Sistemas de Banco de Dados Paralelos

Aluno: Márcio Angelo B. de Lira
mabl@cin.ufpe.br
Prof.ª : Bernadette Farias Lóscio
Ana Carolina Salgado

Pós-Graduação em Ciência da Computação
Universidade Federal de Pernambuco (UFPE)

Sistemas Paralelos de BD

Roteiro

- Arquiteturas de sistemas paralelos de banco de dados
- Localização de Dados Paralelos
- Processamento de Consulta Paralela
- Balanceamento de Carga
- Clusters de Banco de Dados
- Conclusão

Motivação

- Sistemas paralelos melhoram as velocidades de processamento de E/S usando vários processadores e discos em paralelo.
- As máquinas paralelas estão se tornando cada vez mais comuns contribuindo para o avanço dos estudos em sistemas de banco de dados paralelos.

Objetivo

- O objetivo deste seminário é a continuidade da disciplina de Banco de dados Distribuído e Móveis, estimulando o aluno a realizar uma pesquisa minuciosa sobre os sistemas de banco de dados paralelos e compartilhar com a turma o conhecimento adquirido.

Principais conceitos

•Paralelismo de BD: É um tipo especial de sistema distribuído feito de um número de nós compartilhando processadores, memórias e discos conectados por uma rede muito rápida.

•Localização de Dados: Separação de dados de um mesmo banco de dados em diversos nós a fim de aumentar o desempenho e disponibilidade de um banco de dados

•Balanceamento de Carga: Técnica de dividir o processamento em dois ou mais servidores de BD como recurso de tolerância a falhas e ganho de desempenho.

•Cluster de BD: É um conjunto de nós independentes interligados para compartilhar recursos e formar um único sistema.

Estado da Arte

- Aplicações de dados muito grandes (centenas de terabytes ou pentabytes) requerem suporte a esses bancos.
- Exemplos: e-commerce, data warehousing e data mining.
- Apoia grandes bancos de dados tanto para sistemas OLTP como para OLAP, pode ser abordada através da combinação de computação paralela e gestão de BD distribuído.
- A idéia é construir um computador muito potente partindo de muitos computadores pequenos, a um custo muito menor do que computadores equivalentes de grande porte.

Estado da Arte (cont.)

- Aumentar o desempenho (através de paralelismo) e disponibilidade (através da replicação).
- Modelo relacional proporciona uma boa base para os dados baseado em paralelismo.
- Arquiteturas de sistemas paralelos são : Memória compartilhada, Disco compartilhado, Nada compartilhado.

Arquiteturas de sistemas paralelos de BD

Objetivos

- Aumentar o desempenho e disponibilidade.
- Preocupação entre as décadas de 70 e 80.
- Princípios cobertos pelos os mesmos do SGBD distribuído.
- Alto desempenho, alta disponibilidade e extensibilidade.

Arquiteturas de sistemas paralelos de BD

Alto desempenho

Gerenciamento de dados paralelos, otimização da consulta e balanceamento de carga.	Diminui o tempo de resposta das transações, usando paralelismo de intra-consulta.	Divididir a carga igualmente entre todos os processadores.	Dependendo da arquitetura pode ser alcançado estaticamente (banco físico) ou dinamicamente em tempo de execução.

Arquiteturas de sistemas paralelos de BD

Alta disponibilidade

Componentes redundantes, aumentam a disponibilidade de dados e tolerância a falhas.	A replicação de dados em diversos nós é útil para suportar Failover (técnica de tolerância a falhas ...).	Problemas: sobrecarga do server que terá uma cópia disponível.	Soluções: cópias de particionamentos de tal modo que eles possam também ser acessados em paralelo.

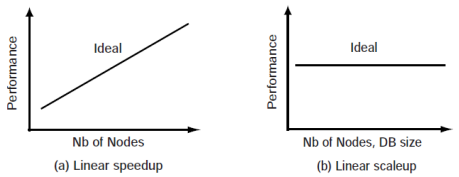
Arquiteturas de sistemas paralelos de BD

Extensibilidade

É a capacidade de expandir o sistema suavemente, através da adição de processamento e armazenamento para o sistema.	Speedup: aumento linear no desempenho para uma base de dados com tamanho constante enquanto que o número de nós são aumentados linearmente.	Scaleup: se refere a um desempenho sustentado por um aumento linear no tamanho do banco de dados e número de nós.

Arquiteturas de sistemas paralelos de BD

- **Extensibilidade (cont.)**



Arquiteturas de sistemas paralelos de BD

- Arquiteturas básicas:
  - Memória compartilhada;
  - Disco compartilhado e;
  - Nada-compartilhado.
- Arquiteturas híbridas:
  - NUMA e;
  - cluster.

Arquiteturas de sistemas paralelos de BD

Arquiteturas básicas

•Memória compartilhada

- Qualquer processador tem acesso a qualquer módulo de memória ou unidade de disco através de uma interligação rápida (por exemplo, um barramento de alta velocidade)
- Todos os processadores estão sob o controle de um único sistema operacional.
- O DB2 foi o primeiro exemplo, utilizando um IBM3090 com 6 processadores [Cheng et al., 1984].

Arquiteturas de sistemas paralelos de BD

- **Memória compartilhada (cont.)**

Vantagens:

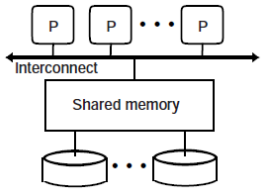
- Simplicidade (paralelismo inter-consulta e intra-consulta) e
- Facilidade para balanceamento de carga.

Desvantagens

- Alto custo (hardware bastante complexo);
- Extensibilidade limitada (até 16 processadores) e;
- Baixa disponibilidade (uma falha de memória pode afetar a maioria dos processadores).

Arquiteturas de sistemas paralelos de BD

Ilustração da arquitetura de Memória compartilhada



Arquiteturas de sistemas paralelos de BD

Arquiteturas básicas(cont.)

- **Discos compartilhados**
  - Qualquer processador tem acesso a qualquer unidade de disco através da interconexão.
  - Cada processador pode acessar páginas de banco de dados sobre o disco compartilhado e carregalas em sua própria memória principal.
  - Processadores diferentes podem acessar a mesma página em modos de atualização conflitantes, para isso a consistência da cache global é necessária.
  - Solução: usar um gestor de bloqueio distribuído.
  - O Oracle possui uma implementação bem eficiente.

Arquiteturas de sistemas paralelos de BD

- **Discos compartilhados (cont.)**

Vantagens:

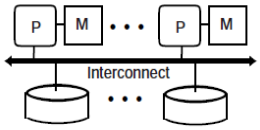
- Menor custo, extensibilidade elevada;
- Disponibilidade, balanceamento de carga e;
- Fácil migração de sistemas centralizados.

Desvantagens:

- Exige protocolos específicos, tais como bloqueio distribuído de duas fases.
- Além disso, manter a consistência do cache pode implicar em um overhead de comunicação entre os nós.

Arquiteturas de sistemas paralelos de BD

Ilustração da arquitetura de Discos Compartilhados



Arquiteturas de sistemas paralelos de BD

Arquiteturas básicas (cont.)

- **Nada compartilhado**
  - Cada processador tem acesso exclusivo para a sua memória principal e da unidade de disco.
  - Cada nó pode ser visto como um site local (com a sua própria base de dados e software).
  - Cada nó (processador, memória e disco) está sob o controle da sua própria cópia do sistema operacional.
  - O primeiro grande produto SGBD paralelo foi o **Computer Teradata,** banco de dados que poderia acomodar até mil processadores em sua versão inicial.

Arquiteturas de sistemas paralelos de BD

- **Nada compartilhado(cont.)**

Vantagens:

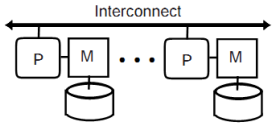
- Menor custo (melhor do que a do disco partilhado que requer uma interconexão especial para os discos).
- Extensibilidade elevada (favorece o crescimento suave incremental de novos nós).
- Grande disponibilidade (replicação de dados em vários nós).

Desvantagens:

- O equilíbrio de carga é mais difícil de alcançar.
- A adição de novos nós no sistema exige uma reorganização do banco de dados para lidar com as questões de balanceamento de carga.

Arquiteturas de sistemas paralelos de BD

Ilustração da arquitetura Nada Compartilhado



Arquiteturas de sistemas paralelos de BD

- **Arquiteturas Híbridas**

Tentam obter as vantagens das diferentes arquiteturas como a eficiência e simplicidade da memória compartilhada e a extensibilidade e custo das arquitetura de disco compartilhado e nada compartilhado.

Arquiteturas de sistemas paralelos de BD

- **Arquiteturas Híbridas (cont.)**

NUMA

- Fornece a uma arquitetura de Memória Compartilhada um modelo de programação e todos os seus benefícios, tornando uma arquitetura escalável, com memória distribuída.
- Qualquer processador tem acesso a todas as memórias dos outros processadores.
- O argumento forte para NUMA é que ela não necessita de qualquer reescrita do software de aplicação.

Arquiteturas de sistemas paralelos de BD

- **Arquiteturas Híbridas (cont.)**

  Cluster

  - Conjunto de nós de servidores independentes interligados para compartilhar recursos e formar um único sistema compartilhando.
  - Na sua forma mais barata a interconexão pode ser uma rede local ou um padrão rápido de interconexões de clusters (ex. Myrinet e Infiniband) que oferecem alta largura de banda em Gigabits/seg com baixa latência para o tráfego de mensagens.
  - Amplamente utilizados pois eles podem fornecer a melhor relação custo/desempenho e escalabilidade com milhares de nós.

Arquiteturas de sistemas paralelos de BD

- Principais tecnologias para compartilhar discos em um cluster:
  - Armazenamento anexado à rede **(NAS):** é um dispositivo dedicado para discos compartilhados em uma rede (normalmente TCP/IP) utilizando um sistema distribuído, protocolo de sistema de arquivos.
  - Adequado para aplicações de baixo custo, como backup de dados e arquivamento de discos rígidos do PC.
  - Porém é relativamente lento, torna-se um gargalo com muitos nós.

Arquiteturas de sistemas paralelos de BD

Principais tecnologias para compartilhar discos em um cluster:

- Área de armazenamento de rede **(SAN**): fornece funcionalidade similar a NAS, mas com uma interface de baixo nível.
- Usa um protocolo baseado em blocos, tornando assim mais fácil de gerenciar a consistência da cache (em nível de bloco).
- Os discos em uma SAN são ligados à rede, em vez de um barramento como acontece em NAS.
- Fornece transferência de dados elevada e pode escalar um grande número de nós.

Arquiteturas de sistemas paralelos de BD

- **Arquiteturas Híbridas (cont.)**
  - Vantagem do NUMA: melhor desempenho com modelo memória compartilhada, de programação simples que facilita a administração e otimização de banco de dados.
  - Vantagem do clusters: usando nós de PC padrão e interconexões, pode proporcionar uma melhor relação de custo total e performance, usando o modelo nada compartilhado, eles podem escalar configurações muito grandes com milhares de nós.

Localização de Dados Paralelos

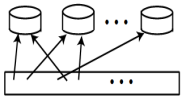
- Se asemelha com a fragmentação de dados em bancos de dados distribuídos.
- Como os dados são muito maiores do que os programas, a execução deve ocorrer, sempre que possível, onde os dados residem.
- Um problema é evitar a contenção de recursos, o que pode resultar na sobrecarga do sistema inteiro (ou seja um nó acaba fazendo todo o trabalho enquanto os demais permanecem ociosos).

Localização de Dados Paralelos

- Finalidade de maximizar o desempenho do sistema, reduzindo a quantidade total de trabalho e do tempo de resposta das consultas individuais.
- O DBA é responsável por examinar periodicamente os acessos aos fragmentos e, quando necessário, mover e reorganizar os fragmentos.
- Uma boa alternativa para a localização de dados é o particionamento completo, em que cada relação é horizontalmente fragmentada em todos os nós do sistema.

Localização de Dados Paralelos

Round-robin (Rodízio)

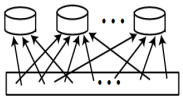


(a) Round-Robin

- É a estratégia mais simples, assegura a distribuição de dados uniforme.
- Varre a relação em qualquer ordem e envia a *i-ésima* tupla ao disco de número (*i mod n*).
- Essa estratégia permite o acesso seqüencial a uma relação em paralelo.
- No entanto, o acesso direto as tuplas individuais, com base em um predicado, exige acesso a relação inteira.

Localização de Dados Paralelos

Hash

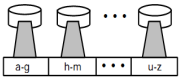


(b) Hashing

- Aplica-se uma função hash para algum atributo tornando-o atributo do particionamento.
- Esta estratégia é mais adequada para consultas pontuais baseadas no atributo de particionamento.
- É útil para varreduras sequenciais da relação inteira.
- Não adequada para consultas de intervalos.

Localização de Dados Paralelos

Range (Faixa ou intervalo)



(c) Range

- Distribui tuplas com base nos intervalos de valor de algum atributo.
- É bem adequada para consultas por abrangência.
- Uma consulta com um predicado entre "A1 e A2" pode ser processado por único nó apenas contendo tuplas cujo valor está na faixa [A1, A2].
- Normalmente a consulta é enviada a um disco em vez de todos os discos.

Processamento de consultas paralelas

Paralelismo de Consultas

- O objetivo do processamento de consulta em paralelo é transformar consultas em planos de execução que pode ser eficientemente executadas em paralelo.

- Permite a execução paralela de várias consultas geradas por transações simultâneas, a fim de aumentar a taxa de transferência transacional.

- Duas técnicas são usadas: Paralelismo intra-operador e Paralelismo Inter-operador.

Processamento de consultas paralelas

Paralelismo Consultas (cont.)

- Inter-consulta: Diferentes consultas ou transações são executadas em paralelo umas com as outras.
  - Forma simples, usa modelo de paralelismo de memória compartilhada

- Intra-consulta: Uma única consulta é executada em paralelo em vários processadores e discos.
  - Usada em consultas complexas de longa duração.

Processamento de consultas paralelas

Intraconsulta

Paralelismo intra-operador

- Agiliza o processamento colocando em paralelo a execução de cada operação individual.

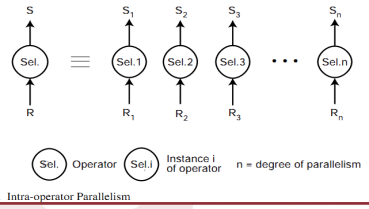
- É baseado na decomposição de um operador de um conjunto de operadores sub-independente, chamados de instâncias do operador.

- Cada instância operador irá então processar uma partição da relação, também chamado de balde.

Processamento de consultas paralelas

Paralelismo intra-operador (cont.)

- O operador de seleção pode ser diretamente decomposto em vários operadores de seleção, cada um em uma partição diferente.



Intra-operator Parallelism

Processamento de consultas paralelas

Intraconsulta (cont.)

- **Paralelismo Inter-operador**
  - Agiliza o processamento de uma consulta executando em paralelo as diferentes operações em uma expressão de consultas.

  Duas formas de inter-operador:
  - Pipeline: vários operadores com uma ligação do produtor-consumidor são executados em paralelo.
  - Independente: é alcançada quando não há nenhuma dependência entre os operadores que são executados em paralelo.

Processamento de consultas paralelas

Algorítmos Paralelos para Processamento de Dados

É importante para um processamento eficiente dos operadores de BD (ou seja, os operadores da álgebra relacional) e consultas de BD que combinam múltiplos operadores.

- **Algoritmo paralelo loop aninhado (PNL);**
  - É o mais simples e mais geral.
  - Compõe o produto cartesiano das relações R e S em paralelo.

Processamento de consultas paralelas

- **Algoritmo paralelo junção associativa (PAJ);**

  Aplica-se apenas no caso de equijoin com uma das relações de operando particionado de acordo com o atributo de junção.

- **Algoritmo paralelo hash (PHJ).**

  Pode ser visto como uma generalização do algoritmo de junção paralela associativa. Também se aplica no caso de equijoin mas não exige qualquer particionamento específico das relações.

Processamento de consultas paralelas

Otimização de consulta paralela

Exibe semelhanças com o processamento de consulta distribuída, se divide em três componentes.

- **Pesquisa Espacial**
  - Os planos de execução são captados por meio de árvores de operadores, que definem a ordem em que os operadores são executados.
  - Podem ser executadas em pipeline que requerem relações temporários para serem materializada.

Processamento de consultas paralelas

Otimização de consulta paralela (cont.)

- **Modelo de Custo**
  - O otimizador é responsável por estimar o custo de um plano de execução.
  - É composto de duas partes: arquitetura dependente e arquitetura independente.
- **Estratégia de Pesquisa**
  - Não diferencia de qualquer otimização de consulta centralizada ou distribuída.
  - Porém, o espaço de busca tende a ser muito maior porque há mais parâmetros que os planos de execução em paralelo.

Balanceamento de Carga

Definições

- É uma técnica para distribuir a carga de trabalho uniformemente entre dois ou mais computadores, enlaces de rede, processadores, discos rígidos ou outros recursos.
- Otimiza a utilização de recursos, maximizando o desempenho, minimizando o tempo de resposta e evitando sobrecarga.
- É crucial para o desempenho de um sistema paralelo.
- Pode sofrer vários problemas incorrendo em tempo de execução.

Balanceamento de Carga

Problemas de execução paralelas

- **Inicialização**
  - Criação do processo, inicialização e comunicação.
- **Interferências**
  - Ocorre quando vários processadores acessam simultaneamente o mesmo recurso de hardware ou software.
- **Inclinação**
  - Variação no tamanho da partição ou variação na complexidade dos operadores.

Balanceamento de Carga

Balanceamento de Carga Inter-Operador

- É necessário escolher, para cada operador, quantas e quais os processadores para atribuir a sua execução.
- Na arquitetura nada-compartilhado é determinado dinamicamente (imediatamente antes da execução) o grau de paralelismo e a localização dos processadores para cada operador.
- É a base para a escolha do conjunto de processadores que serão utilizados para a execução de consulta

Balanceamento de Carga

Balanceamento de Carga Inter-Operador (cont.)

- Na arquitetura disco compartilhado e memória compartilhada, há mais flexibilidade, já que todos os processadores têm igualdade de acesso aos discos.
- Não havendo necessidade para o particionamento relação física, qualquer processador pode ser atribuído a qualquer operador.

Balanceamento de Carga

Balanceamento de Carga intra-consulta

- Até certo ponto as técnicas para qualquer carga balanceada intra ou inter-operador pode ser combinada.
- Nos sistemas híbridos (NUMA ou cluster) os problemas de balanceamento de carga são exacerbados porque têm de ser dirigida a dois níveis, localmente entre os processadores de cada nó de memória partilhada e globalmente entre todos nós.
- A solução geral para o balanceamento de carga em sistemas híbridos é o modelo de execução chamado Processamento Dinâmico (DP).
- O modelo de execução DP é baseada em: ativações, filas de ativação e segmentos.

Clusters de Banco de Dados

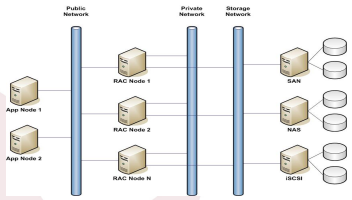
Definições

- Conjunto de dois ou mais Computadores que fornecem uma alternativa rentável para supercomputadores ou multiprocessadores fortemente acoplados.
- Usados com sucesso em computação científica, recuperação de informação web (motor de busca Google) e data warehousing.
- A Oracle possui uma solução bastante comercializada o RAC.
- Real Application Cluster (RAC) é uma solução de alta disponibilidade que fornece a capacidade de cluster em todos os seus nós.

Clusters de Banco de Dados

RAC (Oracle)

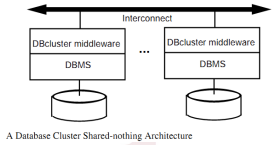
Componentes de softwares, hardwares de rede e storage que proporcionam todos os pontos fortes em alta disponibilidade, performance e balanceamento de carga.



Clusters de Banco de Dados

Arquitetura de Cluster de Banco de Dados

- Como em um SGBD paralelo, o middleware de cluster de BD tem várias camadas de software: balanceador de carga de transação, gerente de replicação, processador de consulta e gerente tolerância a falhas.



A Database Cluster Shared-nothing Architecture

Clusters de Banco de Dados

Arquitetura de Cluster de BD (cont.)

- Uso de SGBD "blackbox" em cada nó.
- Recursos de gerenciamento de dados paralelos devem ser implementados através do middleware.
- Na sua forma mais simples, um cluster de BD pode ser visto como um sistema de multidatabase.

Clusters de Banco de Dados

Replicação

- Para melhorar o desempenho e disponibilidade, os dados podem ser replicados em diferentes nós, usando o SGBD local.
- Em um cluster de BD, o sistema de interconexão e comunicação rápida pode ser explorado para apoiar uma cópia serializável, proporcionando maior escalabilidade.

Clusters de Banco de Dados

Replicação (cont.)

Protocolo de replicação preventiva

- É um protocolo para a replicação preguiçosa distribuído em um cluster de banco de dados.
- Cada transação T de entrada para o sistema tem um ts timestamp cronológico.
- Em cada nó, um atraso de tempo é introduzido antes de iniciar a execução de T. Um sistema síncrono com a computação delimitada e tempo de transmissão assumido.

Clusters de Banco de Dados

O balanceamento de carga

• O "melhor" nó é definido como aquele com carga de transação mais leve.

•O balanceador também garante que cada execução da transação obedeça as propriedades ACID, e em seguida, sinaliza para o SGBD confirmando ou anulando a transação.

•O gestor de replicação gere o acesso aos dados replicados e assegura forte consistência de modo a que as operações que atualizam dados replicados sejam executados na mesma ordem em cada nó.

Clusters de Banco de Dados

Processamento de Consulta

•Explora tanto paralelismo inter-consulta quanto intra-consulta.

•Com paralelismo inter-consulta, o processador de consulta encaminha cada consulta submetida a um nó e após a conclusão da consulta, envia os resultados para o aplicativo cliente.

•À medida que os SGBDs não são caixa-preta de clusters, eles não podem interagir uns com os outros, a fim de processar a mesma consulta.

Clusters de Banco de Dados

Tolerância a Falhas

•Falhas precisam ser detectadas um mecanismo de pulsação compara a nova associação com a associação anterior da transação.

•Falha de réplica detectada (ações devem ser tomadas no cluster de banco de dados).

•Estas ações são parte do processo de Failover, os clientes com operações em aberto se conectam a um nó com nova réplica e reenviam as últimas transações.

Clusters de Banco de Dados

Tolerância a Falhas (cont.)

•Failover é tratado para abortar todas as operações em curso para evitar situações de conflitos.

•Realizar recuperação após a falha.

• Manter o acesso a dados consistentes apesar das falhas.

•O gerente de tolerância a falhas fornece recuperação on-line de Failover.

Clusters de Banco de Dados

Tolerância a Falhas (cont.)

Questionamentos

•Como manter a consistência dos dados apesar das falhas ?

Se utilizando da redundância das Réplicas.

•Com operações em aberto, como é executado o failover?

Aborta as transações e reencaminha para nova Réplica assumida.

•Quando uma réplica é reintroduzida, ou uma réplica recente (fresca) é introduzida no sistema, o estado atual da base de dados tem que ser recuperado?

Depende, é verificado se a réplica tem todas as atualizações das demais, caso não para-se as transações, atualiza-se a réplica e recomeça as transações novamente.

Conclusões

•Paralelismo é a principal solução viável para suportar bancos de dados muito grandes dentro de um único sistema.

•Promessas de alta performance, alta disponibilidade, extensibilidade com baixo custo e bom desempenho.

•Arquiteturas paralelas de memória partilhada, discos, nada compartilhado e arquiteturas híbridas devem ser planejadas.

•Arq. Híbrida, como **NUMA** e **Cluster** podem combinar eficiência e simplicidade da memória-compartilhada e a extensibilidade e custo de disco-compartilhado ou nada-compartilhado.

Conclusões

•Técnicas paralelas de gerenciamento de dados e técnicas de banco de dados distribuídos geram ganhos de desempenho e disponibiliadade.

•Um cluster de banco de dados é um importante tipo de sistema de banco de dados paralelo.

•Novas pesquisas para novas técnicas de replicação, balanceamento de carga, processamento de consultas e tolerância a falhas devem surgir.

Referências utilizadas

- Principles of Distributed Database Systems, M. Tamer Özsu & Patrick Valduriez, Editora Springer, 3rd. Edition, 2011.
- Sistema de Banco de Dados, KORTH, SILBERSCHATZ, SUDARSHAN. Rio de Janeiro: Campus, 5ª edição, 2006.
- Artigo técnico da Oracle, Oracle Database 11g com Alta Disponibilidade, William Hodak, colaboradores: Sushil Kumar, Ashish Ray, Outubro de 2007.

Obrigado!