INE5646 Programação para Web

- Tópico :

Escalabilidade

(estes slides fazem parte do material didático da disciplina INE5646 Programação para Web)

Sumário

- Definição
- Escalabilidade vertical
- Escalabilidade horizontal

Definição

- A escalabilidade de uma aplicação para web significa a sua capacidade, no lado servidor (camadas 2 e 3 ou 2,3 e 4), de manter-se em funcionamento dentro de parâmetros de desempenho previamente estabelecidos considerando-se um número altamente variável de usuários (requisições) simultâneos.
- A escalabilidade ocorre em duas categorias:
  - **Vertical (scale up)**: capacidade de atender mais requisições simultâneas na mesma máquina.
  - **Horizontal (scale out)**: capacidade atender mais requisições simultâneas usando mais de uma máquina.

Escalabilidade Vertical

- O desafio é conseguir explorar o hardware da forma mais eficiente possível.
- Há duas abordagens:
  - Baseada em hardware.
  - Baseada em software.

Escalabilidade Vertical

- **Baseada em hardware**:
  - Solução simples, óbiva e, muitas vezes, cara.
  - Trocar o processador por outro mais rápido. Solução ok se o processador for antigo.
  - Aumentar a quantidade de memória:
    - Maior quantidade de usuários (dados da requisição e dos usuários)
    - Maior eficiência no processamento das requisições. Por exemplo, possibilita a criação de caches (guardar dados em memória).
  - Usar disco rígido mais rápido. Quanto tempo uma aplicação passa lendo/gravando dados?
  - Trocar todo o servidor atual por um novo mega-ultra-super servidor de última geração. Compensa ($)?
  - Observação: algumas vezes, porém, modificar o hardware não resolve o problema. Exemplo: a linguagem usada para desenvolver a aplicação só utiliza 1 das CPUs.

Escalabilidade Vertical

- **Baseada em software**:
  - Aumentar o desempenho dos servidores HTTP tradicionais (ex: APR for Tomcat).
  - Usar servidores HTTP especializados em altíssimas taxas de requisições que exploram melhor os recursos do sistema operacional. Exemplos:
    - nginx - http://nginx.org/
    - HAProxy - http://haproxy.1wt.eu/

Escalabilidade Vertical

- **Baseada em software**:
  - Executar mais de uma tarefa ao mesmo tempo explorando o paralelismo real (hardware) proporcionado pelas múltiplas CPUs (1 thread ativo por CPU).
  - Paralismo combina muito bem com **programação funcional** e seu conceito de imutabilidade.
  - As tecnologias baseadas no modelo “1 thread por requisição (usuário)” não são escaláveis pois threads consomem muita memória. Alternativas:
    - Uma única thread no servidor (como faz o Node.js)
    - Modelo baseado em atores e *futures* (como faz o Play).

Escalabilidade Vertical

- **Baseada em software**:
  - Tecnologias como Node.js e Play (via Akka - http://akka.io/) demonstram, na prática, a questão principal da escalabilidade vertical baseada em software.
  - Em algumas aplicações, a thread que está processando uma requisição fica bloqueada (segurando a CPU) aguardando que uma operação de entrada/saída seja completada.
  - Em outras palavras, durante o ciclo requisição → processamento → resposta, o servidor passa a maior parte do tempo (etapa processamento) não fazendo nada. A ideia é que ele poderia, enquanto aguarda, estar atendendo outras requisições.

Escalabilidade Vertical

- **Baseada em software**:
  - É neste contexto que aparecem as expressões: event-driven, non-blocking I/O, asynchronous.

Node.js:

> Node.js is a platform built on Chrome's JavaScript runtime for easily building fast, scalable network applications. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices.

Play:

Underneath the covers Play uses a fully asynchronous model built on top of Akka. Combined with being stateless, Play scales simply and predictably.

- Stateless Web Tier
- Non-blocking I/O
- Built on Akka
- Real-time enabled

Akka:

Escalabilidade Vertical

- **Baseada em software**:
  - A escalabilidade vertical está relacionada ao conceito web 2.0.
  - Na web 1.0, os dados ligados aos usuários são armazenados no servidor (camada 2) por meio do conceito de sessão (ex: carrinho de compras).
  - Esta abordagem é conhecida como ***statefull***. Não é escalável pois gasta-se memória para cada usuário simultâneo.
  - Na web 2.0 a tendência é usar a abordagem ***stateless***, onde os dados do usuário são mantidos na camada 1 (no computador do usuário).
  - O papel principal da camada 2 passa a ser oferecer serviços, por meio de uma API RESTful (que será apresentada em outro tópico), para a camada 1.

Escalabilidade Horizontal

- Todo computador (servidor), por melhor que esteja configurado, possui um limite na sua capacidade de atender requisições (usuários) simultâneas.
- A escalabilidade vertical, portanto, possui um limite.
- Escalabilidade horizontal: colocar mais de um computador (servidor) para realizar o serviço.

Escalabilidade Horizontal

- Escalabilidade horizontal é um conceito intrinsicamente relacionado a sistemas distribuídos: o sistema passa a ser formado por um conjunto/rede de nodos.
- No contexto das aplicações para web, os nodos envolvem as camadas 2, 3 e 4.
- O objetivo é, **aparentemente**, simples de ser alcançado: N computadores (servidores) conseguirão atender N vezes mais usuários (requisições) simultâneos.

Escalabilidade Horizontal

- A escalabilidade horizontal pode ser um problema bem simples de ser resolvido.
- Exemplo: Em uma aplicação para web organizada em 4 camadas:
  - Situação atual: as camadas 2, 3 e 4 sendo executadas em um único computador.
  - Problema: todas as camadas competindo pelos mesmos recursos (CPU, memória e disco).
  - Solução: cada camada sendo executada em um computador próprio.

Escalabilidade Horizontal

- Em alguns casos, a solução “1 computador por camada” não é suficiente”.
- Ao passar para o patamar “N computadores por camada” entra-se no verdadeiro território da escalabilidade horizontal com seus benefícios e desafios.
- Alguns desafios:
  - Administrar dezenas, centenas ou milhares de máquinas ao mesmo tempo não é fácil.
  - Avaliar se uma aplicação está suficientemente escalável não é uma tarefa trivial. Como avaliar se o custo ($) compensa?
  - Como reorganizar (redistribuir) os dados nos nodos garantindo a consistência das informações?

Escalabilidade Horizontal

- Os conceitos de armazenamento e de processamento nas nuvens (“cloud computing”) tornam, em geral, os desafios menos complicados e mais viáveis ($).
- Os relatos das empresas que lidam na prática com escalabilidade horizontal apontam para uma direção: cada aplicação tem suas particularidades (cargas de trabalho ao longo do tempo e interconexões dos dados).
- A escalabilidade horizontal é, portanto, construída e compreendida à medida que a aplicação precisa ser mais escalável. Não há fórmula pronta.