabalho colaborativo, como seu editor de email e editor de texto.

Resta saber como será, dentro do cenário, a posição dos fornecedores de sistemas. Será que eles deverão se especializar em ferramentas completas para diferentes ramos de negócio (indústria farmacêutica, indústria automobilística, desenvolvimento de produto, manufatura, etc...) ou em ferramentas com funcionalidades específicas conforme as diferentes formas de representação ? Serão os dados centralizados ou não ? Os sistemas *peer-to-peer* serão empregados na prática ? A tecnologia XML será um padrão de fato para a integração desses sistemas ? Será possível integrar as ferramentas de gerenciamento de projetos e de

trabalho em grupo eficientemente ? Essas e outras questões relacionadas parecem distantes de uma resposta conclusiva e muitas transformações ainda devem ocorrer nessa área. A revisão realizada não permite estabelecer uma tendência nesse aspecto, havendo infinitos cenários possíveis, de acordo com o desenvolvimento e aplicação das tecnologias e ferramentas aqui apresentadas.

3. Método

Este capítulo descreve o método empregado. Inicia com a apresentação detalhada do problema de pesquisa, item 3.1, e a discussão sobre sua relevância, item 3.2. Em seguida, são apresentadas as hipóteses assumidas e objetivos a serem atingidos, respectivamente, nos itens 3.3 e 3.4. O item 3.5 apresenta, detalhadamente, os aspectos referentes à escolha do método e instrumentos de pesquisa e o item 3.6 apresenta as etapas do trabalho.

3.1 Problema de Pesquisa

Este trabalho foi motivado e desenvolvido sobre um problema prático de pesquisa, identificado durante o desenvolvimento de um modelo de referência para o processo de desenvolvimento de produto, considerando a participação de fornecedores. Para apresentar, de forma clara e inequívoca, sua importância e fundamentação é necessário explorar mais detalhadamente o problema prático que o motivou e o problema de pesquisa que foi tema de estudo. Por isso a definição do problema, hipóteses e objetivo são descritos dentro da metodologia.

O objetivo inicial do projeto era desenvolver um modelo de referência para o processo de desenvolvimento de produto, considerando a participação de fornecedores nesse processo. Durante esse esforço, percebeu-se que o modelo obtido ficava pouco claro quando o leitor ou usuário não tinha acesso aos conhecimentos que foram utilizados para suportar as inúmeras decisões tomadas durante a elaboração desse modelo. Os conhecimentos evidenciam a lógica empregada nas decisões e fornecem informações sobre os aspectos do processo que receberam maior ou menor atenção, permitindo ao usuário uma análise mais crítica da recomendação exposta no modelo. Levantou-se a hipótese de que os conhecimentos explícitos ou tácitos empregados pelos responsáveis pela modelagem deveriam ser parte integrante do modelo de empresa.

Decidiu-se, então, buscar soluções para essa questão na literatura sobre gestão do conhecimento e modelagem de empresa. Foram avaliadas várias ferramentas e tecnologias de informática, desde linguagens de programações básicas até

ferramentas sofisticadas que incluem suporte para trabalho em grupo e armazenamento de dados distribuídos. Conforme se avançava nesta análise, ficou evidente os problemas apontados na revisão bibliográfica.

Primeiramente, sentiu-se falta de uma integração entre os conhecimentos explícitos registrados em diferentes formatos, característica importante para contextualizar o conhecimento armazenado, em especial, dos relacionamentos entre modelos de referência e os demais conhecimentos explícitos. Pode-se considerar os modelos de referência, como um dos formatos de conhecimentos explícitos sobre o processo, cujo diferencial é registrar a visão holística e integrada deste. Ao consultar o modelo de referência durante a realização de um projeto, deveria ser possível explorar, diretamente do modelo, os conhecimentos registrados nos demais formatos (livros, testes, formulários, opiniões, casos, etc...).

As ferramentas encontradas na maioria dos casos, não deixavam claro o seu papel dentro do esforço maior de gestão do conhecimento; principalmente, descreviam recursos e funcionalidades sem mostrar a forma de empregá-las, seus potenciais e limitações quando inseridas em um programa de gerenciamento de conhecimentos em uma organização.

Diante desses problemas, tornou-se evidente que a dificuldade em obter uma ferramenta que registrasse os conhecimentos explícitos durante a criação dos modelos de referência tinha origem em um problema fundamental, que pareceu bastante relevante. Faltavam conceitos teóricos e discussões que permitissem relacionar os recursos disponíveis nessas ferramentas com o efetivo esforço de gestão do conhecimento e aprendizagem organizacional em si, isto é, havia um desnível entre pesquisas e teorias que lidavam com ferramentas computacionais e outras que lidavam com as ações ligadas à gestão do conhecimento, sem que houvesse conceitos e modelos ligando essas duas áreas. Era este problema, a falta dessa “ponte”, que dificultava a escolha e a adoção de ferramentas para a gestão do conhecimento no PDP.

Portanto faltava conhecimento suficiente que guiasse os profissionais sobre como empregar as tecnologias e funcionalidades das ferramentas comerciais existentes na construção de soluções para o registro de conhecimentos explícitos. Como criar soluções capazes de proporcionar a melhoria da gestão do conhecimento

e, conseqüentemente, melhorias nos processos de negócio. Segundo BOOTH, COLOMB & WILLIANS (1995), esse poderia ser caracterizado como um problema prático, que pode ser formulado com a seguinte questão:

como criar, a partir de ferramentas e tecnologias de informática disponíveis, uma solução para gerenciamento (registro, recuperação, manutenção e validação) de conhecimentos explícitos, de maneira integrada ao modelo de processo e partindo-se de tecnologias e ferramentas existentes no mercado?

Essa foi definida como a questão de pesquisa do trabalho, ou seja, aquela para cuja resposta se pretende contribuir. Esta questão pode ser transformada no Problema de Pesquisa:

proposição de um modelo conceitual que auxilie a criação de soluções para o gerenciamento (registro, recuperação, manutenção e validação) de conhecimentos explícitos sobre o processo de desenvolvimento de produto, de maneira integrada ao modelo de processo e partindo-se de tecnologias e ferramentas existentes.

Este é um legítimo problema de pesquisa na acepção de BOOTH, COLOMB & WILLIANS (1995) pois se caracteriza por demonstrar uma falha ou lacuna no conhecimento e por levar, como conseqüência direta desta lacuna, a uma falha na compreensão de um assunto. A lacuna é, justamente, o potencial da aplicação de ferramentas de Gestão do Conhecimento na solução do problema de registro e apresentação do conhecimento explícito sobre o processo de desenvolvimento de produto. A existência desta lacuna impede a construção, de forma eficiente e eficaz, de ferramentas que auxiliem o registro e a análise do conhecimento explícito sobre o processo de desenvolvimento de produto, que poderiam ser extremamente úteis para a melhoria deste processo.

3.2 Justificativa

A justificativa do trabalho está, em parte, apresentada no item acima, quando se discorreu sobre o problema de pesquisa. Conforme apresentado, ele contribui com conhecimentos e um modelo que permitem a construção de soluções para gerenciar os conhecimentos explícitos no processo de desenvolvimento de produto.

Mas, além disso, espera-se que possa também ser utilizado para criar soluções que auxiliem o compartilhamento de conhecimentos entre pesquisadores e especialistas que atuam neste processo. Dada sua natureza ampla e multidisciplinar, o PDP é constituído por pesquisadores com formação em diferentes áreas de conhecimento, o que, muitas vezes, dificulta a integração desses conhecimentos em um corpo único e integrado. Esse trabalho pode contribuir para o desenvolvimento de soluções que venham facilitar o trabalho multidisciplinar e a troca conhecimentos entre estes pesquisadores e os profissionais de empresa que atuam nesta área.

Outro aspecto importante é que os resultados dessa pesquisa podem beneficiar outras áreas tão interdisciplinares como o processo de desenvolvimento de produto, na medida que o conhecimento gerado poderá ser, em grande parte, aplicado na gestão do conhecimento de outros processos de negócio.

De maneira imediata, este trabalho contribui fortemente para a pesquisa do Grupo de Engenharia Integrada do Núcleo de Manufatura Avançada, do qual o pesquisador faz parte.

3.3 Hipóteses

Do problema de pesquisa exposto no item 3.1, foram derivadas duas hipóteses que servirão de base para o trabalho. A primeira, hipótese H1, é formulada como a seguir:

H1: é viável a criação de um modelo (arquitetura, isto é um conjunto de elementos) que oriente o desenvolvimento de soluções para o gerenciamento de conhecimentos explícitos sobre o PDP, independentemente da tecnologia;

Essa hipótese poderia ser classificada como existencial segundo McGUIGAN(1975), ou seja, tem uma abrangência geral e responde pela garantia da existência de uma solução. Esse modelo seria independente da tecnologia de informação e poderia ser utilizado como base para o desenvolvimento de ferramentas específicas, cada qual utilizando uma ou um conjunto de soluções tecnológicas comerciais.

Por fim, há uma hipótese ligada diretamente ao conteúdo ou estrutura do modelo. Assume-se que:

H2: uma solução que integre os conhecimentos explícitos de diferentes tipos e formatos com os modelos de referência melhoraria a aprendizagem organizacional;

H 2.1: a visualização dos conhecimentos explícitos, a partir do modelo do processo facilita o acesso a esses conhecimentos e sua compreensão;

H 2.2: necessita-se menos esforço para gerar e aprimorar os modelos de referência, baseando-se na arquitetura, do que sem seu apoio;

H 2.3: o modelo gerado é melhor compreendido pois o usuário tem acesso a parte do conhecimento que serviu de base na geração do modelo;

H 2.4: a aprendizagem organizacional seria beneficiada pelo aumento do compartilhamento dos conhecimentos e a melhoria contínua dos modelos de processo de negócio.

Neste trabalho será feita uma primeira verificação das hipóteses H1 e H2. Diz-se primeira verificação, por se tratarem de hipóteses abrangentes e, portanto, cuja avaliação mais “madura” e “confiável” demandará anos de pesquisa. Este trabalho pretende ser um início sólido dessa verificação.

3.4 Objetivos

O objetivo principal deste trabalho pode ser descrito como:

1. criar uma arquitetura para orientar o desenvolvimento de soluções para o gerenciamento de conhecimentos explícitos no PDP de maneira integrada, isto é, na qual conhecimentos explícitos, em diferentes formatos, estão relacionados e podem ser acessados diretamente do modelo do processo.

Esse é o objetivo central do trabalho que, uma vez atingido, implicaria na demonstração de um importante caminho a ser seguido nas pesquisas sobre ferramentas para gestão do conhecimento. A aposta deste trabalho é que, mais do que pesquisar novas ferramentas, levando em consideração apenas a tecnologia de informação, seria impossível compreender melhor quais as funcionalidades

existentes, quais são as mais importantes para a gestão dos conhecimentos explícitos e como elas poderiam ser reunidas, de forma a auxiliar, efetivamente, esse gerenciamento. Responder essas questões é fundamental para os profissionais que enfrentam o problema de avaliar, escolher e implementar sistemas de informação no PDP.

Para atingir este objetivo, optou-se por realizar uma pesquisa profunda. Adotou-se o caminho de propor uma arquitetura, empregá-la como base para o desenvolvimento de uma solução específica e, por fim, avaliá-la a partir da aplicação dessa solução em um estudo de caso.

A vantagem de se optar por esse caminho é a de ter, como subproduto, uma solução específica, que pode ser, futuramente, aprimorada e utilizada em investigações cada vez mais profundas, que venham demonstrar a real potencialidade do modelo conceitual (arquitetura) desenvolvido. Neste trabalho espera-se atingir o objetivo mais modesto, mas não menos complicado, de realizar uma primeira verificação da viabilidade da criação de uma arquitetura desse tipo, dando forma a essa linha de pesquisa.

Uma vez definido este direcionamento, foram desenvolvidos os três objetivos secundários listados nos itens O1.1, O1.2 e O1.3:

O1.1 propor uma arquitetura que oriente o desenvolvimento de soluções de gerenciamento do conhecimento explícito sobre o PDP;

O1.2 criar uma solução para gerenciamento de conhecimentos explícitos sobre o PDP, baseada na arquitetura;

O1.3 aplicar a solução no caso exemplo;

O1.3.1 escolher e implantar a solução no caso;

O1.3.2 analisar os resultados da solução, quanto ao gerenciamento de conhecimentos explícitos;

O1.3.2 analisar a aplicação da solução, quanto ao desenvolvimento de modelos de referência.

3.5 Método

Para a realização da pesquisa foram definidas 4 grandes etapas, guiando-se por um mesmo referencial teórico, abordagem de pesquisa e método de pesquisa, conforme apresentado na figura 3.1. Para cada uma destas etapas, foram definidos instrumentos a ser aplicados. O conhecimento gerado é, portanto, resultante da soma das análises e informações coletadas em cada uma dessas etapas, que se complementam e, em determinados pontos, se sobrepõem, permitindo uma avaliação mais precisa do objeto de estudo. Cada um destes elementos é apresentado, a seguir, junto com a justificativa de suas escolhas, implicações e vantagens e limitações de sua adoção.

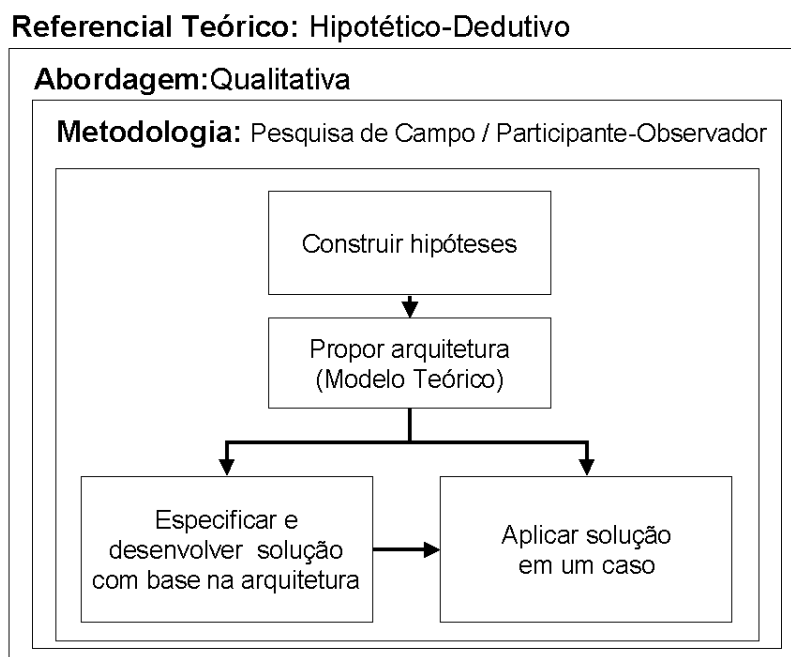


Figura 3. 1 Referencial Teórico e Métodos

3.5.1 Referencial Teórico

Denomina-se referencial teórico à categoria de métodos de pesquisa mais amplos, ou seja, aqueles diretamente relacionados com as correntes que estudam métodos de pesquisa dentro da área de epistemologia (conforme PÁDUA, 1996). Nas pesquisas, em áreas que abordam problemas sociais e de organizações, como a deste trabalho, é fundamental que se identifique e avalie, dentro do contexto da metodologia, o referencial teórico que é assumido. Conforme o referencial adotado, diferentes concepções da realidade, da personalidade e da interação homem-mundo são assumidos, os quais, por sua vez, terão implicações diretas nas análises e resultados obtidos. Diferentes referenciais limitam ou expandem a compreensão do mundo, segundo um conjunto específico de aspectos (PADUA, 1996).

Este trabalho pode ser enquadrado dentro do referencial teórico hipotético-dedutivo, no qual se assume uma postura positivista, alterando-se, porém, o critério de verificabilidade pelo de “falseabilidade” (MARTINS, 1994), isto é, formulam-se hipóteses, com base em teorias e procuram-se *refutações* para estas hipóteses. Enquanto não refutadas (“falseadas”) as hipóteses continuam válidas, garantindo, conseqüentemente, a validade da teoria. Quando algum pesquisador refuta a hipótese, deve-se sair à procura de uma nova teoria, formulando-se novas hipóteses e testando-

as (PÁDUA, 1996). Conforme a visão de mundo positivista desse referencial, a realidade é tida como objetiva e formada por fatos. O papel do pesquisador é identificar esses fatos, montando teorias capazes de controlá-los e predizê-los.

A escolha desse referencial para o trabalho foi motivada por ser sua prova mais robusta que os métodos indutivos e dedutivos (conforme a crítica feita por Karl Popper no início deste século) e, além disso, por serem as variáveis envolvidas na pesquisa suficientemente concretas e circunscritas a um momento específico, para dispensar métodos que considerassem a interpretação do pesquisador e o processo histórico, tal qual os métodos fenomenológicos e dialético o permitem (veja PÁDUA, 1996 e MARTINS, 1994).

3.5.2 Abordagem da Pesquisa

Segundo CERVO & BERVIAN(1983) e BRYMAN(1989), um trabalho científico pode ser classificado, segundo a abordagem, entre Pesquisa Qualitativa e Pesquisa Quantitativa. A pesquisa quantitativa é aquela em que o pesquisador tem a oportunidade de definir, claramente, as variáveis e de medi-las; na abordagem qualitativa, a ênfase deixa de ser a codificação e medição das variáveis e procura-se captar interpretações e discursos e estruturá-los de forma a construir o conhecimento (BRYMAN, 1989). Nesse último tipo, as variáveis não são totalmente quantificáveis. Estas abordagens também não são excludentes podendo coexistir conjuntamente em uma mesma pesquisa.

Como o objeto desta pesquisa é bastante novo e as variáveis não são bem definidas, a abordagem predominante escolhida é a da pesquisa qualitativa. A análise será baseada, principalmente, nas discussões e análises do processo de definição da arquitetura (modelo conceitual), passando pela sua implantação na forma de uma solução e na avaliação de sua aplicação. É importante manter a predominância dessa abordagem, mesmo na avaliação da aplicação (uma etapa que seria teoricamente mais quantificável), porque, dado o grau de inovação e por ser a primeira aplicação do modelo (arquitetura) proposto, esse método permitiria captar uma riqueza maior de detalhes, possíveis variáveis e problemas não previstos.

3.5.3 Método de Pesquisa

A realização do trabalho foi dividida em 4 etapas que, juntas, envolvem a criação de uma arquitetura, seu uso para a construção de uma ferramenta de gestão de conhecimentos explícitos específica para um determinado caso e a avaliação dessa ferramenta quanto à sua aplicação neste caso. Essa opção reflete a adoção de um método que poderia, segundo a classificação de DANE(1990), ser identificado como o de pesquisa de campo do tipo participante-observador (*participant-as-observer*).

Segundo esta autora (DANE,1990, p.147), pesquisa de campo é um rótulo que pode ser atribuído aos métodos de pesquisa que envolvem a observação direta dos eventos, conforme sua ocorrência natural. No tipo de pesquisa participante-observador mencionado, o pesquisador influencia e atua diretamente nas ações do fenômeno e todos os participantes têm conhecimento de que se trata de um pesquisador. A esse mesmo enfoque, há autores que empregariam o rótulo de pesquisa-ação.

Um forte motivo para a escolha desse método que, desde o início mostrou-se mais promissor, foi a não existência de ferramentas ou modelos conceituais anteriores ao proposto, em conformidade com as características explicitadas, impedindo, portanto, que o estudo fosse baseado em análises de casos ou outra técnica de observação direta.

Ao mesmo tempo, outros métodos, como optar por uma pesquisa teórica ou *surveys* e casos com análise profunda sobre os temas Modelagem de Empresa e Gestão do Conhecimento, não pareciam suficientes para gerar os conhecimentos capazes de auxiliar a criação desta ferramenta especial de registro da gestão do conhecimento. Isso porque, partindo-se desses métodos, dificilmente seria possível observar todos os problemas de ordem conceitual e tecnológica que o enfrentamento do problema pelo próprio pesquisador, conforme o método adotado, pode proporcionar.

Porém não se deve desprezar as limitações do método escolhido. A maior delas é o fato de não permitir que sejam facilmente tecidas relações causais entre variáveis (DANE, 1990). Outra limitação importante é quanto aos resultados gerados que apresentam problemas de validade externa, isto é, os resultados encontrados nesses tipos de pesquisa, na grande maioria das vezes, são válidos somente para o

caso estudado e não há como reproduzir o experimento. Em decorrência desses problemas, as discussões não poderão “esgotar” o tema, isto é, refutar totalmente as hipóteses apresentadas, pois estarão sendo discutidas formas particulares de arquitetura e aplicação desenvolvidas por indivíduos (o pesquisador e seu grupo de pesquisa) possuidores de uma bagagem e experiências únicas, não reprodutíveis e que deverão, devido a essa formação básica, focar, mais ou menos, determinados aspectos do problema durante a realização da proposta e sua avaliação.

Essas restrições, se analisadas, estão condizentes com os objetivos propostos no trabalho, o de uma primeira verificação das hipóteses de existência, viabilidade e funcionamento do modelo. O método é importante, também, neste momento, porque servirá como um passo inicial para que, no futuro, possam existir uma proposta de arquitetura e ferramentas a ser exploradas por métodos de pesquisa mais objetivos e generalizáveis e, assim verificar, de maneira mais definitiva as hipóteses aqui apresentadas, de forma que possam ser completamente validadas, ou não, futuramente.

Os cuidados a ser tomados são os mesmos para todas as etapas da pesquisa: caracterização clara dos papéis de pesquisador e membro do grupo de pesquisa, de forma a realizar anotações independentes, quando se assume cada um destes papéis, controle das anotações e registros tomados e complementação das notas do pesquisador, com entrevistas e discussões com outros membros do grupo de Engenharia Integrada, que estejam participando dos processos (de pesquisa e elaboração da arquitetura) descritos nas etapas anteriores.

3.6 Etapas do trabalho

Cada uma das etapas para a realização do trabalho é descrita, a seguir, em maiores detalhes. Ao final, apresenta-se o quadro 3.1 e a figura 3.2, que as sintetizam.

Etapa 1: Construção das hipóteses

Essa etapa não pode ser considerada, propriamente, uma etapa de pesquisa, pois, segundo MARTINS (1994) e PÁDUA (1996), no método hipotético-dedutivo não importam, para sua validade, as origens das hipóteses e nem mesmo o processo que levou a sua construção. Mesmo assim, na análise e dedução das hipóteses desta

pesquisa, foram tomados dois principais cuidados para garantir hipóteses adequadas e robustas. Primeiramente, as hipóteses foram classificadas segundo McGUIGAN(1975) como universais e existenciais. Esta classificação permitiu entender melhor a natureza de cada uma delas e levou, por meio deste questionamento, ao seu refinamento até tornarem-se claras e com o mínimo possível de ambigüidade. Outro cuidado foi o questionamento segundo os sete critérios que, segundo McGUIGAN(1975), devem ser satisfeitos: a) deve ser comparável; b) deve estar em harmonia com outras hipóteses de investigação; c) deve seguir o princípio da parcimônia; d) deve responder ao problema; e) deve ter simplicidade lógica; f) deve ser expressa da forma mais quantificável possível; g) deve ter o maior número possível de conseqüências.

Etapa 2: Proposição da Arquitetura para Gerenciamento de Conhecimentos Explícitos sobre o PDP

Nessa etapa, pretende-se responder ao objetivo 1.1, de propor a arquitetura. Nela o pesquisador desenvolveu a Arquitetura para Gerenciamento de Conhecimentos Explícitos sobre o PDP, sendo possível registrar todas as dificuldades e aspectos interessantes deste processo de criação. Esses elementos formam uma primeira verificação de viabilidade da proposta, na medida que emerge, dessa experiência de desenvolvimento, uma primeira apreciação sobre as dificuldades e reais possibilidades de criação de um modelo como o proposto na arquitetura.

Os resultados desta etapa estão relacionados às hipóteses H1, de que é importante, necessário e possível criar um modelo que auxilie o desenvolvimento de ferramentas de gestão de conhecimentos explícitos para o PDP.

Etapa 3: Desenvolvimento de uma solução para gerenciamento de conhecimentos explícitos no PDP baseada na arquitetura

Essa etapa visa atingir o objetivo 1.2, criando uma solução baseada na arquitetura. Para tanto, definiu-se o grupo de engenharia integrada como caso, mais especificamente, a atuação deste grupo num estudo sobre o relacionamento entre clientes e fornecedores no processo de desenvolvimento de produto. Os instrumentos

de pesquisa aplicados são notas de campo e análise de documentos, uma vez que as únicas fontes de informação são os pesquisadores envolvidos e os documentos gerados durante o trabalho de desenvolvimento.

O desenvolvimento da solução foi realizado em cinco grandes atividades, envolvendo: um desenho geral da solução; a especificação e desenvolvimento de uma intranet; a “customização” da ferramenta de modelagem ARIS Toolset; a preparação de modelos de documentos e diretórios na rede e a análise dos resultados.

Os resultados dessa etapa permitiram uma segunda análise da hipótese H1, relacionada com a possibilidade de criar a arquitetura. Enquanto na etapa anterior os resultados avaliam a obtenção da arquitetura em si, nesta etapa complementa-se a demonstração verificando a capacidade de empregá-la na construção de soluções. Dois aspectos foram verificados: a viabilidade técnica (existência de tecnologia de informática suficiente para a implementação de uma solução compatível com todas as especificações da arquitetura) e a viabilidade prática (as soluções técnicas existentes serem compatíveis com a complexidade de um sistema de informação trivial ou tradicional, tais como sistemas de automação comercial). Caso a arquitetura não permitisse a construção de uma solução a hipótese H1 não seria verificada, mesmo que, anteriormente, fosse encontrada uma arquitetura qualquer.

Essa etapa foi útil, também, como uma primeira verificação da hipótese H2, pois o desenvolvimento da solução permitiu ao pesquisador realizar testes com o sistema e obter uma primeira experiência sobre a utilidade da solução, isto é, se a visualização dos conhecimentos explícitos com modelo de referência traz reais benefícios para a GC.

Etapa 4: Aplicação da Solução Desenvolvida no Caso Escolhido

O objetivo dessa etapa é aplicar a ferramenta ao caso escolhido e analisar os resultados encontrados. Analisou-se a utilização da solução, tanto para o registro de conhecimentos explícitos, como, também, na geração de modelos de referência. Verificou-se os requisitos, em termos de desempenho, vantagens e desvantagens de sua utilização e principais problemas, atingindo, portanto, o objetivo 1.3 e avaliando principalmente a hipótese H2, relacionada com as características da arquitetura. Nesse esforço, será possível realizar uma nova avaliação da hipótese H1, pois

permitiu a análise da viabilidade da ferramenta, frente, agora, à experiência dos usuários, participantes do estudo de caso.

As atividades principais para a aplicação da solução podem ser vistas na figura 3.2.

Quadro 3.1 Síntese das principais etapas do trabalho

	Etapas do Projeto de Pesquisa			
	Propor Arquitetura		Aplicar Arquitetura	
	Etapa 1	Etapa 2	Etapa 3	Etapa 4
Descrição	Gerar Hipóteses	Proposição da Arquitetura	Desenvolvimento de uma solução baseada na arquitetura	Aplicação da solução
Objetivos do Trabalho para o qual a etapa contribui	**	O 1.1	O 1.2	O 1.3
Hipóteses que auxilia a verificar	**	H1	H1 H2 (Parcial)	H2
Método	Análise Dedução*	•Pesquisa de Campo (participante-observador)		
Motivo da Escolha	**	<ul style="list-style-type: none"> •Permite vivenciar” todos os problemas e dificuldades ; •Identificar variáveis e problemas não perceptíveis •Permite avaliar detalhes e aspectos qualitativos e inesperados sobre o desempenho da ferramenta 		
Limitações	**	<ul style="list-style-type: none"> •Dificuldade de tecer relações; •Baixa validade externa •Influência do Pesquisador 		
Ferramentas	**	<ul style="list-style-type: none"> •Notas de pesquisa •Documentos 	<ul style="list-style-type: none"> •Notas de pesquisa •Roteiros de Entrevista 	
Cuidados Adotados	Verificação lógica das hipóteses segundo critérios da literatura	<ul style="list-style-type: none"> •Distinção de Papéis; •Comparar dados de diferentes fontes 		

*Dentro da literatura sobre metodologia de pesquisa consultada, a geração das hipóteses não seriam métodos, pois, considera-se, dentro do método hipotético-dedutivo, que não importa como a hipótese é gerada. A inclusão deste item na tabela é para demonstrar os cuidados que foram tomados com a definição da hipótese.

**Não se aplica, pois é a única maneira possível.

4. Descrição da arquitetura para gerenciamento de conhecimentos explícitos sobre o processo de desenvolvimento de produto

Nesse item, descreve-se a arquitetura proposta para gerenciamento de conhecimentos explícitos sobre o processo de desenvolvimento de produto. Inicia-se com a apresentação do princípio básico e dos requisitos que pautaram a proposição da arquitetura, item 4.1, e a sua descrição geral, item 4.2. Nos itens 4.3, 4.4 e 4.5, são apresentadas, com maiores detalhes, as três classes principais de objetos da arquitetura, respectivamente: repositório, funcionalidades e elementos de apoio. No item 4.6 descrevem-se as considerações sobre como fazer uso dessa arquitetura, isto é, como aplicá-la, de forma a desenvolver sistemas para gerenciamento de conhecimentos explícitos sobre o processo de desenvolvimento de produto.

4.1 Definição, Princípio Básico e Requisitos para a Construção da Arquitetura

A arquitetura descrita nesse item é formada por um conjunto de elementos que auxiliam o desenvolvimento de sistemas para o gerenciamento do conhecimento no processo de desenvolvimento de produto. Trata-se, portanto, de um modelo conceitual contendo descrições de funcionalidades, métodos, conceitos, tipologias e outros elementos, os quais orientam o desenvolvimento destes tipos de sistemas, independentemente das tecnologias de informática empregadas.

Antes de apresentar os elementos da arquitetura, é preciso definir o conceito de gestão de conhecimento empregado. No item revisão bibliográfica, ela foi definida como o conjunto de atividades dedicadas a garantir e incentivar a criação, registro e compartilhamento do conhecimento em uma organização, visando as características de uma organização que aprende.

Embora essa definição seja fortemente consensual, na área de pesquisa sobre gestão do conhecimento, do ponto de vista de construção de uma solução é excessivamente genérica, dando margem a uma ampla gama de caminhos a ser explorados. Este “conjunto de atividades” pode conter atividades ecléticas, como a

formação, treinamento, aprendizagem, incorporação de ferramentas e motivação. Em outro aspecto porque é preciso tornar claro o entendimento de “compartilhar” e “aprender”. Assim, parece imprescindível, quando se trata de propor metodologias e ferramentas nesta área, tornar explícito o referencial teórico empregado para se compreender a gestão do conhecimento.

Adotou-se como referencial neste trabalho o modelo de NONAKA & TAKEUCHI (1995) de ciclos de transformação do conhecimento, apresentado durante a revisão bibliográfica. Dentro desse referencial, intervir na área de gestão do conhecimento significa realizar ações para que o conhecimento da empresa esteja sendo continuamente criado e transformado, de tácito para explícito, e vice-versa, através da organização. Quanto mais intensa a realização do ciclo, maior a geração e o compartilhamento do conhecimento dentro da organização, gerando maiores oportunidades de melhorias nos processos de trabalho e produtos.

Nesse contexto, o papel das ferramentas de tecnologia de informação é apoiar as transformações nas diversas etapas do ciclo. Devem garantir a integração, por exemplo, permitindo que o conhecimento tácito, que é sistematizado e transformado em explícito em uma etapa do ciclo possa ser encontrado e servir de base para a combinação em outra etapa, seja ela realizada por um único ou por vários indivíduos da organização.

A escolha desse referencial foi motivada por parecer o mais prático em termos de “operacionalização” da gestão de conhecimento, pois define bem seu escopo de atuação e está em harmonia com o conceito de *learning organization*. Isso pode ser notado analisando-se a idéia fundamental, por trás do conceito de ciclos. Não é a ferramenta, o treinamento ou as pessoas em si que garantem isoladamente a efetiva gestão do conhecimento. Na verdade, é a existência deste ciclo que permitirá a geração e transmissão contínua do conhecimento, de pessoa para pessoa, dentro dessa organização. Por conseguinte, a efetiva gestão do conhecimento é o esforço de manutenção e “fortalecimento” do ciclo. Cabe às organizações criar as condições ambientais (recursos, valores, motivação, etc...), fornecer treinamento e capacitação, entre outras ações, de forma a criar um ambiente propício para que o ciclo seja cada vez mais “reforçado” dentro da organização, gerando o acúmulo de conhecimento na empresa e este se refletindo em melhorias reais nos processos de negócio e produtos

e, sobretudo, permitindo que esse conhecimento seja um fator de criação de competências internas na solução de problemas dentro da esfera de atuação da organização.

Com base neste referencial e nas barreiras e propostas de sistemas de gestão do conhecimento apresentados na revisão bibliográfica, foram definidos os seguintes requisitos para a arquitetura:

1. **Apoiar as etapas do ciclo de conversão:** a arquitetura deve registrar e gerenciar conhecimentos explícitos, de maneira a apoiar, o máximo possível, as quatro etapas do ciclo de conversão de conhecimento (socialização, externalização, combinação, internalização), fazendo-o da maneira mais eficiente e eficaz possível;
2. **Auxiliar a integração entre as etapas do ciclo de conversão:** deve permitir que os conhecimentos explícitos gerados ou utilizados por uma pessoa, em uma determinada etapa do ciclo de conversão, tornem-se disponíveis para as demais pessoas e/ou outros ciclos, difundindo o conhecimento e “potencializando” cada contribuição realizada à memória organizacional;
3. **Possuir um padrão para validação e sistematização dos conhecimentos explícitos armazenados:** conhecimentos explícitos pouco válidos podem trazer sérios problemas para os usuários, pois dificultam a busca dos demais conhecimentos e geram insegurança quanto à validade e utilidade do sistema. Portanto é fundamental a existência de um mecanismo para garantir a validade e a organização dos conhecimentos, evitando desperdícios e facilitando a busca;
4. **Apoiar a geração de modelos do processo:** os conhecimentos devem ser armazenados e dispostos de forma a permitir a geração de modelos de referência e modelos específicos para o processo de desenvolvimento de produto, empregando os conhecimentos explícitos armazenados;
5. **Gerenciar um conjunto de Modelos de Referência para diferentes tipos de processo de desenvolvimento de Produto:** a arquitetura deve permitir o armazenamento dos modelos de referência, de forma a facilitar

a geração de vários modelos de referência cada qual atendendo as particularidades dos diferentes projetos a serem desenvolvidos;

6. **Possuir metodologia de Modelagem e permitir a comparação de modelos:** embora a arquitetura deva ser capaz de gerenciar modelos de referência em diferentes formatos, é importante que prescreva uma metodologia de modelagem capaz de obter modelos, dentro de um padrão formal, pois a padronização dos modelos é importante para facilitar a comparação e o acesso dos usuários ao seu conteúdo;
7. **Manter uma integração entre modelos de processo e conhecimentos explícitos:** os modelos de processo são um tipo de conhecimento explícito que descreve o processo de negócio, sendo possível inserir, como parte dessa descrição, as competências ou conhecimentos exigidos para a condução das atividades de desenvolvimento de produto descritas no modelo. A arquitetura deve manter o relacionamento entre as atividades e conceitos expressos nos modelos de empresa e os conhecimentos explícitos armazenados, de forma a permitir ao usuário navegar diretamente do modelo para os conhecimentos explícitos e vice-versa. Permite-se, assim, que o conhecimento explícito possa ser acessado a partir do contexto em que é empregado dentro do processo de negócio;

4.2 Visão Geral da Arquitetura

A arquitetura é representada na figura 4.1. No lado esquerdo, representa-se, de maneira estilizada, o ciclo de conversão de conhecimentos de NONAKA & TAKEUCHI (1995), contendo suas quatro etapas: socialização, combinação, externalização e internalização. Reforça-se, assim, o conceito de que o objetivo maior das soluções baseadas na arquitetura é auxiliar nas etapas deste ciclo bem como facilitar a integração entre elas. O quadro à direita representa os elementos principais da arquitetura. Ela é formada por três grandes classes de elementos, que aparecem em cinza: o repositório de conhecimentos explícitos, as especificações das funcionalidades necessárias para a manutenção do repositório e um conjunto de elementos de apoio. Estes últimos são modelos que descrevem aspectos específicos e que devem ser utilizados como base na implementação das funcionalidades.

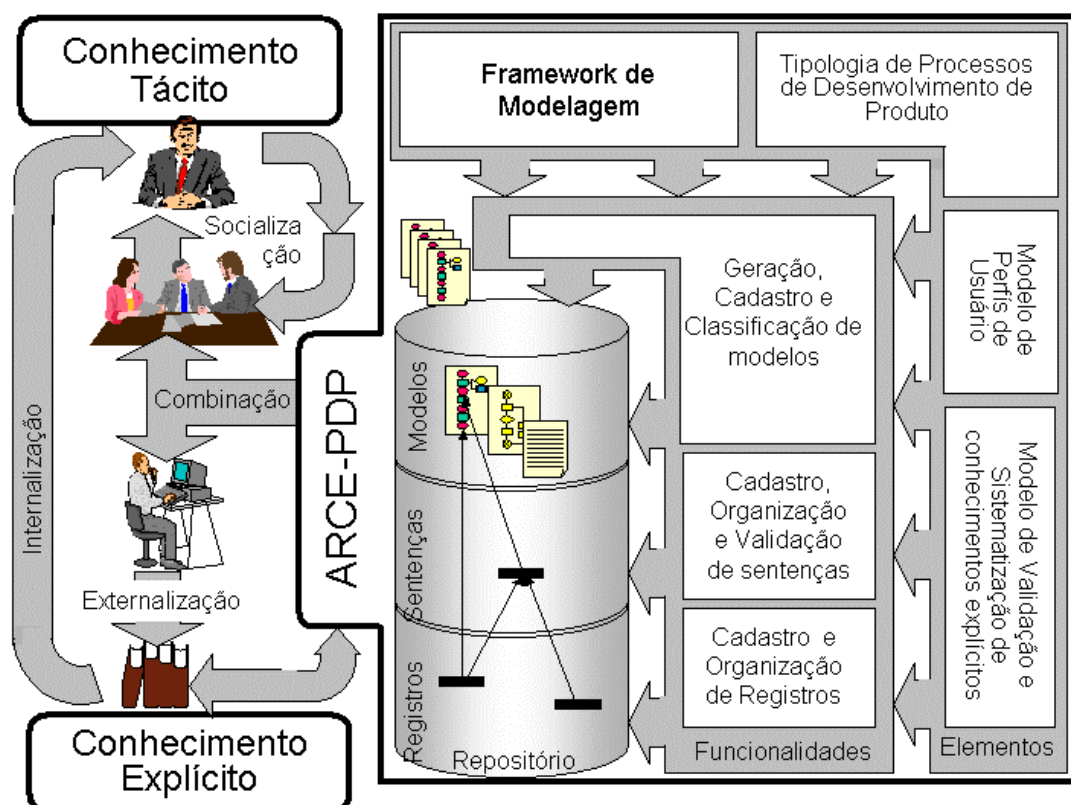


Figura 4. 1 Arquitetura para registro de conhecimentos explícitos sobre o processo de desenvolvimento de produto (ARCE-PDP)

O **repositório** armazena os conhecimentos em 3 **tipos básicos** denominados: registros, sentenças e modelos (ver figura 4.1). Os registros são informações e todo e qualquer documento gerado durante o processo de desenvolvimento de produto: atas de reunião, desenhos, procedimentos, casos, referências bibliográficas de livros e relatórios, entre tantos outros. Representam o equivalente à forma de representação dita não-estruturada, conforme descrito no item 2.3.4 do trabalho. As sentenças são um tipo específico de conhecimento dito estruturado, isto é, que possui um formato padrão. Cada sentença representa uma idéia específica, na forma de regras, contendo um sujeito e um verbo. As sentenças são criadas a partir dos registros, isto é, representam idéias que estão incorporadas em um ou mais deles. As sentenças são peças fundamentais dentro do conceito de validação e sistematização do conhecimento, como se verá mais adiante. Por fim, há os conhecimentos representados na camada de maior nível, os modelos do processo de desenvolvimento de produto. São representações do processo e, por isso, formam a

camada superior do repositório. Os modelos “trazem” em si uma visão sistêmica do processo, a qual auxilia na contextualização dos conhecimentos explícitos armazenados abaixo dele. Esse repositório permite ao usuário efetuar relacionamentos entre estes três tipos de conhecimento.

A segunda classe de elementos é formada pelas **funcionalidades** básicas que precisam ser implementadas, para a criação e manutenção do repositório. Tais funcionalidades abrangem desde o cadastro, armazenamento e atualização dos conhecimentos explícitos, como sua manipulação, incluindo a adição de comentários, comparação, validação, classificação e outros. Foi especificado um conjunto de funcionalidades para cada tipo de conhecimento explícito armazenado.

A arquitetura prevê um conjunto de 4 **elementos conceituais** que definem como implementar as funcionalidades acima especificadas. Os elementos apóiam as atividades de registro, classificação e organização dos conhecimentos explícitos. Um framework de modelagem, uma tipologia de processos de desenvolvimento de produto e um modelo para sistematização e classificação dos conhecimentos explícitos; são representados na figura 4.1, como caixas ao lado do repositório.

As três classes de elementos, em conjunto, permitem gerar as funcionalidades que auxiliam o ciclo de transformação do conhecimento. As etapas do ciclo de criação do conhecimento, potencialmente mais beneficiadas, são as de externalização e internalização. Na externalização, isto é, a transformação do conhecimento tácito em explícito, a arquitetura auxilia, facilitando o registro e o armazenamento dos conhecimentos explícitos. Quanto à internalização, a arquitetura auxilia, suportando a busca e a identificação de conhecimentos explícitos relevantes, os quais poderão ser utilizados pelos indivíduos e ser transformados em novos conhecimentos tácitos. Outro importante papel da arquitetura no ciclo é o de integração entre as etapas, principalmente quando realizadas por diferentes membros da organização. Ela permite, por exemplo, que os conhecimentos explícitos, externalizados por um dos membros, possam ser localizados e utilizados pelos demais membros da empresa, intensificando o ciclo de transformação dos conhecimentos.

A arquitetura pode auxiliar, também, na etapa de combinação, na qual conhecimentos explícitos são combinados, formando novos conhecimentos explícitos. Faz isso facilitando a busca, identificação e apresentação dos diferentes

conhecimentos explícitos a ser combinados. Ainda nesta etapa, ela auxilia o registro do conhecimento explícito obtido dessa combinação. Uma outra característica da arquitetura que pode auxiliar neste aspecto é a incorporação da visão de modelos de empresa. Esses são um tipo de conhecimento explícito, que representa o processo de negócio como um todo, formando uma visão sintética deste. Espera-se que a visão do modelo contribua para a contextualização dos conhecimentos, facilitando sua combinação, principalmente no caso de conhecimentos oriundos de diferentes setores funcionais e áreas de especialização.

Quanto à etapa de socialização, onde conhecimento tácito é transformado em conhecimento tácito, a arquitetura tem pouco a oferecer, conforme demonstra a figura. Essa etapa do ciclo depende fundamentalmente da interação entre as pessoas. Ferramentas de informática podem auxiliar somente no momento anterior a esse encontro, facilitando a identificação de pessoas que tenham um determinado conhecimento ou habilidade.

No decorrer deste capítulo, são descritas, em detalhe, as classes de elementos principais da arquitetura: repositório (item 4.3), funcionalidades (4.4) e os elementos conceituais de suporte (4.5). Ao final deste capítulo, no item 4.6, descreve-se como utilizar a arquitetura para a construção de sistemas para a gestão do conhecimento.

4.3 Repositório e os Tipos de Conhecimentos Explícitos

O repositório de conhecimentos explícitos incorporado na arquitetura é representado pela figura 4.2. Ele é dividido, conforme apresentado na visão geral, em três camadas contendo diferentes tipos de conhecimentos explícitos, definidos como registros, sentenças e modelos do processo. Deve permitir o armazenamento desses diferentes tipos de conhecimento e, ao mesmo tempo, dos relacionamentos entre eles. Com o uso contínuo de uma solução baseada na arquitetura, espera-se a obtenção de um conjunto de modelos de referência, cada qual destinado a diferentes tipos de processos de desenvolvimento realizados dentro de uma organização, os quais são representados no topo da figura 4.2. Nos subitens a seguir, descreve-se, com mais detalhes, a definição e a forma de classificação de cada um dos diferentes tipos de conhecimentos explícitos armazenados, ou cada uma das diferentes camadas do repositório.

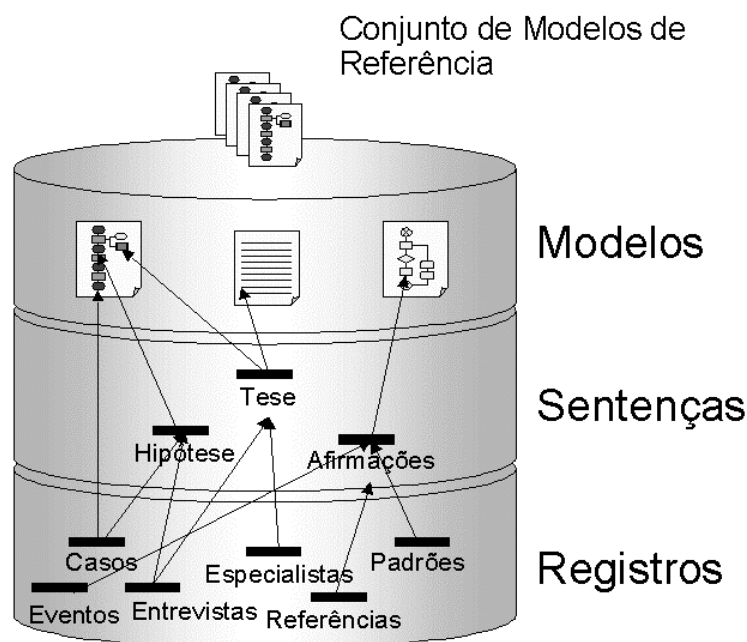


Figura 4. 2 : Repositório de Conhecimentos Explícitos da ARCE-PDP

4.3.1 Registros

Os registros são os conhecimentos explícitos ditos não estruturados ou “em estado bruto”, isto é, aqueles que se encontram na forma como foram manipulados pelo usuário na rotina normal de trabalho. São todos os documentos gerados durante o processo de desenvolvimento de produto ou obtidos, interna ou externamente, por qualquer um dos membros da organização.

Estes conhecimentos são classificados em seis tipos básicos que identificam a estrutura de dados necessária para armazenar as informações que o descrevem: eventos, padrões, especialistas, entrevistas, casos e referências bibliográficas.

Todo registro cujo conteúdo esteja fortemente relacionado com a idéia de atividades no tempo, como, por exemplo, congressos, conferências e reuniões são registrados com a estrutura denominada evento. Em padrões, reúnem-se os registros relacionados com instruções de trabalho, procedimentos, formulários e outros documentos, que servem de modelo ou instrumento durante a execução das atividades de desenvolvimento. Em especialistas, armazenam-se conhecimentos explícitos sobre contatos, isto é, pessoas que reúnem uma habilidade especial, prestam algum tipo de serviço especializado ou detêm um conhecimento bastante

desenvolvido sobre determinado assunto, podendo estar dentro ou fora da organização. Em entrevistas, são armazenados os resultados de conversas com pessoas de outras empresas, fornecedores, consultores e outros. Podem ser atas de reuniões com estas pessoas, resultado de aplicação de um questionário formal ou outro conhecimento da mesma natureza. Como casos, são armazenados os conhecimentos explícitos contendo a descrição de um evento ocorrido dentro e/ou fora da organização. Nos casos deve-se registrar o contexto em que este ocorreu (local, data, motivo, momento e eventos anteriores), o problema que foi enfrentado, a solução adotada e os resultados obtidos dessa solução. Por fim, todos os conhecimentos explícitos, na forma de documentos, sejam eles artigos, livros, periódicos, desenhos, relatórios e outros tipos de publicação, são classificados como referências.

Estes tipos básicos foram especificados de maneira a ser obtido o número mínimo de estruturas de dados capazes de registrar qualquer tipo de conhecimento explícito gerado durante o processo de desenvolvimento de produto. O objetivo é facilitar a implantação dos sistemas e das bases de dados, pois, conforme essa classificação, pode-se detalhar os atributos específicos para cada um desses tipos de registro. Com efeito, qualquer conhecimento explícito do tipo registro poderia então ser armazenado, bastando identificar qual a estrutura mais adequada, de acordo com as características do registro. Em termos de implementação computacional, há, ainda, a vantagem de possibilitar a criação de uma “superclasse” registro, isto é, uma entidade registro, contendo o conjunto de atributos comuns a todos registros, sendo os atributos específicos armazenados numa subclasse, ou entidade específica para cada um destes determinados tipos (caso, referência, evento, etc...). Por exemplo, dados comuns a todos os registros podem ser: título, autor, palavra-chave, data do cadastro, documento e link da internet. E cada tipo de registro poderia ter dados específicos como por exemplo: data, ocorrência, local e duração para evento; localização, referência e ano de publicação para referência; e assim por diante. Um exemplo bastante claro pode ser observado no desenvolvimento da solução apresentada no capítulo 5, mais especificamente, o item 5.2.3.

Para o usuário de um sistema, no entanto, a distinção dos registros segundo este leque limitado de estruturas de dados não é suficiente para as tarefas diárias de

busca de conhecimentos; e, principalmente, para a adequação das soluções geradas à partir da arquitetura às necessidades de sistematização de documentos das organizações. Assim, foram adicionados na arquitetura os conceitos de classe e subclasse de registros. As classes são tipos gerais de registro classificados conforme as necessidades dos usuários, as quais podem ser “customizadas” por eles empregando-se os termos consagrados dentro da organização. As subclasses são divisões das classes, isto é, são entidades com relacionamentos de filhos da classe. Uma subclasse pode apresentar uma ou mais subclasses e cada subclasse é uma divisão de apenas uma única classe.

Empregando-se as classes e subclasses pode-se então criar uma classificação dos registros sob medida para as necessidades da empresa. Pois a definição da forma de organização dos registros precisa respeitar o tamanho e complexidade da organização; e também os termos específicos já consagrados neste ambiente. Isto é fundamental para garantir flexibilidade à solução gerada à partir da arquitetura, de forma que ela possa ser adaptada a diferentes ambientes de aplicação.

Unindo as classificações apresentadas de tipos de registro, classes e subclasses, obtém-se a sistemática para classificação de registros prescrita pela arquitetura: uma maneira prática e interessante de se organizar os registros, conforme demonstrado na figura 4.3. Enquanto o usuário pode “visualizar” os conhecimentos explícitos por meio da classificação dos registros em classe e subclasse, o sistema gerencia e relaciona os dados empregando um conjunto finito e pequeno de seis estruturas de dados. No exemplo da figura o sistema teria sido especificado contendo as classes Reuniões e Dimensionamento de Eixos. Representam, portanto, assuntos significativos para o usuário, facilitando, a escolha do local onde ele deve procurar a informação que está buscando. A primeira classe foi dividida em 4 subclasses e a segunda em 5. Estas subclasses detalham ainda mais as definições anteriores e cada uma delas está relacionada diretamente com um dos 6 tipos básicos de estrutura de dados. Do ponto de vista dos integradores e desenvolvedores da solução, porém, este alto grau de flexibilidade na classificação não representa um esforço maior, pois eles trabalham com um conjunto pequeno e suficiente de estruturas.

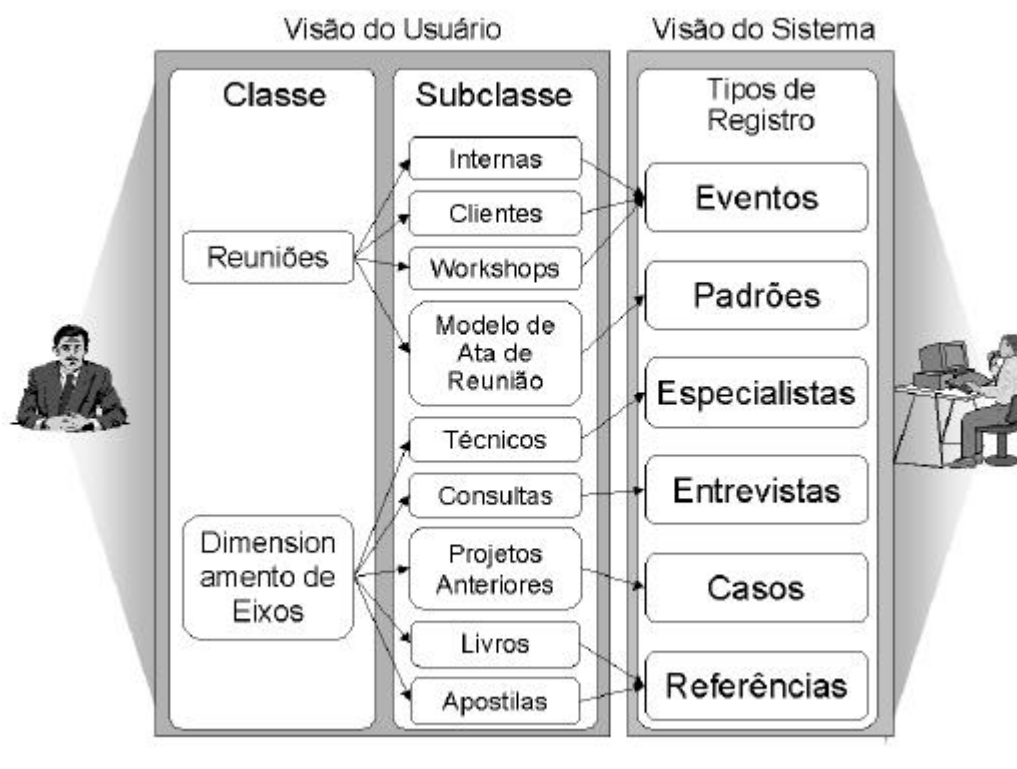


Figura 4.3 : Classificação de Registros segundo a Arquitetura

4.3.2 Sentenças

No contexto da arquitetura, uma sentença é definida, conforme LAROUSSE(1992), como uma unidade sintática que exprime um pensamento completo e constituída de somente um sujeito e um predicado. As idéias transmitidas pelas sentenças estão contidas nos registros e, portanto, cada uma delas deve estar relacionada a um ou mais registros.

Este tipo de conhecimento serve para resumir e organizar o conhecimento explícito que se encontra disperso nos registros, o qual contém em si um conjunto grande de idéias num formato não estruturado e que demanda um esforço grande de leitura e interpretação para ser absorvido. As sentenças permitem que um usuário interessado em obter entendimento melhor sobre algum assunto, ao invés de consultar esta grande quantidade de documentos possa, navegando nas sentenças, avaliar um conjunto restrito e estruturado de afirmações que lhe permitam entender idéias centrais dentro daquele tema. Caso deseje se aprofundar em uma das afirmações, poderá consultar os documentos (registros) que estão relacionados com a

sentença, acessando as fontes de origem e, portanto, permitindo-lhe avaliar criticamente estas afirmações.

As sentenças são classificadas, quanto à sua estrutura, como simples ou compostas. As simples são frases únicas com apenas um sujeito e um verbo, por exemplo: “CAD é uma ferramenta muito utilizada no desenvolvimento de produto” ou “A primeira casa do QFD é a mais utilizada”. As sentenças compostas apresentam a estrutura IF...THEN (SE...ENTÃO), típicas das regras de produção, por exemplo: “IF a interface do produto com o cliente é complexa THEN deve-se utilizar a ferramenta do QFD”.

As sentenças são classificadas também quanto à sua origem entre enunciados ou argumentos. As primeiras representam, como o nome diz, afirmações realizadas por algum dos registros. Já os argumentos foram deduzidos ou induzidos pelo construtor da sentença, com base em outras sentenças e, portanto, devem estar relacionadas a mais de uma sentença. Do ponto de vista do sistema é importante diferenciá-los porque o argumento é uma inferência de um membro da organização e deve ser analisado criticamente como tal.

Um exemplo da aplicação destas classificações é apresentado na figura 4.4. Ela descreve 4 sentenças relacionadas à aplicação da técnica de QFD. Três delas são compostas, possuindo a estrutura IF...THEN e uma é simples, aquela relativa à aplicação da primeira casa do QFD. As três sentenças descritas na parte inferior da figura foram originadas apenas de registros, sendo, portanto, enunciados. A sentença que aparece na parte superior é uma inferência derivada da análise das três outras sentenças, sendo, portanto, classificada como argumento.

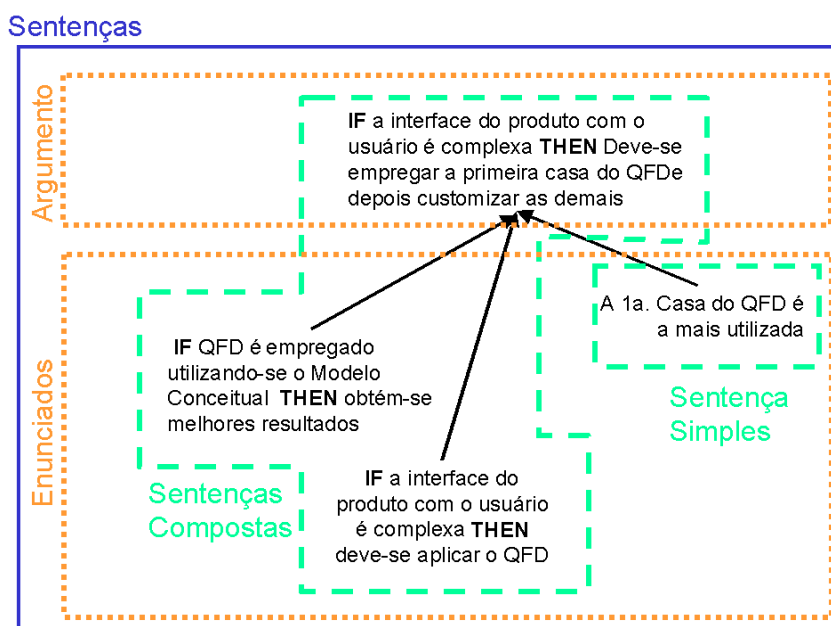


Figura 4. 4 : Exemplo de Classificação de Sentenças

De maneira a facilitar a validação das sentenças, essas são classificadas ainda em teses e hipóteses. Quando baseadas em poucos registros, ou muitos registros com opiniões contraditórias, recebem o rótulo de hipóteses, sendo, portanto, pontos a ser verificados e avaliados cientificamente. Quando a sentença traz muitos registros comprovando-a ou refutando-a completamente, são descritas como teses. Nada impede a transformação, no transcorrer do tempo, de hipóteses em teses e vice-versa. A primeira transformação é bastante natural, basta que surjam novos registros corroborando-a. A segunda deve acontecer quando uma determinada sentença, avaliada unanimemente como verdade por vários registros, começa a “ganhar” registros contradizendo-a, a tal ponto que a antiga tese pode se transformar novamente em uma incógnita merecedora de mais estudos para a comprovação de sua veracidade ou falsidade, isto é, novamente em hipótese. Retomando o exemplo anterior pode-se classificar as sentenças em hipóteses e teses conforme é apresentado na figura 4.5.

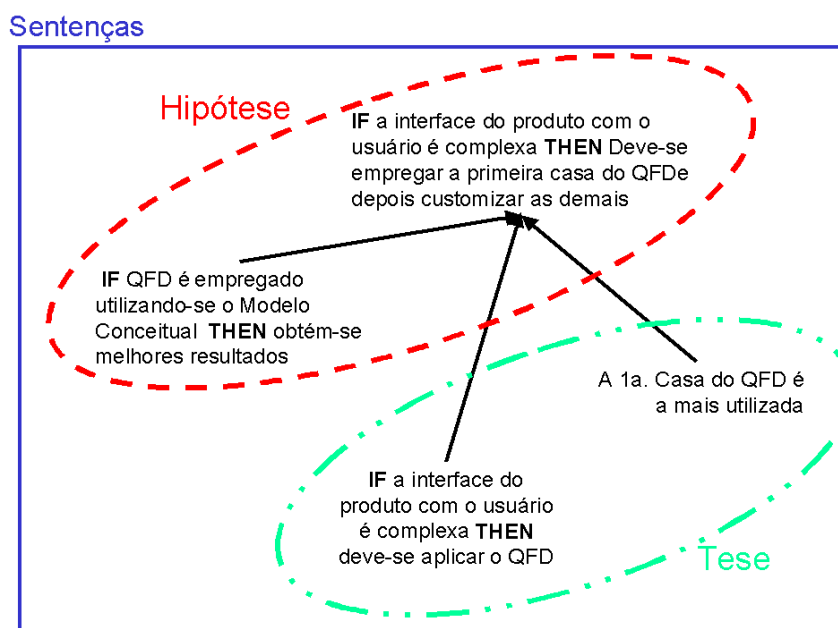


Figura 4. 5: Exemplo de classificação de sentenças conforme o grau de validação

As nomenclaturas e definições empregadas nesse nível de conhecimento explícitos foram baseados nos trabalhos de COSTA(1980) e OLIVA(1980), sendo, portanto, termos correntes na área de lógica-matemática. Essa decisão foi tomada como forma de se manter a parcimônia na definição dos termos, evitando-se, assim, o risco (e a tentação) de se criarem categorias atualizadas quanto ao discurso corrente (como *best-practice*, competência, mapa mental, etc..), porém mais confusas e ambíguas. Também, porque, essa estratégia de criação de sentenças não foge às já amplamente empregadas tentativas de sistematizar conhecimentos com base em regras (veja no item de revisão bibliográfica as representações de conhecimento baseado em regras). O fator diferencial dessa proposta é a utilização das regras em conjunto com outros elementos, assumindo-as como parcialmente suficientes para a representação do conhecimento. Na proposta da arquitetura, as sentenças (regras) formam a parte altamente estruturada e objetiva dos conhecimentos explícitos, que faz sentido, apenas, quando complementada pela outra parte desestruturada, constituída pelo conjunto de registros.

Esse aspecto torna fundamental o papel das sentenças na sistemática de validação do conhecimento explícito. Criando-se sentenças e validando-as, abre-se a possibilidade para que o usuário tenha acesso a um conjunto estruturado de idéias

sobre o tema, que o orienta na leitura e interpretação do conhecimento armazenado nos registros. O Modelo de Validação e Sistematização do Conhecimento, a ser apresentado no item 4.5.1, faz uso intensivo das sentenças.

4.3.3 Modelos de Processo

Os modelos localizados na parte superior da arquitetura são representações de processos de negócios, descrevendo as atividades, informações, recursos, pessoas, etc.. Esta descrição pode ter várias aplicações dentro da empresa, conforme apresentado no item revisão bibliográfica: melhorias no processo de desenvolvimento, planejamento, gerenciamento, definição de custos (aplicação de *Activity Based Costing*), e outras. O papel dos modelos nestas aplicações é o de servir de instrumento para o compartilhamento do conhecimento sobre o processo de negócio, entre os diferentes membros do time. Como representação sintética do processo, eles servem como um referencial comum, padronizando linguagens e facilitando a interação entre os diferentes profissionais envolvidos nesse processo de negócio.

Na arquitetura, procurou-se fazer uso dessa qualidade especial dos modelos, a de viabilizar uma visão geral (“fotografia”) do processo, com o intuito de permitir uma contextualização dos conhecimentos explícitos, armazenados no repositório. Para tanto, os modelos de empresa foram adicionados na camada superior, juntamente com a possibilidade de relacioná-los com os demais conhecimentos explícitos. Assim, deve-se permitir que os usuários analisem o processo como um todo e “naveguem” dentro das atividades específicas, em busca dos conhecimentos explícitos que possam auxiliá-los na condução de tais atividades. O contexto fica claro para o usuário porque o conhecimento explícito, presente nas duas camadas inferiores, é analisado pelo usuário depois que ele já reconheceu um ou mais pontos específicos do processo de negócio em que tal conhecimento pode ser aplicado. Uma das hipóteses desse trabalho, também incorporada nos requisitos da arquitetura, é a de que essa característica facilita a compreensão, pelo usuário, da importância e da potencial contribuição do conhecimento explícito encontrado.

Além disso, o objetivo dessa configuração é permitir que o conjunto de conhecimentos acumulados seja transformado, com o passar do tempo, em alterações na forma de condução dos processos de negócio, por meio das atualizações e

proposições de novos modelos de processo. Com isso, espera-se “potencializar” o emprego dos conhecimentos explícitos armazenados, de forma que eles não sejam apenas aproveitados no formato original, o que demanda tempo e reflexão do usuário, mas também, de forma “indireta”, ao ser incorporados como recomendações dentro dos modelos do processo.

O acúmulo de conhecimentos explícitos e de modelos, complementados com outras atividades relacionadas à gestão de conhecimento dentro da organização, deve, enfim, gerar um conjunto específico de modelos que sirvam como referência para o desenvolvimento de projetos com características semelhantes dentro da empresa. Os modelos de referência representados no topo da figura 4.2.

Os modelos são classificados em dois tipos básicos, segundo a abrangência de sua aplicação: de referência e específico. Os modelos de primeiro tipo são aqueles que servem de base para um conjunto de processos de desenvolvimento, com características específicas. São, portanto, mais genéricos e, como o próprio nome diz, servem de referência na geração de outros modelos. Os modelos específicos descrevem o processo de desenvolvimento de um produto específico ou de uma classe de produtos com características bastante coesas.

Para facilitar o armazenamento e a aplicação desses modelos, cada um deles é caracterizado por meio de uma tipologia de processos de desenvolvimento de produto, apresentada no item 4.5.2, a seguir.

Outra distinção realizada nos modelos é quanto ao formato empregado na sua descrição. O repositório deve permitir o armazenamento de modelos em qualquer formato, isto é, empregando-se diferentes metodologias (*frameworks* ou métodos) e/ou ferramentas de modelagem. Apesar disso, há um formato padrão (dito modelo formal), obtido por meio da geração do modelo com o emprego do *framework* de modelagem especificado pela arquitetura, o qual é apresentado no item 4.5.3. Assim, quanto ao formato, os modelos são classificados entre modelos formais e informais. Os formais são ainda diferenciados, em Web e Relatório, que são as duas formas de apresentação previstas no *framework* (veja item 4.5.3). Os informais seguem classificações específicas da organização, que adota a solução, seja empregando o nome da ferramenta de modelagem ou da metodologia de modelagem empregada. Seriam classificações como: outros, PDF, SADT, etc...

4.4 Funcionalidades

A figura 4.3 descreve, de maneira sintética, as principais funcionalidades previstas dentro da arquitetura, separadas por tipo de conhecimentos explícitos: registros, sentenças e modelos. Parte delas são comuns a todos esses tipos de conhecimentos explícitos, enquanto outras são específicas de um ou mais destes.

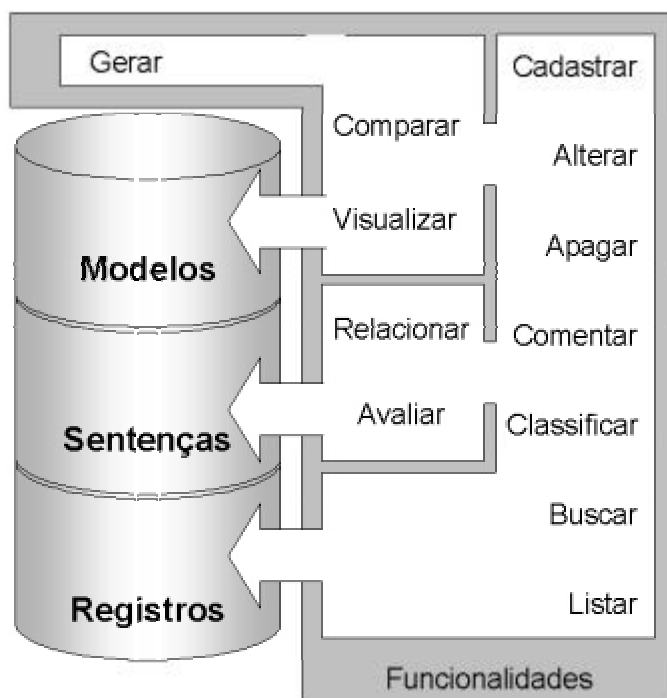


Figura 4. 6: Funcionalidades Básicas Previstas na Arquitetura

As funcionalidades básicas, disponíveis para todos os conhecimentos explícitos, são:

- cadastrar: funcionalidade de armazenamento e registro do conhecimento explícito. Sempre que possível deve ser arquivado o próprio documento no formato eletrônico, de forma que os usuários tenham acesso diretamente a ele à partir da ferramenta. Nos casos em que isto é impossível, deverá haver um campo para indicação clara de sua localização física;
- alterar: funcionalidade de alteração dos dados e do próprio documento eletrônico referente ao conhecimento explícito;

- apagar: funcionalidade de eliminação de conhecimentos explícitos das diferentes áreas do processo de desenvolvimento de produto;
- comentar: possibilidade de adição, pelos usuários, de textos comentando os conhecimentos explícitos. Estes textos devem conter um título, um campo aberto, data e identificação do usuário autor do comentário;
- classificar: deve-se permitir a classificação de cada conhecimento explícito segundo os parâmetros discutidos na apresentação do repositório. Para os conhecimentos do tipo de registro deve-se classificá-los segundo a classe e subclasse (figura 4.3). As sentenças devem ser classificadas segundo a validação em tese e hipótese e segundo a origem em argumento ou sentença (figuras 4.4 e 4.5). Os modelos devem ser classificados quanto a formalização (em formais e informais), quanto ao tipo (de referência ou específicos) e quanto aos critérios da tipologia de processos de desenvolvimento de produto apresentada no item 4.5.2;
- buscar: funcionalidade de busca de conhecimentos explícitos por diferentes meios, tais como palavras-chave, título, usuário, data e palavras abertas inseridas pelo próprio usuário e
- listar: funcionalidade de geração de listas de conhecimentos explícitos conforme as diferentes classificações possíveis. As listas tornam mais prática a busca de um conhecimento explícito quando o usuário conhece de antemão o tipo de documento que deseja recuperar;

Além destas funcionalidades básicas, para os conhecimentos na forma de sentença devem ser disponibilizadas as seguintes funcionalidades:

- relacionar: funcionalidade de atribuir o relacionamento da sentença com os registros que a corroboram. Esse relacionamento deve ser feito de forma bastante interativa, de tal modo que o usuário possa visualizar a sentença e os registros já relacionados enquanto realiza buscas de novos registros, que poderiam ser ligados à sentença em análise;
- avaliar: possibilidade de avaliar a validade da sentença por meio dos registros relacionados, conforme previsto no modelo de validação e sistematização de conhecimentos explícitos apresentados no item 4.5.1.

Para os conhecimentos explícitos na forma de modelo devem ser acrescentadas as seguintes funcionalidades:

- gerar: o conjunto de funcionalidades para auxílio no desenvolvimento de modelos de empresa empregando-se o framework de modelagem especificado na arquitetura (conforme item 4.5.3);
- visualizar: capacidade de visualização do modelo, permitindo a navegação através de todas as suas visões e construtos e
- comparar: conjunto de funcionalidades de apoio à comparação de modelos.

Conforme é possível observar na parte inferior da figura 4.6, os registros não possuem funcionalidades especiais, além das básicas já descritas. Deve-se notar ainda que o acesso a essas funcionalidades depende do tipo de usuário do sistema, conforme especificado no modelo de perfis de usuário (item 4.13)

4.5 Elementos de Apoio

Os elementos de apoio, especificados na arquitetura (veja figura 4.1), são descritos a seguir.

4.5.1 Modelo de validação e sistematização dos conhecimentos explícitos

Um dos elementos de apoio da arquitetura é um modelo para validação e registro dos conhecimentos explícitos, formado por um conjunto de regras para o relacionamento entre registros e sentenças. O objetivo é possibilitar que as sentenças formem uma síntese estruturada do conhecimento armazenado nos registros.

O princípio básico desse modelo é estabelecer regras que façam da camada de sentenças uma síntese estruturada do conhecimento incorporado dentro dos registros, armazenados na camada mais do repositório. Conforme são armazenados registros sobre determinado assunto, um especialista, que será denominado “engenheiro do conhecimento”(veja item 4.5.4), poderá criar uma série de sentenças e relacioná-las com os registros que serviram de sustentação para a sua elaboração. Este conjunto de sentenças forma um pequeno discurso, isto é, um corpo teórico que resume as idéias principais contidas nos registros, sendo permitido, ao especialista, adicionar registros de sua própria autoria. O usuário pode, além de analisar as sentença em si, consultar suas fontes e mesmo, conforme o caso, questionar o modo como o especialista construiu e relacionou as sentenças.

A dimensão de validação é proporcionada por um conceito simples também ligado a esse relacionamento. O especialista, além de cadastrar as sentenças, avalia o quanto o registro corrobora cada sentença, baseando-se num critério que leva em conta a “confiabilidade” ou “validade” da informação contida no registro. Avalia-se, na verdade, a forma como a afirmação ou o conhecimento contido no registro foi obtido, considerando-se tanto mais confiável quanto mais “criterioso” e “profundo” tenha sido o trabalho que levou à criação desse conhecimento.

A figura 4.7 apresenta essa avaliação para o mesmo exemplo empregado na descrição dos tipos de sentenças, item 4.3.2 (veja figura 4.4). Cada relacionamento entre um registro e uma sentença recebe um **Grau de Validação**, isto é, uma nota entre -1 e +1. O valor -1 ou +1 quando o registro afirma de maneira enfática respectivamente, a falsidade ou a veracidade da sentença, ou seja, quando se trata de

um registro contendo uma evidência forte como aquelas baseadas em *surveys* com ampla extensão e análises profundas, casos bem documentados e claros e/ou opiniões convictas de especialistas consagrados. Os valores entre esses dois pólos são atribuídos a relacionamentos com registros que afirmam ou negam a sentença, mas, tratam-se evidências mais fracas (menos precisas) tais como *surveys* parciais ou baseados apenas em opiniões, estudos de caso mais superficiais e opiniões de especialistas desconhecidos. Recebe valor 0 quando, negando ou corroborando a sentença, o nível de confiabilidade da fonte é mínimo.

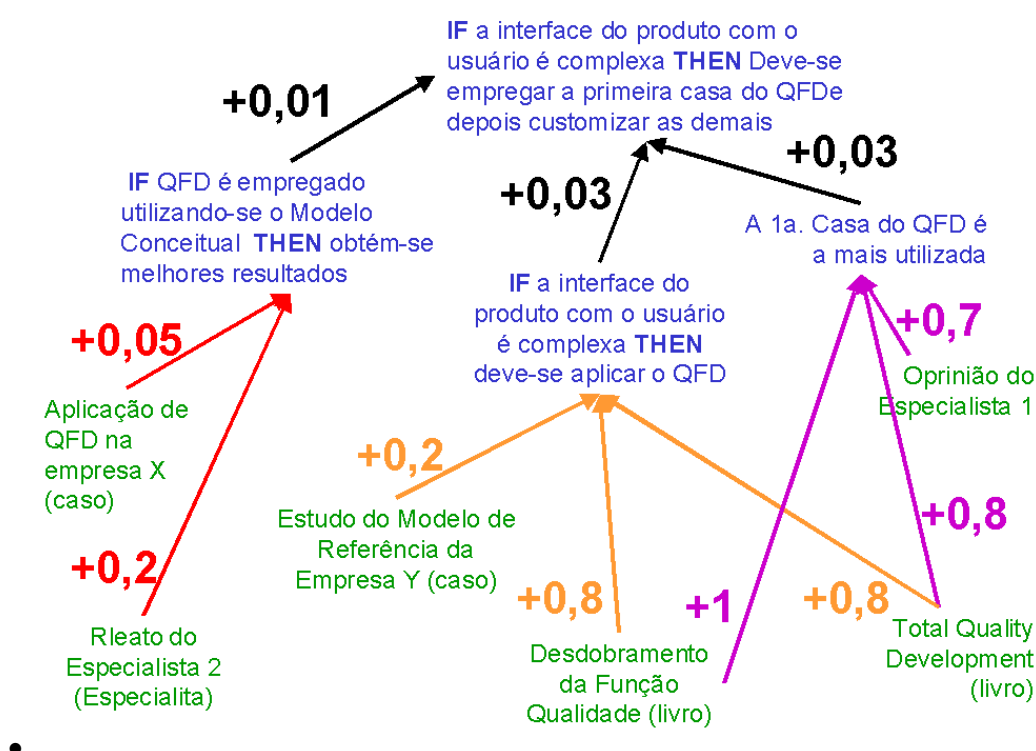


Figura 4. 7: Exemplo do grau de validação de registros

O quadro 4.1 apresenta valores extremos que servem de parâmetro para a realização da avaliação dos relacionamentos, lembrando-se que a nota é um número contínuo. A primeira coluna (geral) apresenta o critério de forma mais genérica. As colunas que a seguem apresentam esse mesmo critério, adaptado às especificidades de cada um dos tipos de registros. Por exemplo, quando na figura 4.7 apresenta-se o valor +0,7 para o relacionamento entre o especialista 1 e a sentença “A primeira casa do QFD é a mais utilizada”, deve-se entender, conforme a linha 2 e coluna 6 do quadro 4.1, que esse especialista corrobora essa afirmação da sentença e que ele tem

atuado com a técnica do QFD por vários anos, tendo podido aplicá-la em várias empresas.

Quadro 4.1 : Critério para avaliação do Grau de Validação entre registro e sentença

GV	Geral	Referências Bibliográficas	Casos	Livros	Especialistas
"+/- 1"	O registro valida (+1) ou refuta (-1) a sentença e se baseia em dados confiáveis, obtidos criteriosamente e de maneira a possuir alta validade externa.	<ul style="list-style-type: none"> • Surveys com ampla extensão; • Análises profundas com evidências fortes e documentais, • Análises considerando todas as visões do problema; • Estudos de caso bem escolhidos e com dados cruzados; 	<ul style="list-style-type: none"> • Um caso ou problema bem escrito e profundo. Com análise cuidadosa do contexto, limitações e implicações 	<ul style="list-style-type: none"> • Riqueza de detalhes e exemplos; • Apresentam de maneira rigorosa as fontes que suportam a proposição do enunciado; e • Autor Conhecido. 	<ul style="list-style-type: none"> • Especialista com muitos anos de atuação especificamente neste assunto; • Vários casos de problemas similares enfrentados.
"+/- 0,5"	O Registro ou enunciado valida (+0,5) ou refuta (-0,5) a sentença e se baseia em dados obtidos criteriosamente mas com limitações	<ul style="list-style-type: none"> • Surveys com extensão menor e/ou superficiais; • Análises abrangentes, mas não muito profundas; • Estudos de caso bem escolhidos mas não muito profundos; 	<ul style="list-style-type: none"> • Casos profundos, mas não suficientemente abrangentes e/ou com pouca explicação dos contextos e limitações 	<ul style="list-style-type: none"> • Falta total clareza na fundamentação; e • O autor é conhecido 	<ul style="list-style-type: none"> • Muito anos atuando nesta área, sem maiores comprovações de casos de sucesso e/ou muitos problemas similares enfrentados
0	O Registro valida ou refuta a sentença, mas é baseado em dados questionáveis ou com sérias limitações;	<ul style="list-style-type: none"> • Qualquer tipo de estudo (Survey, Caso ou Análise) com erros grosseiros ou não muito bem descritos e justificados; 	<ul style="list-style-type: none"> • Casos cujo relato não é confiável 	<ul style="list-style-type: none"> • Não são apresentados os fundamentos em que se baseou a informação para o caso • O autor é desconhecido 	<ul style="list-style-type: none"> • Informações sobre especialista não confiáveis

O valor referência para a validade de uma sentença é obtido pelo **Índice de Validação**, obtido por meio da soma dos valores de cada um dos relacionamentos da sentença, o **Grau de Validação**, multiplicado pelo número de registros relacionados, conforme a fórmula apresentada na Eq.1.

$$Is = \sum_{i=1}^n Gi \times n \quad , \text{ onde:}$$

Is = Índice de validação de uma sentença qualquer s;

G = Grau de validação de um registro qualquer (n) relacionado com a sentença s e

n = Número de registros relacionados com a sentença s.

A figura 4.8 apresenta os Índices de Validação para as sentenças do exemplo apresentado na figura 4.7. O **Índice de Validação** foi elaborado de forma a permitir

uma avaliação comparativa da validade da sentença, de uma forma dinâmica. Uma sentença, que nasce como fortemente positiva, pode tornar-se mais e mais negativa conforme vão sendo incorporados na base registros que a neguem, isto é, afirmem ser ela falsa. Da mesma forma, uma sentença fortemente negativa pode tornar-se positiva com o passar do tempo.

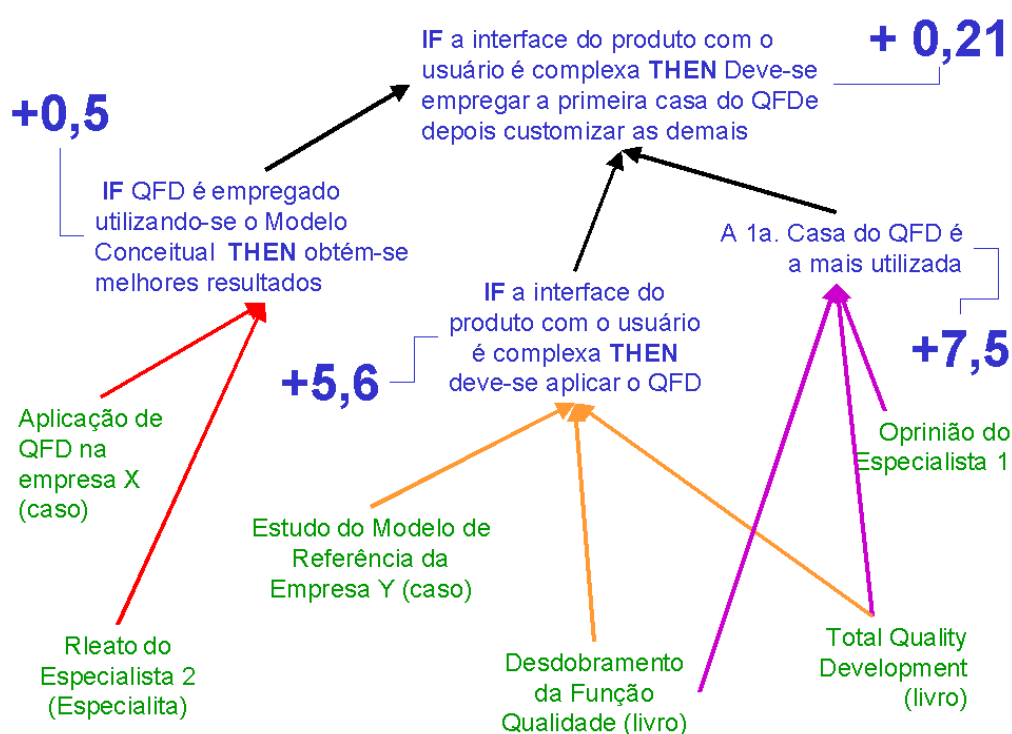


Figura 4. 8 : Exemplo de Índice de Validação de Sentenças

Na medida do **Índice de Validação** de sentenças foi incorporado o número de registros ligados à sentença como fator multiplicador, para garantir uma melhor comparação entre as sentenças. Empregar somente o número bruto, obtido da soma dos **Graus de Validação** dos relacionamentos, facilitaria a confusão entre uma sentença corroborada por poucos registros altamente confiáveis, com outra sentença contendo vários registros insignificantes. Pela soma simples, ambas podem apresentar índices de validação próximos. A multiplicação pelo número de registros não elimina, totalmente, essa possibilidade, mas a atenua, distinguindo melhor os índices, pois as sentenças suportadas por quantidades de registros diferentes terão maior probabilidade de apresentarem valores de grau de validação também distintos.

A figura 4.9 e a tabela 4.1 apresentam um exemplo fictício, contendo 4 sentenças (S1, S2, S3 e S4) e 10 registros (R1, R2, R3, R4, R5, R6, R7, R8, R9 e R10), nas quais pode ficar mais claro o sentido do grau de validação. Na figura, são representados os relacionamentos dos registros com cada uma das sentenças, os mesmos reproduzidos na tabela 4.1, por meio do cruzamento entre as linhas (registros) e as colunas (sentenças). Os valores de **Grau de Validação** para cada registro e sentença são apresentados, tanto nas setas da figura 4.9 como nas células da tabela 4.1. A tabela apresenta ainda o cômputo do Índice de Validação, obtido pela soma dos Graus de Validação, atribuídos no relacionamento com os registros, multiplicado pelo número de registros relacionados. Desta forma, o índice inclui uma medida, tanto do nível de validação (por meio da soma), quanto do número de evidências suportando-a (por meio do número de registros).

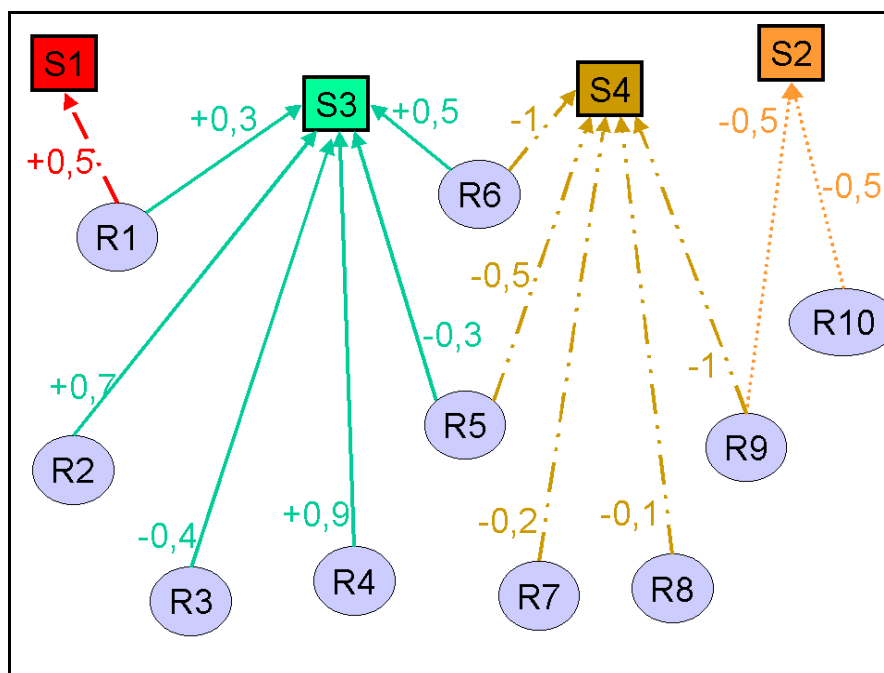


Figura 4.9 : Exemplo Esquemático de sentenças avaliadas

Tabela 4. 1: Cálculo do índice de validação para o exemplo

		Sentenças			
		s1	s2	s3	s4
Registros	R1	0,5		0,3	
	R2			0,7	
	R3			-0,4	
	R4			0,9	
	R5			-0,3	-0,5
	R6			0,5	-1
	R7				-0,2
	R8				-0,1
	R9		-0,5		-1
	R10		-0,5		
Soma de Grau de Validação		0,5	-1	1,7	-2,8
No. de Registros		1	2	6	5
Índice de Validação		0,5	-2	10,2	-14

A sentença mais “validada”, ou com maior grau de verdade, é a de número 4 (S4). Deve-se notar ainda que se trata de um valor negativo, indicando que se tem bastante certeza de que seu inverso, ou a sua negação, é verdadeira. A sentença 3 aparece em segundo, mesmo possuindo um número maior de registros a validá-la. Nota-se que os registros que corroboram essa sentença apresentam Graus de Validação contraditórios, uns negando e outros afirmando, mostrando que ela precisa ser verificada mais detalhadamente. As duas outras apresentam valores bem inferiores, demonstrando ser sentenças baseadas em poucas evidências. A partir desses valores, o usuário teria ao seu dispor uma informação adicional para a análise das sentenças, tendo acesso a uma medida comparativa do grau de validação delas.

Logicamente, como deve ter ficado claro para o leitor, no decorrer do texto, o princípio fundamental deste modelo de sistematização e validação não busca levar os usuários a interpretar estes valores como uma medida de validade do conhecimento. Ao contrário, espera-se que esses valores sirvam como uma referência inicial. O objetivo é que o usuário tenha uma visão comparativa do nível de comprovação das sentenças, permitindo-lhe distinguir afirmações

bastante consolidadas das hipóteses sob cerrada discussão. O usuário desse modelo deve ter em mente que essa avaliação é dependente e fundamentalmente baseada na interpretação dos especialistas que estão atribuindo o **Grau de Validação** de cada registro em relação à sentença. Portanto devem analisar criticamente estes resultados.

O resultado do **Índice de Validação** pode, além disso, ser utilizado pelo engenheiro de conhecimento para a classificação de cada sentença entre tese e hipótese. Foi visto na apresentação do repositório que os especialistas podem classificar uma determinada sentença como uma tese, sempre que a julguem como suficientemente comprovada. O índice de validação pode ser um sinal de possíveis teses, cabendo aos especialistas analisarem tais sentenças e, caso concordem, proceder sua classificação como tal. Para o usuário final, a classificação como tese e hipótese seria mais um indicativo de nível de validação à sua disposição, em certa medida mais simples e direto que os índices. O quadro 4.2 apresenta classificações possíveis de teses e hipóteses para o exemplo fictício apresentado. A figura 4.5 mostra uma possível classificação para o exemplo da técnica do QFD apresentado na figura 4.8.

Quadro 4.2 : Exemplo de avaliação de sentenças

	Grau de Validação	Número de Registros	Somatória dos Índices de Validação	Tipo
Sentença 1	0,5	1	0,5	Hipótese
Sentença 2	-0,5	2	-1	Hipótese
Sentença 3	1,7	6	10,2	Hipótese
Sentença 4	-2,8	5	-14	Tese

4.5.2 Tipologia de processos de desenvolvimento de produto

Uma importante característica da arquitetura é a possibilidade da geração de um conjunto de modelos de referência, os quais podem ser utilizados pelos membros de uma organização, nas tarefas de desenvolvimento. Para tanto, é fundamental identificar as características específicas que distinguem cada processo de desenvolvimento quanto à forma de condução e gerenciamento.

Neste sentido, empreendeu-se um esforço para identificar quais fatores ou critérios seriam fundamentais para diferenciar um processo de desenvolvimento de produto. O resultado deste esforço e a tipologia desenvolvida são descritos detalhadamente em ROZENFELD & AMARAL (1999). Realizou-se, primeiramente,

uma revisão bibliográfica para levantar os fatores que influenciam na forma de se conduzir o processo de desenvolvimento de produto. Em seguida, foram entrevistados alguns especialistas na área, apresentando-se a lista para que avaliassem o conteúdo. Como resultado, obteve-se um conjunto de fatores principais, a ser caracterizados no processo de desenvolvimento de produto.

Para cada um dos fatores foram definidas possíveis dimensões, originando uma tipologia para caracterizar o processo de desenvolvimento de produto. O resultado final foi um conjunto de dimensões e os seus possíveis valores descritos no quadro 4.3.

A hipótese que vem sendo testada, desde o desenvolvimento dessa tipologia, é a de que ela permite identificar as características fundamentais que distinguem os diferentes processos de desenvolvimento de produto. Espera-se, com isto, facilitar a escolha do modelo mais apropriado, permitindo comparações, a geração de novos modelos e a adaptação destes quando na sua utilização em casos de processos de desenvolvimento de produto específicos. Dentro da arquitetura, a tipologia é uma ferramenta crucial para a geração do conjunto de modelos de referência para o processo.

Quadro 4.3 Dimensões da Tipologia de Processos de Desenvolvimento de Produto

NÍVEL	Dimensões		Valores
Mercado	Setor		Automobilístico, aeronáutico, petro-químico, máquinas-ferramenta, eletrodomésticos, linha branca, indústria de base
	Concorrência		Monopólio, oligopólio competitivo, oligopólio diferenciado, concorrência perfeita
	Alvo	Geográfico	local, regional, mundial
		Posição na cadeia de produção	contato com cliente final, intermediário na cadeia de suprimentos
Corporação	Inserção		unidade independente, matriz, filial
	Interação com unidades	Responsabilidade	coordenador de desenvolvimento, participante de co-desenvolvimento
		Equipe	local, regional, mundial
Empresa	Responsabilidade técnica		centro de adaptação tecnológico, aquisição e adaptação de tecnologia, centro de desenvolvimento de produto, centro de desenvolvimento tecnológico de manufatura, centro de manufatura
	Estratégia	Competitiva	custo, qualidade, tempo, misto
		Inter-projetos	novo, simultâneo, seqüencial, posterior
	Informações iniciais		idéia, requisitos de desempenho, parte do projeto do produto, parte do projeto do produto e protótipos, produto em produção
	Complexidade do produto	Tecnologia "core"	mecânica, elétrica, eletrônica, mecatrônica, opto-eletrônica, química
		Interna	número de componentes, números de linhas de softwares, número de insumos e etapas
		Interface com o usuário	alta complexidade, baixa complexidade
	Grau de inovação		pesquisa e desenvolvimento avançados, <i>breakthrough</i> ou radical, plataformas ou nova geração e derivados

4.5.3 Framework de modelagem

O *framework* de modelagem é um elemento de apoio que serve de base para o desenvolvimento de modelos dentro de um padrão comum de representação, os quais são denominados modelos formais dentro da arquitetura. Essa padronização dos modelos deve facilitar a comparação entre eles, permitir a integração dos modelos com os demais conhecimentos explícitos, facilitar a leitura para os usuários que dominam o *framework*, além de facilitar a tarefa de modelagem. Portanto há várias vantagens para o usuário disposto a empregar o *framework* e construir modelos formais, mas eles não são obrigados a fazê-lo. Apenas recomenda-se, fortemente, que os modelos de referência sejam descritos nesse formato.

A seguir, são apresentadas algumas definições básicas, assumidas na proposição do *framework*. Em seguida, são descritos, de forma sucinta, os principais elementos do *framework* de modelagem.

Definições

Os termos arquitetura e *framework*, conforme apresentado na revisão bibliográfica, têm sido empregados com diferentes significados. Assim, para descrevê-lo é importante tornar explícitos os significados atribuídos a cada um desses termos. São eles:

- **Construtos(*Constructs*):** é um elemento primitivo de uma linguagem de modelagem, possuidor de uma semântica bem definida;
- **Formalismos ou métodos de modelagem (*modeling formalisms*):** um formalismo ou método de modelagem é um conjunto de elementos (*constructs* e regras de sintaxe) capaz de representar uma parte da realidade, relativa a um subconjunto do domínio do processo/sistema que está sendo modelado. Um formalismo pode ser aplicado isoladamente para modelar a empresa, e também como parte integrante de um *framework* ou *arquitetura* de modelagem. Os métodos ou formalismos podem ser baseados em três tipos de linguagem: matemática (lógica), natural ou gráfica.
- **Metodologia de Modelagem:** é o conjunto de passos necessários para desenvolver um modelo de empresa, ou seja, construir uma representação de parte ou do todo de uma empresa, baseando-se em um único formalismo ou em um *framework* de modelagem. A metodologia pode fazer parte do *framework* de modelagem;
- **Framework de Modelagem:** é uma coleção de princípios, formalismos de modelagem, ferramentas e metodologias de modelagem, que sejam relevantes para um dado domínio de aplicação da modelagem.

Descrição Geral do Framework

O *framework* de modelagem proposto é baseado na arquitetura de modelagem ARIS. A intenção inicial era utilizar essa arquitetura, mas verificou-se que ela seria demasiado complexa para o objetivo de formalizar os modelos de processo da arquitetura. Portanto decidiu-se criar um *framework* específico para arquitetura, tomando-se como base na arquitetura ARIS. A figura 4.10 mostra os principais elementos do *framework* desenvolvido. Seus elementos principais são:

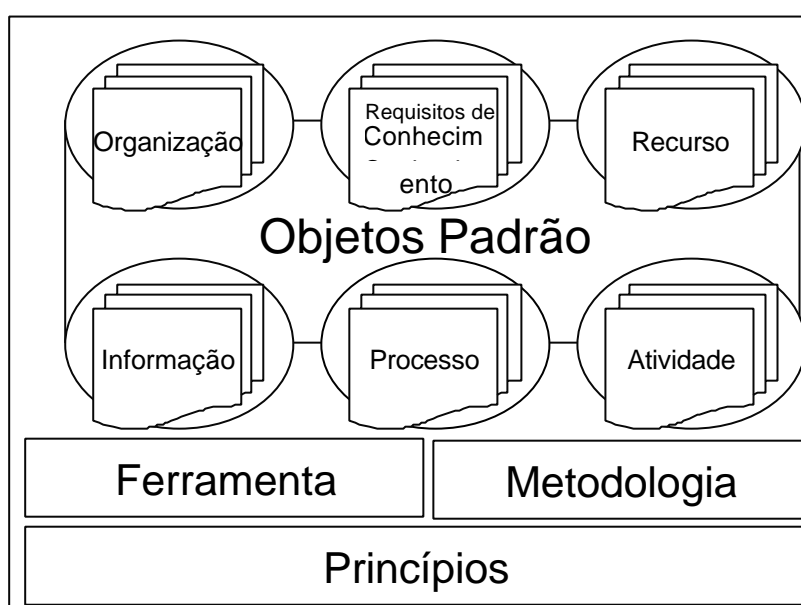

















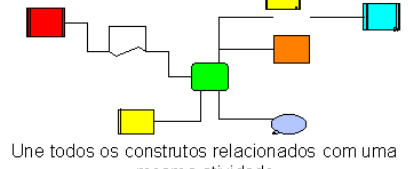
Figura 4. 10 Elementos do framework de modelagem

- **princípios:** como parte do *framework*, foi definido um conjunto de princípios que sustentam a modelagem, que, resumidamente, são: emprego do conceito de processo de negócio; emprego de várias visões; utilização de formalismos ou métodos de modelagem clássicos e bastante difundidos; suporte de ferramenta de modelagem e “portabilidade” dos modelos desenvolvidos, sejam específicos ou de referência;
- **visões:** as visões representam um determinado aspecto do processo de negócio que pode ser visualizado independentemente. Os modelos desenvolvidos segundo o *framework* devem conter seis diferentes visões: Organização; Atividade; Recursos; Informação; Processo e Requisitos de Conhecimento. As quatro

primeiras são similares às clássicas visões de modelagem de empresa empregadas pelo CIMOSA. As duas últimas, Processo e Requisitos de Conhecimento, foram adicionadas para deixar em evidência dois aspectos relevantes para a proposta deste trabalho. A visão de processos é fundamental na comparação de modelos de referência, na descrição do fluxo de atividades do processo e para consultas ou intervenções práticas baseadas no modelo de referência. A visão de Requisitos de Conhecimento é fundamental para mapear os conhecimentos explícitos dentro do processo de negócio e viabilizar a ligação entre modelos de empresa e conhecimentos explícitos. Essa visão permite, também, a descrição das capacitações necessárias para a realização das diversas atividades no decorrer do processo de negócio;

- **Formalismos:** os formalismos empregados em cada visão são apresentados resumidamente no quadro 4.4. Nela, descreve-se, estratificado por visão, cada um dos formalismos incorporados no *framework* e seus construtos. A figura 4.11 apresenta, esquematicamente, como tais construtos se relacionam. Cada visão pode ser descrita por um conjunto pequeno de formalismos. A escolha foi pautada na busca de simplicidade e na utilização de formalismos clássicos. Quanto menor o número de formalismos e construtos, mais simples a compreensão, aprendizado e utilização do *framework*;

Quadro 4. 4: Formalismos e construtos empregados em cada visão

Visão	Formalismo	Construtos	Significado
Organização	Organizational Chart	 Unidade Organizacional	Representa um tipo de unidade organizacional (por exemplo: vendas, marketing ou uma fábrica)
		 Posição	Uma função ou cargo assumido por algum membro da organização
		 Grupo	Um grupo de pessoas tal como um time.
Requisitos de Conhecimento	Knowledge Map	 Knowledge Category	Representa uma área de conhecimento
Recursos	Application System / Application System Type	 Application System Type	Representa uma classe ou um tipo de sistema. Ex.: CAD, PDM, CAPP, etc..
		 Application System	Representa um sistema específico dentro de uma classe. Ex.: AutoCad, etc..
		 Module	Um módulo de um sistema. Ex.: BOM do CAPP,
	eEPC	 General Resource	Qualquer outro tipo de recurso que não seja sistemas (máquinas, equipamentos, etc..)
Informação	Extended Relational Model (eERM)	 Information Carrier	Representa um documento ou outro dispositivo que contenha informações (Etiquetas, etc..)
		 Cluster	Representa a estrutura de dados (complexos) ou repositório de informações. Armazena todos os dados deste repositório.
Atividade	Function Tree	 Activity	Representa uma atividade
Processo	Value-Added Chain (VAC)	 Activity	Representa uma atividade de mais alto nível como uma fase do PDP
	Extended Event Process Chain (eEPC)	 Event	Representa um documento ou outro dispositivo que contenha informações (Etiquetas, etc..)
		 Connectors	Representa uma decisão dentro do Fluxo do Processo, que pode ser "e", "X" ou "ou". No caso de "e" o fluxo segue nos dois carinhos e em "ou" passa para uma delas aleatoriamente
		 Activity	Representa uma atividade
Function Allocation Diagram		Une todos os construtos relacionados com uma mesma atividade	

- **Objetos Padrão:** a arquitetura prevê ainda um conjunto de objetos padrão para cada um dos construtos presentes nos formalismos de modelagem. Um objeto representa uma entidade típica de um processo de negócio de desenvolvimento de produto e é representada por um construto específico. Por exemplo: desenho (informação), detalhar componente (atividade), etc.. Esses objetos devem

abranger todo o domínio da área de desenvolvimento de produto. Com seu emprego, facilita-se a tarefa de modelagem, pois o modelador já tem à disposição um conjunto de objetos “instanciados”, os quais podem ser empregados na modelagem. Como consequência, mantêm-se vocabulários e linguagens padrão para a descrição dos modelos, o que facilita a comparação entre eles. Foram desenvolvidos objetos-padrão apenas para a visão conhecimentos, pois exercem um papel fundamental na integração entre modelos e conhecimentos explícitos;

- **Metodologia de Modelagem:** representa o conjunto de passos que devem ser seguidos para a construção do modelo. A metodologia contém procedimentos específicos para diferentes situações de modelagem, apresentadas no quadro 4.5.

Quadro 4.5 : Situações previstas na metodologia de modelagem

Tipo Básico de Modelo	Situação	Descrição
Referência	Novo formalismo	Projetos onde há um modelo de referência descrito em um formalismo específico e se deseja transcrevê-lo para o framework.
	Totalmente Novo	Quando o modelo de referência está sendo desenvolvido especificamente para a situação sem se basear em nenhum modelo ou processo.
Específico	Melhoria de Processo	Quando se deseja obter um modelo para o processo que seja melhor do que o modelo atual.
	Novo Processo	Quando se trata de um novo modelo para o processo, ou seja, sem que haja um modelo para o processo anterior
	Implantação de Sistemas de Engenharia	Quando deseja-se aprimorar o modelo existente aplicando novas ferramentas de informática.

Desenvolvendo e utilizando modelos conforme o *framework*

Para construir modelos baseados nesse *framework*, basta seguir a metodologia de modelagem, descrita empregando-se o *framework* formal de modelagem. Entre os passos dessa metodologia o usuário deverá classificar seu projeto de modelagem como um dos tipos básicos descritos no quadro 4.5 e partir para a descrição das visões empregando-se os formalismos apresentados no quadro 4.4. A ordem de detalhamento das visões muda conforme o tipo de projeto de modelagem, mas o relacionamento entre os objetos de cada visão é sempre o mesmo, representado, esquematicamente, na figura 4.11. Pode-se observar nessa figura que as visões Informação, Organização, Recursos, Requisitos de conhecimento e

Atividades reúnem, cada qual, objetos específicos. É na visão processo que esses objetos são alocados e organizados, como um processo de desenvolvimento de produto.

Uma vez construído o modelo, os usuários passam a ter acesso às várias visões específicas do processo. Podem, assim, visualizar só a estrutura organizacional necessária para a realização do processo, ou somente os requisitos de conhecimento, as informações relevantes e as atividades e recursos (conforme figura 4.11).

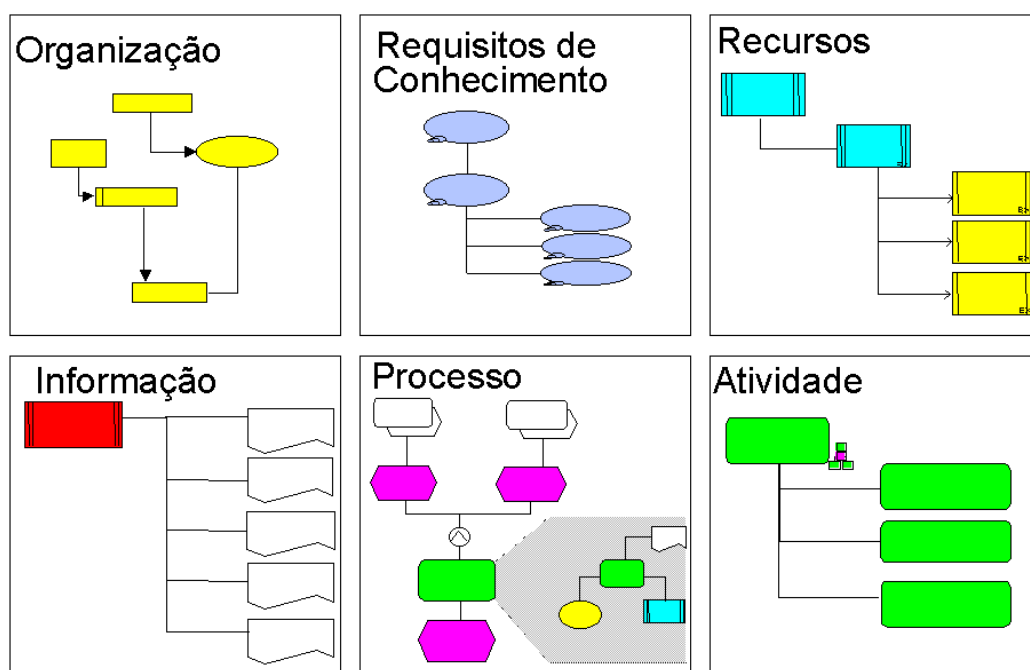
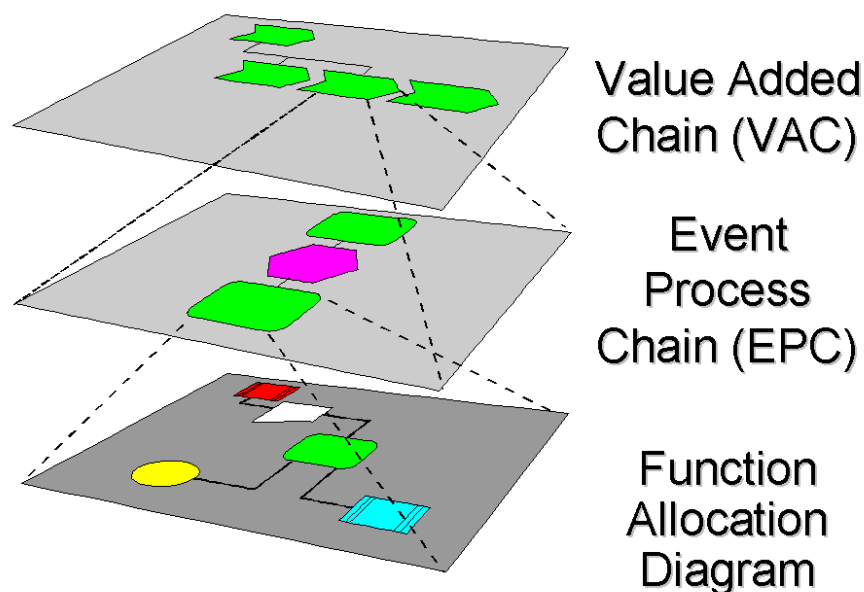


Figura 4. 11 Formalismos de modelagem utilizados em cada uma das visões

A visão por processos contém 3 níveis, conforme demonstra a figura 4.12. Começando com uma visão geral das fases do processo de desenvolvimento de produto descrita no VAC (*Value Added Chain*), entrando em uma das fases específicas no EPC (*Event Process Chain*), até chegar ao nível de detalhe, no *Function Allocation Diagram*, em que são apresentados todos os requisitos necessários para a realização de uma das atividades: informação, recursos, organização e conhecimento.



* objetos conforme o quadro 4.4

Figura 4.12 : Níveis da visão por processo

O formalismo VAC (Value Added Chain) permite representar a visão geral do processo. Conforme representado na figura 4.12, cada uma das fases do processo, representadas no VAC, podem ter seu fluxo descrito por meio do formalismo eEPC (extended Event Process Chain). Por fim, caso interessados nos detalhes de cada uma das atividades, podem acessá-las visualizando as informações que entram e saem desta atividade e quais os recursos, pessoas e conhecimentos são necessários para realizá-la.

4.5.4 Modelo de Perfis de Usuário

Outro elemento constituinte da arquitetura é o modelo de caracterização de perfis de usuários. Esses perfis são úteis para o desenvolvimento do controle de acesso, um item fundamental num sistema de gerenciamento de conhecimentos explícitos, como os que se propõem sejam elaborados, a partir da arquitetura. O controle de acesso deve garantir que os usuários tenham o direito de visualizar toda a informação importante, conforme sua atuação no desenvolvimento de produto e, ao mesmo tempo, impedir a proliferação de “lixo”, ou seja, conhecimentos incorretos, incompletos ou totalmente irrelevantes para a organização.

Apesar de parecer simples, essa não é uma tarefa trivial, pois, há justamente um *“trade-off”*, ou solução de compromisso, entre controle e acesso. Quanto maior o controle e as barreiras para a modificação ou inclusão de um conhecimento explícito, mais se desestimula os usuários a incluí-los. Em sentido contrário, quanto maior o grau de liberdade de acesso, maior é a tendência da proliferação de conhecimentos menos relevantes, incompletos ou incorretos, os quais podem, a longo prazo, inviabilizar o sistema.

Na rotina do dia-a-dia, as atividades e atribuições principais dentro do processo de desenvolvimento de produto são, claramente, prioridades para esses usuários. O registro e armazenamento do conhecimento são geralmente relegados a um segundo plano. Caso as atividades para armazená-los e recuperá-los sejam demasiadamente demoradas ou complexas, haverá uma forte tendência dos usuários deixarem de utilizar o sistema.

No caso da arquitetura proposta, elaborou-se uma definição de perfis capaz de considerar esses dois aspectos do problema. De um lado, procurando evitar ao máximo a “burocracia” para a inclusão dos conhecimentos explícitos e, de outro, permitindo que haja pessoas devidamente capacitadas para filtrar esses conhecimentos e sistematizá-los. Esses especialistas, denominados engenheiros do conhecimento, têm a responsabilidade de eliminar conhecimentos irrelevantes, modificar ou solicitar a alteração de conhecimentos incompletos ou errôneos e de tecer comentários que possam esclarecer o contexto e idéias principais, relativos ao conhecimento explícito armazenado.

Segundo este modelo de perfis de usuário, somente os engenheiros de conhecimento deverão ter acesso ao armazenamento de sentenças, podendo os usuários comuns sugeri-las a esses engenheiros, ficando a cargo deles aceitar ou não a sugestão. Tudo isso para que as sentenças estejam bem estruturadas e sem ambigüidade ou repetição, pois ambas são condições necessárias para que as sentenças “desempenhem” bem seu papel de filtragem e sistematização. Isto é, no caso de surgimento de um emaranhado de sentenças, seu poder de síntese do conhecimento explícito “desestruturado” (aquele armazenado na forma de registros) é eliminado.

Um elemento adicional para este controle de acesso é a classificação de cada conhecimento explícito em dois diferentes níveis: público ou restrito. Os conhecimentos cadastrados como “públicos” poderão ser vistos por qualquer usuário do sistema. Os conhecimentos classificados como “restrito” somente poderão ser acessados pelos engenheiros do conhecimento e tipos de usuário de mais alto nível.

A escolha de dois únicos níveis foi motivada pela busca de simplicidade. Há atualmente, conforme a tecnologia de informática empregada, possibilidade para a inclusão de outros níveis, que podem ser interessantes, dependendo do caso em que a solução a ser derivada da arquitetura será aplicada. Como regra básica, deve-se buscar o menor número de níveis possível, pois uma avaliação errada no cadastro destes níveis torna o conhecimento explícito inacessível e/ou invisível a um usuário que esteja precisando dele. Quanto mais níveis, maior a chance deste erro ocorrer.

O modelo de perfis obtido é representado, esquematicamente, na figura 4.13 (empregando-se o diagrama de atores da UML). Abaixo descreve-se cada um dos diferentes perfis de usuário.

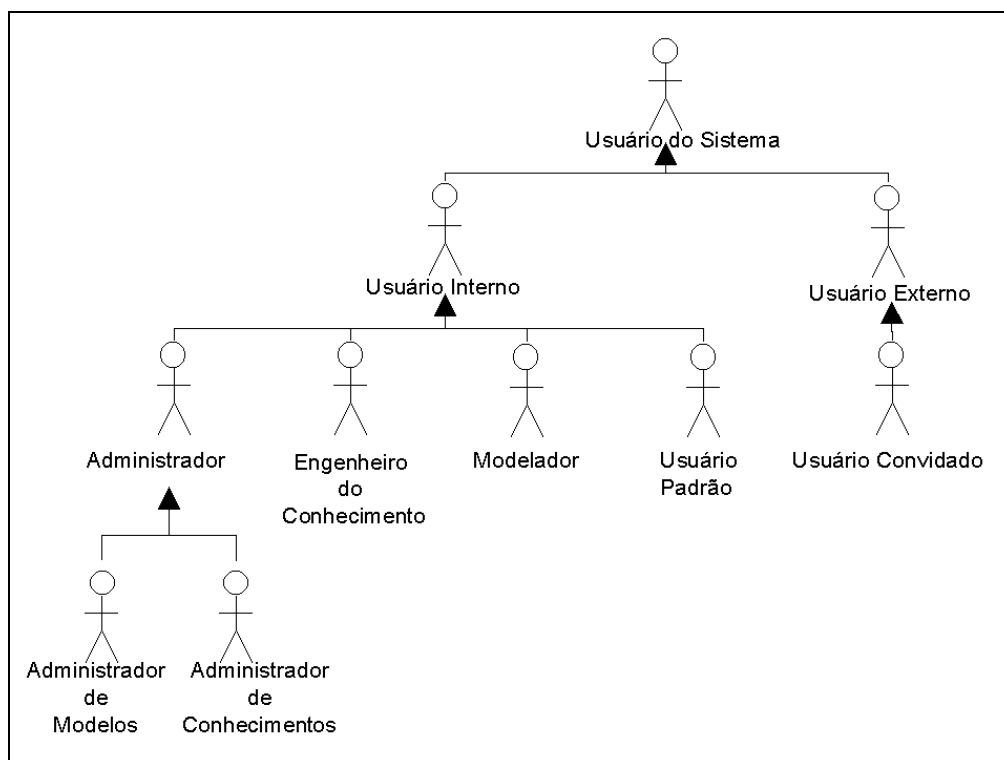


Figura 4. 13 : Representação esquemática do modelo de perfis de usuário

Os usuários são classificados em externos e internos, não tanto pela sua localização geográfica, mas pelo seu envolvimento dentro da organização. Como interno, devem-se incluir todas as pessoas membros que participam diretamente do desenvolvimento de produto, estejam elas localizadas num mesmo espaço físico ou não. Por exemplo, poderia ser classificado com perfil “interno” um funcionário de um fornecedor, um consultor ou especialista, desde que tenham alguma responsabilidade importante no projeto. Deve haver um mínimo de confiança no indivíduo, pois, deve-se estar atento para o fato de que acesso ao sistema significa visualizar a memória organizacional da empresa. Como externos são incluídas pessoas a quem se deseja permitir a visualização do sistema, por alguma razão específica, como demonstrações e apoio em uma atividade ou tarefa específica.

Os usuários internos devem ser classificados em um dos seguintes perfis:

- **administrador:** são os usuários que controlam todo o sistema e devem zelar pelo seu funcionamento, monitorando os conhecimentos armazenados e orientando os demais tipos de usuários quando problemas são identificados. Eles devem ter direito pleno de inclusão, exclusão e modificação de qualquer um dos conhecimentos explícitos. Como é esperada uma grande quantidade de registros, sendo essa tarefa bastante volumosa, foram definidos dois tipos de administradores, delegando-se todo o controle dos modelos a um deles, o administrar de modelos, e as demais tarefas o administrador de conhecimento. Isso porque o monitoramento de modelos exige uma formação específica e conhecimentos sobre *frameworks* e ferramentas de modelagem.
- **administrador de Conhecimento:** os usuários com este perfil devem possuir controle total sobre registros e sentenças e pela manutenção do sistema (melhorias, gerenciamento de *backups* e outros). É também o responsável pelo controle do acesso de usuários, decidindo a inclusão dos novos e a atribuição do perfil adequado a cada um deles. Deve monitorar o trabalho dos engenheiros de conhecimento e demais usuários, sugerindo e cobrando mudanças nos conhecimentos explícitos, e, ainda, identificar quais indivíduos contribuem mais ou menos para o funcionamento do sistema. Deve haver um único administrador de conhecimento ou, em

casos especiais, 2 ou 3. Esse número não deve ser alto para evitar ambigüidades ou atritos na condução da administração do sistema. Lembrando-se que este administrador precisa, mais do que realizar as atividades específicas de manutenção, estar atento também para a motivação dos usuários e avaliação da utilização do seu conteúdo;

- **administrador de Modelo:** é o usuário com controle total sobre a inclusão, exclusão e manutenção dos modelos registrados. Por conseguinte, deve monitorar a atuação dos modeladores garantindo a consistência dos modelos de empresa armazenados. Da mesma maneira que o administrador de conhecimento, deve existir apenas um ou poucos administradores de modelo;
- **engenheiro do conhecimento:** esses usuários são definidos de acordo com sua experiência em uma determinada área do conhecimento, identificada por uma palavra-chave específica. Eles devem ser especialistas naquele determinado assunto e são os responsáveis, dentro do sistema, pela estruturação do conhecimento naquela determinada área. Possuem o direito de incluir, excluir e editar quaisquer novos registros ou sentenças por eles armazenados. Devem ter o direito de adicionar, alterar e apagar registros cuja palavra-chave principal seja a mesma da área de conhecimento sob seu controle. Assim, ele não pode incluir qualquer sentença ou registro, somente aquelas relacionadas ao seu domínio de conhecimento. Deve existir um engenheiro de conhecimento para cada um dos conhecimentos mais importantes (centrais) para a organização. Os engenheiros de conhecimento devem visualizar ambos os conhecimentos, restritos e públicos, sem distinção;
- **usuário Padrão:** deve formar a grande maioria das pessoas cadastradas no sistema. São os usuários que utilizam os conhecimentos explícitos e que registram a grande maioria deles. Devem ter acesso à inclusão de novos registros e à modificação e cancelamento dos registros por eles cadastrados. Devem ter permissão para incluir comentários em qualquer um dos conhecimentos explícitos que lhe esteja disponível e acesso para dar sugestões e opiniões de diferentes tipos: sobre novos usuários para o

administrador de conhecimento, sobre sentenças para os engenheiros de conhecimento e sobre novos modelos para o administrador de modelos e

- **modelador:** os modeladores são usuários-padrão que, além das atribuições desses (descritas anteriormente), têm permissão para incluir novos modelos e para atualizar e apagar os modelos por eles armazenados.

Os usuários externos são classificados no perfil de convidado, recebendo acesso apenas para a visualização dos registros, sentenças e modelos classificados como públicos. Eles não podem adicionar nenhum registro, sentença e modelo e, nem mesmo qualquer comentário. Também não lhes é permitido realizar *downloads* de documentos.

O quadro 4.6 apresenta um resumo das permissões conforme o tipo de usuário.

Quadro 4. 6 Síntese dos diferentes tipos de perfis

Perfil	Tipos de Conhecimentos vistos	Permissões
Administrador de Conhecimento	Restrito/Público	<ul style="list-style-type: none"> • Cadastrar (usuário, registro, sentença, modelo, comentário) • Apagar (usuário, registro, sentença, modelo, comentário) • Modificar (usuário, registro, sentença, modelo, comentário) • Relacionar (registro, sentença, modelo, comentário)
Administrador de Modelo	Restrito/Público	<ul style="list-style-type: none"> • Cadastrar (modelo, registro, sentença e comentário) • Apagar (modelo, registro*, sentença*) • Modificar (modelo, registro*, sentença*) • Relacionar (modelo, registro, sentença)
Engenheiro de Conhecimento	Restrito/Público	<ul style="list-style-type: none"> • Cadastrar (registro, sentença, comentário) • Apagar (registro**, sentença**) • Modificar (registro**, sentença**) • Relacionar (registro**, sentença**)
Usuário-Padrão	Restrito/Público	<ul style="list-style-type: none"> • Cadastrar (registro, comentário) • Apagar (registro*) • Modificar (registro*) • Sugerir (usuário, sentença e modelo)
Modelador	Restrito/Público	<ul style="list-style-type: none"> • Cadastrar (registro, comentário, modelo) • Apagar (registro*, modelo*) • Modificar (registro*, modelo*) • Sugerir (usuário, sentença)
Convidado	Restrito	<ul style="list-style-type: none"> • Visualizar

(*) permissão de realizar somente naqueles conhecimentos explícitos que ele foi o criador

(**) permissão de realizar naqueles conhecimentos explícitos em que foi autor e que têm palavra-chave principal igual ao domínio do conhecimento do perfil

Há um outro item relacionado com o controle de acesso, e que não faz parte integrante deste modelo. Trata-se de proporcionar ao usuário a possibilidade de configurar a interface do sistema conforme suas necessidades, permitindo, dessa maneira, que ele visualize, preferencialmente, a parte do sistema de conhecimentos explícitos que ele utiliza mais intensamente, tanto as novas tecnologias de TI como a Web permitem implementar esta funcionalidade de forma simples. Caso incorporada juntamente com o modelo de perfis de usuário proposto, poderia facilitar ainda mais a utilização do sistema e, sem dúvida, robusteceria o controle de acesso.

4.6 Utilizando a arquitetura

A arquitetura ARCE-PDP deve suportar o desenvolvimento de sistemas de gerenciamento de conhecimentos explícitos. Como foi visto no decorrer deste capítulo, trata-se de um modelo conceitual composto por três principais classes de elementos: uma descrição de como armazenar e classificar os conhecimentos explícitos na forma de um repositório, a especificação de um conjunto de funcionalidades e um conjunto de elementos que orientam como implementar estas funcionalidades. Fazer uso da arquitetura significa, em última instância, assumir as especificações contidas nessas três classes de elementos, como guia para o desenho e construção de soluções para o gerenciamento de conhecimentos explícitos no desenvolvimento de produtos, independentemente das tecnologias de informática empregadas.

Essas três classes de objetos formam um conjunto de especificações de alto nível, cujos detalhes, como, por exemplo, a quantidade de campos para descrever cada tipo de registro e a divisão do conhecimento em classes e subclasses, podem ser definidos conforme a solução. O nível de abstração em que a arquitetura foi especificada torna-a independente da tecnologia de informática, pois, as especificações contidas nos seus elementos podem ser implementadas empregando-se uma ou mais ferramentas e tecnologias de informação. As ferramentas podem ser escolhidas conforme a conveniência da organização, considerando-se as soluções já existentes na empresa e as habilidades e recursos disponíveis.

Este é um ponto forte da proposta, pois permite não somente que a arquitetura possa ser aplicada para uma gama maior de organizações como, também, que seja empregada de diferentes maneiras. Pode-se utilizar a arquitetura com a finalidade de

desenvolvimento de uma nova ferramenta sofisticada, que integre todos os conhecimentos explícitos, ou como uma orientação para a integração dos diversos sistemas de armazenamento de documentos, já disponíveis, e em uso na empresa.

A figura 4.14 mostra esquematicamente exemplos de como diferentes tecnologias poderiam ser empregadas para a construção da ferramenta. No topo dessa figura pode-se observar a representação da arquitetura em si, pela reprodução simbólica de parte da figura 4.1, apresentada no início deste capítulo.

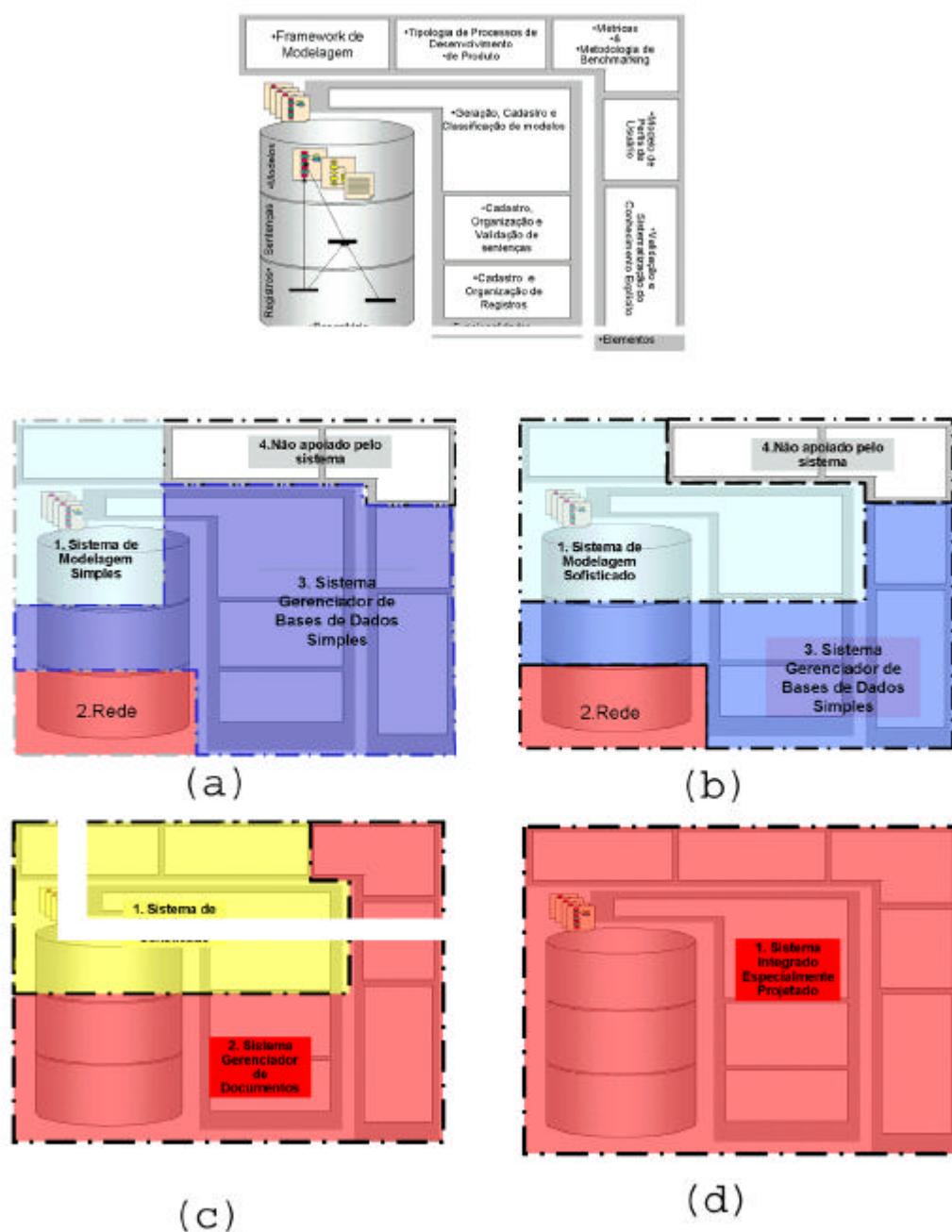


Figura 4. 14 : Representação de diferentes implementações empregando-se a arquitetura

O **item (a)** simboliza uma possível maneira de implementar a arquitetura, empregando-se: uma ferramenta de modelagem simples, isto é, contendo apenas recursos de desenho gráfico (1); uma rede local compartilhada pelos usuários do sistema (2) e um sistema de gerenciamento de base de dados relacional comum (3).

Os modelos seriam gerados pela ferramenta de modelagem (1) e armazenados na rede na forma de arquivos. Os diversos documentos do tipo registro estariam também sendo armazenados na própria rede (2), em um conjunto padronizado de diretórios previamente definido. E, para controlar os cadastros de registros, sentenças e modelos, seria empregado um aplicativo simples, baseado em um sistema gerenciador de base de dados (3). Nesse aplicativo seriam armazenadas as informações sobre cada um dos conhecimentos explícitos (criador, data, etc..), incluindo a descrição de sua localização física na rede. Nesse sistema, ainda que seriam implementadas as sentenças e as principais funcionalidades do sistema: busca, comentários e a de relacionamentos entre registros e sentenças e modelos de referência e demais conhecimentos explícitos. A tipologia de processos e o *framework* de modelagem teriam que ser do conhecimento de cada usuário, ficando sob seu controle a realização.

No **item (b)**, é representada uma configuração semelhante à do item anterior. A principal diferença está no uso de uma ferramenta de modelagem mais sofisticada. Neste caso, não só o desenho, como, também, o gerenciamento dos modelos de processo seriam auxiliados pelo sistema de modelagem (1). Os registros continuariam sendo armazenados na rede (2) e gerenciados pelo sistema de base de dados, na qual seriam registradas e gerenciadas as sentenças.

O **item (c)** representa um caso de aplicação da arquitetura com o auxílio de duas soluções mais sofisticadas: uma ferramenta avançada de modelagem de processos (1) e uma ferramenta de Gerenciamento Eletrônico de Documentos ou Gerenciador de Dados do Produto, GED ou PDM, respectivamente (2). A primeira dessas soluções iria auxiliar o desenvolvimento, gerenciamento e classificação dos modelos de processo(1). A segunda gerenciaria os registros e sentenças, fazendo, também o controle de usuário e assumiria o papel de interface principal com o usuário. Sendo sistemas mais sofisticados, as diversas funcionalidades seriam mais robustas e com níveis maiores de automação e controle de usuário. Além disso, talvez fosse possível implementar uma integração entre ambas as ferramentas, de forma que objetos construídos no sistema de modelagem (1) pudessem ser controlados como documentos pelo sistema gerenciador de documento (2). Conforme o nível de integração, seria possível inclusive atingir o patamar de permitir

ao usuário a sensação de estar manipulando uma única solução integrada, em que todos os conhecimentos explícitos seriam acessados dentro de uma mesma interface, tornando transparente para o usuário qual o sistema que estaria sendo utilizado.

O **item (d)** representa a solução mais radical, na qual seria desenvolvida uma ferramenta específica contendo todas as características especificadas na arquitetura. Essa ferramenta seria construída empregando-se tecnologias básicas de TI, ou seja linguagens de programação e sistemas gerenciadores de bases de dados.

O próximo capítulo descreve em detalhes a construção de uma solução específica seguindo as especificações dessa arquitetura.

5. Desenvolvimento de uma Solução Baseada na Arquitetura

Este capítulo descreve o desenvolvimento e as especificações finais de uma solução para o gerenciamento de conhecimentos explícitos sobre o processo de desenvolvimento de produto, conforme a arquitetura apresentada no capítulo anterior. Inicia com a descrição geral da solução desenvolvida, contendo sua estrutura, componentes principais e o relacionamento entre eles. Em seguida, nos itens 5.2 e 5.3, são descritos os detalhes do funcionamento e desenvolvimento dos elementos principais da solução, respectivamente, a Intranet de Gestão dos Conhecimentos e a Adaptação da Ferramenta de Modelagem.

5.1 Descrição Geral da Solução

O desenvolvimento da solução apresentada neste item teve como objetivo permitir a avaliação da proposta de arquitetura apresentada no item anterior, capítulo 4. A meta foi desenvolver uma solução genérica que pudesse ser aplicada em pequenas e médias organizações que desenvolvessem produtos (conforme exemplificado na figura 5.1). Como consequência, essa solução deveria ser simples o suficiente para permitir seu desenvolvimento, dentro do prazo do projeto de pesquisa, e também facilmente aplicável em diferentes ambientes, de forma a facilitar seu emprego e avaliação na fase de testes. Assim, foram adicionados dois novos requisitos à exigência principal da solução que é, logicamente, a de seguir as especificações da arquitetura apresentados no capítulo 4. Os requisitos adicionais são:

- portabilidade suficiente para a aplicação em diferentes ambientes e
- complexidade e tempo de desenvolvimento compatíveis com a duração do projeto de pesquisa.

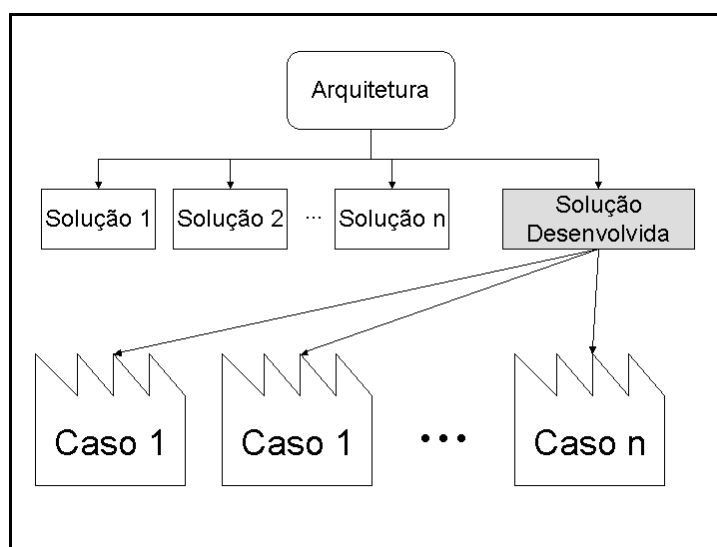


Figura 5. 1 Objetivo da Solução Desenvolvida

Com o intuito de satisfazer estes, objetivos decidiu-se empregar a tecnologia WEB e Banco de Dados Relacional, isto é, desenvolver a solução, empregando-se, principalmente, páginas em HTML (dinâmicas com o emprego da tecnologia de Active Server Pages) e armazenamento dos dados em um Banco de Dados Relacional. Essa seria a maneira de permitir a maior flexibilidade pois não exigiria qualquer ferramenta específica de gestão do conhecimento e/ou gerenciamento de documentos. As exigências para sua aplicação seriam servidores Web e de Base de Dados e uma rede local, contendo computadores dotados de sistemas *browsers*. Mais ainda, caso fosse necessário aplicá-la em uma pequena organização (como uma pequena empresa de alta tecnologia), a solução preparada poderia ser mantida no servidor WEB do laboratório do grupo de pesquisa e ser acessada diretamente dentro da empresa por meio da internet, exigindo-se, para sua aplicação, portanto, apenas computadores com acesso à internet e simples sistemas, do tipo *browsers*, como Netscape ou Internet Explorer, o que permitiria satisfazer o requisito de portabilidade.

Deve-se notar que algumas soluções comerciais de Gestão do Conhecimento (*Knowledge Management*) também poderiam trazer essa última facilidade. Por exemplo, desenvolvendo-se uma solução em um Sistema de Gestão do Conhecimento como Lotus Notes, Lotus KM e as ferramentas da SAP sobre *Knowledge Management*. Adotar este caminho, entretanto, comprometeria o requisito de complexidade, pois o trabalho tornar-se-ia dependente da aquisição

dessas ferramentas, o que envolve um custo de tempo e recursos, em plataformas de *hardware* e *software*, mais elevados e um tempo maior de desenvolvimento a ser consumido no domínio dos diversos aspectos destas ferramentas mais sofisticadas, das quais, inclusive, se tinha pouco conhecimento, naquele estágio de desenvolvimento do projeto. Em oposição a isso, empregar tecnologias básicas de WEB, criando a solução por meio de páginas HTML dinâmicas e Banco de Dados Relacional, exigiria, apenas, ferramentas de desenvolvimento padrões de mercado, mais acessíveis e mais flexíveis.

Uma desvantagem da adoção da tecnologia WEB, de conhecimento dos pesquisadores naquele momento, era a necessidade de se programar todas as funcionalidades necessárias, enquanto as ferramentas mais específicas de Gestão do Conhecimento já disponibilizam muitas delas, bastando complementá-las com integrações e customizações. Assim, no caminho adotado, a robustez e a sofisticação tornam-se proporcionalmente menores, se comparados com a utilização de sistemas comerciais, desenvolvidos e testados durante anos como aqueles citados anteriormente.

Neste mesmo sentido, há funcionalidades especificadas na arquitetura que dificilmente poderiam ser implantadas com o uso exclusivo de tecnologias de programação HTML estático e dinâmico. É o caso, por exemplo, da criação dos modelos de referência. O desenvolvimento de modelos segundo o *framework* de modelagem, especificado no capítulo 4, requer, em termos de funcionalidades, a manipulação de recursos gráficos, em altos níveis de sofisticação e desempenho, o que seria praticamente inviável empregando-se unicamente HTML (mesmo que com tecnologias dinâmicas como ASP), dentro do escopo deste projeto de pesquisa. Assim, para o gerenciamento de modelos de referência, decidiu-se empregar uma ferramenta de modelagem integrada com a solução WEB. Da mesma forma, parte da manipulação de conhecimentos explícitos poderia ser plenamente atendida por meio de padronizações e procedimentos especificados, dentro do ambiente de rede tradicional.

A solução foi desenvolvida baseando-se na integração de três componentes principais, conforme representado nos retângulos da figura 5.2: o ambiente de rede da empresa, a Ferramenta de Modelagem ARIS e uma Intranet especialmente

projetada. O ambiente de rede e a ferramenta de modelagem (adaptada ao framework de modelagem), geram, respectivamente, os registros e modelos de referência (formais). A Intranet, desenvolvida especialmente como parte da solução, permite que estes registros e modelos sejam armazenados, organizados, validados e relacionados pelo usuário. Na intranet, são gerenciados, ainda, os registros, as sentenças e as informações sobre os usuários. Para complementar essa solução, incluem-se padrões, procedimentos e modelos de documentos armazenados de maneira padronizada na rede da empresa, com o intuito de garantir que os procedimentos gerenciados pelo usuário, como o cadastramento de novos modelos e de novos registros, sigam os padrões necessários a fim de facilitar a comunicação e agilizar a criação de novos documentos.

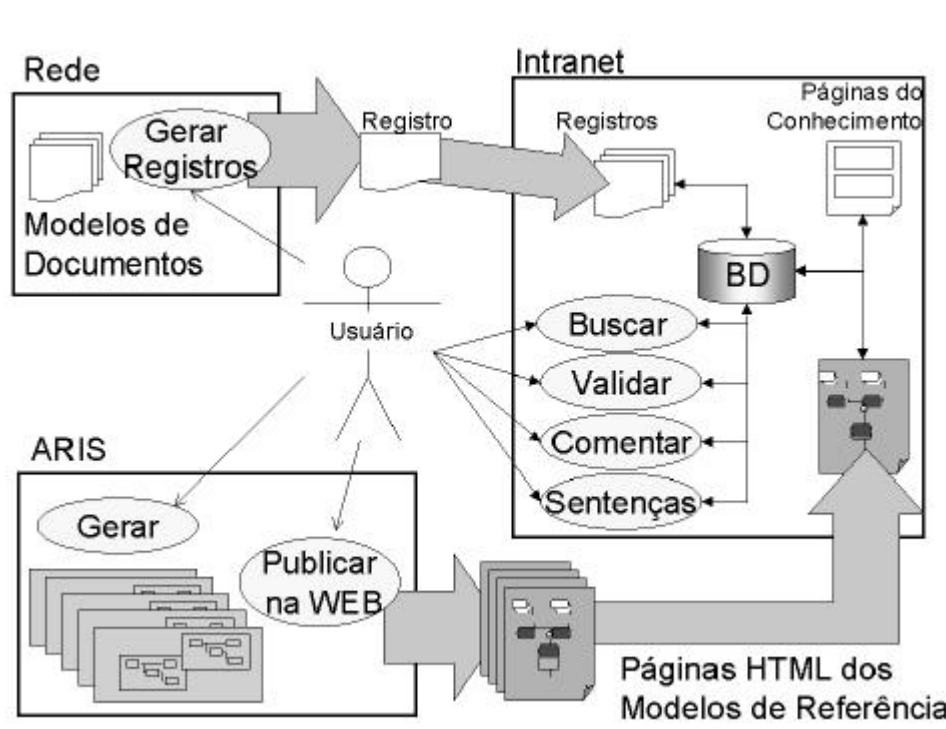


Figura 5. 2: Componentes principais da solução desenvolvida

A solução completa, apresentada na figura acima, foi denominada Sistema de Gestão de Conhecimentos Explícitos sobre o Processo de Desenvolvimento de Produto, e será identificada neste capítulo por SGEN-PDP. Nos próximos dois itens, 5.2 e 5.3, são descritas, respectivamente, os dois principais componentes desta solução: a intranet e a adaptação (“customização”) da ferramenta ARIS.

5.2 Especificação e Descrição da Intranet

Para o desenvolvimento da intranet, procurou-se empregar um processo de desenvolvimento de software estruturado, com o objetivo de garantir a qualidade da ferramenta e gerar, ao final do trabalho, uma documentação rigorosa e clara, permitindo a continuidade das pesquisas futuras, mesmo por outros pesquisadores do grupo.

Foram realizadas buscas na literatura e conversas com especialistas de duas empresas, visando a obtenção de metodologias de desenvolvimento específicas para o ambiente WEB, isto é, que considerassem as peculiaridades desses tipos de sistemas. Quanto à literatura, não foi identificada nenhuma metodologia predominante e de fácil utilização. A de maior evidência, sem dúvida, era a *Unified Modeling Process* (UML). Os especialistas consultados empregavam metodologias próprias, sem muito rigor, principalmente baseadas em prototipação. Isto é, criavam protótipos de telas, empregando sistemas simples como *powerpoint*, ou mesmo diretamente em HTML e gerenciavam as especificações e o processo de software por meio desses protótipos. A decisão final foi a de adotar a UML (Unified Modeling Language). Por várias razões ela parecia mais apropriada: 1) vem se tornando uma linguagem padrão no mundo do *software*, facilitando o entendimento da especificação no futuro; compreende um conjunto robusto e bem documentado de métodos e formalismos e 3) compreende, atualmente, também um processo de desenvolvimento que acompanha a modelagem, denominado de *Unified Process*, que contempla customizações para o desenvolvimento de sistemas para WEB.

Partiu-se para o desenvolvimento da especificação, utilizando-se essa linguagem e o seu processo de desenvolvimento, tendo como base uma ferramenta de modelagem simples denominada VISIO Enterprise 5.0, produto que hoje é comercializado e produzido pela Microsoft. Este caminho foi interrompido, depois de 3 semanas de trabalho intenso, devido à constatação de que, sem uma ferramenta mais poderosa, seria impossível obter uma especificação consistente com essa linguagem de especificação. Dentro da UML o software é especificado em termos de objetos e por meio de um conjunto de diagramas complementares, os quais são divididos em duas categorias básicas: modelos estruturais (que mostram os objetos e seus atributos em si) e modelos dinâmicos (que mostram as interações entre estes

objetos). Ao todo, a especificação do sistema é formada por um conjunto de diagramas, cada um mostrando aspectos particulares do sistema, que, portanto, exigem, para o bom entendimento, uma consistência entre si. Um dos problemas encontrados para o desenvolvimento desta especificação foi o fato da ferramenta empregada, o Microsoft VISIO, não gerenciar estas relações e a consistência entre os objetos. Por isto, uma alteração em um aspecto ou diagrama precisava ser atualizada e confirmada em uma quantidade grande de outros diagramas, tomando tempo e levando a erros e inconsistências. Conforme a especificação do sistema aumentava, esses aspectos foram tornando-se mais e mais críticos até que fosse atingido um ponto onde havia se tornado impossível garantir a consistência da especificação. Nesse momento decidiu-se procurar por uma nova maneira de especificar o sistema.

Baseando-se nesta experiência, no conhecimento sobre programação para a Web do pesquisador e na bibliografia sobre engenharia de software, decidiu-se elaborar um processo simples, contendo fases de desenvolvimento e um conjunto de documentos para a especificação. Esse processo é descrito no item 5.2.2. Os itens 5.2.3 e 5.2.4 descrevem os dois elementos principais da documentação, considerados suficientes para se obter uma boa descrição da estrutura do sistema. São eles: o modelo entidade-relacionamento (5.2.3) e o mapa de telas (5.2.4). Julgou-se que seria difícil entender a lógica da documentação, sem antes apresentar o funcionamento da intranet, portanto, no primeiro subitem, 5.2.1, faz-se uma descrição rápida do funcionamento da intranet, contendo as principais telas e funcionalidades.

5.2.1 Descrição do funcionamento da Intranet

A intranet desenvolvida é dividida em 4 módulos principais:

- **práticas sistematizadas:** esse módulo contém as funcionalidades ligadas aos conhecimentos explícitos dos tipos registro e sentença. Permite gerenciar esses dois tipos de conhecimentos explícitos, oferecendo, portanto, acesso às funcionalidades gerais (listar, cadastrar, comentar, imprimir, editar, apagar) e as específicas das sentenças como a de realizar o relacionamento com registros;
- **modelos de referência:** este módulo contém as funcionalidades para busca, cadastro e edição dos modelos de referência. Conforme o tipo de modelo, pode-se consultá-los diretamente como páginas HTML ou pode-se obter o documento

original, via *download*. Exclusivamente para os casos em que o modelo de referência é formal, isto é, foi descrito segundo o *framework*, é possível acessar os conhecimentos explícitos relacionados com uma determinada atividade do modelo de referência;

- **páginas do conhecimento:** as páginas do conhecimento são páginas especiais que contêm um roteiro de estudo para o usuário iniciar-se em um determinado tema. Elas possuem uma introdução e *links* para os diversos conhecimentos explícitos (registros, sentenças e modelos) sobre esse tema. O módulo permite que um usuário do sistema que seja leigo em um determinado assunto possa ter um roteiro que o oriente no primeiro contato com o assunto. Uma página de conhecimento possui uma estrutura contendo: título, autores, palavras-chave, conceitos básicos e *links* para os demais conhecimentos explícitos (registros, sentenças, modelos de referência e usuários). Contém, ainda, as funcionalidades de cadastro, busca e edição das páginas de conhecimento e
- **pessoas & Conhecimentos:** é o módulo que contém informações sobre o conjunto de usuários cadastrados no sistema, ou seja, a comunidade de pessoas que está compartilhando conhecimentos explícitos, por meio da solução. Nele, pode-se listar pessoas cadastradas e observar informações para contato. Talvez a funcionalidade mais importante desse módulo seja o registro do domínio de conhecimento de cada usuário, isto é, uma declaração de quais conhecimentos e experiências o usuário possui. O domínio é cadastrado e mantido pelo próprio usuário.

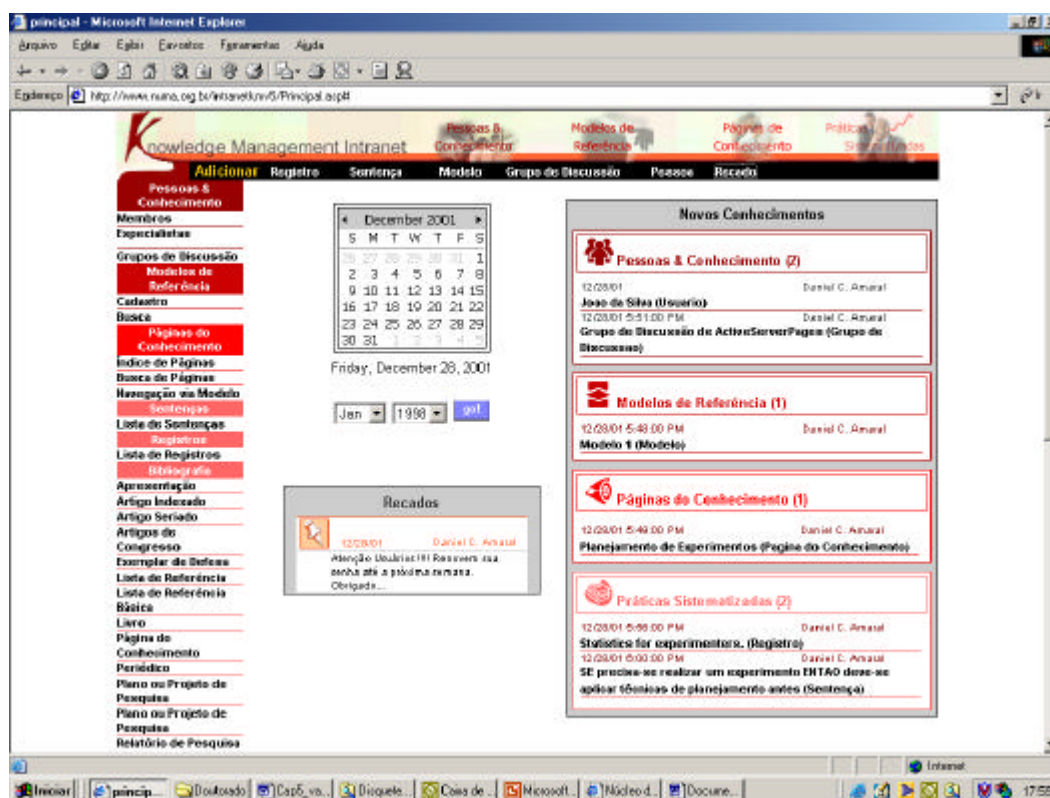
Os conhecimentos armazenados nesses módulos estão relacionados entre si, conforme previsto na arquitetura. Por exemplo, os registros podem ser relacionados com as sentenças e ambos com os modelos. A divisão em módulos tem o intuito apenas de guiar os usuários na busca dos principais tipos de conhecimentos explícitos. Caso o usuário pretenda encontrar informações sobre um modelo, ele pode acessar o módulo específico e efetuar todas as operações relacionadas com esse tipo especial de conhecimento explícito. Da mesma forma, os registros e as sentenças que foram alocados em um mesmo módulo por estarem fortemente relacionados. Os sub-itens abaixo descrevem, em detalhe, cada uma das seções, incluindo figuras das telas do sistema. A descrição se inicia pela página principal. Para facilitar a

localização do usuário dentro do sistema, foi atribuída uma cor específica para cada módulo. A programação visual das páginas segue um mesmo modelo, variando a cor conforme o módulo.

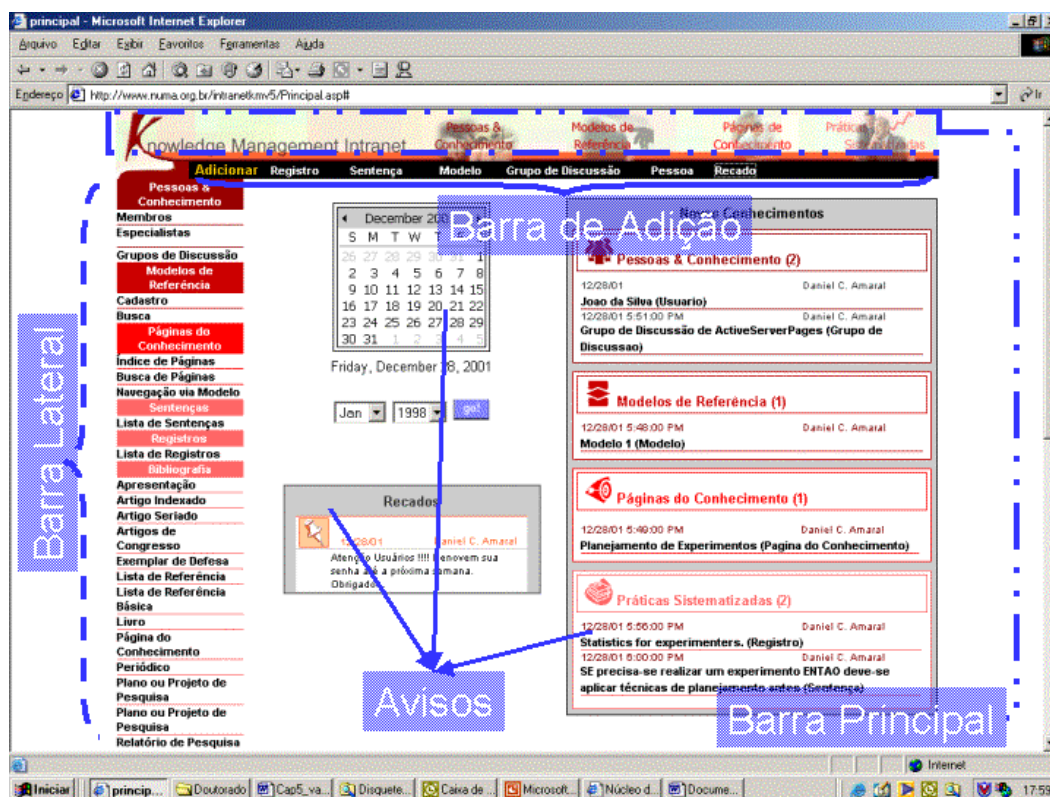
Página Principal

A tela principal pode ser visualizada na figura 5.3a. Essa tela foi projetada de forma a diminuir, ao máximo, a quantidade de “*clicks*” (operações de pressionar o botão esquerdo do *mouse*) necessários para se encontrar uma informação específica. O princípio é de que o *site* (sítio) deve tornar-se uma ferramenta de uso diário para os membros que participam do processo de desenvolvimento de produto da empresa. Priorizou-se uma programação visual que facilitasse a utilização, por parte dos usuários familiarizados com o sistema. Apesar de densamente “carregada”, a página inicial contém o máximo possível de “atalhos”, permitindo que se acesse, diretamente a partir dela, a informação ou operação desejada. Se isso penaliza o usuário “iniciante” com a grande oferta de informações e *links*, em contrapartida, facilita enormemente o uso rotineiro pelo usuário experiente (sendo a última opção privilegiada).

Essa primeira tela é dividida em quatro partes que denominaremos Barra Principal, Barra Lateral, Barra de Adição, Avisos (Calendário, Recados e Novos Conhecimentos), conforme pode ser visto na figura 5.3b.



(a) página principal conforme é apresentada ao usuário



(b) Página principal com identificação das partes principais da tela.

Figura 5.3 : Tela principal do sistema

A Barra Principal contém *links* para os principais módulos do sistema acima descritos. No início de cada um destes módulos, há uma página inicial de busca, a partir das quais pode-se realizar todo tipo de operação relacionada com o determinado tipo de conhecimento explícito armazenado no módulo, como se verá mais adiante.

A Barra Lateral é uma ferramenta de acesso rápido, contendo *links* para pontos específicos de cada uma das seções. Em especial, no caso da seção Práticas Sistematizadas, essa barra contém um *link* para listas de cada uma das classes e subclasses cadastradas no sistema. Ela deverá ser utilizada pelo usuário já familiarizado com a *intranet*, permitindo que ele atinja, em um único *click* uma lista dos conhecimentos de cada uma das subclasses pré-configuradas nesta barra.

Abaixo da Barra Superior encontra-se uma barra negra denominada Barra de Adição, onde o usuário tem à sua disposição “*links*” para inclusão de cada um dos tipos principais de conhecimentos explícitos, isto é: registros, sentenças e modelos. Ele pode cadastrar um novo usuário (*link* pessoa), se devidamente autorizado, e grupos de discussão. O “*link*” recado, presente ao final dessa barra, permite que seja introduzido um recado que ficará armazenado na área de recados que aparece no centro da página. Da primeira página ele pode adicionar qualquer tipo de conhecimento explícito manipulado pelo sistema.

Por fim, a parte Aviso mantém o usuário em contato com as últimas informações adicionadas ao sistema. O quadro “Novos Conhecimentos” apresenta uma lista dos conhecimentos explícitos armazenados no sistema nos últimos 4 dias, separados por módulos. O quadro de “Recados” contém mensagens que outros usuários podem deixar impressas na página por um tempo que varia de 1 dia a 3 meses, sendo o período exato estipulado pelo autor do recado. O calendário mostra os eventos cadastrados no sistema para um dia específico, bastando que o usuário aponte e faça dois “cliques” com o mouse no dia escolhido.

Num futuro próximo, espera-se incluir um mecanismo de busca total nessa primeira página, o qual permitiria encontrar todos os tipos de conhecimentos explícitos contendo uma palavra digitada pelo usuário. Este mecanismo constava da especificação do sistema, mas foi ignorado no protótipo por falta de tempo de programação.

Práticas Sistematizadas

O módulo Práticas Sistematizadas é iniciado com a abertura da tela apresentada na figura 5.4. O corpo desta página contém três mecanismos de busca que permitem ao usuário encontrar todos os registros e sentenças, juntos ou cada um separadamente.

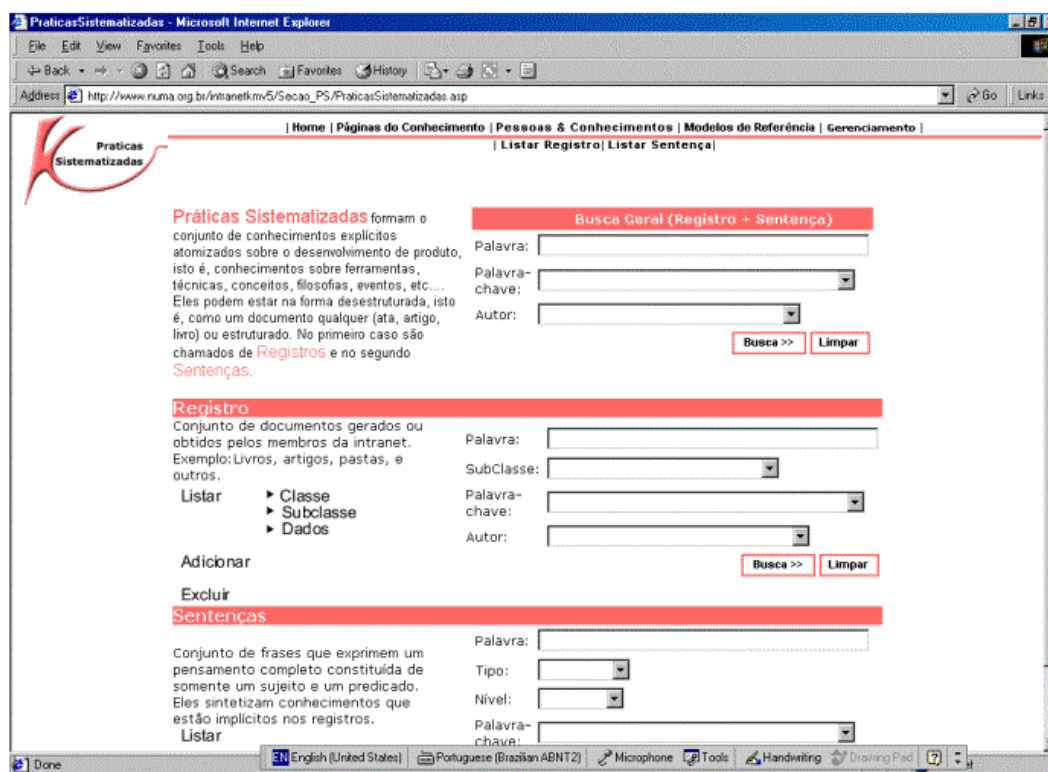


Figura 5.4: Tela inicial do módulo Práticas Sistematizadas

Cada um dos três mecanismos possui campos específicos para a busca. O primeiro deles, que abrange ambos os tipos de conhecimentos explícitos (registros e sentenças), possui 3 campos onde o usuário pode inserir uma palavra qualquer (palavra), uma das palavras-chave cadastradas no sistema ou escolher um dos membros (autor) da intranet. No caso de escolher mais de uma destas opções, a busca será feita baseando-se na intersecção entre esses campos, isto é, buscará registros e/ou sentenças que contenham a palavra digitada, possuam a palavra-chave escolhida e tenham sido cadastrados pelo membro da comunidade identificado no campo autor. No caso de um ou mais destes campos serem deixados em branco, eles serão ignorados pelo mecanismo de busca. O mecanismo específico para registros, que aparece logo abaixo, contém os mesmos campos do anterior, com o acréscimo do

campo Subclasse, que permite ao usuário escolher uma subclasse específica, a ser empregada como “filtro”, durante a busca, isto é, só deverão ser localizados registros oriundos da subclasse escolhida. O último mecanismo, dedicado a buscas de sentenças, repete novamente os mesmos campos gerais do primeiro, havendo o acréscimo de dois campos específicos, que são o tipo e o nível da sentença. O tipo identifica se é um argumento ou enunciado e o nível permite “filtrar” teses ou hipóteses.

Deve-se observar a barra superior da figura 5.4. Ela foi construída dentro de um padrão que acompanha todas as páginas da intranet. É formada pelo “K” que simboliza o sistema onde se lê, ao centro, o título do módulo em que o usuário se encontra. Nesse caso, trata-se do módulo de Práticas Sistematizadas. A linha que cruza a página divide esta barra ao meio formando duas seções de *links*. Acima, podem ser visualizados os *links* para os outros módulos principais da intranet, que, para o caso do módulo de Práticas Sistematizadas, são: Pessoas & Conhecimento, Modelos de Referência, Páginas do Conhecimento e Gerenciamento. Abaixo são apresentados os *links* para as principais páginas do módulo no qual o usuário se encontra. Nesse caso, há *links* para listas de registros e listas de sentenças. Todas as páginas do sistema contêm esta estrutura. Conforme o usuário muda de módulo, são alterados o esquema de cores, o título do módulo no centro do logo e os *links* para os outros módulos. E conforme o usuário se move para uma página na mesma seção alteram-se os *links* abaixo da linha, de forma que ele possa ter, à sua disposição, acesso constante aos principais *links* do módulo que ele está visitando.

Tanto o resultado das buscas como os links da página principal levam o usuário às listas de registros ou sentenças. As listas de registros possuem um mesmo padrão exemplificado na figura 5.5.

Práticas Sistematizadas

| Home | Páginas do Conhecimento | Pessoas & Conhecimentos | Modelos de Referência | Gerenciamento |

| Adicionar Registro | Excluir Registro | Alterar Registro | Listar Registros (Todos, Classe, SubClasse, Dados) |

Registros(Sub) Total de Registros: 88

	Título	Autor	Ano	Subclasse	P. Chave	Dwl	URL	C
<input type="checkbox"/>	Uma proposta de gestão do conhecimento no desenvolvimento de novos produtos	Cristiana Lima da Silva	4/10/01	Artigos de Congresso	Gestão do Conhecimento	S		2
<input type="checkbox"/>	Vivenciando o desenvolvimento de produtos.	Daniel C. Amaral	4/10/01	Apresentação	Desenvolvimento de produto	S		
<input type="checkbox"/>	Definição de formadores	Daniel C. Amaral	4/11/01	Apresentação	Colaboração Cliente-Fornecedor no DP	S		
<input type="checkbox"/>	QFD - Quality function deployment	Fernando Walker	4/11/01	Apresentação	QFD	S		
<input type="checkbox"/>	QS9000	Fernando Walker	4/11/01	Apresentação	QS 9000	S		
<input type="checkbox"/>	Ciclo de desenvolvimento de produtos na VW - caminhões e ônibus	Fernando Walker	4/11/01	Apresentação	Desenvolvimento de produto	S		
<input type="checkbox"/>	Business Process Management	Henrique Rozenfeld	4/12/01	Livro	Business Modeling	S		2
<input type="checkbox"/>	computer world de 1999 até março de 2001	Henrique Rozenfeld	4/12/01	Periódico	general	S		1
<input type="checkbox"/>	Cadastro Notebooks Grupo EI	Henrique Rozenfeld	4/12/01	Cadastro	general	S		
<input type="checkbox"/>	O futuro do desenvolvimento de produto no Brasil (GM)	Fernando Walker	4/12/01	Apresentação	Desenvolvimento de produto	S		
<input type="checkbox"/>	Tendência da engenharia na FIAT do Brasil (FIAT)	Fernando Walker	4/12/01	Apresentação	Desenvolvimento de produto	S		
<input type="checkbox"/>	Trabalhando com o conhecimento no desenvolvimento de produtos: uma proposta de sistematização	Usuarios	4/12/01	Artigos de Congresso	Gestão do Conhecimento	S		
<input type="checkbox"/>	Trabalhando com o conhecimento no DP: uma proposta de sistematização	Daniel C. Amaral	4/12/01	Artigos de Congresso	Gestão do Conhecimento	S		
<input type="checkbox"/>	A knowledge framework for product development	Daniel C. Amaral	4/12/01	Artigos de Congresso	Gestão do Conhecimento	S		

Figura 5. 5: Exemplo de uma lista de registros no módulo Práticas Sistematizadas

Nas listas de registros encontra-se uma tabela contendo o título, autor, ano, subclasse e palavra-chave primária. As três últimas colunas indicam a presença de arquivos para *download* (dwl), *links* para sites (URL) e comentários (C). Sempre que aparecer “S” (de Sim) em em uma destas colunas, há um arquivo para download ou um endereço de site cadastrado. Basta o usuário “clique” na letra “S” para realizar o *download* do arquivo ou visualizar a página na internet. No caso da coluna “C”, de comentários, aparece um número que indica a quantidade de comentários cadastrados sobre àquele determinado registro. Este número é também um *link* que, uma vez acessado, abrirá uma página que exhibe estes comentários. A figura 5.6 mostra o exemplo de uma página de comentários na qual se lê, na parte superior, o nome do registro seguido da expressão “Comentário de...” . No caso dessa figura, são apresentados dois (2) comentários para o registro de título “Business Process Management”.

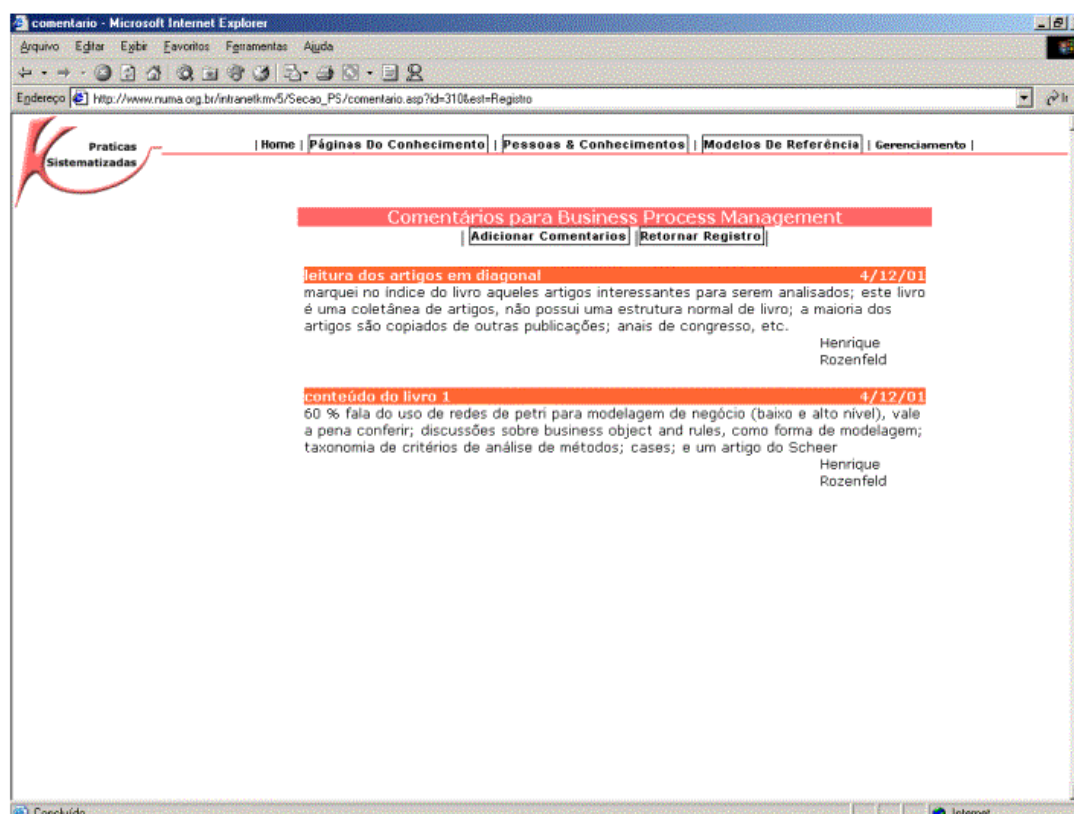


Figura 5. 6: Página de comentários de Registros

Uma outra característica das listas de registro, semelhantes à da figura 5.5, é o conjunto de mecanismos de ordenação adicionados aos títulos das colunas. Basta que o usuário “clique” sobre um dos títulos para que esta lista seja reconstruída, de maneira a conter os mesmos registros, classificados conforme o conteúdo da coluna escolhida. Isto é, acionando-se o link da coluna “título”, será apresentada a mesma lista em ordem alfabética crescente de títulos. Igualmente, para autor, ano, subclasse e palavra-chave.

Pode-se observar ainda na lista de registros (figura 5.5 e 5.7a) um conjunto de “quadrados” ou, de forma mais específica, de controles do tipo *check-box*. Os usuários podem marcar um ou mais registros da lista, que sejam do seu interesse (ver figura 5.7a) e, acionando o botão “Registros Completos”, que aparece em azul abaixo da lista, acessar o conjunto completo de atributos dos registros na página denominada “Detalhes de Registros”, conforme figura 5.7b.

Práticas Sistematizadas

| Home | Páginas do Conhecimento | Pessoas & Conhecimentos | Modelos de Referência | Gerenciamento |
| Adicionar Registro | Excluir Registro | Alterar Registro | Listar Registros (Todos, Classe, SubClasse, Dados) |

Livro(Sub) Total de Registros: 7

	Título	Autor	Ano	Subclasse	P. Chave	Dwl	URL	C
<input type="checkbox"/>	Business Process Management	Henrique Rozenfeld	4/12/01	Livro	Business Modeling	S		2
<input type="checkbox"/>	PMBOK - Project management body of knowledge (versão português)	Cristiane Lima da Silva	4/12/01	Livro	Gestão do Conhecimento	S		
<input checked="" type="checkbox"/>	PMBOK - Project management body of knowledge - versão inglês	Cristiane Lima da Silva	4/13/01	Livro	Gestão do Conhecimento	S		
<input type="checkbox"/>	QFD: planejamento da Função Qualidade	Daniel C. Amaral	5/28/01	Livro	QFD			
<input type="checkbox"/>	Introdução ao desdobramento da Função Qualidade	Daniel C. Amaral	5/28/01	Livro	QFD			
<input checked="" type="checkbox"/>	Total quality development	Daniel C. Amaral	9/19/01	Livro	Desenvolvimento de produto	S		1
<input type="checkbox"/>	Statistics for experimenters.	Daniel C. Amaral	12/28/01	Livro	DOE	S		

[Dados Completos](#)

(a) lista de registros do tipo livro com 2 registros selecionados

Práticas Sistematizadas

| Home | Páginas do Conhecimento | Pessoas & Conhecimentos | Modelos de Referência | Gerenciamento |
| Adicionar Registro | Excluir Registro | Alterar Registro | Listar Registros (Todos, Classe, SubClasse, Dados) |

Detalhes de Registros

PMBOK - Project management body of knowledge - versão inglês

Data: 4/13/01

Autor: Cristiane Lima da Silva

Título: PMBOK - Project management body of knowledge - versão inglês

URL: Não possui

Download: [Clique Aqui](#)

Palavra-Chave: Gestão do Conhecimento

Restrição: 2

Referência Completa: A guide to the project management body of knowledge PMBOK Guide, 2000 Edition

Localização:

Total quality development

Data: 9/19/01

Autor: Daniel C. Amaral

Título: Total quality development

URL: Não possui

Download: [Clique Aqui](#)

Palavra-Chave: Desenvolvimento de produto

Restrição: 2

Referência Completa: Clausling, D. Editora: ASME, 1994

Localização: BC_EESC/

(b) página de detalhes dos registros selecionados

Figura 5. 7: Exemplo de página de detalhes de registro

De forma semelhante, porém mais simples, é possível gerar listas de sentenças. A figura 5.8 apresenta uma listagem de sentenças, contendo 5 sentenças. São apresentados o título, o tipo (enunciado ou argumento), o grau de validação e uma coluna com os *links* para os detalhes de cada uma das sentenças. Uma vez acessado o *link* dos detalhes, o usuário abrirá uma tela semelhante à figura 5.9, que é a “Ficha da Sentença”.

Título	Tipo	Grau de Validação	Detalhes
A primeira casa do QFD é a mais utilizada	Enunciado	0	Clique aqui
SE A interface do produto com o usuário é complexa ENTÃO deve-se aplicar o QFD	Enunciado	2.5	Clique aqui
SE QFD é empregado utilizando-se o modelo conceitual ENTÃO obtêm-se os melhores resultados	Enunciado	1.0	Clique aqui
SE a interface do produto com o usuário é complexa ENTÃO deve-se empregar a primeira casa da qualidade	Argumento	0.6	Clique aqui
		0.7	Clique aqui
		0	Clique aqui
		0	Clique aqui
		0	Clique aqui
		0	Clique aqui
		0	Clique aqui
		0	Clique aqui
		0	Clique aqui
SE precisa-se realizar um experimento ENTÃO deve-se aplicar técnicas de planejamento antes	Argumento	0.7	Clique aqui

Figura 5. 8: Exemplo de lista de sentença

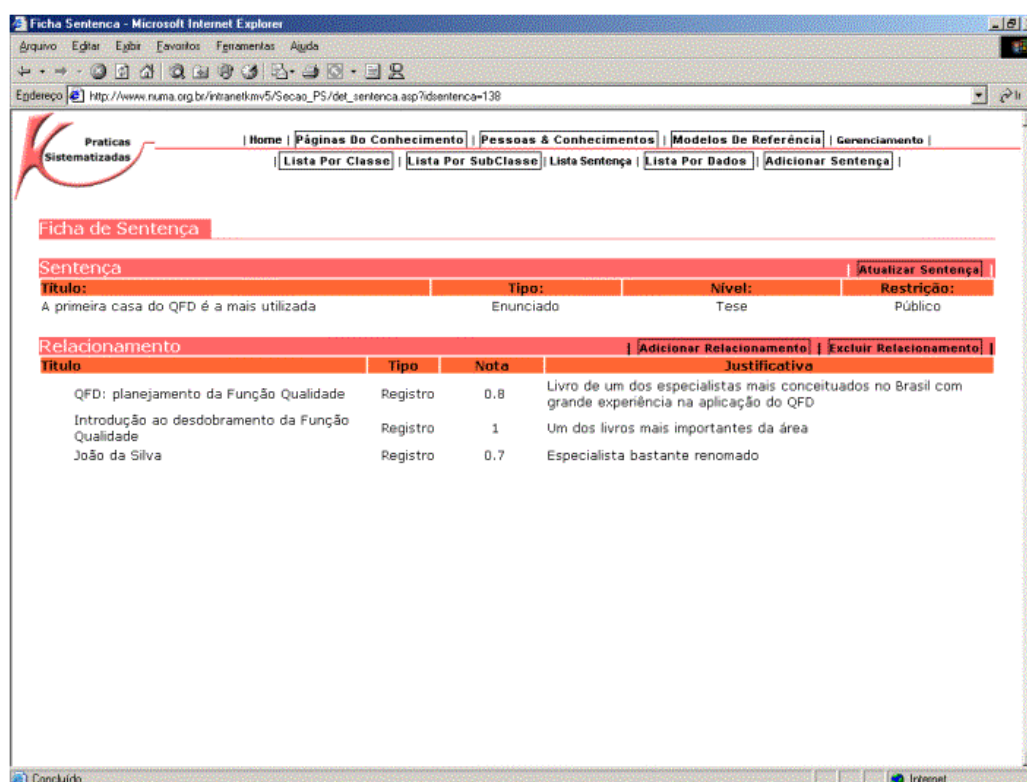


Figura 5. 9: Exemplo de tela de detalhes da sentença

As Fichas de Sentença (Figura 5.9) são divididas em duas partes principais. Na parte superior, são apresentados os dados referentes à sentença: o título, tipo (se é enunciado ou argumento), nível (se é tese ou hipótese) e a restrição (se é pública ou restrita). Na parte inferior, são apresentados os relacionamentos da sentença com outras sentenças ou registros. No caso específico da sentença apresentada na figura 5.9, trata-se de uma sentença do tipo enunciado (isto é, obtida diretamente de registros), no nível de tese e cujo título é “A primeira casa do QFD é a mais utilizada”. Ela é suportada por 3 registros com graus de validação 0,8, 0,7 e 1, todos positivos. Ao lado do grau de validação, aparece a justificativa inserida pelo engenheiro de conhecimento. Esta mesma tela contém os *links* para a edição desta sentença. O primeiro deles, “Atualizar Sentença”, pode ser visto no lado direito do título da primeira parte da tela e permite que sejam alterados os atributos referentes especificamente à sentença, isto é, o título e as classificações de tipo e nível. Na parte de baixo, há o link que permite a edição do relacionamento da sentença com os demais registros e sentenças. São os *links* “Adicionar Relacionamento” e “Excluir Relacionamento”.

É importante notar, neste momento, que todo o cadastro e as atualizações nas diversas seções da intranet são realizadas pelo usuário com telas específicas da intranet. Essas telas não estão sendo apresentadas por se tratarem de cadastros comuns, semelhantes aos encontrados na maioria dos sistemas web, com caixas de controle e listas; sendo, portanto, de menor significância para a descrição das especificidades deste sistema. Deve-se fazer uma exceção para o cadastramento de sentenças, o qual exigiu um formulário mais sofisticado para garantir a facilidade de uso. O intuito era facilitar e permitir o acesso do usuário aos vários registros e sentenças cadastrados no sistema, ao mesmo tempo em que ele está adicionando os dados referentes à nova sentença. Um exemplo desta página pode ser observado na figura 5.10.

The screenshot shows the 'Cadastro de Sentença' page in a Microsoft Internet Explorer browser. The page has a navigation bar with links like 'Home', 'Páginas do Conhecimento', 'Pessoas & Conhecimentos', 'Modelos de Referência', and 'Gerenciamento'. Below the navigation bar, there's a red header for 'Cadastro de Sentença'. The main form includes fields for 'SE' (containing 'A interface do produto com o usuário é complexa'), 'ENTÃO' (containing 'deve-se aplicar o QFD'), 'Tipo' (dropdown menu with 'Argumento'), 'Nível' (dropdown menu with 'Hipótese'), and 'Restrição' (dropdown menu with 'Público'). Below the form is a table with columns 'Título', 'Tipo', 'Nota', and 'Justificativa'. The table contains three rows of data. At the bottom, there's a search section with 'Busca Palavra-Chave' (containing 'QFD') and 'Usuário' (dropdown menu with 'Daniel C. Amorim'). To the right of the search section is a 'Resultado de Busca' table with columns 'Registro', 'Título', and 'Subclasse'. Below that is another table for 'Sentença' with columns 'Título' and 'Tipo'.

Título	Tipo	Nota	Justificativa
QFD: planejamento da Função Qualidade	Registro	1	Livro clássico na área com muitos casos e comprovações.
Introdução ao desdobramento da Função Qualidade	Registro	0.8	Livro clássico na área.
João da Silva	Registro	0.7	Especialista com muita experiência (7 anos) e aplicação em

Registro	Título	Subclasse
Ver	Aplicação de QFD na empresa X	Caso
Ver	Aplicação de QFD no desenvolvimento de um motor da Y	Artigo Seriado
Ver	QFD: planejamento da Função Qualidade	Livro
Ver	Introdução ao desdobramento da Função Qualidade	Livro
Ver	João da Silva	Especialista

Sentença	Título	Tipo
Ver	A primeira casa do QFD é a mais utilizada	Enunciado

Figura 5. 10: Exemplo de tela para o cadastro das sentenças

A tela de cadastro de sentenças é dividida em 4 grandes segmentos. No primeiro deles, logo abaixo ao título da página “Cadastro de Sentença”, encontram-se os campos relativos à sentença em si, isto é, título, tipo, nível e restrição. Na seção logo abaixo, o usuário observa uma lista dos registros e sentenças já relacionados com esta sentença. Para cada um deles, há um controle do tipo lista, em que o usuário determina o grau de validação desse registro e/ou sentença, em relação à

sentença que está sendo cadastrada. Ao lado desse controle, há espaço para que seja digitado um texto curto em que o usuário justifica a escolha do grau de validação.

Nos dois últimos segmentos desta tela, emparelhados na parte inferior da página, o usuário tem, à sua disposição, um mecanismo de busca que ele pode consultar no momento que desejar, enquanto está cadastrando a nova sentença. São duas partes complementares, sendo a da esquerda denominada “Busca” e a da direita, “Resultado da Busca”. O usuário pode escolher palavras-chave e nomes de usuários nas listas contidas na seção de buscas e, ao “clicar” no botão “busca”, deverá surgir uma lista com os registros e sentenças encontrados no segmento ao lado direito, “Resultado de Buscas”. Para cada um dos registros listados são apresentados o título e dois botões ao lado: “ver” e “+”. Ao passar o mouse e clicar sobre o botão ver, será aberta uma nova janela do *browser*, com dimensões reduzidas, apresentando todos os detalhes do registro ou sentença. Com isso, ele pode decidir se, realmente, este é o registro ou a sentença que ele deseja relacionar com a nova sentença. Em caso afirmativo, basta que ele “clique” com o *mouse* sobre o botão “+” para adicionar o relacionamento. Feito isso a tela é reconstruída e o novo relacionamento surge no segmento acima da página, ou seja, uma nova linha será inserida no segmento de relacionamentos. A diferença das demais é que o grau de validação será 0 e o campo justificativa estará vazio. Basta o usuário preencher tanto o grau de validação quanto a explicação, para que a sentença esteja pronta para ser submetida.

O fim da operação de cadastramento se dá quando o usuário efetua dois “cliques” sobre o botão “submeter”, que aparece na parte debaixo do segmento que contém os relacionamentos. Somente depois desta operação, os dados são cadastrados na base de dados. Portanto se houver algum problema durante o cadastramento, nenhum registro incompleto será criado na base. Se o usuário clicar “Redefinir”, todos os relacionamentos já estabelecidos serão apagados da tela, permitindo que o usuário recomece a definição do ponto inicial. Por fim, caso ainda o usuário deseje, antes de criar a nova sentença, eliminar um dos relacionamentos estabelecidos, basta “clicar” no botão com sinal “-” ao lado do relacionamento. Isto fará com que apenas o relacionamento escolhido seja eliminado, mantendo os demais intactos.

Modelos de Referência

O módulo de modelos de referência começa com uma lista com todos os modelos cadastrados, na qual se pode observar: o título do modelo; o tipo que identifica o formato em que se encontra (se é Excel, ARIS, word, PDF ou outro); o autor; a data de criação do modelo; a versão e a coluna que identifica se há algum documento disponível sobre o modelo, a coluna doc. Um exemplo é apresentado na figura 5.11.

Título	Tipo	Formato	Autor	Data	Versao	doc
Cenário - PLM	Específico	Excel	Grupo EI	1/5/01	1.0	
Cenário - PLM	Específico	Excel	Grupo EI	1/5/01	1.0	
Cenário - PLM	Específico	Excel	Grupo EI	1/5/01	1.0	
Empresa X	Específico	Outro	Grupo EI	10/1/01	1.0	S
Metodologia	Referência	ARIS(Web)	Grupo EI	1/22/01	1.0	S
Metodologia	Referência	ARIS (Relatório)	Modelo do processo de modelagem	1/20/01	3.0	S
Metodologia	Referência	Excel	Grupo EI	1/20/01	3.0	S
Metodologia	Referência	Excel	Grupo EI	1/20/01	4.0	S
New Product	Referência	ABC-Flow	Grupo EI	10/1/01	3.0	S
New Product	Referência	Excel	Grupo EI	10/1/01	3.1	S
New Product	Referência	Excel	Grupo EI	10/1/01	3.2	S
New Product	Referência	ARIS (Relatório)	Grupo EI	1/20/01	3.2	S
New Product	Referência	Excel	Xup	10/1/01	4.0	S
New Product	Referência	Excel	Grupo EI	1/20/01	4.1	S
New Product	Referência	ARIS(Web)	Grupo EI	10/10/01	4.1	S
teste	Específico	Excel	tet	1/9/1900	te	

Figura 5.11 : Página inicial do módulo Modelos de Referência

A última coluna, de título doc, conterá o valor “S” (de sim), caso exista uma versão do documento disponível para navegação ou *download*. Portanto “clcando” sobre o “S” o usuário poderá abrir e “navegar” nos modelos em html ou, no caso de documentos em outros formatos, realizar um “*download*” do documento para o seu computador e consultá-lo. Um exemplo de modelo cadastrado em HTML é apresentado na figura 5.12.

Modelo de Referência

| Home | Páginas do Conhecimento | Pessoas & Conhecimentos | Práticas Sistematizadas | Gerenciamento |

Lista de Modelos | Cadastrar Modelos | Retorna Modelo |

Database: Metodologia

Visão Geral do Modelo:

Modelar Processo de Desenvolvimento de Produto

Definir Projeto | Planejar Projeto | Construir Modelos | Implementar Modelo

Fases	
	Definir projeto
	Planejar projeto
	Construir Modelos
	Implementar Modelo

Visão	Modelos
	<ul style="list-style-type: none"> Análise do uso do Modelo de Referência AutoAvaliação do time

Figura 5. 12: Exemplo do resultado da consulta de um *link* “doc” para um modelo de referência armazenado em HTML.

O modelo da figura 5.12 foi desenvolvido dentro do framework de modelagem padrão. A primeira página mostra o modelo VAC (*Value Added Chain*) e, abaixo, duas tabelas: a primeira delas contendo links para cada uma das fases do processo e a segunda contendo links para modelos e objetos de cada uma das seis visões do *framework* de modelagem. Esse modelo foi descrito na ferramenta ARIS Toolset e as páginas HTML foram geradas para dentro da intranet, conforme apresentado no item 5.3 deste capítulo.

Ainda na página inicial da seção (figura 5.11), ao clicar sobre o título, o usuário poderá visualizar uma ficha, contendo uma descrição mais detalhada do modelo, conforme a figura 5.13. A ficha contém também um link para acesso ao documento, e, adicionalmente, há acesso à classificação do modelo, segundo os critérios especificados na Tipologia de Processos de Desenvolvimento de Produto. Basta “clique” no link da palavra “AQUI”, para que aparece na frente do atributo “critério”. Ao fazer isso, o usuário terá acesso a uma página semelhante à da figura 5.14, a qual mostra um conjunto de critérios especificados para um determinado

modelo denominado “PDP-Linha Branca”. Os critérios são adicionados um a um pelo usuário, que cadastra o modelo, de forma a não ser obrigatória a classificação do modelo, segundo “todos” os critérios da tipologia.

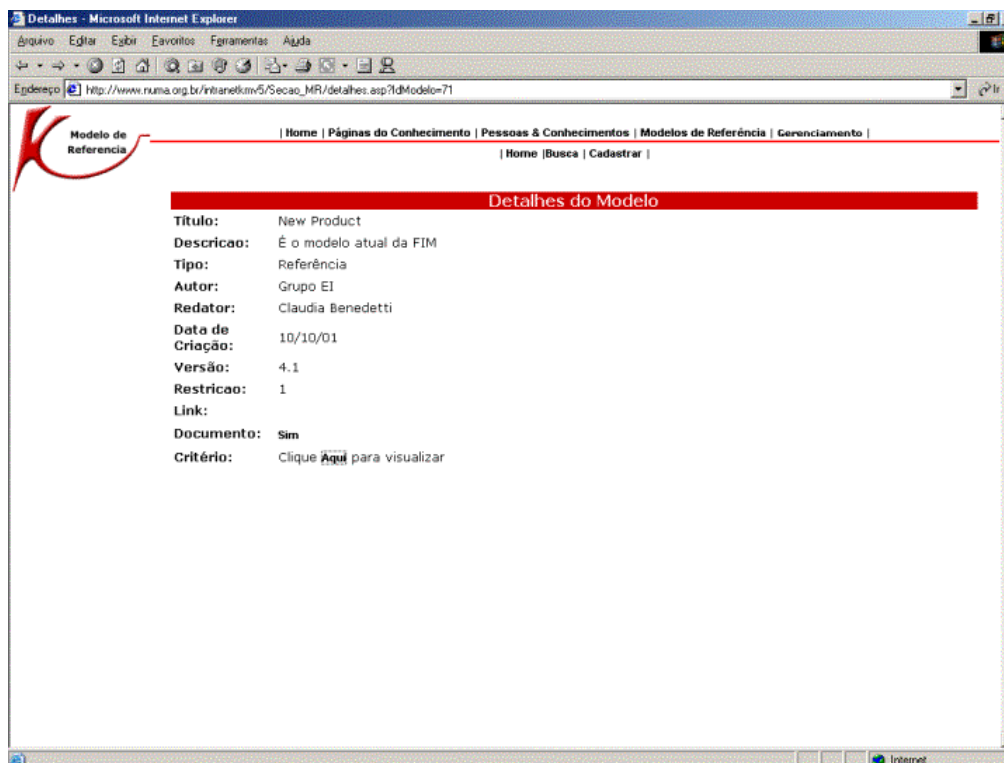


Figura 5. 13: Exemplo de tela contendo uma ficha de modelo de referência

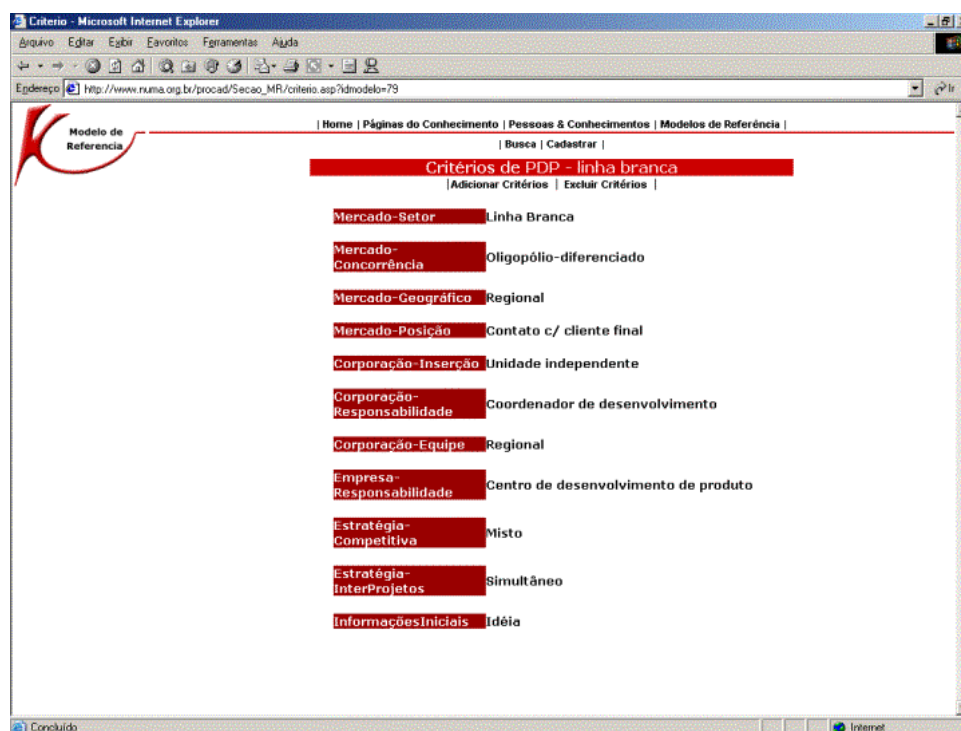


Figura 5. 14: Exemplo de classificação de um modelo segundo a tipologia de processos de desenvolvimento de produto

Páginas do Conhecimento

O módulo de página de conhecimentos é iniciado com uma lista das páginas existentes, conforme pode ser observado na figura 5.15. As páginas são listadas em agrupamentos, segundo 3 tipos básicos: conceitos/filosofias; técnicas/métodos; e ferramentas/sistemas. Os conceitos/filosofias são os conhecimentos essencialmente formados por princípios e definições, que podem orientar o usuário na condução das atividades de desenvolvimento de produto, como, por exemplo, Engenharia Simultânea. Como técnicas/métodos, classificam-se os conhecimentos que se apresentam na forma de um conjunto de passos visando a realização de determinada tarefa ou obtenção de um objetivo. É o caso da metodologia de QFD (*Quality Function Deployment*), FMEA (*Failure Mode and Effect Analysis*), entre outras. Por fim, classificam-se como ferramentas/sistemas os conhecimentos sobre dispositivos que auxiliam uma determinada atividade e que realizam uma função específica. É o caso dos sistemas CAD (Computer Aided Design). No caso do exemplo apresentado na figura 5.15, há apenas uma página do conhecimento cadastrada que versa sobre a técnica do FMEA acima citada.

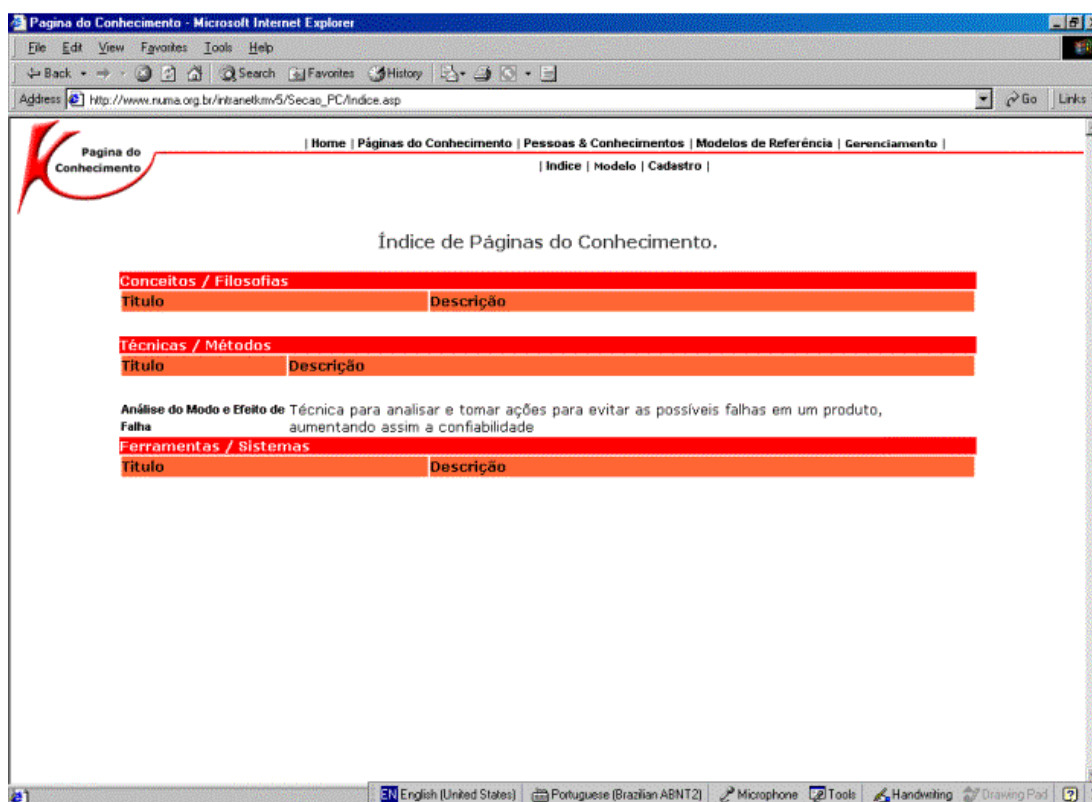


Figura 5. 15: Tela inicial do módulo Páginas do Conhecimento

Ao escolher uma das páginas listadas, “clizando” sobre o título, o usuário tem acesso a uma página semelhante à apresentada na figura 5.16. Nela, pode-se ler o título da página, os autores e observar os *links* contendo os diversos conhecimentos explícitos relacionados àquele assunto: registros, modelos, sentenças e grupos de discussão. No centro da página há um *link* ao lado da palavra “visualizar” onde o usuário tem acesso ao corpo principal da página de conhecimento, que aparece em uma nova tela semelhante à que figura no canto inferior direito da figura 5.16. Essa janela reduzida contém o corpo da página de conhecimento, no qual são apresentados os textos, contendo a definição do conhecimento e uma introdução ao assunto, tais como conceitos-chave e dicas para o usuário iniciante introduzir-se no tema. O conteúdo dessas seções pode variar conforme as necessidades dos diferentes tipos de conhecimentos, mas 3 tópicos são obrigatórios: conceitos básicos, definições e roteiro de estudo. A figura 5.16 apresenta o primeiro deles, conceitos básicos.

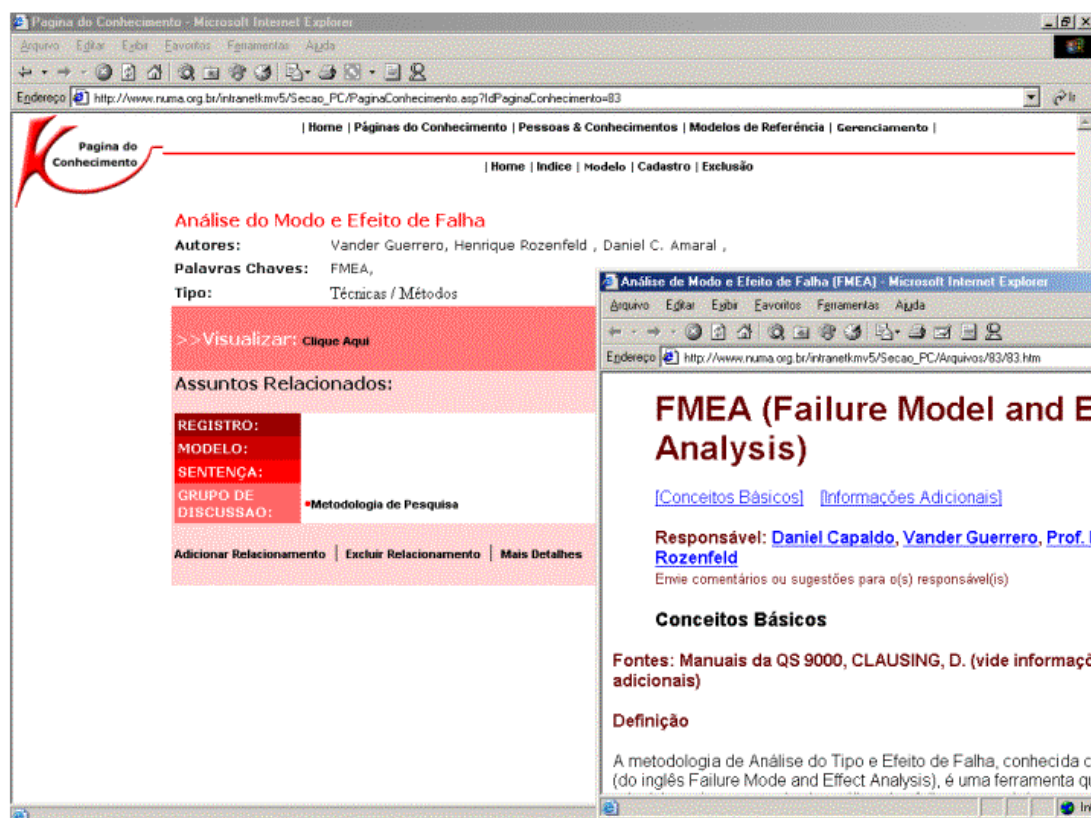


Figura 5. 16: Página do Conhecimento

Na Página do Conhecimento apresentada na figura 5.16, há, ainda, na parte superior, *links* para o cadastro de novas páginas e, no rodapé, *links* para atualizar os relacionamentos, isto é, incluírem-se novos conhecimentos dentro da página apresentada.

O procedimento para o cadastro de novas páginas de conhecimentos é composto pela criação dos textos do corpo principal e pelo cadastramento, via intranet, da página. Nesse cadastro, deverão ser incluídas inicialmente as informações principais sobre a página, tais como título, autores e palavras-chave. Depois disso o usuário faz um *upload* enviando para o servidor a página em html simples que contém o corpo da página de conhecimento. Feito isto, já deverá aparecer a nova página. Por fim, basta o usuário criar os relacionamentos entre a página de conhecimento e os demais conhecimentos explícitos armazenados no sistema.

Pessoas & Conhecimento

Este é o módulo mais simples e contém informações sobre os usuários do sistema e Grupos de Discussão. O módulo é iniciado com a apresentação de uma lista dos diversos usuários do sistema e seus respectivos papéis ou perfis (veja figura 5.17). No caso específico da figura 5.16, todos os usuários estão cadastrados como administradores de conhecimento. Ao “clique” sobre o nome da pessoa abre-se um email com o endereço eletrônico devidamente preenchido.

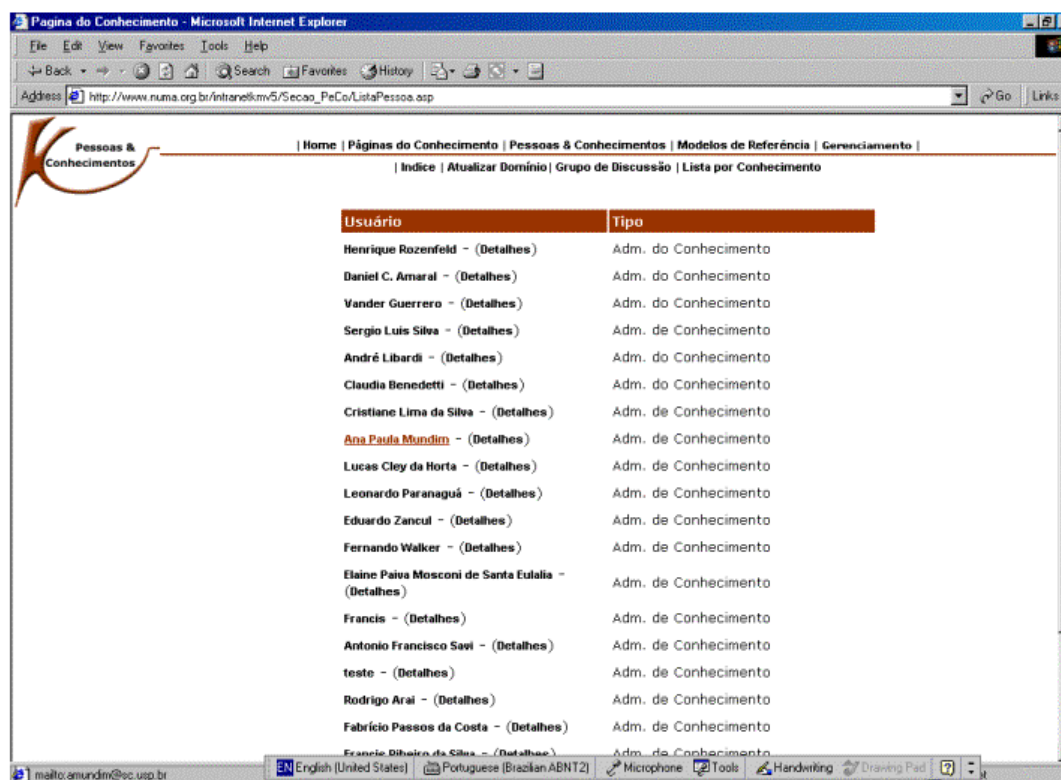


Figura 5. 17: Tela inicial do módulo Pessoas & Conhecimento

Escolhendo-se o link “detalhes”, surgirá uma tela com informações detalhadas sobre o usuário, como endereço e telefone para contatos (ver figura 5.18). Uma informação fundamental propiciada pelo sistema é a área de domínio do usuário, isto é, as palavras-chave onde o usuário tem experiência e/ou é um conhecedor do assunto. Essa é uma característica extremamente importante do sistema pois permite verificar o grau de conhecimento das pessoas sobre as palavras-chave. Essa mesma informação pode ser obtida, em outro formato, a partir da tela inicial acessando o *link* “Lista por Conhecimento”, que aparece na barra superior. Este *link* retorna uma lista com as palavras-chave cadastradas no sistema e, abaixo

delas, os usuários que dominam tais palavras. O usuário pode identificar uma lista de membros da intranet que conhecem determinado assunto, havendo, ainda, uma indicação sobre o nível do conhecimento desses membros sobre o assunto e o porquê.

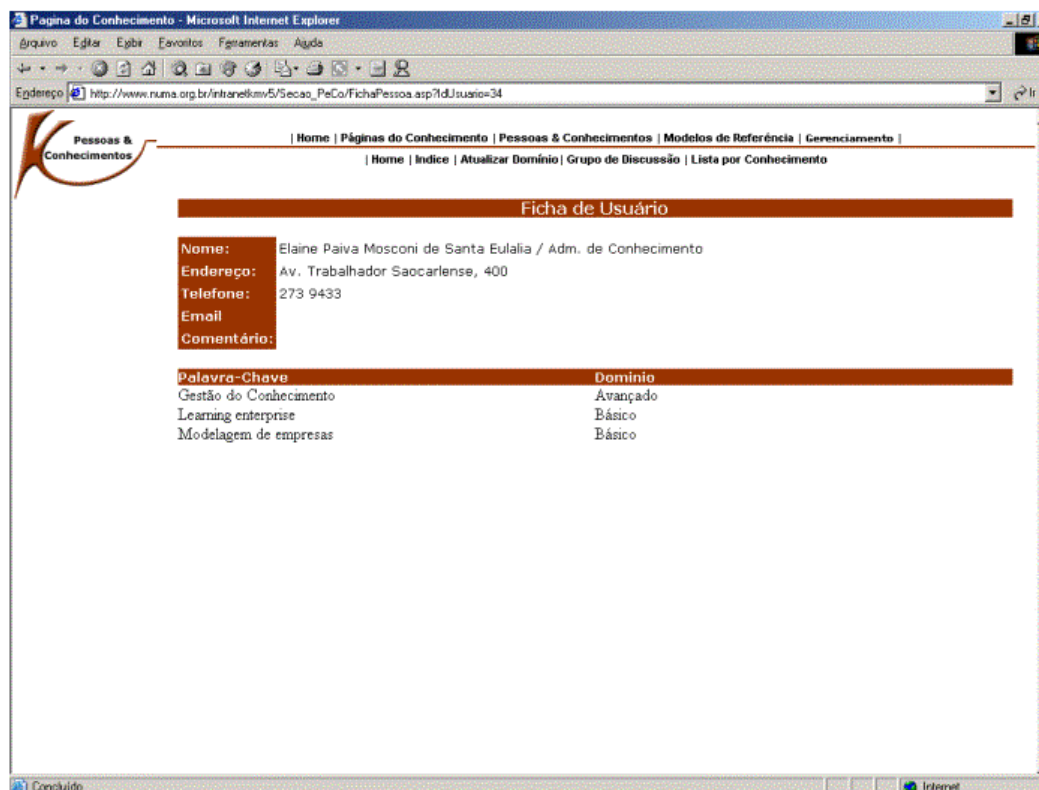


Figura 5. 18: Exemplo de uma ficha com informações detalhadas sobre um usuário

As informações sobre o domínio são efetuadas e mantidas pelos próprios usuários. Basta que acessem o link “Atualizar Domínio” na barra superior da tela principal do módulo de Pessoas & Conhecimentos (Veja figura 5.17). O resultado dessa ação é a apresentação de uma tela parecida a da figura 5.19, onde se lê o nome do usuário que está utilizando o sistema e a lista de domínios atualmente cadastrados. Portanto, empregando os *links* “Adicionar Domínio” e “Excluir Domínio”, o usuário poderá relacionar novas palavras-chave como temas sobre seu domínio ou excluir relacionamentos anteriormente efetuados. Quando se estabelece um relacionamento, o usuário, além de classificar seu domínio entre básico, intermediário e avançado, é solicitado a escrever uma frase justificando a classificação escolhida. Desse modo, o

usuário que busca um especialista entre os membros da intranet tem uma informação adicional sobre o domínio do usuário sobre o assunto. Deve-se notar que o usuário tem acesso apenas ao “seu” próprio domínio, sendo impossibilitado de alterar domínios de outros usuários.

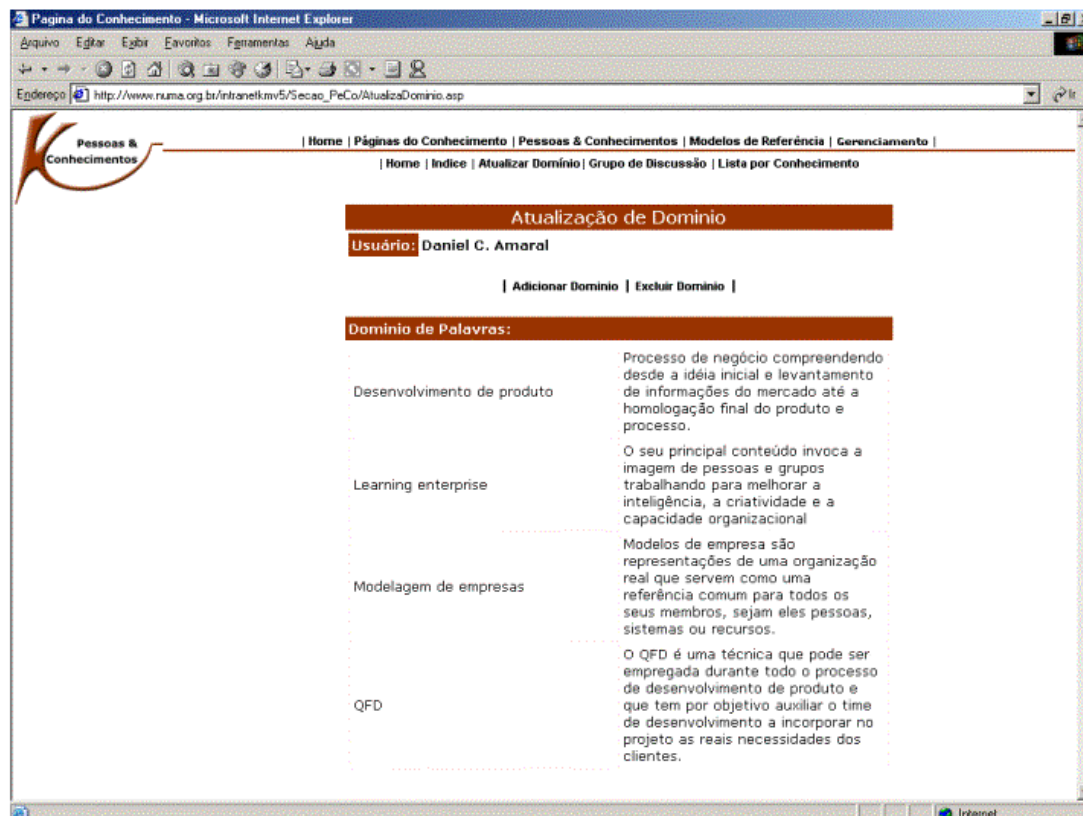


Figura 5. 19: Exemplo de tela para atualização do domínio do usuário

A lista de Grupos de Discussão funciona de maneira similar à lista de pessoas. Esses grupos de discussão podem ser internos ou externos à empresa e essa lista é acessada, também, da página inicial no link “Grupos de Discussão”, que aparece na barra superior. A figura 5.20 mostra um exemplo de lista com dois grupos de discussão cadastrados. Os grupos são importantes pois são canais através dos quais se pode discutir e eliminar dúvidas sobre os diversos assuntos relacionados com o processo de desenvolvimento de produto.

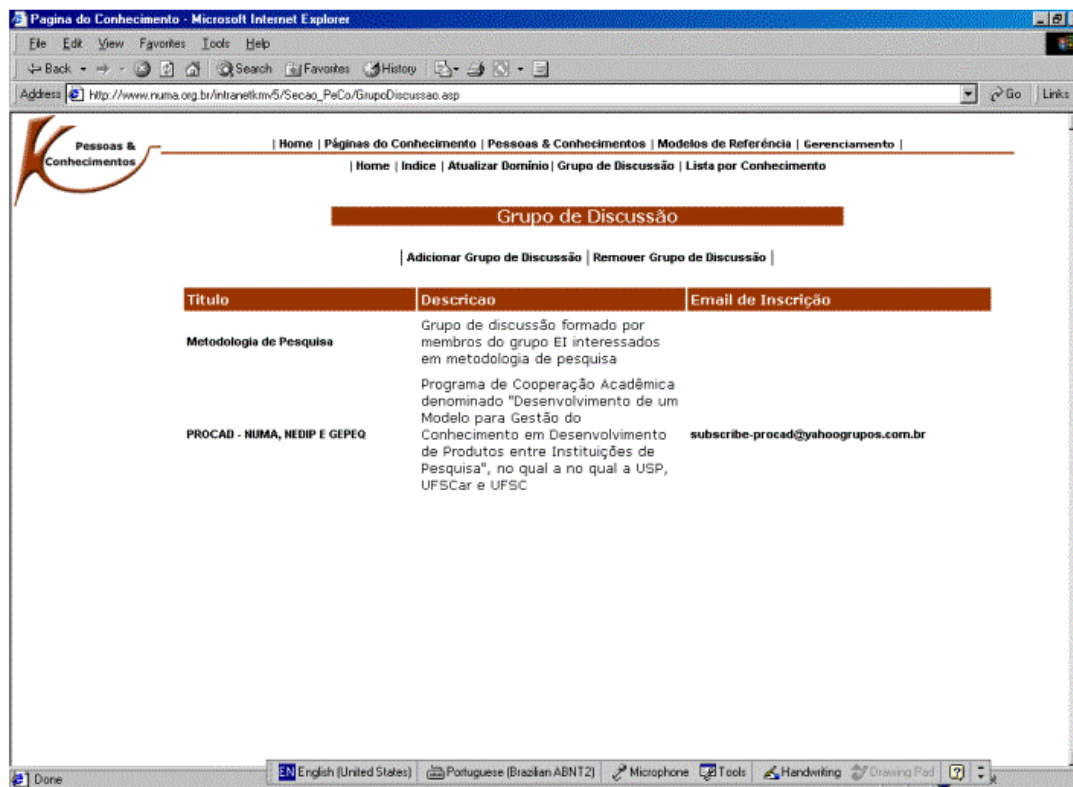


Figura 5. 20: Exemplo de lista de Grupos de Discussão

5.2.2 Descrição do processo de desenvolvimento da intranet

A visão geral do processo de desenvolvimento proposto para a solução é apresentada na figura 5.23. Nessa figura, as caixas representam atividade, as caixas arredondadas resultados e as caixas em cinza, as ferramentas de desenvolvimento utilizadas. O processo de desenvolvimento de *software* é dividido em três grandes fases: especificação, definição e implementação. Cada uma delas é descrita em detalhes a seguir:

especificação: nesta fase, foram definidos os requisitos do sistema, isto é, as especificações em termos de necessidades dos usuários. Iniciou-se pela definição dos contornos ou a abrangência do sistema, por meio da elaboração de um documento denominado “visão”. Ele contém o escopo do sistema e as suas características principais. Em seguida, foram identificados os perfis de cada tipo distinto de usuário que iria utilizá-lo e levantados os requisitos, analisando-se as necessidades de cada um dos perfis. Ambas as atividades tiveram como base os requisitos e especificações da arquitetura descritos no capítulo 4. Ao final, foram gerados os seguintes

documentos: 1) visão do sistema (contendo principais objetivos do sistema e sua abrangência); 2) atores do sistema; 3) lista de Requisitos, contendo um diagrama de contexto e os casos de uso principais do sistema (que em seu conjunto delimitam as funcionalidades principais que precisam ser atendidas);

definição: nesta fase, os requisitos gerados na etapa anterior foram transformados em especificações para o desenvolvimento do sistema, ou seja, em informações capazes de guiar a implementação. Essas informações compreendem o modelo de informações (o desenho da base de dados e o dicionário de dados), o modelo dinâmico (representado pelo mapa de telas) e as especificações de design gráfico. Foram gerados os seguintes documentos: 4) modelo entidade relacionamento da base de dados (MER); 5) dicionário de dados; 6) projeto gráfico (contendo cada um dos elementos gráficos e um protótipo da tela principal); 7) protótipo de telas (gerado em arquivo do software *Microsoft Powerpoint* contendo as informações principais de cada tela, isto é, os dados a serem mostrados e os controles, tais como botões, listas, etc...) e 8) mapa do site ou mapa de telas (diagrama contendo as páginas principais do sistema e as principais interações entre cada uma delas);

implementação: foi a última fase, na qual a especificação gerada na etapa anterior serviu como base para a geração do código, ou seja, dos arquivos de páginas HTML, da base de dados e demais componentes. Essa fase seguiu o clássico modelo de desenvolvimento de *software* em espiral, na qual as especificações obtidas na fase de definição são transformadas em protótipos, depois em código e em seguida são testadas e atualizadas num ciclo que só termina depois da aprovação do sistema (na figura definição, protótipo, implementação e teste). Durante essa fase várias alterações e detalhamentos foram realizados. Foi a mais demorada e a que consumiu o maior esforço, num total de 7 meses (de setembro de 2000 até março de 2001). No decorrer dessa fase, o dicionário de dados e o protótipo de telas deixaram de ser efetivamente utilizados pelos programadores. Tornaram-se desatualizados e, com o passar do tempo, também obsoletos. A documentação principal do sistema atualmente consiste da visão, requisitos, design gráfico, MER e mapa de telas.

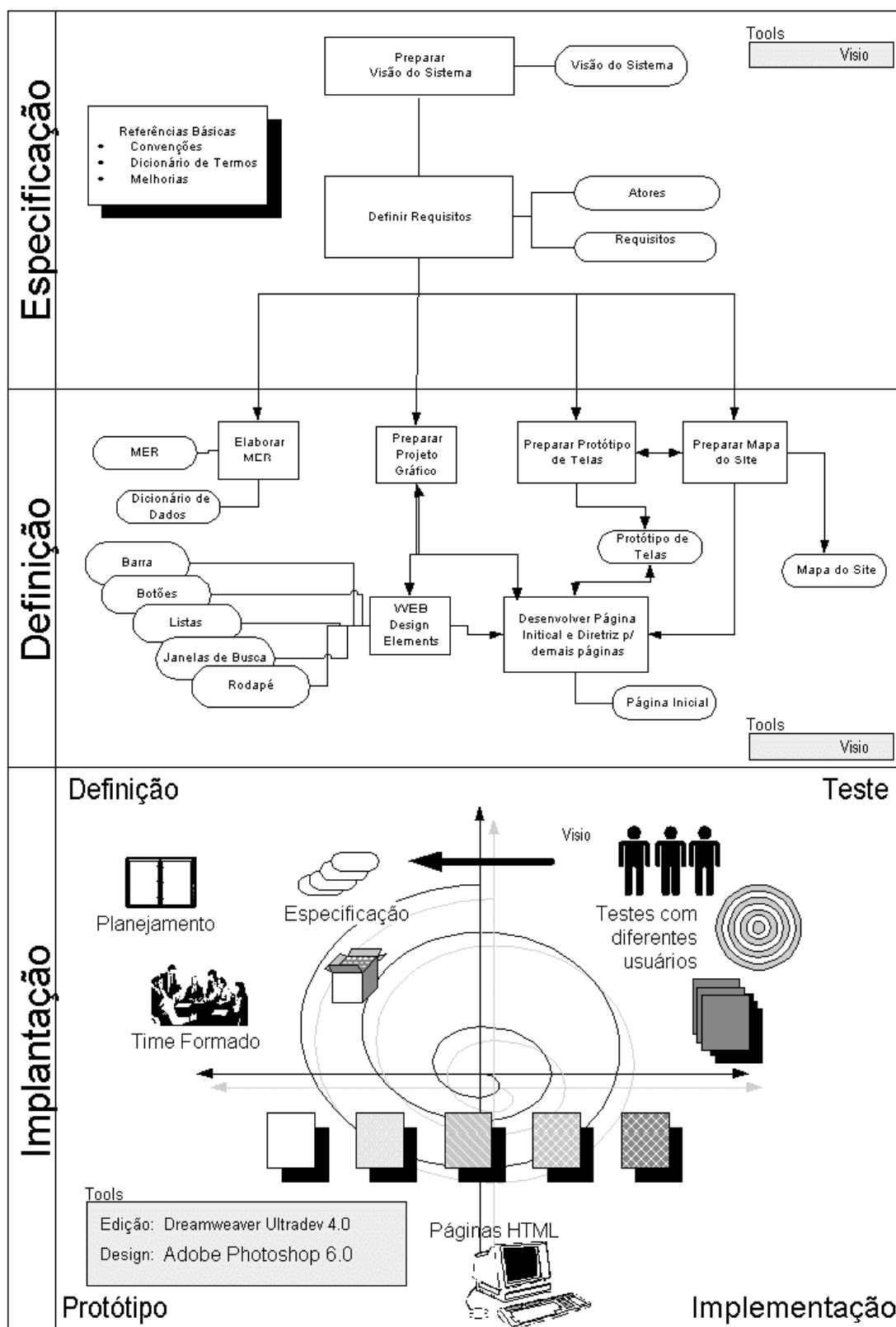


Figura 5. 21 : Processo de desenvolvimento da intranet

5.2.3 Modelo da Base de Dados

O modelo entidade-relacionamento (MER) da base de dados da intranet é apresentado na figura 5.22, destacando-se as entidades mais importantes do sistema. Entre as principais, encontram-se as que representam os conhecimentos explícitos: registro, sentença e modelo. Outras entidades importantes são a Página de Conhecimento, que contém as informações sobre estes “roteiros” de acesso aos conhecimentos explícitos e as entidades palavra-chave e usuário, que permitem o relacionamento entre os conhecimentos explícitos.

A entidade usuário representa as pessoas cadastradas no sistema e se relaciona de duas formas principais com as entidades de conhecimento (registro, sentença, modelo e páginas de conhecimento): como criadora e como “comentarista” de conhecimentos explícitos. O relacionamento de criadora permite que se registre quem foi o responsável pelo cadastramento de cada conhecimento explícito. O papel de “comentarista” é importante pois registra quem é o usuário que criou um comentário, entidade que está relacionada com os vários conhecimentos explícitos. A entidade usuário tem ainda um importante relacionamento com a palavra-chave que é o denominado “domínio”. É esse relacionamento n para n que permite o armazenamento do nível de domínio do usuário sobre as palavras-chave. Esse relacionamento garante a ligação de cada usuário com as palavras-chave que descrevem os temas em que o usuário é conhecedor e referência, dentro da organização. Esse relacionamento tem um atributo denominado nível, que pode assumir um dos valores entre básico, intermediário ou avançado, conforme o nível de conhecimento do usuário daquele determinado domínio.

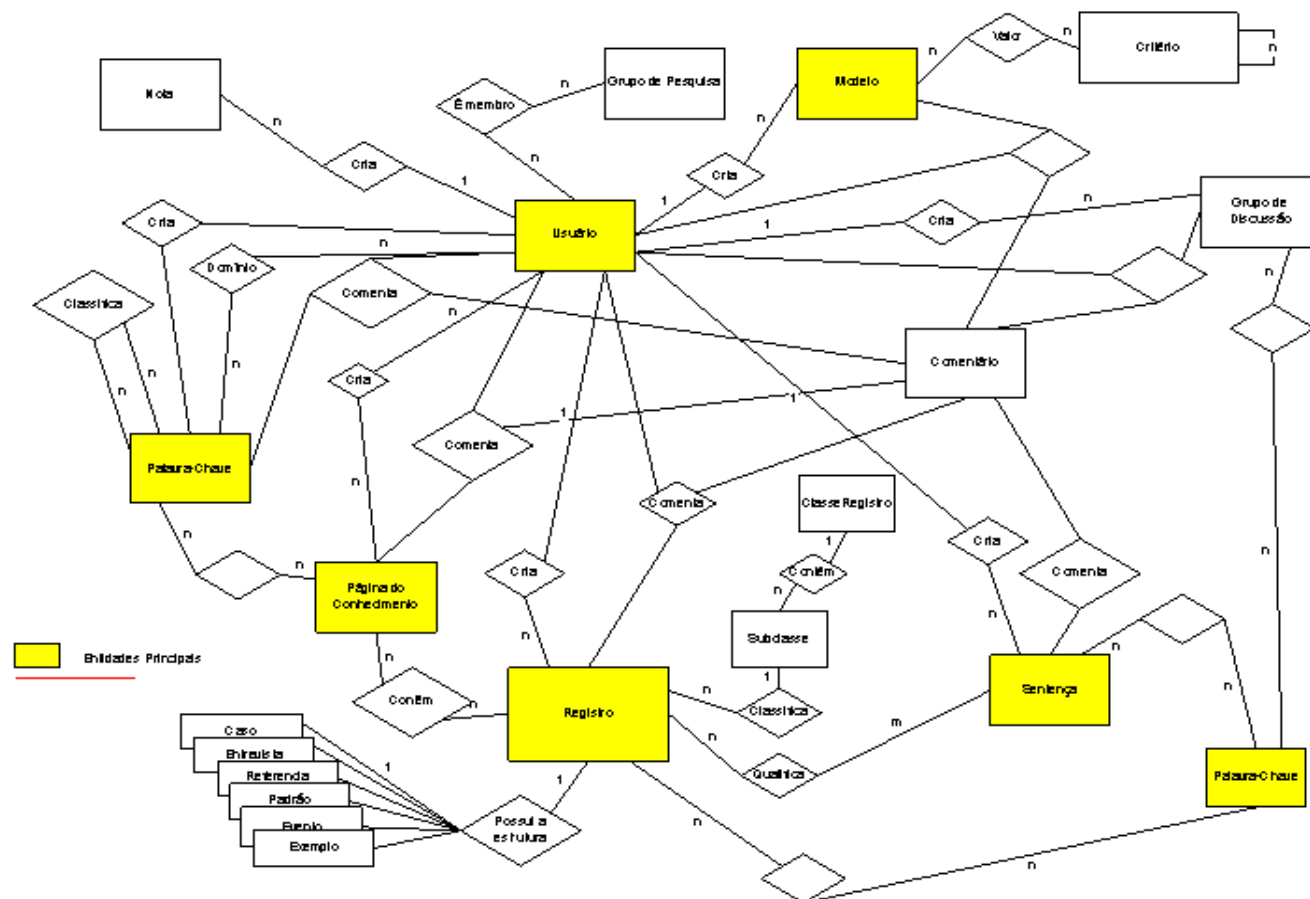


Figura 5.22: Modelo entidade-relacionamento da base de dados

Uma outra entidade, que tem papel fundamental dentro dessa base de dados é a palavra-chave. É essa entidade que, por natureza, permite o relacionamento entre os diferentes conhecimentos explícitos armazenados. Por exemplo, a ligação entre modelos de referência e conhecimentos explícitos é feita por meio de uma busca em que são localizados todos os conhecimentos explícitos ligados a uma palavra-chave, que está relacionada a um objeto requisito de conhecimento do modelo (veja no item 5.3.5). De forma semelhante, uma palavra-chave e o relacionamento entre esta entidade e a Página do Conhecimento permitem que sejam, automaticamente, encontrados vários conhecimentos explícitos, relacionados com a página de conhecimento. Assim, no mesmo instante em que se cria uma nova página do conhecimento, torna-se, automaticamente, disponível um conjunto de conhecimentos explícitos, relacionados com essa página. Outro relacionamento importante da entidade palavra-chave é o relacionamento de “domínio”, estabelecido com a entidade Usuário, conforme apresentado no parágrafo anterior.

Uma característica especial dessa base de dados é o desenho da entidade registro e seus relacionamentos com classe e subclasse. Em primeiro lugar, porque se trata da estrutura mais complexa, na medida que armazena os conhecimentos explícitos de tipo não-estruturado, e portanto, cuja estrutura de dados pode variar enormemente conforme as características deste conhecimento. Um livro tem atributos-chave muito diferentes de um evento, um endereço de especialista ou ou um formulário qualquer. O grande desafio era escolher um conjunto de dados, mais simples possível, que permitisse armazenar todos os diferentes tipos de informação. Esse desenho segue o modelo proposto pela arquitetura (veja item 4.3.1). Criou-se uma estrutura de “supertipo” para o registro, por meio da adição de “filhos” para esta entidade, cada qual armazenando dados específicos de um conjunto de tipos de registros. Os dados básicos, comuns a todos os tipos de registros, são armazenados nessa entidade pai denominada “registro”: identificação do registro, título, data da criação permanecem armazenados nesta entidade. Concomitantemente, foram criados 6 subtipos de registro que armazenam os dados específicos. Esses subtipos, baseados no modelo disposto na arquitetura, formam, idealmente, o conjunto mínimo de entidades, suficientes para armazenar qualquer tipo desejável de registro. Os subtipos que aparecem na base e constam da proposta da arquitetura são:

- **Referências:** guardam os dados de registros do tipo publicação, cobrindo desde seriados, periódicos, relatórios e documentos de diversas natureza até livros publicados. Os atributos principais dessa entidade são referência completa e ano de publicação;
- **Eventos:** armazenam os conhecimentos explícitos, com características de temporalidade, isto é, informações sobre congressos, encontros, reuniões e outros tipos de acontecimentos com datas de início e fim. Os atributos principais desta entidade são data e período de duração;
- **Especialistas:** são conhecimentos explícitos que identificam contatos importantes em uma determinada área. É importante não confundi-los com os usuários. Os especialistas são conhecimentos de contatos, isto é, armazenam nome, descrição e endereço de pessoas ou empresas que dominam um determinado assunto. São, portanto, registros de pessoas (contém endereços, telefones, etc...) que não deverão estar cadastradas no sistema como usuário. Esta entidade serve para registrar contatos como fornecedores, clientes e também pessoas que dominam um determinado assunto ou oferecem algum tipo de serviço especializado em desenvolvimento de produto. O atributo principal é o endereço para contato;
- **Entrevistas:** são os conhecimentos explícitos formados por registros de entrevistas realizadas, tais como transcrições de fitas, respostas de questionários, etc. Essas entrevistas podem ter sido motivadas por diversos fatores, desde pesquisas, até visitas sem qualquer compromisso. Os atributos principais são contexto (hora, local e motivo da entrevista) e descrição;
- **Padrões:** são os conhecimentos explícitos na forma de padrões e registros de trabalho, isto é, são formulários, regras, procedimentos, e outros documentos que estabelecem normas para execução de atividades, ou que servem como instrumentos para registros de resultados de atividades realizadas. Os campos principais são abrangência (local e escopo da aplicação do padrão), data, versão e validade e

- **Casos:** são narrativas de ocorrências (histórias) vividas dentro ou fora da organização e que se deseja compartilhar com as demais pessoas dentro da empresa. Os atributos principais são contexto (local, data e motivos), problema (enfrentado ou ocorrido), solução (caso tenha sido tomada alguma ação frente ao problema) e resultados (resultado do problema, que pode ser positivo ou negativo).

Conforme o modelo proposto na arquitetura (vide novamente o item 4.3.1), a entidade registro se relaciona ainda com duas entidades que auxiliam na organização e classificação destes: entidades Classe e Subclasse (no caso subentendido que são classes e subclasses de registros). Essas entidades armazenam a estrutura de classificação dos registros, de forma que a intranet possa ser “customizada” para cada aplicação, criando-se uma classificação de classe e subclasse específica. Estes dois níveis de classificação dos registros permitem adequar a terminologia de organização dos registros ao linguajar padrão existente no ambiente da organização, conforme especificado na arquitetura.

O desenho da entidade registro, aliado com a classificação permitida pelas entidades classe e subclasse, fornece à base de dados do sistema uma grande flexibilidade, pois alia a capacidade de guardar, em poucos atributos bastante específicos, dados para qualquer tipo de documentos e permite, ainda, que esses documentos possam ser organizados de diferentes formas pelo usuário, conforme as necessidades da organização.

As demais entidades, que aparecem nesta versão simplificada do MER da solução, são menos fundamentais para a ilustração da proposta da arquitetura, portanto não são descritas em maiores detalhes.

5.2.4 Mapas de Telas

Documentos importantes da especificação são os mapas de telas, os quais representam, esquematicamente, as páginas principais do sistema e seus relacionamentos. Abaixo, a figura 5.23 apresenta o mapa de telas de um dos módulos de Práticas Sistematizadas da intranet, o mais complexo de todos. Esses mapas auxiliaram no desenvolvimento, demonstrando as páginas principais que precisariam ser desenvolvidas e, quando necessário, as variáveis que seriam transmitidas entre elas. As páginas são apresentadas pelos símbolos retangulares e as flechas indicam os

links principais do sistema. Páginas de menor importância foram omitidas do desenho. A solução final se tornou consideravelmente grande, possuindo a intranet um total de, aproximadamente, 80 páginas, praticamente, a totalidade delas dinâmicas, com códigos escritos em VBScript e empregando a tecnologia ASP (*Active Server Pages*).

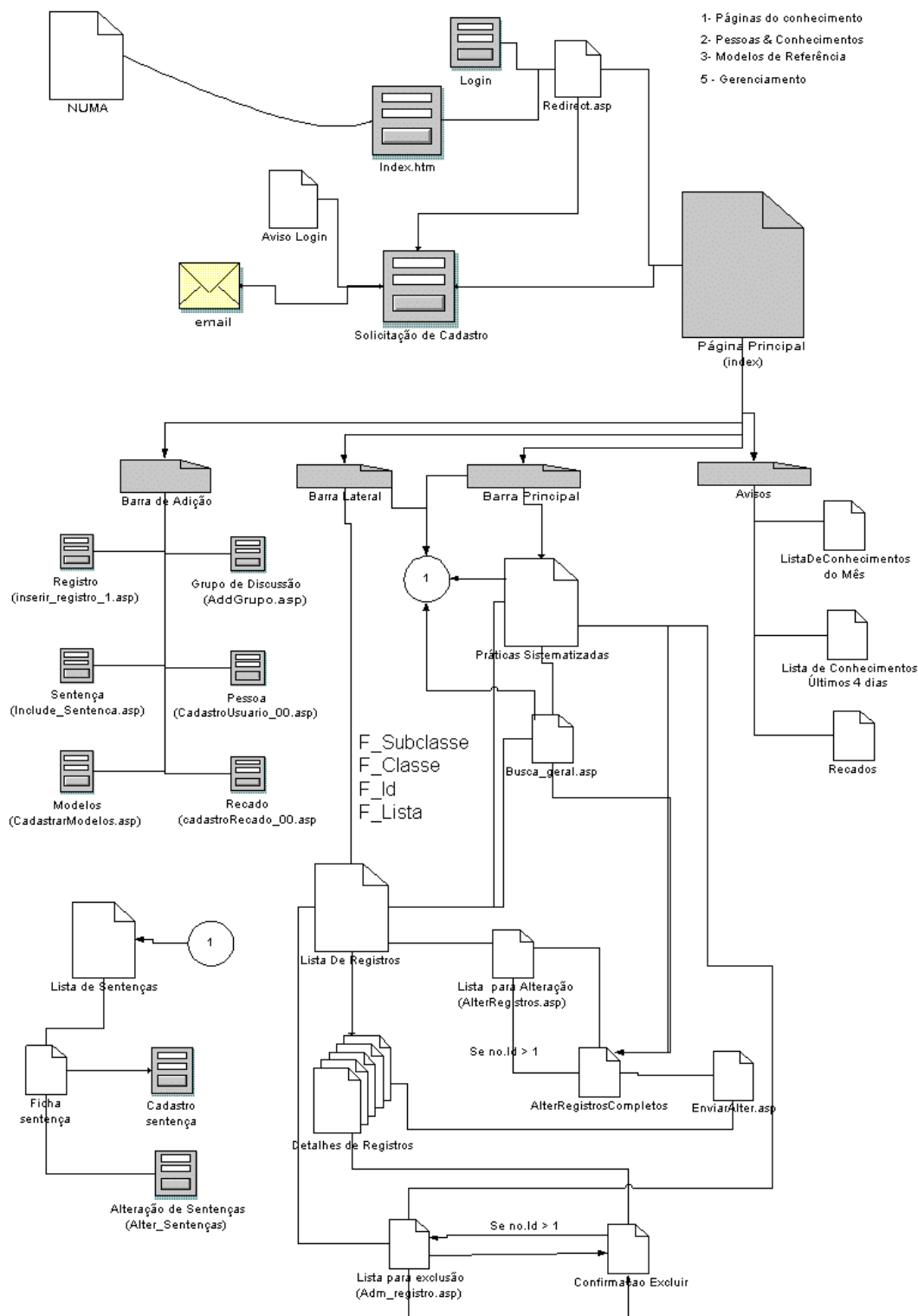


Figura 5. 23: Mapas de telas do módulo de Práticas Sistematizadas

5.3 Adaptação da Ferramenta de Modelagem ARIS Toolset

A ferramenta de modelagem é o componente da solução criado para auxiliar na descrição dos modelos na forma gráfica, conforme as especificações do *framework* de modelagem descritas no capítulo 4. Porém era necessário permitir que páginas contendo a descrição do modelo pudessem ser acessadas diretamente da intranet, sem que fosse necessária a utilização de outro sistema pelo usuário e também possibilitando a integração destes modelos com os conhecimentos explícitos, conforme especificado na arquitetura.

Decidiu-se preparar a solução de forma tal que a ferramenta de modelagem fosse utilizada, principalmente, pelo usuário do tipo modelador, auxiliando-o na construção dos modelos e permitindo-lhe gerar páginas HTML que seriam incorporadas na intranet por meio de simples *“links”*. Assim, os demais usuários poderiam navegar pelos modelos diretamente destas páginas, que estariam interligadas com os conhecimentos explícitos. Dessa forma, a ferramenta de modelagem é exigida, apenas, durante a construção ou alteração dos modelos, enquanto toda a consulta e os dados cadastrais sobre eles seriam gerenciados por meio da intranet.

Conforme a especificação da arquitetura, a existência de um *framework* e de uma ferramenta de modelagem não implica que todos os modelos de referência devam ser armazenados, necessariamente, neste único formato padrão. A intranet aceita cadastros de modelos em qualquer formato, incluindo qualquer tipo de arquivo. A única restrição é que os arquivos não armazenados em HTML, PDF ou outro *software* compatível com os *browsers* não poderão ser visualizados diretamente pelo usuário, ficando disponíveis para *download*. O usuário terá que armazenar o documento em seu computador e abri-lo, empregando o sistema específico no qual o modelo foi gerado. Assim, a intranet reúne os registros e informações de todos os modelos de processo, independentemente do formato do arquivo e do *framework* de modelagem empregado no seu desenvolvimento.

A vantagem de se ter os modelos descritos conforme o *framework* é que, para estes, estará disponível a integração entre modelo e conhecimentos explícitos armazenados. Esta integração permite que todo o modelo gerado segundo o

framework esteja automaticamente conectado a esses conhecimentos explícitos, sem esforço adicional ou conhecimento prévio do modelador sobre a existência de tais conhecimentos. É uma integração limitada na direção modelo-conhecimento explícito, uma vez que o caminho inverso não pode ser realizado, isto é, partir de um conhecimento explícito e listar as partes dos modelos que o empregam. Porém, mesmo com esta limitação a integração desenvolvida já disponibiliza recursos suficientes para garantir um repositório em conformidade com a arquitetura proposta.

No contexto da solução completa, a ferramenta de modelagem empregada deve suportar as funcionalidades da arquitetura listadas a seguir:

- **construir, alterar e apagar parte dos modelos de processo:** a ferramenta armazena os modelos de processo descritos segundo o *framework* de modelagem, suportando a construção e edição desses modelos;
- **gerar:** funcionalidade que permite aos modelos armazenados ser publicados em um conjunto de páginas WEB interligadas, as quais, por sua vez, podem ser acessadas diretamente da intranet. Assim, ao final da construção de um modelo segundo o *framework*, deve-se gerar a versão em HTML e cadastrá-la na intranet. O cadastro oficial de modelos de processo da empresa será sempre a intranet e
- **visualizar:** a ferramenta deve permitir, ainda, a navegação no modelo, para os usuários com permissão de desenvolvê-los e alterá-los.

A ferramenta de modelagem escolhida foi a ARIS-TOOLSET, versão 4.0, uma das mais sofisticadas ferramentas de modelagem do mercado. Ela foi desenvolvida, inicialmente, voltada para o emprego do *framework* de modelagem ARIS (*ARchitecture for Integrated Systems*). Na versão utilizada, v.4.0, a sua interface, antes voltada exclusivamente para o framework ARIS, foi aprimorada e tornou-se genérica, facilitando o emprego da ferramenta com outros *frameworks*.

Esta ferramenta foi “customizada” para se adequar às funcionalidades acima descritas, empregando-se 3 recursos do sistema: filtros, estrutura de diretórios padrão, modelos de relatório e modelos de página *web*, os quais são apresentados,

respectivamente, nos subitens 5.3.1, 5.3.2, 5.2.3 e 5.3.4. No último subitem, 5.3.5, descreve-se a integração entre a ferramenta de modelagem e a *Intranet*.

5.3.1 Filtro de Modelagem

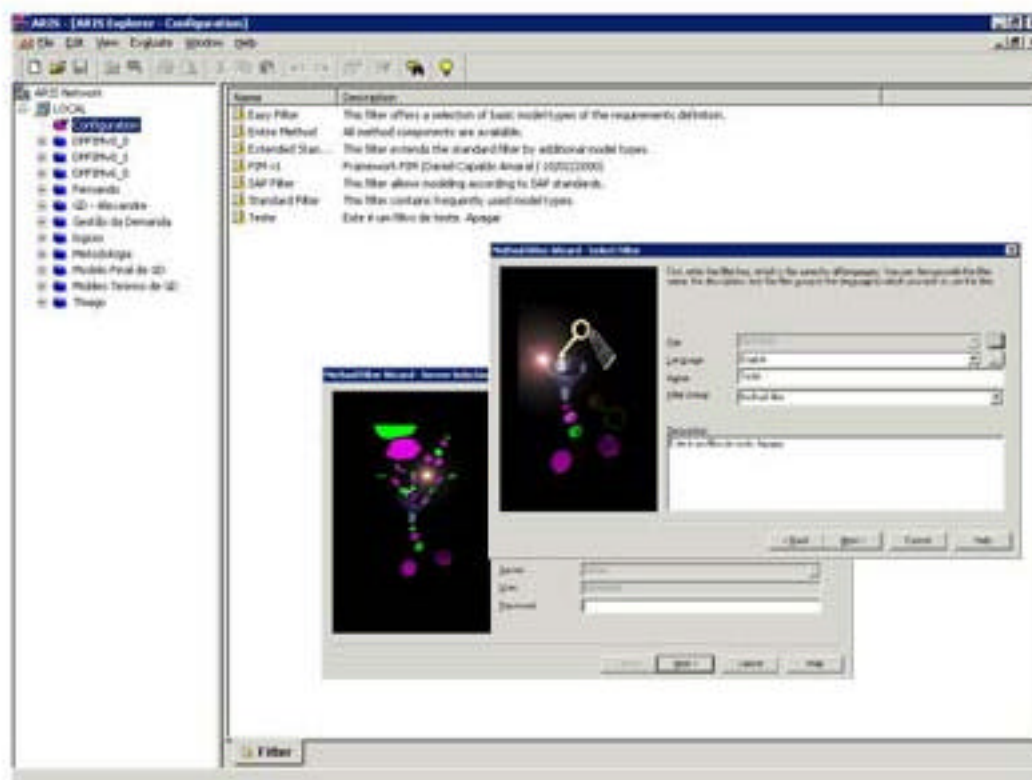
Os filtros desempenham um papel fundamental dentro da ferramenta de modelagem ARIS Toolset v. 4.0. Eles permitem que os usuários selecionem somente os construtos, relacionamentos e formalismos que deverão ser utilizados, sendo todos os demais “escondidos” do usuário. Isto significa que, por exemplo, caso um determinado formalismo possua 3 tipos diferentes de construtos (denominados objetos na ferramenta) e seja criado e aplicado um filtro contendo apenas dois deles, o usuário, no instante da modelagem, terá à disposição, na sua “paleta” de ferramentas (também conhecido como “caixa de ferramentas”), apenas os dois construtos contidos no filtro. O terceiro estará oculto e não poderá ser utilizado e nem visualizado por esse usuário.

A possibilidade de criação desses filtros é importante porque esta ferramenta possui, aproximadamente, uma centena de diferentes tipos de formalismos, cada qual contendo vários construtos e esses, por sua vez, com diferentes tipos de relacionamento com os demais construtos. A ferramenta incorpora os formalismos mais conhecidos nas áreas de análise de sistemas e de modelagem de empresas. Caso todos fossem apresentados ao usuário (durante todo o tempo) seria difícil encontrar os elementos essenciais para a modelagem, além de aumentar o risco de falta de padronização pois diferentes indivíduos, ou um mesmo indivíduo em momentos diferentes, poderiam criar representações semelhantes de partes do processo semelhantes empregando diferentes formalismos, *construtos e relacionamentos*.

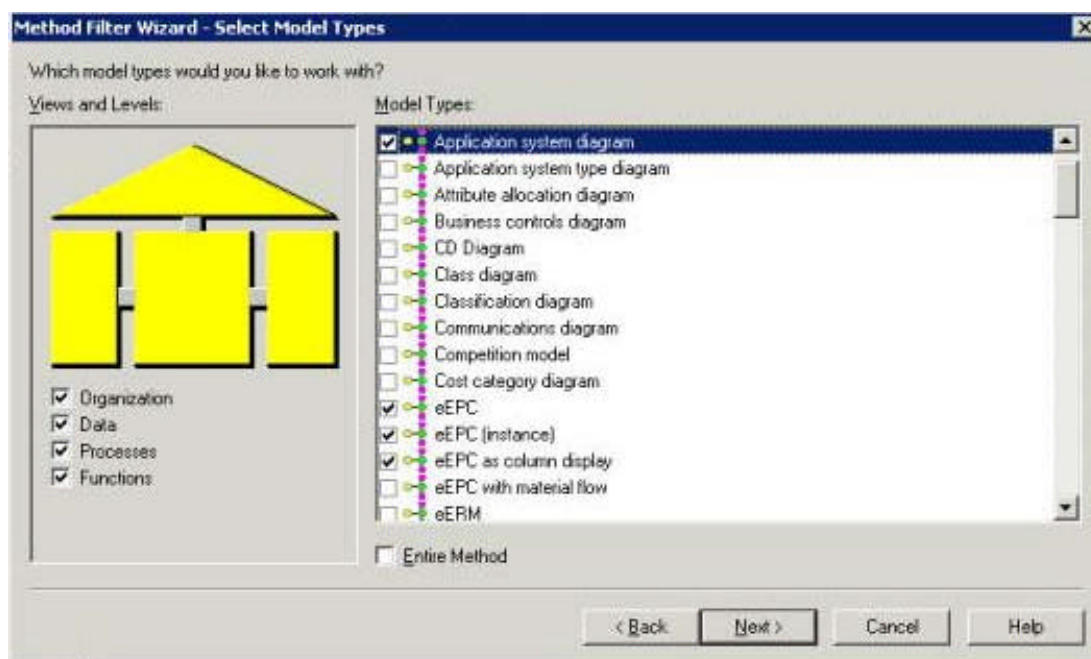
No caso do *framework* de modelagem proposto neste trabalho foi gerado um filtro contendo apenas os formalismos, construtos e relacionamentos previstos no *framework* (conforme quadro 4.4). O modelador tem, à sua disposição, somente os elementos essenciais para a modelagem segundo o *framework*, simplificando o trabalho de modelagem e evitando erros.

A geração do filtro é feita por meio de um assistente, denominado “*Wizard Filter*”, apresentado na figura 5.24. O primeiro passo desse assistente (duas primeiras telas) é solicitar do usuário: senha de acesso, título e a descrição do filtro (veja 5.24 a). Em seguida, surgem várias telas nas quais o usuário configura os

modelos (formalismos), objetos (construtos) e os atributos e relacionamentos entre eles. A primeira delas, segundo passo, solicita que os usuários identifiquem os tipos de modelos (formalismos) que serão utilizados. Isso é feito por meio de uma lista conforme a figura 5.24 b. De maneira semelhante, o usuário irá escolher ao pressionar o botão “next” nessa tela: 3) os atributos a ser empregados para descrever cada tipo de modelo (formalismos); 4) os tipos de objetos que serão utilizados (construtos); 5) os atributos para cada um dos tipos de objetos selecionado; 6) Os tipos de modelo que poderão ser relacionados (anexados) a cada um dos tipos de objetos; 7) os símbolos que deverão empregados para cada tipo de objeto, em cada um dos tipos de modelos específicos; 8) os tipos de relacionamentos que poderão ser estabelecidos entre cada um dos tipos de objeto, para cada um dos tipos de modelos; e, finalmente, (passo 9) os atributos que deverão ser empregados para descrever cada um destes relacionamentos.



(a) Telas iniciais do assistente de criação do filtro



(b) Tela para definição dos tipos de modelos a serem incorporados no filtro

Figura 5. 24 : Assistente para a criação do filtro de modelagem na ferramenta ARIS

5.3.2 Estrutura de Diretórios Padrão na Ferramenta ARIS

O filtro apresentado anteriormente não contempla um aspecto importante do *framework*, que é a dimensão das visões. Para contornar este problema, as visões foram geradas em diretórios pré-definidos dentro da ferramenta, isto é, cada visão possui um diretório próprio onde são alocados modelos e os construtos que descrevem essa visão (ver figura 5.25). Assim, para um único modelo, quando se deseja consultar a visão de atividade basta direcionar-se para a pasta atividade, na qual poderão ser visualizados os modelos que representam essa visão e poderão ser listados os vários construtos (objetos) referentes a essa mesma visão.

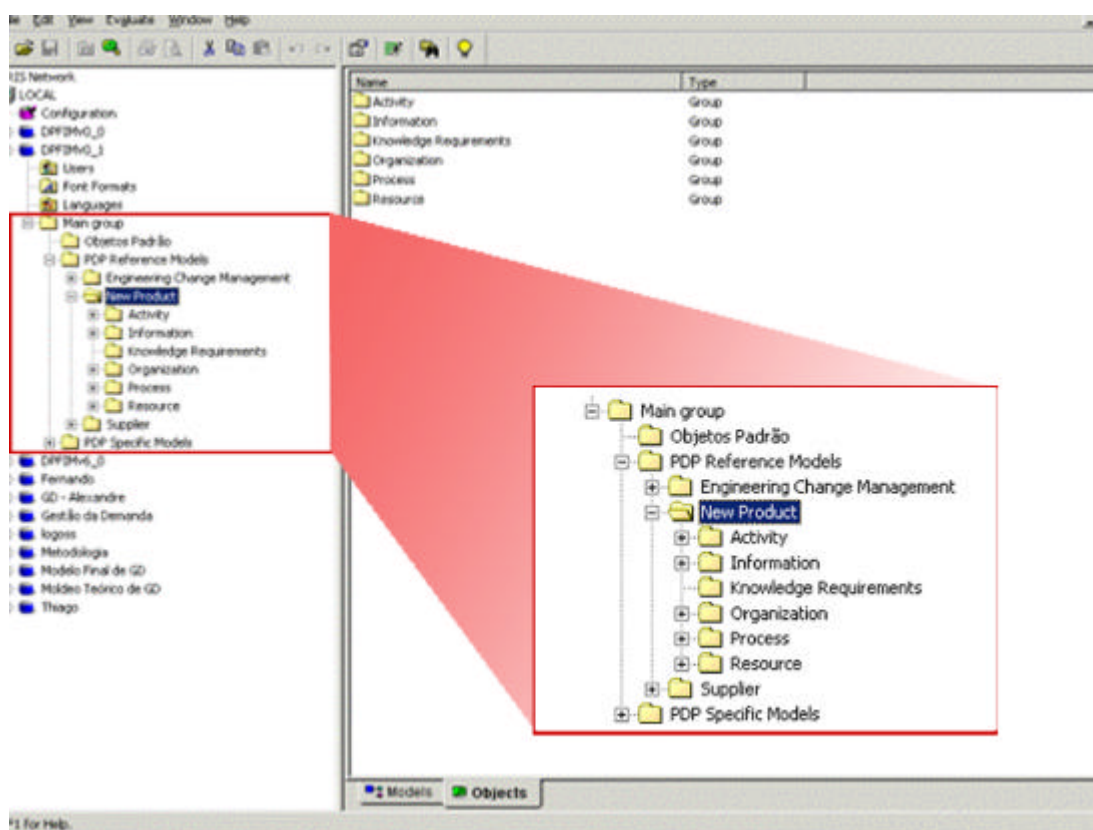


Figura 5. 25: Visões de um modelo armazenado na ferramenta de modelagem ARIS

Cada uma destas pastas contém subdivisões padrão que permitem organizar os construtos e modelos referentes à visão respectiva. Esta subdivisão é especificada na tabela 5.1, que discrimina qual o conteúdo de cada uma delas (vide a Figura 5.26, que apresenta como essa subdivisão é implementada na ferramenta ARIS). O intuito destas subdivisões é facilitar a navegação e busca dentro de uma mesma visão, de forma que o usuário possa listar ou encontrar a informação desejada.

É importante notar as duas “folhas” de título “Models” e “Objects”, que surgem na tela apresentada na figura 5.25, mais especificamente, ao pé da segunda coluna. Para um mesmo diretório, previamente selecionado, cada folha apresenta, respectivamente, listas dos modelos e objetos (neste linguajar construtos) presentes no folder. Durante a modelagem, por convenção, todos os construtos (“objetos”) são localizados no diretório raiz da visão a que ele pertence. Os Objetos “Activity” são colocadas dentro da pasta Atividade. Objetos “Information Carrier” e “Cluster” são localizados no diretório Informação. E assim por diante, conforme a lista de objetos por visão apresentada no quadro 4.4, capítulo 4. Os objetos estarão sempre todos

juntos no diretório raiz. Nos subdiretórios deverão existir modelos que apresentam classificações e/ou diferentes formas de organizar e visualizar esses objetos. Isto é importante para facilitar a localização e reutilização dos objetos.

Quadro 5. 1: Estrutura de diretórios padrão na ferramenta de modelagem

Nível 0 - Visão	Nível 1	Descrição
Atividade	Por Objeto (by object)	Contém modelos que descrevem as atividades, hierarquizadas por tipo de objeto relacionado.
	Por Operação (by Operation)	Contém modelos que descrevem as atividades, hierarquizadas por tipo de operação.
	Por Processo (by Process)	Contém modelos que descrevem as atividades, hierarquizadas por tipo de processo.
Informação	Análise Econômica (Economic Analysis)	Contém modelos com a estrutura das informações relacionadas com a análise de viabilidade econômica do projeto.
	Marketing / Lançamento (Marketing/Launch)	Contém modelos contendo as informações relacionadas com o mercado e as atividades de lançamento do produto.
	Especificação do Produto (Product Specification)	Contém modelos que descrevem as informações relacionadas com a especificação do produto. Inclui-se os requisitos do projeto também;
	Processo de Fabricação (Production Process)	Contém modelos que descrevem as informações relacionadas com o processo de fabricação;
	Gerenciamento de Projeto (Project Management)	Contém modelos que descrevem as informações relacionadas com o gerenciamento do projeto.
Requisitos de Conhecimento		Não há subdivisão, havendo um único modelo que mostra o mapa de conhecimentos, isto é, o conjunto de requisitos de conhecimentos devidamente hierarquizados
Organização	Estrutura Organizacional (Organizational Structure)	Contém os modelos que descrevem a estrutura organizacional da empresa, isto é, os organogramas.
	Times (Teams)	Contém modelos que descrevem a composição dos diversos times de projeto que fazem parte do processo de desenvolvimento de produto.
Recursos	Materiais (Materials)	Contém os modelos que descrevem os recursos do tipo materiais, isto é, equipamentos, instrumentos, dispositivos, sistemas de medida e outras ferramentas empregadas no processo de desenvolvimento de produto e que não são do tipo <i>software</i> (sistemas).
	Sistemas (Systems)	Contém modelos que descrevem a estrutura dos diferentes tipos de sistemas (<i>softwares</i>) empregados no processo de desenvolvimento de produto.
Processos	Diretório Raiz	Contém os modelos do tipo VAC e eEPC, que representam respectivamente a visão geral do processo e a descrição do fluxo de atividades de cada uma das fases
	Activity Details	Contém os diagramas do tipo Function Allocation Diagram (FAD), os quais detalham cada uma das fases.

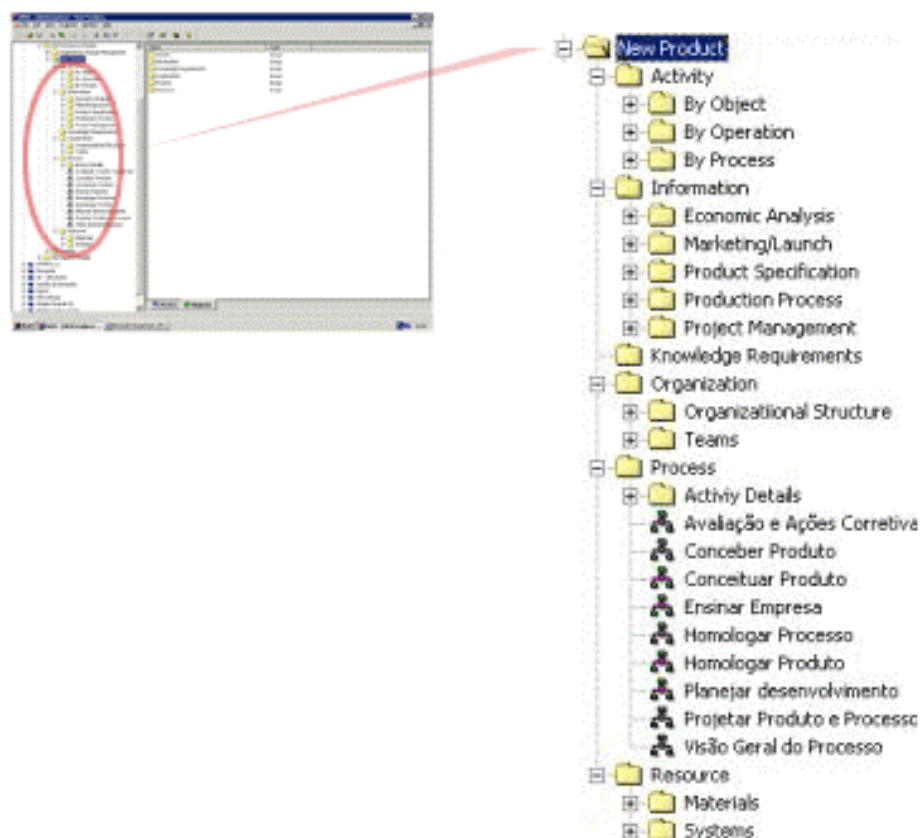


Figura 5. 26: Estrutura de diretórios padrão na ferramenta ARIS

Somando o filtro e a estrutura de diretórios padronizada, o usuário pode navegar nos modelos, seguindo as 6 diferentes visões previstas no *framework* de modelagem, conforme apresentado no capítulo 4.

5.3.3 Macro para a geração de Relatório

Uma vez construído um modelo com o auxílio da ferramenta, é possível gerar relatórios que descrevem as informações (atributos) adicionadas aos modelos, relacionamentos e construtos. Existe um conjunto de relatórios padrão que acompanha a ferramenta, mas, indo além, a ferramenta permite que sejam escritos relatórios “customizados” e sob medida por meio de “macros” (programas anexados ao sistema) na linguagem *Visual Basic Application*.

Esta versão do ARIS Toolset (4.0) possui um modelo de objetos aberto, sendo cada um deles, capaz de ser acessado por meio de macros. Por exemplo, existem métodos do objeto modelo que permitem imprimir todos os construtos (objetos) presentes em um determinado modelo. Empregando-os, é possível escrever um programa (em uma “macro” de relatório) que imprime todos os construtos presentes

no modelo. Assim, pode-se programar um relatório sob medida para uma aplicação, contendo todas as informações que se julgar pertinentes, em diferentes seqüências e com uma programação visual (*layout*) específica. Uma vez construída a macro, basta escolher o diretório do modelo que se deseja imprimir e acioná-la para que seja emitido um relatório do modelo dentro desse padrão pré-definido.

Para facilitar o armazenamento dos modelos, essa potencialidade da ferramenta foi utilizada, gerando um relatório padrão para ser empregado na impressão dos modelos especificados pela ferramenta. Estes relatórios são armazenados em arquivo e podem, também, ser anexados à intranet. Na figura 5.27 apresenta-se a janela empregada para iniciar a geração do relatório. Na figura 5.28, apresenta-se, a título de exemplificação, partes de código de uma macro para relatório do ARIS.

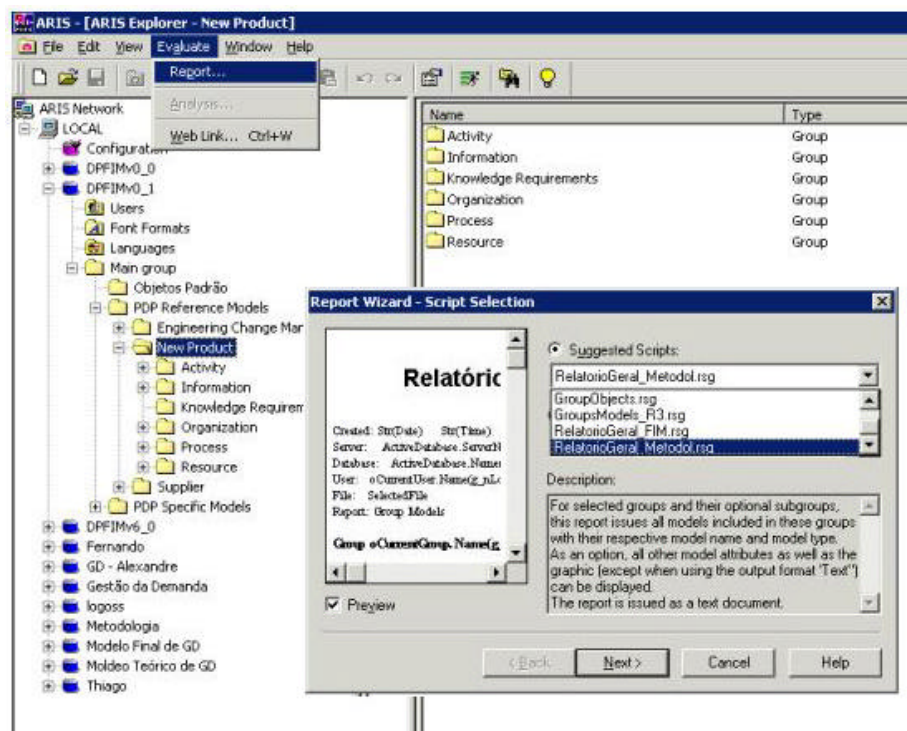
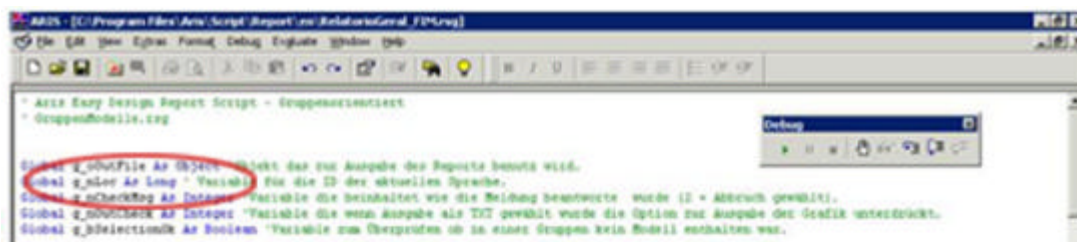


Figura 5. 27: Tela de início da geração do relatório



Global g_oOutFile As Object...

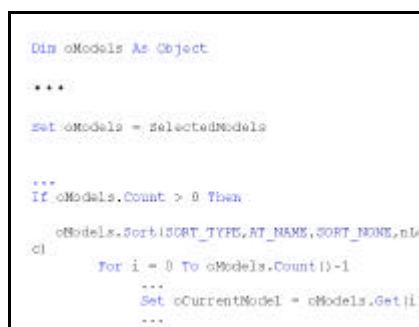


Set g_OutFile = CreateObject (ARIS Output)



```
g_oOutFile.DefineF("REPORT1", "Arial", 24, RGB(0,0,0)
,COLOR_TRANSPARENT,FMT_BOLD Or FMT_CENTER,0,21,0,0,
0,1) g_oOutFile.OutputLnF("ARIS Report", "REPORT1")
```

(a) Segmentos da macro onde é criado um objeto “Saída”



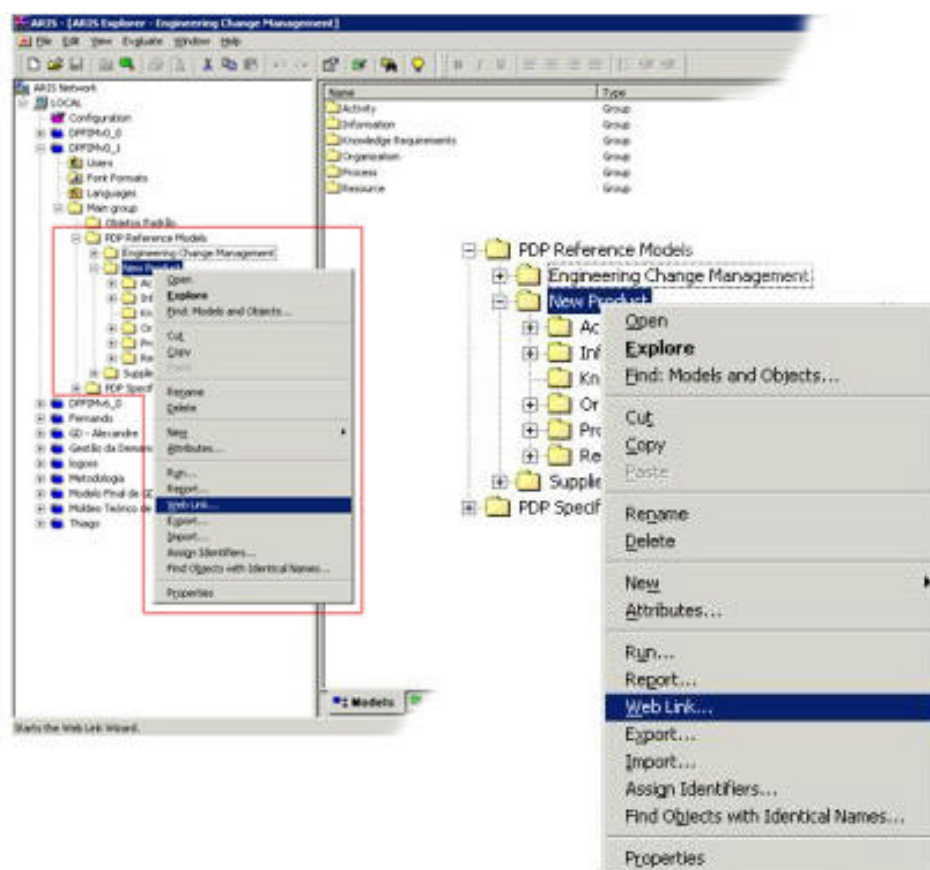
(b) Segmentos de macro para a criação de uma coleção de “Modelos” e a impressão de cada um deles.

Figura 5. 28: Macro para a geração de relatórios de modelos

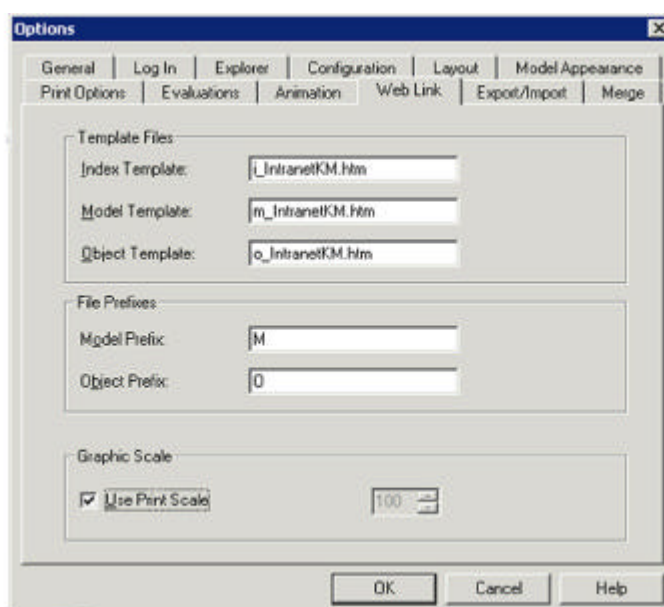
5.3.4 Modelos de Página WEB

Na versão adquirida da ferramenta de modelagem ARIS Toolset, há um módulo denominado *WebLink*, que gera páginas HTML dos modelos armazenados. Esse módulo precisa ser adquirido à parte e permite gerar um conjunto de páginas relacionadas entre si por *links*, as quais contêm o modelo descrito. Esses *links* mantêm relacionamentos idênticos aos do modelo originalmente armazenado na ferramenta. Caso um construto que representa, por exemplo, uma determinada atividade (como desenhar) possuir um eEPC (extended Event Process Chain), que a explica com detalhes e está anexada (*assign*) a ela no modelo original, surgirá um *link* nas páginas web, ligando esse objeto atividade ao modelo anexado. O usuário poderá, a partir do *browser*, “apontar” para esse *link* na página do objeto atividade e “navegar” em direção a este modelo detalhado.

A figura 5.29 mostra o menu utilizado para a geração das páginas na web. Escolhe-se o diretório contendo o(s) modelo(s) que se deseja transcrever para web e inicia-se, a partir do menu, um assistente que gerará as páginas, após solicitar alguns parâmetros como o formato com que se deseja gerar as figuras (JPEG ou GIF) e outros. Estas páginas são geradas em um diretório específico, que pode ser escolhido pelo usuário, de forma que, estando este diretório em um servidor web ou servidor de uma intranet, essas páginas tornam-se, conseqüentemente, disponíveis aos usuários que acessam o site na internet ou a intranet.



(a) Tela para inicialização da geração dos modelos na WEB



(b) Tela para especificação dos arquivos-padrão que servem de modelos para a geração das páginas WEB.

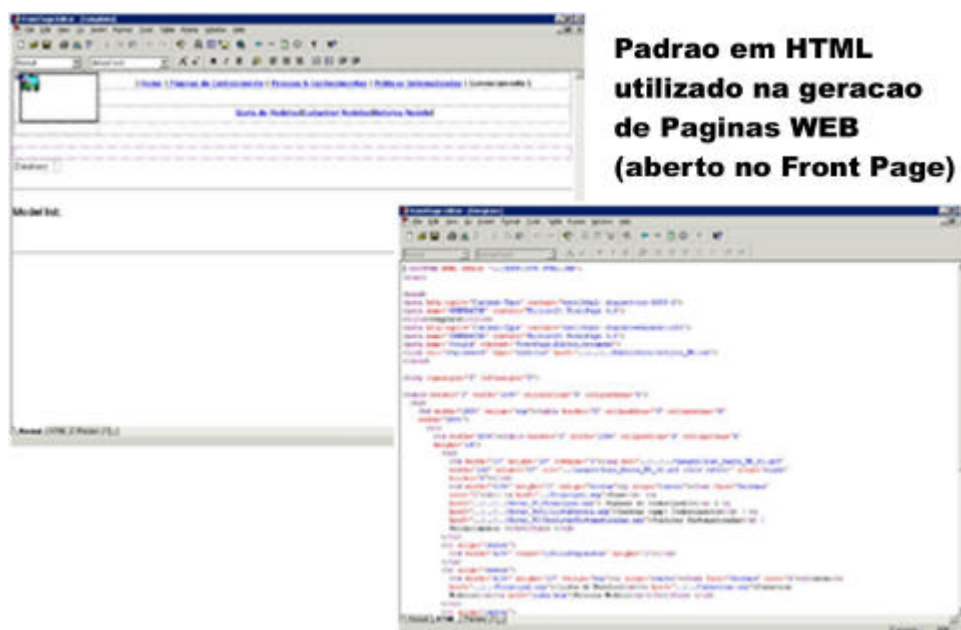
Figura 5.29 : Menu da ferramenta ARIS empregado na geração de modelos em HTML

Este módulo é, na verdade, um objeto que gera essas páginas baseadas em três arquivos modelos (*templates*) em HTML: o primeiro que orienta a construção da página inicial, um modelo para as páginas que descrevem modelos e um de páginas que descrevem os construtos (objetos no sistema). Assim, a estrutura do *site* gerado por este assistente é sempre a mesma, contendo uma página inicial, páginas de modelo e páginas de construto (objetos). Na página inicial, aparece uma janela (um objeto *ActiveX*) que simula a árvore de diretórios da ferramenta, denominada “ARIS explorer”. Abaixo dessa janela, aparece uma lista de *links* para todos os modelos, em ordem alfabética. Desta janela, ou desses *links*, pode-se adentrar as páginas de modelo ou objeto.

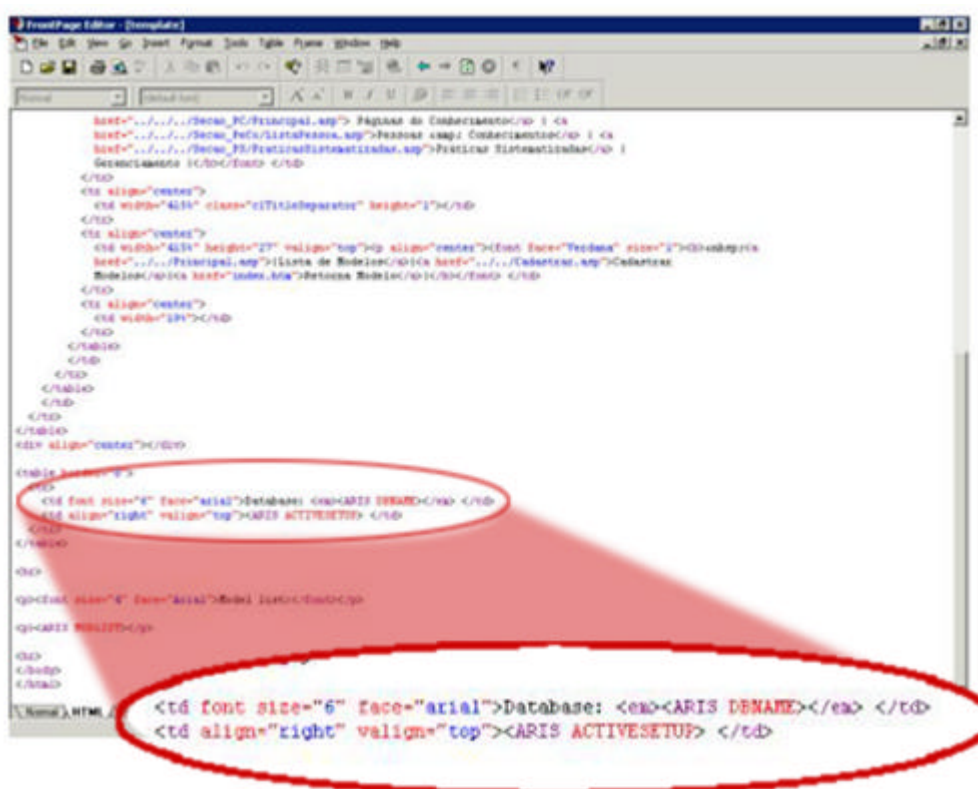
As páginas de modelo, por padrão inicial da ferramenta, contêm o diagrama do modelo e listas, contendo todos os atributos deste modelo, os construtos que pertencem a ele (objetos) e o relacionamento com outros modelos. Recursivamente, seja apontando para as representações dos construtos contidos na figura do modelo ou diretamente da lista de construtos, é possível “navegar” em direção às páginas dos construtos (objetos) ou outras páginas de modelo.

As páginas de construtos (objetos) contêm, como padrão inicial: uma lista dos atributos do construto específico; uma lista dos relacionamentos destes construtos com cada um dos outros construtos e uma lista de ocorrências, isto é, dos modelos nas quais este construto é referenciado. Todas essas listas contêm *links* que podem ser acessados para se chegar até estes outros objetos ou modelos.

Os arquivos *templates*, utilizados pelos usuários para a personalização (“customização”) do *site* gerado pela ferramenta, são arquivos HTML, conforme exemplificado na figura 5.30. Pode-se adicionar estilos, cores e alterar a formatação das páginas, simplesmente formatando tais páginas modelo como se fossem uma página HTML tradicional. Por exemplo, com a ajuda de uma ferramenta de edição em HTML tradicional como *Front Page* ou *Dreamweaver* (veja a figura 5.30 a). A única distinção em relação a uma página tradicional da *web* é que essas páginas modelo (*templates*) contêm “*tags*” (instruções na linguagem HTML) especiais que são utilizadas somente pela ferramenta ARIS. Um exemplo de *tag* especial é apresentada na figura 5.30b. Trata-se de uma *tag* que imprime na página o título da base de dados.



(a) Visão geral da página modelo



(b) Exemplo de tag específica do ARIS

Figura 5. 30: Exemplo de arquivo-padrão em HTML empregado pelo ARIS na geração de páginas WEB de modelos de processo

Portanto o módulo *Weblink* da ferramenta ARIS cria páginas de modelo e objetos inserindo, dentro destas *tags* especiais, as listas e *links* conforme representados nos modelos. Os elementos padrão são mantidos, da mesma forma que os cabeçalhos e outros *links* que existirem no arquivo *template*. Os elementos essenciais da página, como as listas de atributos, de modelos e imagens das representações gráficas dos modelos, são gerados dinamicamente pelo módulo da ferramenta. O resultado final é um conjunto de páginas estáticas que representam todo o modelo descrito na ferramenta.

O primeiro passo da adaptação do módulo de *Weblink* da ferramenta, em direção a adequá-la à solução, consistiu, portanto, na criação de 3 novos arquivos modelos (*templates*) para a geração de páginas WEB: um para a páginas inicial, um para a página de modelo e um para a página de construto (objeto). A criação dessas páginas foi realizada conforme a programação visual padrão da intranet e omitindo “*tags*” especiais do ARIS, de forma a manter, apenas, as informações essenciais para a interpretação do modelo. Portanto algumas listas, como a de relacionamento entre objetos foram eliminadas dos modelos (*templates*).

Excepcionalmente, uma intervenção maior mostrou-se necessária na página inicial. O conjunto de páginas geradas sobre um mesmo modelo, o *site* do modelo, deveria refletir a mesma estrutura de visões definida no *framework* de modelagem, apresentado no capítulo 4. Isso significa poder “navegar” por entre tais páginas segundo a lógica das 6 visões. O objeto que simula o “ARIS Explorer”, presente na primeira página, mostrou-se bastante lento e também confuso. Incluía outras informações além das visões, tais como os diretórios de bases de dados do ARIS. Seria bastante difícil para o usuário final navegar nas visões do *framework*, empregando a estrutura de diretórios-padrão, conforme é feito pelo modelador e administrador de modelos, quando estes utilizam a ferramenta ARIS em si. Seria necessário, portanto, uma alteração radical na estrutura dessa primeira página, que permitisse a visualização das seis visões.

As *tags* específicas do ARIS para a criação do site são bastante restritas nessa versão da ferramenta. Era impossível, portanto, personalizar a primeira página de forma a organizar os modelos conforme as 6 visões por meio da personalização destas *tags*. A única solução possível foi alterar esta página depois de gerado o site

contendo todas as páginas do modelo. Do *template* original, somente a lista de models foi mantida, sendo eliminado, inclusive, o objeto “ARIS-Explorer”. Assim, depois de gerado o site, emprega-se uma ferramenta de edição de páginas web comum para organizar os links dos modelos gerados por ordem alfabética, agrupando-os em uma tabela conforme as 6 visões. Para que o usuário tivesse acesso a uma visão geral do processo foi adicionado o diagrama VAC (*Value Added Chain*), que descreve as fases gerais do processo (veja item 4.5.3), acima da tabela. Segundo esta estratégia, portanto, a página inicial recebe um tratamento diferenciado sendo reconstruída, depois de gerada, para obter-se o resultado apresentado na figura 5.31 (esse é o modelo da metodologia de modelagem). As figuras 5.32 e 5.33 apresentam, respectivamente, exemplos de uma página de modelo e uma página de objeto. Note que as páginas possuem poucas informações, pois contemplam apenas aquelas consideradas essenciais.

The screenshot shows a web browser window displaying a software interface. At the top, there is a navigation bar with the text 'Modelo de Referência' and a menu with options: 'Home / Página de Conteúdos / Processos & Conteúdos / Ferramentas / Gerenciamento'. Below this, the page title is 'Lista de Modelos (Conteúdo e Modelos) Para os Modelos'. The main content area is titled 'Visão Geral do Modelo:' and features a diagram of a process flow. The flow starts with a large green arrow labeled 'Modelo Processo de Desenvolvimento de Projeto', which branches into four smaller green arrows: 'Planeja Projeto', 'Controla Modelo', 'Implementa Modelo', and 'Monitora Projeto'. Below the diagram is a small table with the following rows: 'Planeja', 'Controla Modelo', and 'Implementa Modelo'. The main part of the page is a large table with two columns: 'Visão' and 'Modelos'. The 'Visão' column lists categories like 'Informação', 'Organização', 'Recursos', 'Recursos', 'Conteúdos', 'Atividade', and 'Processo'. The 'Modelos' column lists specific tasks and sub-tasks for each category. For example, under 'Informação', it lists 'Análise do uso do Modelo de Referência', 'Atualização de lista', 'Criação de lista', 'Configuração de modelos', 'Controlar Modelos', 'Conteúdo de modelos', 'Dados projeto', 'Diretrizes', 'Gate E1', 'Gate E2', 'Gate E3', 'Gate E4', 'Lista de atividades', 'Lista de direções', 'Modelo de referência', 'Nome', 'Propriedades', 'Registral de objetos', and 'Estruturação de Processo'. Under 'Atividade', it lists 'Lista de atividades por Fase', 'Controlar Modelos', 'Dados Projeto', 'Implementar Modelo', 'Planejar Projeto', 'Detalhar o(a) Atividade', 'Adaptar processo de desenvolvimento', 'Analisar modelo atual', 'Aprovar o modelo a ser usado no projeto', 'Aplicar projeto', 'Atualizar direções de modelo', 'Classificar o projeto de desenvolvimento', 'Comparar modelos existentes', 'Controlar EPC's', 'Controlar VAC', 'Controlar atividades em família', 'Controlar organização', 'Controlar direção de função', 'Criação planejada modelo em lista', 'Dados relativos ao modelo', 'Dados relativos a responsabilidades', 'Dados relativos a custos de manutenção', 'Dados relativos', 'Dados relativos de distribuição e controle', 'Modelar Item Projeto', 'Identificar requisitos primários', 'Identificar requisitos de integração de recursos', 'Linha de tempo de execução', 'Preparar funcionalidades de projeto', 'Preparar análise de estabilidade econômica', 'Preparar diagrama de rede', 'Preparar processo de implementação e testes', 'Realizar teste E1', 'Realizar teste E2', 'Realizar teste E3', 'Realizar teste E4', 'Realizar teste E5', 'Realizar teste E6', 'Realizar teste E7', 'Realizar teste E8', 'Realizar teste E9', 'Realizar teste E10', 'Realizar teste E11', 'Realizar teste E12', 'Realizar teste E13', 'Realizar teste E14', 'Realizar teste E15', 'Realizar teste E16', 'Realizar teste E17', 'Realizar teste E18', 'Realizar teste E19', 'Realizar teste E20', 'Realizar teste E21', 'Realizar teste E22', 'Realizar teste E23', 'Realizar teste E24', 'Realizar teste E25', 'Realizar teste E26', 'Realizar teste E27', 'Realizar teste E28', 'Realizar teste E29', 'Realizar teste E30', 'Realizar teste E31', 'Realizar teste E32', 'Realizar teste E33', 'Realizar teste E34', 'Realizar teste E35', 'Realizar teste E36', 'Realizar teste E37', 'Realizar teste E38', 'Realizar teste E39', 'Realizar teste E40', 'Realizar teste E41', 'Realizar teste E42', 'Realizar teste E43', 'Realizar teste E44', 'Realizar teste E45', 'Realizar teste E46', 'Realizar teste E47', 'Realizar teste E48', 'Realizar teste E49', 'Realizar teste E50', 'Realizar teste E51', 'Realizar teste E52', 'Realizar teste E53', 'Realizar teste E54', 'Realizar teste E55', 'Realizar teste E56', 'Realizar teste E57', 'Realizar teste E58', 'Realizar teste E59', 'Realizar teste E60', 'Realizar teste E61', 'Realizar teste E62', 'Realizar teste E63', 'Realizar teste E64', 'Realizar teste E65', 'Realizar teste E66', 'Realizar teste E67', 'Realizar teste E68', 'Realizar teste E69', 'Realizar teste E70', 'Realizar teste E71', 'Realizar teste E72', 'Realizar teste E73', 'Realizar teste E74', 'Realizar teste E75', 'Realizar teste E76', 'Realizar teste E77', 'Realizar teste E78', 'Realizar teste E79', 'Realizar teste E80', 'Realizar teste E81', 'Realizar teste E82', 'Realizar teste E83', 'Realizar teste E84', 'Realizar teste E85', 'Realizar teste E86', 'Realizar teste E87', 'Realizar teste E88', 'Realizar teste E89', 'Realizar teste E90', 'Realizar teste E91', 'Realizar teste E92', 'Realizar teste E93', 'Realizar teste E94', 'Realizar teste E95', 'Realizar teste E96', 'Realizar teste E97', 'Realizar teste E98', 'Realizar teste E99', 'Realizar teste E100'. Under 'Processo', it lists 'Dados projeto', 'Planejar projeto', 'Controlar Modelos', and 'Implementar Modelo'.

Visão	Modelos
Informação	<ul style="list-style-type: none"> Análise do uso do Modelo de Referência Atualização de lista Criação de lista Configuração de modelos Controlar Modelos Conteúdo de modelos Dados projeto Diretrizes Gate E1 Gate E2 Gate E3 Gate E4 Lista de atividades Lista de direções Modelo de referência Nome Propriedades Registral de objetos Estruturação de Processo
Organização	<ul style="list-style-type: none"> Cela Organizacional Temas
Recursos	<ul style="list-style-type: none"> Item do modelo
Recursos	<ul style="list-style-type: none"> Item do modelo
Conteúdos	<ul style="list-style-type: none"> Item do modelo
Atividade	<ul style="list-style-type: none"> Lista de atividades por Fase Controlar Modelos Dados Projeto Implementar Modelo Planejar Projeto Detalhar o(a) Atividade Adaptar processo de desenvolvimento Analisar modelo atual Aprovar o modelo a ser usado no projeto Aplicar projeto Atualizar direções de modelo Classificar o projeto de desenvolvimento Comparar modelos existentes Controlar EPC's Controlar VAC Controlar atividades em família Controlar organização Controlar direção de função Criação planejada modelo em lista Dados relativos ao modelo Dados relativos a responsabilidades Dados relativos a custos de manutenção Dados relativos Dados relativos de distribuição e controle Modelar Item Projeto Identificar requisitos primários Identificar requisitos de integração de recursos Linha de tempo de execução Preparar funcionalidades de projeto Preparar análise de estabilidade econômica Preparar diagrama de rede Preparar processo de implementação e testes Realizar teste E1 Realizar teste E2 Realizar teste E3 Realizar teste E4 Realizar teste E5 Realizar teste E6 Realizar teste E7 Realizar teste E8 Realizar teste E9 Realizar teste E10 Realizar teste E11 Realizar teste E12 Realizar teste E13 Realizar teste E14 Realizar teste E15 Realizar teste E16 Realizar teste E17 Realizar teste E18 Realizar teste E19 Realizar teste E20 Realizar teste E21 Realizar teste E22 Realizar teste E23 Realizar teste E24 Realizar teste E25 Realizar teste E26 Realizar teste E27 Realizar teste E28 Realizar teste E29 Realizar teste E30 Realizar teste E31 Realizar teste E32 Realizar teste E33 Realizar teste E34 Realizar teste E35 Realizar teste E36 Realizar teste E37 Realizar teste E38 Realizar teste E39 Realizar teste E40 Realizar teste E41 Realizar teste E42 Realizar teste E43 Realizar teste E44 Realizar teste E45 Realizar teste E46 Realizar teste E47 Realizar teste E48 Realizar teste E49 Realizar teste E50 Realizar teste E51 Realizar teste E52 Realizar teste E53 Realizar teste E54 Realizar teste E55 Realizar teste E56 Realizar teste E57 Realizar teste E58 Realizar teste E59 Realizar teste E60 Realizar teste E61 Realizar teste E62 Realizar teste E63 Realizar teste E64 Realizar teste E65 Realizar teste E66 Realizar teste E67 Realizar teste E68 Realizar teste E69 Realizar teste E70 Realizar teste E71 Realizar teste E72 Realizar teste E73 Realizar teste E74 Realizar teste E75 Realizar teste E76 Realizar teste E77 Realizar teste E78 Realizar teste E79 Realizar teste E80 Realizar teste E81 Realizar teste E82 Realizar teste E83 Realizar teste E84 Realizar teste E85 Realizar teste E86 Realizar teste E87 Realizar teste E88 Realizar teste E89 Realizar teste E90 Realizar teste E91 Realizar teste E92 Realizar teste E93 Realizar teste E94 Realizar teste E95 Realizar teste E96 Realizar teste E97 Realizar teste E98 Realizar teste E99 Realizar teste E100
Processo	<ul style="list-style-type: none"> Dados projeto Planejar projeto Controlar Modelos Implementar Modelo

Figura 5. 31: Página Inicial

http://www.nansa.org.br/procad/Service_Files/Arquivos/71740164.Mini - Microsoft Internet Explorer Ferramentas por Teófilo atalla

Arquivo Editar Editar Favoritos Ferramentas Ajuda

Endereço http://www.nansa.org.br/procad/Service_Files/Arquivos/71740164.Mini

Modelo de Referência | Home | Páginas de Conhecimento | Pessoas & Conhecimentos | Práticas Gerenciais/Gráficas | Gerenciamento |
Lista de Modelos/Cadastrar Modelos (Retorna Modelo)

Visão Geral do Processo
Model: *Visão Geral do Processo*
Value added chain diagram

Model graphic:

Lista de Objetos do Modelo:

Função	Entoar Empresa
Função	Homologar Processo
Função	Homologar Produto
Função	Projetar Produto e Processo
Função	Conceber produto
Função	Conceber Produto
Função	Desenvolver Produto Modular
Função	Analisar e Atuar Agilizar Corretivas

Atributos do Modelo:

Name Visão Geral do Processo

Figura 5. 32: Exemplo da página do tipo modelo gerada depois da customização

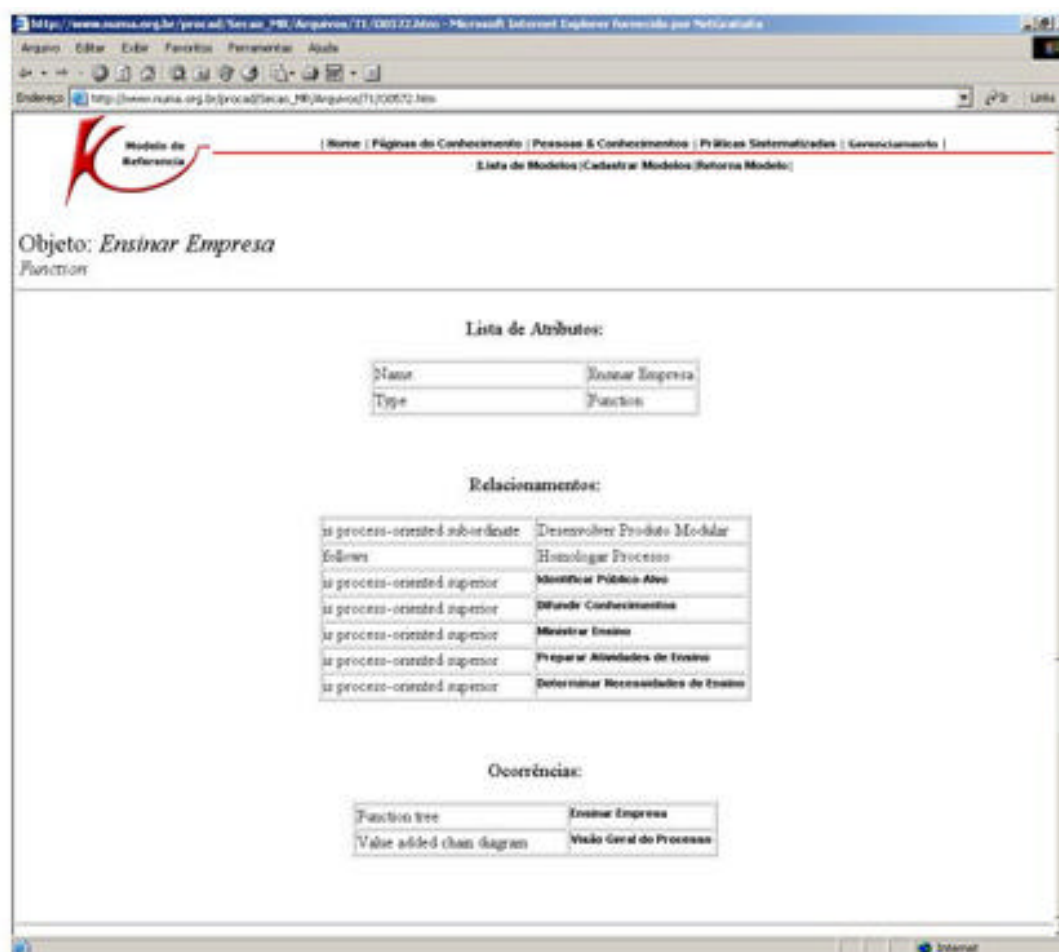


Figura 5. 33: Exemplo de página de um objeto gerada depois da customização

5.3.5 Integração entre a ferramenta de modelagem e a intranet

A integração entre as páginas geradas e a intranet é garantida por meio de uma página especial denominada “Página de Busca” (figura 5.34) e do procedimento representado esquematicamente na figura 5.35. Quando o usuário cria um modelo novo, empregando a ferramenta ARIS customizada, ele tem à sua disposição um conjunto de objetos padrões, conforme especificado no *framework* de modelagem. Entre eles, há um tipo específico de construto (um tipo de objeto), presente na dimensão conhecimento, que é denominado categoria de conhecimento (*knowledge category*), conforme apresentado no capítulo 4. O primeiro passo para a existência da integração, portanto, é a preparação destes objetos padrão (veja figura 5.35). Eles representam requisitos de conhecimento para o desenvolvimento da atividade. Deve existir, no mínimo, um objeto categoria de conhecimento para cada uma das

palavras-chave principais do sistema. Esse construto possui 2 atributos que são do tipo *link*, os quais podem ser vistos na caixa de cadastro da ferramenta (1º. passo da figura 5.35). No segundo deles, por padrão, deve-se inserir o link para a página de busca, o qual é formado pelo link da página, seguido do caractere “?” e o código de identificação da palavra-chave que ele representa no sistema (o Id). Os administradores de conhecimento e os administradores de modelo têm acesso a este código dentro do módulo de gerenciamento do sistema. No caso empregado como exemplo na figura 5.32, o objeto é denominado “QFD” e o código de identificação é 72. O link cadastrado no objeto é, portanto: “http://numa.org.br/intranetkmv3/PaginaDeBusca.asp? IdPalavraChave =72”.

The screenshot displays a web application interface for a search page. The browser window title is "PaginaDeBusca - Microsoft Internet Explorer fornecido por NetGratuita". The address bar shows the URL: "http://www.numa.org.br/intranetkmv3/Secao_MR/PaginaDeBusca.asp?IdPC=72".

The page content includes a navigation menu at the top: "Home | Páginas do Conhecimento | Pessoas & Conhecimentos | Modelos de Referência | Gerenciamento". Below the menu, there are three main sections:

- Registros**: A table with columns: Título, Subclasse, Data, Download, URL.

Título	Subclasse	Data	Download	URL
QFD - Quality function deployment	Apresentação	4/11/01	s	s
QFD (Quality function deployment)	Apresentação	4/16/01	s	s
Aplicação de QFD na empresa X	Caso	5/28/01	s	s
Aplicação de QFD no desenvolvimento de um motor de Y	Artigo Seriado	5/28/01	s	s
QFD: planejamento da Função Qualidade	Livro	5/28/01	s	s
Introdução ao desdobramento da Função Qualidade	Livro	5/28/01	s	s
João da Silva	Especialista	9/19/01	s	s
- Sentença**: A table with columns: Título, Tipo, Validacao, Data, Autor, Download URL.

Título	Tipo	Validacao	Data	Autor	Download URL
A primeira casa do QFD é a mais utilizada	Enunciado	2.5	9/19/01	3	
SE A interface do produto com o usuário é complexa ENTÃO deve-se aplicar o QFD	Enunciado	1.8	9/19/01	3	
SE QFD é empregado utilizando-se o modelo conceitual ENTÃO obtém-se os melhores resultados	Enunciado	0.6	9/19/01	3	
SE a interface do produto com o usuário é complexa ENTÃO deve-se empregar a primeira casa da qualidade	Argumento	0.7	9/19/01	3	
- Página do Conhecimento**: A table with columns: Título, Tipo, Descrição.
- Usuarios**: A table with columns: Nome, Tipo, Endereço, Telefone.

Nome	Tipo	Endereço	Telefone
Daniel C. Amaral	Adm. do Conhecimento		

Figura 5. 34: Página de busca - responsável pela integração entre as páginas de modelo e os conhecimentos explícitos armazenados na intranet

O segundo passo é a construção dos modelos, empregando a ferramenta ARIS e o *framework* de modelagem proposto. Como consequência da aplicação do *framework*, os objetos padrão serão utilizados na representação. Por fim, na etapa 3, basta gerar as páginas HTML de modelos, com o auxílio da ferramenta ARIS. Isso é

feito selecionando-se a pasta (“*folder*”) em que se encontra o modelo e iniciando-se o módulo de “*Weblink*”, conforme apresentado na figura 5.29a . Ao fim da geração do site com o modelo de referência, o modelador poderá então entrar na intranet e efetuar o cadastro do modelo indicando título, versão, formato e autores, conforme os dados descritos no item 5.2. Como último passo deste cadastro o usuário é solicitado a realizar o procedimento de *upload*, no qual poderá enviar os arquivos em html do modelo via internet.

Uma forma alternativa de cadastrar o modelo na intranet, válida somente para usuários com acesso direto ao servidor, é inverter essas duas etapas, cadastrando o modelo na intranet, antes que os arquivos HTML do modelo sejam gerados. Esse procedimento só é possível porque, ao final do procedimento de cadastro, o usuário é avisado sobre o número de identificação do modelo (Id do modelo). Este número é idêntico ao nome da pasta no diretório da intranet no qual o modelo deveria estar armazenado, e que, ao final do processo de cadastro, é criada pelo sistema automaticamente (“Seção_MR/Arquivos”). Assim, basta que o usuário anote esse número e gere os arquivos HTML do modelo a partir da própria ferramenta, direcionando a criação de tais páginas para o diretório, com o número do Id do modelo. No exemplo apresentado na figura 5.34, é o número 63. Esse procedimento foi mais utilizado durante a aplicação da solução, e é preferível porque a quantidade e o tamanho dos arquivos gerados pela ferramenta ARIS são elevados, dificultando o envio por meio de *upload* através da Intranet.

Independentemente do procedimento de geração do modelo, desde que esse procedimento tenha sido realizado corretamente, o modelo estará pronto para ser utilizado. Se ainda não estiver claro para o leitor, basta lembrar que, quando a ferramenta ARIS gera páginas html de modelos, automaticamente, os atributos do tipo *link* são transformados em *links* nas páginas do objeto ou do modelo. Portanto a página que descreve os atributos do construto (objeto) *knowledge Category* estará, automaticamente, apresentando um *link* para a Página de Busca apresentada na figura 5.34. Como a Página de Busca contém uma consulta sobre todos os conhecimentos explícitos cadastrados no sistema com aquela palavra-chave (modelos, registros, sentenças, páginas do conhecimento e usuários) tem-se, enfim, a integração unidirecional entre modelo e conhecimentos explícitos armazenados.

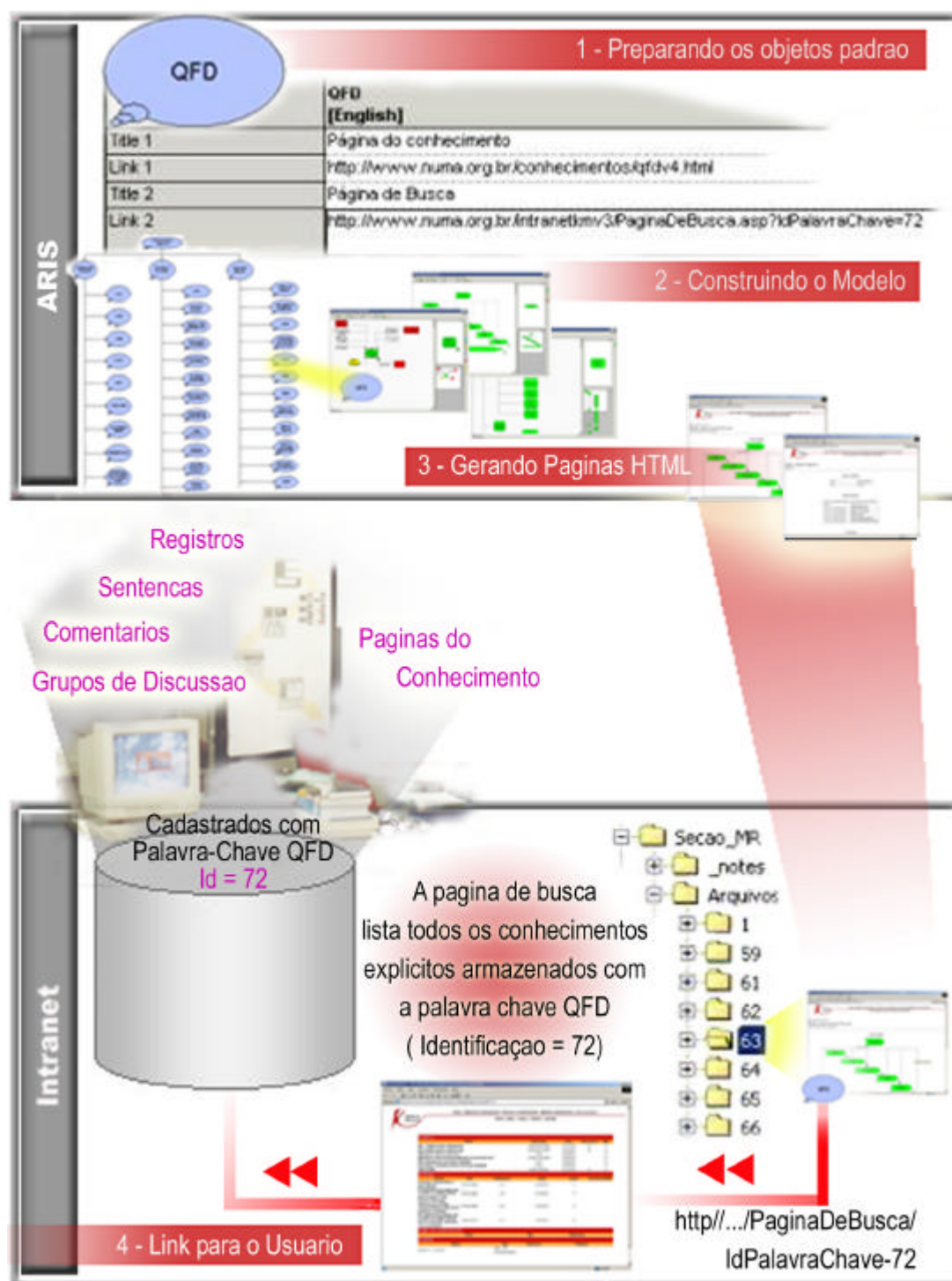


Figura 5. 35: Integração dos modelos com a intranet

Resta esclarecer os detalhes técnicos dessa integração. Ela é realizada, de maneira simples, empregando-se um objeto Request que é criado no servidor ASP (*Active Server Pages*), mais especificamente pelo IIS. O caractere “?” indica ao servidor que a palavra seguinte, o nome `IdPalavraChave`, é uma variável a ser criada no objeto Request, cujo valor atribuído é 72. A Página de Busca é programada para

fazer buscas localizando os conhecimentos que possuem uma palavra-chave com número de identificação igual à variável *IdPalavraChave*. Portanto na prática este link faz com que o browser solicite ao servidor a Página de Busca (*PaginaDeBusca.asp*) listando todos os conhecimentos explícitos que estejam relacionados com a palavra-chave ou que venham a ser criados no futuro também com esse relacionamento.

Uma vez criado o conjunto de objetos padrão, basta que o modelador empregue um desses objetos em seu modelo, para que o *link* entre modelo e conhecimentos explícitos passe a ser válido, sem haver necessidade que o usuário construtor do modelo faça qualquer operação adicional. Na verdade, ele não precisa saber o ID da palavra-chave e nem mesmo criar qualquer tipo de *link*. Ele apenas usa o objeto padrão no modelo e gera as páginas HTML; conseqüentemente, para toda a ocorrência do objeto padrão, estará disponível o *link* para a Página de Busca.

Logicamente, há a necessidade de uma coordenação para garantir o funcionamento coerente dessa integração. É preciso garantir, principalmente, três aspectos: 1) que os objetos-padrão do tipo requisitos de conhecimentos possuam, em seu atributo *link*, o caminho correto para a página de busca (*PaginaDeBusca.asp*); incluindo o código de identificação da palavra-chave; 2) que os objetos padrão do tipo *Knowledge Category* sejam empregados na modelagem e 3) que os modelos gerados sejam direcionados para o diretório correto da Intranet. Como isto depende, fundamentalmente, da disciplina dos usuários do tipo administrador de modelos ou modeladores, foi elaborada uma folha de procedimento para a realização dessa atividade. Esse procedimento, portanto, faz parte da solução e sua observância garante que a integração entre modelo e conhecimentos explícitos possa funcionar com pleno êxito.

6. Exemplo de Aplicação da Solução

Este capítulo descreve a aplicação da solução apresentada no capítulo anterior, a qual foi realizada em um grupo de pesquisa, com o intuito principal de possibilitar uma primeira avaliação da solução desenvolvida. A apresentação desse exemplo é dividida em 3 itens. O item 6.1 apresenta o contexto da aplicação, incluindo os objetivos, critérios de escolha e uma descrição do grupo de pesquisa analisado. No item 6.2, são apresentadas as etapas para o desenvolvimento da solução e no item 6.3, os resultados encontrados. Ao final, no item 6.4, como informação adicional, são apresentados outros dois exemplos, um teórico e outro referente a uma aplicação que se encontra em andamento.

6.1 Contexto da Aplicação

6.1.1 Objetivos e Critérios de Escolha

A aplicação da solução apresentada no capítulo anterior foi realizada visando dois principais objetivos: uma demonstração (exemplificação) do uso da solução e a obtenção de uma primeira avaliação do seu uso. Como se tratava de uma primeira análise, era fundamental que a avaliação fosse a mais minuciosa possível e que todos os detalhes fossem verificados, cobrindo desde aspectos específicos de implementação computacional, até os de nível mais genérico como a adequação aos objetivos principais propostos para o sistema. Esses dois objetivos são fundamentais para que no futuro seja possível aplicá-la num caso mais crítico e de maior extensão, na qual fosse possível uma avaliação mais rigorosa da solução e da arquitetura. Portanto, este estudo de caso visa, antes de mais nada, construir as bases para uma validação mais ampla da arquitetura e da solução desenvolvidas, não havendo a pretensão de verificar definitivamente as hipóteses H2 e H3.

Segundo estes objetivos traçados, os critérios considerados para a escolha do caso foram:

- facilidade e rapidez de implantação;
- exigência de complexidade mínima no gerenciamento de conhecimentos, suficiente para requerer o auxílio de uma ferramenta de gestão de conhecimentos;
- facilidade de acesso às pessoas envolvidas, para facilitar a coleta de dados e

- existência de abertura para uma ampla discussão da aplicação e uso do sistema, visando análises e discussões mais ricas sobre o caso de utilização da ferramenta.

Optou-se, então, por aplicar a ferramenta no próprio grupo de pesquisa de que o pesquisador faz parte, denominado Grupo de Engenharia Integrada (EI), principalmente pelos requisitos de restrição de tempo e de abertura para discussão da solução.

Caso tivesse sido escolhida uma organização mais distante do pesquisador, haveria a necessidade de um período maior de tempo para a personalização (“customização”) do sistema e também para o treinamento das pessoas quanto à sua utilização. Mas a maior barreira seria o tempo necessário para que fosse armazenado um conjunto suficiente de conhecimentos explícitos, necessários para que os usuários pudessem avaliar o sistema contendo um conjunto de conhecimentos cadastrados. Inicialmente, imaginava-se a necessidade de algo em torno de 50 a 100 registros e de 10 a 20 sentenças.

No caso de implantação dentro do Grupo de Engenharia Integrada haveria economia no tempo de implantação, devido ao grau de conhecimento do pesquisador com respeito ao funcionamento do grupo e seus temas principais de atuação. Esse conhecimento permitiria a organização das classes, subclasses e das palavras-chave específicas para o grupo, em um tempo bastante reduzido. Havia, ainda, vantagem na preparação das pessoas para o uso do sistema pois grande parte dos usuários conhecia o trabalho e a proposta da arquitetura. Talvez o que mais pesou nesta decisão tenha sido a existência, dentro do grupo, de um esforço de gerenciamento do conhecimento, havendo um conjunto inicial grande de conhecimentos explícitos armazenados e compartilhados de maneira sistemática. Esse fato permitia que já existissem uma “cultura” e interesses naturais em compartilhar conhecimento. Facilitava, também, a “aceleração” do processo de avaliação pois havia um conjunto de conhecimentos explícitos disponíveis, que estavam cadastrados e organizados em planilhas e arquivos na rede. Estes conhecimentos poderiam ser facilmente empregados para o preenchimento inicial da ferramenta. Outro ponto forte da escolha era a capacidade de se observar e discutir todos os detalhes desta implantação, devido ao acesso do pesquisador aos usuários e também pelo conhecimento profundo do contexto da aplicação, face à experiência do pesquisador em 3 anos de trabalho dentro desse grupo de pesquisa.

As limitações desta escolha eram também bastante evidentes. Sendo o pesquisador e o Grupo responsáveis pela pesquisa, e, portanto, compromissados de alguma maneira com este trabalho, há a tendência natural de avaliar positivamente a proposta tanto da solução como as implicações disso para a arquitetura. Limitação potencialmente significativa, também, era a diferença entre o perfil dos usuários deste grupo, principalmente acadêmico, diferente, portanto, do perfil dos usuários a ser atendidos segundo a proposta da arquitetura: os profissionais de empresa que atuam na área de desenvolvimento de produtos.

Ao final da avaliação considerou-se que, para uma primeira verificação e exemplificação do uso da ferramenta, as vantagens da escolha do caso seriam mais compensadoras que as limitações acima descritas. A tendência de avaliação positiva deixa de ser excessivamente danosa frente ao próprio objetivo do caso que, mais do que obter um juízo definitivo acerca da contribuição ou não da proposta, buscava servir como uma verificação inicial e, mais importante ainda, tinha como objetivo a identificação de pontos para aprimoramento da proposta que permitissem a evolução deste protótipo para um nível suficientemente desenvolvido para ser aplicado em futuros casos, empregando empresas e contextos mais amplos e próximos da realidade de aplicação da arquitetura. Deve-se notar, ainda, que, em um grupo de pesquisa a necessidade de gerenciamento de conhecimento é mais pronunciada e crítica que em uma organização de desenvolvimento de produto de menor porte (semelhantes as que seriam capazes de servir de caso frente ao intervalo de tempo disponível), e que muitas das características típicas de um PDP são encontradas no desenvolvimento de uma pesquisa científica: ambos são processos não estruturados e exigem multidisciplinaridade e trabalho em equipe. Embora o caso escolhido não seja uma organização típica de desenvolvimento de produto, o nível de exigência em termos de gestão do conhecimento foi considerado minimamente suficiente para uma primeira avaliação da arquitetura e da solução.

No próximo item, 6.1.2, descreve-se com maior detalhe o grupo de pesquisa.

6.1.2 Descrição do Grupo de Engenharia Integrada

O grupo de pesquisa em Engenharia Integrada faz pesquisas sobre o processo de desenvolvimento, empregando uma abordagem ampla e integrada. Analisa o gerenciamento do processo como um todo, sistematizando conhecimentos e analisando ferramentas referentes às diversas fases deste processo. Emprega ênfase em sistemas de informação para o gerenciamento desse processo, tais como sistemas

de gerenciamento de informações e dados do produto; sistemas para gerenciamento do projeto e outras.

Uma característica peculiar deste grupo é a realização de pesquisas em conjunto com empresas. Há também o trabalho em times, em que pós-graduandos e estudantes de graduação realizam projetos em conjunto. O grupo possui ainda uma quantidade grande de conhecimentos explícitos já armazenados sobre o assunto. Este grupo, do qual o pesquisador faz parte, era composto no momento da realização do caso, de aproximadamente, 15 pesquisadores, entre 6 alunos de pós-graduação (3 doutorandos e 3 mestrandos) e, sendo os demais, alunos de iniciação científica.

Desde 1999, esse grupo tem aplicado esforços no sentido de gerenciar o conhecimento criado pelos seus membros. Entre as diversas ações neste esforço inicial deve-se destacar:

1. implantação de um sistema de gerenciamento de bibliotecas do laboratório, onde são armazenados todos os artigos, livros e material impresso obtidos ou produzido por membros do grupo;
2. áreas na rede para cadastros de textos e apresentações produzidos ou obtidos pelo grupo, formados por diretórios padrões organizados na rede. Cada texto e apresentação são armazenados, tendo, como nome do arquivo, um código seqüencial. Da raiz desses diretórios constam planilhas onde são registrados apresentações ou textos. A primeira coluna delas mostra o código de identificação do documento e, as demais, vários dados sobre eles, tais como: data de criação, para que foi utilizado, o nome do autor e palavras-chave que o caracterizam. Por meio da planilha pode-se realizar buscas de textos e apresentações por título, tipo, palavra-chave, motivo ou utilização e autores. Estes diretórios contêm atualmente mais de 321 apresentações e 221 textos armazenados pelos diversos membros do grupo e são bastante utilizados;
3. reuniões esporádicas para apresentação de temas em estudo pelos diversos alunos;
4. listas de periódicos importantes em desenvolvimento de produto, que são acompanhados pelo grupo;
5. desenvolvimento de uma lista de palavras-chave padrão utilizadas para o armazenamento de todos os documentos produzidos ou adquiridos pelo grupo de pesquisa e

6. existência de uma intranet, com todos os procedimentos internos do laboratório armazenados;

Ainda que de maneira não totalmente sistematizada, há um certo nível de gerenciamento de conhecimentos explícitos dentro desse grupo, o qual pode ser atestado pela quantidade de textos e apresentações compartilhadas e registradas pelo grupo de pesquisa. Outros materiais, tais como artigos de jornais e revistas, são compartilhados, em menor intensidade, e de maneira predominantemente informal.

6.2 Etapas para a implantação do sistema e realização do exemplo

A aplicação foi conduzida em 4 etapas simples, descritas nos subitens abaixo.

6.2.1 Personalização (Customização) da Solução

A primeira etapa da implantação consistiu da personalização, ou seja, da adaptação do sistema às características da organização em que seria realizado o caso: o Grupo de Engenharia Integrada.

A principal adaptação do sistema consiste do registro de uma estrutura de classe e subclasse para organização dos registros dentro do repositório. A proposição dessa estrutura se baseou, primeiramente, num levantamento de todos os possíveis tipos de conhecimentos explícitos gerados nesta organização, os quais foram listados em uma planilha Excel. Em seguida, por meio de discussões com alguns membros do grupo, essa lista foi refinada e hierarquizada, surgindo, então, as classes e subclasses. Ao final, para cada subclasse, foi escolhida uma estrutura de dados (ficha) que melhor representaria este conhecimento explícito, isto é, identificou-se se era uma referência, entrevista, especialista, caso, padrão ou evento. Obteve-se a estrutura apresentada no quadro 6.1.

Utilizando-se este quadro como referência, a base de dados específica da aplicação para o caso foi devidamente preenchida com os valores de classe, subclasse e tipo. Automaticamente as listas do módulo de Práticas Sistematizadas e a barra lateral da tela principal passaram a apresentar essa classificação. Esperava-se que esta subdivisão fosse natural e significativa para os membros do Grupo de Engenharia Integrada.

Quadro 6. 1 : Classificação de classe e subclasse dos conhecimentos explícitos do Grupo EI

Classe	Subclasse	Tipo de Dado
Bibliografia	Apresentação	Referencia
	Artigo Indexado	Referencia
	Artigo Seriado	Referencia
	Artigos de Congresso	Referencia
	Exemplar de Defesa	Referencia
	Lista de Referência	Referencia
	Lista de Referência Básica	Referencia
	Livro	Referencia
	Página do Conhecimento	Referencia
	Periódico	Referencia
	Plano ou Projeto de Pesquisa	Referencia
	Plano ou Projeto de Pesquisa	Referencia
	Relatório de Pesquisa ou Serviço	Referencia
	Site	Referencia
Documento	Ata de Reunião	Referencia
	Ata de Visita	Padrão
	Documento de Projeto Interno	Referencia
	Entrevista com Especialista	Entrevista
	Entrevista de Pesquisa	Entrevista
	Gates de Projeto de Pesquisa	Referencia
	Padrão Externo	Padrão
	Resultado de Busca	Referencia
Evento	Curso	Evento
	Evento Científico	Evento
	Relatórios	Evento
	Reunião Interna	Evento
Experiência	Caso	Caso
	Caso de Projeto Interno	Caso
	Estratégia de Busca	Referencia
	Relato de Banca	Caso
Ferramenta	Cadastro	Padrão
	Formulário	Padrão
	Método	Referencia
	Procedimento	Padrão
Pessoa	Curriculum	Especialista
	Especialista	Especialista

Outro ponto importante para a “personalização” do sistema é a lista de palavras-chave. Nesse caso, empregou-se uma lista já em uso pelo grupo no armazenamento e controle de livros, textos e apresentações. Portanto tratava-se também de palavras-chave, já de conhecimento prévio dos usuários finais. Bastou incluí-las na base de dados, via o próprio sistema.

No elemento ferramenta de modelagem da solução, não são necessárias personalizações ou adaptações. O único cuidado foi o de permitir acesso aos modelos para os usuários que seriam do tipo modelador, no caso, um dos alunos de iniciação científica do grupo.

Essas atividades relacionadas com a adaptação do sistema foram iniciadas concomitantemente à finalização de seu desenvolvimento, pois, seguindo o modelo de implementação em espiral, assim que o protótipo se tornou maduro o suficiente, foram cadastradas as palavras-chave e também a estrutura de classe-subclasse especificada anteriormente.

Este trabalho foi facilitado por um recurso especial do protótipo: a possibilidade de serem conectadas diferentes bases de dados. O sistema foi programado empregando-se um arquivo de dados de conexão únicos, de forma que, alterando os parâmetros deste arquivo, pode-se alterar a base de dados acessada pelo sistema. Assim, foram criadas duas bases de dados, uma de teste e uma final denominada “Base de Produção”, esta última destinada à primeira aplicação do protótipo. Durante o desenvolvimento, empregava-se a base de testes, permitindo que os programadores inserissem quaisquer dados no sistema para a realização dos testes (“lixos”). Paralelamente, quando o sistema não estava em manutenção, bastava alternar para a base de dados de “Produção”. Nessa base, foram adicionados os dados para a adaptação do sistema ao Grupo de Engenharia Integrada, isto é, a estrutura de classe e subclasse e a lista de palavras-chave apresentadas.

Ao final do mês de março de 2001, foi finalizada a versão 3 do protótipo, considerada como robusta o suficiente para a avaliação da aplicação. Nesse mesmo tempo a base de produção já estava “customizada” com a estrutura de classe e subclasse e as palavras-chaves específicas para o caso de aplicação do grupo de engenharia integrada. Assim, bastava ajustar o sistema para acessar a base em produção, tornando-o apto a ser alimentado.

6.2.2 “Alimentação” inicial do sistema

Com a solução devidamente adaptada às características do grupo, iniciou-se uma alimentação prévia do sistema, para que os usuários tivessem acesso a uma quantidade inicial de conhecimentos explícitos desde o momento inicial de contato.

Essa tarefa foi realizada durante as duas semanas compreendidas entre os dias 26 de março e 09 de abril de 2001. Esse esforço contou com o apoio de 2 funcionários

do laboratório, além do próprio pesquisador. Foram cadastrados artigos, apresentações e textos diversos. Também foram cadastrados os principais modelos de referência sobre o processo de desenvolvimento de produto, mantidos pelo Grupo de Pesquisa.

6.2.3 Treinamento e comunicação aos usuários

Ao final das duas primeiras etapas a solução encontrava-se em funcionamento. Tanto a ferramenta de modelagem como a intranet estavam prontas para ser utilizadas, e os procedimentos necessários para a realização das tarefas se encontravam em diretórios na rede. Havia, ainda, um conjunto inicial de conhecimentos explícitos armazenados. Procedeu-se ao cadastramento de todos os usuários registrando “logins” e senhas para cada um deles. Em seguida, reuniões foram agendadas com alguns dos membros do grupo. O pesquisador e um aluno de iniciação científica, que participava do projeto, apresentavam o sistema e propunham um prazo de algumas horas para que o usuário fizesse uso deste.

A apresentação consistia na descrição do objetivo e do conteúdo dos módulos principais. Algumas operações básicas, como cadastro de registros e de sentenças, foram demonstradas para o usuário. Aos usuários era também informado que se tratava de uma aplicação de testes, que fazia parte do projeto de pesquisa. Portanto estavam eles conscientes de que não se tratava de treinamento para a utilização do sistema, mas de uma primeira avaliação. Ao final da apresentação, solicitava-se que o usuário navegasse e utilizasse o sistema durante um período do dia (uma tarde ou manhã) e que fizesse anotações e sugestões sobre sua utilização.

6.2.4 Avaliação do sistema

Depois de ultrapassado o período estipulado para o uso do sistema, o pesquisador voltava a contatar o usuário e realizava a entrevista de avaliação. Em alguns casos, o usuário se antecipou e procurou o pesquisador para sanar algumas dúvidas. A entrevista era então realizada em frente a um computador acessando a solução. Nela, o usuário era questionado sobre a opinião com relação aos aspectos abaixo listados, os quais foram descritos num roteiro de pesquisa amplo contendo apenas estes itens:

1. facilidade de armazenar e recuperar conhecimentos explícitos;
2. qualidade da interface;
3. desempenho;
4. problemas e sugestões e

5. crença na viabilidade de sua utilização na rotina diária de trabalho.

Ao todo foram treinados e posteriormente avaliados, 4 pessoas, sendo 1 de nível de doutorado, 2 de mestrado e 1 de iniciação científica. Dessa população 2, um doutorando e um de iniciação científica, conheciam bem a proposta e esta pesquisa. As demais, embora já conhecessem o teor do projeto de pesquisa e do sistema, não haviam tido contato mais direto com o conteúdo da arquitetura e do sistema.

6.3 Resultados da Aplicação da Ferramenta no Exemplo

A avaliação da aplicação foi realizada com base nas entrevistas descritas no item anterior e também conforme o parecer do pesquisador sobre a utilidade do sistema. No geral, as pessoas avaliaram a ferramenta como positiva e fácil de ser utilizada. Consideraram a interface (facilidade de recuperação, inclusão e busca) agradável, assim como o sistema de listas. Quanto ao cadastramento e envio dos artigos, no geral a avaliação foi positiva também.

Os principais pontos em consenso com relação ao uso da ferramenta foram:

- os usuários consideraram que, embora o procedimento de cadastro parecesse simples, a maioria julgou que, na rotina normal de trabalho, não interromperiam suas tarefas para alimentar o sistema;
- parte não conseguiu, sem auxílio do pesquisador, entender algumas classificações, pois não havia explicações e nem ajuda. Depois que as recebiam compreendiam bem e passavam a utilizar o sistema de forma mais eficiente, revelando a necessidade de um módulo de Ajuda (“Help”). Todos os entrevistados, durante o período do teste, em que eram deixados sozinhos com a ferramenta, buscaram uma ou mais vezes o pesquisador para eliminar dúvidas sobre o uso da ferramenta;
- tiveram dificuldade de interpretar a tela inicial, em especial a barra lateral devido à grande quantidade de itens. Uma vez explicada sua função e o princípio que norteava sua construção, começavam então a utilizá-la. A grande maioria concordou com a eficiência e vantagem de possuir todos os *links* à mão, acreditando, porém, que a lista de classe e subclasse poderia ser aprimorada e
- alguns tiveram dificuldade de se localizar dentro do sistema após avançar além das telas iniciais, revelando que há um problema de

localização dentro dessa interface, principalmente para as operações de atualização e de exclusão.

Quanto ao procedimento de validação e sistematização, os usuários consideraram interessante e viável de ser utilizada. Porém a maioria concordou que a chance de utilizá-la em uma situação real, dentro de uma rotina de trabalho, seria muito difícil. Eles julgaram que, em condições normais de uso, seriam criadas muitas sentenças, sendo improvável que os especialistas, responsáveis pelo gerenciamento e criação das sentenças, pudessem mantê-las dentro deste contexto de uso intensivo. Essa primeira aplicação, devido à sua característica de exemplificação e curta duração, não permitiu verificar este aspecto, o qual deverá ser tratado de forma mais intensa em avaliações futuras, onde o emprego da ferramenta deverá ser realizado por um período mais longo e numa situação mais crítica e próxima da realidade de um processo de desenvolvimento de produto complexo.

Atualmente, já estão em andamento alguns trabalhos que visam atingir estes objetivos, que são apresentados no item 6.4. Essas avaliações adicionais deverão permitir testar, com maior rigor, tanto a solução desenvolvida como as hipóteses que guiaram o desenvolvimento da arquitetura.

6.4 Avaliações adicionais e em andamento da solução

Essa etapa do trabalho completo, a de desenvolvimento de uma solução específica, propiciou um resultado muito importante e duradouro que terá extrema valia na continuidade das pesquisas para o desenvolvimento e avaliação da arquitetura. Trata-se do protótipo funcional desta solução. Espera-se empregá-lo em outras investigações e, com isso, dar continuidade ao aprimoramento e à validação do conceito da arquitetura e da solução específica. Descreve-se a seguir parte desses trabalhos que já estão em andamento.

Exemplo teórico da aplicação da solução em uma pequena empresa de alta tecnologia

Uma das curiosidades despertadas durante o desenvolvimento dos testes da solução residia em uma limitação específica do caso analisado. Era o aspecto do grupo de pesquisa não se constituir, idealmente, em uma empresa que desenvolve produtos. Como forma de entender um pouco melhor estas diferenças e verificar o uso do sistema dentro deste contexto, realizou-se um exercício conceitual de

“personalizar” a ferramenta, para o caso de uma pequena empresa de alta tecnologia da região de São Carlos, bastante conhecida do pesquisador. Foi gerada uma base de dados específica para essa empresa, a qual foi posteriormente inserida no sistema. A estrutura de classe e subclasse elaborada é apresentada no quadro 6.2.

Quadro 6. 2: Estrutura de classes e subclasses para uma pequena empresa

classe	subclasse	Estrutura de Dados
Reunião	Reunião	Evento
	Workshop	Evento
	Ata de Reunião	Evento
	Curso	Evento
	Modelo de Ata de Reunião	Padrão
	Cronograma	Referencia
Gerenciamento de Projeto	Formulário de Descrição de Times	Referencia
	Relatório de Auto-Avaliação	Referencia
	Relatórios de Gates	Referencia
	Definição do Escopo do Projeto	Referencia
	Avaliação de Risco	Referencia
	Casos sobre Gerenciamento de Projeto	Caso
	Apresentações de Projetos	Referencia
	Modelo de Registros	Padrão
Modelo de Documento	Modelos de Listas de Verificação	Padrão
	Modelos de Procedimentos	Padrão
	Modelo para desenhos CAD	Padrão
	Modelo de Relatório	Padrão
	Relatório de Pesquisa de Mercado	Referencias
Requisito	Instrumentos de Pesquisa de Mercado	Referencias
	Entrevistas para Pesquisa de Mercado	Referencias
	Casas de QFD	Referencias
	Compilações de Dados Secundários	Referencias
	Casos sobre Planejamento e Execução de Pesquisas	Caso
	Casos sobre eficiência dos requisitos	Caso
	Lista de Requisitos Compilados	referencia
	Desenhos	Referencia
Especificação	Lista de Material	Referencia
	Desenho	Referencia
	Plano de Processo	Referencia
	Programa CN	Referencia
	Padrão para aplicação de tolerâncias	Padrao
	Lista de normalização de materiais	Padrao
	Casos de Aplicação de Tolerâncias	Caso
	Casos de Solução de Projeto	Caso
	Relatório de Teste	Referencia
Teste	Padrões para Testes	Padrao
	Casos de Testes	Caso

Quadro 6. 3: Continuação do quadro 6.2

classe	subclasse	Estrutura de Dados
Normas e Padrões	Normas de Qualidade	Padrao
	Normas de Produto	Padrao
	Livro	Referencia
Conhecimento Básico	Periódico	Referencia
	Associação de classe	Especialistas
	Apostila	Referencia
	Site	Referencia
	Apresentação Teórica	Referencia
	Artigos de Periódicos	referencia
Informações Sobre Mercado e Clientes	Livro	referencia
	Site	referencia
	Relatório de Pesquisa	referencia
	Fornecedor	Especialista
Contatos	Cliente	Especialista
	Especialista	Especialista

Este exemplo mostra com maior clareza como a solução, no estágio atual de desenvolvimento, poderia ser aplicada em um caso envolvendo uma empresa que desenvolve produtos. Uma evidência que surgiu deste exercício, é que parece faltar na ferramenta uma entidade (talvez até um módulo) que pudesse organizar os conhecimentos explícitos por projeto. Essa nova “visão” parece ser bastante importante e complementar as duas atuais, notadamente uma, por processo de desenvolvimento de produto e outra por meio das classes e subclasses do repositório. Seria o caso de haver mais um módulo, no qual o usuário pudesse ver somente os conhecimentos explícitos e modelo de referência de um projeto específico.

Aplicação da solução numa comunidade de interesse em desenvolvimento de produto

Na metade do ano 2001, a CAPES (Fundação Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) criou o Programa Nacional de Cooperação Acadêmica (PROCAD), com o objetivo de estimular a formação de recursos de alto nível (em pós-graduação) e a cooperação e intercâmbio científico entre instituições de pós-graduação dentro do país. Surgiu, então, a iniciativa de criar, com o apoio desse programa, uma comunidade de interesse na área de desenvolvimento de produto, envolvendo três importantes grupos de pesquisa que atuam na área de gestão e metodologias de desenvolvimento de produto: Grupo de

Engenharia Integrada (EI) já citado nos itens anteriores; Grupo de Pesquisas para o Aperfeiçoamento da Qualidade e Produtividade (GEPEQ), da Universidade Federal de São Carlos (UFSCar) e Núcleo de Desenvolvimento Integrado de Produtos (NEDIP), lotado na Universidade Federal de Santa Catarina.

O objetivo do projeto é criar um modelo para gestão do conhecimento na área de desenvolvimento de produto gerado pelos três grupos, permitindo: o compartilhamento de conhecimentos e experiências de ensino na área de desenvolvimento de produto, a criação de projetos conjuntos, o intercâmbio de alunos dos programas de pós-graduação dessas instituições e o planejamento das disciplinas, visando uma uniformização e complementaridade entre os cursos de pós-graduação oferecidos por essas três instituições.

A solução para a gestão do conhecimento, apresentada no capítulo anterior, foi “personalizada” e implantada para suportar o compartilhamento de conhecimentos dentro dessa comunidade. Empregando-se o protótipo, foi criado um portal Rede de Conhecimentos em Desenvolvimento de Produto cuja localização na WEB é “<http://www.pdp.org.br>”. O processo de implantação durou vários meses, indo de junho até dezembro de 2001. Por ser um problema bem mais complexo (maior quantidade de usuários e distribuídos nacionalmente), a implantação está sendo feita por etapas. Num primeiro momento, foi lançado o repositório de armazenamento de registros, as funcionalidades de comentários e as de modelos de referência. Está em andamento a inclusão de Grupos de Discussão e, no futuro, deverão ser implantados os módulos de sentença e páginas do conhecimento. O sistema conta, atualmente, com quase uma centena de usuários (98) dos três grupos de pesquisa, 185 registros cadastrados, 54 comentários e 15 modelos de referência. A avaliação do uso da solução está sendo conduzida, desde a implantação inicial do sistema, como parte de um trabalho de mestrado.

7. Conclusões e Perspectivas Futuras

Este trabalho apresentou uma arquitetura que auxilia o desenvolvimento de sistemas para gestão de conhecimentos explícitos, empregando diferentes tipos de Soluções Comerciais e Tecnologia de Informação. A arquitetura foi avaliada por meio da proposição de uma solução, baseada em seus conceitos, e a aplicação dessa solução em uma organização exemplo. Trata-se, portanto, de um escopo de trabalho bastante amplo, cujas conclusões e considerações cobrem uma ampla gama de aspectos.

O primeiro conjunto deles se relaciona com os resultados advindos da proposição da arquitetura. São as questões relativas à viabilidade de criação de um modelo com objetivos iguais ao da arquitetura e às dificuldades de aplicação da tecnologia de informação atual para a criação de sistemas de gerenciamento de conhecimentos explícitos. Outro conjunto de considerações se deve à discussão derivada da proposição da solução, compreendendo uma primeira análise da viabilidade tecnológica e prática de se desenvolverem soluções para a GC, empregando a arquitetura como referência básica. E, por fim, com a exemplificação da ferramenta para um caso, pôde-se tecer, adicionalmente, considerações sobre a viabilidade da solução e, mais importante, comparar a realidade prática de uso da ferramenta com os objetivos e requisitos traçados para o modelo teórico inicial. Nos próximos três itens, são discutidos cada um destes pontos e, ao final, tecem-se as considerações e conclusões gerais, isto é, aquelas relacionadas ao trabalho como um todo. Ao final, último item, apresentam-se ainda as perspectivas para trabalhos futuros.

7.1 Considerações relacionadas à proposição da arquitetura

A mais importante conclusão obtida da proposição da arquitetura diz respeito à própria idéia de elaborá-la. Conforme apresentado no capítulo 3, durante as primeiras tentativas de se obter um sistema que suportasse o desenvolvimento de modelos de referência, integrando-os com conhecimentos explícitos, identificou-se um “descompasso” entre a literatura sobre gestão de conhecimento e as ferramentas comerciais propostas para auxiliar essa atividade. A literatura destacava o fato das ferramentas de TI serem parte do problema, mas não apresentavam com clareza qual o

o papel delas no processo de gestão do conhecimento. Os fornecedores de ferramentas, então, adotavam o rótulo “Gestão de Conhecimentos” para diferentes tipos de tecnologia e soluções, na maioria das vezes sem tornar explícito o que consideravam gestão do conhecimento e como sua solução poderia auxiliar esta atividade. Esta situação dificulta a escolha, comparação e especificação de tais sistemas. Ao se aprofundar na análise deste problema, ficou evidente que se tratava de uma grande barreira para as pessoas envolvidas em projetos de gestão do conhecimento pois impedia a aplicação destas soluções nas empresas, dificultava o planejamento de esforços conjuntos, nos quais a tecnologia de informática seria parte de uma estratégia maior de gestão de conhecimentos.

Esse quadro motivou a criação da Arquitetura, um modelo conceitual capaz de auxiliar no desenho de soluções de gerenciamento de conhecimento, as quais, por sua vez, poderiam ser construídas utilizando-se diferentes sistemas comerciais. A experiência com a proposição da arquitetura reforçou ainda mais, a hipótese de que um modelo conceitual desse tipo é útil e importante. Enquanto avançava-se no desenvolvimento da arquitetura, mais o papel da tecnologia de informática na GC tornava-se claro e crescia a visão crítica do potencial das soluções comerciais no apoio destes sistemas. Percebia-se, também, que as ferramentas comerciais em sua maioria trabalhavam com parte dos conhecimentos explícitos, algumas mais fortemente ligadas ao registro e armazenamento e outras, mais ligadas com a análise e/ou descrições na forma gráfica.

Esta experiência de propor a arquitetura permitiu uma segunda conclusão importante: o modelo da espiral de conhecimento proposto por NONAKA & TAKEUCHI (1995) é uma maneira viável e interessante para se conceber sistemas para gestão do conhecimento. Este é o modelo que, depois de várias tentativas de se desenhar a arquitetura (empregando diferentes abordagens para a GC), pareceu mais promissor, em termos de definição do escopo e missão da dos sistemas de informação na gestão do conhecimento. Trata-se de um modelo que, apesar de partir de categorias bastante abstratas na definição do conhecimento (conhecimento tácito, ciclo de conversão, etc...), permite descrever bem as possíveis contribuições dos sistemas de informação no esforço de GC, mais especificamente, a partir da delimitação desses como um dos habilitadores na geração do ciclo de conversão. Sua aplicação na arquitetura reforçou, ainda mais, essa primeira impressão e, assim, uma outra contribuição importante, que emerge deste trabalho, é a recomendação do modelo da

espiral de conhecimentos como um referencial viável para quem deseja implantar ou atuar na área de sistemas de gestão do conhecimento. Além disso, reforça-se a idéia de que é imprescindível que os trabalhos de escolha, avaliação e desenvolvimento de soluções comerciais sobre gestão do conhecimento sejam feitos empregando-se um referencial teórico sobre GC.

Outra conclusão importante é que o fato de não ser possível encontrar uma solução comercial, que pudesse atender totalmente a arquitetura, mostra que as ferramentas atuais destinadas à área de GC não são capazes de suportar esta atividade, se concebida dentro da abordagem mais abrangente de organização que aprende, conforme a empregada no trabalho. Portanto sendo essas as abordagens mais aceitas e empregadas pelas empresas, reforça-se a idéia de que os sistemas atuais de GC precisam ser repensados e aprimorados.

Não deve tardar o surgimento de soluções completas que unam todas as funcionalidades para o gerenciamento de conhecimentos explícitos, mas, para tanto, é preciso que estas se baseiem em um referencial teórico bastante sólido, como é o caso da arquitetura aqui desenvolvida. Nesse caso, uma outra contribuição, obtida da proposição da arquitetura, é a demonstração de um caminho para o desenvolvimento destas soluções, isto é, da arquitetura como uma proposta conceitual para a proposição de uma ferramenta para a gestão do conhecimento.

Estes resultados permitem, portanto, reforçar a hipótese H1, mostrando que não só é possível, mas também é útil e importante a criação de um modelo que suporte o desenvolvimento de sistemas, para a gestão do conhecimento.

Devem-se notar, ainda, os resultados advindos da criação dos elementos da arquitetura, os quais, por si só, são também contribuições da pesquisa, em especial: a Tipologia para Classificação de Modelos de Referência, o Framework de Modelagem, e o Modelo para Validação do conhecimento.

O esforço para a criação da tipologia, que não é detalhado no corpo deste trabalho, mas na referência citada durante o texto, demonstrou a importância de uma classificação desse tipo para as pesquisas em desenvolvimento de produto, pois permitiu a identificação de uma lista de características críticas a ser levadas em consideração na análise deste processo, e permitiu o desenvolvimento de uma tipologia bastante estruturada, que vem sendo aplicada e testada, em vários estudos do grupo de pesquisa.

A experiência no desenvolvimento do *framework* de modelagem mostrou que, apesar da grande quantidade de propostas para modelagem de processos, ainda há espaço para o desenvolvimento de propostas mais pragmáticas, que permitam a construção de modelos de empresa mais simples e versáteis. Isso é importante principalmente no caso do processo de desenvolvimento de produto, na qual a dinâmica de mudança do processo e a necessidade de se conviver com vários modelos dificultam o emprego dos *frameworks* existentes. O *framework* desenvolvido para a pesquisa foi uma tentativa de simplificação e atualização do *framework* ARIS e de sua adaptação às necessidades de modelagem em desenvolvimento de produto, que trouxe algumas contribuições: a inclusão de uma visão conhecimento, a simplificação na quantidade de construtos e o relacionamento entre eles, a definição de classificações específicas para a modelagem do desenvolvimento de produto e os objetos padrão de conhecimentos para o desenvolvimento de produto. A experiência no desenvolvimento e uso deste *framework* mostrou que ele pode ser bastante aprimorado, principalmente quanto aos formalismos empregados e quanto à personalização da ferramenta de modelagem.

Por fim, uma contribuição da proposição da arquitetura foi o modelo para validação e sistematização de conhecimentos explícitos. Apesar de não se tratar de uma inteira novidade em termos de elementos construtivos, pois faz uso de regras e associações, inova, ao propor uma abordagem conjunta em que se aplicam, concomitantemente, as regras e conhecimentos explícitos desestruturados. Por assumir também um sistema de pontuação, esta sistemática permite um dinamismo e uma contínua reavaliação das regras. O resultado final é uma representação estruturada de um conhecimento, que inclui um indicador de validade e a possibilidade de averiguação da informação pelo próprio usuário. Estas características, unidas, a tornam bastante promissora.

7.2 Considerações relacionadas à proposição de uma solução

A proposição de uma solução foi um trabalho árduo, que permitiu muitas considerações sobre a viabilidade da arquitetura e dos pontos críticos para sua aplicação. Em termos gerais, essa experiência demonstrou que a arquitetura pode ser considerada tecnologicamente viável, isto é, não foram identificadas especificações impossíveis de serem implementadas computacionalmente. A arquitetura pode também ser considerada viável em termos práticos, pois o esforço para a criação d

protótipo empregou, principalmente, tecnologias básicas de programação, apresentando uma magnitude e complexidades em níveis compatíveis com um sistema de informação pequeno e tradicional. As técnicas de programação utilizadas não são distintas dos recursos mais simples e usuais empregados em sistemas de informação para WEB. Logicamente, algumas funcionalidades poderiam ter sido implementadas mais eficientemente (isto é, com maior confiabilidade e melhor “acabamento” do sistema). Mas isso se deu, antes, por uma questão de tempo do que por barreiras técnicas de programação e, portanto, esse fato não coloca em risco a viabilidade tecnológica.

Adicionalmente, em termos mais qualitativos, a experiência de desenvolvimento deixou claro alguns pontos críticos para a criação de ferramentas baseadas na arquitetura são:

- **A implementação da integração entre modelos de referência e os outros conhecimentos explícitos:** durante a proposição da solução, uma das barreiras mais evidentes foi a implementação do relacionamento entre os diferentes tipos de conhecimentos explícitos e os modelos de referência. A criação de modelos de empresa exige funcionalidades sofisticadas, em termos gráficos e controles de dados, difíceis e complexas demais para ser implementadas, empregando somente ferramentas genéricas de desenvolvimento. Por outro lado, os sistemas de modelagem comerciais, embora “eficientes” na descrição gráfica, não são facilmente integrados com sistemas que armazenam informações sobre os conhecimentos explícitos em bases de dados, isto é, outros sistemas que façam o papel do repositório na arquitetura. A estratégia adotada neste trabalho foi o de tentar integrar estes dois “mundos”: repositório (intranet) e ferramenta comercial de modelagem (ARIS). Além de mais simples, esse caminho foi, na época do desenvolvimento da solução, considerado mais generalizável pois pode ser utilizado com qualquer ferramenta de modelagem capaz de gerar modelos em HTML; praticamente todas, uma vez que essa funcionalidade está presente nas principais ferramentas comercializadas atualmente e é considerado como importante para a maioria dos clientes. Entretanto, apesar dessa estratégia ter se mostrado factível, constatou-se que ela é bastante limitada, frente ao tipo de integração unidirecional gerado (modelo para conhecimentos explícitos). Deve-se tomar bastante cuidado na especificação dessa integração, caso a arquitetura venha a ser novamente empregada para a especificação de uma outra solução, incluindo

a consideração da possibilidade do emprego de estratégias diferentes da utilizada neste trabalho;

- **Gerenciamento de palavras-chave:** um aspecto fundamental para o sucesso na implementação de soluções baseadas na arquitetura é a implementação correta das palavras-chave, que são elementos fundamentais na integração entre os diferentes tipos de conhecimento (modelo, registro, sentenças, pessoas, páginas do conhecimento). A utilização de dois níveis de palavras-chave (controladas e não controladas), adotada na solução mostrou-se suficiente, mas, na realidade, sentiu-se falta de uma sistemática de gerenciamento de palavras-chave mais efetiva. Esse aspecto precisa ser melhorado nas futuras soluções derivadas da arquitetura;
- **Interface de apresentação das sentenças:** o modelo de validação e sistematização do conhecimento parecia à primeira vista, bastante simples. Porém o desenvolvimento da solução mostrou dois aspectos importantes: 1) ele não é de simples implementação computacional, como se previa; 2) para que ele seja eficiente é preciso que a interface de cadastro das sentenças seja sofisticada, permita concomitantemente a busca e o relacionamento de conhecimentos explícitos e 3) para que o relacionamento entre as sentenças seja de mais fácil visualização é preciso uma interface mais intuitiva do que listas, isto é, deve-se considerar a possibilidade de fazê-lo por meio de representações na forma de “Mapas Cognitivos”, em que, por exemplo, tais relacionamentos poderiam ser feitos com flechas criadas com o *mouse*. A maior parte do tempo de desenvolvimento da solução foi gasta na programação da interface de registro das sentenças e, mesmo assim, julgou-se que o resultado obtido não foi satisfatório. Embora toda a busca e o relacionamento sejam realizados na mesma página, a interface possui um número grande de elementos visuais (caixas, registros, informações, etc..). Quando existirem muitos conhecimentos e muitas sentenças, será muito complicado utilizar essa interface (conforme a avaliação dos usuários do exemplo de aplicação, vide próximo item). A impressão atual é que, entre todos, este é o ponto mais crítico para a criação de soluções baseadas na arquitetura e
- **Falta de uma metodologia mais robusta para especificação e desenvolvimento da solução:** como apresentado no capítulo 5, a metodologia de especificação empregada foi bastante simples, elaborada pelos próprios pesquisadores e baseada

na experiência. Sentiu-se falta, portanto, de uma especificação mais rigorosa, a qual pudesse, inclusive, fazer a “ponte” entre os requisitos de mais alto nível expressos na documentação da arquitetura, isto é, todos aqueles elementos apresentados no capítulo 4, e os desenho e codificação do sistema (mapas de telas, Modelo Entidade-Relacionamento, entre outros). Durante a especificação ficou claro que a mudança entre estes dois “mundos” não é facilmente realizada.

Desta discussão, pode-se concluir que o resultado da elaboração da solução reforça novamente a hipótese H1. Por ser plenamente possível, com a tecnologia de informação existente, gerar uma solução a partir da arquitetura, tem-se uma validação de H1 em termos da capacidade da arquitetura, de gerar uma solução computacional viável. Além disso, a experiência com a manipulação deste protótipo pelos próprios pesquisadores indicou que a integração entre os 3 tipos de conhecimentos explícitos é bastante promissora e útil, sendo, assim, uma primeira indicação de que a hipótese H2 pode ser válida.

7.3 Considerações relacionadas ao exemplo de aplicação

Um primeiro resultado desta etapa é que o modelo de validação e sistematização dos conhecimentos foi considerado, pelos usuários, como interessante e capaz de ser utilizado. Deve-se ressaltar, porém, que parte deles questionou a viabilidade do sistema quando no uso intenso da ferramenta, dentro de uma rotina de trabalho. Esses usuários julgaram que, caso o sistema estivesse em uso, certamente buscariam e utilizariam os conhecimentos, porém, afirmaram também que haveria grande chance de não alimentarem o sistema. Outro ponto questionado foi a dificuldade para os especialistas, responsáveis pelo gerenciamento e criação dessas sentenças, de mantê-las dentro de um contexto de uso intensivo. Ambos os aspectos necessitariam de uma avaliação após um tempo maior de uso.

A aplicação da solução também permitiu a identificação de vários problemas específicos da implementação: 1) o procedimento de cadastro foi considerado muito complexo; 2) a interface não é suficientemente intuitiva como se desejava; 3) a tela inicial é ainda difícil de ser assimilada para os usuários, que se perdiam dentro da solução; 4) os usuários tiveram dificuldades de se localizar dentro do sistema; 5) dificuldade de entender os nomes das seções e 7) inadequação dos elementos gráficos empregados. A identificação desses problemas foi uma contribuição bastante

importante desta fase da pesquisa, pois, estes pontos deverão ser melhorados em outros casos de aplicação da solução.

Quanto ao aspecto principal, relativo à hipótese H2, deve-se inicialmente considerar a ressalva de que a avaliação foi limitada. Uma avaliação ideal desse aspecto da solução deveria incluir um período extenso de uso da ferramenta pelos usuários, no mínimo de 3 a 6 meses, pois, somente nessas condições, eles teriam a oportunidade de fazer uso do sistema, cadastrando e recuperando conhecimentos explícitos, sob condições normais de trabalho, durante a rotina diária. Só assim, seria possível apurar os resultados práticos da aplicação do sistema. Não havendo este tempo disponível partiu-se então para esta primeira avaliação, de caráter mais restrito, proporcional à limitação de tempo. O sistema foi implementado da maneira mais completa possível. Como resultado, embora a avaliação realizada não permita uma validação robusta acerca da hipótese H2, não foi possível identificar, em nenhum momento, uma posição negativa que contrariasse fortemente tal hipótese. Os usuários entrevistados consideraram interessante e útil a integração entre estes conhecimentos. Pode-se considerar, também, que, na solução desenvolvida, o repositório tomou uma posição central, sendo o ponto mais comentado e discutido da ferramenta durante os testes. Uma possível razão para isso talvez seja a estratégia de integração modelo e conhecimento explícito, conforme comentado no item 7.2.

Embora a avaliação não valide totalmente a hipótese H2, assume-se que resultados foram positivos e indicam que a integração entre modelos de empresa e os demais conhecimentos explícitos são bastante úteis em um sistema de gestão do conhecimento.

7.4 Conclusão Final e Perspectivas Futuras

Considerando o resultado de todas estas etapas em conjunto, pode-se concluir que os resultados da pesquisa não permitem refutar as hipóteses H1 e H2. O trabalho demonstra a necessidade e a importância de um modelo conceitual para o desenvolvimento de sistemas para gerenciamento do conhecimento explícito no processo de desenvolvimento de produto. Demonstra, também, indícios de que a integração dos diferentes conhecimentos explícitos com o modelo do processo é útil e interessante. Como consequência, esses resultados incentivam a continuidade da avaliação da solução e da arquitetura para que essas hipóteses continuem a ser testadas.

Cumpriu-se o objetivo principal do trabalho, traçado no capítulo 3: foram obtidas uma arquitetura e uma primeira validação de sua viabilidade. Na prática, isto significa a proposição de um referencial para a criação de soluções de GC mais robustas, empregando ferramentas comerciais e tecnologias existentes. Como desdobramento, há, um grande potencial para que essa arquitetura possa ser empregada na criação de uma ferramenta comercial específica, que permita a integração dos mais diferentes conhecimentos explícitos existentes, de forma a aprimorar as competências e a aprendizagem organizacional.

Naturalmente, os resultados deste trabalho são as primeiras validações da arquitetura e da solução desenvolvidos. A avaliação do real alcance de ambas será feita futuramente com a continuidade deste trabalho de pesquisa. Nesse sentido, uma contribuição adicional, indireta e bastante significativa do trabalho é a criação do protótipo. Ele pode significar uma sólida base para a continuidade do desenvolvimento da proposta. Os detalhes e problemas identificados durante a aplicação prática aqui relatada poderão ser solucionados e serem realizadas avaliações em casos reais de uso.

Possíveis temas para trabalhos futuros:

- aperfeiçoamento da arquitetura com a inclusão de funcionalidades de ensino à distância;
- avaliação mais profunda da aplicação da solução no Grupo de Pesquisa em Engenharia Integrada;
- realização de casos de aplicação da solução em empresas, especialmente nos setores de desenvolvimento de produto;
- programação de sistemas genéricos de automação de escritório (editores de texto, planilhas, etc.), editores de texto e outros destinados a facilitar a inclusão de conhecimentos explícitos na solução. Por exemplo, criando-se macros em modelos de documentos que incluiriam, automaticamente, o documento exibido na base de dados da intranet;
- criação de uma solução, baseada na arquitetura, empregando-se apenas aplicativos de automação de escritório, tais como as plataformas Microsoft Office ou Linux;

- criação de uma solução baseada na arquitetura empregando-se uma integração entre sistemas Product Data Management (PDM) e sistemas de modelagem de empresas;
- desenvolvimento de uma interface gráfica para facilitar a visualização e relacionamento entre sentenças e registros;
- avaliação e proposição de aprimoramentos no *framework* de modelagem, por exemplo: 1) propor objetos-padrão para o *framework* de modelagem; 2) propor construtos para modelar indicadores de desempenho; 3) repensar a necessidade do construto evento; entre outros.
- proposição de metodologia para benchmarking e avaliação de modelos;
- análise de como empregar a UML (Unified Modeling Language) para especificar a arquitetura, facilitando sua aplicação no desenvolvimento de sistemas e
- avaliação do impacto das possibilidades da tecnologia *peer-to-peer* no desenho da arquitetura.

Um dos possíveis trabalhos futuros é a aplicação da solução como suporte para gestão do conhecimento em comunidades de prática. Conforme apresentado no final do capítulo 6, uma pesquisa com este objetivo já está em andamento.

8. Referências Bibliográficas

- ABECKER, A.; BERNARDI, A.; HINKELMANN, K.; KÜHN, O.; SINTEK, M. Toward a technology for organizational memories. **IEEE Intelligent Systems**, p.40-48, May-June, 1998.
- AGOSTA, L. Behind the business intelligence portal. **Intelligent enterprise**, v.2,n.11,p.31-38, aug, 1999.
- AMARAL, D.C.; ROZENFELD, H. Requisitos para a criação de modelos de referência para o processo de desenvolvimento de produto considerando a participação de fornecedores. **XV Congresso Brasileiro de Engenharia Mecânica**. UNICAMP: Campinas-SP, 1999.
- ANAND, V.; MANZ, C.C.; GLICK, W.H. An organizational memory approach to information management. **Academy of management review**, v.23,n.4,p.796-809, 1998.
- ARIS – Toolset Methods Manual – version 3.0** . IDS PROF. SCHEER, 1995.
- ATLAR. (2002). XpertRule knowledge builder. http://www.attar.com/pages/info_kb.htm (28 jan.).
- BARROSO, A.C.O.; GOMES, E. B.P. **Tentando entender gestão do conhecimento**. Rio de Janeiro: Comissão Nacional de Energia Nuclea, 1999. Mimeo.
- BATTISTI, J. **ASP.NET: uma revolução na construção de sites e aplicações web**. Rio de Janeiro: Axcel books, 2001.
- BLOODGOOD, J.M.; SAILSBURY, W.D. Understanding the influence of organizational change strategies on information technology and knowledge management strategies. **Decision support systems**, v.31. p.55-69, 2001.
- BRANDIMARTE, P. CANTAMESSA, M. Methodologies for designing CIM systems: a critique. **Computers in Industry**, v.25, p.281- 293, 1995.
- BROWN, S.L.; EISENHARDT, K.M. (1995). Product development: past research, present findings, and future directions. **Academy of Management Review**, v. 20, n.2, 343-378, April.
- CANTAMESSA, M.; PAOLUCCI, E. Using organizational analysis and IDEF0 for enterprise modelling in SMEs. **International journal of computer integrated manufacturing**, v.11, n.5, p.416-429,1998.
- CARAYANNIS, E.G. The strategic management of technological learning in project program management:the role of extranets, intranets and intellignt agents in knowledge generation, diffusion, and leveraging. **Technovation**, v.18,n.11, 1998.
- CERQUEIRA, L.A. & OLIVA, A. Introdução à lógica. Rio de Janeiro: Zahar, 1980.

- CHEN, D. VALLESPIR, B. DOUMEINGTS, G. GRAI integrated methodology and its mapping onto generic enterprise reference architecture and methodology. **Computers in Industry**, v.33, p.387-394, 1997.
- CHEN, Y-M.; HSIAO, Y-T. A collaborative data management framework for concurrent product and process development. **International Journal of Computer Integrated Manufacturing**, p. 446-469, v.10, n.6, 1997.
- CIMDATA. (2001). What is cPDM ? <http://www.cimdata.com/index.htm> (12 Dec.)
- CLARK, K.B.. Project scope and project performance: the effect of parts strategy and supplier involvement on product development. **Management Science**, v. 35, n. 10, p.1247-1263, Oct, 1989.
- CLARK, K.B.; FUJIMOTO, T. (1991). **Product development performance: strategy, organization and management in the world auto industry**. Boston-Mass.: Harvard Business School Press.
- CLAUSING, D. Total quality development: a step-by-step guide to world-class concurrent engineering. New York, Asme Press, 1993.
- CONVERSE, T. & PARK, J. **PHP4: a bíblia**. Rio de Janeiro: Campus, 2001.
- COOPER, R.G.; KLEINSCHMIDT, E.J. Winning business in product development: the critical success factors. **Research-technology management**, v.39, n.4, p.18-29, July, 1996.
- COSTA, N. Fundamentos da lógica. São Paulo: HUCITEC e EDUSP, 1980.
- CUSUMANO, M.; NOBEOKA, K. **Thinking beyond lean**. New York, Mc Graw-Hill, 1995.
- DATE, C. **Uma introdução ao sistema de banco de dados**. São Paulo: Edgard Blucher, 1999.
- DAVENPORT, T. H. **Reengenharia de Processos**. Rio de Janeiro: Campus, 1994.
- DAVENPORT, T.; PRUSAK, L. **Conhecimento empresarial**. Rio de Janeiro, Campus, 1998.
- DECHAMPS, J.P.; NAYA, P.R. **Produtos irresistíveis**. São Paulo: Markron Books, 1997.
- DIENG, R.; CORBY, O.; GIBION, A.; RIBIÈRE, M. Methods and tools for corporate knowledge management. **International journal of human-computer studies**, V.51, p.567-598, 1999.
- DORNELAS, J.C.A. **Proposta de um Método de Gerenciamento do Desenvolvimento de Software Aplicado no Contexto de uma Metodologia de Integração de Empresas**. São Carlos: Universidade de São Paulo, 1998 (Dissertação de Mestrado).
- DOUMEINGTS, G. VALLESPIR, B. CHEN, D. Methodologies for designing CIM systems: A survey. **Computers in Industry**, v.25, p.263-280, 1995.
- DUTTON, J. IDC-Newsletter. (2001). Will knowledge management succumb to peer pressure ? <http://www.idc.com/itforecaster/itf20010116.stm> (12 Sep.).
- EDWARDS, J.M.; AGUIAR, M.W.C.; COUTTS, I.A. A top-down and botton-up approach to manufacturing enterprise engineering using the function view.

- International journal of computer integrated manufacturing. **International journal of computer integrated manufacturing**, v. 11, n.4, p.236-376, 1998.
- FUNK, J.L. Concurrent engineering and the underlying structure of design problem. **IEEE transaction on engineering management**, v.44, n.3, p. 305-315, August, 1997.
- GARVIN, D. Building a learning organization. **Harvard Business Review**, v. , n. , July-August, 1993.
- GARVIN, D. **Gerenciando a qualidade**. Rio de Janeiro: Qualitmark, 1992.
- GARVIN, D.A. The processes of organization and management. **Sloan management review**, v.39, n.4, summer, p.33-50, 1998.
- GRIFFIN, A. PDMA research on new product development practices: updating trends and benchmarking best practices. **Journal of product innovation management**, v.14, n.16, november, 1997.
- GROOVE. (2002). Groove networks. <http://www.groove.net/> (28 jan.).
- IMAI, K.; NONAKA, I.; TAKEUSHI, H. Managing the new product development process: how Japanese companies learn and unlearn. In: Clark, K.B.; Hayes, R.; Lorenz, C.. **The uneasy alliance**. Boston-Mass, Harvard Business School Press, p.533-561, 1985.
- INTRANETS.COM. (2001). Intranets.com interactive tour. <http://www.intranets.com/Demo/> (28 Dec.).
- JANDL JUNIOR, P.J. Introdução ao JAVA. São Paulo: Berkley Brasil, 2002.
- JARKE, M.; JEUSFELD, M.A.; PETERS,P.;POHL, K. Coordinating distributed organizational knowledge. **Data & knowledge engineering**, v. 23, p.247-268, 1997.
- KAHN, K.; McDONOUGH, E.F. Na empirical study of the relationship among co-location, integration, performance and satisfaction. **Journal of product innovation management**, v.14, p.161-178, 1997.
- KELLER, G. & TEUFEL, T. (1998). **SAP R/3 Process Oriented Implementation**. Harlow, Addison-Wesley.
- KOSANKE, K. CIMOSA – Overview and status. **Computers in Industry**, v.27, p.101-109, 1995.
- LIM, S.H. JUSTER, N. PENNINGTON, A . Enterprise modelling and integration: a taxonomy of seven key aspects. **Computers in Industry**, v.34, p.339-359, 1997.
- LUNDQVIST, M., SUNDGREN, N., TRYGG, L. Remodularization of a product line: adding complexity to project management. **Journal of product innovation management**, v.13, n.4, p.311-324, July, 1996.
- MALHOTRA, Y. Expert systems for knowledge management. **Expert systems with applications**, v.20, p.7-16, 2001.
- MARCA, D.; MCGOWAN, C.L. **IDEFO/SADT: business process and enterprise modelling**. San Diego – EUA: Eclitic Solutions Corporation, 392p.

- MATTA, N.; CORBY, O.; PRASAD, B. A generic library of knowledge components to manage conflicts in CE tasks. **Concurrent engineering: research and applications**, v. 6, n.4, p.274-287, dec, 1998.
- MAURER, H. Web-based knowledge management. **Computer**, p.122-123, March, 1998.
- MENDES, J.V. **Desenvolvimento de um Método para Modelagem de Operação no Contexto de uma Metodologia de Integração de Empresas**. São Carlos: Universidade de São Paulo, 1996 (Dissertação de Mestrado).
- MERTINS, K. JOCHEM, R. JÄKEL, F.W. A tool for object-oriented modelling and analysis of business processes. **Computers in Industry**, v.33, p.345-356, 1997.
- MEYER, M.H.; TERTZAKIAN, P.; UTTERBACK, J.M. Metrics for managing research and development in the context of the product family. **Management science**, v.43, n.1, p.88-111, January, 1997.
- MICROSOFT. **Knowledge management: produtividade organizacional**. Microsoft Corporaion: 1999.
- MICROSOFT. **The power of intranets: creating workgroup web sites with microsoft office 2000 and Front Page 2000**. <http://www.microsoft.com/office/PwrIntra.htm>. (18 Jul. 2000).
- MOORMAN, C; MINER, A.S. Organization improvisation and organization memory. **Academy of management of review**, v.23, n.4, p.698-723, 1998.
- MULLINS, J.W.; SUTHERLAND, D.J. New product development in rapidly changing markets: an exploratory study. **Journal of product innovation management**, v.15, n.3, p.224-236, May, 1998.
- NISHIGUSHI, T. **Strategic industrial sourcing: the Japanese advantage**. New York, Oxford University Press, 1994.
- NONAKA, I.; TAKEUCHI, H. **Criação de conhecimento na empresa**. Rio de Janeiro, Campus, 1997.
- NOVAK, J.P. Learning, creating, and using knowledge: concept maps as facilitative tools in schools and corporations. Mahwah-NewJersey: Lawrence erlbaum associates, 1998. 251p.
- O'DELL, C.; GRAYSON, C.J. If only we knew what we know: identification and transfer of internal best practices. **California Management Review**, p.154-174, v.40, n.3, Spring, 1998.
- O'LEARY, D.E. Using AI in knowledge management: knowledge bases and ontologies. **IEEE Intelligent Systems**, p.34-39, May-June, 1998.
- ORACLE. (2002). Business intelligence applications. http://www.oracle.com/applications/bus_intelligence/index.html?erpi.html (28 jan.).
- PIMENTEL, M.G.C. & TEIXEIRA, C.A.C. Hiperdocumentos XML na web. In CARVALHO, A.C.P.L.F & MALDONADO, J.C. **VI escola regional de informática**. São Carlos-SP: ICMSC-USP, 2001.
- PRESSMAN, R.S. **Engenharia de Software**. São Paulo: Makron Books, 1995.

- PRIMAVERA. (2001). Primavera: products overview. <http://www.collaboration-tools.com/tools.htm> (28 Dec.).
- PROJECT MANAGEMENT CENTER (2001). Directory of project management software. <http://www.infogoal.com/pmc/pmcswr.htm> (28 Dec.).
- PROSCI. (2001). Collaborative tools and software. <http://www.collaboration-tools.com/tools.htm> (28 Dec.).
- PUGH, S. **Creating innovative products using total design**: the living legacy of Stuart Pugh. Massachusetts: Addison-Wesley, 1996, 533p.
- PUGH, S. **Total Design**. 1978.
- RAMESH, B.; TIWANA, A. Supporting collaborative process knowledge management in new product development teams. **Decision support systems**, v.27, p.213-235, 1999.
- REIMER, U. Knowledge integration for building organization memorues. Freiburg-German: **Knowledge-based systems for knowledge management in enterprises**, September, 1997. Proceedings.
- REITHOFER, W. NAEGER, G. Bottom-up planning approaches in enterprise modelling – the need and the state of the art. **Computers in Industry**, v.33, p.223-235, 1997.
- RENTES, A.F. **Proposta de um Metodologia de Integração com Utilização de Conceitos de Modelagem de Empresas**. São Carlos: Universidade de São Paulo, 1995 (Tese de Doutorado).
- ROBERTSON, D; ULRICH, K.; Planning for product plataforms. **Sloan management review**, v. 39, n.4, p.19-31, Summer, 1998.
- RODGERS, P.A; CALDWELL, N.H.M.;HUXOR, A.P.; CLARKSON, P.J. WEBCADET: a knowledge management support system for new product development. Cambridge-UK: **6th IPDMC**, 1999. Proceedings.
- ROZENFELD, H. Desenvolvimento de produtos na Manufatura Integrada por Computador (CIM). In: AMATO, J. (organiz.) **Manufatura classe mundial: conceitos, estratégias e aplicações**. São Paulo: Atlas, 2001. 230p.
- ROZENFELD, H.; AMARAL, D.C.; TOLEDO, J.C.; CARVALHO, J. O processo de desenvolvimento de produtos e processos na fábrica do futuro. In. ROZENFELD, H.R. (org.) **A Fábrica do Futuro**. São Paulo: Banas, 2000.
- ROZENFELD, H; AMARAL, D.C. Proposta de uma tipologia para processos de desenvolvimento de produto visando a construção de modelos de referência. **I Congresso Brasileiro de Gestão do desenvolvimento de produto**.
- RUBENSTEIN-MONTANO, B.; LIEWBOWITZ, J.; BUCHWALTER, J., McCAW; NEWMAN, B.; REBECK, K. A systems thinking framework for knowledge management. **Decision support systems**, v.31, p.5-16, 2001.
- SAP. (2002). My SAP: business intellingence - Key capabilities - KM. <http://www.sap.com/solutions/bi/keycapabilities/km.asp> (28 jan.).
- SAVOLAINEN, T. BEECKMANN, D. GROUMPOS, P. JAGDEV, H. Positioning of modelling approaches, methods and tools. **Computers in Industry**, v.25, p.255-262, 1995.

- SCHEER, A.W. **Architecture of integrated information systems**. Berlin: Springer, 1992.
- SCHEER, A.W. **ARIS: business process frameworks**. Berlin: Springer, 1998a.
- SCHEER, A.W. **ARIS: business process modeling**. Berlin: Springer, 1998b.
- SCOTT, J.E. Organizational knowledge and the Intranet. **Decision Support Systems**, 23, p.3-17, May 1998
- SELTZER, L. ZDNET. (2002). Peer-to-peer: my favorite stupid fad to 2001. <http://techupdate.zdnet.com/techupdate/stories/main/0,14179,2837666,00.html> (16 jan.).
- SHIN, M.; HOLDEN, T.; SCHIMIDT, R.A. From knowledge theory to management practice: towards an integrated approach. **Information processing and management**, v.37, p.335-355, 2001.
- SMITH, R.P. The historical roots of concurrent engineering fundamentals. **IEEE transactions on engineering fundamentals**, v. 44, n.1, p.67-78, February,1997.
- SONG, M.X.; MOTOYA-WEISS, M.M. Critical activities for really new versus incremental products. **Journal of product innovation management**, v.15, n.2, p.124-135, March, 1998.
- SOUDER, W.E.; SONG, X.M. Analyses of U.S. and Japanese management processes associated with new product success and failure in high and low familiarity markets. **Journal of product innovation management**, v.15, n.3, May, 1998.
- STUDER, R.; BENJAMINS, R., FENSEL, D. Knowledge engineering: principles and methods. **Data & knowledge engineering**, v.25, p.161-197, 1998.
- SU, HWANG & LIU. Knowledge architecture and framework design for preventing human error in maintenance design. **Expert systems with applications**, v. 19, p. 219-228, 2000.
- SWINK, M.L.; SANDVIG, J.C.; MABERT, V.A. Customizing concurrent engineering processes: five case studies. **Journal of product innovation management**, v.13, n.13, p.230-244, May, 1996.
- TECHNOLOGY REVIEW. (2001). Reworking online work: Q&A with Ozzie. http://www.techreview.com/web/print_version/mcdonald081301.html. (14 jan.).
- TERRA, J.C.C. **Gestão do conhecimento: o grande desafio empresarial**. São Paulo: Negócio Editora, 2000.
- THANNHUBER, M.; TSENG, M.M.; BULLINGER, H.J., 2001, An autopoietic approach for building knowledge management systems in manufacturing enterprises., **Annals of the CIRP**, 50/1: 313-318.
- TIWANA, A.; RAMESH, B. A design knowledge management system to support collaborative information product evolution. **Decision support systems**, v.31, p.241-262, 2001.
- TOLEDO, J.C. **Gestão da mudança da qualidade do produto**. São Paulo: Escola Politécnica-USP, 1994a. 231p. Tese (Doutorado em Engenharia de Produção).
- USCHOLD, M.; KING, M.; MORALEE, S.; e ZORGIOS, Y. The enterprise ontology. **The knowledge engineering review**, v.13, n.1, p.31-89, 1998.

- VERNADAT, F.B. **Enterprise Modelling and Integration: Principles and Applications**. London: Chapman & Hall, 1996a.
- VERNADAT, F.B. Enterprise integration: on business process and enterprisa activity modelling. **Concurrent engineering: research and applications**, V.4, n.3, p.219-228, Sept, 1996b.
- VERYZER, R.W. Discontinuous innovation and the new product development process. **Journal of product innovation management**, v.15, n.4, p.304-321, July, 1998a.
- VERYZER, R.W. Key factors affecting customer evaluation of discontinuous new products. **Journal of product innovation management**, v.15, n.2, p.136-150, March, 1998b.
- WHEELRIGHT, S.C.; CLARK, K.B. **Revolutionizing product development: quantum leaps in speed, efficiency, and quality**. New York : The Free Press, 1992. 364p.
- WIELINGA, B.; SANDBERG, J.; SCHREIBER, G. Methods and techniques for knowledge management: what has knowledge engineering to offer? *Expert Systems with Applications*, p. 73-84, July, 1997.
- WIIG, K.M.; HOOG, R.; SPEK, R. Supporting knowledge managment: a selection of methods and techniques. *Expert systems with applications*, p.15-27, July, 1997.
- XEROX.(2001). Software: compartilhamento de conhecimento. (http://www.xerox.com/go/xrx/template/011.jsp?Xcntry=BRA&Xlang=pt_BR&Xseg=corp&family=products&view=Product%20Matrix&path=/taxonomy/BRA/Products/Scanners%20and%20Software/Software/Knowledge%20Sharing (12 Dec.).
- ZELM, M. VERNADAT, F.B.; KOSANKE, K. The CIMOSA business modelling process. *Computers in Industry*, v.27, p.123-142, 1995.

