# Impacto da carga de trabalho com rajadas no desempenho de Serviços Web

**Adriana M. Centurion, Marcos J. Santana, Regina C. Santana, Sarita M. Bruschi**

Instituto de Ciências Matemáticas e de Computação (ICMC)
Universidade de São Paulo (USP)
Caixa Postal 668 – São Carlos – SP– Brasil

{amolina, mjs, rcs, sarita}@icmc.usp.br

***Resumo.*** *Este artigo apresenta uma avaliação de desempenho de Serviços Web considerando diferentes cargas de trabalho. Dentre as características avaliadas, destaca-se a existência ou não dos fenômenos de rajadas. O comportamento da demanda na forma de rajadas é uma característica comum de cargas de trabalho encontradas em Serviços Web e aplicações Web, que chegam ao sistema com alta variabilidade e podem causar gargalos de desempenho do serviço em ocasiões de difícil previsibilidade. Este estudo mostra o impacto das rajadas no desempenho do serviço e como elas podem afetar negativamente os tempos de resposta do sistema.*

***Abstract.*** *This paper presents a performance evaluation of Web services considering different workloads. Among the characteristics evaluated, it is considered the existence or not of the phenomena of bursts. The demand behavior in bursts is a common feature of workloads found in Web services and Web applications, it arrives in the system with high variability and can cause performance bottlenecks in service, making it difficult to predict the behavior. This study shows the impact of bursts in service performance and how they can negatively affect response times of the system.*

## 1. Introdução

Os Serviços Web estão se tornando uma importante solução para prover a comunicação entre aplicações heterogêneas. Diferente das aplicações Web tradicionais, compostas de páginas Web com as quais os usuários interagem através de *browsers*, os Serviços Web são aplicações com baixo grau de acoplamento que são disponibilizadas como serviços e que podem ser consumidas por outras aplicações por meio da internet. Os Serviços Web podem ser considerados como uma importante plataforma tecnológica que promove a instanciação de Arquiteturas Orientadas a Serviço. Essa tecnologia tem sido amplamente pesquisada pela comunidade acadêmica e por empresas como IBM, HP e Microsoft [Al-Moayed and Hollunder, 2010], a qual tem investido esforços na sua evolução apoiando as padronizações junto ao W3C (*World Wide Web Consortium*). No entanto, esta solução ainda continua a ser um grande desafio para a comunidade de tecnologia da Informação, principalmente devido ao fato de serem aplicações distribuídas e heterogêneas, exigindo, portanto, que requisitos de qualidade, confiabilidade e desempenho sejam verificados e validados com maior grau de rigidez [Dilucca, 2005].

O desempenho de um sistema computacional é determinado por suas características, bem como pela composição da carga que estiver sendo imposta a ele [Menascé and Almeida, 2002]. No contexto de Serviços Web, dentre o conjunto de características comuns de carga de trabalho, destaca-se a ocorrência de rajadas. O fenômeno de rajadas em cargas de trabalho se caracteriza pelo aumento inesperado de solicitações de serviços submetidas ao sistema, com consequentes picos temporais irregulares na intensidade de chegadas das requisições dos clientes ou na intensidade de demandas de serviços impostas ao sistema. O desempenho do Serviço Web pode ser afetado significativamente por essa carga de trabalho que apresenta alta variabilidade em um determinado espaço de tempo [Mi et al., 2010].

Diversos trabalhos sobre caracterização e análise de diferentes tipos de cargas de trabalho podem ser encontrados na literatura. Alguns trabalhos baseiam-se em *traces* para realizar o estudo de caracterização de cargas visando à análise de desempenho em servidores Web [Bertolino et al., 2008] [Li et al., 2008]. Outros trabalhos tentam modelar rajadas no processo de chegada das requisições ou na demanda de serviço impostas aos recursos do sistema [Mi et al., 2010] [Lu, et al., 2010] [Casale et al., 2011]. No entanto, a maioria desses trabalhos é voltada para avaliar o desempenho de aplicações Web baseadas no modelo arquitetural multicamada. Em se tratando de Serviços Web, alguns *benchmarks*, com o intuito de avaliar o desempenho desses sistemas, são propostos na literatura [TPC, 2004] [Head et al., 2005] [Oh et al., 2009]. Entretanto, a maioria desses *benchmarks* tem como objetivo sobrecarregar os provedores de serviço ou avaliar fatores específicos. Além disso, não fornecem mecanismos para modelar rajadas em cargas de trabalho. Existem, desta forma, vários trabalhos abordando estudos relacionados à caracterização de cargas de trabalho voltadas para aplicações e servidores Web, bem como *benchmarks* disponíveis para Serviços Web. No entanto, uma lacuna que é observada refere-se ao estudo e caracterização de rajadas nas cargas de trabalho impostas aos Serviços Web. O estudo apresentado neste artigo contribui, portanto, para o preenchimento dessa lacuna encontrada.

Este trabalho apresenta uma avaliação de desempenho voltada para Serviços Web, considerando diferentes características das cargas de trabalho. Dentre as características a serem avaliadas destaca-se a existência ou não dos fenômenos de rajadas. As rajadas nas cargas de trabalho são modeladas com base na metodologia proposta por [Mi et al., 2010] a qual faz uso de duas técnicas: MAP (*Markovian Arrival Process*), para regular a taxa de chegada das requisições no sistema; e índice de dispersão, para regular a intensidade de rajadas no processo de chegada das requisições. Os resultados obtidos mostram a importância de se considerar rajadas no modelo de cargas de trabalho, visto que elas podem degradar o desempenho e tempos de resposta do sistema se não levadas em consideração.

O restante deste artigo está organizado da seguinte forma. A Seção 2 aborda os trabalhos relacionados a respeito de modelos de rajadas em cargas de trabalho. A Seção 3 discorre sobre aspectos relacionados à caracterização de cargas de trabalho de Serviços Web e conceitos referentes ao fenômeno de rajadas no processo de chegada das requisições. As Seções 4 e 5 detalham as aplicações desenvolvidas, o ambiente de experimentação e o planejamento de experimentos utilizados para realização da avaliação de desempenho. Finalmente, nas Seções 6 e 7 são apresentados os resultados obtidos, as principais conclusões deste trabalho e sugestões de trabalhos futuros.

## 2. Trabalhos Relacionados

Vários trabalhos encontrados na literatura têm tratado da modelagem de rajadas em cargas de trabalho. De modo geral estes trabalhos são voltados para avaliar o desempenho de servidores e aplicações Web tradicionais.

Em [Krishnamurthy et al., 2009] é proposto um método de média ponderada (WAM -*Weighted Average Method*) para melhorar a precisão dos modelos analíticos de previsão de desempenho para sistemas com rajadas de clientes simultâneos. Esse método é baseado em *traces* de sessões juntamente com modelos de desempenho, como modelos de redes de filas, levando em consideração o impacto causado pelas rajadas no processo de chegada, para estimar o tempo de resposta médio das requisições. Bodik, et al. (2010) apresentam uma metodologia para modelar picos no volume da carga de trabalho, para refletir uma carga de trabalho mais intensa e picos nos dados, para representar mudanças na distribuição da popularidade de objetos individuais. Esse modelo é baseado na análise de quatro *traces* de servidores Web e de registros parciais de uma rede social popular, conhecida como *Twitter*.

Casale et al. (2011) propõem uma metodologia para construção de *benchmarks* com níveis de rajadas customizadas na demanda de serviço, denominada de BURN (*BURstiness eNabling method*) para avaliação de desempenho de sistemas baseados em sessões. Nessa metodologia são implementadas duas políticas de submissão de sessões: $\mathbf{P}^{trad}$ para gerar a carga de trabalho tradicional de um *benchmark* e $\mathbf{P}^{burst}$ para introduzir rajadas na carga de trabalho. A partir de um conjunto de cargas de trabalho pré-existentes, BURN encontra uma política que intercala a sua execução para estressar a aplicação multicamada e gerar rajadas controladas no consumo de recursos do sistema. Outro trabalho baseado em *benchmarks* é apresentado em [Rolia et al., 2009]. Nesse trabalho é proposta uma metodologia voltada para modelagem de desempenho de instâncias de serviços de *softwares* customizados, chamado de BAP – *Benchmark-driven Algebraic Method*. Essa abordagem permite um conjunto integrado de métodos, entre eles WAM e BURN, que suportam modelagem e testes de desempenho de aplicações Web baseadas em sessão e que levam em conta características importantes como rajadas no processo de chegada e na demanda de serviços.

Em [Lu, et al., 2010] é apresentada uma política de controle de admissão, baseada em sessão, visando fornecer um suporte adicional para tratar chegadas de sessões em forma de rajadas. Essa política ajusta a capacidade da fila de sessões bloqueadas em resposta às cargas de trabalho com rajadas submetidas ao sistema, a fim de manter o nível de desempenho esperado pelo cliente e minimizar a quantidade de sessões aceitas e abortadas. Outros trabalhos importantes relacionados ao fenômeno de rajadas em cargas de trabalho são apresentados em [Mi et al., 2009] [Mi et al., 2010]. No trabalho apresentado em [Mi et al., 2010] é proposta uma metodologia para modelar cargas de trabalhos com rajadas no processo de chegada das requisições. O modelo proposto nesse trabalho foi adotado na avaliação de desempenho voltada para Serviços Web, apresentada neste artigo. Por esta razão, a metodologia proposta nesse trabalho é apresentada com maiores detalhes na seção 3.2.

## 3. Cargas de Trabalho em Serviços Web

O desempenho de um sistema distribuído, com muitos clientes, servidores e redes, como é caso de Serviços Web, depende fortemente das características da carga de trabalho que

são submetidas a ele. Existe um conjunto considerável de trabalhos sobre caracterização de cargas de trabalho de sistemas Web [Barford and Crovella, 1998] [Wang et al., 2003] [Williams et al., 2005] [Bertolino et al., 2008] [Li et al., 2008]. Algumas das características consideradas nesses trabalhos tratam das distribuições do tamanho médio de transferência, tipos de documentos, tempo entre referências para um mesmo documento, distribuição dos tamanhos dos arquivos e auto-similaridade no tráfego da Web.

É importante destacar que outras propriedades foram encontradas nas cargas de trabalho desses sistemas, uma delas é a tendência de se observar rajadas no processo de chegada das requisições ou mesmo na demanda de serviço dos recursos [Menascé et al., 2000] [Wang et al., 2003] [Mi et al., 2010]. As rajadas no processo de chegada das requisições estão relacionadas aos intervalos entre chegadas das requisições, enquanto que rajadas na demanda de serviço estão associadas às demandas impostas aos recursos do sistema, como memória, CPU e servidores, resultantes das solicitações de serviços. Neste trabalho, o foco está no primeiro caso, ou seja, a carga de trabalho é modelada para representar rajadas no processo de chegada das requisições.

### 3.1. Fenômeno de Rajadas em Cargas de Trabalho

Vários estudos mostram que as requisições HTTP submetidas pelos clientes chegam em forma de rajadas. Um estudo de Wang et al. (2003) mostra como a capacidade do serviço pode ser afetada pela ocorrência do fenômeno de rajadas através da analise de registros de *logs* de um *site e-commerce*. Nesse trabalho, observou-se que as rajadas de requisições submetidas ao sistema representava um dos principais motivos para os gargalos encontrados nos tempos de resposta. Em [Menascé, et al., 2000] também é apresentado um estudo do processo de chegada das requisições em uma aplicação Web *e-commerce*. Os resultados obtidos nesse trabalho revelam rajadas e alta variabilidade no processo de chegada das requisições e a importância de se considerar rajadas no modelo de cargas de trabalho, visto que elas podem degradar o desempenho e *throughput* do sistema se não levados em consideração.

As rajadas no processo de chegada das requisições apresentam algumas propriedades específicas desse tipo de carga de trabalho. Dentre elas destacam-se [Mi et al., 2010]:

- **Localidade temporal**: em situações de rajadas, o processo de chegada das requisições provenientes de diferentes clientes não acontece em instantes de tempos aleatórios, mas sim de forma condensada em curtos períodos ao longo do tempo.
- **Variabilidade em diferentes escalas**: O modelo da carga de trabalho precisa não apenas criar picos de intensidade variável no processo de chegada das requisições, mas também criar flutuações dentro dele.
- **Agregação:** Os clientes agem de modo agregado, ou seja, o processo de chegada das requisições e o intervalo entre duas requisições sucessivas (*think times)* não podem ser gerados de forma aleatória e independentes uns dos outros.

### 3.2. Rajadas no Processo de Chegada das Requisições

No estudo apresentado neste trabalho, a carga de trabalho é modelada para representar situações sem e com rajadas com base na metodologia proposta por Mi, et al. (2010).

Nesse modelo é implementado um processo conhecido como MAP (*Markovian*

*Arrival Process*), o qual tem a habilidade de fornecer variabilidade em diferentes níveis como também efeitos de localidade temporal. O MAP pode ser visto como um modelo matemático de séries de tempo e nesse trabalho é adotado para gerar uma sequência de *think times*, ou seja, intervalo de tempo entre a submissão de duas requisições sucessivas provenientes de um mesmo cliente. A Figura 1 ilustra o modelo MAP com dois estados, um representando situações de tráfego normal e outro representando situações de pico de tráfego, causadas pelas rajadas. O primeiro estado é responsável em gerar uma sequência de *think times* longos, associados aos períodos de tráfego normal. O outro estado é responsável em gerar uma sequência de *think times* curtos, implicando que os intervalos de chegada das requisições sucessivas serão menores e, portanto, podendo resultar em picos de tráfego. Os *think times* curtos e longos são gerados com a taxa média de $\lambda_{short}$ e $\lambda_{long}$ respectivamente. Com o intuito de garantir uma correlação entre diferentes eventos, após a geração de uma nova amostra de *think times*, o modelo tem uma probabilidade $p_{s,s}$ de dois *think times* consecutivos serem curtos e $p_{l,l}$ de serem longos. A probabilidade $p_{s,l}$ e $p_{l,s}$ determina a mudança de estado curto para longo e vice-versa.

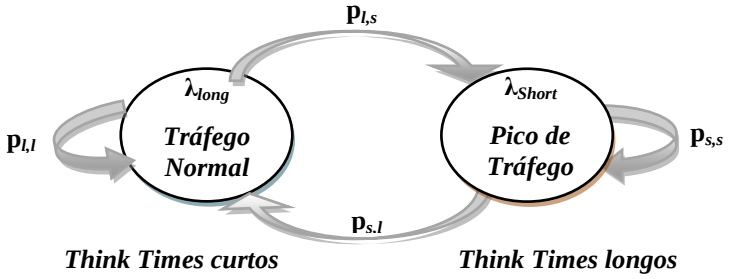


**Figura 1. Modelo de tráfego para regular Think Times [Mi et al., 2010].**

Além da classe MAP de dois estados, o modelo faz uso da medida de índice de dispersão (***I***), a qual é um indicador clássico para representar rajadas em séries de tempo, utilizadas em engenharia e avaliação de desempenho voltada para redes [Gusella, 1991]. O Índice de dispersão tem uma propriedade importante, que garante que o valor de ***I*** cresça proporcionalmente à variabilidade e correlações. Quando não há rajadas, o valor de ***I*** é igual ao coeficiente de variação ao quadrado (*SCV – Squared Coefficient of Variation*) da distribuição, por exemplo, $I = SCV = 1$, para uma distribuição exponencial e à medida que o valor de ***I*** cresce, rajadas de diferentes intensidades surgem no processo de chegada das requisições. Nesse trabalho, o índice de dispersão é utilizado como um regulador da intensidade de rajadas no processo de chegada das requisições.

## 4. Avaliação da Influência das Rajadas no Desempenho de Serviços Web

Para a realização dos experimentos descritos neste trabalho, foi necessário o desenvolvimento de algumas aplicações (Módulo Cliente e Serviço Web), apresentadas na Figura 2. As aplicações foram implementadas em linguagem Java e foi utilizado o *software* Apache Axis2 v1.5.3 para instalação e configuração de um motor de processamento de mensagens SOAP (trocadas entre clientes e o Serviço Web). Além disso, foi utilizado o Apache Tomcat v6.0.29 como servidor de aplicação para publicar os serviços.

A aplicação cliente é responsável por instanciar vários processos concorrentes, onde cada um deles emula um cliente fazendo várias requisições ao Serviço Web. Essa aplicação está integrada ao Módulo Gerador de *Think Times* que é responsável por gerar uma sequência de *think times* de acordo com a política de cargas de trabalho parametrizada pelo usuário: cargas com rajadas ou sem rajadas. Para a política de cargas de trabalho sem rajadas, a amostra de *think times* é gerada utilizando uma distribuição exponencial com *think time* médio E[Z] = 7 segundos e para a política de cargas de trabalho com rajadas, as sequências de *think times* são geradas a partir do índice de dispersão e da classe MAP de dois estados. Desta forma, de acordo com o modelo parametrizado (com rajadas ou sem rajadas), os clientes adquirem *think times* gerados a partir do modelo escolhido.

O Serviço Web implementado fornece um serviço de consulta de endereços nacionais fictícios, via acesso a um banco de dados PostgreSQL. No modelo implementado foram criadas e configuradas duas bases de dados, uma base de dados a qual denominamos de *BDLeve* , com aproximadamente 419 mil registros e uma base de dados *BDPesada* com aproximadamente 700 mil registros.

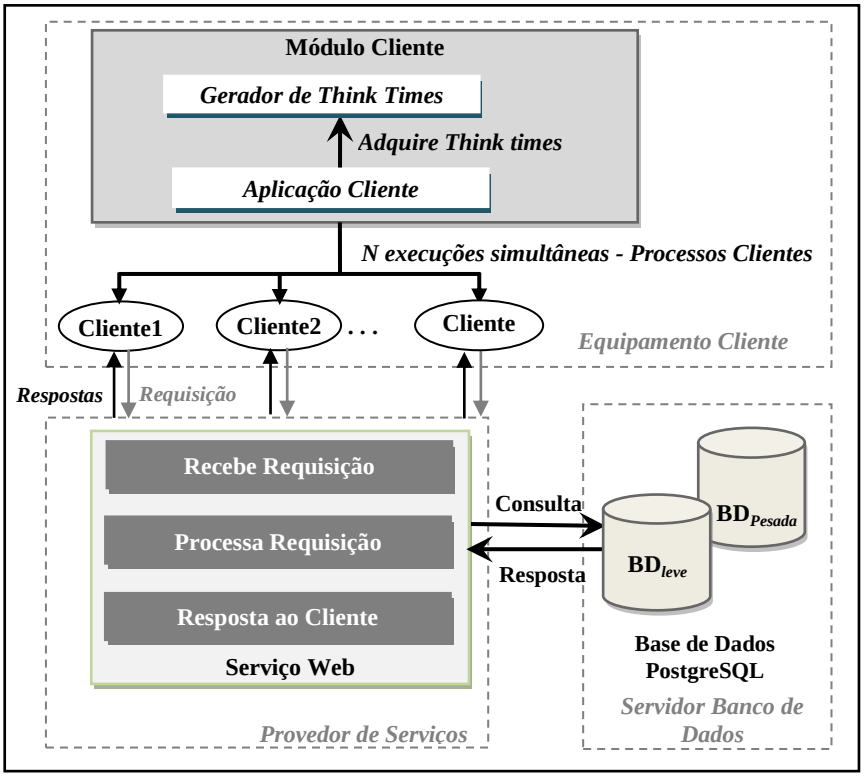


**Figura 2. Aplicações – Clientes e Serviço Web.**

## 5.1. Ambiente de Experimentação

O ambiente físico utilizado para execução dos experimentos é apresentado na Tabela 1. Foram utilizadas quatro unidades de processamento que funcionam como clientes dos serviços, uma unidade de processamento atuando como provedor de serviços e uma unidade executando o banco de dados a ser consultado pelo provedor de serviços. Considerando que há uma quantidade máxima de N clientes no sistema, cada equipamento cliente executa simultaneamente **N/4** aplicações clientes ou *processos*

*clientes* concorrentes, onde cada um deles atua como um cliente enviando várias requisições ao Serviço Web.

**Tabela 1. Ambiente físico para execução dos experimentos.**

| Componente | Quantidade | Tipo | Descrição |
|---|---|---|---|
| Clientes | 2 unidades | Intel Core 2 Quad Q6600 2.4GHz, 4GB de RAM | Clientes que acessam o serviço. |
| Clientes | 2 unidades | Intel Core 2 Quad Q9400 2.66GHz, 4GB de RAM | Clientes que acessam o serviço. |
| Provedor de Serviços | 1 unidade | Intel Core 2 Quad Q8200 2.33GHz, 3GB de RAM | Hospedar e fornecer o serviço implementado aos clientes. |
| Servidor de Banco de Dados | 1 unidade | Intel Core 2 Quad Q9400 2.66GHz, 6GB de RAM | Hospedar a base de dados a ser consultada pelo *Serviço Web*. |

## 5.2. Planejamento dos Experimentos

Os experimentos conduzidos neste estudo consideraram quatro diferentes fatores. O primeiro fator refere-se à quantidade de clientes simultâneos requisitando serviços ao Serviço Web, considerando-se duas quantidades: 100 e 170 clientes. O segundo fator está associado ao processo de chegada das requisições, ou seja, a política de geração de *think times* a ser adotada, para situações sem ou com rajadas. O terceiro fator considera a demanda de serviço imposta ao Serviço Web, desconsiderando o tempo de consulta ao banco de dados. Esse fator representa a possibilidade de serem considerados diferentes tipos de serviço, com demanda computacional distintas. Para esse fator são atribuídos dois níveis: uma demanda de serviço básica, associada ao tempo médio de serviço para que as requisições sejam concluídas pelo Serviço Web e uma demanda de serviço até 67% mais alta do que o tempo médio de serviço básico. Por último, o quarto fator refere-se ao tamanho da base de dados a ser consultada pelo Serviço Web, base $BD_{Leve}$ ou $BD_{Pesada}$. Os níveis dos fatores considerados apresentam um aumento equivalente em torno de 70% de um nível para o outro. Esse aumento foi escolhido com o objetivo de manter uma diferença equilibrada entre todos os valores dos níveis de cada fator.

O planejamento de experimentos utilizado na avaliação de desempenho apresentada neste artigo segue a abordagem do planejamento fatorial completo. Esta abordagem foi escolhida por possibilitar que todas as combinações possíveis de configurações e cargas de trabalho sejam examinadas. Para todos os experimentos foram realizadas 5 execuções, cada uma com duração aproximada de 40 minutos, utilizadas para determinar a média, o desvio padrão e o intervalo de confiança de 95% de acordo com a tabela *T-student*. O número de execuções e duração foi definido considerando-se a necessidade de obter diferença significativa dentre os intervalos de confiança dos experimentos que implica em comparação dos resultados. Os experimentos foram conduzidos com o objetivo de avaliar aspectos relacionados ao desempenho do serviço, compreendendo as seguintes variáveis de resposta:

- **Tempo médio de resposta*:*** intervalo de tempo entre o envio da requisição pelo cliente e chegada da resposta completa processada pelo Serviço Web;
- **Tempo médio de execução WS**: tempo médio de execução das requisições processadas pelo Serviço Web, incluindo tempo médio consumido para estabelecer as conexões com o banco de dados;

- **Tempo médio de consulta ao banco de dados*:*** tempo médio de consulta à base de dados realizada pelo Serviço Web para processar todas as requisições.

## 6. Análise dos Resultados

Nesta seção são apresentados os resultados obtidos com a execução dos experimentos. A análise dos resultados consistiu na observação das médias referentes ao tempo de resposta, tempo de execução do serviço e tempo de consulta à base de dados e de uma análise estatística com cálculo dos desvios padrão, intervalos de confiança, como também das influências dos fatores nas variáveis de resposta.

### 6.1. Tempo Médio de Resposta

Os resultados, referentes ao tempo médio de resposta, obtidos com a execução dos experimentos propostos neste trabalho são apresentados nos gráficos da Figura 3. Os gráficos mostram uma comparação entre os experimentos com carga de trabalho sem e com rajadas de requisições para as quantidades de 100 e 170 clientes. Como pode ser observado, quando são geradas cargas de trabalho com rajadas, há um aumento médio nos tempos de resposta em torno de 350% quando comparado a situações sem rajadas. Este aumento é ainda mais acentuado quando há um crescimento na quantidade de clientes concorrentes, conforme apresentado na Figura 3(b), que apresentam os tempos médios de resposta referentes à quantidade de 170 clientes. Esses resultados mostram que a presença de rajadas no processo de chegada das requisições ocasiona uma significativa degradação de desempenho percebida pelo usuário.

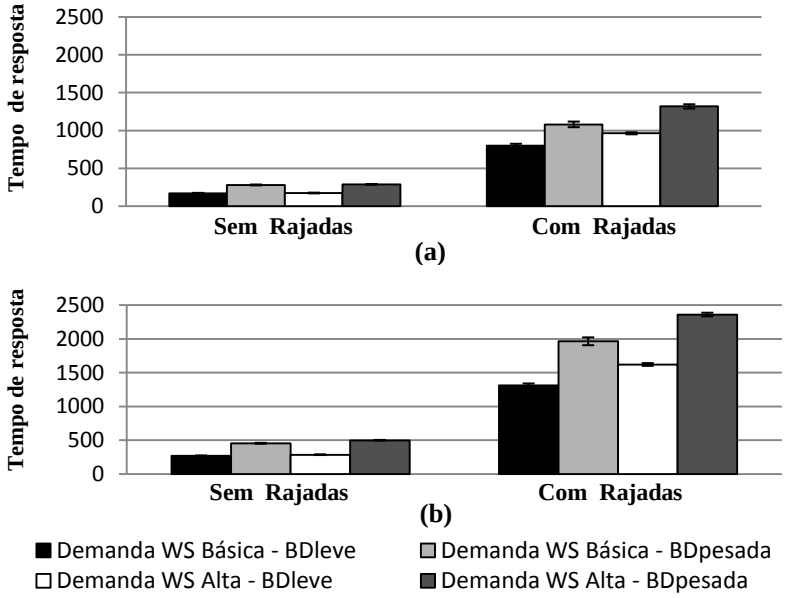


**Figura 3. Tempo médio de resposta em ms: (a) 100 clientes e (b) 170 clientes.**

Quanto ao fator demanda de serviço no Serviço Web, para a situação sem rajadas, o tempo médio de resposta para as duas demandas de serviço consideradas (básica e alta) apresenta valores muito próximos. Entretanto para situações com rajadas os tempos médios de resposta tornam-se mais elevados, em torno de 21%, para uma demanda de serviço mais alta. Isso acontece pois, em condições de rajadas, há um aumento, em alguns períodos ao longo do tempo, no volume de serviço submetido ao Serviço Web. Em relação ao fator relacionado ao tamanho da base de dados ($BD_{Leve}$ e

$BD_{Pesada}$), observa-se que à medida que há um aumento na quantidade de registros da base de dados consultada, o desempenho torna-se pior, principalmente quando são introduzidas rajadas no processo de chegada das requisições. Isso acontece principalmente porque para uma base de dados com tamanho mais elevado, há uma maior sobrecarga imposta tanto no processo de estabelecimento das conexões realizado pelo Serviço Web, como também no tempo médio de consulta à base de dados.

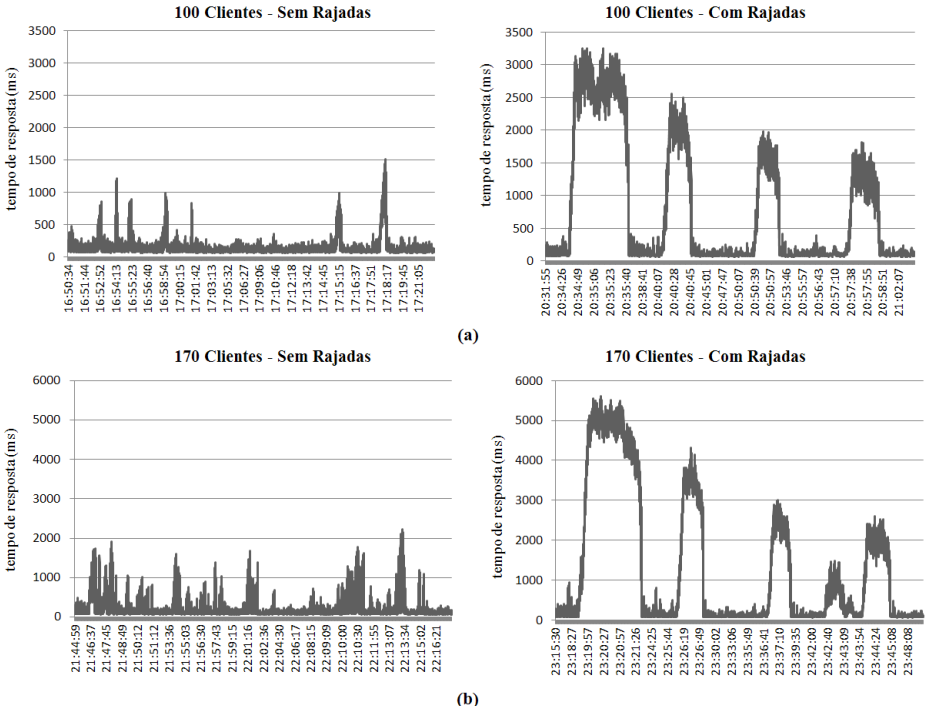


**Figura 4. Amostras dos tempos de resposta: a) 100 clientes e (b) 170 clientes.**

Na Figura 4 é apresentado o tempo de resposta das requisições enviadas ao Serviço Web, em tempo de execução, observados durante uma janela de monitoração de aproximadamente 30 minutos. Os gráficos da Figura 4 mostram os tempos de resposta obtidos para os experimentos sem e com rajadas, para as duas quantidades de clientes analisadas. Nesses experimentos considera-se uma demanda de serviço básica e a base de dados $BD_{Leve}$. Como pode ser observado nos gráficos referentes aos experimentos com rajadas, há momentos em que picos fortes e irregulares em diferentes escalas de tempo aparecem quando são criadas condições de rajadas. Este comportamento não acontece com a mesma intensidade nos casos em que não há rajadas no processo de chegada das requisições.

## 6.2. Tempo de Execução e Consulta

Nesta seção, são apresentados os resultados obtidos nas execuções dos experimentos para as variáveis de resposta: tempo médio de execução das requisições no Serviço Web e tempo médio de consulta ao banco de dados. As Figuras 5(a)(b) mostram os resultados, referentes ao tempo médio de execução. Da mesma forma que foi observado nos resultados obtidos para o tempo médio de resposta, há um aumento considerável no tempo médio de execução das requisições submetidas ao Serviço Web, quando o processo de chegada dessas requisições se apresenta em forma de rajadas. Nota-se ainda que o tempo médio de execução, em condições de rajadas, torna-se ainda mais elevado à medida que a quantidade de clientes passa para 170, conforme é mostrado no gráfico da

Figura 5(b). Outro ponto a ser observado é que, para todos os experimentos executados com 170 clientes, especialmente em condições de rajadas, o fator referente ao tamanho da base de dados apresenta um impacto um pouco maior, associado ao aumento no tempo médio de execução, em relação ao fator demanda de serviço do Serviço Web. Isso ocorre em virtude do tempo médio de execução considerar também o tempo médio de conexão com o banco de dados e para uma base de dados com tamanho mais elevado, há uma maior sobrecarga imposta ao banco de dados para processamento de todas as consultas submetidas a ele e consequentemente, ao processo de estabelecimento de novas conexões concorrentes.

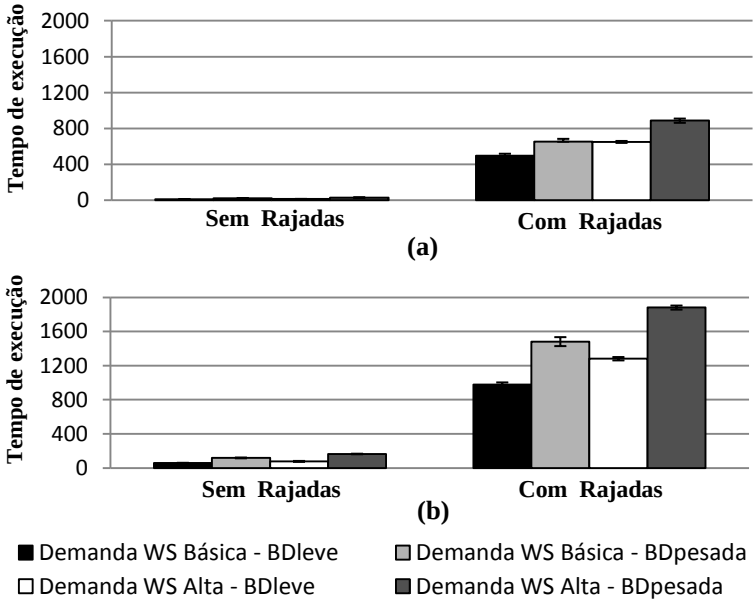


**Figura 5. Tempo médio de execução em ms: (a) 100 clientes e (b) 170 clientes.**

Quanto aos resultados relacionados ao tempo médio de consulta ao banco de dados, na Figura 6 são apresentados os valores obtidos em todos os experimentos. De modo similar aos resultados analisados anteriormente, quando são geradas cargas de trabalho com rajadas, o tempo médio de consulta torna-se pior, porém com um impacto muito menos expressivo do que foi notado para as outras variáveis de resposta analisadas. Nota-se ainda que os valores médios dos tempos de consulta tornam-se maiores para os experimentos que consideram a base de dados $BD_{Pesada}$. Esse resultado, de certa forma já era esperado, visto que o tempo de consulta a uma base de dados com maior número de registros tende a ser mais elevado do que para uma base mais compacta.

A Figura 7 apresenta amostras dos tempos referentes à execução das requisições no Serviço Web e à consulta ao banco de dados, durante uma janela de monitoração de aproximadamente 30 minutos. Observa-se que para os experimentos sem rajadas, para as duas quantidades de clientes, existem algumas variações nos tempos obtidos ao longo do tempo. Para a quantidade de 100 clientes o tempo de consulta se sobressai em relação ao tempo de execução no Serviço Web, enquanto que para uma maior quantidade de clientes, o tempo de execução, em alguns momentos, se equipara ou até mesmo supera o tempo de consulta à base de dados. Entretanto, ao se observar os experimentos com rajadas, ficam evidentes que as variações nos tempos obtidos, no decorrer do período, tornam-se muito mais agressivas, onde picos irregulares e de

diferentes intensidades surgem em ambos os tempos considerados (execução e consulta). No entanto, ao contrário do que foi obtido nas situações sem rajadas, percebe-se que os tempos de execução das requisições ultrapassam os tempos de consulta à base de dados, influenciado principalmente pelo aumento do tempo de conexão com o banco de dados.

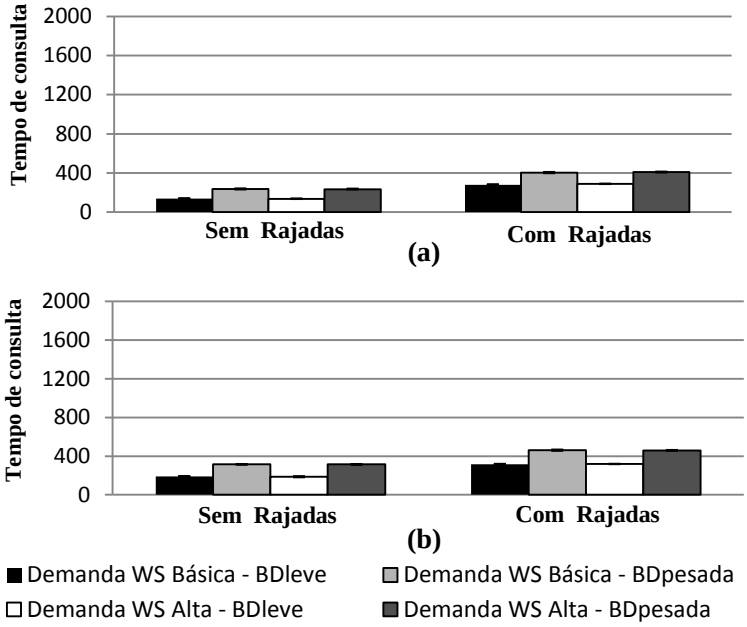


**Figura 6. Tempo médio de consulta em ms: (a) 100 clientes e (b) 170 clientes.**

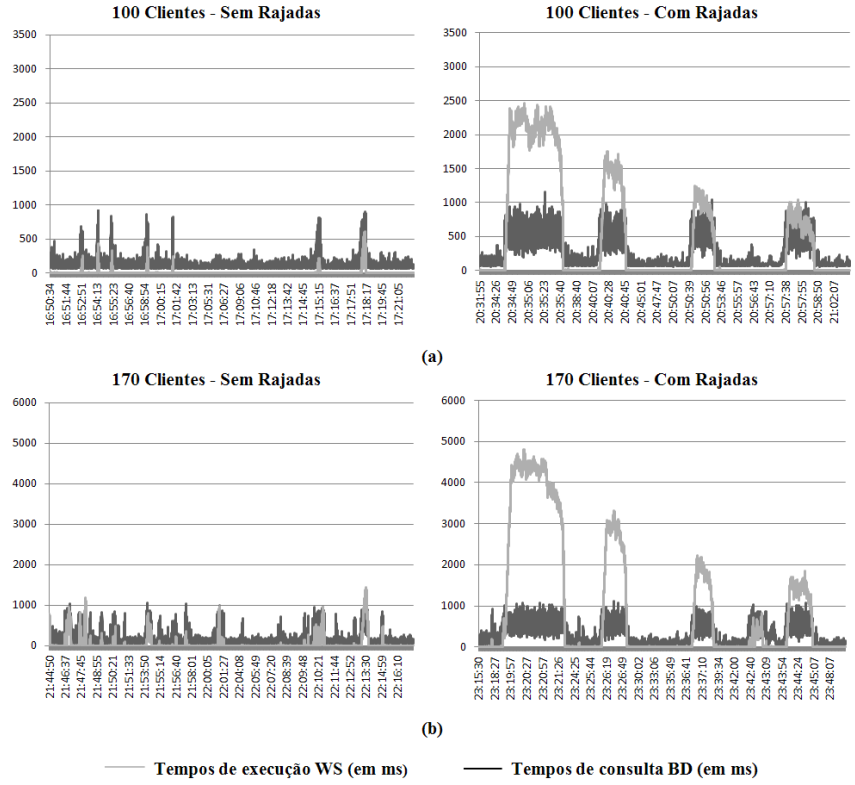


**Figura 7. Amostras - tempos de execução e consulta (a) 100 clientes e (b) 170 clientes.**

### 6.3. Análise das Influências

Nesta seção são apresentadas as influências dos fatores em relação às variáveis de resposta analisadas. Essa análise é feita utilizando modelos de regressão que permitem estimar uma variável aleatória em função de várias outras variáveis (Jain, 1991). Neste caso, a variável estimada corresponde a variável de resposta e as variáveis consideradas para estimá-la representam os fatores. Assim, é possível identificar quais fatores ou quais combinações de fatores que mais influenciam nos resultados. Para tanto, os fatores, quantidade de clientes, processo de chegada, demanda de serviço no Serviço Web e tamanho da base de dados são representados na Tabela 2 pelas letras A, B, C e D, respectivamente. O percentual da interação entre dois fatores é representado por duas ou mais letras. Observa-se na Tabela 2 que para todas as variáveis de resposta, o fator que mais apresenta influência nos resultados corresponde ao processo de chegada das requisições. Esse fator representa uma influência de 70% nos resultados obtidos para o tempo médio de resposta e tempo médio de execução do Serviço Web e 54% para o tempo médio de consulta à base de dados. Essa influência evidencia o impacto negativo no desempenho, provocado pelas rajadas introduzidas no processo de chegada das requisições.

**Tabela 2. Influência dos fatores nas variáveis de resposta.**

| Variáveis de Resposta | A | B | C | D | AB | AC | AD | BC | BD | Outras Interações |
|---|---|---|---|---|---|---|---|---|---|---|
| Tempo médio de resposta | 11,9% | 71,0% | 1,2% | 6,3% | 5,4% | 0,1% | 0,8% | 0,9% | 1,8% | 0,65% |
| Tempo médio de execução WS | 12,3% | 70,2% | 1,6% | 3,2% | 7,7% | 0,1% | 0,8% | 1,2% | 2,0% | 0,81% |
| Tempo médio de consulta BD | 7,5% | 54,5% | 0,0% | 37,1% | 0,3% | 0,1% | 0,4% | 0,0% | 0,3% | 0,02% |

## 7. Conclusões

Conhecer e caracterizar a carga de trabalho é requisito fundamental na avaliação de desempenho e na atividade de planejamento de capacidade de um Serviço Web. O conhecimento adquirido nessa tarefa auxilia a identificação de problemas de desempenho, disponibilidade e confiabilidade do sistema. A caracterização de carga é também essencial para a construção de cargas sintéticas e para a proposição e validação de *benchmarks*.

Este trabalho apresentou uma avaliação de desempenho no contexto de Serviços Web, através do desenvolvimento de diferentes cargas de trabalho e considerando a ocorrência de fenômenos de rajadas. O processo de chegada das requisições em forma de rajadas é um fenômeno específico de carga de trabalho de Serviços Web, que chega com alta variabilidade em um determinado espaço de tempo, podendo afetar significativamente o desempenho do serviço.

O estudo mostrou, considerando os fatores, níveis e experimentos adotados nessa avaliação, que o processo de chegada das requisições em forma de rajadas corresponde ao fator que mais afeta negativamente o desempenho, para todas as variáveis de resposta analisadas. Este comportamento é consistente com as afirmações discutidas por alguns autores [Wang et al., 2003] [Mi et al. 2010] [Lu, et al., 2010] [Rolia et al., 2010] [Casale et al., 2011] que consideram o fenômeno de rajadas uma característica importante de carga de trabalho a ser considerada na avaliação de

desempenho de aplicações Web, visto que essas cargas podem degradar consideravelmente o desempenho de um provedor de serviços, conduzindo para uma significante sobrecarga dos servidores, aumento descontrolado nos tempos de respostas e no pior caso, indisponibilidade do serviço.

Como trabalhos futuros, pretende-se ampliar o estudo de avaliação de desempenho apresentada neste artigo, considerando abordagens diferentes, disponíveis na literatura, para modelar rajadas no processo de chegada das requisições e avaliar como os diferentes modelos impactam o desempenho do serviço oferecido. Objetiva-se também propor um modelo de cargas de trabalho voltado para Arquiteturas Orientadas a Serviços, que considerem rajadas e que possa ser instanciado em *benchmarks* ou em demais técnicas de avaliação de desempenho existentes para avaliar o desempenho não apenas de Serviços Web, mas também de Serviços disponibilizados em Computação em Nuvem.

## Referências

Al-Moayed, A. and Hollunder, B. (2010). "Quality of Service Attributes in Web Services", Fifth International Conference on Software Engineering Advances, Nice, France, p. 367-372.

Barford, P. and Crovella, M. (1998). "Generating Representative Web Workloads for Network and Server Performance Evaluation", In Proceedings of the 1998 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, p. 151-160.

Bertolino, A., De Angelis, G., Frantzen, Al., Polini, A. (2008). "Model-Based Generation of Testbeds for Web Services", In: Testing of Communicating Systems and Formal Approaches to Software Testing – TESTCOM/FATES, Lecture Notes in Computer Science, Springer, p. 266–282.

Bodik, P.; Fox, A.; Franklin, M. J.; Jordan, M. I.; Patterson, D. A. (2010). "Characterizing, modeling, and generating workload spikes for stateful services", In ACM Symposium on Cloud Computing (SOCC), Indianapolis.

Casale, G., Kalbasi, A., Krishnamurthy, D., Rolia, J. (2011). "BURN: Enabling Workload Burstiness in Customized Service Benchmarks", Software Engineering, IEEE Transactions.

Di Lucca, G. (2005). "Testing Web-Based Applications: the State of the Art and Future Trends", In QATWBA'05: Workshop of the International Computer Software and Applications Conference, COMPSAC 2005, p. 65, IEEE Press, Edinburgh-Scotland.

Gusella, R. (1991). "Characterizing the variability of Arrival Process with Indexes of Dispersion", IEEE JSAC, p. 203-211.

Head, M. R., Govindaraju, M., Slominski, A., Liu, P., Abu-Ghazaleh, N., Van Engelen, R., Chiu, K., Lewis, M.J. (2005). "A benchmark suite for soap-based communication in grid services", In SC´05: Proceedings of the 2005 ACM/IEEE Conference on Supercomputing, Whashington, USA, IEEE Computer Society.

Jain, R. (1991). "The art of computer systems performance analysis: Techniques for experimental design, measurement, simulation, and modeling". Wiley.

Krishnamurthy, D., Rolia, J., Xu, M. (2009). "The Weighted Average Method for Predicting the Performance of Systems with Bursts of Customer Sessions", Software Engineering, IEEE Transactions.

Li, H., Lee, W-C., Sivasubramaniam, A., Giles, C. L. (2008). "Workload analysis for scientific literature digital libraries", Journal International on Digital Libraries - Special Issue on Very Large Digital Libraries, p.139–149.

Lu, L., Cherkasova, L., Personè, V. N., Mi, N., Smirni, E. (2010). "AWAIT: Efficient overload management for busy multi-tier web services under bursty workloads", Proceeding ICWE'10 Proceedings of the 10th international conference on Web engineering, Springer-Verlag Berlin, Heidelberg.

Menascé, D., Ribeiro, F., Almeida, V., Fonseca, R., Meira, R.R.W. (2000). "In Search of Invariants for E-Business Workloads", In second ACM conference on Electronic Commerce, Minneapolis.

Menascé, D. A., Almeida, A. F. (2002). "Capacity Planning for Web Services: Metrics, Models, and Methods", Prentice Hall.

Mi, N., Casale, G., Cherkasova, L., Smirni, E. (2009). "Injecting Realistic Burstiness to a Traditional Client-Server Benchmark", Proceedings of the 6th International Conference on Autonomic Computing, p. 149-158, Barcelona, Espanha.

Mi, N., Casale, G., Cherkasova, L., Smirni, E. (2010) "Sizing multi-tier systems with temporal dependence: benchmarks and analytic models", Journal of Internet Services and Application (JISA), p. 117-134.

Oh, S.C, Kil, H., Lee, D. (2009). "WSBen: A Web Services Discovery and Composition Benchmark TOOLKIT", In: International Journal of Web Services Research, p. 1-9.

Rolia, J., Krishnamurthy, D., Casale, G, Dawson, S. (2010). "BAP: a benchmark-driven algebraic method for the performance engineering of customized services". Proceedings of the first joint WOSP/SIPEW international conference on Performance engineering, San Jose, California.

TPC. (2004). "TPC Benchmark App (Application Server) specification, version 1.0", Transaction Processing Performance Council, http://www.tpc.org/tpc_app/.

Wang, Q., Makaroff, D. J., Edwards, H. K., Thompson, R. (2003). "Workload Characterization for an E-commerce Web Site", In: Proceedings of the 2003 Conference of the Centre for Advanced Studies on Collaborative research, p. 313-327, Toronto, Ontario, Canada.

Williams, A., Arlitt, M., Williamson, C., Barker, K. (2005). "Web Workload Characterization: Ten years later", In Web Content Delivery, v. 2, p. 3-21.