Treinamento PostgreSQL
Cluster de Banco de Dados - Aula 01

Eduardo Ferreira dos Santos

SparkGroup
Treinamento e Capacitação em Tecnologia
eduardo.edusantos@gmail.com
eduardosan.com

13 de Junho de 2013

Sumário

Ementa - Administração de Dados (Parte 1)

Introdução Conceito de banco de dados, Histórico do PostgreSQL, Comunidade, Desenho Conceitual, Representação de dados, Armazenamento, Indexação;

Teoria Relacional Modelo de dados e definições, Gerencimento de tabelas, Restrições e integridade referencial, Herança entre tabelas;

Conjuntos Álgebra relacional e operações de conjuntos

Linguagem SQL Sintaxe, Definição de dados, Manipulação de dados, Consultas, Tipos de dado, Funções e operadores, Conversão de tipos, Índices, Busca textual (*Full Text Search*), Controle de concorrência, Análise de performance;

Administração do Banco de Dados

PostgreSQL em GNU/Linux Instalação e Configuração;

Manipulação da estrutura do banco de dados Manipulação de *tablespaces* e *schemas*;

Administração do Servidor Segurança lógica e física, Monitoramento, Ferramentas administrativas e Backup;

PTR *PITR - Point-in-Time Recovery*;

Migração Desenhando um projeto de migração de dados para o PostgreSQL.

Alta disponibilidade

Cluster Virtualização, Arquitetura e definições;

Escalabilidade horizontal Soluções de replicação de base de dados *Master-Slave*, *Multimaster*;

Escalabilidade vertical Banco de dados distribuídos, Replicação de discos;

Escalabilidade nativa *Streaming Replication/Hot Standby*;

Administração do cluster Balancamento de carga, Alta disponibilidade com Heartbeat.

Performance Tuning

PostgreSQL por dentro Estrutura do PostgreSQL, Sistema Operacional, Disco e Regras gerais de performance;

Otimização de SQL Análise de consultas e Plano de execução;

Otimizando os discos Configurações de I/O, Tabela de partições e Tipos de disco;

Configuração do PostgreSQL *shared_buffers* e referência do arquivo postgresql.conf;

Otimizando o SO Otimização de Kernel para GNU/Linux, Memória e Disco;

Cronograma

Semana 1: 27 de Maio a 3 de Junho Administração de Dados

Semana 2: 4-11 de Junho Administração de Banco de Dados

Semana 3: 13-18 de Junho Alta disponibilidade

Semana 4: 19-24 de Junho Performance Tuning

Cluster

- Definição:

  Implementação de compartilhamento de recursos computacionais, utilizando dois ou mais dispositivos de computação [MPOG, 2006, p. XXXI]

- Organizados em:
  - Cluster de Processamento de alto Desempenho (HPC)
  - Cluster de Balanceamento de Carga e Alta Disponibilidade
  - Cluster de Banco de Dados
  - Cluster de Armazenamento

Grid

- Definição:

  Rede de execução de aplicações paralelas em recursos geograficamente dispersos e pertencentes a múltiplas organizações [MPOG, 2006, p. XXXII]

- Aplicação: serviços sob demanda
- Prover sob demanda qualquer serviço computacional

O paradigma c10k

A Internet é um lugar grande, e já é hora de tratar 10.000 requisições simultâneas [Kegel, 2011]

- Baixo custo do hardware para plataforma baixa
- Diferentes estratégias para tratamento de requisições
- Foco: mecanismos de I/O (Entrada e saída)
  1. Threads, I/O não bloqueante e notificação através de gatilhos;
  2. Threads, I/O não bloqueante e notificação por detecção de alterações;
  3. Threads, I/O assíncrono e notificação de completude;
  4. Um cliente para cada thread;
  5. Código do servidor dentro do Kernel.

Demandas Computacionais

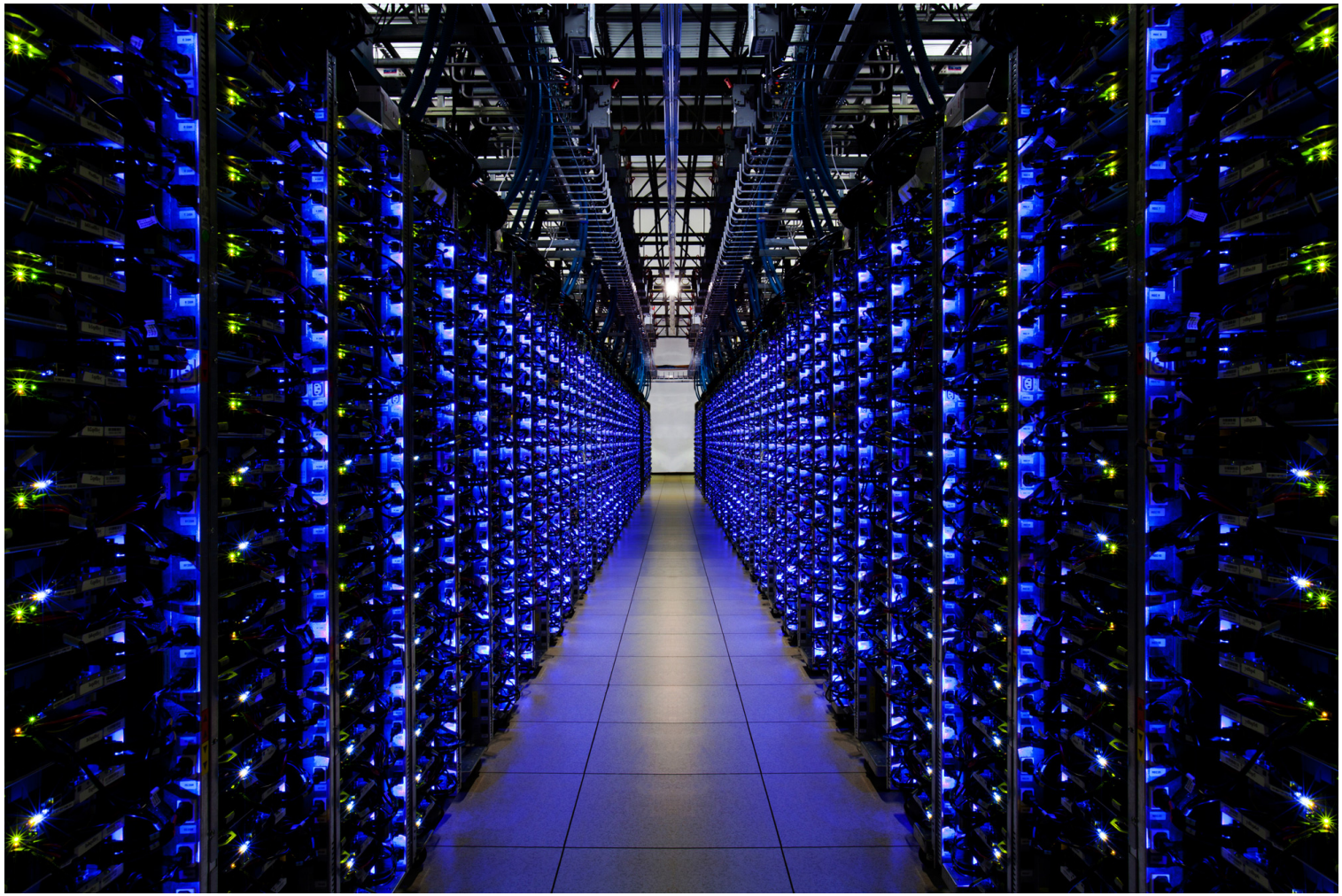
O momento mudou: agora precisamos responder 1.000.000 de requisições por segundo!

- Alta disponibilidade;
- Suporte a milhões de usuários simultâneos;
- Alta capacidade de processamento;
- Capacidade de trabalhar com bancos de dados da ordem de milhões de registros;
- Tolerância a falhas de hardware e software;
- Facilidade de integração e interoperabilidade;
- Armazenamento massivo da ordem de terabytes de dados.

Ontem

Mainframe

Hoje



Google Server Farm

Dois paradigmas computacionais

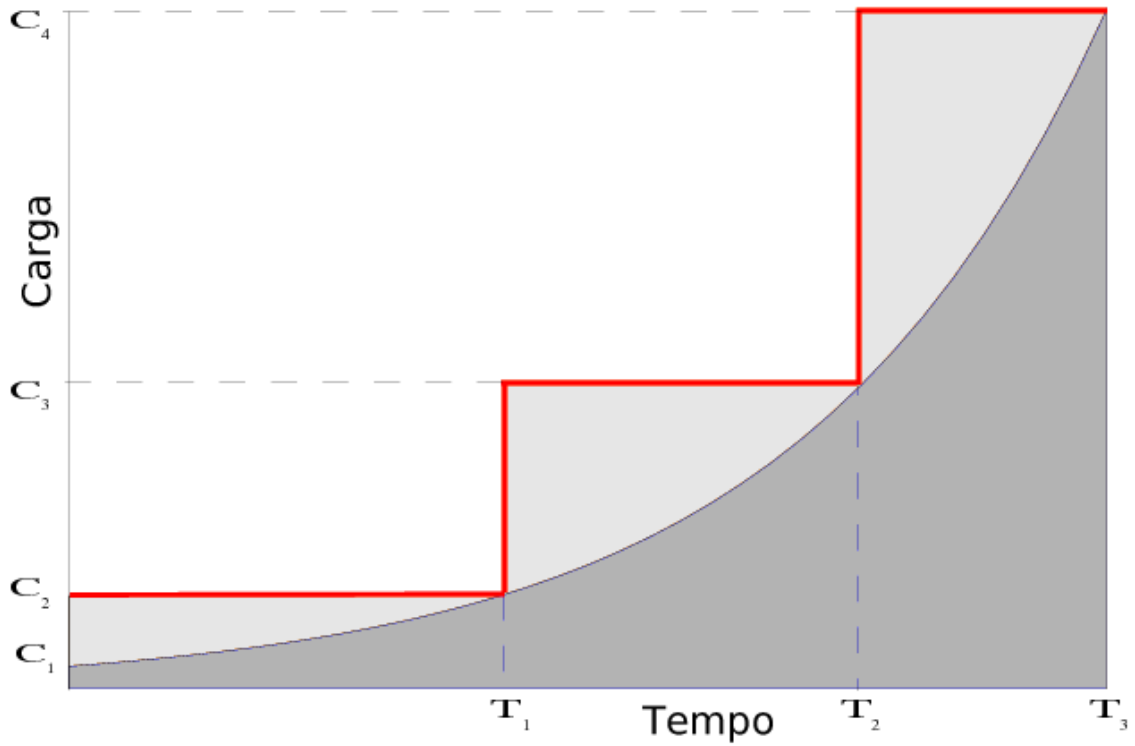
Grande Porte Computadores com alto poder de processamento:
Mainframes

- Processamento de grande volume de informações
- Alto grau de confiabilidade nos dados inseridos
- Mainframes x Supercomputadores
- Difícil expansão da capacidade

Computação distribuída Computadores comuns agrupados em Cluster

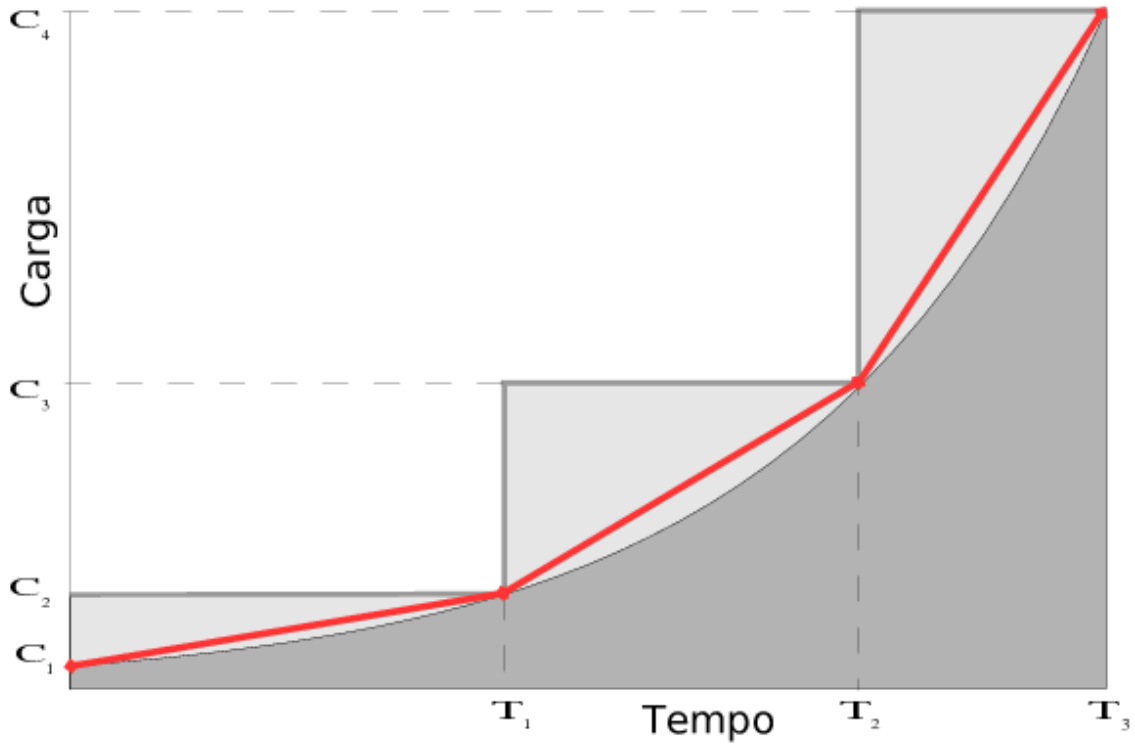
- "Elasticidade" da capacidade de processamento
- Dimensionamento como uma função da necessidade de carga
- Facilidade de expansão

Custo x Capacidade - Grande porte



Evolução da carga de processamento e a utilização da computação de grande porte. [MROG, 2006, p. 29]

Custo x Capacidade - Cluster



Evolução da carga de processamento e a utilização da solução de

Comparativo

Grande Porte	***Cluster* e *Grid***
- Alto custo de implantação; - Dependência de fornecedor único; - Utilização de *hardware* específico; - Alto custo de manutenção;	- Baixo custo de implantação; - Independência de fornecedores – facilidade de negociação; - Utilização de *hardware* comum – padrão PC; - Baixo custo de manutenção;

Comparativo

Grande Porte	***Cluster* e *Grid***
- Utilização parcial da capacidade de processamento; - Grande custo total de propriedade; - Tecnologia estabelecida no mercado.	- Maximização da capacidade de processamento; - Baixo custo total de propriedade; - Tecnologia inovadora.

Cenários

Alta Disponibilidade O banco de dados está sempre no ar, independente das falhas que possam acontecer

Paralelização de consultas Aumentar a velocidade de processamento das consultas SQL. Úteis em aplicações OLAP[1] e *Datawarehouse*[2]

Banco de dados distribuído Distibuição da aplicação de banco de dados em múltiplos servidores, até mesmo geograficamente distribuídos

[1] *On-Line Analytical Processing* é a capacidade de manipular e analisar um grande volume de dados sob múltiplas perspectivas

[2] Aplicação de OLDAP para visualização dos dados gerenciais consolidados

Definições

Escalabilidade Capacidade de aumento de processamento da aplicação

Escalabilidade Horizontal Aumento do número de servidores para processamento das requisições. Cluster

Escalabilidade Vertical Aumento da capacidade de processamento de requisições no mesmo servidor. Tuning ou Ajuste Fino

Master-Slave Arquitetura em cluster onde um servidor se comporta como Mestre possuindo capacidade de escrita apenas recebem os dados

Multi-Master Mais de um nó do Cluster está disponível para escrita dos dados

Alta Disponibilidade

- Objetivo: manter o banco de dados sempre no ar
- Tecnologias:
  - **DRBD** Replicação de disco
  - **Heartbeat** Resposta automática a eventos de indisponibilidade

Restrições

- Cluster é Matemática. Ex.: se eu tenho apenas um servidor, como saber que um deles caiu?
- Se a ferramenta de alta disponibilidade falhar, de quem é a culpa?
- Qual o risco a que está submetida uma ferramenta de resposta automática

Diminuindo os riscos

- Escalabilidade nativa: ferramentas de replicação internas do PostgreSQL

  Warm Stand By Replica o banco de dados em outra máquina, mas não distribui processamento

  Hot Stand By Replica o banco de dados em outra máquina e permite distribuição de processamento

- Ainda assim há muitos riscos. Lembre-se: o PostgreSQL só vai de um estado consistente para outro estado consistente

Multi-master

- Replicação de disco
  - **DRBD** Replicação do disco do postgreSQL em muitas máquinas
- Commit distribuído
  - **Commit síncrono** Ao consolidar a transação o PostgreSQL envia para mais de um servidor
  - **Commit assíncrono** As transações são executadas em cascata em mais de um servidor diferente
    - **Bucardo** Ferramenta livre para replicação
- Pense: você realmente precisa de replicação Multi-master?

Distribuindo a carga

- Se trabalha com Cluster, em algum momento vai precisar de pgPool-II
- Objetivo: distribuir a mesma transação entre vários servidores PostgreSQL
- Necessário para qualquer cenário:

  Master-slave Manda as transações para o master e distribui as consultas entre os nós Slave

  Multi-master Distribui transações entre os diferentes nós master e as consultas entre os slaves
- Enorme gama de configurações e heurísticas de Cluster

Definição

Modo de apresentação ou agrupamento de um subconjunto lógico de recursos computacionais de modo que possam ser alcançados resultados e benefícios como se o sistema estivesse executando sobre a configuração nativa. [MPOG, 2006, p. 341]

Tipos de virtualização

Por software O SO é simulado dentro de um outro ambiente, denominado "hospedeiro"

- **Emulação** A máquina virtual simula todo o hardware, que pode ser diferente do nativo
- **Nativa** Simula parcialmente o hardware, mas a máquina virtual precisa ser projetada para o tipo de processador do hardware hospedeiro
- **Paravirtualização** A máquina virtual não simula o hardware, mas se comunica com o hospedeiro através de uma API que requer compilação especial do sistema hospedeiro
- **No nível do SO** Isola os nós em um mesmo servidor físico, mas todas as máquinas compartilham o mesmo Kernel.

Por hardware Utilização de *microkernel* ou camadas de abstração

Tipos de virtualização

Por software O SO é simulado dentro de um outro ambiente, denominado "hospedeiro"

- **Emulação** Bochs, PearPc, Qemu sem aceleração
- **Nativa** VMware, Parallels Desktop, Adeos, Mac-on-Linux, XEN
- **Paravirtualização** As chamadas de sistema ao hypervisor do XEN
- **No nível do SO** Linux-VServer, Virtuozzo e OpenVZ, Solaris Containers, User Mode Linux e FreeBSD Jails

Por hardware Partições lógicas da IBM

Em banco de dados...

- Em todas as opções acima, o disco é o mesmo
- Em banco de dados, o que mais interfere na performance é o disco

Em banco de dados...

- Em todas as opções acima, o disco é o mesmo
- Em banco de dados, o que mais interfere na performance é o disco
- Resumo: não virtualize o banco de dados

Em banco de dados...

- Em todas as opções acima, o disco é o mesmo
- Em banco de dados, o que mais interfere na performance é o disco
- Resumo: não virtualize o banco de dados
- Se não tiver jeito?

Em banco de dados...

- Em todas as opções acima, o disco é o mesmo
- Em banco de dados, o que mais interfere na performance é o disco
- Resumo: não virtualize o banco de dados
- Se não tiver jeito? Aí utilize um storage no disco

Em banco de dados...

- Em todas as opções acima, o disco é o mesmo
- Em banco de dados, o que mais interfere na performance é o disco
- Resumo: não virtualize o banco de dados
- Se não tiver jeito? Aí utilize um storage no disco
- Se não tiver jeito?

Em banco de dados...

- Em todas as opções acima, o disco é o mesmo
- Em banco de dados, o que mais interfere na performance é o disco
- Resumo: não virtualize o banco de dados
- Se não tiver jeito? Aí utilize um storage no disco
- Se não tiver jeito? Vai ficar lento. Ponto.


Kegel, D. (2011).
O paradigma c10k.
http://www.kegel.com/c10k.html Acessado em 13/06/2013.


MPOG (2006).
Guia de Estruturação e Administração do Ambiente de Cluster e Grid.
SLTI.
http://www.governoeletronico.gov.br/anexos/guia-de-cluster Acessado em 13/06/2013.

Contato

Eduardo Ferreira dos Santos
Sparkgroup
Lightbase Consultoria em Software Público

eduardo.santos@lightbase.com.br
eduardo.edusantos@gmail.com

www.postgresql.org.br
www.eduardosan.com

+55 61 3347-1949