

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
CURSO DE GRADUAÇÃO EM CIÊNCIA DA
COMPUTAÇÃO**

Ivan Ricardo Schuster

**Desenvolvimento de um suporte temporal de dados
para uma aplicação de curva de carga**

Trabalho de Conclusão de Curso

**Prof. Ronaldo dos Santos Mello, Dr.
Orientador**

Florianópolis, Maio de 2004

Desenvolvimento de um suporte temporal de dados para uma aplicação de curva de carga

Ivan Ricardo Schuster

Este Trabalho de Conclusão de Curso foi aprovado em sua forma final pelo Curso de em
Ciência da Computação da Universidade Federal de Santa Catarina.

Prof. Ronaldo dos Santos Mello, Dr.

Orientador

Prof. José Mazzucco Jr., Dr.

Coordenador do Curso

Banca Examinadora

Rubem Guimarães Dias Neto

Prof. Mário Dantas, PhD.

Agradecimentos

Agradeço a todas as pessoas que me ajudaram direta ou indiretamente na realização deste trabalho e que talvez não tenham sido citados aqui mas que sabem que o agradecimento deve ser estendido a eles.

Agradeço aos meus amigos pela ajuda, nem sempre na forma de conhecimento técnico, mas também pelas festas e por tudo de bom que aconteceu desde que vim morar em Floripa.

Um agradecimento especial à família 992, minha turma que permaneceu unida durante estes 5 anos de curso.

Aos Jailsons, que sempre que eu posso, combino umas festas em Blumenau.

Ao pessoal da Airgate, principalmente a Sheila, que sempre se prontificou a revisar o texto antes de eu enviá-lo para a banca, ao Rubem, que foi colocado no fogo quando convidei para fazer parte da banca, e também ao Carlucci, que me deu várias dicas de \LaTeX .

Ao meu pai e minha mãe, que sempre quebram o galho quando preciso. A Suzana, minha irmã que me aguentou por cinco anos morando com ela (e ainda aguenta!).

Agradeço também ao meu orientador, Ronaldo dos Santos Mello, pela sua orientação e confiança de que eu faria um bom trabalho e que muitas vezes deixou tudo de lado para dar uma atenção de última hora ao meu TCC.

Sumário

| | |
|---------------------------------------------------------|------------|
| Lista de Figuras | vii |
| Lista de Tabelas | ix |
| Lista de Símbolos | x |
| Resumo | xi |
| Abstract | xii |
| 1 Introdução | 1 |
| 1.1 Objetivos | 2 |
| 1.1.1 Objetivo Geral | 2 |
| 1.1.2 Objetivos Específicos | 2 |
| 1.2 Materiais e Métodos | 3 |
| 1.3 Motivação e Justificativa | 4 |
| 1.4 Trabalhos Correlacionados | 4 |
| 1.5 Organização do Texto | 5 |
| 2 Conceitos Básicos de Bancos de Dados Temporais | 6 |
| 2.1 Introdução | 6 |
| 2.2 Dimensão Temporal | 7 |
| 2.3 Ordem no Tempo | 7 |
| 2.4 Tempo Absoluto e Tempo Relativo | 8 |
| 2.5 Variação Temporal | 8 |

| | | |
|----------|---------------------------------------------------------------------|-----------|
| 2.6 | Granularidade Temporal | 10 |
| 2.7 | Rótulos Temporais | 10 |
| 2.7.1 | Instante no Tempo | 11 |
| 2.7.2 | Intervalo Temporal | 11 |
| 2.7.3 | Elemento Temporal | 11 |
| 2.8 | Representação Temporal Explícita e Implícita | 12 |
| 3 | Bancos de Dados Temporais | 14 |
| 3.1 | Introdução | 14 |
| 3.2 | Bancos de dados instantâneos | 14 |
| 3.3 | Bancos de Dados de Tempo de Transação | 15 |
| 3.4 | Bancos de Dados de Tempo de Validade | 16 |
| 3.5 | Bancos de Dados Bitemporais | 16 |
| 4 | Formas de armazenamento de dados temporais em BD relacionais | 19 |
| 4.1 | Introdução | 19 |
| 4.2 | Relação Instantânea/Temporal | 19 |
| 4.3 | Relação Instantânea e Temporal | 20 |
| 4.4 | Relação Temporal Delta | 20 |
| 5 | Aplicação de Curva de Carga | 22 |
| 5.1 | Introdução | 22 |
| 5.2 | Sistema | 23 |
| 5.2.1 | Coleta de dados | 23 |
| 5.2.2 | Administração | 24 |
| 5.2.3 | Banco de dados | 24 |
| 5.2.4 | Web | 25 |
| 5.3 | Proposta | 26 |
| 6 | Tecnologias Utilizadas | 27 |
| 6.1 | Introdução | 27 |

| | | |
|----------|-------------------------------------------------------------|-----------|
| 6.2 | Banco de Dados | 27 |
| 6.2.1 | Sistema de Gerenciamento de Banco de Dados Oracle | 27 |
| 6.2.2 | PL/SQL developer | 28 |
| 6.3 | Páginas WEB | 28 |
| 6.3.1 | IIS | 28 |
| 6.3.2 | ASP | 28 |
| 6.4 | Conclusão | 29 |
| 7 | Sistema Proposto | 30 |
| 7.1 | Introdução | 30 |
| 7.2 | Análise | 30 |
| 7.2.1 | Funcionalidades | 31 |
| 7.3 | Projeto | 31 |
| 7.3.1 | Modelagem dos dados | 32 |
| 7.3.2 | Manipulação dos dados | 35 |
| 7.3.3 | Caso Real | 37 |
| 7.3.4 | Consultas temporais | 40 |
| 7.3.5 | Escalabilidade | 42 |
| 7.4 | Conclusão | 42 |
| 8 | Considerações Finais | 43 |
| 8.1 | Sugestões para Trabalhos Futuros | 44 |
| | Referências Bibliográficas | 45 |
| A | Fontes | 46 |

Lista de Figuras

| | | |
|-----|--------------------------------------------------------------|----|
| 2.1 | Exemplo de dado convencional | 7 |
| 2.2 | Exemplo de atributo temporal em uma tabela | 8 |
| 2.3 | Varição por escala | 9 |
| 2.4 | Varição ponto a ponto | 9 |
| 2.5 | Varição definida por função | 10 |
| 2.6 | Reencarnação sem elemento temporal | 12 |
| 2.7 | Reencarnação com intervalo temporal | 12 |
| 3.1 | Banco de dados instantâneo [EDE 94] | 14 |
| 3.2 | Banco de dados de tempo de transação [EDE 94] | 15 |
| 3.3 | Banco de dados de tempo de transação [EDE 94] | 16 |
| 3.4 | Banco de dados de tempo de validade [EDE 94] | 17 |
| 3.5 | Banco de dados bitemporal [EDE 94] | 18 |
| 4.1 | Relação Instantânea/Temporal [MEL 03] | 20 |
| 4.2 | Relação Instantânea e Temporal [MEL 03] | 20 |
| 4.3 | Relação Temporal Delta [MEL 03] | 21 |
| 5.1 | Tabelas de coleta de dados | 25 |
| 5.2 | Energia entregue ou recebida | 26 |
| 7.1 | Uma tabela genérica a ser adaptada | 32 |
| 7.2 | Modelagem de dados temporal para a tabela genérica | 34 |

| | | |
|-----|-----------------------------------------------------------------------------------------|----|
| 7.3 | Modelagem de dados da tabela genérica aplicada a um banco de dados relacional | 34 |
| 7.4 | Tabelas com suporte temporal geradas a partir da tabela genérica | 35 |
| 7.5 | Aplicação da procedure sobre os dados temporais | 36 |
| 7.6 | Consulta à view | 37 |

Lista de Tabelas

| | | |
|-----|-----------------------------------------------|----|
| 5.1 | Campos da tabela de energia | 24 |
| 7.1 | Dados inseridos por tabela temporal | 39 |

Lista de Siglas

| | |
|-------|--------------------------------------------|
| ASP | Active Server Pages |
| BD | Banco de Dados |
| BDR | Banco de Dados Relacional |
| BDT | Banco de Dados Temporal |
| IIS | Internet Information Server |
| MS | Microsoft |
| SBC | Sociedade Brasileira de Computação |
| SGBD | Sistema de Gerenciamento de Banco de Dados |
| SQL | Structured Query Language |
| UFRGS | Universidade Federal do Rio Grande do Sul |
| WEB | World Wide Web |

Resumo

Aplicações de coleta de dados de energia geram grande quantidade de dados e estes, por serem inseridos em intervalo de tempo fixo, muitas vezes permanecem constantes por algum tempo, tornando os valores redundantes. Uma forma de reduzir a constância destes dados é aplicar métodos temporais à estrutura das tabelas de armazenamento. Com elas, pode-se eliminar redundâncias e permitir que consultas temporais sejam realizadas com menor custo. A idéia do trabalho é realizar um estudo da viabilidade da construção de um suporte temporal a um sistema já existente de coleta de dados de energia.

Palavras-chave: Banco de Dados Temporal, Bancos de Dados não Convencionais, Relação Temporal Delta.

Abstract

Colocar aqui o abstract

Capítulo 1

Introdução

Uma das maiores preocupações dos projetistas de sistemas que trabalham com dados sendo inseridos constantemente é a quantidade de bytes que esta aplicação vai gerar com o passar do tempo. Os sistemas que se utilizam de dados históricos precisam ter discos com imensa capacidade para guardar tanta informação.

Sistemas que coletam dados de forma periódica podem gerar dados redundantes, ou seja, valores constantes que a cada intervalo de tempo são registrados no banco de dados.

Um exemplo são sistemas que lidam com equipamentos de medição que, habitualmente, geram dados de forma temporal, em intervalos de tempo específicos. Estes dados podem ser apresentados de minuto em minuto, de hora em hora, etc. Muitas vezes, são valores constantes e não há a necessidade de inseri-los novamente no banco de dados. Um exemplo disso é a medição de energia.

Supondo um determinado medidor eletrônico que registra o consumo de energia, valores de tensão e corrente em intervalos fixos de 1 minuto, o sistema (inclui-se aí o medidor) não tem a capacidade de avaliar se o valor do dado que está sendo gerado neste instante é o mesmo que foi inserido há 5 minutos.

Diante do que foi explanado acima, percebe-se a importância da utilização de sistemas com técnicas que visem a otimização do armazenamento das informações, excluindo redundâncias e armazenando um dado novo somente se for diferente do dado

antigo, para minimizar o espaço em disco.

1.1 Objetivos

1.1.1 Objetivo Geral

O objetivo geral desse TCC é realizar um estudo sobre BDs temporais, seus benefícios e sua performance, verificando a possibilidade da utilização destas técnicas em aplicações que envolvam medição de energia elétrica e outros sistemas que colem dados periodicamente e necessitem de uma análise sobre os dados que estão sendo inseridos.

1.1.2 Objetivos Específicos

Os objetivos específicos deste TCC são:

1. Demonstrar as vantagens da utilização da abordagem de bancos de dados temporais (BDTs);
2. Levantar a importância da aplicação dos BDTs em sistemas que avaliam a medição de energia de grandes consumidores;
3. Estudar formas de armazenamento temporal e quais delas se adaptariam melhor ao sistema existente;
4. Traçar um comparativo entre a modelagem temporal e a relacional, usada atualmente;
5. Realizar um projeto de BDT para a aplicação de curva de carga com implementação em um BDR;
6. Desenvolver uma extensão desta aplicação, que faça o gerenciamento de dados temporais em um BDR.

1.2 Materiais e Métodos

O trabalho será desenvolvido nos moldes do Sistema de Curva de Carga, sistema desenvolvido pela Airgate [AIR 03] e os materiais utilizados são, basicamente, os mesmos utilizados no sistema atualmente em produção.

O hardware utilizado no sistema pode ser uma única máquina que sirva Banco de Dados Oracle, servidor de paginas ASP e cliente.

Na implementação de teste realizada no desenvolvimento deste trabalho foram utilizados os seguintes recursos:

Hardware:

- Máquina 1: Computador servidor de banco de dados. No caso prático foi utilizado um Intel Xeon 2.4 GHz, 2 GB de RAM, com sistema operacional Linux Red Hat 9.0
- Máquina 2: Computador servidor de paginas ASP. No caso prático foi utilizada uma máquina Intel Pentium 833 MHz, rodando em sistema operacional Windows 2000 Server.
- Máquina 3: Computador cliente, que pode ser o mesmo computador servidor. No caso prático foi utilizado um Intel Pentium III 750 MHz com sistema operacional Windows 2000 Client instalado.

Software:

- Oracle 9i Server, rodando na máquina 1, com razoável espaço em disco livre;
- Microsoft Internet Information Server com suporte para ASP, na máquina 2;
- Ainda na máquina 2, devemos ter instalado um cliente Oracle, para que o mesmo faça a integração das páginas com o BD;
- Browser Internet Explorer ou compatível na máquina 3.

1.3 Motivação e Justificativa

Uma barreira cultural está criada quando o assunto é BDs. A frase “a propaganda é a alma do negócio” descreve exatamente o que acontece nesse meio. Talvez por força do departamento de marketing de empresas do setor, hoje em dia quando se pensa em BDs, remete-se geralmente aos relacionais. Mas, como cada caso é um caso, muitas aplicações são desenvolvidas com técnicas impróprias para seu objetivo.

Não se pretende aqui avaliar se as abordagens de BD são boas ou ruins, mas faz-se imperativo frisar que algumas são mais apropriadas que outras para determinadas aplicações. O que se pretende com este TCC é aplicar novas técnicas de BDs, muitos deles específicos para determinadas aplicações, possibilitando assim uma formação mais aberta para novas tecnologias e metodologias mais variadas.

Um domínio de aplicação que necessita controlar dados históricos, como o sistema de curva de carga, juntamente com a teoria sobre BD temporal, fornece os instrumentos necessários para o desenvolvimento de um módulo que preencha os requisitos temporais, aliando performance e funcionalidade.

1.4 Trabalhos Correlacionados

Alguns poucos trabalhos podem ser encontrados sobre Bancos de Dados Temporais. A técnica não é nova, mas até hoje é muito pouco utilizada pelos sistemas de computação. A professora Nina Edelweiss da UFRGS [EDE 94], [EDE 98] já escreveu muito sobre este tipo de tecnologia e mantém um grupo de estudos juntamente com o prof. Clesio Saraiva dos Santos. O grupo, “Versions and Time in Databases Research Group” estuda controle de versões e bancos de dados temporais.

Além disso, pode-se encontrar projetos de sistemas de gerenciamento de BDT em alguns sites como o Tiger [TIG 04] da universidade Aalborg, na Alemanha ou o TimeConsult [Tim 04], da Suíça, mas seus projetos foram abandonados e as versões mais atuais são do final da década de 90.

Outros projetos de BDT podem ser encontrados no endereço <http://www.db.informatik.uni->

rostock.de/bg/tdb_res.html.

1.5 Organização do Texto

Neste capítulo foi apresentada uma introdução do presente trabalho, os recursos utilizados para a implementação, os motivos para que fosse realizado este estudo e uma visão geral de como está atualmente o estudo de BDT no Brasil e no mundo.

No capítulo 2 são apresentados alguns conceitos básicos de BDT, conceitos estes que foram apresentados no Congresso Nacional da Sociedade Brasileira da Computação, em 1998 e permite que o leitor tenha de forma resumida algumas técnicas utilizadas na implementação.

O capítulo 3 traz uma breve apresentação dos quatro tipos de bancos de dados temporais, passando no quarto capítulo por exemplos de algumas formas de armazenamento de dados temporais em BDR.

No capítulo 5 será apresentadas algumas das funcionalidades da aplicação de Curva de Carga, o sistema em estudo para o desenvolvimento do suporte temporal.

O Capítulo 6 apresenta uma descrição de todas as tecnologias utilizadas no desenvolvimento do Sistema. O objetivo do capítulo é o de apresentar ao leitor os conhecimentos específicos necessários de cada ferramenta.

Já no capítulo 7, o sistema proposto é apresentado, juntamente com sua análise e seu projeto. São apresentadas também as duas modelagens de dados, a antiga e a nova e por fim a descrição de seu funcionamento.

No capítulo 8 serão apresentadas as considerações finais do trabalho.

Capítulo 2

Conceitos Básicos de Bancos de Dados Temporais

2.1 Introdução

A maioria dos sistemas de armazenamento de dados utiliza informações temporais, na forma de valores temporais, restrições de validade ou características evolutivas e dados históricos [EDE 94]. Ao se construir uma modelagem de dados para um sistema, os aspectos temporais vêm à tona e deve-se definir quais dados devem ser temporais e determinar aspectos evolutivos no sistema.

O objetivo deste capítulo é dar uma base ao leitor de quais são os conceitos principais no que diz respeito ao armazenamento de dados temporais, para um completo entendimento do que será realizado neste trabalho.

Os conceitos aqui apresentados foram extraídos do artigo publicado no Congresso Nacional da Sociedade Brasileira de Computação em 1998 e podem ser encontrados por completo em [EDE 98].

2.2 Dimensão Temporal

Os BDs tradicionais são implementados em duas dimensões: as linhas de uma tabela e as colunas. Cada par linha/coluna apresenta um único valor. Não há representação desses dados na história. Como exemplo, pode-se tomar uma tabela que guarda informações de salário. Se o salário de um determinado funcionário for alterado, o valor anterior será substituído, conforme pode ser visto na figura 2.1:

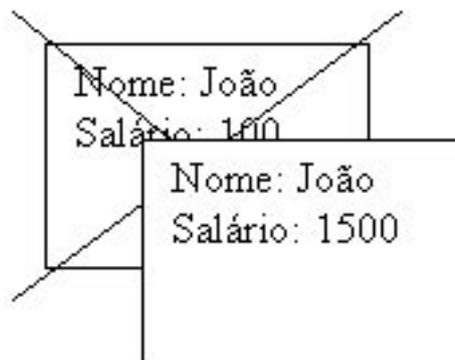


Figura 2.1: Exemplo de dado convencional

A proposta da modelagem temporal é fazer uma alteração nesta modelagem tradicional para incluir dados temporais, mantendo dados antigos ao invés de substituí-los. Com isso, cada dado novo possui um rótulo que o identifica no tempo (figura 2.2).

2.3 Ordem no Tempo

Uma seqüência ordenada de pontos compõe a dimensão temporal, sendo chamada de eixo temporal. Assim sendo, três abordagens de ordem no tempo podem ser apresentadas:

- Tempo linear: o mais comum é que se assuma que o tempo flui de forma linear e seqüencial. É o caso de uma tabela de logs, onde dados são inseridos de forma seqüencial.

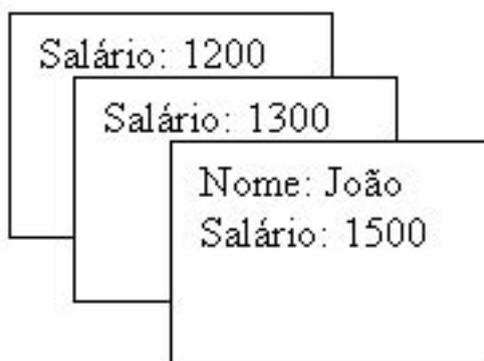


Figura 2.2: Exemplo de atributo temporal em uma tabela

- Tempo ramificado: nesta abordagem pode-se dizer que dois pontos são sucessores ou antecessores imediatos de um único. Um exemplo seria no desenvolvimento de hipóteses, como diferentes suposições sobre a história futura.
- Tempo circular: assume-se que a ordem do tempo é cíclica. Essa abordagem pode ser utilizada, por exemplo, quando se desconsidera a informação de dia, mês e ano e trabalha-se com eventos que sempre ocorrem em uma determinada hora do dia.

2.4 Tempo Absoluto e Tempo Relativo

A definição de tempo em uma abordagem temporal pode ser relativa a um determinado evento ou possuir um rótulo de tempo absoluto. Um exemplo de tempo relativo seria um evento que ocorreu 05 minutos após o desligamento de um determinado equipamento. Em um tempo absoluto poder-se-ia dizer que este evento ocorreu às 3 horas do dia 05/07/88.

2.5 Variação Temporal

Outro conceito importante é o que define de que forma o tempo varia num sistema. Esse tempo pode variar continuamente ou discretamente. Por natureza, o tempo é conti-

nuo, mas com pouca perda de precisão pode-se passar a considerar o tempo discreto. A representação discreta facilita consideravelmente a implementação de modelos de dados.

Modelos de dados que utilizam a noção discreta de variação temporal baseiam-se em uma linha de tempo em que o intervalo de tempo entre registros é fixo e esse intervalo de tempo é chamado de cronon.

Considerando-se uma variação discreta nas informações, o conceito pode ser definido de três maneiras: ponto a ponto, por escala ou por função.

Na variação por escala, o valor definido fica constante enquanto não ocorre uma alteração de valor, conforme pode ser visto na figura 2.3. Em uma variação ponto a

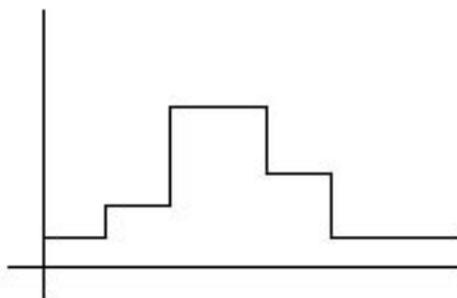


Figura 2.3: Variação por escala

ponto o valor vale somente para a estampa de tempo definida e não há informação entre um e outro ponto da escala de tempo (figura 2.4). Já na variação temporal definida por

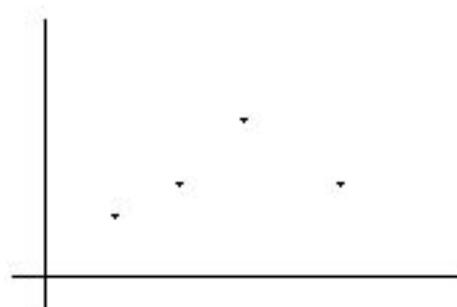


Figura 2.4: Variação ponto a ponto

função, existe uma função que interpola uma curva que define os valores de cada ponto no intervalo (figura 2.5). Essa função pode ser definida na modelagem conceitual ou pelo

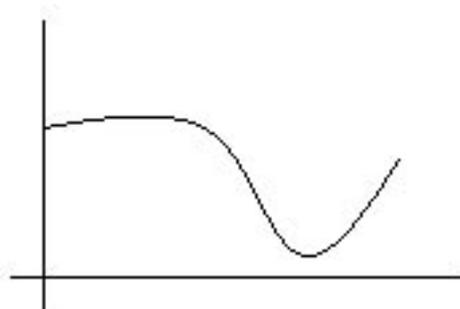


Figura 2.5: Variação definida por função

usuário.

2.6 Granularidade Temporal

A granularidade temporal basicamente é definida por um cronon. Exemplos de granularidade temporal são mês, dia ou minuto. Dependendo da aplicação pode-se necessitar uma granularidade mais fina (tempo mais detalhado) ou mais grossa (tempo menos detalhado). Entretanto esse cronon pode variar de acordo com várias características e pode ser ajustado para uma melhor representação dos dados em uma tabela. Enquanto em um segmento os dados são inseridos em intervalos de 5 minutos, outro segmento pode apresentar a necessidade de uma precisão de 1 minuto.

2.7 Rótulos Temporais

Os rótulos temporais são formas de definição de atributos temporais. São eles:

2.7.1 Instante no Tempo

O instante no tempo é o mínimo valor que pode ser atribuído a um intervalo de tempo no sistema. Se o tempo for contínuo, o instante de tempo tem duração infinitesimal. Caso a linha de tempo tenha sido definida como discreta, o instante é definido como um cronon, e neste caso os instantes serão isomórficos aos números inteiros.

Considerando a ordem de variação no tempo linear, tem-se um instante especial, que é chamado de instante atual (now) e é a referência para que os dados sejam considerados passados ou futuros.

2.7.2 Intervalo Temporal

Intervalo temporal é o tempo decorrido entre dois instantes de tempo não necessariamente consecutivos.

O intervalo temporal é simbolizado por $[t_1, t_2]$ onde t_1 é o instante inicial e t_2 é o instante final do intervalo, mas o próprio eixo temporal T pode ser considerado um intervalo temporal representado por $[\ll, \gg]$, onde o símbolo \ll representa o início da contagem do tempo e \gg representa o final. Para que um intervalo temporal seja válido, é necessário que t_1 seja um instante de tempo menor ou igual a t_2 , e no segundo caso, este intervalo tem duração de um cronon. Um exemplo de intervalo temporal é o tempo de permanência de um veículo em um estacionamento, em que o veículo entrou as 09 hs e saiu as 12 hs. Este intervalo de tempo pode ser simbolizado por $[9:00, 12:00]$.

2.7.3 Elemento Temporal

Um elemento temporal é um conjunto finito de intervalos temporais, sendo formado tanto por um único intervalo como pela união de vários deles. Pode-se utilizar elementos da teoria dos conjuntos em sua representação, como operações de união, intersecção e complemento. Exemplos de elementos temporais são $[56,59]$ ou $[10,85] \cup [142,162]$ ou $[147,269] \cap [185,185]$.

Em termos de modelagem, o elemento temporal apresenta vantagens em relação

ao intervalo temporal. Utilizando o intervalo temporal na definição de rótulos temporais, pode haver fragmentação de dados em várias tuplas, pois esta primitiva não permite a representação de “reencarnação” de objetos (figura 2.6).

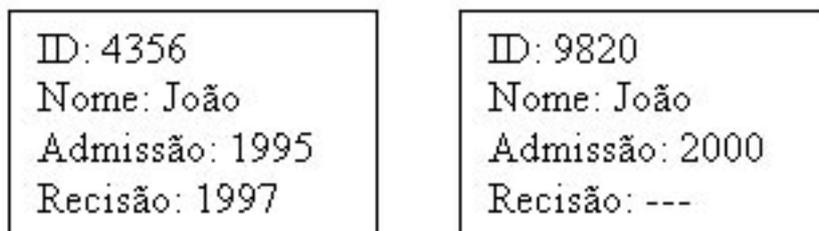


Figura 2.6: Reencarnação sem elemento temporal

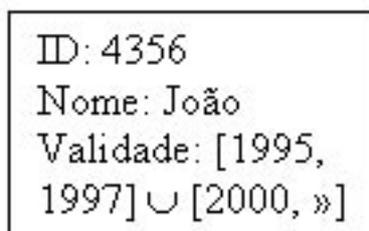


Figura 2.7: Reencarnação com intervalo temporal

Um exemplo de reencarnação é o de uma pessoa que trabalhou em uma determinada empresa entre 1995 e 1997 e foi readmitida em 2000. A validade deste objeto pode ser determinada pelo rótulo de tempo na forma de elemento temporal, como [1995,1997] ∪ [2000, »]. Na figura 2.7 pode-se visualizar a vantagem da utilização do elemento temporal.

2.8 Representação Temporal Explícita e Implícita

A definição de tempo pode ser simbolizada de forma explícita, usando a associação de instantes de tempo ou de forma implícita, com o uso de algum tipo de linguagem de

lógica temporal. Na forma implícita pode-se citar como exemplo a afirmação “a aula de lógica ocorreu ontem”. Na representação explícita podemos dizer que “a aula de lógica ocorreu no dia 5 de março de 2001”

Capítulo 3

Bancos de Dados Temporais

3.1 Introdução

Um BD temporal é todo BD que permite o armazenamento e a apresentação de informações temporais de alguma forma. De acordo com a comunidade de BD temporais, conforme descrito em [EDE 94], pode-se classificar os BDs temporais em quatro tipos diferentes: BD instantâneo, de tempo de transação, de tempo de validade e bitemporais.

3.2 Bancos de dados instantâneos

Nos BDs convencionais, como os BDs relacionais, cada dado possui um único valor. Caso este dado for alterado, o valor anterior é perdido. Os valores das propriedades do estado atual do BD são os únicos valores disponíveis (Figura 3.1).

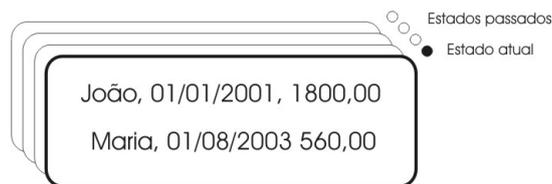


Figura 3.1: Banco de dados instantâneo [EDE 94]

Vários artifícios foram desenvolvidos para contornar este problema, como arqui-

vos de log que guardam estados passados, mas estes arquivos não podem ser facilmente consultados pelo processador de consultas do BD.

3.3 Bancos de Dados de Tempo de Transação

Para contornar este problema, uma forma de adaptação sugerida seria atribuir um rótulo de tempo (timestamp) para cada alteração que fosse feita nos atributos de uma tabela. Neste caso, se for alterado o valor de um determinado atributo, ao invés de apagar-se o valor antigo, é inserido o novo valor com o rótulo atual de tempo. A figura 3.2 representa algumas transações no banco de dados de tempo de transação. Cada atualização dos dados é associada ao dia em que ela foi feita e este valor torna-se válido até a realização de uma próxima atualização. Neste caso pode-se tanto fazer a consulta atual, buscando o valor atual do atributo, quanto consultar algum valor de estado passado sem muita dificuldade (Figura 3.2).

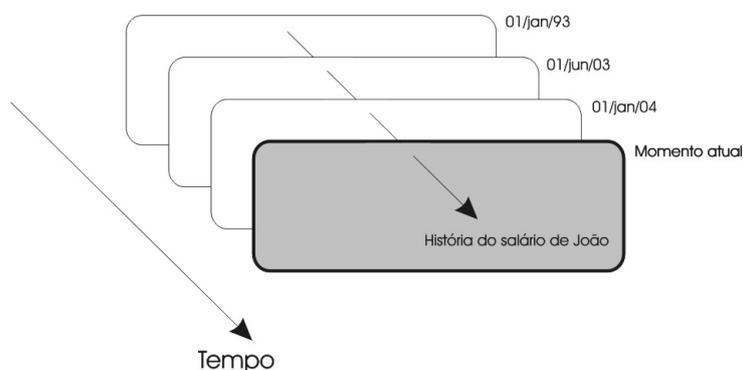


Figura 3.2: Banco de dados de tempo de transação [EDE 94]

Pode-se por exemplo, alterar o valor do salário de um determinado funcionário sem prejuízo na guarda do histórico de todos os salários do mesmo. Neste modelo, nenhum dado é apagado ou sobrescrito - o dado novo é inserido com nova estampa de tempo e passa a ter uma validade.

Como o próprio nome já diz, este BD lida com o conceito de tempo de transação, ou seja, cada modificação feita é registrada na forma de tuplas e identificada pelo times-

tamp da alteração, como se fosse um log (Figura 3.3).

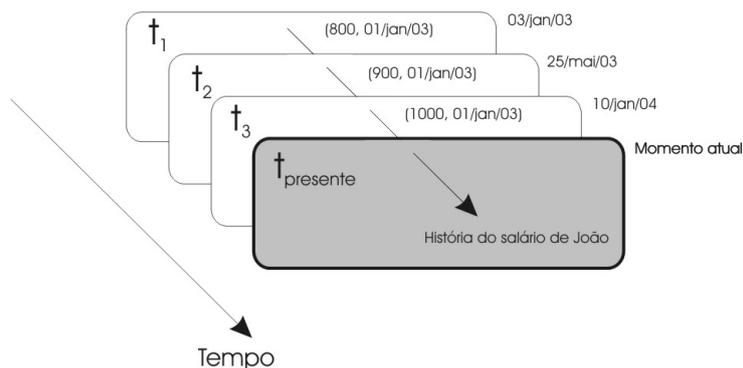


Figura 3.3: Banco de dados de tempo de transação [EDE 94]

3.4 Bancos de Dados de Tempo de Validade

Muitas vezes, a representação dos dados na forma de transações não é suficiente para que se tenha uma representação completa do histórico dos dados. Por vezes um dado pertence a um determinado período no tempo que não é mais válido no momento atual. Outra situação possível é a de dados que passarão a ter validade no futuro. Para estes casos utiliza-se o conceito de tempo de validade, onde cada valor pode ter definido o instante em que ele passou a se tornar verdadeiro e quando ele perdeu a validade, tendo ou não outro dado para substituí-lo. A figura 3.4 ilustra isto.

3.5 Bancos de Dados Bitemporais

Dos tipos de BDs, o BD bitemporal é o que armazena de forma mais completa os dados temporais. Nele, utiliza-se os conceitos de tempo de validade e tempo de transação ao mesmo tempo. Em um dado alterado, há tanto a informação do momento em que ele foi alterado no BD, quanto informações sobre a validade deste dado.

Neste tipo de modelagem, é possível voltar a um instante passado e saber o valor exato dos dados que acreditava-se que fossem válidos. Pode-se também voltar ao mesmo

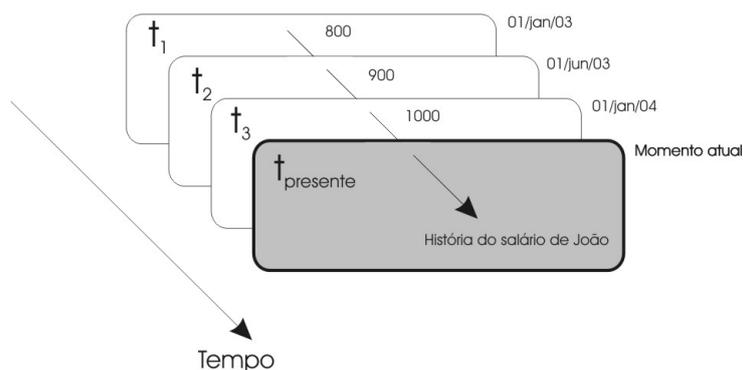


Figura 3.4: Banco de dados de tempo de validade [EDE 94]

instante passado e verificar os dados que realmente eram válidos naquele instante. Um exemplo prático é o de um funcionário que recebeu aumento no dia 01/01/04, mas, apesar de contar com seu salário reajustado desde o primeiro dia do mês, o funcionário encarregado de alterar o salário dos funcionários da empresa no sistema só o fez no dia 10. Neste caso, a validade do salário novo do funcionário é a partir do dia primeiro, mas esta informação só ficou disponível no dia 10 (Figura 3.5). Suponhamos que outro funcionário resolveu fazer o cálculo do custo que a empresa estava tendo com a folha de pagamento de seus funcionários, e o fez justamente no dia 04/01/04. Neste dia os dados ainda não estavam atualizados, portanto o cálculo feito não apresentava a realidade da empresa. Para justificar-se, o funcionário poderia facilmente no futuro consultar os dados tidos como verdadeiros no dia em que foi feita a pesquisa.

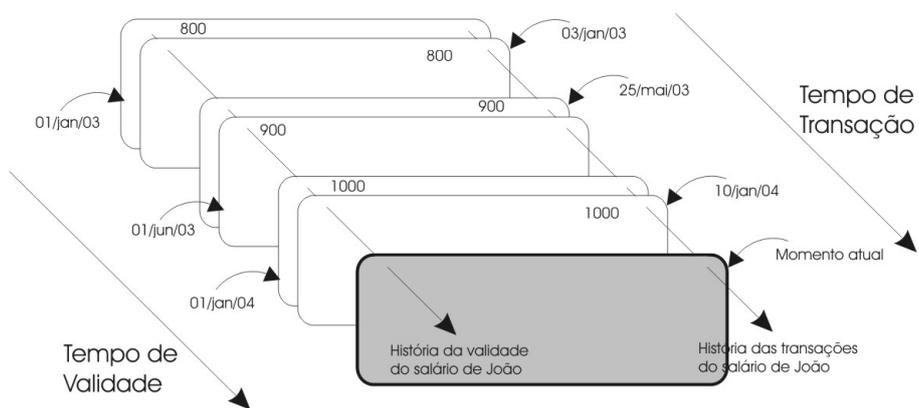


Figura 3.5: Banco de dados bitemporal [EDE 94]

Capítulo 4

Formas de armazenamento de dados temporais em BD relacionais

4.1 Introdução

Os BDs temporais foram e são alvo de muita discussão e estudo, mas pouca prática. Geralmente, suas implementações são bastante específicas e pouco genéricas. Muitos experimentos são apresentados na forma de protótipos, como TimeDB [Tim 04], Tiger [TIG 04], TIGUCAT [MEL 03] e TF-ORM [EDE 98].

Nos estudos realizados, algumas alternativas podem ser consideradas na implementação de um modelo de dados temporal em uma base de dados relacional [MEL 03]: Relação Instantânea/Temporal, Instantânea e Temporal ou Temporal Delta.

4.2 Relação Instantânea/Temporal

Esta relação mantém dados instantâneos e temporais em uma única relação. Esse tipo de implementação permite que o sistema possua um número reduzido de relações. Em compensação, há a possibilidade da existência de grande quantidade de dados redundantes, o que degrada o desempenho de consultas instantâneas. A figura 4.2 ilustra uma relação que mantém dados instantâneos e dados temporais.



Figura 4.1: Relação Instantânea/Temporal [MEL 03]

4.3 Relação Instantânea e Temporal

Nesta relação, os dados instantâneos e temporais são armazenados em relações separadas, melhorando a performance das consultas instantâneas. Neste caso, também temos grande quantidade de dados redundantes. Além disso, a quantidade de relações é maior. A figura 4.2 mostra este conceito.

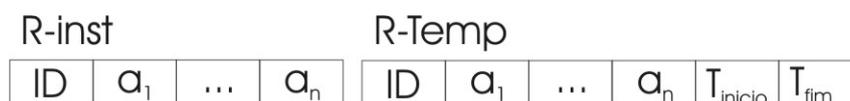


Figura 4.2: Relação Instantânea e Temporal [MEL 03]

4.4 Relação Temporal Delta

Na relação temporal delta, os dados de cada atributo são mantidos em relações temporais separadas, sendo definidas somente para os atributos que necessitam de suporte temporal. Com isso, evitamos a redundância existente nas propostas anteriores. A desvantagem é a dificuldade em consultar todos os atributos de um determinado dado em um certo tempo e a grande quantidade de relações. A figura 4.3 ilustra uma relação temporal do tipo Delta.

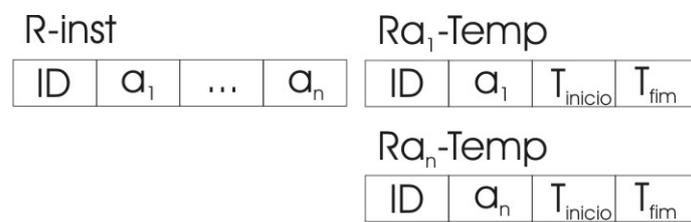


Figura 4.3: Relação Temporal Delta [MEL 03]

Capítulo 5

Aplicação de Curva de Carga

5.1 Introdução

Este capítulo descreve o domínio de aplicação para o qual este trabalho pretende desenvolver um suporte temporal.

A aplicação de Curva de Carga foi desenvolvida pela Airgate Telecomunicações Ltda [AIR 03], empresa em que trabalho atualmente, e consiste em coletar dados de medidores de energia elétrica, fazer uma análise de consistência desses dados e apresentá-los aos usuários na forma de gráficos, relatórios ou eventos. Este sistema é destinado principalmente às empresas de geração e de distribuição de energia elétrica, mas também pode ser aplicado às empresas que consomem muita energia e/ou que geram parte dela.

Pode-se tomar como exemplo uma empresa de distribuição de energia elétrica. Ela recebe a energia que é distribuída de empresas geradoras de energia. A cada ponto de fronteira entre geradora e distribuidora são instalados medidores que controlam o recebimento dessa energia. Por outro lado, o enfoque principal se dá nos clientes dessa distribuidora. Cada cliente (ou somente os maiores) tem um (ou mais) medidor(es) de energia instalado(s) e cada medidor é conectado via rede a um servidor que coleta estes dados e os armazena num BD relacional. Com o sistema, cada cliente pode verificar em tempo real qual o consumo atual, a energia gasta durante o mês/ano, fazer comparativos com meses/anos anteriores, etc. Além disso, pode-se verificar problemas ocorridos na

entrega de energia, distorções harmônicas e outros eventos [DIA 03].

Três módulos formam o sistema. São eles:

- Coleta de Dados
- Banco de Dados
- Web

Estes módulos serão explicados na próxima seção.

5.2 Sistema

O sistema é dividido em módulos que funcionam de forma independente e são ligados entre si pelo banco de dados. São eles a **Coleta de dados** e o módulo de **Administração**. No presente trabalho, o módulo administração será deixado de lado, por falta de dados interessantes ao propósito.

5.2.1 Coleta de dados

O módulo de Coleta de Dados foi desenvolvido utilizando uma ferramenta Scada que se encarrega de comunicar com medidores de energia instalados em diversos pontos geograficamente distribuídos.

Estes medidores são conectados ao servidor de coleta utilizando redes *Ethernet*, *RS 485* e/ou modem e se comunicam com o aplicativo *Scada* através de *drivers* específicos. Estes medidores são ligados à rede elétrica, recebem dados de **consumo, qualidade de energia, eventos** e outros. As informações são armazenadas em uma memória interna do medidor e ficam disponíveis para serem lidas pelo aplicativo *Scada*. Este aplicativo, por sua vez, lê as informações armazenadas em memória e joga os dados colhidos em tabelas do banco de dados. A tabela 5.1 mostra os campos e tipos de dados da tabela analisada.

| Campo | Tipo de dado |
|-----------|---------------|
| timestamp | data |
| cd_ponto | inteiro |
| eneat_del | <i>double</i> |
| eneat_rec | <i>double</i> |
| enere_del | <i>double</i> |
| enere_rec | <i>double</i> |
| iamean | <i>double</i> |
| ibmean | <i>double</i> |
| icmean | <i>double</i> |
| vamean | <i>double</i> |
| vbmean | <i>double</i> |
| vcmean | <i>double</i> |

Tabela 5.1: Campos da tabela de energia

5.2.2 Administração

O módulo de administração permite que os servidores de coleta de dados sejam configurados, os pontos de medição sejam cadastrados e seus dados sejam visualizados de na forma de gráficos e relatórios.

Este módulo foi desenvolvido basicamente em ASP e utiliza componentes ActiveX para a visualização dos gráficos.

O estudo deste módulo não será aprofundado, pois a alteração do banco de dados não alterará a forma com que este módulo realiza consultas.

5.2.3 Banco de dados

O módulo de BD é o “coração” do sistema. Este módulo guarda informações diversas, desde configurações e permissões de acesso de usuários até dados para geração de contas de energia elétrica.

Poder-se-ia usar páginas e mais páginas para descrever detalhadamente a estrutura do BD, mas o foco estará concentrado nas tabelas de coleta de dados (figura 5.1), tabelas estas que serão objeto de estudo e alvo de otimizações.

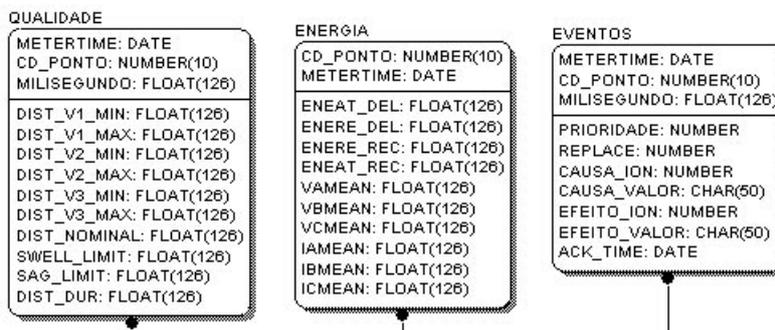


Figura 5.1: Tabelas de coleta de dados

A tabela de **energia** guarda informações de consumo de energia elétrica em intervalos discretos de 05 minutos. O período discreto desses dados interessa bastante, pois alterações estruturais podem ser efetuadas para que estes dados sejam otimizados. Além disso, outra característica interessa nesta tabela: quando há informação de energia entregue, não há informação de energia recebida. Isso acontece porque uma determinada empresa pode tanto receber energia de uma distribuidora como ceder energia (empresas com geradores próprios), e o dado recebido de um determinado intervalo é a totalização desse intervalo de tempo, conforme pode-se verificar na figura 5.2.

5.2.4 Web

O módulo Web, por sua vez, se encarrega de apresentar os dados colhidos durante as etapas anteriores na forma de gráficos, tabelas e formulários. Este módulo será pouco afetado pela modificação na modelagem dos dados, pois suas consultas são todas feitas utilizando-se *stored procedures*, e não há consultas diretas a tabelas.

Novas opções de consultas temporais serão disponibilizadas ao usuário através deste trabalho, visto que a forma com que estes dados serão armazenados em tabelas facilitará esses tipos de consulta.

| Meter time | Energia Ativa Entregue | Energia Reativa Entregue | Energia Ativa Recebida | Energia Reat. Recebida |
|-------------------|-------------------------------|---------------------------------|-------------------------------|-------------------------------|
| 14/2/2004 05:30 | 56425,5687 | 9857,3568 | 0 | 0 |
| 15/2/2004 05:35 | 56425,5682 | 9857,3501 | 0 | 0 |
| 16/2/2004 05:40 | 56425,4859 | 9857,3475 | 0 | 0 |
| 17/2/2004 05:45 | 56425,4581 | 9857,3421 | 0 | 0 |
| 18/2/2004 05:50 | 56425,4167 | 9857,3375 | 0 | 0 |
| 19/2/2004 05:55 | 2,3578 | 1,4145 | 0 | 0 |
| 20/2/2004 06:00 | 0 | 0 | 2457,6951 | 857,1457 |
| 21/2/2004 06:05 | 0 | 0 | 2457,6847 | 857,1141 |
| 22/2/2004 06:10 | 0 | 0 | 2457,6914 | 856,9414 |
| 23/2/2004 06:15 | 0 | 0 | 2457,6867 | 857,1094 |
| 24/2/2004 06:20 | 0 | 0 | 2457,6848 | 857,1072 |

Figura 5.2: Energia entregue ou recebida

5.3 Proposta

A proposta desse trabalho é unir a idéia de BDs Temporais com a situação apresentada na **aplicação de curva de carga**, tornando consultas temporais mais eficientes e utilizando menos espaço em disco para armazenar os dados sem perda de precisão.

Conforme viu-se no item 5.2, algumas tabelas de dados possuem características interessantes para o estudo do tema. Pode-se encontrar comportamentos harmônicos em intervalos de tempo e seqüências grandes de tuplas com dados nulos em colunas específicas.

Capítulo 6

Tecnologias Utilizadas

6.1 Introdução

Este capítulo apresenta as ferramentas e tecnologias utilizadas para o desenvolvimento do projeto, incluindo linguagens de programação, ferramentas e aplicativos, servidores Web e sistema de gerenciamento de banco de dados. Foi feito um estudo sobre cada uma das tecnologias e ferramentas empregadas procurando encontrar no histórico, vantagens e desvantagens, comparativos com outras tecnologias da mesma linha de funcionamento e o estado da arte, que representa o atual estado em que se encontra cada tecnologia.

6.2 Banco de Dados

6.2.1 Sistema de Gerenciamento de Banco de Dados Oracle

O *Oracle* [ORA 04] é um sistema de gerenciamento de banco de dados relacional completo, bastante robusto e utilizado em sistemas que necessitem de disponibilidade em tempo real, excessivas transações e armazenamento de grande volume de dados.

Ele possui vários recursos que ajudam tanto no desenvolvimento quanto na manutenção do BD, com suporte a *triggers*, *jobs* e outros recursos bastante úteis no desenvolvimento de *software*.

Este BD faz parte do **Sistema de Curva de Carga**, já instalado em alguns clientes da *Airgate* [AIR 03] e por este motivo foi o SGBD escolhido para o desenvolvimento do suporte temporal.

6.2.2 PL/SQL developer

O *PL/SQL Developer* é um ambiente de desenvolvimento e manutenção de objetos de banco de dados *Oracle*, que facilitou bastante a organização e o desenvolvimento das tabelas, *procedures* e *views* utilizadas no desenvolvimento do projeto.

Este ambiente foi desenvolvido pela *Allround Automations* [All 04] e é largamente utilizado por desenvolvedores de sistemas que utilizam BD *Oracle* e possui versões de demonstração disponíveis para *download* no site da empresa.

6.3 Páginas WEB

O ASP é um ambiente de programação por *scripts* que roda do lado do servidor

6.3.1 IIS

O IIS é um dos mais utilizados servidores de páginas WEB existentes no mercado. Por estar disponível na instalação do Microsoft Windows 2000, este aplicativo foi escolhido por conveniência.

Outro motivo para a escolha é a alta compatibilidade com o módulo ASP, que permite que sejam executados scripts de fácil programação.

6.3.2 ASP

O ASP é uma tecnologia capaz de processar scripts em um servidor da WEB,

6.4 Conclusão

Este capítulo apresentou uma descrição das principais tecnologias utilizadas no desenvolvimento deste trabalho, bem como suas funcionalidades.

Capítulo 7

Sistema Proposto

7.1 Introdução

A idéia do trabalho é alterar a forma com que os dados são inseridos nas tabelas do sistema de **curva de carga**, fazendo com que os dados utilizem um paradigma mais orientado ao tempo e menos relacional. Os dados continuam sendo inseridos em tabelas relacionais, mas agora todas as tabelas relacionais contém informações temporais, e um dado novo só será inserido caso ele seja diferente do dado antigo. Com isso, a idéia é diminuir o espaço em disco utilizado, pois dados repetidos serão armazenados uma única vez na tabela. Além disso, o sistema permitirá a realização de consultas temporais com mais facilidade.

7.2 Análise

Entre as funcionalidades do sistema, podemos citar:

- Modificação de tabelas para que determinados atributos se comportem como dados temporais;
- Criação de novas tabelas temporais;
- Funções para inserção e alteração destes dados de forma correta;

- Criação de *views* que listam os dados como se fossem tabelas relacionais comuns;
- Criação de consultas temporais.

Trata-se de um sistema (*software*) que altera o modo como os dados são inseridos no banco de dados relacional. Como o sistema atual utiliza *procedures* para inserção e alteração de dados e *views* para consultas, o sistema proposto pode utilizar os mesmos programas de inserção e consulta utilizados atualmente, pois a interface externa continuará sendo a mesma (utilizando-se *procedures* e *views*).

7.2.1 Funcionalidades

1. O sistema permite a criação de novas tabelas. Nelas podem ser definidos quais campos terão informações temporais.
2. Permite alterar uma tabela já existente, criando-se campos que passarão a ser armazenados de forma temporal.
3. Um grupo de rotinas permite inserir e alterar dados temporais nas tabelas criadas. Estas rotinas devem avaliar os dados válidos até o momento para então decidir as ações que serão tomadas para realizar a operação completa.
4. Consultas que simulam uma listagem na tabela original, com intervalos de 5 em 5 minutos e listando dados repetidos, dados estes que não serão armazenados na tabela temporal.

7.3 Projeto

O sistema proposto utiliza basicamente a linguagem *PL/SQL* compatível com *Oracle 9i* por estarem as tabelas implementadas em um banco de dados relacional. A escolha pelo *Oracle* confere grande flexibilidade e uma gama de recursos excelente, tanto em funções existentes quanto em documentação disponível na internet e na literatura.

Além disso a interface de criação e alteração do layout das tabelas é feito utilizando linguagem *ASP* e o servidor de páginas utiliza o *Internet Information Server*, da *Microsoft*.

7.3.1 Modelagem dos dados

Os dados de coleta do sistema geralmente seguem o seguinte padrão:

- Campos específicos com valores constantes (geralmente com valor 0)
- Valores constantes por algum tempo.
- Necessidade de buscas por valores críticos e quanto tempo eles ficaram neste patamar.

Analisando o padrão dos dados e dos requisitos do sistema, chega-se a conclusão que o melhor modelo para a adaptação relacional - temporal é a **relação temporal delta**, apresentada no item 4.4, pois cada campo da tabela segue um padrão único, e a constância de um campo não significa a constância de outro.

Deve-se então, criar uma tabela para cada coluna de dados coletados. Inicialmente será apresentada uma tabela genérica para melhor explicar a técnica desenvolvida. Essa tabela segue os padrões apresentados nas tabelas de medição de dados do sistema. A figura 7.1 ilustra a tabela de exemplo, ainda na forma convencional.

| tb_original | | | | |
|-------------|-----------------|--------|--------|--------|
| cd_ponto | timestamp | Valor1 | Valor2 | Valor3 |
| 1 | 20/5/2004 05:30 | 50 | 30 | 0 |
| 1 | 20/5/2004 05:36 | 51 | 30 | 0 |
| 1 | 20/5/2004 05:40 | 51 | 30 | 0 |
| 1 | 20/5/2004 05:45 | 51 | 29 | 0 |
| 1 | 20/5/2004 05:50 | 52 | 29 | 0 |
| 1 | 20/5/2004 05:55 | 51 | 30 | 0 |

Figura 7.1: Uma tabela genérica a ser adaptada

Esta tabela possui os seguintes campos:

- cd_ponto

- timestamp
- Valor1
- Valor2
- Valor3

O campo *cd_ponto* é a identificação do ponto de medição ao qual se refere o dado. Este campo não deve ser armazenado de forma temporal, pois faz parte da chave única da tabela, juntamente com o campo *timestamp*, que por sua vez identifica a data e o horário ao qual o dado se refere.

Os campos *Valor1*, *Valor2* e *Valor3* são dados de coleta, que no modelo temporal proposto serão inseridos em tabelas temporais específicas.

Os dados apresentados não são dados reais de leitura, mas seu comportamento é semelhante a estes, obtidos em medidores de energia. Pode-se perceber que os dados do campo *Valor3* permanecem constantes durante o intervalo apresentado. Esta característica é bastante comum, pois os medidores de energia utilizados pelo sistema coletam dados de consumo de entrada e de saída de energia elétrica. Como a maioria dos clientes somente consome energia do distribuidor, os campos que se referem a geração de energia permanecem nulos.

Tendo de modo simplificado a forma como estes dados são armazenados, pode-se criar uma modelagem temporal para reorganizar as informações de modo a otimizar o armazenamento e a consulta temporal aos dados. Esta modelagem pode ser vista na figura 7.2.

Nesta nova modelagem, o campo *timestamp* está implícito nos atributos de dados temporais. Esta informação deve ser armazenada no banco de dados temporal como um intervalo de validade, ou seja, com uma coluna indicando o início da validade e outra indicando o final.

A modelagem temporal da figura 7.2 aplicada a um banco de dados relacional resulta na estrutura relacional demonstrada na figura 7.3:

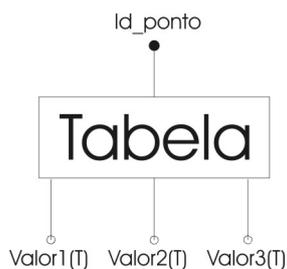


Figura 7.2: Modelagem de dados temporal para a tabela genérica

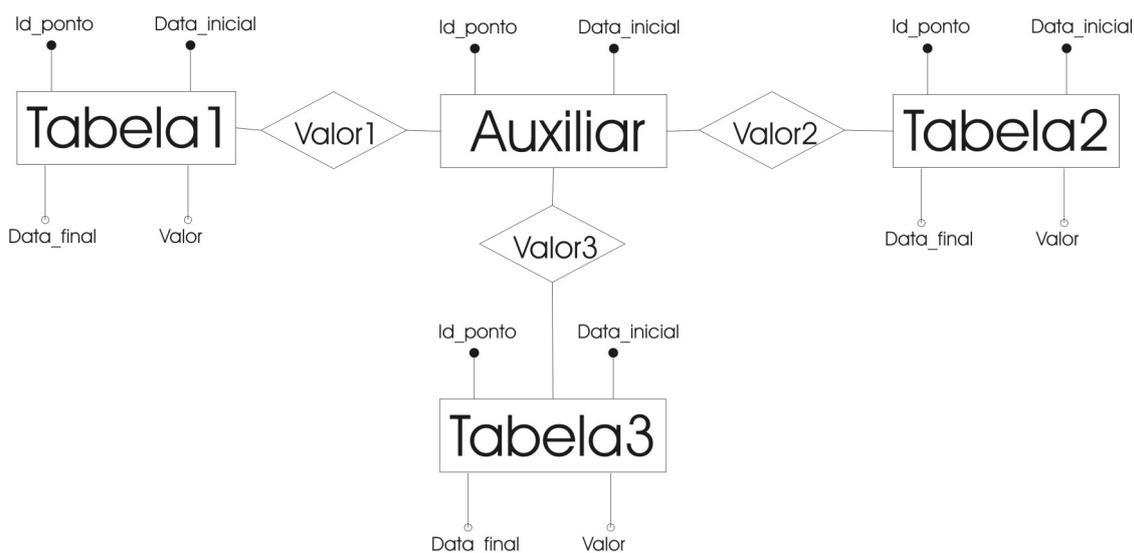


Figura 7.3: Modelagem de dados da tabela genérica aplicada a um banco de dados relacional

Uma tabela auxiliar deve ser criada para que os comandos SQL de consulta possam ser realizados sem perda de informações. A cada dado que é inserido com um timestamp novo, este timestamp é inserido na tabela para consultas futuras. Na figura 7.4 podemos visualizar os mesmos dados da tabela da figura 7.1, mas agora armazenados em tabelas com suporte temporal. Podemos verificar no exemplo que dados repetidos são inseridos uma única vez, retirando a redundância de dados.

| tb_aux | |
|----------|-----------------|
| cd_ponto | timestamp |
| 1 | 20/5/2004 05:30 |
| 1 | 20/5/2004 05:35 |
| 1 | 20/5/2004 05:40 |
| 1 | 20/5/2004 05:45 |
| 1 | 20/5/2004 05:50 |
| 1 | 20/5/2004 05:55 |

| tb_valor1 | | | |
|-----------|-----------------|-----------------|-------|
| cd_ponto | dt_inicial | dt_final | valor |
| 1 | 20/5/2004 05:30 | 20/5/2004 05:35 | 50 |
| 1 | 20/5/2004 05:35 | 20/5/2004 05:50 | 51 |
| 1 | 20/5/2004 05:50 | 20/5/2004 05:55 | 52 |
| 1 | 20/5/2004 05:55 | 20/5/2004 05:55 | 51 |

| tb_valor2 | | | |
|-----------|-----------------|-----------------|-------|
| cd_ponto | dt_inicial | dt_final | valor |
| 1 | 20/5/2004 05:30 | 20/5/2004 05:45 | 30 |
| 1 | 20/5/2004 05:45 | 20/5/2004 05:55 | 29 |
| 1 | 20/5/2004 05:55 | 20/5/2004 05:55 | 30 |

| tb_valor3 | | | |
|-----------|-----------------|-----------------|-------|
| cd_ponto | dt_inicial | dt_final | valor |
| 1 | 20/5/2004 05:30 | 20/5/2004 05:55 | 0 |

Figura 7.4: Tabelas com suporte temporal geradas a partir da tabela genérica

7.3.2 Manipulação dos dados

Com a alteração da modelagem dos dados, novas consultas devem ser criadas para substituir as já existentes. Além disso, outros métodos devem ser utilizados para executar ações como a inserção, remoção e atualização de um dado.

A idéia do projeto é manter a funcionalidade atual do sistema e incrementar novas funções. Atualmente, o sistema utiliza-se unicamente de *procedures* para fazer consultas e alteração de dados em tabelas. Com isso, certifica-se que a segurança dos dados não seja comprometida, além de disponibilizar total flexibilidade ao modo como estes dados são armazenados em disco.

A seguir são comentados os métodos implementados para dar o suporte temporal aos dados da aplicação.

7.3.2.1 Inserção

A inserção de um dado no modelo com suporte temporal segue o algoritmo abaixo:

```
insira data/hora na tabela auxiliar;
para cada coluna, faça
    selecione o valor verdadeiro para o timestamp a ser
    inserido;
```

```

se o valor constante na tabela for diferente do dado novo,
entao
    atualize o campo data/hora final, colocando o timestamp do
    dado a ser inserido;
    insira um novo registro na tabela, com data/hora inicial
    igual ao timestamp do dado e data/hora final igual ao fim
    da validade do dado anterior;
fim se;
fim para;

```

Para cada campo da tabela, haverá uma consulta para verificar se o dado a ser inserido é diferente do dado válido para aquele instante. Se for igual, não há a necessidade de inserir um novo registro, pois os dados constantes na tabela já satisfazem a situação real. Caso seja diferente, a validade do dado antigo é alterada para que seja inserido o novo dado, conforme ilustra a figura 7.5.

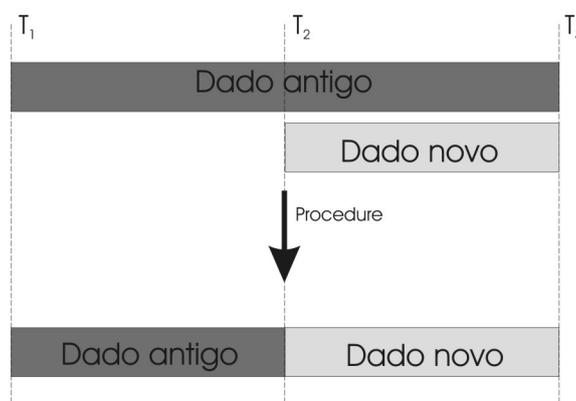


Figura 7.5: Aplicação da procedure sobre os dados temporais

7.3.2.2 Atualização

Na atualização de um dado temporal, pode-se utilizar o mesmo algoritmo proposto para a inserção, pois as duas ações se comportam de maneira semelhante, ou seja, uma atualização em um BD convencional corresponde a inclusão de um novo dado e uma nova validade em um BD temporal.

7.3.2.3 Consulta

Para possibilitar a consulta aos dados da mesma forma com que é feita no sistema atual, foi criada uma *view*. Esta *view* consulta todas as tabelas temporais pertencentes ao domínio dos dados da tabela consultada e retorna valores idênticos aos obtidos em consultas à tabela original, conforme visto na figura 7.6.

| | TIMESTAMP_INICIO | CD_FONTO | VALOR1 | VALOR2 | VALOR3 | VALOR4 | VALOR5 | VALOR6 | VALOR7 | VALOR8 |
|----|--------------------|----------|------------------|------------------|--------|--------|------------------|------------------|------------------|------------------|
| 1 | 15/3/2004 17:00:00 | 2 | 2172.58032226563 | 2411.30883789063 | 0 | 0 | 7975.392578125 | 7984.03759765625 | 7982.8857421875 | 1645.43103027344 |
| 2 | 15/3/2004 17:05:00 | 2 | 2136.11206054688 | 2330.89916992188 | 0 | 0 | 7987.13427734375 | 7995.73828125 | 7994.478515625 | 1645.43103027344 |
| 3 | 15/3/2004 17:10:00 | 2 | 2209.5539507813 | 2313.22689664063 | 0 | 0 | 7990.634765625 | 7999.8740234375 | 7998.20849609375 | 1645.43103027344 |
| 4 | 15/3/2004 17:15:00 | 2 | 2129.00809564963 | 2395.3639570313 | 0 | 0 | 7989.83349603375 | 7998.2527734375 | 7996.4760741875 | 1596.25842295156 |
| 5 | 15/3/2004 17:20:00 | 2 | 2203.59106445313 | 2308.94702148438 | 0 | 0 | 7983.591736875 | 7993.15234375 | 7991.3251953125 | 1596.25842295156 |
| 6 | 15/3/2004 17:25:00 | 2 | 2210.88916015625 | 2328.94482421875 | 0 | 0 | 7984.43072265625 | 7993.6688453125 | 7992.3115234375 | 1596.25842295156 |
| 7 | 15/3/2004 17:30:00 | 2 | 2209.68994140625 | 2338.09195273438 | 0 | 0 | 7982.7900390625 | 7991.94168453125 | 7990.6142578125 | 1618.09777832031 |
| 8 | 15/3/2004 17:35:00 | 2 | 2215.79149414063 | 2338.31884765625 | 0 | 0 | 7991.9033375 | 7990.515625 | 7989.38564453125 | 1618.09777832031 |
| 9 | 15/3/2004 17:40:00 | 2 | 2129.6435646875 | 2348.39916992188 | 0 | 0 | 7984.68509593375 | 7993.41357421875 | 7992.1323828125 | 1618.09777832031 |
| 10 | 15/3/2004 17:45:00 | 2 | 2113.0263871875 | 2328.30883789063 | 0 | 0 | 7983.06889453125 | 7991.81982421875 | 7990.6640625 | 1603.97546386719 |
| 11 | 15/3/2004 17:50:00 | 2 | 2129.57693429688 | 2328.28125 | 0 | 0 | 7982.71484375 | 7991.15771484375 | 7989.79513671875 | 1603.97546386719 |
| 12 | 15/3/2004 17:55:00 | 2 | 2232.62709545088 | 2330.47094726563 | 0 | 0 | 7982.36572265625 | 7992.504296875 | 7993.7333984375 | 1603.97546386719 |
| 13 | 15/3/2004 18:00:00 | 2 | 2123.23510742188 | 2321.19458007813 | 0 | 0 | 7983.17919921875 | 7991.91845703125 | 7990.6376953125 | 1590.267578125 |
| 14 | 15/3/2004 18:05:00 | 2 | 2104.76513671875 | 2324.64428710938 | 0 | 0 | 7984.53759765625 | 7992.9990234375 | 7991.8076171875 | 1590.267578125 |
| 15 | 15/3/2004 18:10:00 | 2 | 2100.61157226563 | 2314.17114257813 | 0 | 0 | 7982.85009765625 | 7991.2608953375 | 7990.046875 | 1590.267578125 |
| 16 | 15/3/2004 18:15:00 | 2 | 2232.62709545088 | 2307.91942353375 | 0 | 0 | 7977.07934453125 | 7986.0049046875 | 7985.19139453125 | 1590.2677441406 |
| 17 | 15/3/2004 18:20:00 | 2 | 2154.74536132813 | 2301.00634765625 | 0 | 0 | 7986.78759765625 | 7996.044921875 | 7994.5517578125 | 1590.2677441406 |
| 18 | 15/3/2004 18:25:00 | 2 | 2120.32846679688 | 2297.05908203125 | 0 | 0 | 7984.22314453125 | 7993.2841796875 | 7991.64892578125 | 1590.2677441406 |
| 19 | 15/3/2004 18:30:00 | 2 | 2112.58962304688 | 2299.71801757813 | 0 | 0 | 7981.6748046875 | 7990.89013671875 | 7989.62744140625 | 1590.2677441406 |
| 20 | 15/3/2004 18:35:00 | 2 | 2110.02709603088 | 2307.97412105375 | 0 | 0 | 7983.3115234375 | 7992.615234375 | 7991.02493234375 | 1590.2677441406 |
| 21 | 15/3/2004 18:40:00 | 2 | 2118.21411132813 | 2298.16870117188 | 0 | 0 | 7988.57666015625 | 7996.25244140625 | 7996.603515625 | 1590.2677441406 |
| 22 | 15/3/2004 18:45:00 | 2 | 2108.12475685938 | 2291.4765625 | 0 | 0 | 7984.568393375 | 7994.25146484375 | 7992.767578125 | 1574.63999517188 |
| 23 | 15/3/2004 18:50:00 | 2 | 2113.36974023438 | 2300.37744140625 | 0 | 0 | 7982.18896484375 | 7991.78076171875 | 7993.995078125 | 1574.63999517188 |
| 24 | 15/3/2004 18:55:00 | 2 | 2110.06719302013 | 2317.04321209063 | 0 | 0 | 7980.97914453125 | 7990.6201171875 | 7990.322421075 | 1574.63999517188 |
| 25 | 15/3/2004 19:00:00 | 2 | 2110.21506789063 | 2292.57373046875 | 0 | 0 | 7979.580078125 | 7989.447265625 | 7986.97265625 | 1577.57019042969 |
| 26 | 15/3/2004 19:05:00 | 2 | 2108.0633359375 | 2298.83568335938 | 0 | 0 | 7980.2685546875 | 7990.52587890625 | 7988.2931640625 | 1577.57019042969 |
| 27 | 15/3/2004 19:10:00 | 2 | 2114.28125 | 2294.99194335938 | 0 | 0 | 7977.625 | 7987.28759765625 | 7985.1064453125 | 1577.57019042969 |

Figura 7.6: Consulta à view

7.3.3 Caso Real

Para demonstrar a utilização do projeto temporal em uma situação real, aplica-se este projeto na tabela de coleta de dados de energia da aplicação de curva de carga. Os detalhes desta aplicação são descritos a seguir.

7.3.3.1 Preparação

A tabela de coleta de dados de energia armazena dados de consumo e engenharia e possui os seguintes campos:

- Energia ativa delivery;
- Energia ativa recebida;

- Energia reativa delivery;
- Energia reativa recebida;
- Corrente na fase A;
- Corrente na fase B;
- Corrente na fase C;
- Tensão na fase A;
- Tensão na fase B;
- Tensão na fase C.

Para tornar esta tabela temporal, deve-se seguir os seguintes passos:

1. Criar uma tabela temporal para cada campo de medição. As tabelas são:
 - tb_temp_revenue_eneat_del,
 - tb_temp_revenue_eneat_rec,
 - tb_temp_revenue_enere_del,
 - tb_temp_revenue_enere_rec,
 - tb_temp_revenue_iamean,
 - tb_temp_revenue_ibmean,
 - tb_temp_revenue_icmean,
 - tb_temp_revenue_vamean,
 - tb_temp_revenue_vbmean,
 - tb_temp_revenue_vcmean
2. Criar uma tabela auxiliar, com o nome tb_temp_revenue_aux;
3. Criar a procedure pr_ins_revenue, que será usada para inserção e alteração de dados;

| Nome da Tabela | Quantidade de dados |
|---------------------------|---------------------|
| tb_temp_revenue_eneat_del | 285 |
| tb_temp_revenue_eneat_rec | 1 |
| tb_temp_revenue_enere_del | 315 |
| tb_temp_revenue_enere_rec | 1 |
| tb_temp_revenue_iamean | 112 |
| tb_temp_revenue_ibmean | 112 |
| tb_temp_revenue_icmean | 112 |
| tb_temp_revenue_vamean | 141 |
| tb_temp_revenue_vbmean | 141 |
| tb_temp_revenue_vcmean | 141 |

Tabela 7.1: Dados inseridos por tabela temporal

4. Criar a view `vi_temp_revenue`, que listará os dados de forma análoga à tabela original.

Feito isso, substitui-se a tabela convencional e passa-se a trabalhar somente com a modelagem temporal. Os dados continuam sendo inseridos via *procedure* e são consultados através da *view*.

7.3.3.2 População de dados

Para o teste real utiliza-se dados de coleta do medidor do prédio do Celta, no Parque Tecnológico onde a Airgate está instalada. Trata-se de um medidor de modelo ION 7500, da marca Power Measurement [Pow 04]. O sistema de coleta continua o mesmo, mas é feita a alteração no trigger de inserção da tabela de interface, apontando a inserção para a nova *procedure*.

A medição foi realizada entre 15/3/2004, às 20:00 hs e 17/3/2004 às 11:45 hs, ou seja, 478 dados de medição foram inseridos neste intervalo. Analisando por tabela, os 478 dados foram reduzidos, conforme mostra a tabela 7.1

Cada dado inserido na tabela original possui os seguintes tipos de dados:

- 10 campos do tipo Double, 16 bytes
- 1 campo do tipo Inteiro, 1 byte
- 1 campo do tipo Data, 8 bytes

Com 478 dados inseridos, temos:

$$478 * (10*16 + 1*1 + 1*8) = 80782 \text{ bytes}$$

Na tabela de dados temporais, cada dado é armazenado como:

- 1 campo do tipo Double, 16 bytes
- 1 campo do tipo Inteiro, 1 byte
- 2 campos do tipo Data, 8 bytes

Considerando os números apresentados na tabela 7.1, temos:

$$(285+1+315+1+112+112+112+141+141+141) * (16+1+2*8) = 44913 \text{ bytes}$$

Com isso, pode-se verificar uma economia de aproximadamente 50% de espaço em disco, ganho considerável visto que o custo de armazenamento e acesso torna-se alto, a medida que a massa de dados aumenta.

7.3.4 Consultas temporais

A implementação do suporte temporal abre o horizonte do sistema para que sejam permitidas algumas consultas temporais que antes necessitavam de muitas linhas de código e grande quantidade de recursos computacionais para realizá-las.

Estas consultas são específicas da aplicação a qual foi aplicado o suporte temporal e podem ser mais ou menos aplicáveis dependendo do contexto. No caso da aplicação de Curva de Carga, pode-se exemplificar algumas:

Consultas a intervalos de tempo em que faltou energia:

```
select
    *
from
    tb_temp_revenue_eneat_rec
where
    valor = 0;
```

Lista de intervalos em que os valores chegaram a um pico:

```
select
    *
from
    tb_temp_revenue_iamean
where
    valor = (select
                max(valor)
            from
                tb_temp_revenue_iamean);
```

Análise da variação no consumo ou da tensão:

```
select
    count(*)
from
    tb_temp_revenue_vamean
where
    timestamp_inicio between '15/03/04'
                        and '16/03/04';
```

7.3.5 Escalabilidade

O sistema foi implantado e testado na tabela de dados de energia do Sistema de Curva de Carga, mas pode ser expandido para que as demais tabelas de dados do sistema utilizem a mesma tecnologia. A utilização ou não do paradigma de dados temporais deve ser analisada tanto do ponto de vista do consumo de dados em disco, como de eficiência nas consultas e operações com os dados.

7.4 Conclusão

Este capítulo descreveu de forma sucinta como foi projetado o protótipo do sistema, suas funcionalidades e aplicações práticas. Os casos de uso descrevem passo a passo as operações principais do sistema, quais os testes que devem ser feitos no momento de uma inserção ou alteração de um determinado valor.

Além disso, foram apresentados alguns resultados simulados a partir de dados reais, comparando espaço em disco ocupado em cada uma das implementações e foram sugeridos alguns exemplos de consultas que são permitidas no modelo temporal implementado.

Capítulo 8

Considerações Finais

BDTs são úteis em vários domínios de aplicação que necessitam de histórico. Eles facilitam certas pesquisas temporais e muitas vezes diminuem o espaço em disco ocupado pelos dados. Dada a carência de SGBDs comerciais deste tipo, a extensão temporal apresentada neste estudo tem a vantagem de dar algum suporte temporal a uma aplicação totalmente desenvolvida em uma plataforma convencional.

Em contraponto às vantagens apresentadas, a performance das consultas tende a ser pior em um cenário inicial, porém, a medida que a sumarização dos dados vai aumentando, a grande diminuição no volume de dados tende a compensar esta desvantagem.

Segundo métricas, o suporte temporal apresentou uma economia de aproximadamente 50% de espaço em disco, ganho considerável visto que o custo de armazenamento e acesso torna-se alto, a medida que a massa de dados aumenta. Embora estas métricas não tenham uma confiabilidade totalmente confirmada, servem como parâmetro para se observar que o suporte temporal trouxe vantagens à aplicação analisada.

Outra questão é que novas consultas temporais podem ser adicionadas ao sistema, permitindo assim que este tipo de informação seja melhor analisada pelo sistema de Curva de Carga.

Apesar do estudo ter sido focado em uma aplicação de medição de energia, este conceito pode ser facilmente relacionado a outras aplicações que armazenem grande quantidade de dados.

Enfim, o estudo de outras tecnologias de armazenamento de dados ajuda a otimização, tanto na utilização de recursos como no incremento de funções às aplicações cliente.

8.1 Sugestões para Trabalhos Futuros

- A otimização do intervalo de validade de um dado, evitando que sejam inseridas duas informações de data em uma única tupla;
- Análise da possibilidade de apagar dados temporais, apesar de esta não ser uma implementação necessária no domínio da aplicação de medição estudada;
- Criação de uma interface própria para o tratamento e consulta destes dados.

Referências Bibliográficas

- [AIR 03] AIRGATE. **Airgate**. Disponível em <<http://www.airgate.com.br/>>. Acesso em: Maio, 2003.
- [All 04] Allround Automations. **Allround Automations**. Disponível em <<http://www.allroundautomations.nl>>. Acesso em: 15/03/2004.
- [DIA 03] DIAS, R. Especificação técnica do projeto de curva de carga. Airgate Telecomunicações Ltda., Setembro, 2003. Relatório técnico.
- [EDE 94] EDELWEISS, N.; OLIVEIRA, J. P. M. Modelagem de aspectos temporais de informação. **IX Escola de Computação**, [S.l.], v.1, 1994.
- [EDE 98] EDELWEISS, N. Bancos de dados temporais: Teoria e prática. **XVII Jornada de Atualização em Informática - Anais do XVIII Congresso Nacional da Sociedade Brasileira de Computação "Rumo à Sociedade do Conhecimento"**, [S.l.], v.1, 1998.
- [MEL 03] MELLO, R. D. S. **Notas de aula da disciplina de Bancos de Dados Não Convencionais**. Disponível em <<http://www.inf.ufsc.br/~ronaldo/ine5342/bd3-6-bdt.pdf>>. Acesso em: 15/03/2003.
- [ORA 04] ORACLE. **Oracle**. Disponível em <<http://www.oracle.com>>. Acesso em: 15/03/2004.
- [Pow 04] Power Measurement. **Power Measurement**. Disponível em <<http://www.pwrm.com>>. Acesso em: 15/03/2004.
- [TIG 04] TIGER. **Tiger**. Disponível em <<http://www.cs.auc.dk/tiger/>>. Acesso em: 15/03/2004.
- [Tim 04] Time Consult. **Time Consult**. Disponível em <<http://www.timeconsult.com>>. Acesso em: 15/03/2004.

Apêndice A

Fontes