

ESTABELECIMENTO DE PADRÕES PARA BANCO DE DADOS EM GRADE COMPUTACIONAL

Patterns establishment for database on grid computing

Celso Henrique Poderoso de Oliveira e Maurício M. Almeida

CEETEPS – Centro Paula Souza

Rua dos Bandeirantes, 169 - Bom Retiro – São Paulo-SP

cpoderoso@uol.com.br, malmeida@inteligenciaartificial.eti.br

RESUMO

Este artigo analisa a situação atual dos bancos de dados em Grades Computacionais e identifica os padrões de serviços para banco de dados. Inicia abordando o estado da arte das Grades Computacionais e dos sistemas gerenciadores de banco de dados e como os fornecedores de banco de dados irão adaptar seus produtos à esta nova necessidade. Para isso, requisitos de banco de dados e dos serviços da Grade Computacional são destacados, com objetivo de identificar o padrão para o banco de dados federalizado.

Palavras-chave: sistema gerenciador de banco de dados, banco de dados federalizado, OGSA, grade computacional, arquitetura orientada a serviços.

ABSTRACT

This paper is about the present situation of database on Grid Computing and it identifies the standard services for databases. It begins coming to grips with the state of art of Grid Computing, database management systems and how database suppliers will adapt their products to this new need. So, Grid Computing service and database computing requirements are highlighted to identify the pattern for a federalized database.

Keywords: database management system, federalized database, OGSA, grid computing, oriented architecture for services.

c) Grade: grupo de recursos heterogêneos, distribuídos e integrados compartilhando diversos recursos como se fossem um único e que utilizam redes de altíssima velocidade. Permite balanceamento de carga, segurança de acesso e é tolerante a falhas (FOSTER, 2001).

Neste artigo, pretendemos apresentar o estado atual da tecnologia de computação em grade, no que diz respeito à utilização dos sistemas gerenciadores de bancos de dados (SGBD) comerciais e identificar como a comunidade acadêmica trabalha para integrar estes produtos à Grade Computacional.

2. COMPUTAÇÃO EM GRADE

Ainda na década de 1960, imaginou-se que o acesso às informações e ao processamento seria realizado da mesma forma como hoje em dia utilizamos a energia elétrica ou o telefone. Ou seja, da mesma forma como não se precisa de uma usina geradora de energia em casa para ter acesso à energia elétrica, também não haveria necessidade de um computador para ter acesso ao processamento e armazenamento de dados (OLIVEIRA, 2004).

Este tipo de computação deveria ser elaborado exatamente como as grades de energia elétrica. Assim, quando fosse necessário utilizar um recurso, simplesmente o usuário se conectaria a grandes servidores, que forneceriam todo o poder de processamento e armazenamento que precisasse. Haveria uma taxa que deveria ser paga de acordo com o uso, exatamente como acontece nos serviços de telefonia ou energia elétrica.

Este modelo é claramente mais distribuído e atenderia melhor às necessidades dos usuários. Contudo, há uma série de etapas que se devem ultrapassar para que isso se torne realidade. Padrões devem ser estabelecidos para que as aplicações possam trocar informações e os servidores tenham condições de compreender as requisições e enviar os resultados.

2.1. Abrangência

Uma Grade Computacional pode ser formada a partir de diversos níveis de utilização. Conceitualmente divide-se em:

- Local: são grades internas de uma única empresa. Também conhecida por intragrid, enterprise ou campus.

- Regional: são grades formadas entre empresas parceiras. Também conhecida por extragrid ou partner.

- Global: são grades amplas, formadas por grandes redes interligadas, com abrangência em todo mundo. Também conhecida por intergrid (CHEDE, 2004).

2.2. Características

Podemos diferenciar as Grades Computacionais em dois grandes grupos:

- Científicos: utilizados para testes e avaliação de teorias e modelos, representa a maior quantidade de grades ativas atualmente. O desafio tem sido o de unir diversos conhecimentos e processar as informações em altas velocidades.

Dado que as fontes de informação encontram-se dispersas, é necessário que haja integração entre elas, para se chegar a resultados satisfatórios (GIL, 2004).

- Uso geral: diversas empresas e segmentos poderão fazer uso da grade computacional para otimizar o uso dos recursos computacionais, diminuindo significativamente o investimento em novos equipamentos. Além disso, alguns processos empresariais tendem a utilizar altas cargas de processamento. Simulações e cenários estão entre estas aplicações.

Uma Grade Computacional deve ter o menor número possível de pré-requisitos, visto que o objetivo é integrar recursos heterogêneos. Por agregar estes

física dos arquivos, contém mecanismos de cobrança por uso e acesso, além de guardar informações de usuários e recursos.

Este middleware é capaz de selecionar, criar e manter réplica dos dados, requisito importante quando se trabalha com distribuição de dados. Apesar de haver alguma tendência a utilizá-lo em conjunto com bancos de dados para distribuição de informações, há algumas limitações que precisam ser vencidas, como a transferência dos dados para o servidor. Esse processo tende a gerar perda no desempenho e escalabilidade das aplicações.

Os demais middlewares têm utilização e características específicas, que não diferem muito do objetivo dos anteriores.

2.5. Identificação

Deve-se identificar se uma tecnologia é classificada como uma Grade Computacional (FOSTER, 2002):

- Recursos não devem estar subordinados a um controle central: uma vez que a grade integra diversos recursos, usuários e domínios, caso esteja vinculado a um controle único, tem-se um sistema de gerenciamento local e não uma grade.

- Utilização de protocolos e interfaces padronizados, de uso múltiplo e abertos: como a grade trabalha com recursos heterogêneos, é necessário adotar padrões abertos de comunicação. Do contrário haverá um sistema de aplicação específica.

- Entrega de serviços de alta qualidade: o objetivo da grade é fornecer poder de processamento e armazenamento onde se integrem diversos recursos dispersos pela rede.

Caso o resultado final não seja melhor que a soma das partes, não faz sentido utilizá-la.

2.6. Organização virtual

Entende-se como um grupo de organizações ou indivíduos que compartilham recursos de forma controlada. Desta forma, os membros da comunidade podem colaborar entre si para atingir o objetivo compartilhado (FOSTER, 2001). Ao agir por interesses comuns, empresas e pessoas podem definir o que, como e quando compartilhar tais recursos. Isso quer dizer que teremos uma grade apenas quando permitimos que outras pessoas ou empresas acessem indiscriminadamente nossos recursos (processamento, armazenamento etc.).

As Organizações Virtuais são formadas por interesses comuns. Desta forma, o compartilhamento dos recursos será mais próximos ao P2P do que ao ambiente cliente/servidor.

Um computador cliente pode servir a um computador central, trocando completamente sua função na rede. Um computador cliente pode se utilizar outro computador cliente para realizar o processamento, que, por sua vez, pode solicitar dados que estejam no computador central. Este é o aspecto mais importante de colaboração em uma Organização Virtual.

É necessário que haja controle e informações sobre os recursos para que a grade tenha condições de encaminhar serviços para serem executados. Métricas de desempenho, prioridade, expectativas e limitações de cada recurso são extremamente importantes quando se trabalha com Organizações Virtuais.

2.7. Desenvolvimento baseado em serviços

Uma Grade Computacional é uma grande Organização Virtual (OV) porque reúne todos os recursos espalhados pela rede como se fosse um único. Isso faz com que seja possível aumentar o desempenho de aplicações e a capacidade de armazenamento. Sistemas que necessitam de um alto poder de processamento e armazenamento são os que mais se beneficiam do

mitisse realizar os principais serviços dos bancos de dados, tirando o máximo proveito dos serviços da Grade Computacional. A este mecanismo que, conforme analisado anteriormente, deve ser baseado em serviços, dá-se o nome de Banco de Dados Federado ou Banco de Dados Virtual. Tanto no meio acadêmico e científico como no ambiente empresarial, é necessário trabalhar com dados estruturados, semi-estruturados e não-estruturados.

Este Banco de Dados Virtual deve prever a utilização desses tipos de dados. O mecanismo sugerido guarda alguma semelhança com os padrões ODBC e JDBC, mas há interesse em não limitar ou nivelar por baixo os bancos de dados, como acontece com estas APIs (Application Program Interface). Como a arquitetura proposta é baseada em serviços, os produtos deverão indicar o que se pode ou não realizar (WATSON, 2003).

Imagina-se que, com a adoção crescente das Grades Computacionais por parte das empresas, haverá interesse por parte das empresas fornecedoras de SGBDs de portar seus produtos para o Banco de Dados Virtual (WATSON, 2003).

3.1. SGBD comerciais

A plataforma básica para disponibilizar um SGBD em uma Grade Computacional é formada por:

- Cluster: permite que mais de um gerenciador de banco de dados, do mesmo fornecedor, seja instalado em diferentes servidores. Estes bancos de dados devem ser compartilhados e administrados como se fossem um único. Pode ou não haver mecanismos de replicação dos dados para garantir a disponibilidade da informação e a paralelização das transações para agilizar a recuperação de dados.

- Autogerenciamento e auto-ajuste: como não é possível determinar a carga de trabalho de cada computador clusterizado, o SGBD deve ser capaz

de gerenciar e ajustar-se à demanda de dados e processamento. Sabe-se que cada gerenciador possui características próprias, que permitem melhorar o próprio desempenho. Normalmente, este serviço é realizado por um profissional, no caso, o Administrador de Banco de Dados (DBA). Como em uma grade a demanda não é contínua e nem previsível, não se pode depender exclusivamente de um profissional para realizar esta tarefa. Deve-se criar um repositório que armazene os dados históricos da utilização do banco e, com base no volume de dados e utilização, são feitas alterações nos parâmetros do SGBD para otimizar o desempenho.

- Armazenamento autogerenciado: é um mecanismo que permite o acesso a dados que não estejam fisicamente no mesmo servidor onde está instalado o SGBD, ou, caso esteja, não dependa do sistema operacional para gerenciá-lo. Com este mecanismo, é possível distribuir os dados em diferentes servidores, sendo, contudo, gerenciado pelo próprio SGBD e não pelo sistema operacional. Isso faz com que a carga e armazenamento de dados possam estar distribuídos através de diversos computadores.

Este mecanismo permite que o próprio SGBD replique os dados que julgar necessário, para diminuir a interferência de um profissional. Isso faz com que a segurança da informação seja aumentada, mais ou menos como acontece com RAID.

3.2. Requisitos de um banco de dados em grade

Os sistemas gerenciadores de banco de dados (SGBD) alcançaram um nível de complexidade bastante alto. Isso faz com que a Grade Computacional tenha que fornecer mecanismos complexos para efetuar buscas e controlar transações. Cada implementação de banco de dados possui uma estrutura própria de gerenciamento e segurança que deve se integrar às aplicações da grade. Por outro lado, pelo conceito de grade, não é admissível que se utilize uma única implementação de banco de dados, como Oracle, SQL Server ou

recursos do servidor, uma vez que a maior parte dos gerenciadores já tem mecanismos de integração com o equipamento e sistema operacional; cópias de segurança dos dados armazenados, visto que possuem diversos mecanismos para agilizar e controlar este processo; replicação de dados controlada pelo mesmo servidor e, finalmente, o controle de concorrência em processamento de transações dos dados.

3.3. Padronização

Fica claro que há necessidade de estabelecer padrões entre os bancos de dados, para que estes possam ser utilizados em uma Grade Computacional. A padronização passa pelo estabelecimento de protocolos de comunicação que contenham basicamente: acesso aos metadados dos SGBD, para localização de dados e execução de comandos transacionais, controle de acesso aos dados, notificação de onde está e quanto tempo poderá durar uma determinada transação, linguagem de programação, definição de custo e segurança não apenas no acesso aos dados, mas também quanto à política de backup e recuperação de dados.

O escopo de uma proposta de padronização deve seguir alguns parâmetros aceitáveis tanto do ponto de vista da Grade Computacional quanto dos SGBD (PATON, 2002):

- Independência de grade: não deve estar vinculada a uma grade específica. Atualmente, as grades evoluem rapidamente e vincular uma proposta a uma única implementação pode ser caótico, por utilizar um conjunto específico de funcionalidades e ferramentas que não estejam disponíveis para todas as grades.

- Não criar um novo banco de dados: estabelecer mecanismos de controle de transações, buscas e outros serviços que existem nos principais SGBD. Os bancos de dados mais completos, contudo, terão maior aderência à grade e os demais vão precisar se adaptar às necessidades.

- Independência de banco de dados: não se deve

seguir uma implementação única para modelo de dados, comandos transacionais e programação de banco de dados.

Conforme visto, a idéia básica é criar um banco de dados virtual que se integre aos SGBD atuais e que tenha serviços para oferecer à Grade Computacional. Com isso, os SGBD poderiam informar o nível de integração e como resolveriam determinadas questões. Este banco de dados virtual tomaria a decisão de utilizar ou não um ou outro SGBD. Os principais serviços estariam relacionados com a distribuição dos dados, busca distribuída e replicação seletiva de dados (PATON, 2002).

3.4. Desafios

Os grandes desafios de aplicações robustas para atuar em uma Grade Computacional e que podem ser extrapolados para a utilização de SGBD são classificados por Gil (2004):

- Captura do conhecimento: identificação dos recursos disponíveis – SGBD, dados, relacionamentos entre os dados e entre os SGBD e a capacidade de processamento e armazenamento de cada servidor.

- Utilidade: deve haver plano de contenção para falhas de equipamentos, sistemas operacionais e SGBD. Da mesma forma, é necessário estabelecer uma política para utilização dos recursos da grade. Deve-se prever formas de diminuir o trabalho dos usuários das aplicações, facilitando a programação (schedule) de utilização dos recursos.

- Robustez: deve-se prever o maior nível de utilização dos servidores para cada serviço requisitado. Os servidores devem, preferencialmente, estar disponíveis 24 horas por dia e sete dias por semana. Isso envolve sistema operacional, software de rede e SGBD.

- Acesso: a quantidade de usuários em aplicações típicas da Grade Computacional é muito grande. Estes usuários estão dispersos em Organizações Virtuais. O controle de acesso e permissões é fundamental para o

grade e podem ser requisitadas ao mesmo tempo por diversos clientes e de várias formas diferentes. Pode-se realizar o processamento seqüencial de cada solicitação, prevendo mecanismos de controle de concorrência na própria interface.

Um recurso de dados é virtualizado por um serviço de dados que é encapsulado por uma implementação da interface e não é visível nem acessível por outros usuários do serviço. Naturalmente, a fonte de dados pode ser acessada por outros mecanismos externos à OGSA, mas, nesse caso, não fazem parte da grade. O gerenciador do recurso (por exemplo, o banco de dados) realiza a comunicação com os serviços de dados e possui características próprias, que independem do modelo de serviços da grade (FOSTER, 2003), mas deve, para prestar serviços à grade, indicar quais serviços é capaz de implementar.

3.5.2. Serviços e interfaces de dados

Conforme visto, um serviço de dados implementa uma (ou mais) interface associada a um comportamento, para manipulação dos dados virtualizados. As interfaces especificadas até o momento são (FOSTER, 2003):

- **DataDescription**: é utilizada para indicar aos clientes os detalhes dos serviços de virtualização dos dados. Utiliza-se da descrição dos dados para realizar solicitações para as interfaces **DataAccess**, **DataFactory** e/ou **DataManagement**. São os parâmetros básicos para encapsular a virtualização de dados pelo serviço.

- **DataAccess**: permite acesso para conexão, busca e modificações nos conteúdos de dados virtualizados que são encapsulados pelo serviço. Alguns exemplos dessa interface incluem **SQLAccess**, para realizar buscas e atualizações na base de dados; **CursorRowSetAccess**, que retorna uma linha dos dados virtualizados; **XMLCollectionAccess**, que permite o acesso e manipulação de dados armazenados no formato XML; **StreamAccess**, que realiza operações de leitura e escrita nos dados virtualizados; **FileAccess**, que possui um ponteiro para ser utilizado em conjunto

com o **StreamAccess**; **BlockAccess**, que realiza leitura e escrita em blocos de dados virtualizados; **TransferSourceAccess** e **TransferSinkAccess**, que realizam transferência de dados entre dois serviços de dados.

- **DataFactory**: responsável pela criação do serviço que retorna os dados virtualizados da fonte de dados (exemplo, banco de dados). É responsável por criar um novo serviço de dados virtualizados e deve ser utilizado sempre que se quiser criar um nome para a virtualização dos dados, uma sessão para o cliente ou para criar uma virtualização de dados vazia. Deve haver diversas extensões para esta interface no futuro. Para cada exemplo utilizado na interface **DataAccess**, pode haver uma especificação em **DataFactory**. Esta interface é responsável por realizar a federalização de várias fontes de dados, pois permite criar uma virtualização de dados vazia, que será acrescentada com diversas outras fontes virtualizadas. Com esta interface, pode-se imaginar que uma organização virtual (OV) será capaz de criar uma virtualização própria de dados baseada em diversos serviços de dados que disponha dentro da Grade Computacional.

- **DataManagement**: é utilizada para monitorar e gerenciar o serviço de dados e os dados virtualizados. Inclui o monitoramento e gerenciamento da fonte de dados. Com esta interface, é possível controlar como está o desempenho e disponibilidade do recurso. O controle de utilização, programação (schedule) deve ser realizado através desta interface.

4. CONCLUSÃO

A tecnologia de Grade Computacional ainda não está completamente adaptada para receber os SGBD e estes, por sua vez, não estão suficientemente padronizados para trabalhar na grade. A grade foi criada para ser um ambiente distribuído. Os SGBD foram criados para atender aos usuários de uma organização.

Para resolver esse descompasso, é necessário estabelecer (e seguir) padrões que possibilitem a integração entre ambos. As recentes pesquisas apontam

