



UNIVERSIDADE PRESIDENTE ANTÔNIO CARLOS
Faculdade de Ciência da Computação e Comunicação Social
Curso de Ciência da Computação

Marcos Flávio Rocha Conceição

BANCO DE DADOS MÓVEIS

BARBACENA

2005

Marcos Flávio Rocha Conceição

BANCO DE DADOS MÓVEIS

Monografia apresentada como pré-requisito para obtenção do grau de Graduação em Ciência da Computação Sob a orientação do Prof^o . Gustavo Campos Menezes.

BARBACENA

2005

Marcos Flávio Rocha Conceição

Bancos de Dados Móveis

Aprovada em _____ / _____ / _____

Gustavo Menezes (Orientador) – UNIPAC-Barbacena

Luis Augusto Mattos Mendes – UNIPAC-Barbacena

Wender Magno Cotta – UNIPAC-Barbacena

Agradeço ao Professor e Orientador Gustavo Campos Menezes, pelo apoio e encorajamento contínuos na pesquisa, aos demais Professores da casa, pelos conhecimentos transmitidos, e à Diretoria da graduação da UNIPAC, pelo apoio institucional e pelas facilidades oferecidas.

“De uma maneira geral, os brinquedos documentam como o adulto se coloca com relação ao mundo da criança”.

Walter Benjamin

RESUMO

Este trabalho apresenta um estudo sobre Banco de dados móveis, enfatizando suas principais características. Mostrando que o seu uso hoje vem expandindo de forma incrível principalmente em telefonia móvel. Apresentando algumas características e vantagens, como também algumas deficiências, visto que é usado em um ambiente móvel. Algumas dúvidas sobre o tema serão debatidas, abordando questões sobre transações e mobilidades neste setor muito interessante, sendo uma variação da computação que mais atrai a curiosidade de todos.

SUMÁRIO

CAPÍTULO 1. INTRODUÇÃO.....	01
CAPÍTULO 2. BANCO DE DADOS DISTRIBUÍDOS E MÓVEIS.....	04
CAPÍTULO 3. COMPUTAÇÃO MÓVEL E SGBD.....	08
3.1. Processo de Transações no Ambiente de Computação Móvel.....	08
3.2. Estados de Operação de um Host Móvel.....	10
3.3. Estados de Operações Desconectadas de um Host Móvel.....	11
3.3.1. Hoarding.....	12
3.3.2. Operações desconectadas.....	12
3.3.3. Reintegração.....	13
3.4. Checkpoint.....	15
CAPÍTULO 4. GERÊNCIA DE DADOS MÓVEIS.....	17
4.1. Difusão de Dados ou Disseminação	18
CAPÍTULO 5. ARQUITETURA DE BANCO DE DADOS DE LOCALIZAÇÃO.....	21
5.1. Esquemas em duas camadas.....	21
5.2. Esquemas hierárquicos.....	22
CAPÍTULO 6. O MERCADO DE SGBD MÓVEL.....	24
6.1. Sybase Anywhere Studio.....	24
6.2. Oracle Lite.....	26
6.3. DB2 Everyplace.....	27
6.4. Microsoft SQL Server CE.....	28
CONCLUSÃO	30
REFERÊNCIA BIBLIOGRAFICAS.....	32

LISTA DE ILUSTRAÇÕES

Figura 1 - Componentes de um Sistema de Bancos de Dados.....	01
Figura 2 - Arquitetura de banco de dados distribuído.....	05
Figura 3 - Estados de operação de um host móvel.....	10
Figura 4 - Forma de broadcast push-based.....	19
Figura 5 - Forma de broadcast com push-based e pull-based.....	20
Figura 6 - Armazenamento e busca com localização hierárquica.....	23

LISTA DE TABELAS

Tabela 1 - Principais problemas e questões relacionadas ao estado de desconexão de uma unidade móvel.....	14
Tabela 2 - Quadro comparativo dos bancos de dados.....	30
Tabela 3 – Pesquisa de SGDB no mercado.....	30

LISTA DE SIGLAS/ABREVIATURAS

ADO	ActiveX Data Objects;
ADOCE	ActiveX Data Objects for Windows CE
ASA	Adaptive Server Anywhere;
BD	Banco de Dados
CODASYL	Comitee for Data Systems Language;
CSS	Common Channel Signaling;
DAO	Data Access Objects;
DDL	Data Defition Language;
ER	Entidade-Relacionamento;
IIS	Internet Information Server;
IMS	Information Management System;
MGP	Message Generator and Processor;
ODBC	Open Database Connectivity;
OKAPI	Object Kernel API;
RDA	Remote Data Access;
RDO	Remote Data Objects;
SBD	Sistema de Banco de Dados;
SGBD	Sistemas de Gerência de Banco de Dados;
SGBDR	Sistema de Gerenciamento de Banco de Dados Relacional;

CAPÍTULO 1. INTRODUÇÃO

O Banco de dados pode ser visto como o equivalente eletrônico a um armário de arquivamento, em outras palavras é um repositório ou recipiente para uma coleção de arquivos de dados computadorizados[1]. Um Sistema de Banco de Dados é basicamente um sistema computadorizado de armazenamento de registros; isto é um sistema computadorizado cujo propósito geral é armazenar informações e permitir ao usuário buscar e atualizar essas informações quando solicitado. As informações em questão podem ser qualquer coisa que tenha significado para o indivíduo ou a organização a que o sistema deve servir - em outras palavras, tudo o que seja necessário para auxiliar no processo geral de tomada de decisões desse indivíduo ou dessa organização[2]. A seguir a Figura 1 mostra a representação simplificada de um Sistema de Banco de Dados, onde contém quatro componentes principais: dados, *hardware*, *software* e usuário[1].

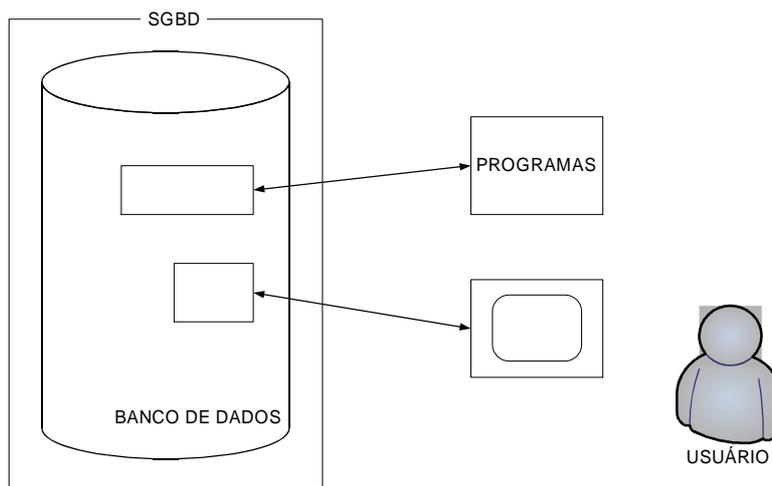


Figura 1 - Componentes de um Sistema de Bancos de Dados(Adaptada de [1]).

Hoje em dia são diversas as maneiras com que as pessoas se envolvem diretamente com um Sistema de Banco de Dados, desde consultas a saldos bancários, consultas a itens de um supermercado até a interação com banco de dados na *internet* para realização de aquisições de produtos.

Os Sistemas de Banco de Dados e seus componentes básicos fornecem todos os mecanismos para:

- Confiança;
- Segurança;
- Disponibilidade;
- Integridade;
- Acesso eficiente a suas bases de dados.

Desta maneira, o Sistema de Banco de Dados pode estar tanto em redes convencionais, fazendo papel de servidor de dados para as aplicações, como também, atuando em uma nova forma de servidor, agora residindo em um equipamento móvel [3]. Esta nova forma de atuar do Sistema de Banco de dados, exigirá um novo perfil e uma nova adaptação nos Sistema Gerenciador de Banco de Dados, para que o mesmo possa atuar em seu papel de administrador dos dados, em relação aos perfis do ambiente móvel. Entre as características destaca-se a constante desconexão dos aparelhos móveis de suas redes, devido à sensibilidade das redes sem fio, nas quais os aparelhos móveis estão conectados e os recursos limitados dos equipamentos que hospedarão os Sistemas gerenciadores Móveis.

Este trabalho tem como objetivo destacar o conceito de banco de dados no meio da computação móvel, colocando em ênfase as novas requisições e suas funções vitais, apresentando alguns modelos que possam vir a ser utilizados com os

bancos de dados, onde a mobilidade é um dos fatores mais importantes da computação, e mostrar algumas questões problemáticas encontradas e como os bancos de dados podem a vir solucionar tais questões.

CAPÍTULO 2. BANCO DE DADOS DISTRIBUÍDOS E MÓVEIS

Um sistema de banco de dados distribuído é uma coleção de dados que é distribuída por diferentes computadores, possivelmente em diferentes locais. Os computadores estão conectados por uma rede de comunicação. O sistema deve suportar aplicações locais em cada computador, bem como aplicações globais nas quais mais de um computador estejam envolvidos[4]A Figura 2 fornece um exemplo de arquitetura de um banco de dados distribuído.

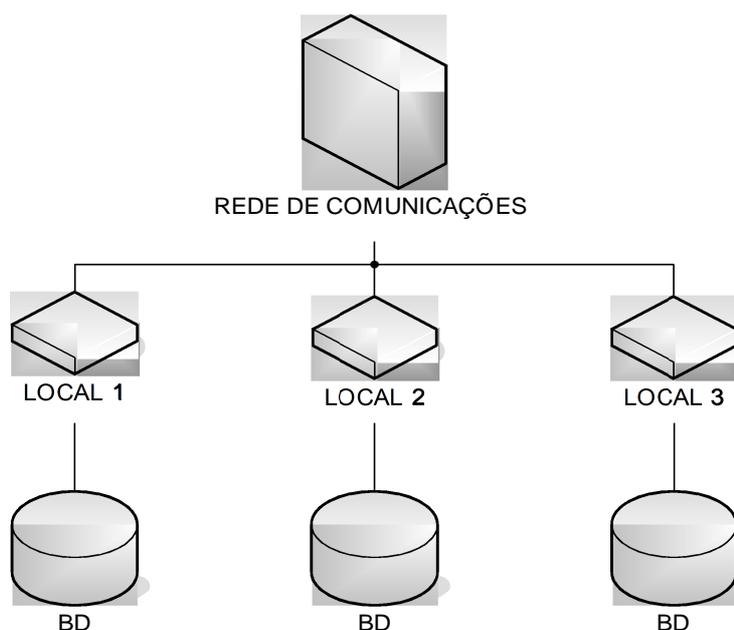


Figura 2 - Arquitetura de banco de dados distribuído(adaptada de[4]).

Um banco de dados distribuído deve possuir como características:

- Independência de replicação: usuários devem ser capazes de se comportar, pelo menos do ponto de vista lógico, como se os dados não

fossem replicados de modo algum [8],sem necessitar lidar com as redundâncias introduzidas pelo processo físico.

- Gerenciamento das transações distribuídas: os gerenciadores devem ser capazes de possibilitar alterações nos estados de base de dados local por uma única transação de forma consistente[3].Isto significa que, independente das possibilidades de falhas que possam ocorrer no processo em um ambiente distribuído, o sistema deve possibilitar a garantia[8] de dois aspectos,o controle de recuperação e o controle de concorrência.

- Autonomia para transações locais: todos os dados distribuídos pela rede devem ser de propriedade e gerência local.Um sistema gerenciador possibilita o acesso aos dados de forma integrada, embora cada *site* continue a ser administrado independentemente do sistema local distribuído[8].

- Independência em relação a um site central: não existe um local que possui autoridade sobre os demais.Issso implica que em cada *site* deve haver uma cópia do dicionário de dados e dos mecanismos de segurança[8].

- Confiabilidade: traz melhorias porque sistemas distribuídos não seguem a proposta do tudo ou nada.Eles podem continuar a funciona, em nível reduzido, mesmo diante da falha de algum componente individual, como um *site* isolado[3].

- Independência de localização: aplicações e usuários não precisam conhecer a localização dos bancos de dados nas redes de computadores ou lidar com protocolos utilizados na comunicação entre os diferentes sistemas[8].

- Independência de fragmentação: significa que as aplicações e os usuários têm sempre uma visão única dos dados independente da forma como as relações estejam fragmentadas entre os diferentes *sites*[3].
- Independência da rede: um sistema gerenciador pode implicar a utilização de diversos protocolos, requerendo eventualmente a conversão de protocolos. Todos esses detalhes não devem ser visíveis aos usuários devendo ser resolvidos pelo próprio gerenciador[8].
- Independência do Gerenciador de Banco de Dados: deve ser possível armazenar e recuperar dados de diferentes bancos de dados locais, independente da arquitetura do banco de dados e do próprio gerenciador utilizado em cada um daqueles locais[8].
- Processamento distribuído das consultas: o usuário deve poder efetuar consultas sobre o conjunto de dados, cabendo ao sistema realizar o processamento necessário para viabilizar o acesso aos subconjuntos de dados de cada local. Isto significa, por exemplo, que seria possível a realização de uma consulta única que resultasse em acesso aos dados dispersos por vários *sites*[8].
- Independência de hardware: uma aplicação que necessite ter acesso a dados dispersos por máquinas diferentes deve poder realizar o seu processamento sem restrições decorrentes das diferenças entre as plataformas[8].
- Independência de Sistemas Operacional: de forma semelhante ao que foi observado na relação ao *hardware*, é necessário que uma aplicação tenha acesso aos dados dispersos por plataformas com sistemas operacionais distintos sem restrições decorrentes de suas diferenças[8].

Banco de dados móvel é uma base de dados de ampla portabilidade e separada de seu servidor de base de dados local, porém é capaz de se estabelecer comunicação por meio de *sites* distantes, admitindo o compartilhamento do dado local.

A computação móvel pode ser considerada um segmento ou uma variação da computação distribuída, em sistemas distribuídos o foco principal está na transparência da distribuição, já nos sistemas móveis o foco fica em torno da informação da localidade em que se encontra[3].As bases de dados de um sistema móvel podem ser repartidas pelos seus integrantes com e sem fio, onde os compromettimentos do gerenciamento são divididos entre as unidades móveis e a estação base.Também são distribuídas entre os integrantes ligados através de cabos, onde uma base torna-se gerente de si mesma, como um sistema gerenciador com atributos agregados para localizar unidades móveis e também gerenciar futuras consultas e operações do ambiente.

Segue a seguir algumas peculiaridades sobre Sistema de banco de dados móvel:

- Primeiramente tem-se a questão de que o equipamento móvel pode atuar com cliente ou até mesmo como um servidor de banco de dados;
- Pode ser acessado um banco de dados através de um equipamento fixo ou até mesmo móvel;
- Diversos bancos de dados são ligados por uma rede de sem fio(*wirelles*);
- Estes bancos podem ter autonomia em sua totalidade, serem completamente distribuídos e serem compostos por partes diferentes.

Em relação à gerência e conseqüentemente à administração dos dados, diversos obstáculos podem e devem afetar o perfeito funcionamento e o pleno exercício do banco de dados no meio computação móvel[5]. Seguem alguns destes empecilhos:

- Devido à mudança permanente de localidade além de poder causar a volatilidade de informação, pode também ocasionar a ausência de conexão e também possível cancelamento de operações;
- Pelo limitado tamanho de memória dos computadores móveis, é pouco provável que aconteça a técnica de replicação ;
- Devido ao fato de serem conectados por redes sem fio(*wireless*) ocorrem a desconexão, conseqüentemente fim de transações, ampliando a necessidade de *checkpoint*;
- Devido à limitada autonomia das baterias: tem-se modo sem conexão da rede e pode surgir cancelamento de operações;
- Bancos de dados em franco crescimento causam impacto nas operações de consultas e com isso diminui dados nos bancos de dados que estão em *hosts* móveis;
- Congestionamento e com isso baixa o nível de velocidade das ligações sem fio, acabam com isso por ocasionar o retardamento nas consultas entre *host* móveis e sua base de dados;
- Tem sempre o anseio de poder se fazer uma administração para localizar *hosts* móveis durante as consultas;
- Devido a constante troca de células, entre uma estação base e outra(*handoff*) necessita que aumente o controle e conseqüentemente a localidade dos *host* móveis.

CAPÍTULO 3. COMPUTAÇÃO MÓVEL E SGBD

Na computação móvel tem-se vários problemas, diretamente estão as transações de banco de dados onde constantemente ocorre a falta de conexão entre os clientes móveis e suas redes tradicionais, também pode ocorrer pela pouca expressividade das redes sem fio e também a freqüente mudança dos clientes móveis por um extenso território e as freqüentes falhas de recuperação das transações[3].Serão analisadas algumas questões mencionadas anteriormente, com o objetivo de dar ênfase a gerência de dados dos Sistemas móveis.

3.1. Processo de Transações no Ambiente de Computação Móvel

Transação é considerada um mecanismo de unidade lógica para realizar processos em banco de dados.Transação, por exemplo, pode ser qualquer tipo de operação como incluir, excluir, modificar ou recuperar dados.Operações que formam transações entre banco de dados e servidores podem ser inseridas dentro do próprio software da aplicação ou também podem ser especificadas de maneiras interativas através de uma linguagem como a SQL.Para se manter dados íntegros há uma necessidade que as transações devam possuir, e que são garantidas pelos Sistemas Gerenciadores móveis:

- Transações no momento da execução de uma operação não devem ser reconhecidas por outras que concorre no mesmo sistema, conhecido como isolamento;

- Quando uma transação é completada, persiste no banco de dados, mesmo que aconteça falha, este é o conceito de durabilidade.
- Transação deve ser uma unidade atômica, sendo completamente executada ou totalmente descartada, esta é a atomicidade.
- Transação deve ser consistente, levar a base de dados de um estado consistente a outro estado.

Todo conjunto de instruções de dados que solicita algum tipo de dados é normalmente considerada estruturada como operação atômica, com a meta de deixar os dados sempre consistentes quando se tratar de concorrência e falhas no sistema. Porém, um conjunto de instruções móveis que tenta acessar algum tipo de dado não pode ser considerada estruturada usando uma operação atômica. Isto ocorre porque transações admitem o ato de serem excluídas isoladamente, o que não permite que as mesmas se dividirem em sua computação é o compartilhamento de seus estados e considerações práticas deste meio, exige que uma unidade de suporte a mudança de localidade conhecida por estação, tanto para computação com em comunicação de dados. Implica em dizer que a computação móvel precisa ser coerentemente estruturada como um conjunto de operações, que executem na unidade móvel, onde outras são executadas na unidade móvel de suporte a mudança de localidade. Em transação móvel, a computação é dividida e executada um período no *host* móvel e outro período no *host* fixo, o processo de transação é afetado de várias maneiras, devido do uso de meio sem fio e a constante mudança de localidade resultante dos clientes e produtores de dados.

A utilização de conexão *wireless* resulta em transações extensas, em virtude dos constantes atrasos da rede. Por este motivo é que os clientes de unidades

móveis evitam constantemente o uso das conexões sem fio, vendo que este tipo de conexão não é tão barata, tanto financeiramente quanto em relação do poder de consumo das unidades móveis, ou também os recursos de rede não estão acessíveis em sua localização.

3.2. Estados de Operação de um *Host* Móvel

Em um ambiente de sistema distribuído tradicional, as operações em suas unidades móveis são totalmente conexo ou sem conexão[4]. No caso do meio móvel existem várias etapas de momento desconexo que vão desde a falta de conexão por completa até uma conexão bem fraca. Como consequência, uma unidade móvel pode operar em alguns estados diferentes. Também existe um outro estado que é conhecido como cochilo, serve para conservar bateria. Antes de desconectar da rede um *host* móvel pode fazer o chamado protocolo de execução. Este protocolo tem como objetivo dar garantia de realizar a transferência de dados para que o próprio continue a operar o mínimo necessário. Uma unidade móvel pode trocar seu estado desconexo para poder executar no estado parcialmente conexo. Enquanto estiver neste estado, todo o contato com a rede deve ser reduzido em apenas uma célula. Nesse momento todos os dados pertencentes ao estado do *host* móvel devem ser transferidos à estação base da célula, por meio de um protocolo conhecido por *handoff*. A Figura 3 representa estes estados em que um *host* móvel se encontra.

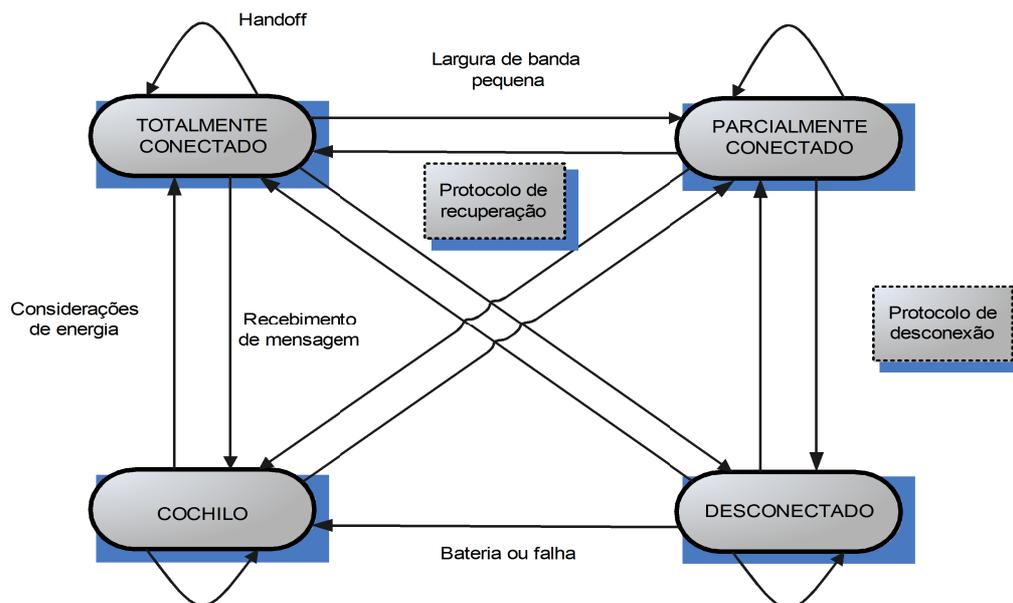


Figura 3 – Operações conectadas de um host (Adaptada de[4]).

3.3. Estados de Operações Desconectadas de um *Host* Móvel

Para diminuir o consumo de bateria, essencialmente no *palmtops*, uma operação no *host* móvel é realizada, o cochilo. O estado de cochilo diminui consideravelmente a velocidade do *clock* do processador e nenhum tipo de operação do cliente é executado. A unidade móvel aguarda em modo passivo o recebimento de mensagens enviadas por parte dos integrantes da rede, voltando ao estado normal de execução quando recebe uma mensagem. Como no estado sem conexão, está é uma operação espontânea. Porém no estado de cochilo a unidade móvel é alcançada pelo sistema, com isso pode ser instigado pelo sistema a voltar em seu estado de operação normal.

Uma falta de conexão é vista, com isso ocorre que dados são imediatamente transferidos para o usuário móvel, para com isso pode conceder a operação durante

um tempo de falta de conexão. E esta desconexão poderá ocorrer em três estados distintos.

3.3.1. *Hoarding*

Previamente à perda da conexão, o *host* móvel está em um estado de carga prévia de dados. Nesse estado os dados mínimos necessários para poder realizar uma operação são carregados no *host* móvel. Os dados podem ser simplesmente localizados novamente ou levados do *host* fixo para a unidade móvel. Porém esse dados não serão acessíveis para outros *sites*. Tem como saída as informações transferidas para a unidade móvel depende diretamente da aplicação e também do tipo de dados que o embasa.

3.3.2. Operações desconectadas

Acontece quando há uma quebra de conexão com a rede convencional, o *host* móvel entra em estado desconexo. O tempo que ficar neste estado, somente poderá usar dados locais. Quando tentar solicitar dados não locais esta não será atendida e voltará em forma de erro. Neste caso estes itens pendentes serão organizados em uma fila, para poderem ter atendimento quando este estiver conectado[4]. Aplicações com solicitações que retornam erro, por não poderem ser atendidas no momento poderão suspender a execução ou trabalhar em processos que não dependa de dados locais.

3.3.3. Reintegração

Uma nova conexão entre o host móvel e a rede convencional acontece, chamamos de estado de reintegração. Nesse processo atualizações da unidade móvel reintegradas através de sites, realizando novamente sua entrada no host fixo. Discussões referentes a possíveis concorrências das transações são fixadas em torno de cada sistema em peculiar, objetivando resolver esta questão da atualização em cima do referente objeto. A Tabela 1 mostra os evidentes problemas e questionamentos levantados em relação ao estado desconexo.

Estado desconexo	Problemas e discussões	Análise
Hoarding	Unidade de carga	Depende do sistema
	Que itens carregar?	<ul style="list-style-type: none"> - Especificado pelo usuário - Usar as informações sobre o histórico dos acesso aos dados - Depende na aplicação para que o sistema é usado
	Quando executar a carga dos dados? Solicitação para todos não disponíveis localmente	<ul style="list-style-type: none"> - Antes da desconexão - Em uma base regular - Retorna uma exceção - Solicita uma fila
Desconexão	O que por no LOG?	<ul style="list-style-type: none"> - Valores dos dados - Timestamps - Operações
	Quando otimizar o LOG?	<ul style="list-style-type: none"> - Nas operações locais - Antes de integração
	Como otimizar o LOG	Depende do sistema
Reintegração	Como integrar?	Reexecutando as operações do LOG
	Como resolver os conflitos?	<ul style="list-style-type: none"> - Usar semântica das aplicações - Soluções automáticas - Ferramentas para ajudar ao usuários

Tabela 1 - Principais problemas e questões relacionadas ao estado de desconexão de uma unidade móvel.[4]

3.4. CHECKPOINT

Quando uma falha de sistema ocorre, devemos consultar o log para determinar aquelas transações que necessitam ser refeitas e aquelas que necessitam ser inutilizadas[1].A princípio, para isso, deveríamos pesquisar todo o log.Há duas grandes dificuldades nessa abordagem:

1. O processo de pesquisa consome tempo.
2. Muitas das transações que, de acordo com nosso algoritmo, necessitam ser refeitas já escreveram suas atualizações no banco de dados.Embora refaze-las não cause dano algum.a recuperação torna-se mais longa.

Para reduzir esses tipos de overhead, introduzimos os *checkpoints*(pontos de controle),que na realidade são pontos de monitoração.Em certos intervalos de predeterminado, em geral, sempre que algum número preestabelecido de entradas é gravado no *log*,o sistema automaticamente marca um *checkpoint*.Marcar *checkpoint* envolve (a) gravar fisicamente o conteúdo dos *buffers* do banco de dados no banco de dados físico e (b) gravar fisicamente um registro de *checkpoint* especial no *log* físico O registro de checkpoint fornece uma lista de todas as transações que estavam em andamento no momento em que o checkpoint foi marcado.

Os checkpoints guardam em intervalos regulares o estado de uma transação em um tipo de armazenagem estável, o que bem compreendido, porque é determinado no caso de recuperação de falhas.Para recuperar uma aplicação é utilizado e guardado em um estado dito global.Este estado compreende o estado de cada um dos processos participantes e também inúmeras mensagens.Com isso para se obter um certa recuperação,este processo deve admitir o estado geral desta aplicação.

Conforme Date 2000[1]tem como classificar de duas maneiras os algoritmos de checkpoint,os coordenados,onde esses precisam que cada componente coordene seus respectivos checkpoints para poder dar garantia de recuperação do checkpoint geral.Já os sem coordenação deixam os componentes do checkpoint sem dependência em seu respectivo estado.No decorrer da recuperação, um momento coordenado é suficiente para poder escolher para os checkpoint gerais com concisão.Por causa da constante mudança de localidade, largura estreita e falta de conexão, estes checkpoints não são adequados para a computação em movimento.

CAPÍTULO 4. GERÊNCIA DE DADOS MÓVEIS

Neste capítulo foca-se técnicas de como gerenciar informações e como recuperar as mesmas em sistemas que se movem constantemente. Ressalta-se a considerada técnica e também a mais importante dentro das técnicas, a chamada disseminação, conhecida também como difusão de dados ou broadcast. Esta técnica faz a entrega de informação às unidades móveis. Hoje em dia a grande parte das tecnologias wireless suportam a broadcast para os elementos que estão inseridos em uma célula. No caso do cliente-servidor, o servidor que reside na estação base pode aproveitar a vantagem de possuir o perfil para disseminar dados para os clientes de sua célula. Com isso se estabelece um novo meio de se enviar dados aos clientes que difere da forma convencional de envio de informações, que apenas acontece quando um cliente a solicita. Para se localizar um cliente móvel, deve haver na computação móvel um perfeito gerenciamento de consultas, de forma eficaz. Com o objetivo de se localizar o cliente móvel, aparecem alguns problemas relacionados aos acúmulos e atualizações destes dados. O gerenciamento destas consultas deve levar em conta alguns problemas encontrados nos hosts móveis.

4.1. Difusão de Dados ou Disseminação

Em um modelo convencional, dados são emitidos para o usuário somente quando há uma solicitação, o servidor recebe este pedido, faz o processamento e retorna ao usuário o dado solicitado. Esta forma de entrega de informações é denominada de pull-based, que significa buscar na base. Na computação wireless, os

servidores são munidos de canais com largura suficientemente para se fazer a entrega de informações por difusão de dados aos usuários em suas células. Este meio de entrega deu origem, a uma forma diferente de entrega de informações denominada push-based, que significa impulsionada pelo servidor (base), que se baseia em o servidor insistentemente jogar informações aos seus usuários sem qualquer tipo de solicitação dos usuários. Estes usuários vigiam esta difusão e adquirem dados que eles necessitam, da forma como eles chegam na conexão. A Figura 4 mostrar a seguir a forma broadcast push-based.

Na forma broadcast push-based, a entrega de informações é importantíssima para uma enorme gama de aplicações que precisam fazer a difusão de informações para um grande universo de usuários. Algumas aplicações podem ser mencionadas, como por exemplo, avisos de emergência, controle de trânsito terrestre, informações climáticas.

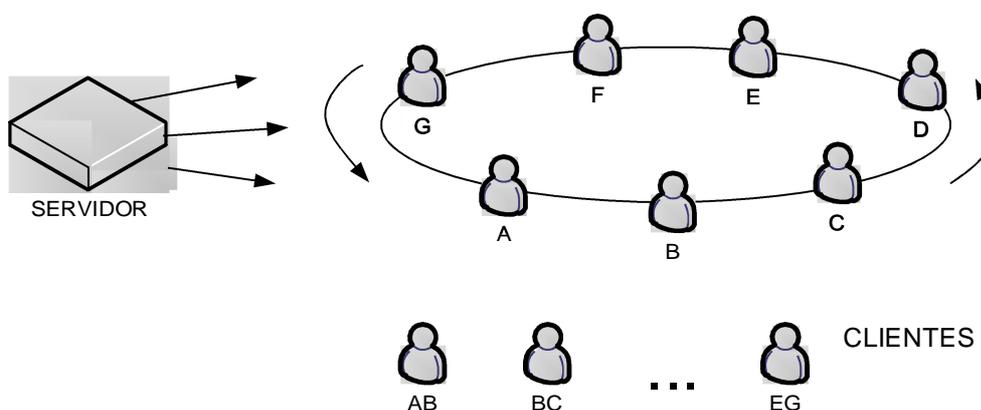


Figura 4 - Forma de broadcast push-based.(Adaptada de[4]).

Já na forma pull-based a entrega de informações está limitada à capacidade de seu servidor de rede. Em broadcast a entrega de informações tem uma limitação e o acesso a estes dados é apenas em seqüência. Usuários necessitam aguardar até

que a solicitação de dados apareça no canal de contato entre o cliente e o servidor. Em pullbased, usuários podem atuar mais ativamente e deixar explícita a solicitação de dados ao servidor.

No caso pode-se combinar estas duas técnicas, a pullbased e a pushbased, levando em conta, que os sistemas possuem um canal broadcast e também possuem um canal de comunicação com o servidor denominado backchannel. Um perfil característico desta análise é que o próprio canal é usado quando clientes solicitam informações, então tem a obrigatoriedade de se usar técnicas para compartilhar e torna-se eficiente este canal.

De formas bem diferentes podem se usar o backchannel, pode solicitar informações ao servidor em casos que os dados necessitem de sigilo. Com isso não poderia aparecer em um canal de difusão de dados. A Figura 5 demonstra esta forma de ajuste entre as duas formas.

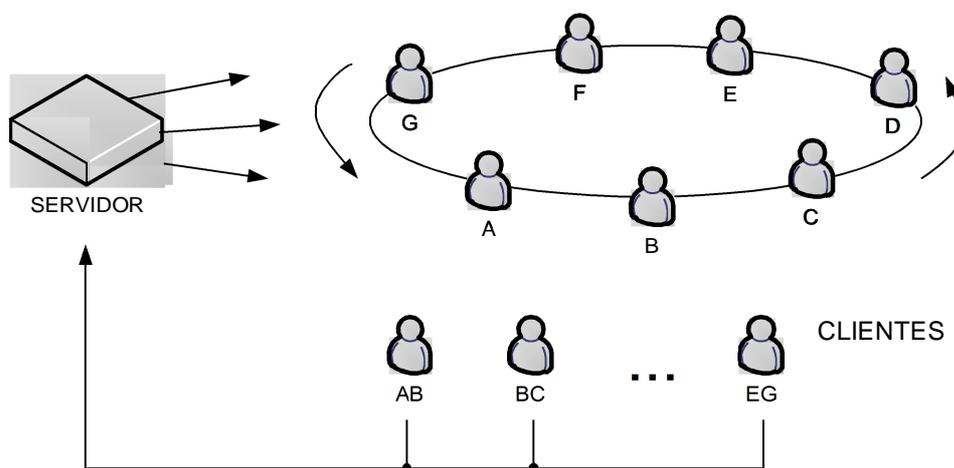


Figura 5 – Combinação entre push-based e pull-based.(Adaptada de[4]).

CAPÍTULO 5. ARQUITETURA DE BANCO DE DADOS DE LOCALIZAÇÃO

Das abordagens mais ordinárias referente a arquitetura de BD's de localização tem-se a chamada esquema de duas camadas, parecido com o modelo utilizado em telefonia celular, onde a localização atual para cada cliente em mudança de localidade em localidades distintas da rede, já o esquema de hierarquia, é a decomposição hierárquica do espaço subdividido em regiões[6]. Entraremos nesta discussão a seguir.

5.1. Esquemas em duas camadas

Cada cliente móvel tem referido a ele um registro de localidade, este registro é residido em uma área de rede especificada com antecipação para cada usuário. Este registro guarda a posição atual de cada cliente em mudança e localidade como parte da característica do cliente. Os métodos de busca é bem simples, em um caso que se quer localizar um determinado usuário, identifica-se o registro deste usuário e com isso realiza-se a consulta. No caso deste usuário movimentar-se para uma zona diferente, contata-se o registro deste usuário e atualiza-se esta nova localização.

Este esquema foi melhorado, um registrador de localidade visitante é conservado em cada área. Este registrador de localização visitante dos clientes que não pertencem à área e estão localizados correntemente dentro de sua área. Ao se estabelecer uma chamada da zona X para determinado usuário, o registro de localidade do usuário é contatado. Se vir a acontecer o movimento do usuário de uma área X para uma área Y, em adição à modificação do registro de localidade de

Y, o registro é deletado do registro visitante da zona X e um novo registro para Y é acrescentado na registro de visitante da zona Y.

5.2. Esquemas hierárquicos

Um esquema de localidade no modelo de hierarquia é uma variação do esquema tradicional em dupla camada, considerando o banco de localidade. Neste método hierárquico, um banco de dados inclui informações de localização, do topo do nível para os clientes incluídos nos níveis mais baixos. Comumente este método hierárquico é considerado como uma árvore. Nesta hipótese, o BD guarda uma folha apenas em área e incluem as entradas para os clientes registrados nesta célula. Nós mais introduzidos, guardam informações referente aos clientes registrados no agrupamento de zonas das subdivisões destas árvores. Em cada cliente móvel, estes dados são considerados ponteiros para as possíveis entradas de baixo nível no BD's ou a localidade corrente do cliente. Os banco de dados são comumente interligados por conexões de redes inteligentes. Na telefonia móvel, os BD's podem ser empregados as switches da rede. A Figura 6 ressalta exemplos de processos de armazenagem e procura neste modelo hierárquico. A Figura 6 a em modelo de como acontece e como é registrado um cliente móvel em um modelo hierárquico. Já na Figura 6 b é apresentado um processo de percorrer dentro do modelo para localizar um cliente que recebe uma solicitação. E na Figura 6 c é mostrado processo de registrar uma nova localização de um usuário em mudança de localidade para o exterior de sua localidade. E temos na Figura 6 d demonstra a técnica de particionamento, onde se busca diminuir o custo em um modelo hierárquico.

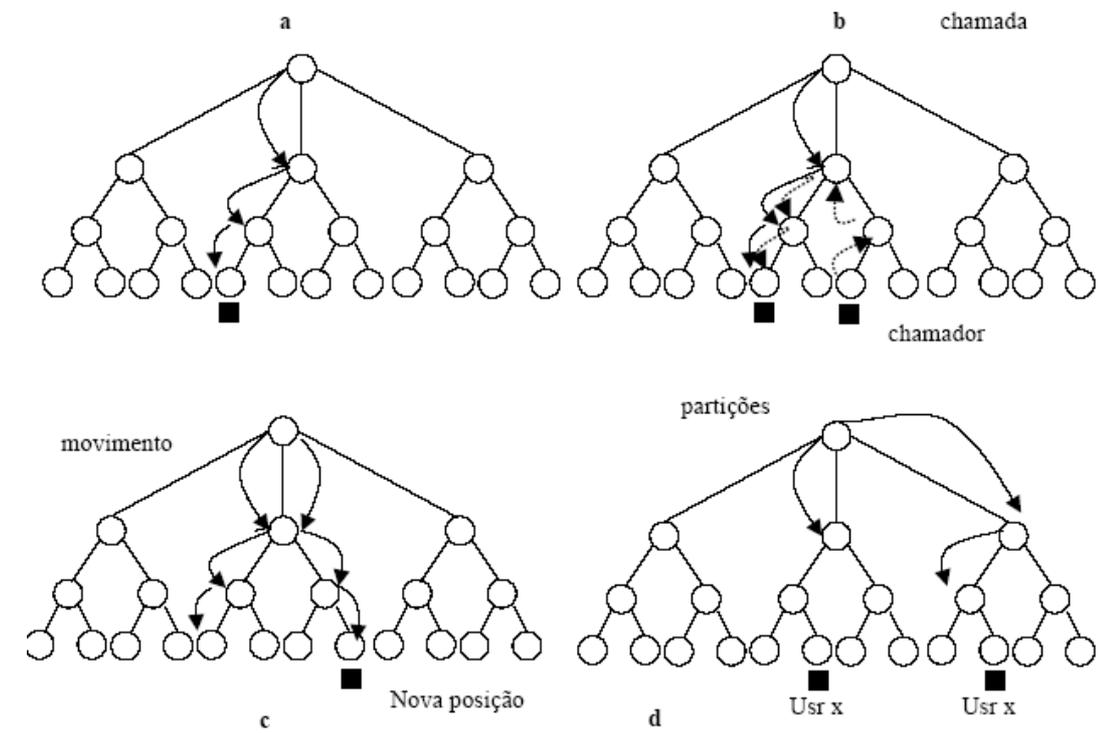


Figura 6 – Modelos hierárquicos.(Adaptada de[4]).

CAPÍTULO 6. O MERCADO DE SGBD MÓVEL

Nos dias atuais, grandes empresas visam um indispensável modelo de gerenciamento de seus negócios, por meio de operações de dados e transações mais remotas de seus postos de trabalho tradicional, as principais fábricas de aplicativos se mobilizaram para esta crescente e importantíssima ramificação da computação e aparecem neste ambiente oferecendo as mais diferentes ferramentas para acolher a essa solicitação e também a solicitação do cliente pessoal, que tem o crescimento significativo de um período pra cá devido à evolução, principalmente dos celulares.

Dentre os softwares para banco de dados móvel, segue alguns dos essenciais no mercado.

6.1 Sybase Anywhere Studio

O SQL Anywhere é um produto que contém como principal, o gerenciamento de dados e também contém efeito de resolver problemas de BD's inserido chamado UltraLite. Possui também a técnica de sincronia, com os próprios sistemas de Sybase, e pode ter comunicar com softwares de diferentes fabricantes[9]. Este possui sistema de BD's de modelo relacional, tem Adaptive Server e também o UltraLite. No caso do Adaptive, ele pode proporcionar níveis diferentes de usabilidade para clientes finais e administradores de BD's. Ele contém sistema de sintonização automática, com isso não exige tanto envolvimento do administrador de BD's. Possui um utensílio gráfico que proporciona uma simples administração de manusear um BD e também o meio de sincronia em plataformas de personal

computer e Unix. Uma vantagem adicionada para controle e possíveis programações de eventos admitem que o administrador possa a vir fazer configuração de seu servidor pra realizar ações, em período estabelecido ou também quando ocorrer algum tipo de evento, como por exemplo o caso de o disco atingir alguma taxa de sua capacidade total.

Considerando banco de dados menos densos, o Ultralite[9] é o mais indicado para dispositivos móveis. Confeccionado com a intenção de diminuir os requisitos e também as aplicações relacionadas diretamente por dados que visam preceitos de memória, como por exemplo, equipamento de fácil portabilidade, aparelhos com certa inteligência. Concede às empresas atribuir apenas partes de BD's que realmente tenham necessidades para certas aplicações. Também concede a aplicativos Java para qualquer tipo de plataforma que possua a máquina Java. Podem ser utilizados em vários tipos e perfis de produtos, porque tem integridade, modelagem e estrutura compatíveis. Alguns dados pertinentes ao Ultralite.

Seu banco de dados pode a vir possuir um espaço em disco de dois GB; Quatro Kbytes é limite estabelecido para as linhas; Banco de dados possui até mil tabelas; No caso das linhas tem por volta de sessenta e cinco mil; Dependendo do limite do BD, o número de linha pode ser alterado.

No *Anywhere Studio*, no caso de sincronia e replicar de informações são três os diferentes facultades-

Com base na sincronia o *Mobilink*[9] é o tipo de conexão de dados por meio de aplicativos que contenham em equipamentos móveis, pode ocorrer troca de dados automáticas com BD's de corporação; Em plataformas não iguais ele concede uma maneira que proporciona segurança, permite sincronizar informações de mão e

contra mão durante o período de contato. Se houver mudanças na base de dados central ela fará uma cópia no equipamento móvel, e a mesma coisa ao contrário, proporcionando monitoração, controle e integridade com relativo vigor. *Mobilink* também possui um procedimento de sincronia com BD's mais distantes ao *Sybase*, como outros tipos de bancos móveis. Também suporta a transmissões de dados por qualquer tipo de comunicação.

Em relação à réplica ela pode ser feita em um número bem limitado de BD's. Comumente não é viável para o meio que as máquinas estão distantes estejam em constantes mobilidades ou constantes falta de conexão aconteçam.

6.2 Oracle Lite

O *Oracle Lite*[10] é feito em cima do *Oracle 9i* sua estrutura é baseada em *Xml* e pode ser desenvolvido aplicações que venham a usar voz, compartilhamento de mensagens e acesso *wireless*. Algumas de suas aplicações podem ocorrer sem conexão no local.

Hoje o *Oracle Lite* é considerado uma plataforma essencial para poder realizar e ter controle sobre aplicações em equipamentos móveis, tem suporte Java em várias plataformas diferentes, como os *handhelds*, telefones celulares e os computadores de bolso.

Considerada uma base restrita, o *Oracle Lite* foi confeccionado para aparelhos de pequeno porte, pode a vir ocorrer réplicas de dados em um servidor maior que a sua base de dados com informações que tenham a base de dados com equipamentos móveis, concedendo que informações possam ser alteradas e operadas mesmo quando o cliente está sem conexão. Ao acontecer a reconexão à

base, as informações tem sincronia automática do equipamento móvel com a sua base de dados da corporação e também acontece o contrário. Dados guardados na central, pode em qualquer tempo ser acessado por qualquer clientes em diferentes lugares.

6.3 DB2 Everyplace

DB2 é considerado um BD relacional, possuindo perto de cento e cinquenta Kbytes. É um BD local quando o host está em desconexão ou quando tenta acessar o servidor com a ligação com a rede convencional. O DB2 possui sincronia em duas direções com os sistemas gerenciadores de corporações[11].

6.4 Microsoft SQL Server CE

Como no DB2 o Microsoft SQL pode ser usado como BD local, quando o host móvel se encontrar desconexo ou também quando o usuário tenta o acesso com o servidor com ligação com a rede convencional[12].

Algumas características primordiais que compõem o SQL Server.

Plataformas como *windows* e *linux* são suportadas; SQL é a sua sintaxe; Pode haver operacionalização com softwares de outros fabricantes; Para manter informações seguras tem o emprego de criptografia; Melhoria em processos que necessitem consultas e operações; Banco de dados com no máximo um mega Byte.

Um grupo de bibliotecas é a base do SQL Server, que tem algumas tecnologias como, Ado e Adoxce.

Na Tabela 2 é apresentada uma comparação entre estes produtos.

Produto	Sistema Operacionais	Espaço em disco	Ferramentas de sincronização	Tecnologias de desenvolvimento e gerenciamento
Sybase SQL Anywhere Studio 8.1	Windows 9x/Me/NT/2000/XP; Windows CE/Pocket PC; UNIX/Solaris/Linux; PalmOS	3Mb-8Mb	MobiLink SQL Remote	Sybase Informaker; PowerDesign; Sybase Central; Interactive SQL
IBM DB2 Everyplace 8.1	Windows 9x/Me/NT/2000/XP; Windows CE; Linux; Palm OS; Symbian 6	175 Mb (servidor) e 150Kb (cliente)	DB2 Everyplace Sync Svrer; DB2 Everyplace Sync Client	Mobile Application Builder; Mobile Devices Administration Center; DataPropagator
Microsoft SQL Server CE	Windows NT/2000/XP; Windows CE	800Kb-3Mb; 30Mb-45Mb (desenvolvimento)	Remote DataAccess; Replicação Intercalada (Merge Replication)	Microsoft Visual Studio.NET ADOCE; ADOXCE, OLE, OLE.NET
Oracle9i Lite 5.0.1	Windows 9x/NT/2000; Windows CE; Palm Os.	200Mb (sevidor)	Mobile Server; Message Genarator and Processor (MGP)	Mobile Developement Kit

Tabela 2 – Análise comparativa entre os produtos (Adaptada de[10,11,12,13]).

Uma parte bem complicada um sistema de informação é a escolha do sistema gerenciador de banco de dados. A questão é, neste universo dominado por batalhas entre as empresas fabricantes de software, como realmente acreditarmos em tudo que eles dizem. Uma questão fica em aberto, existe um banco de dados ideal e perfeito? Realmente é muito difícil dizer que existe, apesar de das apelações feitas pelos fabricantes e suas distribuidoras. Existem bons sistemas de gerenciamento de banco de dados como também existem os piores.

Entre os bons, cada um deles ressalta por algum motivo em especial. A pergunta é qual base de dados eleger, é a melhor que adapte à sua necessidade, principalmente no que diz respeito à concorrência de dados e nas características físicas de rede na qual rodará. Segue agora algumas características dos principais bancos de dados no mercado. Primeiramente vamos falar do Oracle, este é um dos mais conhecidos entre os bancos de dados. O Oracle é potente e extremamente eficiente. O Oracle suporta todas as funções esperadas de um servidor conciso: uma linguagem de extensão ao SQL muito boa, que permite implementar recursos com atividades, como os triggers e os procedimentos registrados, com uma excelente integridade referencial bastante potente. Para se tornar mais eficiente usa o artifício de partição, de réplicas e também admite banco de dados distribuído. O software do servidor pode ser usado em alguns sistemas operacionais. O maior inconveniente do Oracle é o preço, inclusive a licença do Personal Oracle não é barata. Um problema é sempre a necessidade de sintonia, ao se configurar mal o Oracle, pode torná-lo muito lento.

Alguns problemas são encontrados no SQL Server, páginas são bloqueadas, o crescimento de dispositivos são manuais, sua página tem tamanho fixo e bem estreito. A principal qualidade é seu preço e facilidade de se adaptar aos produtos da Microsoft. Tem mais um outro ponto a favor do SQL Server é a sua interface universal, ele é um dos primeiros a aceitá-lo. Sua linguagem é sempre uma questão a ser olhada com muita atenção e a integridade referencial é um pouco restrita.

O DB2 tem uma arquitetura parecida com a do Oracle. Várias plataformas são aceitas pelo DB2, outra questão parecida com o Oracle é o valor, a grande parte de seus usuários é do mercado industrial. Tem muita potência no SQL do DB2, os triggers tem uma fantástica forma de se implementar. É sempre difícil saber qual

sistema gerenciador de banco de dados optar, no caso do DB2 tem o seu preço elevado, o SQL server é mais barato, mas por ter aplicações em tripla camada complica em regras de negocio. O Oracle tem seu preço acessível, mas exige uma qualidade muito grande de seu administrador. A seguir na Tabela 3 é mostrada uma pesquisa sobre tecnologia de banco de dados, realizada pela ComputerWorld, em uma escala de 0 a 5, seus clientes deram suas opiniões.

SGBD	Satisfação	Qualidade	Serviço/Suporte	Custo de propriedade	Conforto com vendedor
DB2	4,17	4,19	4,2	3,86	4,22
SQL Server	3,71	3,71	3,55	3,71	3,75
Oracle	3,62	3,78	3,55	3,27	3,65
Sybase	3,41	3,47	3,33	3,4	3,39

Tabela 3. Pesquisa sobre Sgbd's no mercado (Adaptada de [14]).

CONCLUSÃO

A meta desta apresentação foi mostrar como os sistemas móveis se comportam em um meio de computação em constante mobilidade, mostrando também problemas enfrentados por tal tecnologia. E também ressaltar como esta computação está se expandindo de forma escalar, fornecendo ao cliente a informação em qualquer parte em que ele se encontre.

O tema proposto teve a intenção de apresentar a computação móvel em um ambiente bem complexo, não entrando a fundo em questões de redes, sua estrutura e qualquer outro requisito desta nova e crescente ramificação da computação.

Não é uma coisa fácil inserir SGBD neste ambiente mobilidade constante. Neste ambiente, os dados não são tão concisos e íntegros como em uma computação convencional, pois dependem de estarem conectados. Dependem claramente de estarem conectados para poder ser feitas transações entre seus bancos de dados móveis e os modelos de bancos centralizados e também distribuídos. Sendo assim, inúmeras pesquisas estão sendo desenvolvidas no intuito de se realizar modelos de transações que possam dar garantias de que operações móveis mantêm seus requisitos básicos.

A indústria é a que mais se beneficia com a computação móvel, e pesquisas não param de acontecer, com isso é um mercado que cresce cada vez mais e em franca expansão principalmente tecnologia de telefonia celular. Com este crescente esta tecnologia já abrange muitos segmentos como:

- Companhias de seguro;
- Serviços de emergência;
- Controle de tráfego;
- Empresas de táxi;

- E-commerce.

E cada vez mais, estaremos acessíveis à utilização desses serviços.

REFERÊNCIA BIBLIOGRAFICAS

- [1]DATE, C.J.. **Introdução a Sistemas de Banco de Dados**, 2000 Editora Campus Ltda.
- [2]SILBERSCHATZ, Abraham;KORHT, Henry F.;SUDARRSSHAN S. **Sistema de Banco de Dados**.Terceira edição, Editora Makron Books, 1999.
- [3]MATEUS,Geraldo R.; Loureiro, Antonio A. F. **Introdução a Computação Móvel**. 11ª Escola de Computação, Rio de Janeiro, RJ, 1998.
- [4]ELSMARI,R.Navate,S.B.. **Sistemas de Banco de Dados;Fundamentos e Aplicações**. Terceira Edição.Rio de Janeiro.LTC, 2002.
- [5]COUCEIRO, Luiz Antonio Carneiro da Cunha; BARRENECHA,Hugo Fernando Spencer. **Sistemas de Gerência de Banco de Dados Distribuídos**. Edição Original,1984,Livros Técnicos e Científicos Editora S. A.
- [6]LEITE,Leonardo L.Pereira. **Introdução aos Sistemas de Gerencia de Banco de dados**,Editora Edgar Blucher,1980.
- [7] SQL.Disponível em <<http://dev.mysql.com/doc/mysql/pt/apache.html>> Acesso em: 09 jun. 2005.
- [8] MELO,Rubens.;SILVA,Sideney Dias; TANAKA, Asteiro K.Banco de Dados em Aplicações Cliente-servidor.Sexta Edição,Editora Infobook,1998.
- [9] SYBASE.Disponível em <http://www.sybasebrasil.com/sqlanywherestudio> Acesso em: 01 jun.2005.
- [10]ORACLE Corporation Oracle9i database.Disponível em: <<http://www.oracle.com/lang/pt/database/index.html> >. Acesso em: 09 jun. 2005.

- [11] IBM Corp. **DB2 Everyplace.** Disponível em:
<<http://www.ibm.com/software/data/db2/everyplace/>>. Acesso em: 10 jun. 2005.
- [12] MICROSOFT,SQL Server.Disponível em: <
<http://www.micosot.com/products/sqlserver>> Acesso em 01 jun.2005.
- [13] COMPUTERWORLD,Disponível em < <http://www.computerworld.com.br>> Acesso em
04 jun.2005.

