

UNIVERSIDADE ESTADUAL DE MARINGÁ
CENTRO DE TECNOLOGIA
DEPARTAMENTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

CLARA APARECIDA MILANÊZ

Uma arquitetura de *data warehouse* para apoio à gestão de projetos em desenvolvimento distribuído de software

Maringá
2010

CLARA APARECIDA MILANÊZ

Uma arquitetura de *data warehouse* para apoio à gestão de projetos em desenvolvimento distribuído de software

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Departamento de Informática, Centro de Tecnologia da Universidade Estadual de Maringá, como requisito parcial para obtenção do título de Mestre em Ciência da Computação

Orientador: Profa. Dra. Tania Fatima Calvi Tait

Maringá
2010

"Dados Internacionais de Catalogação-na-Publicação (CIP)"
(Biblioteca Setorial - UEM. Nupélia, Maringá, PR, Brasil)

M677u Milanêz, Clara Aparecida, 1981-
Uma arquitetura de *data warehouse* para apoio à gestão de projetos em desenvolvimento distribuído de software / Clara Aparecida Milanêz. -- Maringá, 2010. 115 f. : il. color.

Dissertação (mestrado em Ciência da Computação)--Universidade Estadual de Maringá, Dep. de Informática, 2010.
Orientador: Prof.ª Dr.ª Tania Fátima Calvi Tait.

1. Banco de dados - Gerência. 2. Engenharia de software. 3. Sistemas de suporte de decisão. I. Universidade Estadual de Maringá. Departamento de Informática. Programa de Pós-Graduação em Ciência da Computação.

CDD 22. ed. -005.74
NBR/CIP - 12899 AACR/2

FOLHA DE APROVAÇÃO

CLARA APARECIDA MILANÊZ

Uma arquitetura de *data warehouse* para apoio à gestão de projetos em desenvolvimento distribuído de software

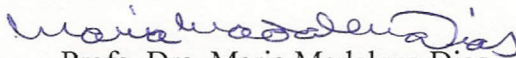
Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Departamento de Informática, Centro de Tecnologia da Universidade Estadual de Maringá, como requisito parcial para obtenção do título de Mestre em Ciência da Computação pela Banca Examinadora composta pelos membros:

BANCA EXAMINADORA



Profa. Dra. Tania Fatima Calvi Tait

Universidade Estadual de Maringá – DIN/UEM



Profa. Dra. Maria Madalena Dias

Universidade Estadual de Maringá – DIN/UEM



Prof. Dr. Marcelo Morandini

Universidade de São Paulo – EACH/USP

Aprovada em: 30 de agosto de 2010.

Local da defesa: Sala 101, Bloco C56, *campus* da Universidade Estadual de Maringá

AGRADECIMENTOS

Primeiramente, a Deus, por tudo.

Aos meus pais, José e Emília, pela paciência, dedicação e pelo maravilhoso exemplo que são na minha vida.

À minha irmã querida, Paula, pela proteção, cuidado e pelas longas e deliciosas conversas ao redor da mesa da mamãe.

Ao meu amado companheiro, Junior, por ter segurado minha onda nos momentos difíceis, pelas canequinhas de leite quente e potinhos de guloseimas que surgiam na minha escrivaninha...

À minha orientadora, Tania, pelo auxílio, pelos puxões de orelha, pelo apoio e pela compreensão nas minhas mudanças de emprego e cidade.

Aos meus colegas de turma, Ju, Camila, Wellington, Shido, Gabriel, Aldo, Borth, Biasão, Ruffles, Montanha, Carlos e Cassolato, pela companhia nas aulas, pela ajuda nos intermináveis exercícios, pelas longas horas no laboratório, pelos cafezinhos na cantina...

À Inês, por ser essa pessoa maravilhosa, com quem sempre pudemos contar.

À professora Elisa, por me permitir realizar o estágio de docência na sua turma.

Ao pessoal do DiSEN, pela ajuda com o ambiente.

Aos meus professores, João Angelo, Elisa, Dante, Itana, Constantino e Ronaldo, pelos ensinamentos.

Aos colegas da Copel, pelas escalas malucas que me permitiram cursar os créditos e pela oportunidade de trabalhar na área.

Aos colegas da Celepar, pelo compartilhamento incondicional de conhecimento, que contribuíram enormemente para meu aprendizado profissional e, conseqüentemente, para esta pesquisa.

Uma arquitetura de *data warehouse* para apoio à gestão de projetos em desenvolvimento distribuído de software

RESUMO

O gerenciamento de projetos envolve a tomada de decisão que deve ser fundamentada em dados sólidos, completos e consistentes. A complexidade da gestão pode aumentar quando esta envolve o projeto de um software desenvolvido de forma distribuída e quando os dados, que devem ser analisados pelo gerente de projetos, são muitos e dispersos geograficamente. Uma solução é reunir e armazenar os dados dos projetos em um *data warehouse* (DW) – que cresce com o passar do tempo e armazena o histórico; para então extrair informações que auxiliem o planejamento e acompanhamento de projetos com desenvolvimento distribuído. Para construir um DW é necessário definir uma arquitetura com base nas necessidades dos usuários finais – nesta pesquisa, os usuários do DW são os gerentes de projeto de software. Este trabalho apresenta uma arquitetura de DW para apoiar a gestão de projetos de desenvolvimento distribuído de software (DDS). A avaliação da arquitetura foi realizada por meio da aplicação desta arquitetura em um DW para o DiSEN, um ambiente de desenvolvimento distribuído de software criado pelo Grupo de Estudos em Engenharia de Software Distribuído da Universidade Estadual de Maringá. A arquitetura apresentada é uma arquitetura de DW global em camadas, composta por *data marts* (DMs) incrementais por assunto. Para propor a arquitetura, necessitou-se realizar o levantamento de requisitos com base das necessidades dos gerentes de projetos em DDS e definir as áreas de assunto a serem atendidas pelos DMs.

Palavras-chave: *Data warehouse*. Desenvolvimento distribuído de software. Gestão de projetos de software.

A data warehouse architecture for distributed software development project management

ABSTRACT

Project management involves decision-making, which must be based on solid, complete and consistent data. The complexity of the management may increase when it involves the design of distributed software development (DSD) and when there are many data that will be evaluated by a project manager and geographically dispersed. One solution is to gather and store project data in a data warehouse (DW) – that grows over time and stores the history, then to extract information to help planning and controlling projects with distributed development. To build a DW is necessary to define an architecture based on the end users needs – in this research of the DW users are the software project managers. This paper presents architecture for DW to support management project of DSD and describes the implementation of this architecture for the DW to DiSEN, a distributed software engineering environment created by the Study Group on Distributed Software Engineering, from State University of Maringá (UEM). The proposed architecture is architecture in layers for global DW, composed of incremental, subject-oriented data marts (DMs). To propose architecture, was needed to elicit requirements from the project managers of DSD and to identify the subject areas to be used as a basis for to implement of the DMs.

Keywords: Data warehouse. Distributed software development. Project management software.

LISTA DE FIGURAS

Figura 1 – Etapas do desenvolvimento da pesquisa	15
Figura 2 – Ciclo de geração de informações nas fases de planejamento e acompanhamento .	21
Figura 3 – Carregamento do DW.....	26
Figura 4 – Extração de Informações e Conhecimento	26
Figura 5 – Arquitetura Bus de Kimball.....	33
Figura 6 – Modelo de Dados Conceitual para GP-DDS	41
Figura 7 – Arquitetura de DW para GP-DDS	53
Figura 8 – Arquitetura do DiSEN	68
Figura 9 – Modelo de dados conceitual em alto nível da GP no ambiente DiSEN	71
Figura 10 – DiSEN – Esquema Estrela para o fato Treinamento do Recurso Humano	75
Figura 11 – Análise do fato Treinamento: Tempo de treinamento por Mês por Treinamento	77
Figura 12 – Análise do fato Treinamento: Nível de aproveitamento por Treinamento	78
Figura 13 – Esquema Estrela para o fato Atribuição de atividade	94
Figura 14 – Esquema Estrela para o fato Conhecimento e Treinamento da Equipe em Tecnologia	95
Figura 15 – Esquema Estrela para o fato Conhecimento e Treinamento do Recurso Humano em Tecnologia	96
Figura 16 – Esquema Estrela para o fato Custo da Atividade	97
Figura 17 – Esquema Estrela para o fato Experiência	98
Figura 18 – Esquema Estrela para o fato Manutenção do Requisito	99
Figura 19 – Esquema Estrela para o fato Tempo e Quantidade de Atividade.....	100

LISTA DE QUADROS

Quadro 1 – Facilidades e dificuldades no desenvolvimento distribuído de software.....	19
Quadro 2 – Comparação de tipos de tabela de fatos.....	31
Quadro 3 – Fatos e Medidas para um DW de apoio à GP-DDS	56
Quadro 4 – Matriz de barramento por Fatos para o DW de GP-DDS	60
Quadro 5 – Definição do Tipo e da Granularidade das Tabelas de fatos.....	61
Quadro 6 – Matriz de barramento por Área de Assunto para DW de GP-DDS.....	62
Quadro 7 – Fatos e Dimensões a partir das necessidades dos gerentes de projeto.....	88
Quadro 8 – Dados do DiSEN - Área de assunto Comum (1).....	103
Quadro 9 – Dados do DiSEN - Área de assunto Comum (2).....	104
Quadro 10 – Dados do DiSEN - Área de assunto Comum (3).....	105
Quadro 11 – Dados do DiSEN - Área de assunto Qualidade.....	105
Quadro 12 – Dados do DiSEN - Área de assunto Recurso (1).....	106
Quadro 13 – Dados do DiSEN - Área de assunto Recurso (2).....	107
Quadro 14 – Dados do DiSEN - Área de assunto Risco	108
Quadro 15 – Significado dos Dados do DiSEN - Área de assunto Comum	109
Quadro 16 – Significado dos Dados do DiSEN - Área de assunto Qualidade	110
Quadro 17 – Significado dos Dados do DiSEN - Área de assunto Recurso	110
Quadro 18 – Significado dos Dados do DiSEN - Área de assunto Risco	111
Quadro 19 – Possíveis valores dos Dados do DiSEN - Área de assunto Comum.....	112
Quadro 20 – Possíveis valores dos Dados do DiSEN - Área de assunto Qualidade	113
Quadro 21 – Possíveis valores dos Dados do DiSEN - Área de assunto Recurso	113
Quadro 22 – Possíveis valores dos Dados do DiSEN - Área de assunto Risco	114

LISTA DE ABREVIATURAS E SIGLAS

ASI	Arquitetura de Sistemas de Informação
BI	<i>Business Intelligence</i> (Inteligência de negócios)
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
CNPq	Conselho Nacional de Desenvolvimento Científico e Tecnológico
DDS	Desenvolvimento Distribuído de Software
DER	Diagrama Entidade Relacionamento
DiSEN	<i>Distributed Software Engineering Environment</i> (Ambiente de desenvolvimento distribuído de software)
DM	<i>Data mart</i>
DW	<i>Data warehouse</i>
EDW	<i>Data warehouse</i> Empresarial
ERP	<i>Enterprise Resource Planning</i> (Sistemas integrados de gestão)
ETL	<i>Extract, Transform, Load</i> (Extrair, transformar e carregar)
GEESD	Grupo de Estudos em Engenharia de Software Distribuído
GP	Gestão de Projetos
GP-DDS	Gestão de Projetos com Desenvolvimento Distribuído de Software
ITIL	<i>Information Technology Infrastructure Library</i> (Biblioteca de infraestrutura de tecnologia da informação)
MD	Modelagem Dimensional
MER	Modelo Entidade Relacionamento
MOLAP	<i>Multidimensional Online Analytical Processing</i> (Processamento analítico online multidimensional)
OLAP	<i>Online Analytical Processing</i> (Processamento analítico online)
OLTP	<i>Online Transaction Processing</i> (Processamento de transações online)
P&D	Pesquisa e Desenvolvimento
RDW	<i>Real Data warehouse</i> (<i>Data warehouse</i> em tempo real)
ROLAP	<i>Relational Online Analytical Processing</i> (Processamento analítico online relacional)
SAD	Sistema de Apoio à Decisão
SI	Sistema de Informação
SQL	<i>Structured Query Language</i> (Linguagem de consulta estruturada)
TI	Tecnologia da informação

SUMÁRIO

Capítulo 1 – Introdução.....	11
1.1. Objetivos da Pesquisa	13
1.1.1. Objetivo geral	13
1.1.2. Objetivos específicos	13
1.2. Justificativa.....	13
1.3. Limitações	14
1.4. Metodologia de Desenvolvimento da Pesquisa	14
1.5. Organização do Trabalho	16
Capítulo 2 – Gestão de Projetos de Software e o Desenvolvimento Distribuído	17
2.1. Desenvolvimento Distribuído de Software	18
2.2. Gestão de Projetos de Software com Desenvolvimento Distribuído	19
2.2.1. Desafios da gestão de projetos com desenvolvimento distribuído de software	20
2.3. Considerações finais	22
Capítulo 3 – <i>Data warehouse</i>.....	23
3.1. Definição de <i>Data warehouse</i>	24
3.2. <i>Data marts</i>	27
3.3. Componentes do <i>Data warehouse</i>	27
3.4. Metadados	29
3.5. Modelo de Dados para o <i>Data warehouse</i>	29
3.5.1. Granularidade	31
3.6. Arquitetura de um <i>Data warehouse</i>	32
3.7. Trabalhos Relacionados	34
3.8. Considerações finais	35
Capítulo 4 – Arquitetura de <i>Data warehouse</i> para Gestão de Projetos em DDS	36
4.1. Requisitos de Apoio à Decisão na Gestão de Projetos em Desenvolvimento Distribuído de Software.....	37
4.1.1. Objetivos globais das organizações que desenvolvem software	37
4.1.2. Necessidades dos gerentes de projeto de software	38
4.1.3. Modelo de dados conceitual do negócio	40
4.1.4. Áreas de assunto para gestão de projetos em desenvolvimento distribuído de software	42
4.1.5. Informações para gestão de projetos de software	43

4.2.	Arquitetura de Date Warehouse para Gestão de Projetos em Desenvolvimento Distribuído de Software.....	49
4.2.1.	O processo de escolha da arquitetura.....	49
4.2.2.	A arquitetura proposta.....	52
4.2.3.	Modelo de dados para o <i>data warehouse</i> – um modelo multidimensional de alto nível	55
4.2.4.	Roteiro para projetar um DW para GP-DDS utilizando a arquitetura proposta....	63
4.3.	Considerações finais	65
Capítulo 5 – Construção do DW sobre o DiSEN utilizando a Arquitetura Proposta.....		66
5.1.	O Ambiente DiSEN	67
5.1.1.	Arquitetura do DiSEN.....	67
5.2.	Aplicação da Arquitetura Proposta sobre o DiSEN.....	68
5.2.1.	Modelo de dados conceitual do DiSEN	70
5.2.2.	Definição do escopo para o projeto inicial do <i>data warehouse</i>	72
5.2.3.	Granularidade	72
5.2.4.	Fonte de dados.....	72
5.2.5.	Análise dos dados do ambiente DiSEN	72
5.2.6.	Recomendações para o ambiente DiSEN.....	73
5.2.7.	Modelagem dimensional para o ambiente DiSEN	74
5.2.8.	Aspectos da implementação física para o DW sobre o ambiente DiSEN.....	76
5.3.	Construção do DW sobre o DiSEN	76
5.4.	Considerações finais	79
Capítulo 6 – Conclusões.....		80
6.1.	Dificuldades Encontradas.....	81
6.2.	Contribuições.....	81
6.3.	Trabalhos Futuros	82
Referências Bibliográficas		83
Apêndice A		87
Apêndice B		93
Apêndice C		101

Introdução

O gerenciamento de projeto de software engloba o planejamento e o acompanhamento do projeto. Antes de desenvolver um software é necessário planejar, e é no decorrer do seu desenvolvimento que acontece a fase de acompanhamento do projeto.

A fase de planejamento envolve a definição das estratégias que irão conduzir o processo de desenvolvimento de software como um todo (Prikladnicki et al., 2002). Sem um planejamento, a empresa perde a visão geral dos projetos em desenvolvimento, dos projetos futuros e da prioridade de cada um (Audy, 2001).

Durante a fase de acompanhamento do projeto, busca-se identificar: o progresso do projeto conforme o planejado em termos de aprimoramento do desempenho; a conclusão de determinadas atividades; o uso eficaz dos recursos; a obediência a cronogramas; e a eficácia no trabalho da equipe (Smith, 1997).

A gestão de projetos (GP) de software suscita novos desafios quando envolve o desenvolvimento distribuído de software (DDS), em que equipes dispersas geograficamente trabalham num mesmo projeto de software.

Um gerente de projeto de software pode utilizar um sistema de apoio à decisão (SAD) para facilitar a execução de suas atribuições. Os SADs analisam dados para extrair informações úteis e apoiar a tomada de decisão dos gerentes de nível médio, aumentando o conhecimento sobre o negócio e provendo vantagem competitiva. Existem três níveis de gerência, sendo que cada grupo possui diferentes necessidades de informação, de acordo com

suas responsabilidades: gerentes seniores ou de alto nível tomam decisões estratégicas e necessitam de informações resumidas para mostrar rapidamente o desempenho geral da empresa; gerentes de nível tático ou médio conduzem programas e planos determinados pelos gerentes de alto nível e necessitam de informações específicas sobre os resultados de cada área funcional; os gerentes operacionais monitoram atividades diárias e precisam de informações detalhadas em termos de transação (Laudon, Laudon, 2007).

Considerando o DDS, que envolve variáveis adicionais específicas para esse tipo de desenvolvimento, o volume de dados que serão analisados pelo SAD pode ser grande e os mesmos podem estar distribuídos em diferentes locais.

A solução proposta por este trabalho é reunir e armazenar as informações históricas dos projetos em um *data warehouse* (DW), possibilitando a compreensão desses dados ao longo do tempo e, assim, auxiliando o planejamento e acompanhamento do DDS.

Esta pesquisa inclui a descrição de uma arquitetura para DW de apoio à GP de software desenvolvidos de forma distribuída e também, a sua aplicação em um DW para o ambiente de DDS denominado DiSEN (*Distributed Software Engineering Environment*) como forma de avaliar a arquitetura proposta.

O DiSEN, que está em desenvolvimento pelo Grupo de Estudos em Engenharia de Software Distribuído (GEESD) da Universidade Estadual de Maringá (UEM), é composto de diversas ferramentas e tem como objetivo auxiliar o processo de DDS em aspectos como o compartilhamento de artefatos, o gerenciamento de versões, a realização de reuniões virtuais, o planejamento e acompanhamento de projetos, entre outros.

Esta pesquisa está relacionada ao projeto de pesquisa em andamento denominado “Suporte à Percepção e ao Contexto em Ambientes de Desenvolvimento Distribuído de Software” (Huzita, 2008), coordenado pela professora Dra. Elisa Hatsue Moriya Huzita, que tem como objetivo contribuir com novas pesquisas no ambiente DiSEN, oferecendo suporte aos problemas gerados pela distância temporal, geográfica e sócio-cultural presentes no DDS.

1.1. Objetivos da Pesquisa

1.1.1. Objetivo geral

O objetivo principal desta pesquisa é definir uma arquitetura de DW para apoiar a gestão de projetos em desenvolvimento distribuído de software (GP-DDS). Um DW baseado no ambiente DiSEN foi construído para avaliação dessa arquitetura.

1.1.2. Objetivos específicos

Os objetivos específicos desta pesquisa são:

- Estudar as necessidades dos gerentes de projeto em DDS, levantando as informações relevantes que auxiliam no processo de tomada de decisão e os aspectos que contribuem para a elaboração da arquitetura;
- Definir as áreas de assunto ou processos de negócio relacionados à GP-DDS;
- Estudar as informações disponíveis no ambiente DiSEN que contribuem para a GP;
- Implementar um DW com base no ambiente DiSEN aplicando a arquitetura proposta, com a finalidade de validar a proposta;
- Contribuir para o desenvolvimento de pesquisas para o DiSEN na área de GP;
- Contribuir para o aprimoramento das áreas de DW, GP e DDS.

1.2. Justificativa

A GP compreende a tomada de decisão. A qualidade da decisão está diretamente associada à qualidade dos dados em que ela é baseada. No DDS, ocorre um aumento na complexidade da GP, pois os dados estão em maior quantidade e podem estar espalhados em diferentes locais e formatos. Considerando essas informações, é possível ponderar que existe uma maior dificuldade na tomada de decisão para o planejamento e acompanhamento de projetos de software quando estes são desenvolvidos de forma distribuída.

Uma arquitetura para construção de um DW possibilita o agrupamento dos registros históricos – de planejamentos e acompanhamentos anteriores, integrados em um mesmo local e disponíveis de forma rápida e segura para serem utilizados por diferentes técnicas de

extração de dados, com o objetivo de melhorar o processo decisório do gerente de projetos de software.

A principal justificativa na escolha do DW é a inviabilidade na busca dessas informações de suporte à decisão diretamente dos bancos de dados operacionais, especialmente por que esses bancos de dados não armazenam o histórico das mudanças; os dados podem estar distribuídos em diferentes sistemas e/ou formatos; e com o passar do tempo a quantidade de informações detalhadas armazenadas pode impedir que uma consulta seja realizada em tempo hábil para uma tomada de decisão.

1.3. Limitações

As limitações desta pesquisa são apresentadas a seguir:

- A pesquisa desenvolvida considera a GP de softwares que são desenvolvidos de forma distribuída, no entanto, é possível adaptá-la para a GP de softwares desenvolvidos de forma localizada (não distribuída) após adequações no modelo dimensional de dados;
- Não existem dados armazenados no banco de dados operacional do DiSEN, o DW foi desenvolvido considerando a estrutura atual do banco de dados (suas tabelas, atributos e relacionamentos);
- Como o ambiente DiSEN não está sendo utilizado em produção por uma organização, a definição dos objetivos da organização e definição das necessidades específicas dos gerentes de projeto da organização não foram adaptadas, tendo sido utilizado o modelo genérico da arquitetura de DW proposta. A análise da qualidade dos dados e a definição de aspectos da implementação física, que envolvia análise de volume e de frequência de atualização dos dados, não foram realizadas pelo mesmo motivo.
- A arquitetura de DW proposta enfoca os gerentes de nível tático, não envolvendo informações estratégicas para gerentes de alto nível.

1.4. Metodologia de Desenvolvimento da Pesquisa

Esta pesquisa foi conduzida em três etapas: fundamentação teórica, definição da arquitetura de DW e aplicação da arquitetura proposta como forma de validação; conforme ilustrado na Figura 1.

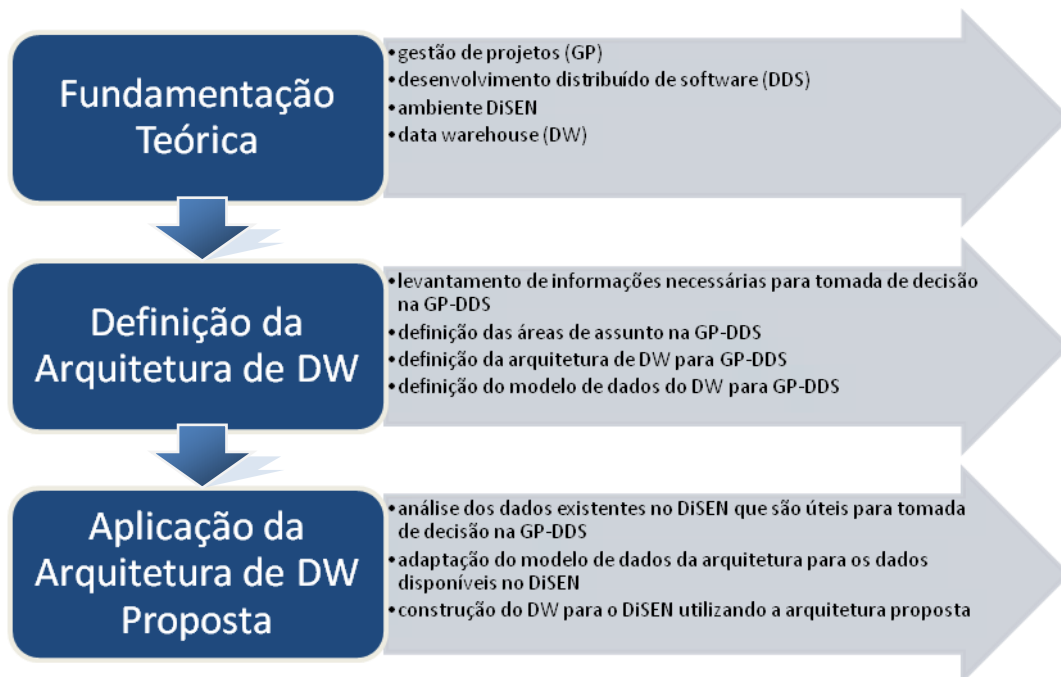


Figura 1 – Etapas do desenvolvimento da pesquisa

Conforme apresentado na Figura 1, a fundamentação teórica, primeira etapa da metodologia de desenvolvimento, englobou os assuntos: GP, DDS, ambiente DiSEN e DW. Foram estudadas as características de DDS, sua relação com GP e os desafios enfrentados pelos gerentes de projetos nesse tipo de desenvolvimento. O ambiente DiSEN foi estudado sob o ponto de vista da GP e no último item foram estudados os conceitos, componentes e arquiteturas existentes para DW.

A definição da proposta de arquitetura de DW para GP-DDS envolveu o levantamento das informações necessárias para a tomada de decisão na GP-DDS, a definição das áreas de assunto existentes, a definição da arquitetura de DW e de um modelo de dados a partir dos requisitos levantados.

Com o propósito de avaliar a construção de um DW para GP-DDS com base na arquitetura foram realizadas as seguintes atividades: análise dos dados relacionados à GP disponíveis no ambiente DiSEN; adaptação do modelo de dados proposto na arquitetura de DW para atender ao ambiente em questão; e a construção do DW com base na arquitetura proposta.

1.5. Organização do Trabalho

O presente documento está estruturado da seguinte maneira: no Capítulo 2 é apresentada uma introdução sobre o DDS e a GP envolvendo essa particularidade de desenvolvimento. No Capítulo 3 são abordados conceitos sobre DW, incluindo uma revisão das arquiteturas existentes e a apresentação de trabalhos relacionados. O Capítulo 4 foi dividido em duas partes, na primeira é realizado o levantamento dos requisitos para apoiar a tomada de decisão dos gerentes de projetos em DDS, incluindo a definição de um modelo de dados conceitual do negócio; na segunda parte é apresentada a arquitetura de DW proposta para GP-DDS. No Capítulo 5 são detalhados o ambiente DiSEN e a construção do DW sobre esse ambiente, utilizando a arquitetura de DW proposta no Capítulo 4. As conclusões são apresentadas no Capítulo 6.

Gestão de Projetos de Software e o Desenvolvimento Distribuído

Este Capítulo é parte da fundamentação teórica relacionada à GP-DDS. Ele é composto de duas partes. A primeira trata dos conceitos de DDS, mostrando as principais características, as facilidades e as dificuldades dessa forma de desenvolvimento de software.

A segunda parte apresenta aspectos da GP sob a perspectiva do DDS, incluindo os desafios dessa configuração de desenvolvimento e demonstrando como uma arquitetura de DW – como a proposta por esta pesquisa, pode auxiliar o gerente de projetos na GP-DDS.

2.1. Desenvolvimento Distribuído de Software

A melhoria dos métodos e ferramentas de Engenharia de Software está permitindo que grupos de pessoas, de diferentes locais e culturas, formem equipes para trabalhar em projetos distribuídos (Prikladnicki, 2003). Superando o tempo e a distância, muitas organizações começam a trabalhar com desenvolvimento de software geograficamente distribuído para capitalizar os recursos globais, aproveitar as estruturas de custo atrativas e o desenvolvimento vinte e quatro horas, com o intuito de acelerar o tempo de desenvolvimento e atender aos mercados locais (Damian, Moitra, 2006). Essa maneira de desenvolver software é conhecida como desenvolvimento distribuído de software (DDS), em contraste com a maneira convencional chamada de desenvolvimento centralizado ou desenvolvimento local.

As principais características de DDS são a colaboração e cooperação entre departamentos de organizações e a criação de grupos de desenvolvedores que trabalham em conjunto, apesar de estarem localizados em cidades ou países diferentes (Prikladnicki, 2003).

Vários fatores incentivam o desenvolvimento distribuído, tais como: necessidade de recursos a qualquer hora, proximidade com o mercado local (conhecimento dos clientes e condições locais), rápida formação de equipes para explorar oportunidades de mercado e desenvolvimento *follow-the-sun* (quase contínuo, pois equipes trabalham em fusos horários diferentes) (Prikladnicki, 2003; Enami, Tait, Huzita, 2006). A distribuição pode ser feita por meio da escolha de regiões onde se concentram especialistas, onde existam incentivos do governo ou onde existam os menores custos para desenvolvimento do software (Bennatan, 2002).

Apesar de apresentar diversas vantagens em relação ao desenvolvimento local, como menor custo e maior produtividade, o DDS traz também alguns problemas. As dificuldades de comunicação podem gerar, por exemplo, diferentes interpretações de uma mesma documentação por equipes dispersas (Bennatan, 2002). Além disso, a distância e as características culturais provocam um aumento na complexidade das tarefas de planejamento e controle sobre a produção do software, por exemplo: diferenças de comportamento e relacionamento dos membros da equipe diante de problemas e conflitos que possam surgir, e atitudes em relação à hierarquia (Lanubile, Damian, Oppenheimer, 2003; Huzita, Tait, 2006).

Alguns problemas introduzidos pelo DDS são as dificuldades de comunicação, controle, supervisão, coordenação, criação de vínculos sociais e construção da confiança. Presumiu-se que algumas interações (comunicação, coordenação e colaboração) pudessem ser

resolvidas com a tecnologia e, portanto, a localização física não seria um problema, o que permitiria que as empresas montassem suas equipes em qualquer parte do planeta onde pudessem encontrar pessoas com as habilidades e produtividade necessárias (Carmel, Abbott, 2007). Porém, como era de se esperar, a tecnologia não resolveu todos os problemas de interação, ainda existem diversos desafios aguardando por pesquisas nesta área, especialmente na área de coordenação, como pode ser visto na Seção 2.2.1 – Desafios da gestão de projetos com desenvolvimento distribuído de software.

O Quadro 1 resume as facilidades e dificuldades encontradas quando o desenvolvimento de software ocorre de maneira distribuída. As facilidades abrangem a possibilidade de utilizar recursos em diferentes locais, a redução de custos, o desenvolvimento *follow-the-sun* e a possibilidade de estar próximo do mercado local. Entre as principais dificuldades estão os problemas de comunicação, a distância física e diferenças horárias que pode dificultar algumas atividades, como reuniões, por exemplo, e a diferença cultural entre os membros da equipe.

Quadro 1 – Facilidades e dificuldades no desenvolvimento distribuído de software

Facilidades	Dificuldades
Recursos globais	Comunicação
Redução de custos	Distância física
Tempo (desenvolvimento 24h)	Fuso horário
Proximidade com mercado local	Cultura

Fonte: Elaborada pela autora com base em Prikladnicki (2003); Damian, Moitra (2006); Bennatan (2002); Carmel, Abbott (2007)

2.2. Gestão de Projetos de Software com Desenvolvimento Distribuído

Gerenciar um projeto significa tomar decisões que envolvem recursos – materiais e humanos, para realizar atividades temporárias com o objetivo de fornecer um resultado. Ao considerar-se a construção de um software como um projeto, o resultado é o próprio software. O gerenciamento de um projeto de software abrange o seu planejamento e controle, englobando também a estimativa de esforço humano utilizado, os custos do projeto, a

medição de software, o gerenciamento de riscos e o gerenciamento do cronograma do projeto (Huzita, Tait, 2006).

O DDS parece ter tornado-se uma necessidade de negócio por muitas razões que incluem: custos, escassez de recursos e a necessidade de localizar desenvolvedores mais próximos dos clientes. Por isso, práticas e estratégias eficazes para obter sucesso na organização e no gerenciamento do desenvolvimento de software global, estão tornando-se críticas (Damian, Moitra, 2006).

A GP-DDS requer técnicas e métodos especiais. Bennatan (2002) define três itens a serem considerados para o sucesso da GP-DDS: a) definição de um único processo de desenvolvimento de software para o projeto; b) uso de ferramentas e tecnologias disponíveis (comunicação, tomada de decisão, cronograma, análise de riscos); c) criação de um plano de gerenciamento para o projeto, estudando as características de cada equipe distribuída e identificando as áreas que requerem soluções específicas (comunicação, cultura, regulamentações do governo, viagens, avaliação dos especialistas, avaliação de recursos, dentre outros).

2.2.1. Desafios da gestão de projetos com desenvolvimento distribuído de software

Alguns desafios na área de GP foram identificados por Prikladnicki e Audy (2006) em sua análise comparativa com empresas que desenvolvem de forma distribuída no Brasil e no exterior, são eles: desafios relacionados ao planejamento do projeto, padronização de relatórios de acompanhamento, padronização de atividades gerenciais, maior controle das atividades que estão sendo realizadas, necessidade de ferramentas integradas de GP, correta distribuição das atividades para os colaboradores, necessidade de caracterizar as equipes distribuídas como uma única equipe dentro do projeto, idiomas, diferenças culturais, necessidade de um processo de aquisição de confiança (necessidade de viagens para aumentar confiança e melhorar processo de engenharia de requisitos e planejamento do projeto), gerenciamento de expectativas entre as equipes distribuídas, falta de compartilhamento de contexto e *awareness* (percepção do que cada colaborador está executando).

Os sete primeiros itens apontados acima pelas empresas de DDS durante a análise de Prikladnicki e Audy (2006) envolvem gerenciamento de dados com vistas a apoiar a gestão do projeto de desenvolvimento de software, sugerindo dessa forma, a necessidade de mais pesquisas para auxiliar o gerente na definição das estratégias que guiarão o projeto.

O aumento da complexidade durante o planejamento e acompanhamento da produção de software deve-se ao acréscimo de variáveis não consideradas no desenvolvimento local, como possibilidade de desenvolvimento vinte e quatro horas, diferenças de fuso horário, recursos distribuídos, dificuldades de comunicação e diferenças culturais.

Conforme apresentado na Figura 2, durante a fase de planejamento de um projeto são geradas informações para serem utilizadas na fase de acompanhamento, na qual, da mesma forma, são geradas informações que podem ser aproveitadas para tomada de decisão durante a fase de planejamento de novos projetos.



Figura 2 – Ciclo de geração de informações nas fases de planejamento e acompanhamento

Fonte: Elaborada pela autora com base em Kerzner (2006)

Segundo Smith (1997), incorporar em um documento de fácil consulta e utilização todas as informações relacionadas à GP – por exemplo, as atividades elementares, as pessoas responsáveis por cada atividade, a sequência, duração e prazo de cada atividade, os recursos necessários, os custos e as etapas eventuais; é uma tarefa difícil. O mesmo autor recomendava que fossem utilizadas ferramentas simples e compreensivas para o planejamento e controle dos projetos.

Porém, os dados relacionados à GP são muitos e inter-relacionados, e uma visão bidimensional cruzando apenas duas áreas de assunto como atividades e pessoas, por exemplo, pode ser insuficiente. Uma perspectiva multidimensional, cruzando diversos assuntos, é mais intuitiva e facilita a exploração e compreensão dos dados corporativos (Singh, 2001).

O DW proposto neste trabalho procura cumprir essa tarefa, agrupando todas as informações em um único local, mantendo os registros históricos e simplificando assim a utilização desses dados para a tomada de decisão dos gerentes de projeto.

2.3. Considerações finais

Este Capítulo apresentou conceitos sobre DDS e sobre GP-DDS. As informações apresentadas incluíram um conjunto de desafios enfrentados pelos gerentes de projetos em DDS, que resultou na motivação desta pesquisa em definir uma arquitetura de DW para GP-DDS.

O próximo Capítulo dá continuidade à fundamentação teórica, descrevendo os conceitos relacionados à DW, os tipos de arquiteturas existentes na literatura e trabalhos relacionados à definição de arquiteturas para contextos específicos – como pretende esta pesquisa.

Data warehouse

Este Capítulo apresenta a tecnologia de DW e seus conceitos. Também são descritos os quatro componentes principais de um DW: as fontes de dados, a área de estagiamento dos dados, a área de apresentação dos dados e a área de acesso aos dados. É ressaltada a importância dos metadados para o ambiente de DW e apresentada a técnica de modelagem de dados mais utilizada para DWs – a modelagem dimensional, incluindo os conceitos de tabelas de fatos, dimensões e granularidade.

Por fim, são apresentados trabalhos que estão relacionados à definição de arquiteturas de DWs para contextos específicos.

3.1. Definição de *Data warehouse*

Inmon (1997, p. 33) define *data warehouse* (DW) como um “conjunto de dados baseado em assuntos, integrado, não-volátil, e variável em relação ao tempo, de apoio às decisões gerenciais”. DW pode ser considerado um exemplo de banco de dados temporal, pois armazena dados históricos (referentes ao futuro e ao passado), ao invés de dados atuais (Date, 2003). Já a tecnologia de *data warehousing* significa o processo de se fazer um DW (Machado, 2006).

Para Singh (2001), DW é uma tecnologia de gestão e análise de dados, ele a descreve como a separação física entre os sistemas de dados operacionais e os SADs de uma organização, que compreende um único repositório de informações construído a partir da integração dos dados corporativos, de forma que os administradores do negócio possam executar consultas, gerar relatórios e fazer análises.

A definição de DW para Turban et al. (2009) é de um repositório de dados atuais e históricos, que possam interessar aos gerentes de toda a organização; em que os dados são estruturados com o objetivo de estarem disponíveis em um formato pronto para as atividades de processamento analítico (como por exemplo: processamento analítico online (OLAP – *Online Analytical Processing*), mineração de dados, consultas, geração de relatórios e outras aplicações de suporte à decisão).

Um DW fornece uma origem de dados única, limpa e consistente permitindo análise complexa, descoberta do conhecimento e apoio à tomada de decisão, de forma a não causar impacto sobre os sistemas operacionais (Elmasri, Navathe, 2005; McGoveran, 2003). Em geral, o DW é armazenado em um ambiente específico, separado do ambiente operacional.

Tecnologias como o DW surgiram com a necessidade de elaboração de melhores sistemas de suporte ao processo de tomada de decisão devido à grande quantidade de informações disponíveis atualmente, ao aumento da concorrência mundial decorrente da globalização e à pressão para que os gerentes conheçam melhor seus clientes e negócio (Rezende et al., 2005).

Os principais objetivos de um DW são: facilitar o acesso às informações da empresa, apresentar informações de modo consistente, adaptar-se às mudanças, proteger e controlar o acesso às informações, subsidiar a tomada de decisão e ser aceito pelos usuários (Kimball, Ross, 2002).

A maior diferença entre bancos de dados tradicionais e os DWs é que os primeiros são transacionais, enquanto que os DWs são direcionados principalmente para aplicações de apoio à decisão (Elmasri, Navathe, 2005). DWs são otimizados para recuperação de dados e não para processamento rotineiro de transações, devem ser considerados como um meio e não como um fim, pois contém informações integradas de forma concisa e confiável, de interesse da organização, para serem utilizadas pelos sistemas de apoio à decisão (SADs) (Tait, Dias, 2004). Um banco de dados tradicional tende a ser não-redundante, normalizado e relacional, enquanto que os DWs são redundantes, não-normalizados e multidimensionais (Turban et al., 2009).

As principais operações que ocorrem em um banco de dados tradicional são: inserir, consultar, atualizar e excluir; enquanto que em um DW ocorrem essencialmente operações de inserção e consulta, dessa forma, é possível ter acesso ao histórico dos dados armazenados e suas transformações no decorrer do tempo.

Os dados armazenados no DW são coletados ao longo do tempo para serem usados em comparações, tendências e previsões. Em DW comuns, os dados não são atualizados em tempo real e sim regularmente, a partir dos sistemas aplicativos (Singh, 2001). Os DWs comuns podem evoluir para um DW em tempo real (RDW), também conhecidos como DW ativos (ADW), que carregam e fornecem dados conforme tornam-se disponíveis (Turban et al., 2009). Este trabalho não trata de RDW, ficando seu estudo e aplicação neste contexto de GP-DDS como uma sugestão para trabalhos futuros.

Os DWs costumam ser grandes e o volume de informações armazenadas cresce muito (em média, 50% ao ano). Suas cargas de trabalho utilizam consultas ocasionais intensivas e o ajuste de desempenho é complicado (McGoveran, 2003). Porém, eles fornecem armazenamento, funcionalidades e capacidade de responder consultas acima das capacidades de bancos de dados orientados por transação.

O DW é construído a partir de diferentes fontes de dados, como mostra a Figura 3.

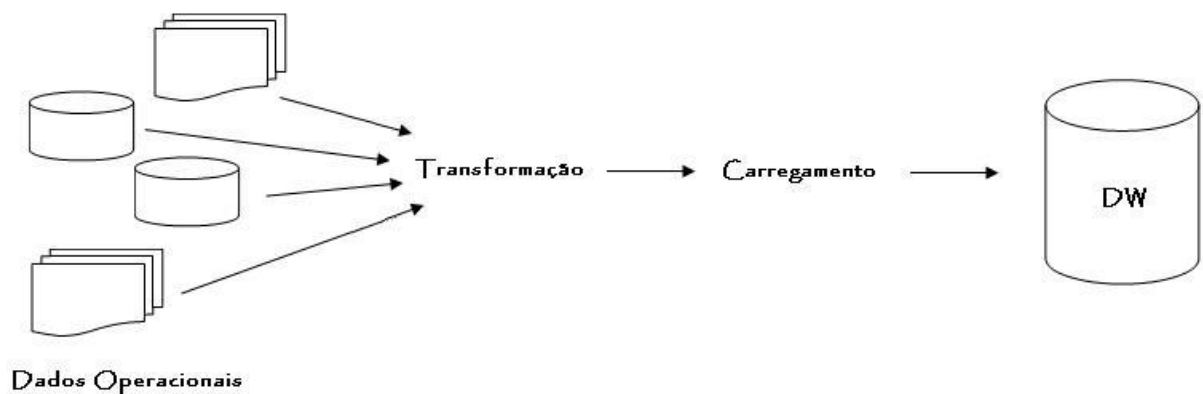


Figura 3 – Carregamento do DW

Fonte: Elaborada pela autora com base em Inmon (1997)

Sobre este DW criado, é possível aplicar diferentes técnicas para obtenção de informações a partir dos dados armazenados, tais como: OLAP para a extração de informações; mecanismos de visualização de dados (gráficos, árvores, entre outros); SQL (*Structured Query Language*) para consultas às informações; ou mineração de dados (*data mining*) para a extração de conhecimento embutido nos dados, conforme ilustra a Figura 4. Esse processo de extração sobre um DW é realizado de forma otimizada, a partir de dados que já estão limpos, agregados e consolidados.

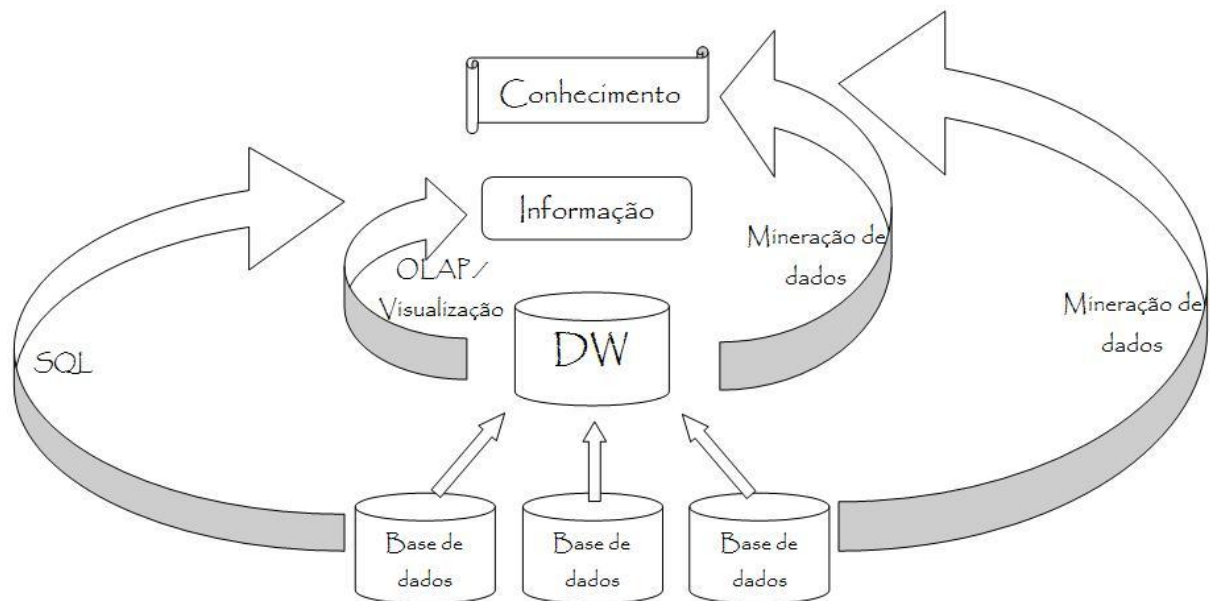


Figura 4 – Extração de Informações e Conhecimento

Fonte: adaptada de Rezende et al. (2005)

3.2. *Data marts*

Um *data mart* (DM) é um subconjunto de um DW (Turban et al., 2009). Eles foram criados com o objetivo de serem limitados e de uso especial, para tornar eficiente as operações de relatórios e análise de dados que com frequência alguns usuários realizavam sobre um mesmo subconjunto de dados de um DW, sendo que, em alguns casos, o subconjunto estava armazenado localmente (McGoveran, 2003).

Singh (2001) apresenta DM como um subconjunto do DW “empresa-inteira”, que pode desempenhar o papel de um DW departamental, regional ou funcional. Os DMs podem ser construídos ao longo do tempo e posteriormente agrupados em um DW.

Para McGoveran (2003, p. 604), DM é um “depósito de dados especializado, orientado por assunto, integrado, volátil e variável no tempo, que fornece apoio a um subconjunto específico de decisões da gestão”.

3.3. Componentes do *Data warehouse*

Kimball e Ross (2002) consideram que o ambiente do DW é formado por quatro componentes separados e distintos. São eles: fontes de dados, área de estagiamento dos dados, área de apresentação dos dados e ferramentas de acesso a dados.

As fontes de dados, também chamadas sistemas operacionais de origem, alimentam o DW. As informações que serão armazenadas no DW podem ser extraídas de diversos sistemas independentes, inclusive de fontes externas – como os dados do Censo ou o índice Dow Jones, por exemplo. Em geral, os dados vêm de sistemas que capturam as transações da empresa (OLTP – processamento de transações online) e que possuem um volume de dados históricos pequeno; mas podem vir também de sistemas integrados de gestão (ERP), dados da Web (*logs*), dados armazenados em *mainframe*, dados em formato de planilhas ou textos, entre outros.

A área de estagiamento dos dados não é acessível aos usuários do DW. É nela que ocorrem os processos de extração, transformação e carga para preparação dos dados operacionais brutos, também conhecidos como ETL (*Extract, Transform, Load*). É possível realizar o processo ETL utilizando softwares comerciais específicos ou por meio de softwares personalizados para o DW.

A primeira etapa do processo de obtenção dos dados é a extração, nessa etapa ocorre a leitura e a compreensão dos dados de origem. Em seguida, ocorrem diversas transformações:

filtragem dos dados, exclusão de dados duplicados, atribuição de chaves e combinação de dados de diferentes origens. Essa integração dos dados é necessária para obter-se consistência de nomes, de variáveis de medidas, de codificação das estruturas e de atributos físicos dos dados (Inmon, Hackathorn, 1997).

Após a transformação, os dados são carregados na área de apresentação. É na área de apresentação que os dados ficam armazenados em esquemas dimensionais para serem acessados pelos usuários do DW e pelas aplicações de análise. Kimball e Ross (2002) definem a área de apresentação de dados como um conjunto de DMs integrados, em que cada DM representa uma área de assunto, contendo dados detalhados e atômicos.

O último componente é formado pelas ferramentas de acesso a dados que consultam dados da área de apresentação de dados. As ferramentas incluem desde simples operações de consultas até complexas operações de exploração de dados.

O acesso aos dados do DW pode ser feito empregando-se consultas simples, como comandos SQL acessando diretamente o banco de dados relacional, onde estão armazenados os dados do DW, e retornando as linhas acessadas em uma forma tabular. Outra forma de consulta é por meio de *stored procedures* – consultas armazenadas no banco de dados, utilizadas quando as consultas são pré-definidas pelo usuário. Um aplicativo de consulta pode ser construído para facilitar a interface entre as consultas pré-definidas e o usuário final, armazenando as consultas e permitindo que elas sejam executadas novamente.

Outras ferramentas permitem que as consultas sejam agendadas para serem executadas periodicamente ou oferecem sistemas de alerta que monitoram os dados do DW para oferecer as informações aos usuários quando um evento crítico ocorrer (Turban et al., 2009).

O acesso aos dados do DW também pode ser feito por meio de ferramentas específicas. Exemplos de ferramentas de acesso a dados são as ferramentas de análise multidimensional que manipulam dados agregados em categorias ou dimensões, permitindo ao usuário sintetizar a informação e obter uma visão corporativa, personalizada e projetada para a análise de dados históricos (Inmon, Welch, Glassey, 1999). A visão corporativa dos dados em diferentes perspectivas facilita a exploração da informação e é mais intuitiva do que a visão tabular. Essas ferramentas, em geral, oferecem a opção de salvar uma consulta, permitindo sua reexecução para obter informações atualizadas do DW.

3.4. Metadados

Metadados são dados sobre dados (Inmon, 1997). São documentadores dos dados que auxiliam a converter dados e informações em conhecimento (Turban et al., 2009). Os metadados estão situados em uma camada diferente do DW porque seus dados não são retirados diretamente do ambiente operacional (Singh, 2001).

Por descreverem a estrutura, organização e significado dos dados, os metadados contribuem para que o seu uso seja eficiente. Eles podem ser classificados em termos de uso como: metadados técnicos e metadados de negócios; ou em termos de padrão como: metadados sintáticos, que descrevem a sintaxe dos dados; metadados estruturais, que descrevem a estrutura dos dados; e metadados semânticos, que descrevem o significado dos dados em um determinado domínio (Turban et al., 2009).

O papel do metadado no ambiente de DW é diferente do seu papel no ambiente operacional. No ambiente operacional ele tem a mesma importância da documentação e muitas vezes é considerado como opcional, enquanto que no ambiente de DW assume um papel de maior destaque, especialmente por gerenciar o mapeamento entre o ambiente operacional e o ambiente do DW (Inmon, 1997). O metadado pode permitir algumas tarefas, tais como: localizar o conteúdo do DW, localizar a fonte de dados que alimenta o DW, verificar quais dados estão disponíveis no DW, mapear a transformação do dado do ambiente operacional para o ambiente do DW, guiar os algoritmos usados para a sumarização, verificar o histórico de extração e transformação, analisar estatísticas de uso dos dados, prever o tamanho das tabelas do DW e mapear as entidades dimensionais para as entidades relacionais (Singh, 2001; Inmon, 1997). Os metadados podem ainda permitir que mudanças feitas no DW sejam refletidas nas aplicações de acesso aos dados (Singh, 2001).

É normal que o conteúdo e a estrutura dos dados armazenados no DW sejam alterados, e, portanto, é necessário garantir que essas alterações sejam registradas no repositório de metadados, de forma a manter um histórico (Inmon, 1997).

3.5. Modelo de Dados para o *Data warehouse*

Os usuários de DW têm necessidades diferentes dos usuários de bancos de dados tradicionais. Enquanto usuários de bancos de dados operacionais constantemente inserem, atualizam e excluem dados individuais no banco, os usuários de DW necessitam de consultas realizadas sobre grandes conjuntos de dados e que retornem não apenas relatórios contendo textos, mas

cálculos e premissas para suporte à análise. As ferramentas de acesso a dados do DW devem permitir que os usuários mudem o nível de detalhe, rotacionem ou adicionem itens aos relatórios (Singh, 2001).

A maneira mais apropriada para realizar a modelagem de dados para um DW, visando atender às necessidades dos usuários, é a modelagem dimensional (MD), conforme aconselha Kimball et al. (1998). A MD é baseada em recuperação de informações e suporta acessos com grande volume de consultas (Turban et al., 2009). A idéia da MD é de que os dados podem ser representados por cubos, em que as células representam valores medidos e os lados representam as dimensões (Kimball et al., 1998). No mundo real, os cubos podem possuir mais de três dimensões, porém o conceito permanece, é possível fatiar um cubo, rotacioná-lo e visualizar os dados com maior ou menor nível de detalhe.

Na MD existem dois tipos de tabela: a tabela de fatos (a principal) e a tabela de dimensão. Uma tabela de fatos contém um grande volume de linhas que correspondem aos negócios ou fatos observados. A tabela de fatos contém atributos chaves que se referem às tabelas de dimensão e atributos com valores, geralmente numéricos ou aditivos – também chamados de medição, medidas ou métricas – que são utilizados para análise de decisão. As tabelas de dimensão contêm descritores textuais que permitem descrever, classificar e agregar os dados contidos nas tabelas de fato. Em geral, cada tabela de dimensão possui um grande número de atributos. Um exemplo de tabela de dimensão sempre presente em um DW é a tabela de dimensão Tempo, que pode possuir os seguintes atributos: data, hora, dia da semana, mês e ano.

Há duas formas de reunir os fatos e as dimensões: o modelo estrela e o modelo floco de neve. O modelo estrela é o mais simples: uma tabela de fatos central associada a diversas tabelas de dimensão. Já no modelo floco de neve, ocorre a normalização de algumas tabelas de dimensão, criando-se tabelas de dimensão auxiliares. Kimball e Ross (2002) sugerem utilizar o modo floco de neve com critério, pois o seu uso pode reduzir a facilidade de uso e o desempenho do DW.

Existe diferença entre a modelagem de um sistema de informação (SI) e a MD para um DW, no primeiro caso, é necessário um “entendimento operacional do negócio” que está sendo modelado, sem considerar, neste momento, a aplicação ou a tecnologia. Já na MD, que é o desenho de soluções de gestão de negócios, é necessário um “entendimento da gestão do negócio”, é preciso entender como os executivos e gestores da organização avaliam as informações para a adoção de estratégias e para a tomada de decisões (Costa et al., 2000).

3.5.1. Granularidade

A granularidade é o nível de detalhe ou de resumo dos dados contidos no DW. Quanto menor a granularidade, mais detalhado; quanto maior a granularidade, menos detalhado. Afeta profundamente o volume de dados do DW e também o tipo de pergunta que pode ser respondida pelo DW. O balanceamento da granularidade deve ser examinado cuidadosamente no início do projeto do DW (Inmon, 1997).

Uma granularidade muito alta pode impedir solicitações que examinam os detalhes que compõem um resumo (operação conhecida como *drill down*). Enquanto que granularidades muito baixas aumentam a quantidade de dados armazenados e o tempo de resposta das consultas. Por essas razões, Turban et al. (2009) orientam que o nível de granularidade necessário deve ser identificado durante a definição do escopo do projeto de DW.

Se granularidade é o nível de detalhe do DW, o grão é o significado de uma linha da tabela de fatos (Kimball, Ross, 2002). Com relação à variação do grão, existem três diferentes tipos de tabelas de fatos, conforme apresentado no Quadro 2. Nas tabelas de fato de transação, cada linha é um evento de uma transação; nas tabelas de instantâneo periódico, cada linha representa um período de tempo (previsível); e, por fim, nas tabelas de instantâneo cumulativo, cada linha significa o tempo de vida (indeterminado) de uma transação.

Quadro 2 – Comparação de tipos de tabela de fatos

Característica	Tipo de tabela de fatos		
	de Transações	de Instantâneo Periódico	de Instantâneo Cumulativo
Período de tempo representado	Momento	Intervalos regulares, previsíveis	Intervalo de tempo indeterminado, normalmente de vida breve
Grão	Uma linha por evento da transação	Uma linha por período	Uma linha por vida
Cargas da tabela de fatos	Inserção	Inserção	Inserção e atualização
Atualizações da linha de fatos	Não revisitado	Não revisitado	Revisitado em momento de atividade
Dimensão Data	Data da transação	Data do final do período	Múltiplas datas para marcos padrão
Fatos	Atividade da transação	Desempenho para intervalo de tempo predefinido	Desempenho em tempo de vida finito

Fonte: Kimball e Ross (2002, pág. 153)

Os DWs podem apresentar diferentes níveis históricos e diferentes níveis de sumarização e detalhe. Os dados atuais representam os acontecimentos mais recentes, geralmente os de maior interesse, e são os dados que apresentam o maior volume – por possuírem o menor nível de granularidade. Os dados antigos são acessados com menor frequência e em um nível de detalhe mais baixo (maior nível de granularidade), porém consistente com o detalhe dos dados atuais. Os dados atuais e/ou antigos podem ser sumarizados em diferentes níveis de acordo com a necessidade de processamento e armazenamento (Singh, 2001).

3.6. Arquitetura de um *Data warehouse*

Segundo Dias et al. (2008), a arquitetura de DW não se originou da Arquitetura de Software que compreende apenas o conjunto de programas, procedimentos, dados e documentação associada a um software. A arquitetura de DW teve origem nos conceitos e modelos definidos para Arquitetura de Sistemas de Informação (ASI) que abrange – desde a definição de estratégia de negócio até a implementação, todos os componentes necessários para o desenvolvimento e implantação de SI.

Portanto, para definir uma arquitetura de DW, os componentes relacionados à arquitetura de dados, infraestrutura e tecnologias necessárias na construção e utilização do DW devem ser considerados (Dias et al., 2008)

A arquitetura de um DW pode ser definida como uma descrição dos componentes, com detalhes que mostram como eles se encaixam e como o sistema irá crescer ao longo do tempo (Hadley, 2002). O crescimento de um DW pode ocorrer em função do aumento dos dados e da necessidade de expansão para atender novas funcionalidades do negócio (Turban et al., 2009).

Os documentos, planos, modelos, projetos e especificações da arquitetura devem possuir detalhes suficientes de cada componente para permitir sua implementação. Hadley (2002) afirma ainda que sempre existe uma arquitetura, seja ela planejada ou não, porém, arquiteturas planejadas têm mais chance de sucesso.

Segundo Sell e Pacheco (2001), definir a arquitetura do DW, determinando a organização de seus componentes, é tarefa crucial para o projeto. E, com o objetivo de evitar desperdício de tempo e consequentes atrasos no projeto, esta arquitetura deve ser determinada no início do projeto (Machado, 2006).

A escolha de uma arquitetura é necessária para cada tipo de aplicação em que o DW é implementado, de forma a suprir as necessidades identificadas na análise de requisitos. Assim, é necessário fazer um estudo das arquiteturas de DW existentes e escolher a mais adequada ou criar uma nova arquitetura, adaptando arquiteturas já existentes (Menolli, Dias, 2006).

Vários autores descrevem as arquiteturas de DW existentes (Clemes, 2001; Sell, Pacheco, 2001; Menolli, Dias, 2006; Machado, 2006), porém não existe um consenso com relação às nomenclaturas. Ariyachandra e Watson (2008) consideram cinco arquiteturas de DW como principais e observam ainda que muitas outras são apenas variações das principais. São elas:

- Arquitetura de DMs independentes: os DMs são independentes de outros dados armazenados e atendem necessidades localizadas e específicas, como prover dados para determinado departamento ou área de negócio, não há foco corporativo, nem integração com outros DMs.
- Arquitetura *bus*, apresentada por Kimball (1998): possui DMs que suportam várias áreas de assunto (ou processos de negócio). O primeiro DM é construído para uma área de assunto única usando dimensões que serão reaproveitadas em outros DMs. DMs adicionais são desenvolvidos usando estas mesmas dimensões, o que resulta em DMs integrados e visão corporativa dos dados. A arquitetura de Kimball é mostrada na Figura 5.

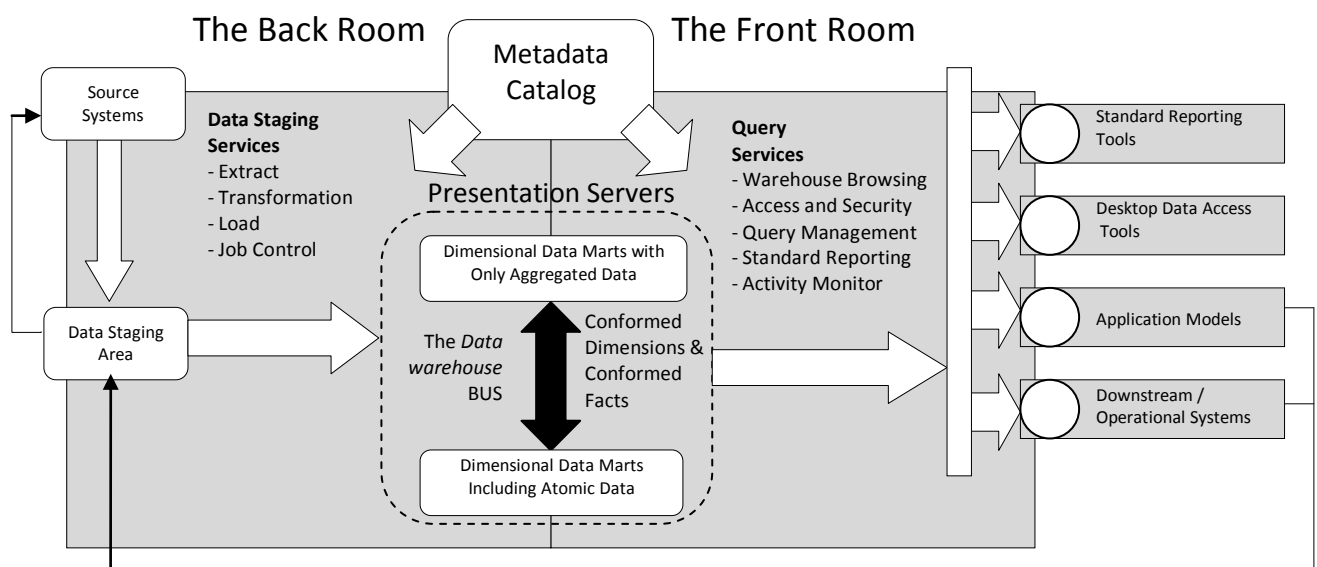


Figura 5 – Arquitetura Bus de Kimball

Fonte: Adaptado de Kimball et al (1998, pág. 329)

- Arquitetura *hub and spoke*, apresentada por Imnon (1997): inicia com uma análise corporativa dos requisitos de dados. O foco está na construção de uma infraestrutura escalável e manutenível. Com base nesta visão corporativa dos dados, a arquitetura é desenvolvida de forma interativa, assunto por assunto. DMs dependentes podem ser desenvolvidos por departamento, por área funcional ou finalidades específicas, por exemplo, para mineração de dados.
- Arquitetura centralizada: é similar à arquitetura *hub and spoke*, exceto pelo fato de não haver DMs dependentes.
- Arquitetura *federated*: é utilizada quando há um ambiente de dados de suporte à decisão fragmentado e há necessidade de integrar ao menos alguns dados. Esse é o caso, por exemplo, de quando ocorrem fusões, aquisições e reorganizações das empresas.

3.7. Trabalhos Relacionados

Vários trabalhos (Clemes, 2001; Sell, Pacheco, 2001; Menolli, Dias, 2006, Novello, 2006; Ruggiero Junior, 2007) já foram desenvolvidos com o objetivo de definir uma arquitetura de DW para dar suporte a diferentes domínios. A seguir é apresentada uma breve descrição desses trabalhos:

Clemes (2001) apresentou uma arquitetura de DW para suporte ao sistema de informações gerenciais em uma Instituição de Ensino Superior, validando sua proposta com um estudo de caso na Universidade Federal de Santa Catarina.

Sell e Pacheco (2001) sugeriram uma arquitetura para DW que permite a inclusão de componentes isolados de forma que a construção do DW seja incremental, a validação da arquitetura ocorreu por meio do desenvolvimento de um SI para a Sociedade Brasileira de Oncologia Clínica.

Menolli e Dias (2006) definiram uma arquitetura de DW para gestão em ciência e tecnologia no Brasil, validando sua proposta sobre as bases de dados da Plataforma Lattes (CNPq – Conselho Nacional de Desenvolvimento Científico e Tecnológico) e da Plataforma Coleta (CAPES – Coordenação de Aperfeiçoamento de Pessoal de Nível Superior).

Novello (2006) apresentou uma arquitetura de DW para análise de processos de desenvolvimento de software e validou sua proposta por meio de uma experiência numa empresa prestadora de serviços de desenvolvimento e manutenção de software.

Ruggiero Junior (2007) propôs uma arquitetura para um DW distribuído, usando programação paralela, com o objetivo de analisar o comportamento dos clientes em um ambiente bancário. O autor validou sua proposta implementando o DW, carregando dados de uma base aleatória de nomes e executando consultas pré-estabelecidas para medir o desempenho em comparação com um DW centralizado.

Os trabalhos acima apresentados não integram os temas DW e GP de software, distribuídos ou não, como elaborado nesta pesquisa.

3.8. Considerações finais

Os conceitos de DW apresentados neste Capítulo – que incluíram a descrição dos principais componentes de um DW e das arquiteturas, foram necessários para fundamentar a proposta desta pesquisa.

O Capítulo a seguir apresenta a arquitetura proposta para um DW de apoio à GP-DDS. Nele são apresentados os requisitos de apoio à decisão, a arquitetura de DW proposta, um modelo de dados criado a partir dos requisitos levantados e um roteiro para apoiar a implementação de um DW baseado na arquitetura definida nesta pesquisa.

Arquitetura de *Data warehouse* para Gestão de Projetos em DDS

Para melhor entendimento da arquitetura proposta, este Capítulo foi dividido em duas partes. A primeira parte trata do levantamento dos requisitos de apoio à decisão na GP-DDS como base para a arquitetura.

Na segunda parte é apresentada a arquitetura de DW para apoio à GP-DDS contendo uma análise de diversos aspectos considerados para a escolha da arquitetura, bem como a descrição dos componentes da arquitetura proposta, com destaque para a modelagem de dados para o DW.

4.1. Requisitos de Apoio à Decisão na Gestão de Projetos em Desenvolvimento Distribuído de Software

Antes de definir a arquitetura para apoiar a construção do DW, foram definidas algumas etapas para o levantamento de requisitos de apoio à decisão, com base em Thomsen (2002), Corey et al. (2001), Machado (2006) e Costa et al. (2000). As etapas são: definição dos objetivos globais das organizações que desenvolvem software, levantamento das necessidades dos gerentes de projeto de software, construção do modelo de dados conceitual do negócio, definição das áreas de assunto para GP-DDS e levantamento das informações para GP de software.

Inicialmente, foram definidos os objetivos globais da organização onde será implementado o DW. Em seguida é realizado um levantamento das necessidades dos usuários finais do DW, neste caso, os gerentes de projeto em DDS, definindo perguntas (requisitos de informação do usuário) cujas respostas auxiliariam no processo de tomada de decisão. Com base nessas necessidades, um modelo de dados conceitual do negócio (ou modelo de dados empresarial) foi construído, em alto nível, apenas representando as entidades principais e seus relacionamentos, o que permite uma visualização do todo de forma integrada e facilita o trabalho de inclusão das áreas de assunto. A partir desse modelo e das necessidades dos gerentes de projeto, foram definidas áreas de assunto ou processos de negócio que serão tratados pelo DW. A última etapa de requisitos abrange a definição das informações necessárias para atender os requisitos dos usuários e a classificação dessas informações dentro das áreas de assunto já definidas.

4.1.1. Objetivos globais das organizações que desenvolvem software

O DW é conduzido de forma diferente de um projeto tradicional. Durante a etapa de análise de requisitos, é necessário identificar os objetivos da organização onde se pretende implementar o DW (Thomsen, 2002). Sendo a primeira versão do DW focada no gerente de nível tático ou gerencial, pode-se considerar, de modo genérico, que as empresas de desenvolvimento de software, estando trabalhando com desenvolvimento local ou distribuído, têm como principais objetivos a redução de custo e o aumento da produtividade e da

qualidade. Para alcançar estes objetivos, os gerentes de projeto precisam tomar decisões pautadas em informações concretas.

Shimizu (2001) apresentou diversos problemas para exemplificar a diversidade e a abrangência das decisões na área de tecnologia da informação (TI). Alguns exemplos de decisões relacionadas à GP em nível tático são: desenvolvimento e manutenção com equipes próprias ou externas (*outsourcing* e *insourcing*); centralização *versus* descentralização das atividades de TI; seleção de tecnologias de pesquisa e desenvolvimento (P&D); escolhas para implantação e substituição de plataformas de hardware e software; escolha de sistemas (comunicação, suporte, análise, modelagem e outros); custos *versus* benefícios de projetos, tecnologias e estratégias; alocação de recursos e projetos; e avaliação de riscos.

Para dar suporte a essas e outras decisões, os gerentes de projeto de software têm necessidade de analisar fatos e acontecimentos que ocorreram ou que estão ocorrendo. A arquitetura de DW para GP-DDS proposta nesta pesquisa pretende facilitar essa tarefa de análise por meio de consultas aos dados disponíveis no DW.

4.1.2. Necessidades dos gerentes de projeto de software

Os gerentes de projetos de software possuem diversas ferramentas para auxiliar seu trabalho, como gerenciadores de cronograma e atividades, planilhas eletrônicas, entre outros. Porém, os relatórios gerados com base nas informações armazenadas nessas ferramentas podem apresentar algumas limitações como: não permitirem determinados tipos e quantidades de cruzamentos de dados, não armazenarem informações históricas suficientes, algumas informações podem não estar disponíveis em um mesmo formato, não oferecer a resposta em tempo hábil.

Com base nos diversos tipos de decisões que um gerente precisa tomar ao longo do planejamento e acompanhamento de um projeto de desenvolvimento de software, foram relacionadas abaixo algumas perguntas cujas respostas poderiam auxiliar nesse processo, com foco para as decisões dos gerentes cujos projetos são desenvolvidos de forma distribuída.

As perguntas agrupam diversas possibilidades de perspectivas utilizando o termo “por”, porém nem todas as perspectivas apresentadas em uma mesma questão podem ser usadas em conjunto, foram apresentadas na mesma questão para evitar repetição de perguntas e para apresentar as diversas opções de cruzamento relacionadas ao mesmo assunto.

1. Quanto tempo foi gasto para executar determinada atividade, por funcionário, por localização, por domínio, por fase, por nível de dificuldade, por tecnologia¹, em determinado período de tempo?
2. Quão próximo ficou a estimativa de custo em relação ao realizado, por equipe, por domínio, por local, por metodologia, por projeto, por fase, por atividade, por gerente, no decorrer de determinado período de tempo?
3. Quão próximo ficou a estimativa de tempo em relação ao realizado, por equipe, por domínio, por local, por metodologia, por projeto, por fase, por atividade, por funcionário, por gerente, no decorrer de determinado período de tempo?
4. Quanto tempo de experiência os desenvolvedores possuíam em determinado domínio, por localização, por equipe, no decorrer de determinado período de tempo?
5. Quantas solicitações de mudança nos requisitos ocorreram por analista de requisitos, por tecnologia, por domínio, por local, em determinado período de tempo?
6. Quantas solicitações de mudança na interface ocorreram, por projeto, no decorrer de determinado período de tempo?
7. Quantos funcionários possuíam conhecimento em algum tipo de tecnologia, por equipe, historicamente, no decorrer de determinado período de tempo?
8. Quanto tempo foi gasto com o desenvolvimento de projetos em um domínio, por localidade, em determinado período de tempo?
9. Quanto tempo de treinamento foi realizado por funcionários, por tecnologia, no decorrer de um período de tempo?
10. Qual a variação no nível de conhecimento em uma tecnologia, historicamente, antes e depois da realização de treinamentos relacionados àquela tecnologia, por funcionário, por determinado período de tempo?
11. Qual o índice de atraso para atividades concluídas de determinada tecnologia, mensalmente, por fase, em um determinado período de tempo?
12. Qual o índice de atividades concluídas no prazo e antecipadamente, por tecnologia, por fase, por funcionário, no decorrer de determinado período de tempo?
13. Qual o índice de reprogramação de atividades, por funcionário, por fase, no decorrer de determinado período de tempo?

¹ O termo “Tecnologia” foi aqui utilizado como forma de abranger toda a gama de recursos, não humanos e não materiais, que podem ser utilizados em um projeto. Assim, podem existir treinamentos ou níveis de experiência ou de conhecimentos relacionados a essas “tecnologias”. Exemplos de tecnologias, no contexto desta pesquisa: linguagens de programação, metodologias, bancos de dados, frameworks, ferramentas, entre outros.

14. Qual a quantidade de atividades retornadas para correção (não homologadas), por desenvolvedor, por local, por projeto, por equipe, por fase, por tecnologia, por linguagem de programação, por metodologia, por framework, por arquitetura, por domínio, no decorrer de determinado período de tempo?
15. Qual o número de atividades atribuídas a cada funcionário, no decorrer de determinado período de tempo, por localidade, por equipe, por fase, por projeto?
16. Qual o número de atividades atribuídas a cada equipe, no decorrer de determinado período de tempo, por localidade, por equipe, por fase, por projeto?
17. Qual o tempo de duração de cada tipo de atividade historicamente em diferentes projetos, por domínio, por tecnologia, por fase, por linguagem?
18. Qual o custo de cada tipo de atividade historicamente em diferentes projetos, por domínio, por tecnologia, por fase, por artefato, por linguagem?
19. Qual a variação no nível de conhecimento da equipe, historicamente, por tecnologia, em determinado período de tempo? Qual a relação desta variação com o tempo de treinamento recebido neste mesmo período?
20. Quanto tempo de experiência os desenvolvedores possuíam em determinada tecnologia, por papel, por localização, por equipe, no decorrer de determinado período de tempo?

Com base nos conceitos de DW, pode-se nomear as respostas a essas perguntas como fatos, sendo que os valores são as métricas dos fatos e os assuntos e perspectivas são as dimensões dos fatos.

4.1.3. Modelo de dados conceitual do negócio

Com base nas necessidades apresentadas, foi possível identificar algumas entidades chaves necessárias ao apoio da GP de software. Essas entidades foram reunidas para representar o modelo lógico em alto nível do conhecimento do negócio de GP-DDS para tomada de decisão. Foi empregado um diagrama entidade relacionamento (DER) com as entidades básicas e seus relacionamentos mais importantes.

Para a construção do DER foi utilizada a versão 2.0 da ferramenta brModelo, desenvolvida por Cândido (2007).

O modelo apresentado na Figura 6 define a atividade como a menor unidade do projeto, sendo que cada atividade pertence a uma fase de um projeto. Uma atividade possui um custo e um cronograma e pode gerar vários artefatos. A atividade depende de recursos materiais e é executada por pessoas de acordo com papéis. As pessoas e os recursos materiais estão fisicamente situados em um determinado local. As atividades fazem uso de diversas tecnologias e as pessoas podem possuir conhecimento e/ou experiência nessas tecnologias. Existem treinamentos relacionados às tecnologias que podem ser cursados pelas pessoas. As pessoas são organizadas em equipes que podem ser responsáveis por fases do projeto.

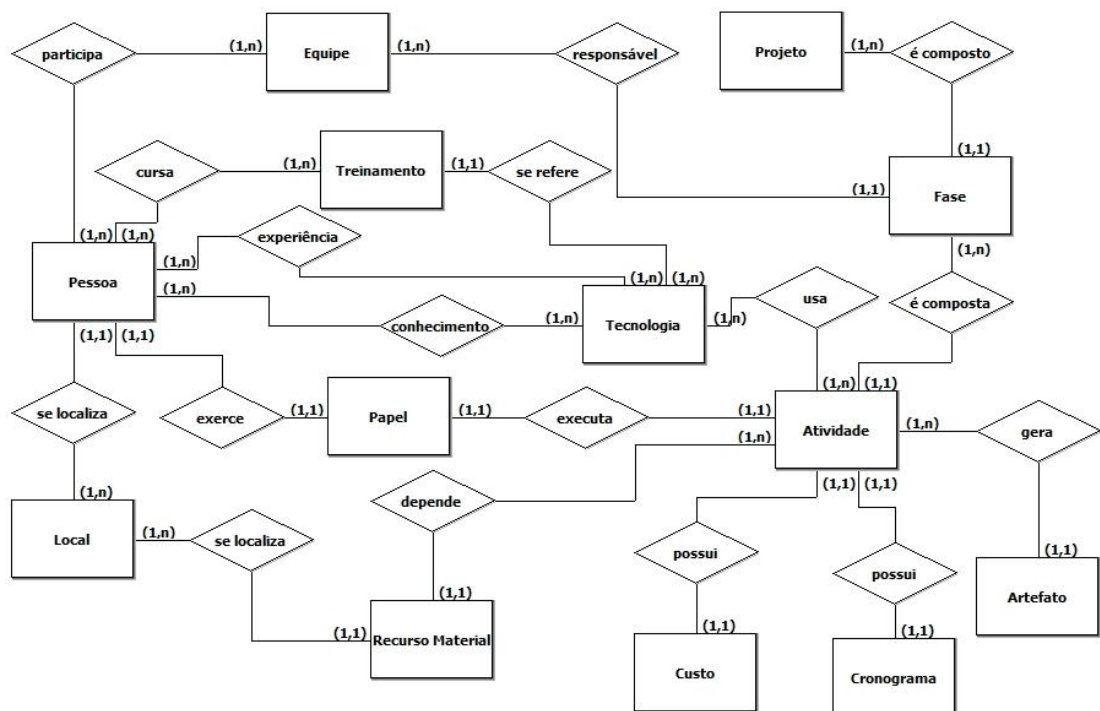


Figura 6 – Modelo de Dados Conceitual para GP-DDS

O modelo da Figura 6 deve ser adaptado ao contexto de cada empresa. Existem empresas que consideram a atividade como composta de várias tarefas, sendo que uma atividade pode estar relacionada a vários recursos humanos. Em outras, as equipes são representadas pelos locais onde os recursos estão alocados e, portanto, não seria necessário representar a equipe e o local.

As considerações feitas no DER apresentado aqui foram necessárias para que se pudesse realizar o levantamento de requisitos para o DW, sendo indispensável sua adaptação ao ambiente da empresa para a qual o DW será desenvolvido.

Com o modelo conceitual definido, foi possível realizar um levantamento mais detalhado das informações essenciais para atender as necessidades dos gerentes de projeto de DDS e classificar essas informações em áreas de assunto.

4.1.4. Áreas de assunto para gestão de projetos em desenvolvimento distribuído de software

Para definir as prováveis áreas de assunto – também conhecidas como processos de negócio, existentes na GP-DDS, foram estudadas as seis perspectivas apresentadas por Magalhães e Pinheiro (2007) para a implantação de ITIL (*Information Technology Infrastructure Library* – a biblioteca de melhores práticas para gestão de serviços de TI): Progresso, Custo, Recursos, Escopo, Riscos e Qualidade. Foram selecionadas três que possuem uma maior relação com o nível organizacional que esta arquitetura de DW pretende atender – o nível tático. As três perspectivas escolhidas foram: Progresso, Custo e Recursos.

A área de assunto Progresso envolve cronogramas e procura responder questões sobre o quanto se está cumprindo o cronograma previsto para o projeto, analisando indicadores como: índice de atividades terminadas, índice de atividades terminadas com atraso, índice de atividades reprogramadas. No caso específico do DDS, é necessário considerar situações como o desenvolvimento *follow-the-sun* e a maior probabilidade de atividades ocorrerem em paralelo. Os históricos armazenados no DW podem favorecer a criação dos futuros cronogramas, auxiliando no planejamento de novos projetos, assim como facilitando a realização dos ajustes necessários durante o acompanhamento de projetos em andamento.

A área de assunto Custo refere-se ao orçamento do projeto e pretende auxiliar o controle de custos, cujo sucesso é completar o projeto no ou abaixo do orçamento. As metas orçamentárias precisam ser realistas e o histórico das previsões e mudanças que ocorreram em projetos anteriores auxilia na estimativa de novos orçamentos. Alguns indicadores importantes para o controle do projeto em andamento são: o índice do orçamento realizado, o índice de variação do custo, o índice de desempenho dos custos orçados e o índice de desempenho dos custos para o término do projeto. Para DDS, os diferentes custos existentes em cada localidade, sejam eles relacionados às pessoas, estruturas físicas ou tributos, precisam ser levados em consideração. Além disso, caso as equipes estejam distribuídas em diferentes países com diferentes moedas, é necessário escolher uma unidade de moeda, por exemplo, o dólar, e registrar e tratar as conversões de moedas.

Na área de assunto Recursos espera-se que o DW auxilie a responder perguntas sobre o desempenho, a disponibilidade, a utilização e a adequação dos recursos alocados aos projetos. Esses recursos podem ser tanto humanos (pessoas) quanto físicos. A análise dos recursos pode ser realizada por meio de indicadores como índice de disponibilidade, índice de utilização, índice de *turnover* voluntário mensal, índice de horas extras mensal, índice de clima interno e índice de capacitação. Especificamente para DDS, é importante considerar a localização e o período de disponibilidade destes recursos.

4.1.5. Informações para gestão de projetos de software

Com base na literatura (Enami, Tait, Huzita, 2006; Huzita et al., 2007; Huzita, Tait, 2006; Kerzner, 2006; Lima, 2004; Magalhães, Pinheiro, 2007; Pedras et al., 2004; PMI, 2009; Prikladnicki, 2003; Shimizu, 2001; Smith, 1997; Turban, McLean, Wetherbe, 2004; Vieira, 2007) e em algumas ferramentas disponíveis para GP, foram identificadas informações necessárias para suportar os processos de decisão relacionados à GP de software. Nesta Seção estão relacionados os dados que podem ser aplicáveis tanto para a GP com desenvolvimento distribuído como para GP com desenvolvimento centralizado.

Para o levantamento das informações, foram observados os dados armazenados por diversas ferramentas. Entre elas, algumas das ferramentas analisadas por Vieira (2007), que são listadas a seguir: Task Tracking, PERT Chart Expert, MindManager, Microsoft Project, Primavera TeamPlay, PMOffice, PS8. Além dessas aplicações, também foram analisadas: a ferramenta DIMANAGER (Pedras et al., 2004) que mais tarde foi incorporada ao DiSEN, e o próprio ambiente DiSEN.

Algumas informações ganham maior importância no contexto do DDS para a GP, entre elas: a localização de cada recurso (seja ele um participante do projeto ou um recurso material – como equipamentos e ferramentas), a localização do cliente (para verificação da proximidade com a equipe), o período de disponibilidade de cada recurso, as responsabilidades dentro do projeto, o perfil e a aptidão de cada membro, os idiomas falados por cada membro, o país de origem, a moeda local e as restrições culturais (Enami, Tait, Huzita, 2006).

O levantamento realizado nesta Seção possui foco no desenvolvimento distribuído, porém, é possível adaptá-lo ao contexto do desenvolvimento local, analisando e, se necessário, retirando as informações que não são relevantes na tomada de decisão dos gerentes de projetos de software no contexto do desenvolvimento centralizado.

Os dados necessários à tomada de decisão na GP-DDS que foram obtidos nesse levantamento, são listados a seguir. Eles foram agrupados conforme as áreas de assunto definidas na Seção anterior – Progresso, Custo, Recursos, Escopo, Riscos e Qualidade. Considerando que existem alguns dados que são essenciais para todas as áreas de assunto, como é o caso dos projetos, foi criada uma área de assunto denominada “Comum” para conter esses dados especiais:

a) Área de assunto Comum

- Artefato: produto gerado por uma atividade, por exemplo: diagramas, modelos, manuais, casos de teste, ata de reunião, códigos, entre outros. Pode fazer parte de um pacote de trabalho ou entrega. Pode ser classificado em tipos, está relacionado com um recurso humano, uma atividade, uma fase e um projeto. Pode estar relacionado a um ou mais recursos materiais e tecnológicos.
- Domínio de Problema: representa a área de domínio associada ao projeto, como por exemplo: fiscal, transporte, vendas, educação, comércio eletrônico, entre outras.
- Local: um local pode estar relacionado a recursos humanos, recursos materiais, clientes, usuários e, também, a um ou mais idiomas.
- Atividade: tarefa necessária para conclusão do projeto, por exemplo: projeto físico dos dados, projeto de arquitetura, submissão de especificação para aprovação, testes de desempenho, desenvolvimento de um componente, modelagem de um caso de uso, entre outros. Inclui detalhes como: datas de início e término (previstas e realizadas), quantidade de horas (previstas e realizadas), estado (em planejamento, atribuída, em andamento, concluída, cancelada), porcentagem de conclusão e nível de complexidade. Pode ser classificada em tipos (desenvolvimento, manutenção, adaptação, entre outros). Uma atividade pertence a um projeto e está relacionada a um recurso humano por meio de um papel, que deve ser desempenhado para atender a atividade. Pode estar relacionada a um ou mais recursos materiais e tecnológicos, pode depender de outra atividade e pode ser requisito de outra atividade (atividade sucessora e atividade predecessora).
- Fase: etapas em que um projeto é dividido. A fase pode ser baseada em uma metodologia de desenvolvimento e inclui detalhes como datas de início e término (previstas e realizadas), quantidade de horas (previstas e realizadas), estado (em andamento, concluída) e porcentagem de conclusão. Está relacionada a um projeto, a

um ou mais recursos humanos, materiais e tecnológicos. Exemplos de fases: requisitos, análise, projeto, implementação e teste.

- **Projeto:** conjunto de atividades com objetivo de desenvolver um software. Inclui detalhes como: recurso humano responsável, datas de início e término (previstas e realizadas), quantidade de horas (previstas e realizadas), estado (em contratação, em andamento, concluído), porcentagem de conclusão, nível de prioridade, nível de complexidade, restrições do projeto, pode pertencer a um portfólio. Está relacionado a um ou mais recursos humanos, materiais e tecnológicos.
- **Reunião:** tipo especial de atividade. Uma reunião possui participantes, pode ser virtual ou presencial, gera uma ata, armazena decisões e pode gerar um mapa mental. Está relacionada a um projeto e uma fase e, também, pode estar relacionada a uma atividade. Pode gerar novos requisitos e atividades ou alterações nos requisitos e nas atividades existentes.

b) Área de assunto Custo

- **Custo:** diz respeito aos valores previstos e realizados relacionados aos recursos humanos e materiais e que podem estar associados a uma unidade de moeda, podendo ser classificados em tipos ou contas contábeis (despesas com viagem, equipamentos, contas telefônicas, despesas trabalhistas, entre outros). Pode ser agrupado para gerar o custo de uma atividade, de uma fase e de um projeto.

c) Área de assunto Qualidade

- **Problema:** inclui detalhes como solução proposta, solução aplicada e resultado da aplicação da solução. Pode ser classificado em tipos (técnico, organizacional, comportamental; interno ou externo; previsto ou não previsto). Pode estar relacionado a um projeto, uma fase e uma atividade.
- **Solução:** solução proposta e aplicada. Está relacionada a um problema.
- **Métrica:** relacionada a um projeto, pode estar relacionada a uma atividade ou fase. Exemplos: pontos por função, pontos por caso de uso, estimativa de esforço, entre outras.

d) Área de assunto Progresso

- Cronograma: inclui detalhes como: quantidade de tempo prevista e realizada, período de tempo previsto e realizado (data inicial e data final), estado (adiantado, atrasado, em dia), porcentagem de conclusão e pode indicar os marcos do projeto (*milestones*). Está relacionado a uma atividade, uma fase, um projeto, a um ou mais recursos humanos e materiais. O cronograma deve armazenar também o sequenciamento (ordem de dependência) das atividades e das fases do projeto.

e) Área de assunto Recurso

- Recurso: pode ser humano, material ou tecnológico, pode estar relacionado a um local ou ser global, pode estar disponível ou não, pode ser atribuído a uma ou mais atividades de um ou mais projetos. Pode ser requisito de uma atividade ou pode estar atribuído a uma atividade.
- Recurso Material: itens necessários ao desenvolvimento do projeto. Pode ser próprio ou de terceiros, móveis ou fixos, duráveis ou de consumo, compartilhável ou exclusivo. Podem ser classificados em tipos e possui custo. Está relacionado a um local que pode variar com o tempo. Exemplo: ferramenta, software, licença, rede, servidor, computador, material de escritório, acesso à Internet, linha telefônica, imóvel, entre outros.
- Recurso Tecnológico: nesta pesquisa, o termo recurso tecnológico agrupa os recursos não humanos e não materiais necessários ao desenvolvimento de sistemas, como por exemplo: linguagens de programação, metodologias de desenvolvimento, bancos de dados, frameworks, arquiteturas e ferramentas. Pode possuir custo ou não, pode ser classificado em tipos (metodologia de desenvolvimento, processo de desenvolvimento, linguagem de programação, linguagem de modelagem, entre outros). Exemplos de recursos tecnológicos: *Scrum*, RUP, XP (*extreme programming*), Java, JSF, C, *Natural*, UML, OO (orientação a objetos), abordagem estruturada, entre outros.
- Recurso Humano: pode ter afinidade com outro recurso humano (rede social), tem relação com competências (que pode ocorrer de formas diferentes: como competências já adquiridas e como competências sugeridas para alinhamento com os objetivos de carreira). Pode possuir experiência, possui um custo, possui tipos de disponibilidade (disponível, em férias, afastamento médico, treinamento, folga, entre outros), pode assumir diferentes papéis em diferentes atividades, está relacionado a um ou mais

idiomas, está relacionado a um local que pode variar com o tempo e que pode ser do tipo *home-office* ou não.

- **Competência:** relacionada ao recurso humano, é composta por conhecimentos, habilidades e atitudes. Pode possuir níveis (por exemplo, baixo, médio, alto, muito alto), pode ser requisito essencial ou desejável para atribuição de uma atividade a um recurso humano.
- **Conhecimento:** pode estar relacionado a um recurso tecnológico ou não, pode estar relacionado a um domínio de problema ou não, pode estar relacionado a uma fase ou não. Exemplo: lógica de programação, ferramenta de desenvolvimento, modelagem de dados, análise de processos, domínio de problema, banco de dados, entre outros.
- **Habilidade:** Exemplo: comunicação verbal e escrita, fluência em idioma, trabalho em equipe, capacidade de atuar sobre pressão, liderança, entre outras.
- **Atitude:** características não-técnicas. Exemplo: iniciativa, pró-atividade, comprometimento com qualidade, observância de prazos, entre outras.
- **Cargo:** registrado em contato de trabalho em determinado período de tempo, está relacionada a um recurso humano. Exemplo: analista de sistemas júnior, analista de produção sênior, administrador de banco de dados, designer gráfico, entre outros. Um recurso humano contratado para determinado cargo pode assumir diferentes papéis em diferentes projetos.
- **Equipe:** grupo de pessoas. Está relacionado com recursos humanos e pode estar relacionada com local, atividade, projeto ou papel.
- **Experiência:** pode estar relacionada a um recurso humano, pode possuir quantidade de horas, período de tempo e recenticidade. Pode ser interna (dentro da organização, possível de ser rastreada por meio dos projetos em que participou) ou externa (fazem parte do currículo). Pode ser requisito essencial ou desejável para atribuição de uma atividade a um recurso humano. Pode estar relacionada a um recurso tecnológico e/ou a um domínio de problema e/ou a uma fase.
- **Papel:** assumido por um recurso humano por determinado período de tempo em uma atividade. Exemplo: programador, analista de requisitos, testador, gerente de projetos, entre outros.
- **Treinamento:** possui nível, carga horária, recenticidade e pode ser classificado em tipos (curso presencial, palestra, *workshop*, pós-graduação, curso à distância, entre

outros). Está relacionado a recursos humanos, pode ter relação com recursos tecnológicos ou não, tem relação com conhecimentos e habilidades.

f) Área de assunto Risco

- Risco: inclui detalhes como a probabilidade de o risco ocorrer. Pode ser classificado em tipos de risco (para o negócio, para o produto, para o projeto). Pode possuir tratamento (preventivo ou corretivo) e medidas de contenção, pode estar relacionado a um ou mais projetos e pode estar relacionado a um local.

4.2. Arquitetura de Date Warehouse para Gestão de Projetos em Desenvolvimento Distribuído de Software

Para a definição de uma arquitetura de DW para GP-DDS foram analisados vários fatores que influenciam no processo de escolha. A análise levou em consideração os levantamentos de requisitos e das necessidades dos gerentes de projeto realizados na primeira Seção deste Capítulo.

Após a escolha do tipo de arquitetura que melhor se adapta às necessidades dos gerentes de projeto em DDS, é apresentada uma proposta de arquitetura de DW para GP-DDS, contendo uma descrição dos seus componentes.

Em seguida, é recomendado um modelo de dados dimensional para a arquitetura de DW proposta, com base no modelo de dados conceitual definido na Seção 4.1.3 – Modelo de dados conceitual do negócio. O modelo evidencia os fatos e dimensões necessários para atender às necessidades estabelecidas nos requisitos de apoio à decisão na GP-DDS.

A última parte desta Seção apresenta um roteiro para implementar um DW, com vistas a facilitar a utilização da arquitetura de DW para GP-DDS proposta nesta pesquisa.

4.2.1. O processo de escolha da arquitetura

Machado (2006) cita diversos fatores e variáveis que devem ser avaliados para a escolha da arquitetura ideal para o projeto de DW: infraestrutura disponível, ambiente de negócio (porte da empresa), escopo de abrangência desejado, recursos disponibilizados ou projetados para investimento, retorno do investimento a ser realizado, satisfação do usuário e local onde o DW ou os DMs estarão residindo. Esses fatores são tratados abaixo levando em conta que o DW proposto deve armazenar dados de projetos de software que são desenvolvidos por equipes distribuídas e que a utilização desses dados tem como objetivo auxiliar a GP.

4.2.1.1. Infraestrutura disponível

A arquitetura de DW deve ser genérica o suficiente para atender infraestruturas diferentes. É possível que os dados coletados para serem armazenados no DW estejam fisicamente distribuídos ou não; os dados também podem estar armazenados em diferentes sistemas (formatos).

É possível prever que em alguns assuntos do DW para GP de software, como por exemplo, no conhecimento da equipe em determinada tecnologia, dados mais recentes (com menos de cinco anos, por exemplo) tendem a ser mais importantes do que dados mais antigos – visto que as tecnologias avançam rapidamente. Portanto, uma arquitetura com dois níveis de granularidade deve ser avaliada, de forma que os dados mais recentes sejam mais detalhados (menor granularidade) do que dados muito antigos; sendo que estes últimos poderiam ser armazenados no DW de modo resumido (sumarizado).

4.2.1.2. Ambiente de negócio (porte da empresa)

Mesmo em empresas de maior porte, que produzam um grande número de softwares e que possuam um grande número de equipes e funcionários de desenvolvimento, o volume de dados estimado para um DW que atenda as necessidades da GP tende a ser muito menor comparado a DWs que armazenem, por exemplo, dados de produtos vendidos em uma rede de supermercados.

Desse modo, o espaço para armazenar o DW não é um ponto crítico neste domínio, e isso deve ser considerado no momento da definição da granularidade da arquitetura do DW.

4.2.1.3. Escopo de abrangência desejado

O escopo do DW proposto envolve inicialmente apenas informações para gerentes de nível tático, porém, prevendo uma possível ampliação do escopo para atender informações estratégicas para gerentes de alto nível da organização, a arquitetura de DW deve ser flexível o bastante para absorver esse acréscimo, mantendo a integração dos assuntos e a visão corporativa dos dados.

Os DMs incrementais mostram-se como uma boa alternativa, pois permitem que novos DMs sejam integrados ao DW atual com o aproveitamento de dimensões já definidas.

4.2.1.4. Recursos disponibilizados ou projetados para investimento

O recurso destinado ao DW pode variar de acordo com a empresa de desenvolvimento de software na qual ele será aplicado, nesse caso, uma opção interessante que atende essa variação é a arquitetura de DW composta por DMs incrementais, abrangendo os assuntos na medida em que os recursos vão sendo disponibilizados.

4.2.1.5. Retorno do investimento a ser realizado

O retorno esperado com a criação do DW é a melhoria nas diversas decisões tomadas pelo gerente de projetos, como as que envolvem alocação de pessoas para equipe, alocação de equipes para o projeto, estimativas de tempo e custo. Essa melhoria reflete na satisfação do cliente, aumento do lucro, redução de riscos nas estimativas e conseqüentemente nos prejuízos decorrentes de estimativas fora da realidade. Os benefícios da utilização das informações disponibilizadas no DW são imediatos, o gerente de projetos pode mudar a gestão de um projeto em andamento e melhorar o planejamento dos próximos projetos com base nas informações disponibilizadas pelo DW.

A arquitetura de DMs incrementais oferece a possibilidade de que esse retorno seja analisado ao longo do tempo, sem a necessidade de aguardar até que todos os assuntos estejam disponíveis no DW para favorecer-se dos resultados (Machado, 2006).

4.2.1.6. Satisfação do usuário

É preciso definir estratégias de atualização do DW para que as informações sejam recentes o suficiente de modo a satisfazer as necessidades dos gerentes de projetos – usuários finais do DW. A arquitetura de DW deve permitir que as regras de atualização sejam configuradas conforme a necessidade da empresa.

4.2.1.7. Local onde o *data warehouse* ou *data marts* estarão residindo

Uma alternativa é manter os DMs em uma instalação central de forma que o gerente possa tomar decisões com uma visão global de todos os recursos que estão fisicamente distribuídos (pessoas, equipes, equipamentos, softwares e instalações físicas).

Outra opção é distribuir os diversos DMs entre os vários sites, criando um DW distribuído. Porém, apesar de um mesmo projeto de software permitir que o desenvolvimento esteja geograficamente distribuído entre equipes, diversas características sugerem que um DW distribuído não é a opção mais adequada neste contexto: a necessidade de controle local é mínima; as empresas em questão, em geral, possuem um único negócio (o desenvolvimento de software); os dados não têm importância exclusiva para o local, pelo contrário, é o agrupamento dos dados locais gerando um DW global que apresenta maior relevância por permitir uma visão corporativa dos recursos pelo gerente e facilitar a distribuição.

4.2.2. A arquitetura proposta

Considerando os sete fatores apresentados por Machado (2006) e as justificativas apresentadas na Seção 4.2.1 (O processo de escolha da arquitetura), optou-se por uma arquitetura baseada na arquitetura *bus* de Kimball (1998), que é composta por DMs integrados suportando diferentes áreas de assunto e oferecendo uma visão corporativa dos dados – sendo o tipo de arquitetura que melhor se aplicou ao contexto do DDS.

Portanto, esta pesquisa propõe uma arquitetura de DW global, também chamado de DW empresarial (EDW), composta por DMs incrementais por assunto, dentro de um plano global que prevê a integração desses DMs. A implementação deve iniciar por um projeto pequeno, com a escolha de uma área de assunto, para permitir um retorno rápido sobre o investimento.

O plano global envolve a definição de um DW empresarial lógico que considera as principais áreas de assunto do DW e seus relacionamentos. Durante a definição deste DW lógico, um macro modelo de dados é gerado, com o objetivo de garantir a consistência dos dados e a coerência entre os vários DMs que serão integrados, de forma a permitir que os usuários possam extrair dados que cruzem o limite entre as áreas de assunto.

A escolha de DMs incrementais torna a arquitetura flexível. Sendo assim, as áreas de assunto não estão limitadas a estas aqui apresentadas já que outros DMs, representando outras áreas de assunto que porventura sejam criadas, poderão ser acrescentados ao DW global a qualquer momento.

A seguir é apresentada a arquitetura de DW proposta para apoiar a GP-DDS. Como pode ser verificado na parte inferior da Figura 7, onde encontra-se a primeira camada, a arquitetura prevê duas situações: o momento da carga inicial (que deve ocorrer uma única vez) e os momentos de cargas periódicas.

A camada mais baixa da arquitetura chama-se Camada de Inicialização e Atualização. É por meio dela que ocorre a primeira carga do DW e a monitoração das fontes de dados para as cargas seguintes. Nesta camada é descrito como deve ser realizada a primeira extração das fontes de dados originais para o DW. Para esta atividade, são necessárias regras que definem como será inserida a indicação de tempo na tabela de dimensão Tempo. Essas regras podem definir que, para os registros que apresentem alguma indicação da data e hora de criação, esta data e hora de criação seja utilizada como indicação de tempo na tabela de dimensão Tempo, enquanto que os demais registros utilizam a data e hora do instante da carga.

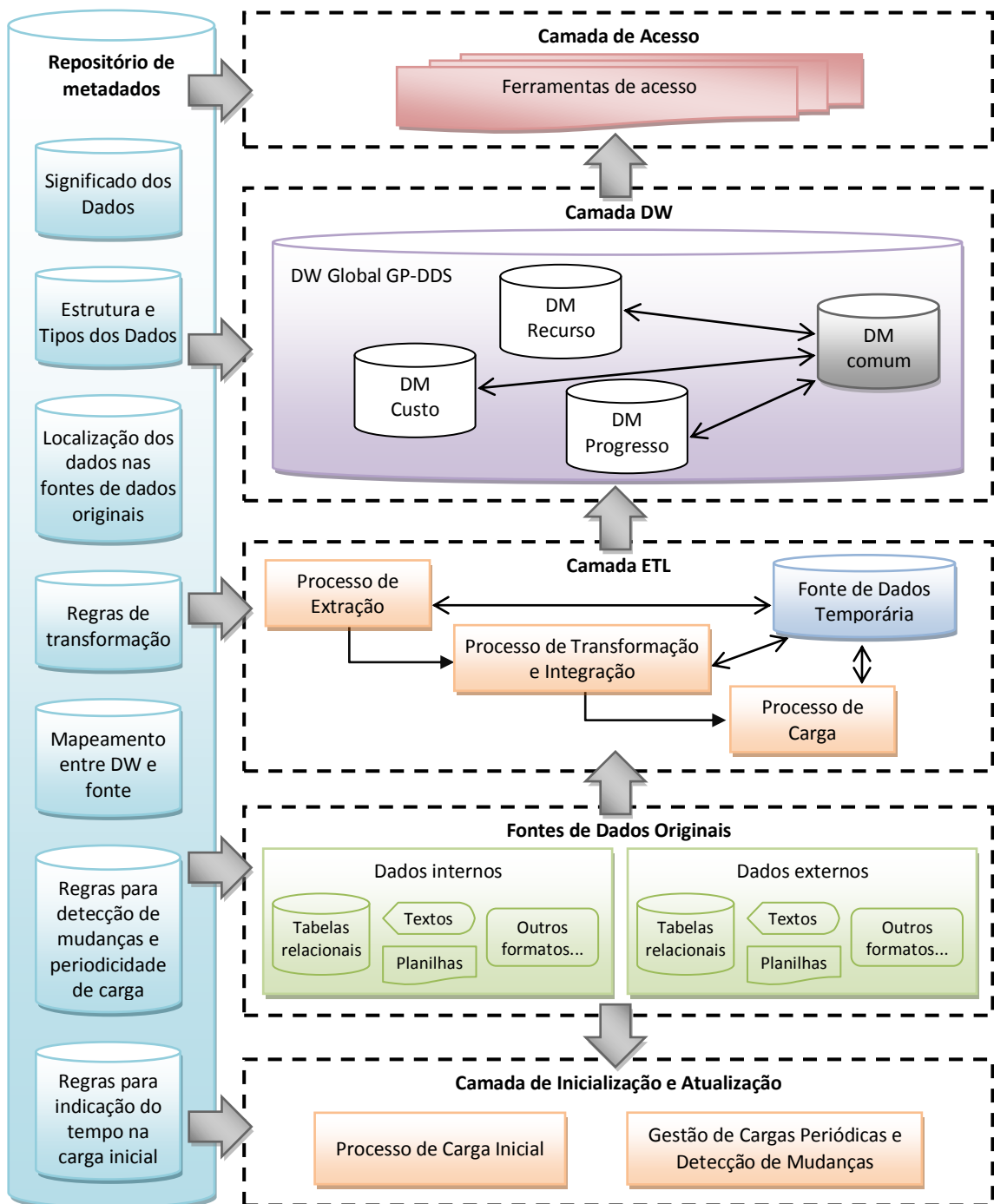


Figura 7 – Arquitetura de DW para GP-DDS

Fonte: Elaborada pela autora com base em Kimball e Ross (2002), Inmon (1997), Singh (2001), Corey et al. (2001)

A Camada de Inicialização e Atualização é responsável, também, por controlar as migrações periódicas de dados dos sistemas operacionais de origem (fontes de dados) para o DW. Essas atualizações ocorrem de acordo com regras definidas para decidir em que momento os dados devem ser reciclados. É possível utilizar uma ou mais técnicas, entre elas: comparações periódicas, com regras para determinar a frequência para determinada área de

assunto, por exemplo, semanalmente ou quinzenalmente; ou por meio de acionadores nas tabelas de origem. O uso de acionadores só é viável caso sejam permitidas alterações no sistema transacional de origem para detectar quando um registro é alterado, inserido ou excluído (Corey et al., 2001).

As fontes de dados podem ser divididas em fontes internas e externas. As fontes internas são os dados pertencentes à empresa, que podem estar armazenados em diferentes formatos como tabelas de banco de dados relacionais povoadas por sistemas transacionais, planilhas eletrônicas ou arquivos de textos. Já as fontes externas, que podem existir ou não, são dados externos que poderão ser utilizados como parâmetro de comparação entre a empresa ou organização onde foi implementado o DW e as demais organizações do mercado.

Ainda sobre as fontes de dados externas, alguns relatórios e pesquisas publicados periodicamente podem ser utilizados no contexto da GP, é o caso do Estudo de Benchmarking em Gerenciamento de Projetos do PMI (PMI, 2009) – um relatório anual sobre o gerenciamento de projetos no Brasil com perspectivas por setor e por porte de projeto. Este relatório apresenta informações como: a porcentagem de profissionais, por nível hierárquico, que recebeu treinamento formal em GP nos últimos 12 meses; o salário médio dos gerentes de projeto; o perfil dos profissionais que atuam no GP; as habilidades mais valorizadas e as deficientes nas organizações no GP.

A segunda camada, chamada Camada ETL, está presente em todas as arquiteturas de DW e é encarregada de extrair, transformar e carregar os dados no DW. Apesar de toda a arquitetura estar ligada diretamente ao Repositório de Metadados, a Camada ETL é a que mais depende dele. Nesta camada, encontra-se a Fonte de Dados Temporária, um local utilizado para armazenar os dados que vêm das fontes de dados originais enquanto ocorrem os processos de transformação e preparação, até que eles sejam carregados no DW.

O Repositório de Metadados armazena todas as referências para vincular cada registro do DW a sua fonte de origem. É ele também o responsável por indicar o significado e tipo de cada informação arquivada no DW, as regras para indicação do tempo na carga inicial, regras para detecção das mudanças no sistema de origem, regras para gestão das cargas periódicas e regras de transformação dos dados do sistema de origem para o DW. Todos esses dados sobre dados ficam localizados no Repositório de Metadados.

Diversas ferramentas auxiliam no gerenciamento automático de parte dos metadados, como por exemplo, ferramentas de ETL que gerenciam automaticamente metadados relacionados às regras de transformação e ao mapeamento dos sistemas de origem para o DW. Porém, apesar das ferramentas proverem recursos para o gerenciamento dos metadados de

negócio, cabe aos desenvolvedores do DW utilizarem esses recursos para a criação e manutenção desse tipo de metadado.

É na terceira camada, na Camada DW, que está localizado o DW físico, composto por um DW global e seus DMs integrados. Na arquitetura proposta, com foco nas necessidades dos gerentes de nível técnico, foram escolhidos três DMs iniciais, além do DM comum: DM Recurso, DM Custo e DM Progresso. Sendo que na primeira versão, o DW global conterà um DM comum e um DM de uma área de assunto a ser escolhida, conforme as necessidades da organização. O DM comum armazenará as tabelas comuns que podem ser utilizadas por diferentes DMs, como a tabela de dimensão Tempo.

A última camada da arquitetura de DW é a Camada de Acesso, por meio da qual os usuários do DW terão acesso aos dados por meio de consultas SQL ou ferramentas específicas.

4.2.3. Modelo de dados para o *data warehouse* – um modelo multidimensional de alto nível

Durante a modelagem dos dados, são determinados quais os dados necessários, quais as regras de negócios associadas aos dados e quais agregações e cálculos são necessários (Turban et al., 2009).

Com base nas necessidades apresentadas na Seção 4.1 (Requisitos de Apoio à Decisão na Gestão de Projetos em Desenvolvimento Distribuído de Software), foram identificadas as possíveis dimensões, fatos e hierarquias de um DW para GP-DDS. O modelo, que é apresentado a seguir, tem como objetivo servir de referência para DWs com foco na GP de software, sendo necessária sua adaptação conforme as necessidades e dados armazenados na organização onde o DW for implantado.

Para cada uma das perguntas apresentadas na Seção 4.1.2 (Necessidades dos gerentes de projeto de software) foram definidos os fatos e as medidas, com a intenção de reunir as perguntas que tenham como resposta as mesmas métricas. Foram definidas, também, as dimensões relacionadas a cada pergunta, tanto as dimensões necessárias para obter o fato resumido, como as possíveis perspectivas para cruzamento, representadas nas perguntas pela palavra “por”.

Como resultado da análise das perguntas propostas como necessidades dos gerentes de projeto foram obtidos sete fatos e doze dimensões. Um quadro contendo as definições dos fatos, medidas e dimensões para cada pergunta pode ser encontrado no Apêndice A. Com a

definição dos fatos e medidas foi possível verificar que várias perguntas poderiam ser respondidas pelo mesmo fato apenas variando o conjunto de dimensões. Sendo assim, foi realizado um agrupamento por fato e medida, resultando no Quadro 3 apresentado a seguir, que é detalhado na sequência.

Quadro 3 – Fatos e Medidas para um DW de apoio à GP-DDS

Fatos	Medidas
Atribuição de atividade	Contagem de registro de atribuição
Conhecimento e treinamento da equipe em tecnologia	Quantidade máxima de recursos humanos com conhecimento, Média de recursos humanos com conhecimento, Média do nível de conhecimento, Nível máximo de conhecimento, Tempo total de treinamento da equipe em horas, Tempo médio de treinamento por recurso humano em horas
Conhecimento e treinamento do recurso humano em tecnologia	Tempo de treinamento em horas, Nível de conhecimento
Custo da atividade	Valor do custo planejado em dólar, Valor do custo realizado em dólar
Experiência	Tempo de experiência em projetos internos em horas
Manutenção de requisito	Quantidade de manutenções
Tempo e quantidade de atividade	Tempo gasto em horas, Tempo planejado em horas, Quantidade de alterações no cronograma, Quantidade de atividades retomadas, Contagem de registros da atividade

4.2.3.1. Fatos

Os fatos apresentados no Quadro 3 são descritos a seguir, com uma referência para os esquemas estrelas correspondentes que estão ilustrados no Apêndice B. Durante a modelagem dos esquemas estrelas, foi possível visualizar vários outros agrupamentos possíveis, além de descobrir que outras dimensões poderiam ser associadas a determinados fatos. Tudo isso amplia a quantidade de perguntas que podem ser respondidas com os fatos e dimensões aqui definidos. A arquitetura incremental com DMs integrados foi construída para comportar-se dessa forma, à medida que dimensões e fatos vão sendo definidos e utilizados, novas possibilidades de cruzamentos são encontradas e novos atributos, dimensões e medidas são integrados ao DW para atender aos novos requisitos.

- Atribuição de atividade: permite a contagem de atividades atribuídas a um recurso humano, portanto, uma linha deve ser inserida nessa tabela todas as vezes que ocorrer o evento de atribuição de atividade. Ver Figura 13.
- Conhecimento e treinamento da equipe em tecnologia: permite verificar a quantidade de recursos humanos de uma equipe que possui conhecimento em determinada tecnologia, o nível de conhecimento e o tempo de treinamento da equipe. Considerando que as equipes podem sofrer alterações na sua formação com o decorrer do tempo, este fato representa as informações calculadas sobre os membros da equipe naquele momento do tempo. Este fato considera ainda que uma mesma equipe possa ser composta por membros de diferentes locais e, por isso, a dimensão Local não aparece nesse fato. O conceito de equipe é muito variável de empresa para empresa, pode representar o local onde ficam os recursos, ou o projeto em que trabalham, ou o cliente que atendem, ou uma determinada fase do projeto, entre outros. Por isso, outras medidas e dimensões podem ser adicionadas a este fato, de acordo com o conceito de equipe utilizado pela empresa. Ver Figura 14.
- Conhecimento e treinamento do recurso humano em tecnologia: esse fato representa o nível de conhecimento e a quantidade de horas de treinamento realizadas por cada recurso humano. Sendo o dia a menor granularidade da tabela Tempo, a quantidade de horas é uma medida aditiva, representando as horas de treinamento realizadas naquele dia. Ver Figura 15.
- Custo da atividade: armazena medidas de custo planejado e realizado relacionadas ao menor nível de granularidade, a atividade. A hierarquia da dimensão Atividade engloba a fase e o projeto, permitindo que sejam realizadas operações de agregação (*drill up*) até o nível máximo e operações de detalhamento (*drill down*) até o nível mais baixo, permitindo consultas sobre o custo da atividade, da fase e do projeto. Ver Figura 16.
- Experiência: representa a experiência já adquirida em projetos da empresa. Uma linha pode ser inserida nessa tabela de fatos todas as vezes que um recurso humano trabalhar com determinada tecnologia. O esquema estrela é apresentado na Figura 17.
- Manutenção de requisito: as atividades do tipo manutenção – sejam elas: corretivas, evolutivas ou preventivas; foram consideradas como uma dimensão separada da dimensão atividade que se limitou aos demais tipos de atividade com exceção do tipo manutenção. Isso permite que situações específicas de manutenção sejam medidas e

filtradas de maneiras específicas, não aplicáveis aos demais tipos de atividades, além de relacioná-las a outras dimensões, como a dimensão de requisitos do projeto. Ver Figura 18.

- Tempo e quantidade de atividade: esse fato contém relacionamentos muito parecidos com o fato Custo da atividade, porém, além de medir o tempo realizado e planejado, ele permite realizar uma contagem de atividades filtrando por situação, como: em andamento com atraso, em andamento no prazo, concluída com atraso, concluída antecipadamente, concluída no prazo, cancelada, entre outras. Ver Figura 19.

4.2.3.2. Dimensões

As dimensões obtidas após análise dos requisitos, apresentadas no Apêndice A, são descritas abaixo:

- Dimensão Artefato: resultado produzido por uma atividade. Como podem existir atividades que não estão associadas a nenhum artefato, é recomendável manter um registro com a descrição “Não aplicável” para ser utilizado por essas atividades durante os cruzamentos.
- Dimensão Atividade (Fase, Projeto): é a menor granularidade do projeto e possui a fase e o projeto como atributos e níveis de hierarquia.
- Dimensão Atividade de manutenção (Projeto): um tipo específico de atividade que possui detalhes específicos e que por isso foi separada da dimensão atividade. Essa dimensão possui o projeto como atributo e nível de hierarquia.
- Dimensão Domínio do problema: aborda a área de domínio associada ao projeto.
- Dimensão Equipe: formada por recursos humanos conforme critério do gerente e da empresa. Nesta pesquisa com foco em DDS, a equipe foi definida como composta por pessoas que podem estar em diferentes locais.
- Dimensão Local: local físico onde se encontra um recurso humano ou material.
- Dimensão Papel: assumido por um recurso humano em determinada atividade de determinada fase de um projeto.
- Dimensão Recurso Humano: pessoa responsável pela execução das atividades.
- Dimensão Requisito: condição do cliente a ser atendida pelo sistema em desenvolvimento, esta dimensão possui como atributo o tipo do requisito que pode assumir valores como, por exemplo, funcional, usabilidade, confiabilidade, desempenho, portabilidade, interface de usuário, entre outros.

- Dimensão Situação da Atividade: pode agrupar informações como a situação de execução da atividade em dado momento e situação da atividade dentro de um cronograma previsto, como por exemplo, em andamento com atraso, em andamento no prazo, concluída com atraso, concluída antecipadamente, concluída no prazo, cancelada, retomada, entre outras.
- Dimensão Tecnologia: no contexto desta pesquisa, o termo tecnologia agrupa os recursos não humanos e não materiais necessários ao desenvolvimento de sistemas, como por exemplo: linguagens de programação, metodologias de desenvolvimento, bancos de dados, frameworks, ferramentas, entre outros.
- Dimensão Tempo: principal dimensão de todo DW, possui atributos como mês e ano que representam os níveis da hierarquia tempo para comportar operações de *drill up* e *drill down*.

4.2.3.3. Matriz de barramento para os fatos e dimensões

Com os fatos e as dimensões definidos, as perguntas foram agrupadas conforme as áreas de assunto apresentadas na Seção 4.1.4 (Áreas de assunto para gestão de projetos em desenvolvimento distribuído de software): Custo, Recurso, Escopo, Progresso, Qualidade e Risco. Essa classificação foi realizada com base nos fatos e medidas, associando cada fato a uma única área de assunto. O resultado é a matriz de barramento apresentada no Quadro 4.

As marcações em negrito, na matriz de barramento, apontam as dimensões “essenciais” cujo fato não poderia existir sem o relacionamento com essas dimensões. Pelo próprio significado de DW, a dimensão Tempo é necessária a todos os fatos. As demais marcações não-negritas indicam as dimensões que permitem um maior número de cruzamentos e filtros. Como a arquitetura incremental inicia com um único DM prevendo sua futura integração, é possível iniciar por uma das áreas de assunto, implementando as dimensões essenciais e uma ou outra que julgar mais conveniente, de acordo com as necessidades do gerente de projetos. À medida que novos DMs vão sendo integrados, as novas dimensões vão sendo relacionadas aos fatos já existentes, ampliando o número de questões que podem ser respondidas pelo DW.

O Quadro 5 é um complemento da matriz de barramento, definindo o tipo de cada tabela de fato e a granularidade correspondente. Como pode ser observado neste quadro, a maioria das tabelas de fatos aqui definidas é do tipo instantâneo periódico, cuja atualização ocorre a intervalos regulares. As regras para carga do DW após a primeira carga devem ser armazenadas no Repositório de dados e realizadas por meio da gestão de cargas periódicas, na primeira camada da arquitetura.

Quadro 5 – Definição do Tipo e da Granularidade das Tabelas de fatos

Área de assunto	Tabela de fatos	Tipo	Granularidade
Progresso	Tempo e Quantidade de Atividade	Instantâneo periódico	1 linha para cada atividade, por dia
Custo	Custo da Atividade	Instantâneo periódico	1 linha para cada atividade, por dia
Qualidade	Manutenção de requisito	Transação	1 linha para cada atividade de manutenção, para cada requisito, por dia
Recurso	Conhecimento e treinamento da equipe em tecnologia	Instantâneo periódico	1 linha para cada equipe, para cada tecnologia, por dia
	Conhecimento e treinamento do recurso humano em tecnologia	Instantâneo periódico	1 linha para cada recurso humano, para cada tecnologia, por dia
	Atribuição de atividade	Transação	1 linha para cada atividade, para cada recurso humano, por dia
	Experiência	Instantâneo periódico	1 linha para cada recurso humano, por dia

Os fatos Manutenção de Requisito e Atribuição de Atividade são do tipo transação, pois mantém um ciclo de vida de período finito, porém indeterminado. Nesses casos, as regras para as cargas que são realizadas após o carregamento inicial devem ser definidas por meio da gestão de detecção de mudanças, na Camada de Inicialização e Atualização.

A junção das áreas de assunto – Recurso, Progresso, Custo e Qualidade, formam a matriz de barramento apresentada no Quadro 6, contendo as doze dimensões. Existe um grande compartilhamento das dimensões entre as áreas de assunto que, no contexto da arquitetura de DW proposta, representam os DMs. Porém, isso não impede a escolha de uma área de assunto para iniciar a construção de um DM, apenas revela que a integração de um novo DM ao DW deve levar em conta não somente os fatos relacionados àquela área de assunto, mas o aumento de perspectivas nos cubos já existentes.

Quadro 6 – Matriz de barramento por Área de Assunto para DW de GP-DDS

Áreas de assunto	Artefato	Atividade	Atividade de manutenção	Domínio do problema	Equipe	Local	Papel	Recurso Humano	Requisito	Situação da Atividade	Tecnologia	Tempo
Progresso	X	X		X	X	X	X	X		X	X	X
Custo	X	X		X	X	X	X	X		X	X	X
Qualidade			X	X		X	X	X	X		X	X
Recurso		X		X	X	X	X				X	X

4.2.3.4. Granularidade e Hierarquias

Com base nas necessidades dos gerentes de projeto, nas considerações sobre o ambiente de negócio, nos fatos e nas dimensões encontradas, esta pesquisa aponta que, para um DW de apoio à GP-DDS, a menor granularidade da dimensão Tempo seja de um Dia. O agrupamento das informações ao longo de vinte e quatro horas não traz perda significativa de informações para a GP de software, enquanto que uma granularidade mínima superior a um dia pode prejudicar análises relacionadas ao andamento das atividades.

Tendo sido definida a granularidade da dimensão Tempo, a hierarquia para permitir operações de *drill up* e *drill down* apresentaria os níveis: Dia → Mês → Ano.

Para a dimensão Projeto – principal dimensão neste contexto, a granularidade mínima sugerida é de uma Atividade. Para satisfazer ao esquema estrela, de modo a facilitar o entendimento e aumentar a eficiência das consultas, optou-se por definir Projeto como um atributo da dimensão Atividade, da mesma forma, Fase também foi tratada como atributo dessa dimensão. Criando-se, dessa forma, uma hierarquia que permite operações de *drill up* e *drill down* com os níveis: Atividade → Fase → Projeto.

Da mesma forma, Projeto foi definido como um atributo da dimensão Atividade de Manutenção, ficando os níveis de hierarquia da seguinte forma: Atividade de manutenção → Projeto.

A atividade pode ser tratada de forma diferente por diferentes empresas, que podem optar por dividi-la em subatividades associando as dimensões de maneira diferente, como por exemplo, uma atividade associada a uma equipe e as subatividades associadas aos recursos

humanos pertencentes à equipe. As hierarquias apresentadas seguem as considerações feitas no modelo conceitual do negócio.

Inicialmente, a Equipe apresentava-se como candidata à hierarquia junto com o Recurso Humano. Porém, tendo como base as considerações sobre o termo Equipe feitas na modelagem conceitual – em que uma pessoa pode participar de várias equipes e que as equipes podem sofrer alterações durante o projeto, optou-se por mantê-las como dimensões separadas. Isso permite que as alterações nas equipes sejam refletidas no DW, uma vez que, nos fatos em que houver sentido manter uma visualização por equipe, cada linha na tabela de fatos que está relacionada a um recurso humano, também estará relacionada a uma equipe.

É indispensável realizar um levantamento sobre como esses conceitos comuns são representados na empresa específica onde será implantada a solução de DW, e assim realizar as adaptações necessárias nos modelos de referência aqui abordados.

4.2.4. Roteiro para projetar um DW para GP-DDS utilizando a arquitetura proposta

Um roteiro para projetar um DW de GP-DDS foi desenvolvido, com objetivo de facilitar a aplicação da arquitetura de DW para apoio à GP-DDS proposta nesta pesquisa. Algumas etapas foram adaptadas do roteiro proposto por Costa et al. (2000). Conforme as particularidades e necessidades de cada organização, na qual pretenda-se implantar uma solução de DW para GP, algumas etapas apresentadas aqui podem ser desnecessárias. As etapas abaixo, incluindo a ordem em que são apresentadas, são sugestões que devem ser ajustadas, além disso, algumas etapas podem ser executadas paralelamente.

Etapa 1 – Análise do negócio a ser modelado

1.1 Definir o negócio objeto de gestão:

1.1.1 Definir os objetivos da organização;

1.1.2 Definir as necessidades dos gerentes de projeto.

1.2 Modelar um MER (modelo entidade relacionamento) para descrever o modelo de dados conceitual do negócio, representando o conhecimento do negócio para tomada de decisão – com o objetivo de facilitar a posterior MD;

1.3 Definir o escopo do projeto:

- a) Determinar quais os projetos mais críticos para o negócio e que precisam de decisões sobre ele;

b) Identificar o nível de granularidade necessário para o DW.

1.4 Analisar onde estão as informações que alimentarão o DW.

Etapa 2 – Análise dos sistemas operacionais de origem e das bases de dados de origem

2.1 Analisar o ambiente computacional da empresa para facilitar o processo de MD:

a) Tipo dos dados das bases de dados de origem:

a. Bancos de dados (relacionais, em rede, hierárquicos);

b. Arquivos sequenciais;

c. Planilhas.

b) Modelos de dados das bases de dados de origem;

c) Dicionários de dados dos sistemas operacionais de origem;

d) Relatórios gerenciais dos sistemas operacionais de origem.

2.2 Definir os metadados de negócio;

2.3 Analisar a qualidade dos dados;

2.4 Verificar a necessidade e a viabilidade de alterar os sistemas operacionais de origem para obter dados faltantes para análise.

Etapa 3 – Modelagem dimensional

3.1 Definir os fatos (as métricas de controle);

3.2 Definir as dimensões (o que descreve o negócio);

3.3 Definir a granularidade da dimensão tempo;

3.4 Definir a hierarquia das dimensões de consulta;

3.5 Definir a técnica de MD (estrela ou floco de neve);

3.6 Definir uma ferramenta CASE para documentar o modelo de dados, o dicionário de dados, os domínios de atributos e para facilitar a geração de esquemas físicos do banco de dados;

3.7 Detalhar as demais granularidades;

3.8 Definir as agregações.

Etapa 4 – Definição dos aspectos da implementação física

4.1 Analisar o volume de dados:

a) Levantar o volume de dados;

b) Prever o crescimento do volume de dados;

c) Verificar a necessidade de sumarizar os dados (agregações);

4.2 Definir as regras para indicação do tempo na carga inicial;

4.3 Definir as regras para detecção de mudanças;

- 4.4 Definir as regras de periodicidade de carga;
- 4.5 Definir as regras de transformação dos dados;
- 4.6 Definir o tempo de armazenagem dos dados;
- 4.7 Definir o tipo do banco de dados: relacional (ROLAP) ou multidimensional (MOLAP), com base na análise do volume de dados;
- 4.8 Definir os índices, caso a escolha do tipo de banco de dados tenha sido ROLAP;
- 4.9 Definir esquema para realização de *backup*;
- 4.10 Definir esquema de controle de acesso.

4.3. Considerações finais

Neste Capítulo foi apresentada a proposta de arquitetura de DW para apoiar a GP-DDS. Inicialmente apresentou-se o levantamento de requisitos com base nas necessidades dos gerentes de projeto, em seguida, a partir das necessidades apresentadas, uma arquitetura de DW foi proposta, contendo a descrição dos seus componentes e destacando a modelagem dimensional para a arquitetura.

A arquitetura de DW foi definida com o objetivo de atender as necessidades da área de GP-DDS: adaptando-se a infraestruturas diferentes; mantendo o foco no gerente de nível tático, sem impedir que outros níveis gerenciais sejam atendidos no futuro com a integração de novos assuntos; possibilitando que o retorno do investimento inicie com a implementação dos primeiros fatos e dimensões; e permitindo uma visão corporativa dos recursos fisicamente distribuídos.

No próximo Capítulo é descrita a aplicação da arquitetura de DW na construção de um DW sobre um ambiente de DDS denominado DiSEN. A aplicação tem como objetivo avaliar a arquitetura que foi proposta neste Capítulo.

Construção do DW sobre o DiSEN utilizando a Arquitetura Proposta

Este Capítulo é composto por três partes. A primeira parte trata da fundamentação teórica sobre o ambiente DiSEN, descrevendo os trabalhos relacionados ao ambiente, sua arquitetura em camadas e o relacionamento desta arquitetura com a arquitetura de DW proposta nesta pesquisa.

A aplicação da arquitetura proposta sobre o ambiente DiSEN é tratada na segunda parte deste Capítulo, abrangendo uma adaptação do modelo de dados conceitual do DiSEN.

A terceira parte apresenta detalhes sobre a construção do DW, demonstrando alguns resultados obtidos sobre o DM inicial.

5.1. O Ambiente DiSEN

O DiSEN (*Distributed Software Engineering Environment* – Ambiente de desenvolvimento distribuído de software) é um ambiente para auxiliar o processo de DDS. Este ambiente está em desenvolvimento pelo Grupo de Estudos em Engenharia de Software Distribuído (GEESD) da Universidade Estadual de Maringá (UEM) (Huzita et al., 2007). O DiSEN vem recebendo importantes contribuições ao longo dos últimos anos por meio de módulos que tratam de partes específicas do ambiente e que vão sendo integrados a ele.

Com relação à GP, algumas ferramentas já foram desenvolvidas pelo GEESD para o DiSEN, tais como: DIMANAGER (Pedras et al., 2004) para atender às necessidades do gerente de projetos, como alocação de atividades e recursos, VIMEE (Trindade, Tait, Huzita, 2008), para dar suporte à comunicação síncrona, explícita e formal e CostDDS (Pagno, 2010) para estimar custos em projetos.

Diversos trabalhos desenvolvidos pelo GEESD estão relacionados à GP-DDS. São eles: Uma ferramenta de estimativa de custos para o desenvolvimento distribuído de software (Pagno, 2010); Um modelo de planejamento estratégico de sistemas de informação para organizações que atuam em desenvolvimento distribuído de software (Cibotto, 2009); Uma ferramenta para gerenciar a comunicação em um ambiente distribuído de desenvolvimento de software (Trindade, Tait, Huzita, 2008); Uma estratégia para o gerenciamento de riscos em desenvolvimento distribuído de software (Leme, Tait, Huzita, 2007); Um modelo de gerenciamento de projetos para um ambiente de desenvolvimento distribuído de software (Enami, Tait, Huzita, 2006); Mecanismo de apoio ao gerenciamento de RH no contexto de um ambiente distribuído de software (Lima, 2004); e Uma ferramenta de apoio ao gerenciamento de desenvolvimento de software distribuído (Pedras et al., 2004).

Estes trabalhos produziram informações gerenciais que podem ser armazenadas no ambiente DiSEN e que foram utilizadas como fonte de dados para a aplicação da arquitetura de DW em GP-DDS proposta por esta pesquisa.

5.1.1. Arquitetura do DiSEN

A arquitetura do DiSEN é composta por três camadas: dinâmica, aplicação e infraestrutura. A camada dinâmica permite a inserção, configuração e remoção dos componentes de software e serviços em tempo de execução. A camada de aplicação oferece: suporte às metodologias de

desenvolvimento de software, repositório para armazenamento dos artefatos e informações necessárias ao ambiente e gerenciadores de *workspace*, objetos e agentes. Já a camada de infraestrutura provê suporte às tarefas de nomeação, persistência e concorrência, e incorpora o canal de comunicação (Huzita et al., 2005).

Conforme apresentado na Figura 8, é na camada de aplicação que fica o Repositório onde serão armazenadas as informações do ambiente. Também nesta camada está presente o Gerenciador de Objetos que é constituído pelo gerenciador de acesso, gerenciador de atividades, gerenciador de recursos, gerenciador de artefatos, gerenciador de projetos, gerenciador de processos e gerenciador de versões e configurações (Pascutti, 2002).

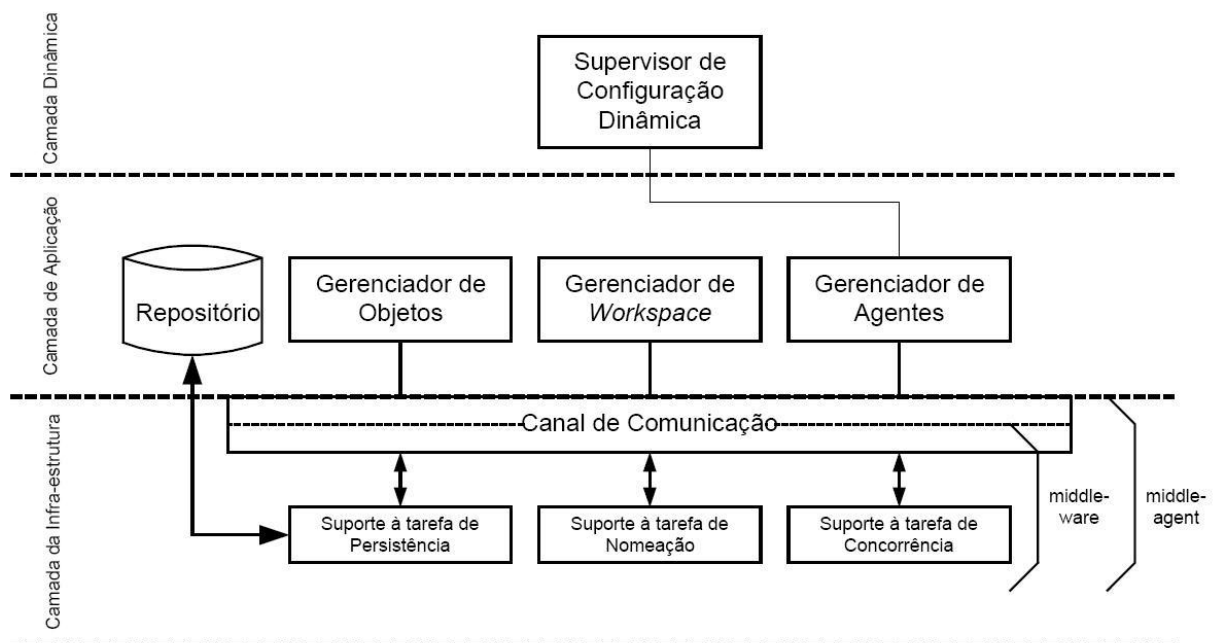


Figura 8 – Arquitetura do DiSEN

Fonte: Pascutti (2002)

5.2. Aplicação da Arquitetura Proposta sobre o DiSEN

Como forma de avaliar a aplicabilidade do resultado desta pesquisa, a arquitetura de DW para apoio à GP-DDS foi aplicada em um DW que utilizou como fonte de dados o ambiente de desenvolvimento distribuído DiSEN.

Seguindo o roteiro sugerido na Seção 4.2.4 (Roteiro para projetar um DW para GP-DDS utilizando a arquitetura proposta) verificou-se que algumas etapas não poderiam ser realizadas já que não há uma organização utilizando o DiSEN em produção. Dessa forma, algumas etapas foram desconsideradas e outras etapas foram adaptadas ao ambiente, conforme apresentado a seguir.

Etapa 1 – Análise do negócio a ser modelado

1.1 Definir o negócio objeto de gestão:

1.1.1 Definir os objetivos da organização: neste item, foram considerados os objetivos já apresentados na definição da arquitetura;

1.1.2 Definir as necessidades dos gerentes de projeto: neste item foram consideradas as necessidades já apresentadas na definição da arquitetura.

1.2 Modelar um MER para descrever o modelo de dados conceitual do negócio: com base nos trabalhos desenvolvidos pelo GEESD relacionados à GP-DDS, um diagrama contendo o modelo de dados conceitual do DiSEN foi desenhado;

1.3 Definir o escopo do projeto:

a) Determinar quais os projetos mais críticos para o negócio e que precisam de decisões sobre ele: uma área de assunto foi escolhida para construção do primeiro DM;

b) Identificar o nível de granularidade necessário para o DW: os níveis de granularidade para o ambiente DiSEN foram definidos.

1.4 Analisar onde estão as informações que alimentarão o DW: a localização física dos dados do ambiente DiSEN foi analisada.

Etapa 2 – Análise dos sistemas operacionais de origem e das bases de dados de origem

2.1 Analisar o ambiente computacional da empresa para facilitar o processo de MD: foi utilizada uma ferramenta de engenharia reversa sobre o banco de dados para obtenção do modelo de dados atualizado, além disso, foram estudados os trabalhos do GEESD sobre o DiSEN para melhor entendimento do significado dos dados que podem ser armazenados pelo DiSEN;

2.2 Definir os metadados de negócio: os dados do ambiente DiSEN relacionados à GP-DDS, seus significados e possíveis valores são apresentados no Apêndice C;

2.3 Analisar a qualidade dos dados: item não verificado nesta aplicação, por não haver dados armazenados no banco de dados;

2.4 Verificar a necessidade e viabilidade de alterar os sistemas operacionais de origem para obter dados faltantes para análise: foram recomendadas modificações no ambiente DiSEN para aumentar a capacidade de respostas do DW.

Etapa 3 – Modelagem dimensional: os itens desta etapa de modelagem referentes ao escopo definido para a aplicação foram adaptados ao ambiente DiSEN;

Etapa 4 – Definição dos aspectos da implementação física: apenas as regras para indicação do tempo na carga inicial, as regras de transformação dos dados e o tipo de banco de dados foram definidos para permitir a implementação física. Não foi possível verificar os demais itens desta etapa por não haver dados armazenados no banco de dados que permitissem análise de volume e frequência de atualização.

5.2.1. Modelo de dados conceitual do DiSEN

O modelo de dados de alto nível referente à GP no ambiente DiSEN, representado pela Figura 9, foi construído com base nos trabalhos desenvolvidos pelo GEESD. Neste modelo, verifica-se que o projeto é dividido em fases que por sua vez são divididas em atividades que são compostas por tarefas. O projeto tem relação direta com um processo de desenvolvimento de software que é usado como base para a criação das fases e atividades, não impedindo, porém, que existam fases e atividades associadas ao projeto e que não foram herdadas do processo de desenvolvimento. Riscos que afetem o projeto podem ser identificados por um local específico.

Cada atividade pode estar relacionada a vários artefatos, que são gerados pelas tarefas. Uma tarefa é executada por um recurso humano que desempenha um papel. Além do recurso humano, a tarefa também necessita de recursos materiais para ser executada. Uma tarefa pode apresentar problemas.

Os recursos, tanto humanos como materiais, estão fisicamente localizados em um local. Os recursos humanos podem possuir relação de afinidade com outros recursos humanos. Eles podem possuir aptidões e habilidades e participar de reuniões e treinamentos.

Existe relação de dependência entre as atividades e as fases, tanto no projeto, quanto no processo de desenvolvimento escolhido como base para o projeto.

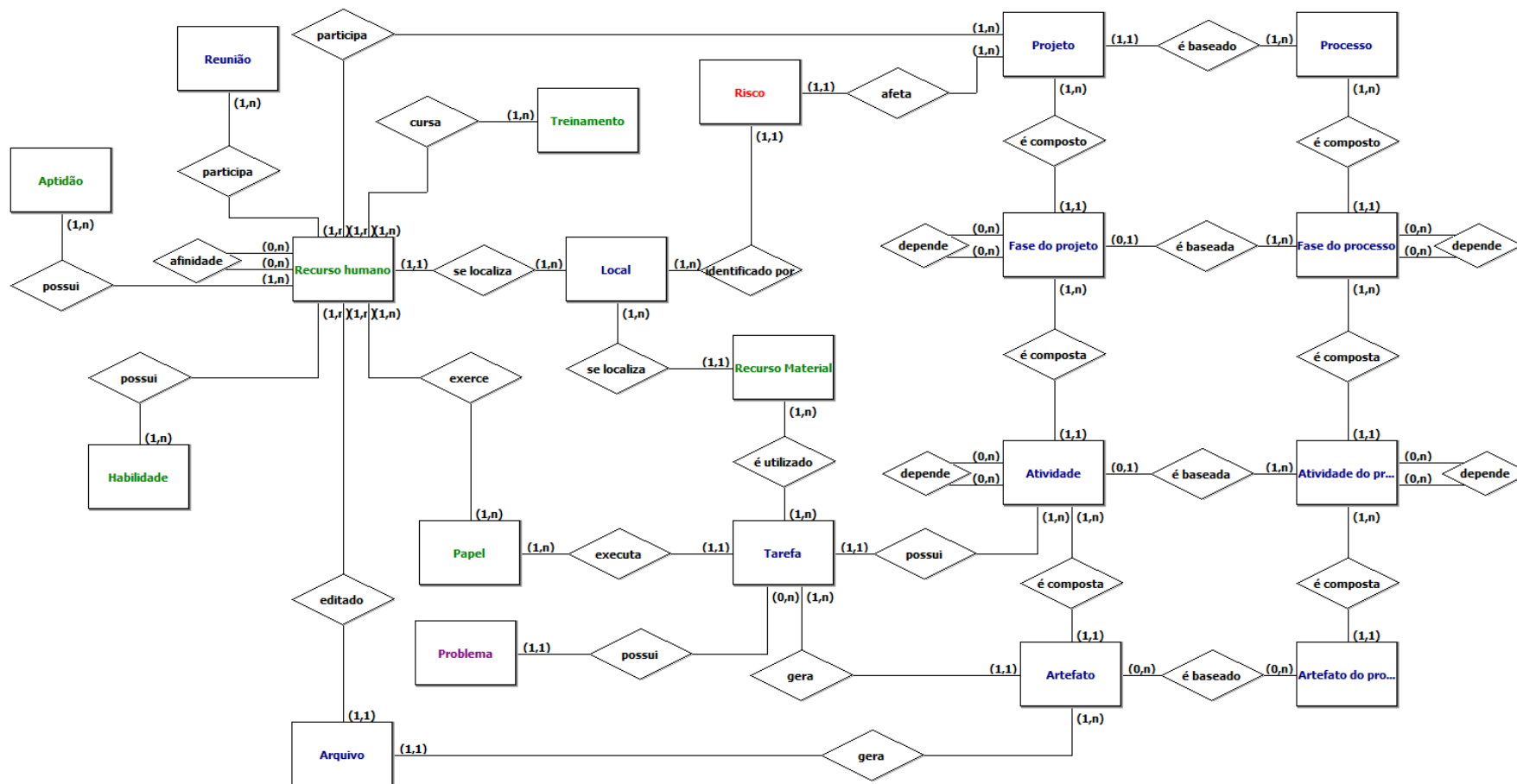


Figura 9 – Modelo de dados conceitual em alto nível da GP no ambiente DiSEN

5.2.2. Definição do escopo para o projeto inicial do *data warehouse*

Para a construção do DW baseado na arquitetura proposta nesta pesquisa, foi necessário escolher uma área de assunto para a criação do primeiro DM. A área de assunto escolhida foi Recurso, e o projeto inicial trata do treinamento dos recursos humanos. As escolhas devem-se à existência de informações sobre este tema no banco de dados do ambiente DiSEN e, também, ao grau de importância que este assunto tem para a tomada de decisão de um gerente de nível tático no planejamento e acompanhamento de um projeto de DDS.

5.2.3. Granularidade

A partir da granularidade sugerida na arquitetura de DW proposta e após análise do modelo conceitual de negócio do ambiente DiSEN, optou-se por manter a granularidade da Dimensão Tempo e alterar a granularidade referente ao Projeto. No ambiente DiSEN, cada atividade é composta por tarefas, portanto a Tarefa é o menor nível de granularidade com relação ao projeto. Dessa forma, a hierarquia da tarefa ficou definida da seguinte maneira: Projeto → Fase → Atividade → Tarefa.

5.2.4. Fonte de dados

Para a primeira versão do DW foram consideradas apenas as informações possíveis de serem armazenadas no repositório de dados do ambiente DiSEN. Dessa forma, o esforço de integração será mínimo, pois os dados estão em um mesmo formato (tabelas relacionais) e centralizados em um único local (banco de dados *Postgre*). No entanto, a arquitetura prevê a integração de novas fontes de dados, ampliando o escopo do DW.

5.2.5. Análise dos dados do ambiente DiSEN

Apesar de não haver diagramas completos e atualizados do repositório de dados do ambiente DiSEN, foi possível realizar engenharia reserva utilizando a ferramenta *Power Architect*, facilitando o trabalho de análise do banco de dados *Postgre* utilizado pelo ambiente DiSEN.

A complexidade e a maior demanda de tempo ficaram por conta da análise do significado dos dados.

Para levantar as informações úteis à tomada de decisão dos gerentes de nível tático e para compreender o significado dos campos nas tabelas do banco de dados utilizado pelo DiSEN, foram estudados: o próprio banco de dados e o código fonte do ambiente DiSEN, os trabalhos do GEESD que abordam temas relacionados à GP (Pagno, 2010; Cibotto, 2009; Trindade, Tait, Huzita, 2008; Leme, Tait, Huzita, 2007; Enami, Tait, Huzita, 2006; Lima, 2004; Pedras et al., 2004), além de outros trabalhos do GEESD cujos temas não estão relacionados diretamente à GP mas que auxiliaram na compreensão do significado dos dados, como: DiSEN-SCV: Gerenciador de versões de artefatos para um ambiente de desenvolvimento distribuído de software (Silva, 2008). Os dados relacionados à tomada de decisão na GP-DDS que foram encontrados no banco de dados, seus significados e possíveis valores são listadas no Apêndice C, agrupados por assunto.

5.2.6. Recomendações para o ambiente DiSEN

Algumas sugestões de alterações no ambiente DiSEN para armazenamento de dados, que podem aumentar a quantidade de informações a serem obtidas do DW para apoiar a GP-DDS, são apresentadas a seguir:

- a) Incluir relacionamentos entre a Reunião e a Fase do Projeto, entre a Reunião e o Projeto e quando oportuno, entre a Reunião e as Atividades; permitindo que as reuniões sejam analisadas sob a perspectiva da atividade, da fase ou do projeto;
- b) Armazenar um atributo que indique a forma de acesso ao Recurso Material, se local ou global. Caso seja possível acessar o recurso globalmente, manter o relacionamento do Recurso Material com o Local – para indicar onde encontra-se o recurso fisicamente, e armazenar atributos que indiquem o meio de acesso e os requisitos para acesso global. Exemplos de recursos globais são os Data Centers e Servidores Web, que podem estar localizados fisicamente em determinado endereço, mas que podem ser acessados via rede de outros locais;
- c) Armazenar atributo que indique a carga horária do Treinamento realizado pelos recursos humanos;
- d) Incluir Recurso Tecnológico relacionando com Treinamento, Atividade e/ou Tarefa, gerando uma nova dimensão no modelo dimensional para ser utilizada no

agrupamento por tecnologia, como sugerido na arquitetura (linguagem de programação, banco de dados, arquitetura, framework, entre outros).

5.2.7. Modelagem dimensional para o ambiente DiSEN

Considerando o assunto escolhido – treinamento de recursos humanos, um esquema estrela foi desenvolvido utilizando como referência o esquema apresentado na arquitetura de DW para esse tema (ver Figura 15).

Foram realizadas algumas adequações na modelagem dimensional proposta na arquitetura de DW sobre o fato “Conhecimento e treinamento do recurso humano em tecnologia” e nas dimensões relacionadas a este fato, de modo a satisfazer o modelo conceitual do DiSEN apresentado na Figura 9. O modelo dimensional para análise do treinamento dos recursos humanos adaptado ao DiSEN é apresentado na Figura 10.

As principais adequações realizadas devem-se aos seguintes fatos: (a) o treinamento não está relacionado diretamente a uma lista de tecnologias, a tecnologia ou assunto relacionado ao treinamento deve ser obtido a partir da interpretação dos atributos nome ou descrição do treinamento que são do tipo texto; (b) as medidas disponíveis no sistema operacional do ambiente DiSEN relacionadas à treinamento são: nível de aproveitamento do treinamento e tempo de treinamento em dias.

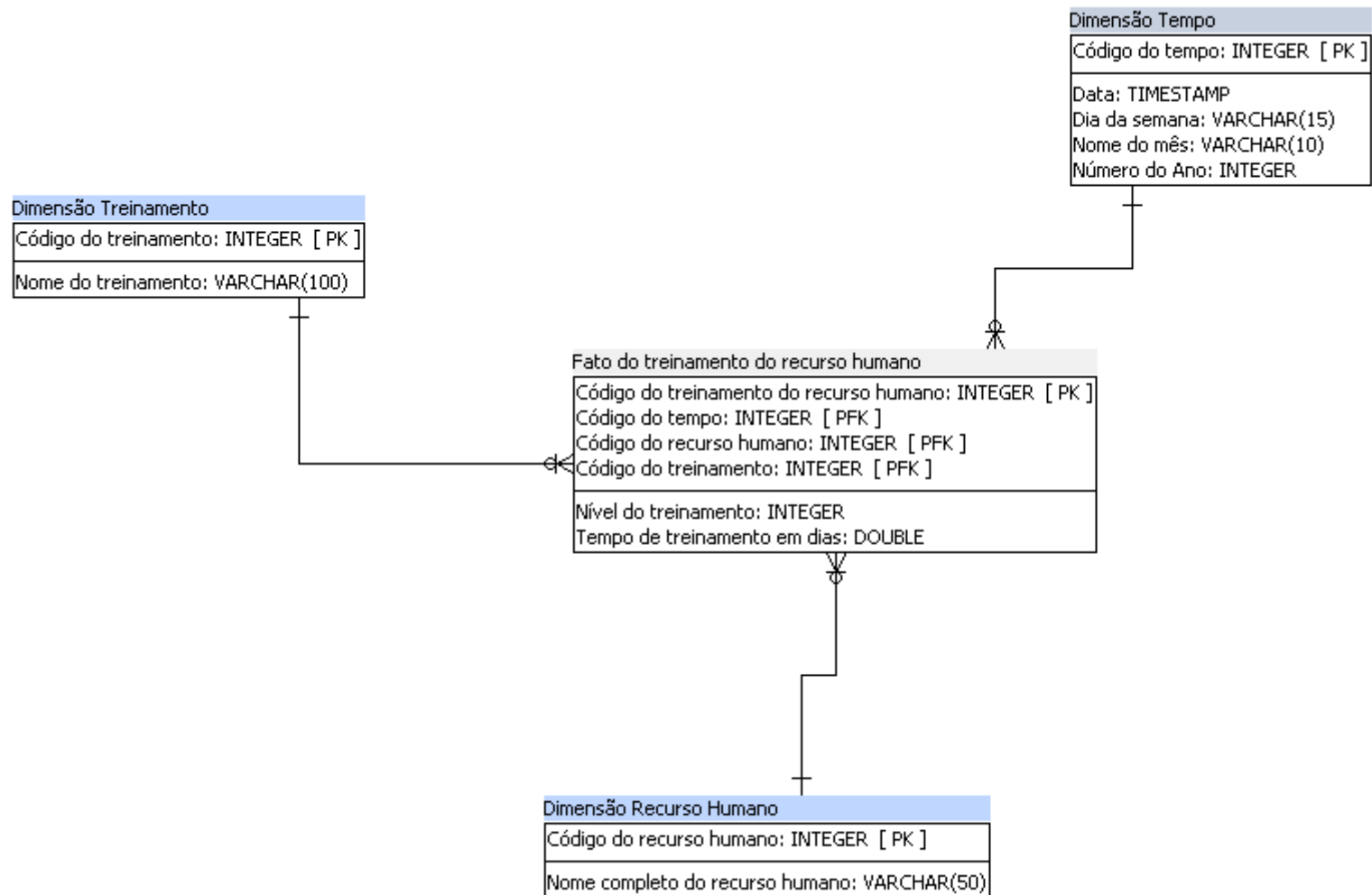


Figura 10 – DiSEN – Esquema Estrela para o fato Treinamento do Recurso Humano

5.2.8. Aspectos da implementação física para o DW sobre o ambiente DiSEN

Com relação ao carregamento do DW a partir dos dados do ambiente DiSEN, são necessárias algumas transformações nas medidas relacionadas à treinamento: (a) o nível de aproveitamento do treinamento é um atributo do tipo texto, que deve passar por um processo de transformação para substituir os possíveis valores, apresentados no Quadro 21 (pouco, razoável, bom e ótimo), por valores numéricos (1, 2, 3, 4); (b) o tempo de treinamento deve ser obtido pelo cálculo da diferença entre as datas de fim e de início disponíveis no banco de dados de origem.

Foram definidas três dimensões para os fatos relacionados a treinamento: Treinamento, Tempo e Recurso Humano. A dimensão Tempo permite as três hierarquias sugeridas na arquitetura: dia, mês e ano. Sendo que o fato só é adicionado quando o treinamento é finalizado, ou seja, a data de fim apresentada no Quadro 12 é utilizada para preencher o atributo data da dimensão Tempo. Nas operações de *drill up*, deve ser calculada a média dos níveis de treinamento, enquanto que para a medida de tempo de treinamento deve ser realizada a operação de soma.

Não sendo possível analisar o volume de dados, optou-se por armazenar os dados do DW em um banco de dados relacional. Assim, o processamento das consultas é do tipo ROLAP. Para essa aplicação da arquitetura de DW sobre o ambiente DiSEN, o *Postgres* foi escolhido como sistema gerenciador de banco de dados para armazenar as tabelas de fatos e as tabelas de dimensão do DW.

5.3. Construção do DW sobre o DiSEN

A primeira versão do DW foi construída com o apoio da ferramenta de modelagem Power Architect, da suíte Pentaho *Open Source Business Intelligence* – uma plataforma de código aberto composta por diversas ferramentas para BI e do sistema gerenciador de banco de dados *Postgres*.

Alguns dados foram inseridos para permitir simular uma análise dos dados armazenados no DW. A seguir, são apresentados os resultados de duas análises utilizando o Pentaho, que foram realizadas sobre o cubo criado a partir da tabela Fato do treinamento do recurso humano, contendo as dimensões Tempo, Recurso Humano e Treinamento.

Na Figura 11, a quantidade de pessoas/dia gasta com treinamento é analisada sob a perspectiva tempo após uma operação de *drill up* – subindo do nível dia para o nível mês na hierarquia da dimensão tempo. Esta análise considerou todos os recursos humanos, sem distingui-los, o que resultou na soma da quantidade de dias utilizada em treinamento por todos os recursos em cada treinamento e em cada mês. É possível verificar os treinamentos que foram ministrados em diferentes meses, quais treinamentos foram ofertados em quais meses, comparar a quantidade de pessoas/dias utilizada por treinamento e verificar a menor e maior quantidade de pessoas/dias por treinamento.

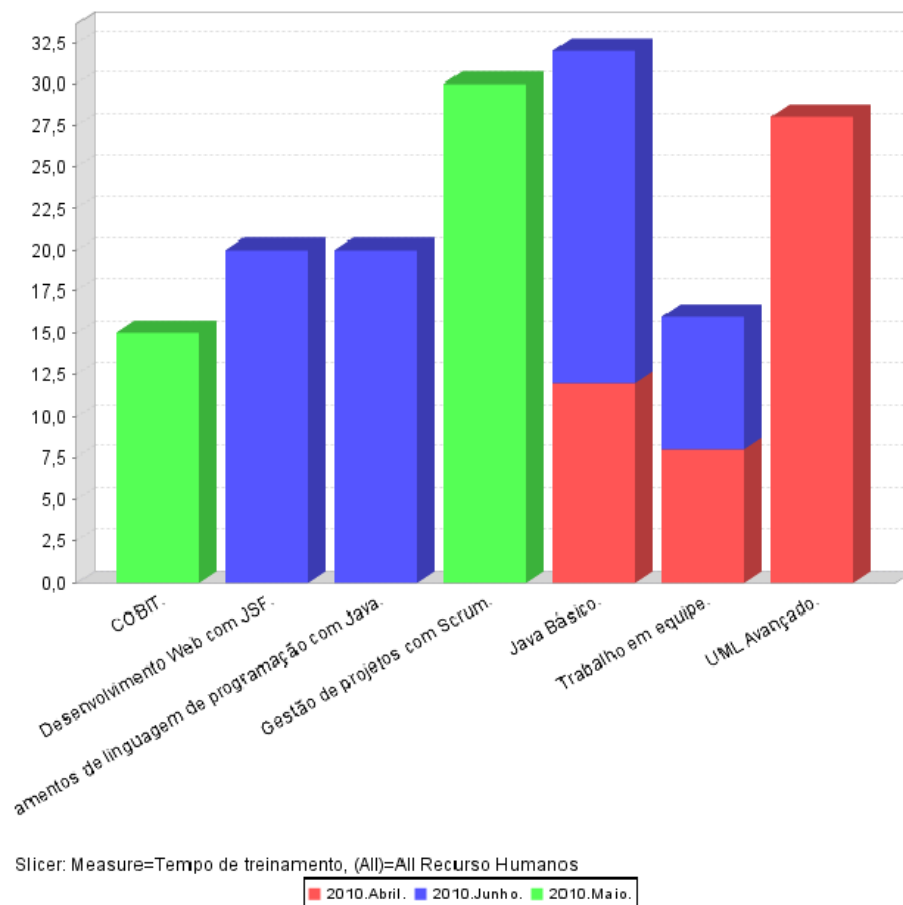


Figura 11 – Análise do fato Treinamento: Tempo de treinamento por Mês por Treinamento

O nível de aproveitamento pode variar de um a quatro, conforme definido na modelagem dimensional para o DiSEN. Esses valores representam respectivamente os níveis: pouco, razoável, bom e ótimo. A Figura 12 permite comparar esses níveis de aproveitamento entre os treinamentos, sem distinguir o período em que esses treinamentos ocorreram e sem

distinguir os recursos humanos, de forma que o nível foi obtido pela média dos níveis de aproveitamento de todos os recursos humanos que participaram daquele treinamento.

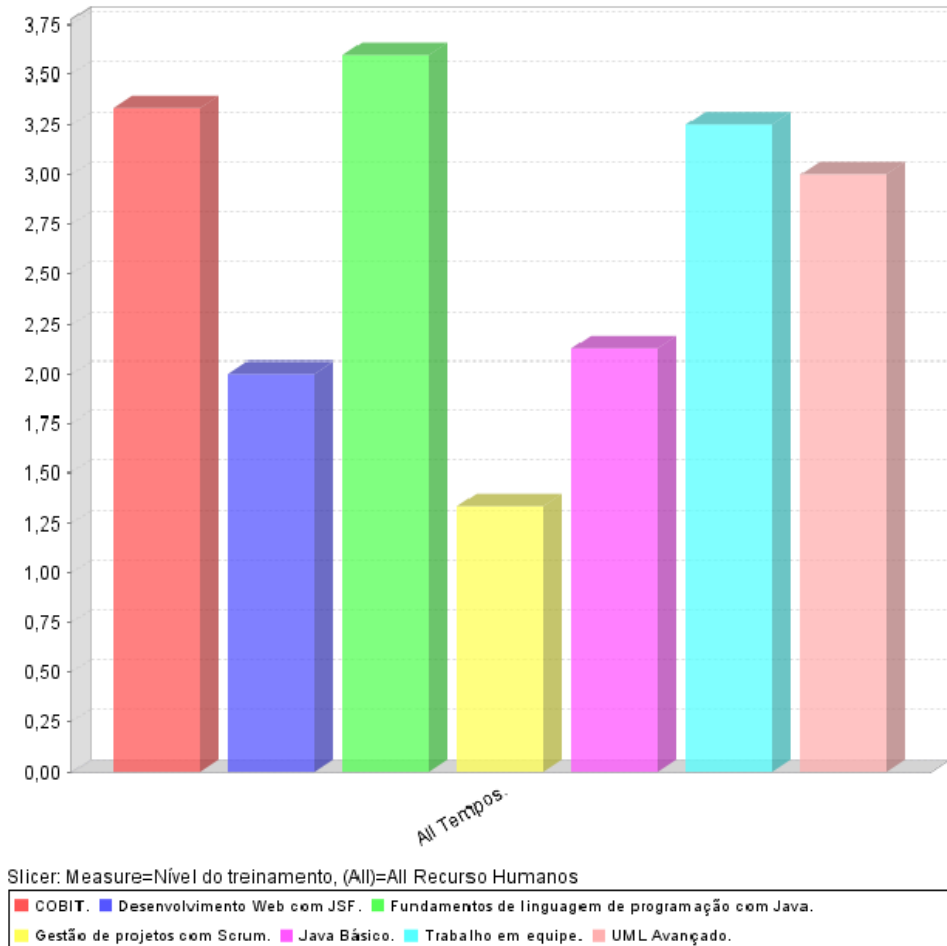


Figura 12 – Análise do fato Treinamento: Nível de aproveitamento por Treinamento

Muitas outras análises podem ser feitas sobre este mesmo cubo, variando as medidas e as dimensões, realizando operações de *drill up* e *drill down*, filtrando dados ou girando o cubo. Portanto, é possível concluir que após a criação do primeiro DM, com vistas a atender um determinado número de requisitos de informações, muitas outras informações são conseguidas com o mesmo DM. E, após o início da utilização pelos gerentes de projeto, com a análise dos primeiros resultados obtidos com os fatos e dimensões existentes, novos requisitos surgirão, podendo envolver o mesmo DM, um novo DM ou o cruzamento entre diferentes DMs.

5.4. Considerações finais

Neste Capítulo foi apresentada a construção de um DW de apoio à GP-DDS sobre o ambiente de desenvolvimento distribuído DiSEN, que utilizou como base a arquitetura de DW descrita no Capítulo 4. Esta aplicação teve como objetivo avaliar a arquitetura de DW para apoio à GP-DDS proposta nesta pesquisa.

Primeiramente, foi realizado um estudo sobre o ambiente DiSEN que seria utilizado na aplicação. Em seguida, utilizando como guia a arquitetura de DW proposta, foi realizada a adaptação do modelo de dados conceitual, a escolha de uma área de assunto para iniciar o projeto, a análise dos dados armazenados no repositório de dados do DiSEN, a adequação do modelo de dados dimensional proposto na arquitetura e, por fim, a criação da primeira versão do DW para o ambiente DiSEN, com o auxílio de ferramentas gratuitas.

A arquitetura de DW proposta nesta pesquisa guiou a construção do DW e permitiu que algumas análises sobre os dados pudessem ser realizadas no início do primeiro DM, com um número reduzido de dimensões. Além disso, tão logo foram definidos os primeiros fatos, foi possível identificar possíveis alterações no ambiente DiSEN para melhorar a capacidade de resposta do DW.

No próximo Capítulo são apresentadas as conclusões desta pesquisa, descrevendo as dificuldades encontradas, as contribuições e as indicações de trabalhos para dar continuidade a este trabalho.

Conclusões

Este trabalho apresentou a definição de uma arquitetura de DW com a finalidade de coletar dados históricos de projetos que sejam úteis para o processo de tomada de decisão do gerente de projetos durante o planejamento e acompanhamento de projetos de software. A proposta focou na GP de softwares que são desenvolvidos de maneira distribuída, levando em consideração informações que são desconsideradas ou possuem pouca importância no contexto do desenvolvimento centralizado, como é o caso da localização dos recursos.

A validação da arquitetura foi realizada por meio da implementação de uma primeira versão do DW sobre o ambiente DiSEN, cujo resultado demonstrou ser possível iniciar a construção a partir de um DM inicial. A definição dos requisitos e do escopo e a modelagem dimensional foram realizadas sobre os requisitos e modelagens sugeridos na arquitetura, mesmo sendo necessárias algumas adaptações. O tempo foi bastante reduzido se comparado à definição de requisitos e modelagem a partir da literatura e ferramentas de GP. A aplicação evidenciou ainda que, após a montagem do cubo, é possível obter muitas outras informações além das inicialmente previstas. Também com a aplicação foi possível verificar a existência de ferramentas livres, disponíveis gratuitamente, que permitem desde a modelagem das tabelas de fatos e dimensão, criação automática de *scripts* para o banco de dados do DW, modelagem do cubo, ETL, até a análise dos dados pelo usuário final.

6.1. Dificuldades Encontradas

Uma das limitações relaciona-se à análise do significado dos dados, a qual está em formatação nos vários trabalhos que compõem o ambiente DiSEN. Neste sentido, a presente pesquisa pode contribuir para fornecer uma documentação atualizada sobre o ambiente de desenvolvimento distribuído DiSEN.

Outra dificuldade refere-se aos dados armazenados no DiSEN que, por tratar-se de um ambiente construído por diversos pesquisadores e ainda estar em desenvolvimento, não foi povoado. Dessa forma, a aplicação foi construída com base na estrutura do banco de dados utilizado como repositório dos dados pelo ambiente DiSEN e alguns dados foram inseridos para permitir simular a análise dos dados do DW.

6.2. Contribuições

A principal contribuição desta pesquisa é descrição de uma arquitetura de DW que pode ser utilizada como guia para a construção de DW na área de gestão de projetos de software. Outras contribuições desta pesquisa estão relacionadas ao aprimoramento das áreas de DDS e da área de DW, referentes ao item GP. Inclui ainda contribuições para o ambiente de desenvolvimento distribuído DiSEN, com a catalogação dos dados, relacionados à GP, que estão armazenados no repositório do ambiente e sugestões de alterações no modelo de dados. Mais uma contribuição é a reunião de material organizado sobre uma arquitetura de DW aplicada à GP em ambientes de DDS.

Esta pesquisa também contribui para a identificação de informações sobre o projeto que devem ser armazenadas durante as fases de planejamento e acompanhamento, de forma a permitir seu uso histórico no planejamento e estimativas de futuros projetos e acompanhamento de projetos em andamento. O levantamento dos requisitos e dados necessários à tomada de decisão de um gerente de projetos DDS, incluindo as sugestões de modelagem dimensional para a área de GP-DDS, são contribuições importantes para o desenvolvimento de novos trabalhos de DW nesta área.

6.3. Trabalhos Futuros

Como pesquisas futuras relacionadas à DW com foco na GP-DDS indica-se:

- Ampliação do escopo do DW, armazenando informações estratégicas para auxiliar os gerentes de alto nível da organização.
- Estudo para inclusão das demais perspectivas de Magalhães e Pinheiros (2007) como áreas de assuntos de DMs: Escopo, Riscos e Qualidade.
- Estudo de planos de contingência para resolução de problemas com base em problemas e soluções já utilizadas, com vistas a incluir as informações no DW e auxiliar a tomada de decisões nessa área.
- Estudo da viabilidade de evolução do DW para um RDW de forma a abranger a tomada de decisões operacionais.

Como pesquisas futuras relacionadas ao DW de apoio à GP-DDS para o ambiente DiSEN aponta-se:

- Aumento da abrangência da primeira versão do DW para o DiSEN, com o desenvolvimento de outras áreas de assunto e inclusão de novos DMs de forma incremental.
- Nova análise do banco de dados utilizado pelo DiSEN, verificando a importância dos novos dados, incluídos a partir do ano de 2010, para o apoio à GP-DDS.
- Alterações no ambiente operacional do DiSEN de forma a armazenar informações adicionais que são úteis à GP-DDS, mas que ainda não estão presentes no banco de dados do ambiente, prevendo posterior inclusão no DW.
- Analisar incorporação de novas fontes de dados externas ao banco de dados do DiSEN, como modelos de referência e/ou pesquisas de mercado para análise e comparação.

Referências Bibliográficas

- ARIYACHANDRA, T.; WATSON, H. J. Which Data warehouse Architecture is Best? In: Communications of the ACM, v. 51, n. 10, p. 146-147, out. 2008.
- AUDY, J. L. N. Modelo de planejamento estratégico de sistemas de informação: contribuições da aprendizagem organizacional e do processo decisório. Porto Alegre, 2001. Tese (Doutorado em Administração) – Escola de Administração. Universidade Federal do Rio Grande do Sul.
- BENNATAN, E. M. What is happening to the global software village? In: Agile Project Management Advisory Service. Cutter Consortium: v. 3, n. 1, jan. 2002.
- CÂNDIDO, C. H. brModelo 2.0, jun., 2007. Disponível em: <http://www.sis4.com/brModelo/download.aspx>. Acessado em: 29/05/2010.
- CARMEL, E.; ABBOTT, P. Why “nearshore” means that distance matters. Communications of the ACM, v. 50, n. 10, p. 40-46, out. 2007.
- CLEMES, M. Data warehouse como suporte ao sistema de informações gerenciais em uma Instituição de Ensino Superior: estudo de caso na UFSC. Florianópolis, 2001. Dissertação (Mestrado em Engenharia de Produção) – Universidade Federal de Santa Catarina.
- COREY, M.; ABBEY, M.; ABRAMSON, I.; TAUB, B. Oracle 8i Data warehouse. Rio de Janeiro: Campus, 2001.
- COSTA, C. M.; LIMA, F. A. P.; FIGUEIREDO, F. A.; MIORELLI, H. S.; SOUZA, M. V. S. Proposta de um Roteiro para Projetar um Data warehouse. Curitiba, 2000. Monografia (Especialização em Tecnologia da Informação e Comunicação) – Faculdade Católica de Administração e Economia.
- DAMIAN, D.; MOITRA, D. Global Software Development: How far have we come? IEEE Software, p. 17-19, set. e out. 2006.
- DATE, C. J. Introdução a sistemas de bancos de dados. 8. ed. Rio de Janeiro: Elsevier, 2003.
- DIAS, M. M.; TAIT, T. F. C.; MENOLLI, A. L. A.; PACHECO, R. C. Data warehouse Architecture through Viewpoint of Information System Architecture. In: International Conference on Computational Intelligence for Modelling Control & Automation. Viena, Áustria, 2008.
- ENAMI, L. N. M.; TAIT, T. F. C.; HUZITA, E. H. M. A Project Management Model to a Distributed Software Engineering Environment. In: International Conference on Enterprise Information Systems. Pafos, Chipre, 2006.

- HADLEY, L. Developing a Data warehouse Architecture. 2002. Disponível em: <http://www.users.qwest.net/~lauramh/resume/thorn.htm>. Acessado em: 21/09/2009.
- HUZITA, E. H. M. 2008. Projeto de Pesquisa. Suporte à Percepção e ao Contexto em Ambientes de Desenvolvimento Distribuído de Software 2008. Disponível em: <http://www.sgp.uem.br:8080/sgp/super/detalhesProjeto.html?id=13973>. Acessado em: 12/07/2010.
- HUZITA, E. H. M.; STEINMACHER, I.; WIESE, I. S.; POZZA, R.; AMORIM, É.; PASCUTTI, M. C. D. Uma Proposta de Arquitetura para Ambientes de Desenvolvimento Distribuído de Software. In: XI Congreso Argentino de Ciencias de la Computación (CACIC), Entre Ríos, 2005.
- HUZITA, E. H. M.; TAIT, T. F. C.; COLANZI, T. E.; QUINAIA, M. A. Um Ambiente de Desenvolvimento Distribuído de Software – DiSEN. In: Workshop de Desenvolvimento Distribuído de Software. João Pessoa, PB, 2007.
- HUZITA, E. H. M.; TAIT, T. F. C. Gerência de Projetos de Software. In: FERREIRA, K. (org.). XII Escola Regional de Informática. Bandeirantes, PR: Faculdade Luiz Meneghel, 2006.
- INMON, W. H. Como construir o data warehouse. 2. ed. Rio de Janeiro: Campus, 1997.
- INMON, W. H.; HACKATHORN, R. D. Como usar o Data warehouse. Rio de Janeiro: Infobook, 1997.
- INMON, W. H.; WELCH, J. D.; GLASSEY, K. L. Gerenciando Data warehouse. São Paulo: Makron Books, 1999.
- KERZNER, H. Gestão de projetos: as melhores práticas. 2. ed. Porto Alegre: Bookman, 2006.
- KIMBALL, R.; REEVES, L.; ROSS, M.; THORNTHWAITTE, W. The Data warehouse Lifecycle Toolkit: expert methods for designing, developing, and deploying data warehouses. New York: Wiley Computer Publishing, 1998.
- KIMBALL, R.; ROSS, M. Data warehouse toolkit: o guia completo para modelagem multidimensional. 2. ed. Rio de Janeiro: Campus: 2002.
- LANUBILE, F.; DAMIAN, D.; OPPENHEIMER, H. L. Global Software Development: Technical, Organizational, and Social Challenges. ACM SIGSOFT Software Engineering Notes, v. 28, n. 6, nov. 2003.
- LAUDON, K. C.; LAUDON, J. P. Sistemas de informação gerenciais. 7. ed. São Paulo: Pearson Prentice Hall, 2007.
- LEME, L. H. R.; TAIT, T. F. C.; HUZITA, E. H. M. Strategy of Risk Management for a Distributed Software Engineering Environment. In: International Conference of Enterprise Information Systems. Funchal, Madeira, Portugal, 2007.
- LIMA, F. Mecanismo de apoio ao gerenciamento de RH no contexto de um ambiente distribuído de software. Maringá, 2004. Dissertação (Mestrado em Ciência da Computação) – Universidade Estadual de Maringá.
- MACHADO, F. N. R. Tecnologia e Projeto de Data warehouse: uma visão multidimensional. – 2. ed. São Paulo: Érica, 2006.
- MAGALHÃES, I. L.; PINHEIRO, W. B. Gerenciamento de Serviços de TI na prática: uma abordagem com base na ITIL. São Paulo: Novatec Editora, 2007.

- MCGOVERAN, D. Apoio à decisão. In: DATE, C. J. Introdução a sistemas de bancos de dados. 8. ed. Rio de Janeiro: Elsevier, 2003.
- MENOLLI, A. L. A.; DIAS, M. M. A Data warehouse Architecture in Layers for Science and Technology. In: Software Engineering and Knowledge Engineering, San Francisco, United States of America, 2006.
- NOVELLO, T. C. Uma abordagem de Data warehouse para Análise de Processos de Desenvolvimento de Software. Porto Alegre, 2006. Dissertação (Mestrado em Ciência da Computação) – Faculdade de Informática. Pontifícia Universidade Católica do Rio Grande do Sul.
- PAGNO, R. T. Uma ferramenta de estimativa de custos para o desenvolvimento distribuído de software. Maringá, 2010. Dissertação (Mestrado em Ciência da Computação) – Universidade Estadual de Maringá.
- PASCUTTI, M. C. D. Uma proposta de arquitetura de um ambiente de desenvolvimento de software distribuído baseada em agentes. Porto Alegre, 2002. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática. Universidade Federal do Rio Grande do Sul.
- PEDRAS, M. E. V.; HUZITA, E. H. M.; TAIT, T. F. C.; SANTIAGO, G. P. Dimanager: A Tool for Distributed Software Development Management. In: International Conference on Enterprise Information Systems. Porto, Portugal, 2004.
- PMI. Estudo de Benchmarking em Gerenciamento de Projetos 2009, Project Management Institute – Chapters Brasileiros. Disponível em: http://www.pmi.org.br/benchmarking/2009/Relatorio_Final_do_Estudo_de_Benchmarking_em_G_P_2009_-_Relatorio_Principal_-_Perspectiva_Geral.pdf, acessado em 21/07/2010.
- PRIKLADNICKI, R.; AUDY, J. L. N. Uma análise comparativa de práticas de desenvolvimento distribuído de software no Brasil e no exterior. In: XX Simpósio Brasileiro de Engenharia de Software. Florianópolis, 2006.
- PRIKLADNICKI, R. MuNDDoS - um modelo de referência para desenvolvimento distribuído de software. Porto Alegre, 2003. Dissertação (Mestrado em Ciência da Computação) – Faculdade de Informática. Pontifícia Universidade Católica do Rio Grande do Sul.
- PRIKLADNICKI, R.; PERES, F.; AUDY, J. L. N.; MÓRA, M. C.; PERDIGOTO, A. Requirements Specification Model in a Software Development Process inside a Physically Distributed Environment. In: International Conference on Enterprise Information Systems. Ciudad Real, 2002.
- REZENDE, S. O.; PUGLIESI, J. B. ; MELANDA, E. A.; PAULA, M. F. Mineração de dados. In: REZENDE, S. O. (coord.) Sistemas Inteligentes: fundamentos e aplicações. 1. ed. Barueri, SP: Manole, 2005.
- RUGGIERO JUNIOR, W. Data warehouse utilizando processamento paralelo em ambiente distribuído. São Paulo: 2007. Dissertação (Mestrado em Engenharia) – Escola Politécnica da Universidade de São Paulo.
- SELL, D.; PACHECO, R. C. S. Uma Arquitetura para Distribuição de Componentes Tecnológicos de Sistemas de Informações Baseados em Data warehouse. In: XXI Encontro Nacional de Engenharia de Produção – ENEGEP, Salvador, 2001.
- SHIMIZU, T. Decisão nas organizações: introdução aos problemas de decisão encontrados nas organizações e nos sistemas de apoio à decisão. São Paulo: Atlas, 2001.

- SINGH, H. S. Data warehouse: conceitos, tecnologias, implementação e gerenciamento. São Paulo: Makron Books, 2001.
- SMITH, S. Faça acontecer: ferramentas testadas e aprovadas para o gerenciamento de projetos. São Paulo: Clío Editora, 1997.
- TAIT, T. F. C.; DIAS, M. M. O papel do data warehouse no desenvolvimento de sistemas de informação executiva. In: Revista Tecnológica. UEM: n.13, p. 25-34, out. 2004.
- THOMSEN, E. OLAP: construindo sistemas de informações multidimensionais. Rio de Janeiro: Campus, 2002.
- TRINDADE, D. F. G.; TAIT, T. F. C.; HUZITA, E. H. M. A tool for supporting the communication in distributed software development environment. In: Journal of Computer Science and Technology. La Plata: v. 8, n. 2, p. 118-124, jul. 2008.
- TURBAN, E.; McLEAN, E.; WETHERBE, J. Tecnologia da informação para gestão. 3. ed. Porto Alegre: Bookman, 2004.
- TURBAN, E.; SHARDA, R.; ARONSON, J. E.; KING, D. Business Intelligence: um enfoque gerencial para a inteligência do negócio. Porto Alegre: Bookman, 2009.
- VIEIRA, M. F. Gerenciamento de projetos de tecnologia da informação. 2. ed. Rio de Janeiro: Elsevier, 2007.

Apêndice A

O Quadro 7 a seguir representa algumas necessidades dos gerentes de projetos de software de nível tático, relacionadas a fatos, medidas e dimensões.

Quadro 7 – Fatos e Dimensões a partir das necessidades dos gerentes de projeto

Necessidade do Gerente de Projetos		Fato	Medidas	Dimensões
1	Quanto tempo foi gasto para executar determinada atividade, por funcionário, por localização, por domínio, por fase, por nível de dificuldade, por tecnologia, por artefato, em determinado período de tempo?	Tempo e Quantidade de Atividade	Tempo gasto em horas	Artefato
				Atividade (Fase, Projeto)
				Domínio do problema
				Local
				Recurso Humano
				Tecnologia
				Tempo
2	Quão próximo ficou a estimativa de custo em relação ao realizado, por equipe, por domínio, por local, por metodologia, por projeto, por fase, por atividade, por situação da atividade, por gerente, no decorrer de determinado período de tempo?	Custo da Atividade	Valor do custo planejado em dólar Valor do custo realizado em dólar	Atividade (Fase, Projeto)
				Domínio do problema
				Equipe
				Local
				Papel
				Recurso Humano
				Situação da Atividade
				Tecnologia
Tempo				
3	Quão próximo ficou a estimativa de tempo em relação ao realizado, por equipe, por domínio, por local, por metodologia, por projeto, por fase, por atividade, por funcionário, por gerente, no decorrer de determinado período de tempo?	Tempo e Quantidade de Atividade	Tempo gasto em horas Tempo planejado em horas	Atividade (Fase, Projeto)
				Domínio do problema
				Equipe
				Local
				Papel
				Recurso Humano
				Tecnologia
				Tempo

Quadro 7 – Fatos e Dimensões a partir das necessidades dos gerentes de projeto (continuação)

Necessidade do Gerente de Projetos		Fato	Medidas	Dimensões
4	Quanto tempo de experiência os desenvolvedores possuíam em determinado domínio, por localização, por equipe, no decorrer de determinado período de tempo?	Experiência	Tempo de experiência em projetos internos em horas	Domínio do problema
				Equipe
				Local
				Recurso Humano
				Tempo
5	Quantas solicitações de mudança nos requisitos ocorreram por analista de requisitos, por tecnologia, por domínio, por local, em determinado período de tempo?	Manutenção de requisito	Quantidade de manutenções	Atividade de manutenção (Projeto)
				Domínio do problema
				Local
				Papel
				Recurso Humano
				Requisito
				Tecnologia
Tempo				
6	Quantas solicitações de mudança na interface ocorreram, por projeto, no decorrer de determinado período de tempo?	Manutenção de requisito	Quantidade de manutenções	Atividade de manutenção (Projeto)
				Requisito
				Tempo
7	Quanto funcionários possuíam conhecimento em algum tipo de tecnologia, por equipe, historicamente, no decorrer de determinado período de tempo?	Conhecimento e treinamento da equipe em tecnologia	Quantidade máxima de recursos humanos com conhecimento Média de recursos humanos com conhecimento	Equipe
				Tecnologia
				Tempo

Quadro 7 – Fatos e Dimensões a partir das necessidades dos gerentes de projeto (continuação)

Necessidade do Gerente de Projetos		Fato	Medidas	Dimensões
8	Quanto tempo foi gasto com o desenvolvimento de projetos em um domínio, por localidade, em determinado período de tempo?	Tempo e Quantidade de Atividade	Tempo gasto em horas	Domínio do problema
				Local
				Tempo
9	Quanto tempo de treinamento foi realizado por funcionários, por tecnologia, no decorrer de um período de tempo?	Conhecimento e treinamento do recurso humano em tecnologia	Tempo de treinamento em horas	Recurso Humano
				Tecnologia
				Tempo
10	Qual a variação no nível de conhecimento em uma tecnologia, historicamente, antes e depois da realização de treinamentos relacionados àquela tecnologia, por funcionário, por determinado período de tempo?	Conhecimento e treinamento do recurso humano em tecnologia	Nível de conhecimento Tempo de treinamento em horas	Recurso Humano
				Tecnologia
				Tempo
11	Qual o índice de atraso para atividades concluídas de determinada tecnologia, mensalmente, por fase, em um determinado período de tempo?	Tempo e Quantidade de Atividade	Contagem de registro de atividade	Atividade (Fase, Projeto)
				Situação da Atividade
				Tecnologia
				Tempo
12	Qual o índice de atividades concluídas no prazo e antecipadamente, por tecnologia, por fase, por funcionário, no decorrer de determinado período de tempo?	Tempo e Quantidade de Atividade	Contagem de registro de atividade	Atividade (Fase, Projeto)
				Recurso Humano
				Situação da Atividade
				Tecnologia
				Tempo
13	Qual o índice de reprogramação de atividades, por funcionário, por fase, no decorrer de determinado período de tempo?	Tempo e Quantidade de Atividade	Quantidade de alterações no cronograma	Atividade (Fase, Projeto)
				Recurso Humano
				Tempo

Quadro 7 – Fatos e Dimensões a partir das necessidades dos gerentes de projeto (continuação)

Necessidade do Gerente de Projetos		Fato	Medidas	Dimensões
14	Qual a quantidade de atividades retornadas para correção (não homologadas), por desenvolvedor, por local, por projeto, por equipe, por fase, por tecnologia, por linguagem de programação, por metodologia, por framework, por arquitetura, por domínio, no decorrer de determinado período de tempo?	Tempo e Quantidade de Atividade	Quantidade de atividades retomadas	Atividade (Fase, Projeto)
				Domínio do problema
				Equipe
				Local
				Recurso Humano
				Tecnologia
				Tempo
15	Qual o número de atividades atribuídas a cada funcionário, no decorrer de determinado período de tempo, por localidade, por equipe, por fase, por projeto?	Atribuição de atividade	Contagem de registro de atribuição	Atividade (Fase, Projeto)
				Equipe
				Local
				Recurso Humano
				Tempo
16	Qual o número de atividades atribuídas a cada equipe, no decorrer de determinado período de tempo, por localidade, por equipe, por fase, por projeto?	Atribuição de atividade	Contagem de registro de atribuição	Atividade (Fase, Projeto)
				Equipe
				Local
				Tempo
17	Qual o tempo de duração de cada tipo de atividade historicamente em diferentes projetos, por domínio, por tecnologia, por fase, por linguagem?	Tempo e Quantidade de Atividade	Tempo gasto em horas	Atividade (Fase, Projeto)
				Domínio do problema
				Tecnologia
				Tempo

Quadro 7 – Fatos e Dimensões a partir das necessidades dos gerentes de projeto (continuação)

Necessidade do Gerente de Projetos		Fato	Medidas	Dimensões
18	Qual o custo de cada tipo de atividade historicamente em diferentes projetos, por domínio, por tecnologia, por fase, por artefato, por linguagem?	Custo da Atividade	Valor do custo realizado em dólar	Artefato
				Atividade (Fase, Projeto)
				Domínio do problema
				Tecnologia
				Tempo
19	Qual a variação no nível de conhecimento da equipe, historicamente, por tecnologia, em determinado período de tempo? Qual a relação desta variação com o tempo de treinamento recebido neste mesmo período?	Conhecimento e treinamento da equipe em tecnologia	Média do nível de conhecimento Nível máximo de conhecimento Tempo total de treinamento da equipe em horas Tempo médio de treinamento por recurso humano em horas	Equipe
				Tecnologia
				Tempo
20	Quanto tempo de experiência os desenvolvedores possuíam em determinada tecnologia, por papel, por localização, por equipe, no decorrer de determinado período de tempo?	Experiência	Tempo de experiência em projetos internos em horas	Domínio do problema
				Equipe
				Local
				Papel
				Recurso Humano
				Tecnologia
Tempo				

Apêndice B

As figuras a seguir apresentam os Esquemas Estrela para os fatos extraídos das necessidades dos gerentes de projeto, que são apresentados no Capítulo 5. Os esquemas foram modelados com o auxílio da ferramenta *Power Architect* que possui um módulo específico para OLAP. Esse módulo facilita a modelagem das tabelas de fato e dimensão e a geração dos *scripts* para o banco de dados a partir do modelo.

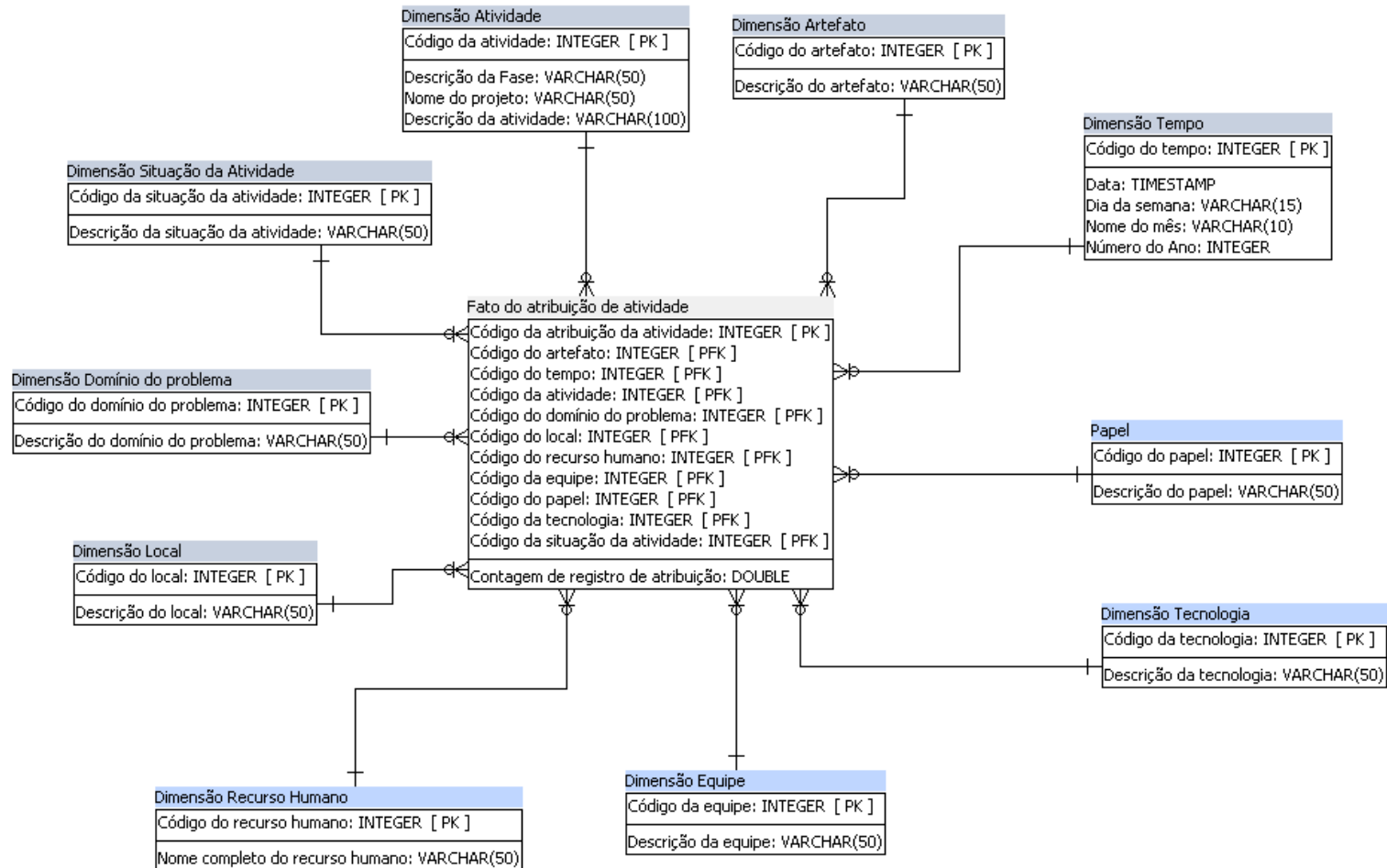


Figura 13 – Esquema Estrela para o fato Atribuição de atividade

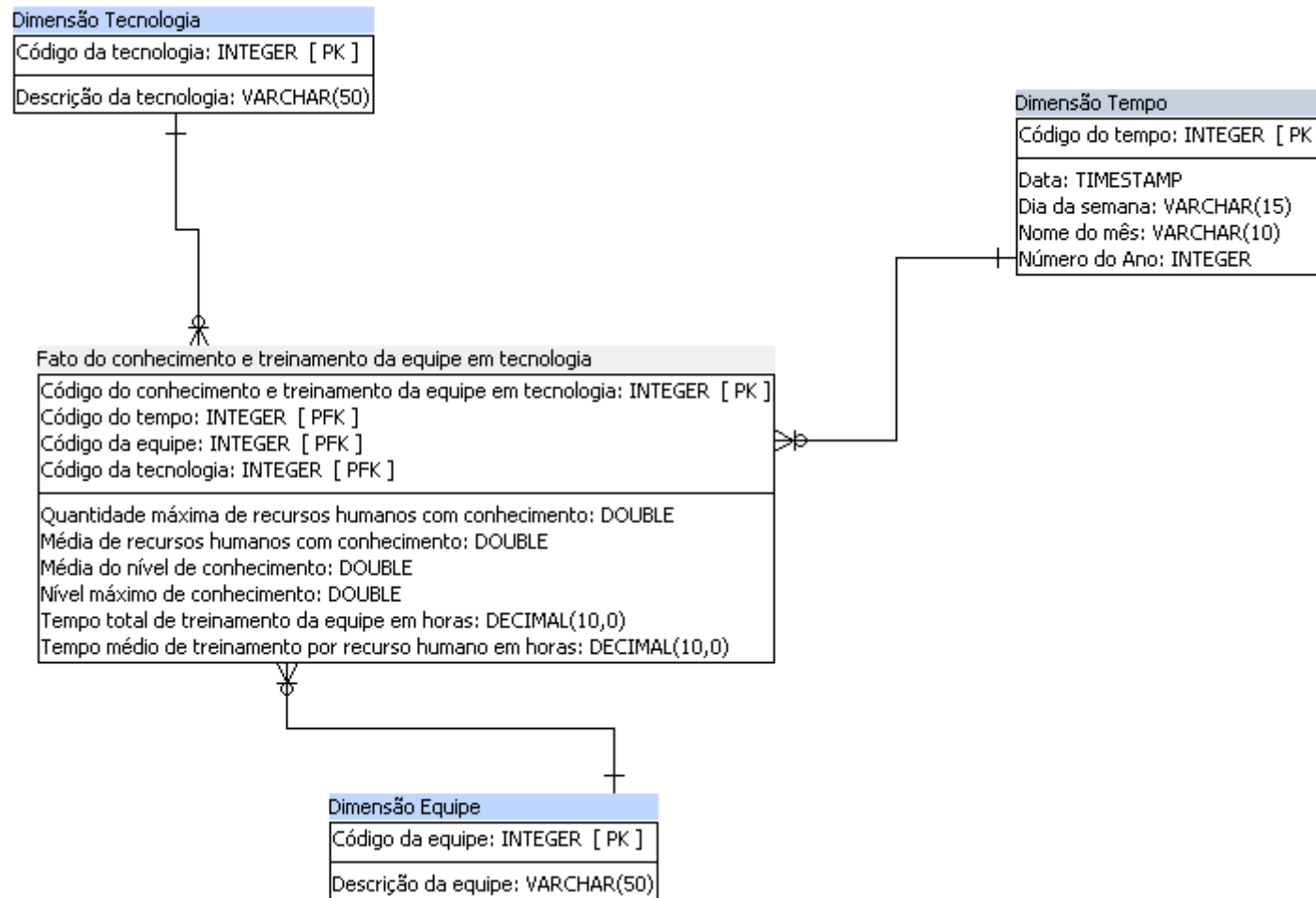


Figura 14 – Esquema Estrela para o fato Conhecimento e Treinamento da Equipe em Tecnologia

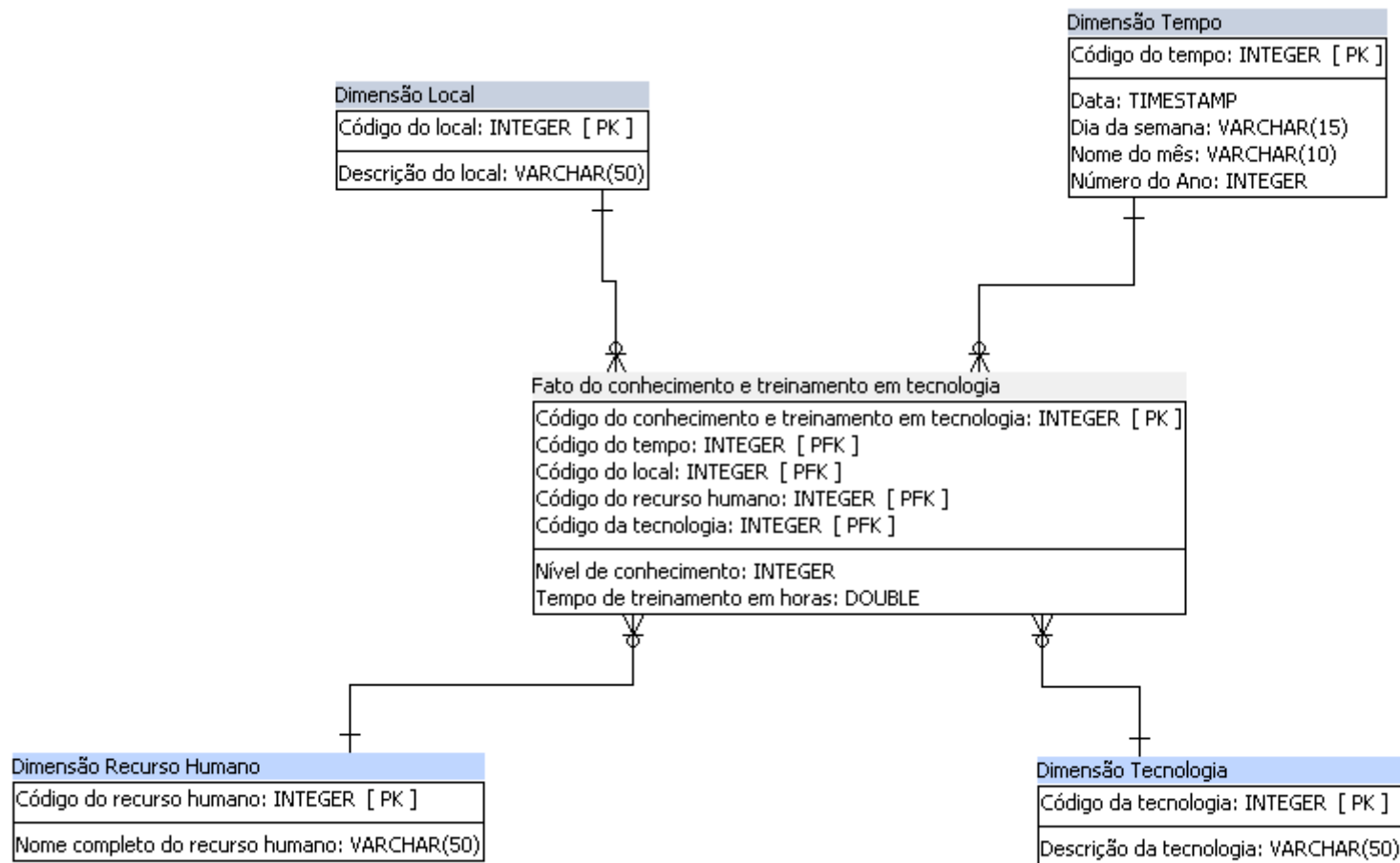


Figura 15 – Esquema Estrela para o fato Conhecimento e Treinamento do Recurso Humano em Tecnologia

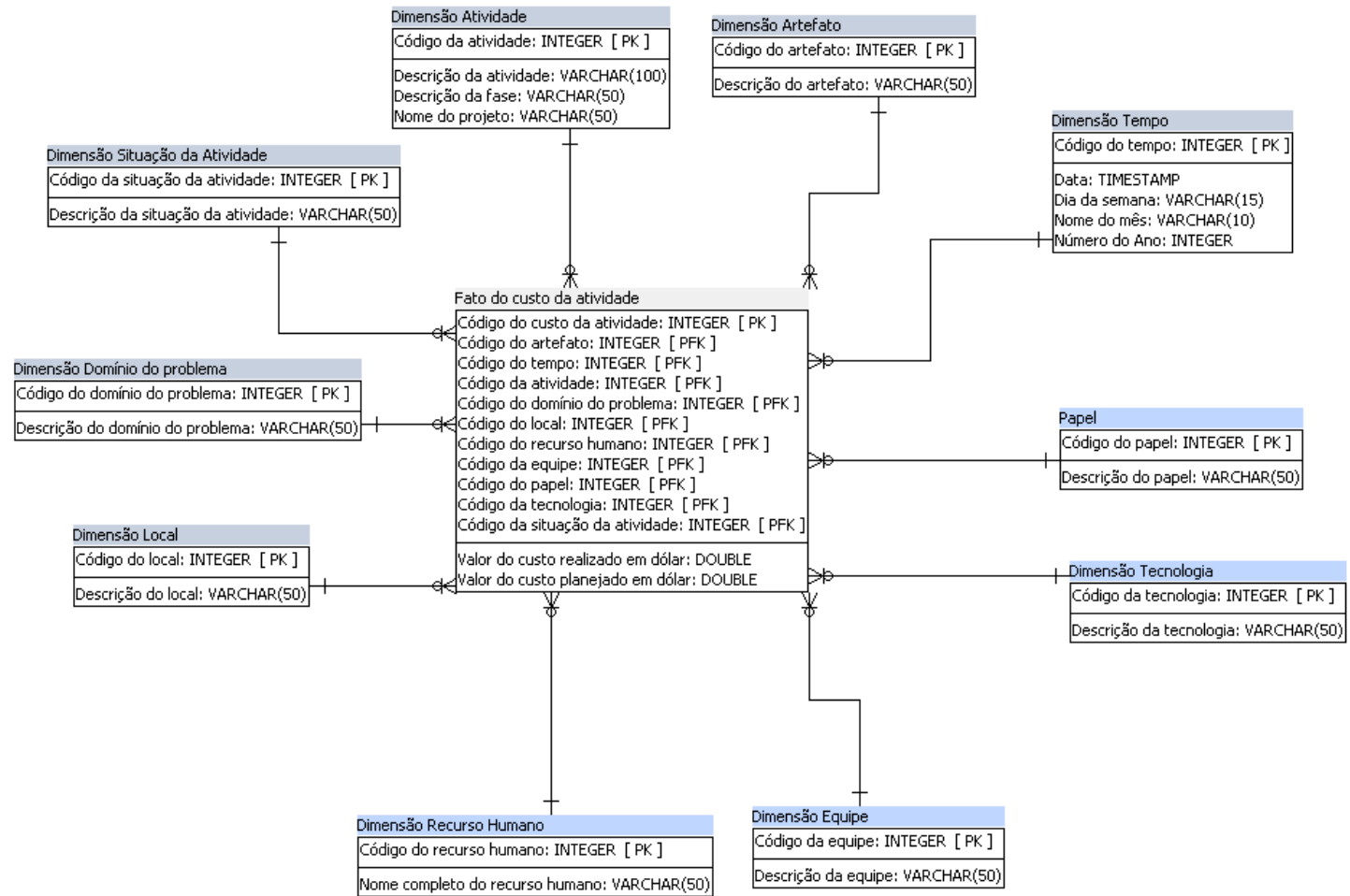


Figura 16 – Esquema Estrela para o fato Custo da Atividade

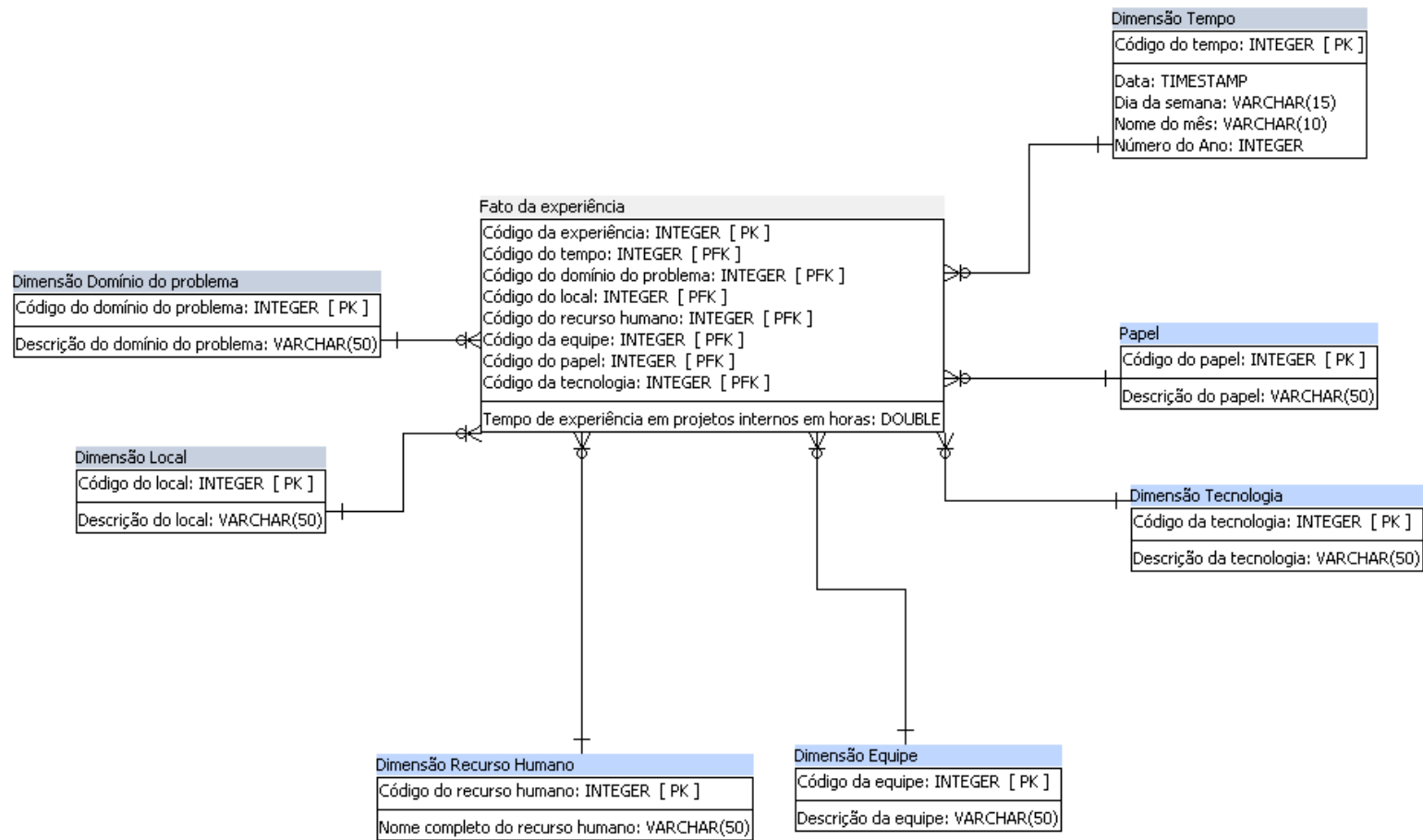


Figura 17 – Esquema Estrela para o fato Experiência

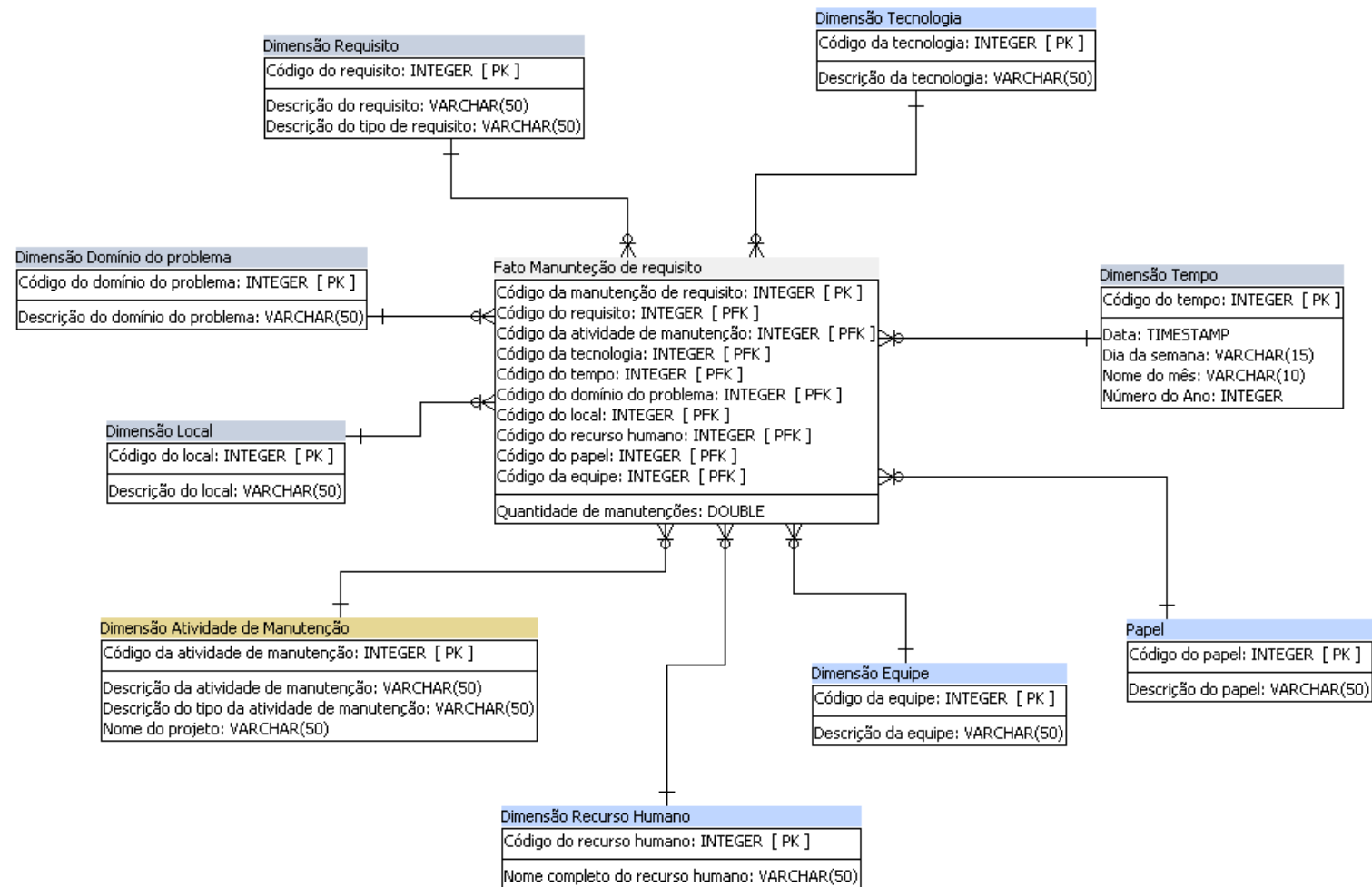


Figura 18 – Esquema Estrela para o fato Manutenção do Requisito

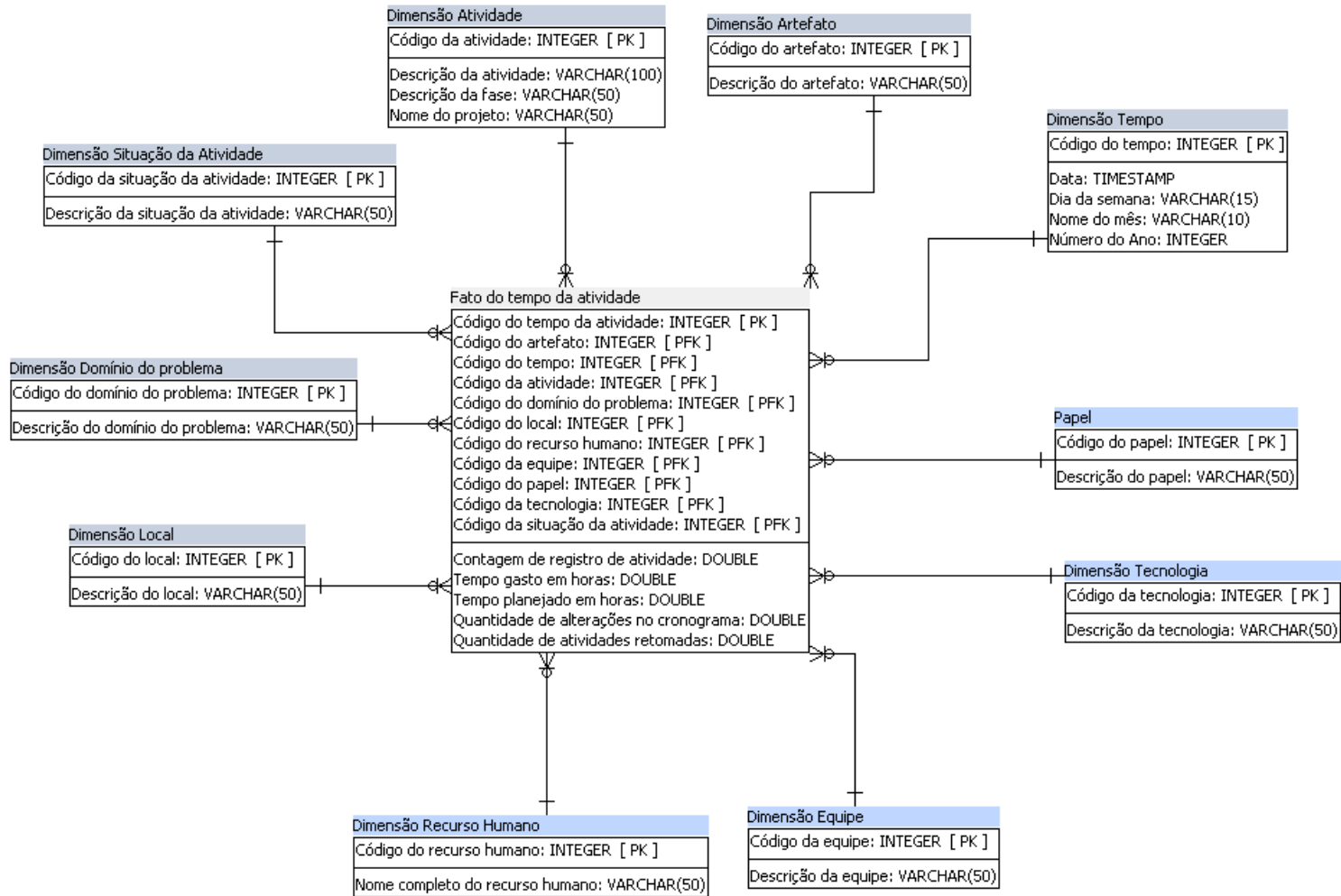


Figura 19 – Esquema Estrela para o fato Tempo e Quantidade de Atividade

Apêndice C

Uma análise do banco de dados utilizado como repositório de dados pelo ambiente DiSEN foi realizada com o auxílio das ferramentas *pgAdmin* e *Power Architect*.

Para obter os metadados do banco de dados como tabelas, campos, chaves estrangeiras e tipos de dados, foram executadas as seguintes consultas em SQL com o apoio da ferramenta *pgAdmin*:

Para obtenção das chaves estrangeiras e tabelas referenciadas:

```
SELECT
  n.nspname AS esquema,
  cl.relname AS tabela,
  a.attname AS coluna,
  ct.conname AS chave,
  nf.nspname AS esquema_ref,
  clf.relname AS tabela_ref,
  af.attname AS coluna_ref,
  pg_get_constraintdef(ct.oid) AS criar_sql
FROM pg_catalog.pg_attribute a
JOIN pg_catalog.pg_class cl ON (a.attrelid = cl.oid AND cl.relkind = 'r')
JOIN pg_catalog.pg_namespace n ON (n.oid = cl.relnamespace)
JOIN pg_catalog.pg_constraint ct ON (a.attrelid = ct.conrelid AND
  ct.confrelid != 0 AND ct.conkey[1] = a.attnum)
JOIN pg_catalog.pg_class clf ON (ct.confrelid = clf.oid AND clf.relkind = 'r')
JOIN pg_catalog.pg_namespace nf ON (nf.oid = clf.relnamespace)
JOIN pg_catalog.pg_attribute af ON (af.attrelid = ct.confrelid AND
  af.attnum = ct.confkey[1]);
```

Para obtenção dos campos e tipos de dados das tabelas:

```

SELECT
    rel.nspname AS Esquema,
    rel.relname AS Tabela,
    attrs.attname AS Campo,
    "Type",
    "Default",
    attrs.attnotnull AS "NOT NULL"
FROM (
    SELECT
        c.oid,
        n.nspname,
        c.relname
    FROM pg_catalog.pg_class c
        LEFT JOIN pg_catalog.pg_namespace n ON n.oid = c.relnamespace
    WHERE pg_catalog.pg_table_is_visible(c.oid) ) rel
    JOIN (
        SELECT
            a.attname,
            a.attrelid,
            pg_catalog.format_type(a.atttypid, a.atttypmod) as "Type",
            (SELECT
                substring(d.adsrc for 128)
            FROM pg_catalog.pg_attrdef d
            WHERE d.adrelid = a.attrelid AND d.adnum = a.attnum AND a.atthasdef
            ) as "Default",
            a.attnotnull,
            a.attnum
        FROM pg_catalog.pg_attribute a
        WHERE a.attnum > 0 AND NOT a.attisdropped )
    attrs ON (attrs.attrelid = rel.oid )
WHERE relname in (
    SELECT tablename AS tabela
    FROM pg_catalog.pg_tables
    WHERE schemaname IN ('public')
    ORDER BY tablename)
ORDER BY relname;

```

Por meio da função de conexão com o banco de dados, a ferramenta *Power Architect* foi utilizada como apoio para análise visual dos relacionamentos entre as tabelas.

Após a análise, os dados considerados úteis para apoiar a tomada de decisão na GP-DDS foram agrupados por assunto. O resultado da análise incluindo assunto, significado, tabela de origem e possíveis valores assumidos, é apresentado a seguir.

Os dados relacionados à GP-DDS, que foram localizados no banco de dados do DiSEN, pertencem aos seguintes assuntos: Qualidade, Recurso e Risco. Os dados que são Comuns às várias áreas de assunto, foram agrupados em uma área denominada Comum. Além da área de assunto, uma segunda classificação foi realizada separando as tabelas por assunto.

Os quadros a seguir apresentam as tabelas do banco de dados, seus atributos, identificação de chaves primárias e estrangeiras, tabelas de referência e o tipo de dado dos atributos.

Quadro 8 – Dados do DiSEN - Área de assunto Comum (1)

		Tabela	Atributo	Chave	Tabela de referência	Tipo de Dado
artefato		arquivo	id	PK		integer
		arquivo	id_artefato	FK	artefato	integer
		arquivo	id_participante	FK	participante	integer
		arquivo	descricao	-		text
		arquivo	estado	-		character varying(30)
		arquivo	nome	-		character varying(100)
		artefato	id	PK		integer
		artefato	id_tarefa	FK	tarefa	integer
		artefato	id_tipo_artefato	FK	tipo_artefato	integer
		artefato	descricao	-		text
		artefato	estado	-		character varying(30)
		artefato	nome	-		character varying(100)
		tipo_artefato	id	PK		integer
		tipo_artefato	descricao	-		text
		tipo_artefato	nome	-		character varying(100)
local		local	id	PK		integer
		local	gerente	FK	recurso / usuario	integer
		local	localizacao	-		character varying(100)
		local	nome	-		character varying(100)
		local	padrao	-		boolean
processo		atividade_processo	id	PK		integer
		atividade_processo	id_fase_processo	FK	fase_processo	integer
		atividade_processo	id_tipo_artefato	FK	tipo_artefato	integer
		atividade_processo	descricao	-		text
		atividade_processo	nome	-		character varying(100)
		dependencia_atividade_processo	id	PK		integer
		dependencia_atividade_processo	id_atividade_processo	FK	atividade_processo	integer
		dependencia_atividade_processo	id_atividade_processo_dependente	FK	atividade_processo	integer
		dependencia_atividade_processo	dados	-		boolean
		dependencia_atividade_processo	sequencia	-		boolean
		dependencia_fase_processo	id	PK		integer
		dependencia_fase_processo	id_fase_processo	FK	fase_processo	integer
		dependencia_fase_processo	id_fase_processo_dependente	FK	fase_processo	integer
		fase_processo	id	PK		integer
		fase_processo	id_processo	FK	processo	integer
		fase_processo	descricao	-		text
		fase_processo	nome	-		character varying(100)
processo	id	PK		integer		
processo	descricao	-		text		
processo	nome	-		character varying(100)		
processo	versao	-		character varying(10)		

Quadro 9 – Dados do DiSEN - Área de assunto Comum (2)

	Tabela	Atributo	Chave	Tabela de referência	Tipo de Dado
Área de assunto Comum projeto	atividade_projeto	id	PK		integer
	atividade_projeto	id_artefato	FK	artefato	integer
	atividade_projeto	id_atividade_processo	FK	atividade_processo	integer
	atividade_projeto	id_fase_projeto	FK	fase_projeto	integer
	atividade_projeto	datafim	-		date
	atividade_projeto	datafimprevisto	-		date
	atividade_projeto	datainicio	-		date
	atividade_projeto	datainioprevisto	-		date
	atividade_projeto	descricao	-		text
	atividade_projeto	estado	-		character varying(255)
	atividade_projeto	nome	-		character varying(100)
	dependencia_atividade_projeto	id	PK		integer
	dependencia_atividade_projeto	id_atividade_projeto	FK	atividade_projeto	integer
	dependencia_atividade_projeto	id_atividade_projeto_dependente	FK	atividade_projeto	integer
	dependencia_atividade_projeto	dados	-		boolean
	dependencia_atividade_projeto	sequencia	-		boolean
	dependencia_fase_projeto	id	PK		integer
	dependencia_fase_projeto	id_fase_projeto	FK	fase_projeto	integer
	dependencia_fase_projeto	id_fase_projeto_dependente	FK	fase_projeto	integer
	fase_projeto	id	PK		integer
	fase_projeto	id_fase_processo	FK	fase_processo	integer
	fase_projeto	id_projeto	FK	projeto	integer
	fase_projeto	descricao	-		text
	fase_projeto	estado	-		character varying(255)
	fase_projeto	nome	-		character varying(100)
	projeto	id	PK		integer
	projeto	id_processo	FK	processo	integer
	projeto	data_fim	-		date
	projeto	data_fim_previsto	-		date
	projeto	data_inicio	-		date
	projeto	data_inicio_previsto	-		date
	projeto	descricao	-		text
	projeto	estado	-		character varying(255)
	projeto	nome	-		character varying(100)
	projeto	objetivos	-		text
	tarefa	id	PK		integer
	tarefa	id_atividade_projeto	FK	atividade_projeto	integer
	tarefa	data_fim	-		date
	tarefa	data_fim_previsto	-		date
	tarefa	data_inicio	-		date
	tarefa	data_inicio_previsto	-		date
tarefa	descricao	-		text	
tarefa	estado	-		character varying(255)	
tarefa	nome	-		character varying(100)	

Quadro 10 – Dados do DiSEN - Área de assunto Comum (3)

		Tabela	Atributo	Chave	Tabela de referência	Tipo de Dado
Área de assunto Comum	reuniao	vimee_item_pauta	id	PK		integer
		vimee_item_pauta	id_reuniao	FK	vimee_reuniao	integer
		vimee_item_pauta	assunto	-		character varying(1024)
		vimee_item_pauta	duracao	-		bigint
		vimee_participante_reuniao	id	PK		integer
		vimee_participante_reuniao	id_reuniao	FK	vimee_reuniao	integer
		vimee_participante_reuniao	id_usuario	FK	recurso / usuario	integer
		vimee_participante_reuniao	tipoparticipantereuniao	-		character varying(255)
		vimee_participante_reuniao	estadoparticipantereuniao	-		character varying(255)
		vimee_reuniao	id	PK		integer
		vimee_reuniao	id_tipo_reuniao	FK	vimee_tipo_reuniao	integer
		vimee_reuniao	ata	-		character varying(255)
		vimee_reuniao	descricao	-		character varying(1024)
		vimee_reuniao	estadoreuniao	-		character varying(255)
		vimee_reuniao	fim	-		timestamp without time zone
		vimee_reuniao	fimprevisto	-		timestamp without time zone
		vimee_reuniao	inicio	-		timestamp without time zone
		vimee_reuniao	iniocioprevisto	-		timestamp without time zone
		vimee_secao	id	PK		integer
		vimee_secao	itempauta_id	FK	vimee_item_pauta	integer
		vimee_secao	estadosecao	-		character varying(255)
		vimee_secao	horafim	-		timestamp without time zone
		vimee_secao	horainicio	-		timestamp without time zone
		vimee_secao	parecerfinal	-		text
		vimee_tipo_reuniao	id	PK		integer
		vimee_tipo_reuniao	descricao	-		character varying(255)
		vimee_votacao	id	PK		integer
		vimee_votacao	secao_id	FK	vimee_secao	integer
		vimee_votacao	pergunta	-		character varying(255)
		vimee_votacao_item	id	PK		integer
		vimee_votacao_item	votacao_id	FK	vimee_votacao	integer
		vimee_votacao_item	item	-		character varying(255)
vimee_votacao_item	votos	-		integer		

Quadro 11 – Dados do DiSEN - Área de assunto Qualidade

		Tabela	Atributo	Chave	Tabela de referência	Tipo de Dado
Área de assunto Qualidade	problema	log	id	PK		integer
		log	id_tipo_log	FK	tipo_log	integer
		log	descricao	-		text
		log	solucao	-		text
		log_tarefa	id	PK		integer
		log_tarefa	log	FK	log	integer
		log_tarefa	tarefa	FK	tarefa	integer
		log_tarefa	contexto	-		text
		log_tarefa	data	-		date
		tipo_log	id	PK		integer
		tipo_log	descricao	-		text
		tipo_log	nome	-		character varying(100)

Quadro 12 – Dados do DiSEN - Área de assunto Recurso (1)

	Tabela	Atributo	Chave	Tabela de referência	Tipo de Dado
recurso	recurso	id	PK		integer
	recurso	estado	-		character varying(30)
	recurso	nome	-		character varying(100)
	recurso	type	-		character varying(20)
Área de assunto Recurso recurso humano	afinidade_usuario	id	PK		integer
	afinidade_usuario	id_usuario	FK	recurso / usuario	integer
	afinidade_usuario	id_usuario_afinidade	FK	recurso / usuario	integer
	afinidade_usuario	nivel	-		integer
	aptidao	id	PK		integer
	aptidao	id_usuario	FK	recurso / usuario	integer
	aptidao	porcentagem_ne	-		double precision
	aptidao	porcentagem_nese	-		double precision
	aptidao	porcentagem_no	-		double precision
	aptidao	porcentagem_none	-		double precision
	aptidao	porcentagem_noso	-		double precision
	aptidao	porcentagem_se	-		double precision
	aptidao	porcentagem_so	-		double precision
	aptidao	porcentagem_ose	-		double precision
	habilidade	id	PK		integer
	habilidade	descricao	-		character varying(100)
	habilidade	nome	-		character varying(100)
	habilidade_usuario	id	PK		integer
	habilidade_usuario	id_habilidade	FK	habilidade	integer
	habilidade_usuario	id_usuario	FK	recurso / usuario	integer
	habilidade_usuario	nivel	-		character varying(255)
	papel	id	PK		integer
	papel	descricao	-		text
	papel	nome	-		character varying(100)
	papel_tarefa	id	PK		integer
	papel_tarefa	id_papel	FK	papel	integer
	papel_tarefa	id_participante	FK	participante	integer
	papel_tarefa	id_tarefa	FK	tarefa	integer
	participante	id	PK		integer
	participante	id_projeto	FK	projeto	integer
	participante	id_usuario	FK	recurso / usuario	integer
	participante	custo_por_hora	-		double precision
	participante	estado	-		character varying(255)
participante	gerente	-		boolean	
treinamento	id	PK		integer	
treinamento	descricao	-		character varying(100)	
treinamento	nome	-		character varying(100)	
treinamento_usuario	id	PK		integer	
treinamento_usuario	id_treinamento	FK	treinamento	integer	
treinamento_usuario	id_usuario	FK	recurso / usuario	integer	
treinamento_usuario	data_fim	-		date	
treinamento_usuario	data_inicio	-		date	
treinamento_usuario	nivel	-		character varying(255)	
usuario	id	PK	recurso	integer	
usuario	id_local	FK	local	integer	
usuario	custo_por_hora	-		double precision	
usuario	excluido	-		boolean	

Quadro 13 – Dados do DiSEN - Área de assunto Recurso (2)

		Tabela	Atributo	Chave	Tabela de referência	Tipo de Dado
Área de assunto Recurso	recurso material	local_recurso_material	id	PK		integer
		local_recurso_material	id_local	FK	local	integer
		local_recurso_material	id_recurso_material	FK	recurso / recurso material	integer
		local_recurso_material	quantidade	-		integer
		local_recurso_material	quantidade_utilizada	-		integer
		recurso_material	id	PK	recurso	integer
		recurso_material	id_tipo_recurso_material	FK	tipo_recurso_material	integer
		recurso_material	configuracao	-		text
		recurso_material	custo	-		double precision
		recurso_material	descricao	-		text
		recurso_material	fabricante	-		character varying(100)
		recurso_material	versao	-		character varying(100)
		recurso_material_tarefa	id	PK		integer
		recurso_material_tarefa	id_recurso_material	FK	recurso / recurso material	integer
		recurso_material_tarefa	id_tarefa	FK	tarefa	integer
		recurso_material_tarefa	quantidade	-		integer
		tipo_recurso_material	id	PK		integer
		tipo_recurso_material	descricao	-		text
		tipo_recurso_material	nome	-		character varying(100)
		tipo_recurso_material_atividade_projeto	id	PK		integer
		tipo_recurso_material_atividade_projeto	id_atividade_projeto	FK	atividade_projeto	integer
		tipo_recurso_material_atividade_projeto	id_tipo_recurso_material	FK	tipo_recurso_material	integer
		tipo_recurso_material_atividade_projeto	quantidade	-		integer

Quadro 14 – Dados do DiSEN - Área de assunto Risco

	Tabela	Atributo	Chave	Tabela de referência	Tipo de Dado
risco	risco	id	PK		integer
	risco	nome	-		character varying(100)
	risco	risco_escopo	-		character varying(255)
	risco	risco_negocios	-		boolean
	risco	risco_produto	-		boolean
	risco	risco_projeto	-		boolean
	taxonomia1	id	PK		integer
	taxonomia1	descricao	-		character varying(100)
	taxonomia2	id	PK		integer
	taxonomia2	taxonomia1_id	FK	taxonomia1	integer
Área de assunto Risco	risco_analise	id	PK		integer
	risco_analise	risco_projeto	FK	risco_projeto	integer
	risco_analise	distributividade	-		character varying(255)
	risco_analise	exposicao	-		integer
	risco_analise	fator	-		character varying(255)
	risco_analise	impacto	-		integer
	risco_analise	probabilidade	-		integer
	risco_controle	id	PK		integer
	risco_controle	risco_projeto	FK	risco_projeto	integer
	risco_controle	contencao_aplicada	-		boolean
	risco_controle	contencao_eficaz	-		boolean
	risco_controle	detalhes	-		character varying(255)
	risco_controle	risco_ocorreu	-		boolean
	risco_controle	tratamento_eficaz	-		boolean
	risco_controle	tratamento_sugerido	-		character varying(255)
	risco_projeto	id	PK		integer
	risco_projeto	local_identificador	FK	local	integer
	risco_projeto	projeto	FK	projeto	integer
	risco_projeto	risco	FK	risco	integer
	risco_projeto	taxonomia2	FK	taxonomia2	integer
	risco_projeto	estado_risco_projeto	-		character varying(255)
	risco_projeto	restrito_local	-		boolean
	risco_projeto	risco_ocorreu	-		boolean
	risco_projeto	tratamento_eficaz	-		boolean
	risco_projeto_contencao	id	PK		integer
	risco_projeto_contencao	alternativa	-		character varying(255)
	risco_projeto_contencao	custo	-		double precision
	risco_projeto_contencao	data_fim_previsto	-		date
	risco_projeto_contencao	data_inicio_previsto	-		date
	risco_projeto_contencao	descricao	-		character varying(255)
	risco_projeto_contencao	disparador	-		character varying(255)
	risco_projeto_contencao	justificativa	-		character varying(255)
	risco_projeto_tratamento	id	PK		integer
	risco_projeto_tratamento	atividades	-		character varying(255)
	risco_projeto_tratamento	custo	-		double precision
	risco_projeto_tratamento	data_fim_previsto	-		date
	risco_projeto_tratamento	data_inicio_previsto	-		date
	risco_projeto_tratamento	documentos	-		character varying(255)
	risco_projeto_tratamento	recursos	-		character varying(255)
	risco_tratamento	id	PK		integer
	risco_tratamento	risco_projeto	FK	risco_projeto	integer
	risco_tratamento	risco_projeto_contencao	FK	risco_projeto_contencao	integer
risco_tratamento	risco_projeto_tratamento	FK	risco_projeto_tratamento	integer	
risco_tratamento	tratamento	-		character varying(255)	

O significado de alguns atributos é apresentado nos quadros a seguir, agrupado por tabela e assunto.

Quadro 15 – Significado dos Dados do DiSEN - Área de assunto Comum

		Tabela	Atributo	Significado
Área de assunto Comum	artefato	arquivo	nome	Arquivo criado por uma pessoa e relacionado a um artefato, um artefato pode possuir diversos arquivos.
		artefato	nome	Produto gerado pela execução de uma atividade. Uma atividade pode gerar um artefato que pode ser composto por vários arquivos.
		tipo_artefato	nome	Tipo do artefato gerado pela atividade.
	local	local	nome	Região ou unidade administrativa a qual pertence o recurso humano ou material
		local	padrao	Local padrão da organização (apenas um para toda a organização).
	processo	atividade_processo	nome	Atividade que faz parte de um processo. Um projeto terá uma instância de processo que já terá atividades padrão mas caso haja alguma atividade extra, será adicionada para o processo instanciado para o projeto em questão.
		dependencia_atividade_processo	dados	Depende de dados?
		dependencia_atividade_processo	id_atividade_processo_dependente	Dependência entre uma atividade do processo e outra.
		dependencia_atividade_processo	sequencia	Depende sequencialmente?
		dependencia_fase_processo	id_fase_processo_dependente	Dependência entre uma fase do processo e outra.
		fase_processo	nome	Fases referentes ao Processo utilizado para o desenvolvimento do projeto.
		processo	nome	Processo utilizado para o desenvolvimento do projeto
	projeto	atividade_projeto	descricao	Atividade relacionada ao projeto, pode estar relacionada diretamente a uma das atividade do processo ou pode ser uma atividade extra que não tem relação com as atividades do processo vinculado ao projeto. Armazena detalhes específicos do projeto, como datas de início e fim e estado.
		dependencia_atividade_projeto	dados	Depende de dados?
		dependencia_atividade_projeto	id_atividade_projeto_dependente	Dependência entre uma atividade do projeto e outra.
		dependencia_atividade_projeto	sequencia	Depende sequencialmente?
		dependencia_fase_projeto	id_fase_projeto_dependente	Dependência entre uma fase do projeto e outra.
		fase_projeto	nome	Fases referentes ao processo de desenvolvimento específicas para um projeto. Podem ser herdadas das fases do processo relacionado ao projeto ou pode ser criada uma fase extra para o projeto, sem vínculo com o processo.
		projeto	nome	Conjunto de atividades com objetivo de desenvolver um software
	tarefa	nome	Menor unidade do projeto, cada atividade é dividida em tarefas.	
	reunião	vimee_item_pauta	assunto	Assunto tratado na reunião.

Quadro 16 – Significado dos Dados do DiSEN - Área de assunto Qualidade

Área de assunto Qualidade		Tabela	Atributo	Significado
problema	log	descricao	Qualquer ocorrência adversa (problema) na tarefa do projeto, descreve um tipo (técnico, organizacional, entre outros) e uma solução.	
	log	solucao	Solução proposta ou aplicada para resolver um problema.	
	tipo_log	nome	Tipo de problema	

Quadro 17 – Significado dos Dados do DiSEN - Área de assunto Recurso

Área de assunto Recurso		Tabela	Atributo	Significado
recurso	recurso	nome	Recurso para ser utilizado no projeto que pode ser humano ou material e pode estar disponível ou indisponível.	
	recurso	type	Tipo do recurso	
recurso humano	afinidade_usuario	id_usuario_afinidade	Afinidade de uma pessoa com outra	
	afinidade_usuario	nivel	Assume valores de 0 a 5 (0:Não tem Afinidade a 5:bastante Afinidade)	
	aptidao	id	Aptidão de uma pessoa, formada por valores percentuais definidos para cada um dos oito perfis, com base nas respostas obtidas a partir de um questionário aplicado à pessoa	
	aptidao	porcentagem_ne	Perfil criatividade	
	aptidao	porcentagem_nese	Perfil criativo/interpessoal	
	aptidao	porcentagem_no	Perfil raciocínio lógico	
	aptidao	porcentagem_none	Perfil intelectual	
	aptidao	porcentagem_noso	Perfil técnico/organizacional	
	aptidao	porcentagem_se	Perfil comunicação	
	aptidao	porcentagem_so	Perfil organização	
	aptidao	porcentagem_sose	Perfil operacional	
	habilidade	nome	Capacidade de utilizar um conhecimento.	
	papel	nome	Papel exercido por uma pessoa em uma tarefa de um projeto em que a pessoa seja participante.	
	participante	id	Todo usuário que faz parte de uma equipe de projeto.	
	participante	gerente	É o gerente do projeto?	
	treinamento	nome	Aptidão adquirida pela participação de curso de aprendizagem.	
	treinamento_usuario	id	Relacionamento entre um treinamento e um recurso humano.	
	treinamento_usuario	nivel	Nível do treinamento realizado pelo usuário.	
	usuario	id	Recurso humano	
	usuario	excluido	Recurso humano não faz mais parte da organização?	
recurso material	recurso_material	descricao	Recurso necessário para execução de uma tarefa. Pode ser produto ou serviço.	
	tipo_recurso_material	nome	Tipo do recurso material	
	tipo_recurso_material_atividade_projeto	id	Relacionamento entre um recurso material e uma atividade do projeto.	

Quadro 18 – Significado dos Dados do DiSEN - Área de assunto Risco

	Tabela	Atributo	Significado
risco	risco	nome	Problemas que pode ocorrer na execução de um projeto.
	risco	risco_escopo	Tipo de risco
	risco	risco_negocios	Tipo de categoria do risco
	risco	risco_produto	Tipo de categoria do risco
	risco	risco_projeto	Tipo de categoria do risco
	taxonomia1	descricao	Categoria da taxonomia do risco (estrutura de classificação do risco)
taxonomia2	descricao	Taxonomia do risco (estrutura de classificação do risco)	
Área de assunto Risco risco do projeto	risco_analise	id	Avaliação de um risco identificado.
	risco_analise	distributividade	Fator relacionado à distribuição
	risco_analise	exposicao	Cálculo da exposição ao risco
	risco_analise	fator	Fator de risco
	risco_analise	impacto	Medida ou consequência de risco
	risco_analise	probabilidade	Probabilidade de um risco ocorrer
	risco_controle	id	Controle e aprendizagem sobre o risco, verificando se ocorreu e se tratamento ou contenção foram eficazes.
	risco_controle	contencao_aplicada	Atividades de contenção foram executadas quando risco se tornou um problema?
	risco_controle	contencao_eficaz	Medidas tomadas contra o risco que se tornou problema foram eficazes?
	risco_controle	risco_ocorreu	O risco ocorreu?
	risco_controle	tratamento_eficaz	O tratamento ao risco foi eficaz?
	risco_projeto	id	Risco relacionado a um projeto.
	risco_projeto	restrito_local	Risco é restrito ao local?
	risco_projeto	risco_ocorreu	O risco ocorreu?
	risco_projeto	tratamento_eficaz	O tratamento ao risco foi eficaz?
	risco_projeto_contencao	descricao	Atividades para reduzir o impacto de um risco em um projeto, por meio de uma ação planejada.
	risco_projeto_tratamento	id	Escolha de tratamento preventivo ou corretivo para o risco de um projeto.
	risco_projeto_tratamento	atividades	Atividades e tarefas para tratar o risco.
	risco_projeto_tratamento	documentos	Documentos de referência sobre tratamento de riscos.
	risco_projeto_tratamento	recursos	Recursos de tratamento e sua alocação.
risco_tratamento	id	Escolha de tratamento preventivo ou corretivo para o risco.	
risco_tratamento	tratamento	Tipo de tratamento	

Para melhor compreensão do significado dos dados armazenados no banco, são listados a seguir possíveis valores para alguns atributos.

Quadro 19 – Possíveis valores dos Dados do DiSEN - Área de assunto Comum

		Tabela	Atributo	Possíveis valores
Área de assunto Comum	artefato	arquivo	estado	Modificado, incluído, excluído, alocado, entre outros
		artefato	estado	Válido, inválido, excluído
		artefato	nome	diagramas, modelos, manuais, códigos fonte, códigos objeto
		tipo_artefato	descricao	diagrama de use-case, um diagrama de seqüência, documento, camadas do sistema, descrição das funcionalidades, diagrama de classes, diagrama de pacotes, diagrama de seqüência, lista de use-cases e atores
	local	local	padrao	Verdadeiro, falso
	processo	atividade_processo	nome	Levantamento de funcionalidades com os usuários, elaborar modelo de domínio, elaborar diagrama de use-case inicial
		dependencia_atividade_processo	dados	Verdadeiro, falso
		dependencia_atividade_processo	sequencia	Verdadeiro, falso
		processo	nome	RUP, CATALYSIS
	projeto	atividade_projeto	descricao	levantar funcionalidades
		atividade_projeto	estado	Em planejamento, em andamento e atribuída, em andamento e para antecipar, em andamento e para executar, em andamento e antecipando, em andamento e executando, em andamento e abortada, suspensa, cancelada e concluída.
		dependencia_atividade_projeto	dados	Verdadeiro, falso
		dependencia_atividade_projeto	sequencia	Verdadeiro, falso
		fase_projeto	estado	Em planejamento, em andamento, suspensa, cancelado e concluído
		projeto	estado	Em planejamento, em andamento, suspensa, cancelado e concluído
		tarefa	estado	Aguardando recurso, em andamento e para antecipar, em andamento e para executar, em andamento e antecipando, em andamento e executando, em andamento e abortada, suspensa, cancelada e concluída.
	reunião	vimee_participante_reuniao	estadoparticipantereuniao	Aguardando confirmação, Presença confirmada, Presença rejeitada, Presente, Ausente
		vimee_participante_reuniao	tipoparticipantereuniao	Secretário, solicitante, mediador, ouvinte, participante
		vimee_reuniao	descricao	Aquisição de tecnologia, definição do escopo do projeto A
		vimee_reuniao	estadoreuniao	Agendada, em andamento, adiada, suspensa, concluída, cancelada
vimee_secao		estadosecao	Aberta, suspensa, adiada, concluída	
vimee_tipo_reuniao		descricao	Reunião estratégica, acompanhamento de projetos, entre outros.	

Quadro 20 – Possíveis valores dos Dados do DiSEN - Área de assunto Qualidade

	Tabela	Atributo	Possíveis valores
Área de assunto Qualidade problema	tipo_log	nome	Técnico, organizacional

Quadro 21 – Possíveis valores dos Dados do DiSEN - Área de assunto Recurso

	Tabela	Atributo	Possíveis valores	
Área de assunto Recurso	recurso	recurso	estado	Disponível, indisponível
		recurso	type	Recurso humano, recurso material
	recurso humano	afinidade_usuario	nivel	0, 1, 2, 3, 4, 5
		habilidade	nome	Linguagem de programação, análise de requisitos
		habilidade_usuario	nivel	pouca, razoável, boa, ótima
		participante	estado	Disponível, indisponível
		participante	gerente	Verdadeiro, falso
		treinamento_usuario	nivel	pouco, razoável, bom, ótimo
		usuario	excluido	Verdadeiro, falso
	recurso material	recurso_material	descricao	Ferramenta CASE, IDE de desenvolvimento, ferramenta de modelagem UML, computador, notebook, papel, caneta, plotter, projetor, ADSL, entre outros.
		tipo_recurso_material	nome	Hardware, software, material de escritório consumível, rede, acesso à Internet, linha telefônica, entre outros.

Quadro 22 – Possíveis valores dos Dados do DiSEN - Área de assunto Risco

		Tabela	Atributo	Possíveis valores
Área de assunto Risco	risco	risco	risco_escopo	Técnico, organizacional
		risco	risco_negocios	Verdadeiro, falso
		risco	risco_produto	Verdadeiro, falso
		risco	risco_projeto	Verdadeiro, falso
		taxonomia1	descricao	Pessoas, Tecnologia, Processo, Ambiental
		taxonomia2	descricao	Pessoas (Clientes, usuários finais, financiadores, stakeholders, funcionários, organização, habilidades, política, moral), Tecnologia (Segurança, Ambiente de desenvolvimento e teste, Ferramentas, Entrega, Suporte, Ambiente operacional, Disponibilidade), Processo (Missão e objetivos, Tomada de decisão, Características do projeto, Orçamento e custo e cronograma, Requisitos, Design, Construção, Teste), Ambiental (Legal, Regulador, Competição, Economia, Tecnologia, Negócios)
	risco do projeto	risco_analise	distributividade	Distância percebida, cultura nacional, metodologia de sistemas, estrutura de tarefa, distância organizacional
		risco_analise	exposicao	impacto X probabilidade
		risco_analise	fator	Ambiente corporativo, patrocínio / propriedade, gerenciamento de relacionamento, gerenciamento e planejamento de projeto, escopo, requisitos, cronograma, processos de desenvolvimento, pessoal e equipe, tecnologia, dependências externas
		risco_analise	impacto	percentual
		risco_analise	probabilidade	percentual
		risco_controle	contencao_aplicada	Verdadeiro, falso
		risco_controle	contencao_eficaz	Verdadeiro, falso
		risco_controle	risco_ocorreu	Verdadeiro, falso
risco_projeto		tratamento_eficaz	Verdadeiro, falso	
risco_projeto		estado_risco_projeto	Pode ocorrer, ocorreu, não ocorreu, eliminado	
risco_projeto	restrito_local	Verdadeiro, falso		
risco_projeto	risco_ocorreu	Verdadeiro, falso		
risco_projeto	tratamento_eficaz	Verdadeiro, falso		
risco_tratamento	tratamento	Pesquisar, aceitar, evitar, transferir, mitigar, conter		