



INSTITUTO DE COMPUTAÇÃO

UNIVERSIDADE ESTADUAL DE CAMPINAS

MC437 - Grupo06 - Relatório 2

Gabriel Oliveira João Fidélis Lucas Moraes
Matheus Figueiredo Pedro Grijó

Technical Report - IC-16-02 - Relatório Técnico

April - 2016 - Abril

The contents of this report are the sole responsibility of the authors.
O conteúdo do presente relatório é de única responsabilidade dos autores.

MC437 - Grupo06

Gabriel Bueno de Oliveira João Guilherme Daros Fidélis
Lucas Henrique Moraes Matheus Yokoyama Figueiredo
Pedro Rodrigues Grijó

Resumo

Utilizamos o benchmark TPC-W, que modela uma livraria online, através de um ambiente controlado, para simular atividades num servidor WEB. Em conjunto com o simulador RBE, que gera três diferentes perfis de carga (Shopping, Ordering e Browsing), pudemos checar o desempenho do servidor instalado num cluster no IC, ao vermos o número de WIPS (WEB Interactions per Second) gerados por diferentes cargas. O relatório refere-se a segunda parte do projeto da disciplina de MC437 (Projeto de Sistemas de Informação) e tem como objetivo realizar testes com dois bancos de dados diferentes, sendo um primário que é replicado para um secundário em hot-standby, testando quais as melhores cargas que o primário consegue atender e depois medir o tempo de recuperação quando o primário é desligado e o secundário promovido.

1 Introdução

Este trabalho é um relatório da segunda parte do projeto da disciplina, que consistiu em preparar além da máquina remota com o servidor Tomcat, utilizar duas máquinas desse cluster com instâncias do PostgreSQL, uma com um banco de dados atuando como primário e outra atuando como um banco secundário em hot-standby para integrar todas essas funcionalidades e fazer um site de compras com dados gerados aleatoriamente.

Também foi instalado na máquina o aplicativo TPC-W que é um benchmark de transações web, que é uma aplicação Java. Para utilizar o TPC-W, foi utilizado o RBE (Remote Browser Emulator), que emula conjuntos de clientes que acessam o lado servidor do TPC-W, que implementa uma loja de livros.

O RBE é um simulador escrito completamente em Java que simula o tráfego HTTP que seria feito por um usuário que estivesse acessando o site através de um navegador.

Também utilizamos o HAProxy, que atua como um proxy para aplicações baseadas em TCP e HTTP. Ele oferece alta disponibilidade e balanceamento de carga para servidores web. Sua função nesse trabalho é assim que detectar que o banco primário falhou, redirecionar as requisições feitas para o antigo banco secundário que deve ser promovido a banco primário.

O TPC-W gera um número, o WIPS (Web Interactions per Second, número de interações Web por segundo). O fluxo de trabalho é gerado pelo RBE e pode ser de três tipos diferentes de perfis. O perfil de compras (*shopping*), onde 80% das ações são de consulta e 20% de escrita no banco de dados. O perfil de navegação (*browsing*), tem 95% das ações de leitura

e 5% de escrita. Já o perfil de compras (*ordering*) tem metades de suas operações de leitura e a outra metade de escrita.

Rodamos três experimentos diferentes, o primeiro apenas com o banco primário ativo, o segundo com o primário ligado e o secundário ligado em hot-standby e o terceiro onde na metade do experimento, o primário era desativado para que o secundário pudesse ser promovido.

2 Condições Experimentais

Nesta seção serão descritas as configurações de hardware e software utilizadas nos experimentos.

Nesta fase, contamos com quatro máquinas, todas iguais e fornecidas pelo Instituto de computação.

A primeira, chamada CBN6, é onde está instalado o HAProxy X.XX, Tomcat versão 7 e TPC-W. Ela que atua como servidor web.

Duas máquinas, a dbmaster2 e dbslave2, é onde ficam os banco de dados primário e secundário, respectivamente. Também tem instalados o PostgreSQL versão 9.5.1.

Já o RBE foi rodado a partir de outra máquina, a CBN 7.

Todas as máquinas tem a mesma configuração: sistema operacional Ubuntu 14.04, CPU Intel(R) Core(TM)2 Quad CPU Q8400 2.66GHz e memória RAM de 4GB e 1333 MHz.

3 Metodologia de Pesquisa

Foi necessário executar três diferentes experimentos. Cada experimento consiste em rodar o RBE para simular uma carga no servidor e avaliarmos as medições feitas.

O primeiro experimento, foi utilizar o RBE com apenas o banco primário ligado. No segundo experimento, estavam ligados os bancos primários e o secundário em hot-standby.

Esses dois experimentos, são feitos para caracterizar nosso servidor. O objetivo é medir a maior carga para a qual ele se comporta de forma consistente.

Para isso, variamos os valores de carga entre 1000 e DESCUBRA, variando de 500 em cada iteração, isso para cada perfil de usuário descrito anteriormente. Assim, podemos fazer gráficos que nos mostram até qual carga o servidor manteve um bom nível de WIPS ao longo de todo o teste (que dura 100s no total).

Com esse valor achado, utilizamos esse valor de carga e utilizamos o RBE para simular essa "carga ótima" no servidor, causando um failover manualmente (isto é, matando o banco de dados primário), enquanto promovemos o banco de dados secundário, para ver em quanto tempo nosso servidor consegue se recuperar e voltar a ficar ativo.

É importante ressaltar que sempre rodamos o RBE de outra máquina remota (CBN7), diferente de nosso servidor, e que os parâmetros utilizados foram Ramp-Up Time: 5s, Ramp-Down Time: 5s e Measurement Interval: 90s. O valor de máximo número de erros tolerados foi colocado em 0. Todos os outros valores são os padrões do RBE.

4 Análise e Resultados

Seguem os gráficos gerados pela execução do RBE para cada experimento

Experimento 1: sem replicação. - Cargas: WIPSB (browsing), WIPS (shopping) e WIPSo (ordering). Réplica primária ativa, réplica secundária desligada.

Experimento 2: sobrecarga gerada pela replicação. - Cargas: WIPSB (browsing), WIPS (shopping) e WIPSo (ordering). Réplica primária ativa, réplica secundária ativa como hot standby.

Os experimentos 1 e 2 devem ser utilizados para caracterizar o desempenho (operações/s e tempo de resposta) em função da carga aplicada.

O objetivo é encontrar a carga que produz o melhor resultado de desempenho. Essa carga deve ser empregada no experimento 3.

Experimento 3: injeção de falha no primário. - Cargas: WIPSB (browsing), WIPS (shopping) e WIPSo (ordering). Replicado, com falha do primário e promoção manual do secundário. Com falha injetada aproximadamente na metade da duração do experimento.

O objetivo do experimento 3 é medir o período em que o sistema não está disponível: quanto menor, melhor. Realizem um número suficiente de experimentos que permita que o grupo argumente que o resultado é confiável (reprodutível) e discuta as potenciais causas para a indisponibilidade.

- Think Time (TT) deve ser otimizado para obtenção de melhor desempenho.
- Banco de Dados: utilizar o dump fornecido.

Configuração para execução:

rbes em um ou mais nós do bloco, tpc-w (tomcat) e haproxy no servidor dedicado a cada grupo e bancos de dados em seus respectivos nós (dbmaster, dbslave ou dbmaster2, dbslave2).

5 Conclusão