Gustavo João Roberto Gorgulho Franco
Orientador: Wagner Meira Jr.

Uma Metodologia de Caracterização de Comportamento de Usuários de Serviços *Internet*

Dissertação apresentada ao Curso de Pós-graduação em Ciência da Computação da Universidade Federal de Minas Gerais, como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

Belo Horizonte
22 de Outubro de 2004

Agradecimentos

Gostaria de agradecer primeiro a Deus, por estar sempre presente em minha vida, e me dar saúde e paz para buscar sempre a felicidade.

Agradeço a meus amados pais, Tetê e José Roberto, pelo infinito amor, apoio e confiança. Vocês são exemplos para mim e a base para a minha vida. A Mia e Titila, por sempre terem me dado tanto carinho e amor. Admiro muito e tenho paixão por vocês. A Lelinha, que nesses últimos 10 anos me trouxe tanta alegria e me fez sentir uma amor diferente, de "paidrinho". Amo muito a todos!

A Poli, minha gatinha, por seu grande amor, respeito, companheirismo, confiança e paciência. Agradeço também por saber entender minhas ausências. Obrigado por estar ao meu lado e fazer minha vida mais feliz a cada dia. Te amo muito!

Sou eternamente grato a minhas queridas avós, Mãe Luíza, por todo amor e cuidado que sempre teve comigo desde bebê, Vovó Janilda, sinônimo de paz e serenidade. A meus tios, Inha, Padrinho, Sirlene, Edivaldo, Clarice e Celso, primos e primas, que completam essa maravilhosa família. Merecem ainda minha recordação Marcelo, Eduardo, Teresa, Octacílio, Dê e Flávio.

Dinda, obrigado pelas "conversas" à noite. Sei que você sempre esteve ao meu lado me iluminando nos momentos mais difíceis.

Agradeço ao professor Wagner Meira Jr., orientador deste trabalho, pelos valiosos ensinamentos, pela confiança, incentivo e amizade. Obrigado por todas as oportunidades, desafios e orientações.

Aos demais membros da banca, professores Virgílio, Lucila e Dorgival, agradeço pelos conhecimentos transmitidos e importantes conselhos.

Agradeço aos amigos e companheiros de pesquisa Adriano Pereira, Leonardo Silva e Walter Santos pelas inúmeras contribuições a este trabalho. Ao Adriano gostaria de fazer um agradecimento especial pelo exemplo de organização, perseverança, trabalho, e por todos os conhecimentos que me passou nesses anos de convivência.

O meu obrigado a todos meus amigos que torceram por mim, me apoiaram e estiveram

do meu lado nesses dois anos. Aos professores e toda a equipe do Synergia, que acreditaram em mim. Léo e Fabi que me proporcionaram um enorme crescimento profissional e pessoal.

A todos do laboratório *e-SPEED*, pelo ambiente harmonioso e pela convivência diária sempre tranquila.

Por fim, agradeço o privilégio de ter estudado no Departamento de Ciência da Computação (DCC) da Universidade Federal de Minas Gerais (UFMG).

Resumo

A constante evolução dos *web sites* tem atraído cada vez mais usuários para a *Internet*. Com isso, o desempenho dos servidores e a qualidade dos serviços oferecidos se tornam fatores cruciais para a manutenção do interesse dos usuários pelo *site*. Esses aspectos causaram a necessidade de se pesquisar formas de melhorar o desempenho dos servidores *Web*. Entender as características das cargas de trabalho permite quantificar os fatores relacionados à interação do cliente com o servidor, que são importantes para projetar sistemas com melhor desempenho e escalabilidade. Com esse conhecimento, torna-se possível analisar as demandas de serviço que um usuário impõe ao sistema, bem como as tendências do comportamento desse usuário durante o tempo em que utiliza o serviço.

Para tratar o problema da caracterização de carga de trabalho de servidores *Web*, essa dissertação propõe uma metodologia hierárquica de caracterização baseada no comportamento do usuário. Considera-se comportamento a maneira como a interação dos usuários com as aplicações *Web* é afetada pela variação de latência (tempo de resposta do servidor). A metodologia sugere a caracterização em quatro níveis de abstração - usuário, sessão, ação e requisição. Para demonstrá-la e validá-la, foram utilizados dados atuais do servidor *proxy-cache Squid* da Universidade Federal de Minas Gerais (UFMG). Ao realizar a caracterização da carga de trabalho foram definidos seis perfis de usuário, identificados durante a análise do fluxo de ações dos clientes e os respectivos tempos de resposta do servidor. Com isso, foi possível capturar e simular as reações dos diferentes tipos de usuário às variações de latência apresentadas.

A caracterização pode ser também o ponto de partida para a construção de modelos analíticos e geradores de cargas de trabalho sintéticas. Esses geradores são usados com eficiência para exercitar a capacidade de servidores e estudar o seu desempenho. A simulação do comportamento dos usuários trará um grande ganho na qualidade da carga de trabalho gerada, reduzindo a distância entre os modelos tradicionais de geração de carga, baseados em distribuições estatísticas típicas, e a carga de trabalho real.

Conteúdo

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

A WWW (*World Wide Web*), ou simplesmente *Web*, foi desenvolvida originalmente para que pesquisadores compartilhassem suas idéias e discutissem seus trabalhos. Atualmente, ela se tornou uma das principais formas de comunicação mundial e permite que uma grande variedade de serviços seja acessada a qualquer momento por todos os tipos de usuário, seja ele humano ou robô - agentes de *software* que navegam pela rede com o objetivo de executar tarefas para os usuários. No início, o que transitava pela rede eram apenas documentos textuais. Nos dias de hoje, são feitas transações bancárias, declaração de imposto de renda de milhões de pessoas, compra e venda de produtos através das mais modernas técnicas de negociação, dentre outros serviços.

Os sítios *web*[1] (*web sites* ou apenas *sites*) evoluíram tanto em questão de conteúdo quanto na parte técnica. Os documentos científicos trocados pelos primeiros usuários (pesquisadores) foram substituídos por qualquer tipo de informação (por exemplo, cotação das ações da bolsa de valores e álbum de figurinhas *on-line*). No aspecto técnico, as páginas que antes eram em HTML passaram a agregar funções e aplicativos. De estáticas, passaram a ser dinâmicas.

Para ilustrar essa variedade de serviços e informações que povoam a *Web* podem ser citados alguns *sites* e serviços que são muito utilizados atualmente, tais como:

[1]Sítios *web*: conjunto de páginas *web*, desenvolvidas por exemplo em HTML

- *sites* institucionais de empresas, universidades e eventos;
- *Internet banking*, que provê várias funcionalidades de um banco tradicional, como pagamento de contas, consulta de saldos e extratos e transferência de dinheiro entre contas, através da rede;
- máquinas de busca, que permitem a busca de informação pela *Web*. Através de uma ou mais palavras informadas pelo usuário é feita uma busca e são retornados links para documentos ou páginas relacionadas a essa(s) palavra(s);
- *sites* de comércio eletrônico, que implementam um negócio *on-line*. Por exemplo, há várias lojas virtuais que comercializam bens e serviços virtuais e reais na *Internet*;
- portais que agregam recursos e serviços variados tais como *e-mail*, fóruns de discussão, páginas pessoais, *shoppings* virtuais, entre outros serviços;
- *sites* de governo eletrônico (por exemplo, utilizado para declaração do imposto de renda de milhões de contribuintes).

É fácil perceber, pela evolução dos aplicativos para a *Web*, que a quantidade de usuários da *Internet* passou de poucos cientistas para o público em geral, ou seja, pessoas comuns passaram a acessar a rede. Hoje, basta ter um computador e uma linha telefônica para ter acesso a rede. Muitas vezes não é preciso nem ter um computador em casa, pois existem bares que permitem que seus usuários naveguem na *Internet*. Além da facilidade de acesso, a explosão de utilização da *Web* se deve ao seu uso para prestação de serviços como, por exemplo, comercialização de mercadorias, *home banking*, além de ser uma forma de comunicação individual - *e-mail*, *chats* e fóruns de discussão - e em massa - jornais e revistas *on-line*.

A grande variedade dos serviços oferecidos na *Internet* e a sua frequente utilização por milhões de pessoas fizeram com que os usuários se tornassem cada vez mais exigentes, requerendo serviços que sejam fáceis de usar e que respondam de forma rápida e precisa às sua requisições. Esses aspectos causaram a necessidade de se pesquisar formas de melhorar

o desempenho de tal acesso. A motivação para essa melhora está em dois objetivos principais: redução no volume do tráfego de rede produzido por clientes e servidores, e maior eficiência no atendimento a requisições, culminando na diminuição do tempo de resposta para os usuários.

A caracterização de cargas de trabalho[2] de servidores *Web* permite quantificar os fatores relacionados à interação do cliente com o servidor, que são relevantes para determinar o desempenho do mesmo. Além disso, através dessa caracterização é possível definir uma representação da carga que permita replicá-la de maneira fiel, para fins de teste e avaliação prévia de implementações de servidores *Web*. Com isso, a caracterização de carga é muito importante para projetar sistemas com melhor desempenho e escalabilidade, aspectos que afetam diretamente a qualidade do serviço experimentado pelos usuários. Com esse conhecimento, torna-se possível analisar as demandas de serviço que um usuário impõe ao sistema, bem como as tendências do comportamento desse usuário durante o tempo em que utiliza o serviço. A melhora da qualidade dos serviços pode ser vista tanto como a diminuição no tempo de resposta a um usuário, quanto como a personalização do *site*. Um dos benefícios da caracterização de carga é que ela permite a construção de modelos analíticos e simuladores que possam replicar o comportamento do usuário de modo que se possa estudar o desempenho de sistemas similares em um ambiente de laboratório. Nesse ambiente fica possível medir com maior precisão os efeitos dos vários tipos de requisições de usuários, e ainda construir modelos mais precisos da utilização dos recursos do sistema.

A variedade e a complexidade das interações entre os usuários e o *site* tornam o processo de caracterização da carga muito interessante. No entanto, pesquisas anteriores sobre caracterização de carga de servidores *Web* tratam esse aspecto de forma superficial, em sua maioria analisando somente o padrão de navegação dos usuários. Poucos estudos foram publicados fazendo uma análise mais detalhada do comportamento dos usuários, tentando modelar a forma como a interação deles com a aplicação *Web* é afetada pelas variações de tempo de resposta do servidor. Isso se deve à dificuldade de se determinar o impacto real

[2]Carga de trabalho: carga imposta a um sistema, no caso a um servidor *Web*, causada pelo acesso dos usuários

que a qualidade do serviço oferecido tem sobre a interação dos usuários com o *site*.

Para realizar essa modelagem é importante considerar aspectos ligados à reação do usuário a um dado tempo de resposta. Analisando-se pela perspectiva de um servidor *Web* ou de um servidor *proxy*, o comportamento do usuário é materializado pela sequência de ações realizadas, onde cada ação pode ser descrita como o clique que resulta em uma ou mais requisições recebidas pelo servidor. Estudando a sequência de ações é possível determinar como o usuário reagiu ao tempo de resposta da ação anterior e, dessa forma, analisar qual o impacto da qualidade do serviço oferecido sobre essa reação.

Caracterizar e replicar o comportamento dos usuários é um desafio discutido e analisado nesse trabalho, tendo como foco principal a relação entre o tempo de chegada entre requisições (*Inter-Arrival Time* ou *IAT*) e a latência (o tempo para processar e responder uma requisição) observada na resposta do servidor a cada ação do usuário.

1.1 Objetivo

O objetivo desse trabalho é propor uma metodologia para caracterizar e simular o comportamento de usuários de serviços *Internet*. A essa metodologia foi dado o nome de *USAR*, uma vez que ela é uma metodologia hierárquica, baseada em quatro níves de caracterização: *Usuário*, *Sessão*, *Ação* e *Requisição*. A *USAR* pode ser dividida em duas partes principais:

- Caracterização de cargas de trabalho de servidores *Web* com foco no comportamento dos usuários. A modelagem do comportamento dos usuários estuda aspectos reativos da interação dos usuários com o *site*, analisando como a qualidade do serviço oferecido afeta as ações dos usuários sobre o sistema.

- Simulação do comportamento dos usuários, validando a metodologia de caracterização proposta. A partir da caracterização realizada, é possível simular o comportamento modelado através da geração de ações de usuário que imitam a interação real do usuário com a aplicação.

1.2 Contribuições

Durante este trabalho ficou claro que diversas pesquisas na área tratam o problema de caracterização de cargas de trabalho *Web*, mas poucas fazem uma análise mais detalhada do comportamento dos usuários e nenhuma delas modela esse comportamento através do estudo da reação dos usuários à qualidade de serviço oferecida pelo *site*.

As principais contribuições deste trabalho foram publicadas em [Pereira et al., 2004b], [Franco and Meira, Jr., 2004] e [Pereira et al., 2004a]. São elas:

- Desenvolvimento de uma metodologia hierárquica para caracterização e simulação do comportamento de usuários de serviços *Internet*.
- Caracterização de *logs* reais. A *USAR* é uma metodologia genérica que pode ser aplicada a qualquer carga de trabalho de servidores *Web*.
- Simulação realística do comportamento de usuários de serviços *Internet*. Pode ser utilizada em outras áreas de aplicação como, por exemplo, geração de cargas de trabalho de servidores *Web*.

1.3 Organização do texto

A dissertação está organizada em 5 capítulos. O Capítulo 2 refere-se aos estudos existentes na área, bem como uma análise crítica dos mesmos. A metodologia *USAR*, os passos para se realizar a caracterização de cargas de trabalho *Web* capturando o comportamento dos usuários e a simulação das reações dos usuários às variações de latência são apresentados no Capítulo 3. No Capítulo 4 é apresentado um estudo de caso utilizando o *log* do servidor *proxy-cache* da UFMG, onde é realizada a caracterização da carga e a simulação do comportamento dos usuários. Conclusões e trabalhos futuros são apresentados no Capítulo 5. Por fim, é apresentada a bibliografia que foi utilizada como base para o desenvolvimento deste trabalho.

Capítulo 2

Trabalhos Relacionados

Este capítulo apresentará alguns trabalhos sobre caracterização e geração de cargas *Web* e sobre a modelagem da interação entre o homem e o computador, que estão relacionados com a dissertação. A Seção 2.1 apresenta trabalhos referentes à caracterização de cargas de trabalho *Web*, mostrando a caracterização feita utilizando-se instrumentação do *browser* e a caracterização das cargas de servidores *Web* e de comércio eletrônico. A Seção 2.2 descreve alguns trabalhos que envolvem geração de cargas de trabalho *Web* e a Seção 2.3 discute um pouco sobre pesquisas que tentam modelar o comportamento dos usuários estudando a interação entre o homem e os sistemas de computação. Por fim, a Seção 2.4 traz uma discussão conclusiva sobre todos esses estudos relacionados.

2.1 Caracterização de Cargas de Trabalho *Web*

A caracterização de cargas de trabalho é um instrumento essencial para avaliar sistemas *Web* e tem motivado vários estudos nos últimos anos, como será discutido a seguir. Entretanto nenhum desses trabalhos modela as características dos sistemas *Web* considerando os fatores que afetam a interação dos usuários com esses sistemas, como por exemplo a qualidade do serviço oferecido.

2.1.1 Caracterização do Lado do Cliente

A caracterização do comportamento do usuário, que permite identificar como o usuário reage à qualidade de serviço oferecida pelo servidor *Web*, capturada no *browser*[1] tem muito valor. No entanto, existe a dificuldade de se instrumentar os *browsers* e a limitação de se ter somente a visão do lado do cliente, além da possibilidade do experimento estar voltado para uma parcela específica de usuários. Apesar desses problemas, métodos como capturar as requisições no *proxy*, monitoração de eventos no nível do sistema operacional e falhas de privacidade nas implementações de alguns *browsers* permitiram a realização da caracterização no nível de *browsers*. Mas, mesmo assim, sem ter acesso ao código fonte dos *browsers* ou possuir APIs suficientes para a instrumentação dos mesmos, nenhuma das técnicas citadas acima permite capturar o contexto de todos os eventos de interface do usuário, limitando o escopo dos trabalhos.

Um dos primeiros trabalhos na caracterização do comportamento de clientes *Web* foi [Catledge and Pitkow, 1995], que realizou um experimento de três semanas de duração com a participação de 107 pessoas. Para tal foi utilizada uma versão instrumentada do *Xmosaic*, o que permitiu, apesar de limitações como a abrangência e a diversidade de clientes, um estudo da atividade de interface do usuário, incluindo a verificação das características dos usuários.

Ainda utilizando a instrumentação de *browsers* dos clientes, [Cunha et al., 1995] e [Crovella and Bestavros, 1996] coletaram e caracterizaram uma grande quantidade de dados que refletia a situação dos acessos de usuários à *Web* na época. Em seu trabalho [Cunha et al., 1995] mostraram que as distribuições de cauda pesada são as que melhor descrevem os dados coletados para as seguintes características: tamanho de objetos e popularidade de objetos em função do tamanho seguem *Pareto*, e o número de referências a documentos em função de sua classificação de acordo com popularidade segue *Zipf* [Zipf, 1949]. Já [Crovella and Bestavros, 1996] demonstraram e explicaram a natureza auto-similar dos pacotes que trafegam na *Internet* (tráfego WWW), concluindo que essa auto-similaridade

[1] *Browser*: interfaces gráficas para navegação na *Internet*

se deve mais ao armazenamento da informação e aos sistemas de processamento do que aos protocolos de rede e às preferências dos usuários.

Todos esses estudos são importantes, apesar de terem sido realizados há quase 7 anos atrás, pois mostram algumas características presentes até hoje na análise de tráfego WWW como, por exemplo, as distribuições de cauda pesada observadas em [Cunha et al., 1995]. Entretanto, eles não apresentam características que permitam identificar aspectos ligados à reatividade dos usuários e possibilitem a modelagem do comportamento relacionado à qualidade de serviço.

2.1.2 Caracterização do Lado do Servidor

A caracterização de cargas do lado do servidor é o estudo e entendimento da natureza e características das cargas de trabalho realizado a partir dos *logs*[2] de acesso desses servidores. A maioria das referências em relação à caracterização de carga de trabalho de servidores *Web* é focalizada apenas na caracterização de *sites* provedores de informações [Pitkow, 1998]. Os trabalhos se concentram em cargas de servidores *Web* que são compostas por uma sequência de arquivos de requisições [Arlitt and Williamson, 1996].

Em [Dilley, 1996], o autor fez um relatório técnico analisando o *log* de um servidor *Web* com o intuito de entender melhor os padrões de tráfego da época e analisar os recursos do sistema como função da carga do servidor. O relatório é bem detalhado e descreve as requisições feitas ao servidor e as características das respostas, incluindo tempo de resposta e distribuições dos tamanhos das respostas do servidor.

[Arlitt, 1996] e [Arlitt and Williamson, 1997] apresentam sugestões e avaliações sobre questões de desempenho e da utilização de *caches*. Esses dois estudos, juntamente com [Almeida et al., 1996], [Menascé et al., 2000] e [Arlitt and Williamson, 1996], apresentam ainda conclusões interessantes a respeito de propriedades e invariantes da carga de servidores *Web*. Algumas invariantes encontradas e que estão relacionadas com a caracterização apresentada nessa dissertação foram: o número de requisições a arquivos HTML e ima-

[2] *Logs*: arquivos texto que guardam informação a respeito das interações dos usuários com o servidor

gens compreende de 90% a 100% das requisições; a quantidade média de *bytes* transferidos é $\leq$ 21 KB; a distribuição dos tamanhos dos arquivos é uma distribuição de cauda pesada, seguindo a distribuição de *Pareto*; a popularidade de objetos também segue uma distribuição de cauda pesada, sendo que 10% dos arquivos acessados contabilizam 90% das requisições do servidor e a popularidade de páginas estáticas servidas por um *site* provedor de informação segue a lei de *Zipf*; e 10% dos domínios que acessam o *site* contabilizam mais de 75% do uso. Além dessas características outros trabalhos também obtiveram conclusões importantes, como por exemplo [Crovella and Bestavros, 1996], que encontraram a característica da *Web* de possuir um tráfego em rajadas, em várias escalas de tempo.

Em [Cherkasova and Phaal, 1998] os autores introduzem a noção de sessão de usuário, que consiste em requisições HTTP individuais. Essa análise prioriza o ganho de *throughput* obtido com a admissão de um mecanismo de controle cujo objetivo é garantir que qualquer sessão aceita seja completada.

[Paliouras et al., 2000] apresenta uma caracterização de grupos de usuários. Para tal, as requisições do *log* de acesso são agrupadas em comunidades, que representam padrões de uso associados a diferentes tipos de usuários. O objetivo é entender as necessidades, os interesses e o conhecimento dos usuários do *site*. Entretanto, essa análise não modela as características do sistema considerando os fatores que afetam a interação dos usuários.

Analisando trabalhos mais voltados para as cargas de trabalho de servidores de comércio eletrônico, tem-se [Krishnamurthy and Rolia, 1998], que estuda as sequências típicas de URLs que os usuários seguem enquanto navegam a fim de completar uma transação. Porém, os autores não apresentam caracterização ou propriedades de cargas geradas por usuários de comércio eletrônico atuais.

Em [Menascé et al., 1999] e [Menascé and Almeida, 2000], os autores propuseram uma metodologia baseada em grafos para a caracterização de cargas de comércio eletrônico e aplicaram-na em uma carga real para conseguir métricas relacionadas à interação dos usuários com o *site*. Em [Menascé and Almeida, 2000] são apresentados vários modelos para caracterização de carga, como por exemplo *Customer Behavior Model Graph*, o CBMG, e o modelo de visitas do usuário. Além disso mostra como obter esses modelos

dos *logs* HTTPs.

Recentemente, a caracterização de cargas de trabalho de servidores de mídia contínua (*streaming media*) tem sido estudada em alguns trabalhos, como [Almeida et al., 2001a], [Almeida et al., 2001b], [Cherkasova and Gupta, 2002] e [Veloso et al., 2002]. A análise dessas cargas é muito interessante pois elas possuem características muito próprias. Contudo, esses estudos fazem uma análise muito superficial do comportamento dos usuários, focando nas propriedades específicas do fluxo de mídia contínua.

Finalizando os trabalhos relacionados à caracterização de carga do lado do servidor, [Menascé et al., 2003] e [Veloso et al., 2002] propõem metodologias hierárquicas para caracterização de cargas de trabalho de *e-business* e *streaming media*, respectivamente, baseadas numa estratégia multi-nível que considera três níveis de caracterização: requisição, função e sessão. Contudo elas se preocupam principalmente com os aspectos mais tradicionais de caracterização de cargas *Web*, fazendo uma análise simplista do comportamento dos usuários.

Nota-se, pela quantidade e diversidade de trabalhos relacionados, que a caracterização de cargas de trabalho de servidores *Web* tem sido um assunto muito estudado ao longo do tempo e que a preocupação com a análise do comportamento dos usuários aumentou muito nos últimos anos. No entanto, nenhum dos trabalhos apresentados realiza uma modelagem mais detalhada desse comportamento, abordando aspectos importantes como o estudo da reação dos usuários às variações no tempo de resposta dos servidores.

2.2 Geração de Cargas de Trabalho *Web*

Geradores de carga de trabalho são ferramentas desenvolvidas para gerar um *log* sintético, composto por requisições que simulam requisições reais de usuários, capaz de exercitar os servidores através do envio repetitivo dessas requisições. Assim como a caracterização, a geração de cargas de trabalho é um instrumento essencial para avaliar sistemas *Web*. Sendo assim, geradores de carga de trabalho *Web* têm sido estudados extensivamente em muitos trabalhos.

Atualmente existem ferramentas como *benchmarks* - *SPECweb99* [SPECweb99, 1999] e *WebBench* [WebBench, 2002] - e geradores de carga de trabalho, tais como *SURGE* [Barford and Crovella, 1998] e *httperf* [Mosberger and Jin, 1998], que são desenvolvidas para exercitar servidores *Web* através do envio repetitivo de requisições. Além desses existem outros geradores que são mais específicos, como *Gismo* [Jin and Bestavros, 2001] e *MediSyn* [Tang et al., 2003], que submetem requisições a servidores de mídia contínua (*streaming media*), e *ProWGen* [Busari and Williamson, 2002], que é um gerador desenvolvido com o intuito de avaliar servidores *proxy-cache*.

Apesar de não ser foco dessa dissertação, a geração de cargas de trabalho *Web* está diretamente relacionada a uma das contribuições da metodologia *USAR*, que é a simulação do comportamento dos usuários. Todos esses geradores de carga citados acima são ferramentas poderosas, mas não são capazes de simular padrões de comportamento de usuário, adotando um processo de chegada de requisições que não considera o desempenho do servidor. Com isso, eles geram sempre a mesma carga independentemente das variações observadas nas interações dos usuários, causadas por mudanças na qualidade do serviço oferecido.

A partir da análise da interação dos usuários com o sistema será possível reproduzir cargas de trabalho mais realísticas, causando um aperfeiçoamento dos geradores de carga e proporcionando análises mais robustas dos servidores *Web*. Com isso será possível melhorar não somente o desempenho dos servidores, mas também outros aspectos cruciais como escalabilidade e planejamento da capacidade.

2.3 Modelagem de Comportamento de Usuário

A modelagem da interação do usuário com o sistema é um ponto muito importante tratado nessa dissertação. É através da análise das ações do usuário e as respostas dadas pelo servidor que serão modelados os aspectos do comportamento dos usuários durante a caracterização da carga de trabalho. Nessa seção serão apresentados alguns trabalhos relacionados que tratam o comportamento dos usuários e sua interação com os sistemas de computação.

Em seu estudo, [Henderson, 2001] utilizou a latência para estudar o comportamento dos usuários, mas no contexto específico de uma aplicação de jogo. Ele detectou que atrasos na rede afetam o comportamento dos jogadores, podendo provocar desistência por parte de alguns clientes e prejudicando o *site* que provê a aplicação.

[Chatterjee et al., 1998] tentou modelar o fluxo de cliques dos usuários no contexto de anúncios de propaganda via *Web*. Já [Costa et al., 2004] analisa a correlação entre requisições em aplicações *streaming media*, tentando determinar tendências no processo de interação do usuário.

Outro trabalho que estuda essa interação é [Balachandran et al., 2002], que caracteriza o comportamento dos usuários de uma rede pública sem fio, considerando a distribuição de usuários, duração das sessões, popularidade da aplicação e mobilidade. O autor pretende com essa caracterização otimizar o acesso dos usuários às estações que provêem os diversos serviços *wireless*.

[Hlavacs and Kotsis, 1999] propôs um *framework* para modelagem do comportamento do usuário, estruturado e construído de maneira *top-down*, que consiste de várias camadas e é baseado em modelos matemáticos. Este trabalho se preocupa principalmente em modelar o tráfego HTTP que é utilizado para simulação do tráfego na rede. Ele também foi usado como base para a construção de um gerador de carga orientado ao usuário [Hlavacs et al., 2000].

Nenhum desses estudos, entretanto, modela o comportamento dos usuários analisando aspectos ligados à reatividade da interação entre usuário e servidor.

2.4 Sumário

As seções anteriores apresentaram diversos trabalhos relacionados à caracterização e geração de cargas de trabalho *Web* e à modelagem de comportamento de usuário. Essa seção apresenta uma síntese da discussão apresentada sobre todos esses trabalhos.

A caracterização de cargas de trabalho é um instrumento essencial para avaliar sistemas *Web* e tem motivado vários estudos nos últimos anos, como foi discutido na Seção 2.1. Essa

caracterização pode ser realizada tanto do lado do cliente, tentando capturar características do comportamento dos usuários a partir da sua interação com o *browser*, como do lado do servidor, que procura entender a natureza e as características das cargas de trabalho a partir da análise dos *logs* de acesso dos servidores *Web*. Essa caracterização permite a melhora da qualidade dos serviços oferecidos, que pode ser vista tanto como melhora no tempo de resposta a um usuário, como a personalização do serviço.

A geração de cargas de trabalho *Web*, apesar de não ser foco do trabalho, foi discutida por estar diretamente relacionada a um dos objetivos propostos pela metodologia *USAR*, que é a simulação do comportamento dos usuários. Geradores de carga de trabalho são ferramentas muito estudadas atualmente e são desenvolvidas para gerar um *log* sintético capaz de exercitar os servidores através do envio repetitivo de requisições.

Um dos principais benefícios da caracterização e da geração de carga é que elas permitem a construção de modelos analíticos e simuladores que possam replicar o comportamento do usuário de modo que se possa estudar o desempenho de sistemas similares em um ambiente de laboratório, onde é possível medir com maior precisão os efeitos dos vários tipos de requisições de usuários, e ainda construir modelos mais precisos da utilização dos recursos do sistema.

Outra área importante relacionada é a que trata a modelagem do comportamento dos usuários, realizada através do estudo da interação dos usuários com o sistema. Na Seção 2.3 foram apresentados alguns trabalhos relacionados que tratam o comportamento dos usuários e sua interação com os sistemas de computação. A maioria desses trabalhos se baseiam em modelos matemáticos para descrever aspectos como duração das sessões e popularidade de objetos e, a partir desses aspectos, modelar, simular e otimizar o tráfego na rede.

A principal limitação dos trabalhos citados nesse capítulo é que eles fazem análise apenas superficial do comportamento dos usuários, deixando de analisar aspectos importantes desse comportamento como, por exemplo, o impacto que a qualidade de serviço oferecida pelos servidores *Web* tem sobre as ações realizadas pelos clientes na aplicação. Apesar das limitações, todos esses trabalhos mostram a evolução do processo de caracterização

de carga com o foco no comportamento do usuário. Seguindo nessa mesma direção, e visando aprimorar esse processo, a metodologia *USAR* traz três inovações à caracterização de comportamento de usuários de serviços *Internet*:

- a metodologia hierárquica de caracterização de cargas de trabalho *Web*, que adiciona o nível de usuário;
- a modelagem do comportamento dos usuários, realizada pelo estudo detalhado das reações dos usuários às variações na qualidade do serviço oferecido, baseado na análise da relação entre IAT e latência; e
- a simulação realística do comportamento desses usuários.

Capítulo 3

A Metodologia *USAR*

Neste capítulo será apresentada a metodologia *USAR*, que envolve uma metodologia de caracterização de cargas de trabalho *Web*, com foco na modelagem do comportamento dos usuários e uma metodologia de simulação de comportamento de usuário, baseada na simulação de reações dos usuários às variações de latência observadas.

A Seção 3.1 apresenta o modelo de interação dos usuários com os servidores *Web*, a fim de ilustrar como será analisada a carga de trabalho durante a caracterização. A Seção 3.2 descreve a metodologia hierárquica - composta pelos níveis de requisição, ação, sessão e usuário - para caracterização de cargas de trabalho *Web*. Por fim, a Seção 3.3 apresenta a metodologia de simulação de comportamento de usuário.

3.1 Modelo de Interação de Usuários com Servidores *Web*

Nesta seção será apresentado um modelo de interação dos usuários com os servidores *Web*, com o intuito de esclarecer alguns conceitos utlizados durante a dissertação e ilustrar como será analisada a carga de trabalho por todo o processo de caracterização.

Durante uma visita a um *web site*, o usuário interage com o sistema realizando ações, que são recebidas pelo servidor como requisições à execução de funções. As funções pre-

sentes em um *site* dependem da natureza do mesmo e variam desde a visualização do conteúdo de uma página estática (em *sites* antigos era a única função disponível) até as funções dinâmicas, como "busca" ou "navegação", muito utilizadas atualmente. Em *sites* de comércio eletrônico, que comercializam produtos e serviços, existem ainda funções mais complexas como "pagamento" de uma compra, ou "transferência" de uma certa quantia de dinheiro entre contas correntes (no caso do *site* de um banco).

A execução de uma função sempre é realizada por uma ação do usuário. Uma simples ação pode gerar muitas requisições HTTP para o *site*. Por exemplo, em resposta a uma ação de navegação poderão vir várias figuras para serem mostradas na página junto com a resposta à ação. Além dessas figuras, existem ainda outros tipos de requisições, como é o caso de anúncios de propagandas, que têm sido muito utilizados atualmente junto das respostas de execução de funções.

A Figura 3.1 mostra o que acontece durante a interação do usuário com o servidor. Quando uma ação (que possui uma função associada) é executada, são geradas a requisição que corresponde ao objeto solicitado pelo usuário e outras requisições HTTP a imagens embutidas. O servidor as recebe, processa e envia uma resposta ao usuário contendo os objetos requisitados. Essa resposta é a página visualizada no *browser* do cliente.

A visita do usuário a um *web site*, todo o conjunto de ações realizadas, requisições geradas, e funções que estejam próximas no tempo, é chamada de sessão. Visto o protocolo de funcionamento de requisições e ações executadas em um *web site*, a próxima seção irá mostrar como é realizada a caracterização da carga em diferentes níveis.

3.2 *USAR* - Caracterização de Cargas de Trabalho *Web*

Como descrito no Capítulo 2, pesquisas anteriores se esforçaram em criar metodologias, algumas delas hierárquicas, para caracterização de cargas de trabalho considerando métricas tanto do lado do usuário quanto do servidor, mas ignorando a correlação entre esses dois lados. Para preencher esse espaço a metodologia *USAR* propõe uma nova estratégia de caracterização de cargas de trabalho *Web*, chamada nessa dissertação de metodologia de

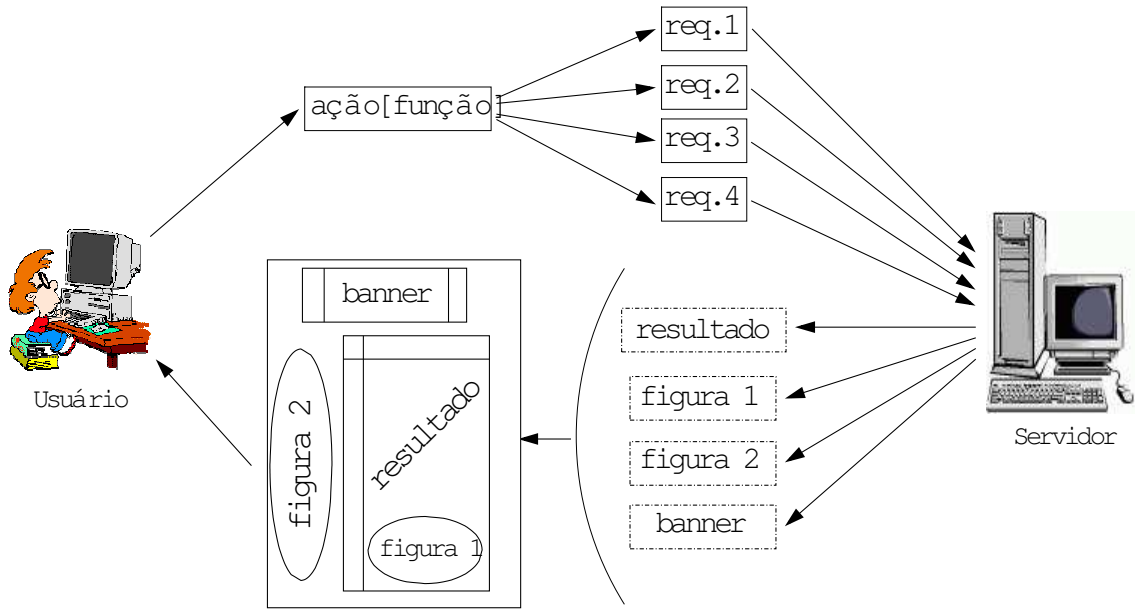


Figura 3.1: Interação do usuário com um *web site*

caracterização *USAR*, que estende [Menascé et al., 2000, Menascé et al., 2003] e tem o foco na análise e modelagem do comportamento do usuário.

Baseada no modelo hierárquico proposto em [Menascé et al., 2000, Menascé et al., 2003], a metodologia de caracterização *USAR* define níveis de acordo com as diferentes formas de abstrair a interação do usuário (representada na Figura 3.1) com o *site* em uma sessão. Dessa forma, a metodologia de caracterização *USAR* consiste de quatro níveis - *Usuário*, *Sessão*, *Ação* e *Requisição* - para caracterizar a carga de trabalho, capturando e modelando aspectos do comportamento dos usuários.

A Figura 3.2 mostra o modelo de hierarquia multinível, proposto nessa dissertação, para a caracterização de carga de trabalho de servidores *Web*, onde o nível de *Usuário* permite analisar e modelar o comportamento reativo[1] dos usuários.

A idéia dos níveis hierárquicos (veja Figura 3.2) pode ser usada para capturar as mudanças no comportamento do usuário e mapear os efeitos dessas mudanças nas camadas mais baixas do modelo. Com isso, a análise da carga de trabalho é conduzida por diferentes perspectivas e facilita o processo de caracterização uma vez que pode ser feito de acordo

[1]Comportamento reativo: comportamento refletido na reação a uma latência observada

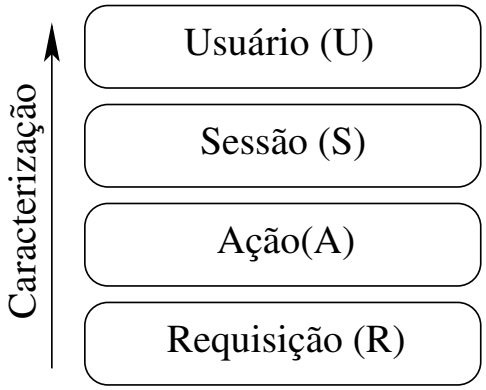


Figura 3.2: Hierarquia da metodologia de caracterização *USAR*

com diferentes visões associadas a cada nível, tornando-o mais claro e produzindo uma caracterização mais completa e detalhada.

A abordagem desse trabalho é analisar cada camada individualmente para obter uma caracterização dos diferentes níveis, com o foco na interação do usuário com o *site* e em como a qualidade do serviço oferecido afeta o seu comportamento. Os quatro níveis de caracterização serão descritos em maior detalhe a seguir.

Caracterização do Nível de *Requisição*

A caracterização do nível de *Requisição* requer a preparação de um *log L*, gerado a partir da junção dos *logs* HTTP de todos os servidores *Web*, caso haja mais de um, dos *sites* para os quais será realizada a caracterização da carga.

Com a geração do *log L*, neste nível é realizada a análise das requisições HTTP que chegam ao *site*, sem levar em consideração a ação, sessão e usuário associado. São estudadas características como a distribuição de chegada de requisições e as distribuições de tamanho dos objetos e de popularidade associada ao tamanho dos objetos. A análise da distribuição de chegada de requisições em diferentes escalas de tempo permite determinar se há uma forte dependência entre as requisições, o que é caracterizado por longas sequências de acréscimo ou decréscimo na intensidade do tráfego em escalas de tempo intermediárias (por exemplo 5 minutos).

Outra análise importante do nível de *Requisição* é a quantificação multi-escalar da dependência do processo de chegada, que é realizada pelo gráfico VTP (*variance time plot*). O VTP é um gráfico *log-log* da variação no processo de chegada de requisições pela escala de tempo e pode ser usado para detectar e quantificar auto-similaridade.

Caracterização do Nível de *Ação*

Como foi explicado na seção 3.1, uma simples ação do usuário pode gerar várias requisições. Com isso, a caracterização do nível de *Ação* demanda a geração de um *log La* com as requisições relevantes, que estão diretamente relacionadas às ações. Para tal são retiradas todas as requisições relativas a imagens embutidas ou erros.

Com a geração do *log La*, neste nível são analisadas as ações realizadas pelo usuário, que são normalmente cliques que ativam uma ligação (*link*), e as funções oferecidas por um *web site*. Deve ser determinada a frequência de execução de cada função e analisada a sua execução em diferentes escalas de tempo, comparando com os padrões observados para o processo de chegada de requisições em escalas de tempo similares. Além disso, é importante analisar a variabilidade das latências correspondentes às ações, a fim de determinar a influência que essas variações de latência têm nas ações dos usuários sobre o sistema.

Deve-se lembrar que as funções variam de acordo com a natureza do *site* em questão, podendo ser desde "visualizar uma página" até "enviar a declaração de imposto de renda". Com isso é importante durante a caracterização observar tanto as funções que são mais requisitadas quanto aquelas que são mais importantes para o objetivo do *site*.

Caracterização do Nível de *Sessão*

A caracterização do nível de *Sessão* requer a preparação de um *log Ls*, gerado a partir do agrupamento das ações de um mesmo usuário e a determinação de um *threshold* τ para delimitar cada uma das sessões. Normalmente define-se que uma sessão é delimitada pelo período de inatividade do usuário. Em outras palavras, se um usuário não faz nenhuma

requisição ao servidor por um período maior que τ, sua sessão é considerada finalizada.

Após a geração do *log* Ls, a caracterização deste nível compreende o estudo da distribuição do tamanho (em quantidade de requisições) das sessões, além de aspectos quantitativos como duração das sessões e a composição das sessões em termos de funções.

No nível de *Sessão* é realizada também a análise do processo multi-escalar de início das sessões. Para tal, deve ser feito um gráfico do número de sessões iniciadas por unidade de tempo, em diferentes escalas de tempo. Essa análise é importante porque os recursos do *site* são normalmente alocados para cada sessão.

Caracterização do Nível de *Usuário*

No contexto dessa dissertação o objetivo principal é descrever a caracterização no nível de *Usuário*. Este nível modela o comportamento do usuário, mais especificamente, a reação do usuário a latências variadas. Com isso, é possível modelar como o tempo de resposta afeta as ações do usuário sobre o sistema.

É importante ressaltar que a caracterização de carga aqui apresentada é realizada sobre o *log* de acesso de servidores *Web*. Dessa forma, o comportamento dos usuários é definido por uma sequência de ações que permitem determinar a maneira como os usuários interagem com o *site*, reagindo à qualidade de serviço observada.

A seguir serão descritos os passos da metodologia de caracterização *USAR* para analisar e modelar o comportamento dos usuários:

1. Análise dos usuários pelas seguintes perspectivas: *IATs* entre ações de um mesmo usuário, latência associada a essas ações e correlação entre *IAT* e latência:

    - Discretização das medidas de *IAT* e latência usando uma função que as correlacione. Nesta dissertação, foi utilizada uma função que leva em consideração a razão e a diferença entre elas, como será melhor discutido no Capítulo 4. Essa discretização permite classificar e agrupar as ações que são similares em termos das duas medidas.

- Transformação das sessões de usuário em sequências de ações usando, por exemplo, o critério de discretização citado acima.

2. Análise do padrão de acesso dos usuários a partir das ações executadas. Pode ser realizada utilizando-se o CBMG (*Customer Behavior Model Graph*) [Menascé et al., 2000]. Determina quais são as transições mais frequentes entre duas ações quaisquer.

3. Identificação de tendências de comportamento a partir das transições entre as ações. Define grupos de sequências de ações que representem padrões de interação do usuário com o sistema. Para essa tarefa pode-se usar, por exemplo, algum algoritmo de mineração de sequências como [Zaki, 2001].

4. Definição dos diferentes tipos de comportamento de usuário presentes no *log*:

   - Agrupamento das sessões de usuário de acordo com a sua similaridade, em termos de tendências de comportamento. Métodos eficientes utilizam técnicas de clusterização, como por exemplo a técnica *K-Means* [Garner, 1995].
   - Análise dos grupos de sessões de usuário que possuem tendências similares de comportamento, definindo os tipos distintos de comportamentos de usuário observados no *log*.

Com a realização desses passos, a modelagem do comportamento dos usuários está completa, partindo da análise de *IAT* e latência das ações e terminando na definição dos tipos de comportamento de usuário presentes no *log*. A próxima seção mostra como é realizada a simulação do comportamento modelado.

3.3 *USAR* - Simulação de Comportamento de Usuário

Nesta seção será apresentada a metodologia para simulação do comportamento dos usuários, chamada de metodologia de simulação *USAR*. A metodologia de simulação *USAR* é baseada

na geração de um *log* sintético composto por ações de usuário, que imita o comportamento dos usuários reais.

Assim como na metodologia de caracterização *USAR*, a metodologia de simulação *USAR* também utiliza um modelo de hierarquia multinível, mostrado na Figura 3.3, para guiar a simulação do comportamento reativo dos usuários. Além disso, o processo de simulação do comportamento se baseia na modelagem do comportamento dos usuários, realizada durante a caracterização de carga.

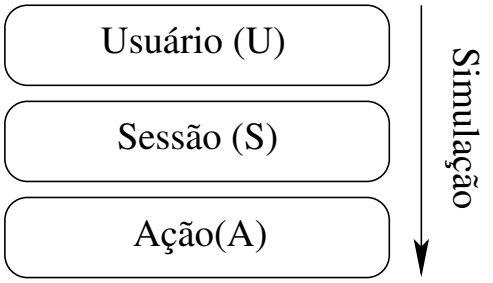


Figura 3.3: Hierarquia da metodologia de simulação *USAR*

Analisando a Figura 3.3 nota-se duas diferenças em relação à hierarquia de caracterização:

1. A simulação é realizada de maneira *top-down* (de cima para baixo), começando pelo nível de *Usuário*, e a caracterização é realizada de maneira *bottom-up* (de baixo pra cima), começando pelo nível de *Requisição*.

2. A simulação não envolve o nível de *Requisição*, uma vez que ela começa pelo nível de *Usuário* e compreende a geração de ações dos usuários.

É importante ressaltar que a geração das ações de usuário é suficiente para simular o comportamento dos usuários, uma vez que esse comportamento foi modelado a partir do estudo da relação entre *IAT* e latência de ações. Entretanto, trabalhos futuros que desejem realizar a geração de carga de trabalho utilizando essa simulação, devem considerar também o nível de *Requisição* para serem capazes de gerar requisições que possam ser submetidas a servidores *Web*.

Simulador de Ações de Usuário

Com o intuito de replicar as propriedades do comportamento do usuário, foi implementado um simulador de ações de usuário, como mostrado na Figura 3.4, que segue o modelo de hierarquia multinível e trabalha com três tipos de dados:

1. Dados de entrada: especificam as características dos usuários que interagem com o sistema. Os dados de entrada definem os tipos de comportamentos de usuários, suas tendências de comportamentos e as seguintes distribuições que são obtidas durante a caracterização de cada nível:

    - Usuário: são definidas a distribuição de probabilidade dos tipos de usuário, a distribuição de probabilidade das tendências de comportamentos que compõem cada tipo.
    - Sessão: é definida a probabilidade de distribuição dos diversos tamanhos de sessão (definidos pela quantidade de requisições). Essa distribuição será utilizada para determinar o número de ações de usuário em cada sessão.
    - Ação: é determinada a distribuição das ações de usuário em sequências que estão diretamente relacionadas às tendências de comportamento. Como resultado, é especificada a popularidade das ações em cada tendência identificada.
    - Requisição: o último nível envolve a geração das requisições que compõem a carga de trabalho *Web*. Para uma geração de carga precisa deve-se determinar as distribuições de popularidade e de tamanho dos objetos. Essas distribuições não são usadas na geração das ações dos usuários, portanto não são tratadas neste trabalho, mas são importantes para transformar essas ações em requisições reais que podem ser submetidas a um servidor real.

2. Parâmetros de execução: parâmetros que são informados para cada execução, tal como o número de sessões que serão simuladas.

3. Dados de saída: um arquivo de *log* composto de ações de usuário. Essas ações são agrupadas em sessões de usuário e simulam de forma precisa o comportamento real modelado durante a caracterização.

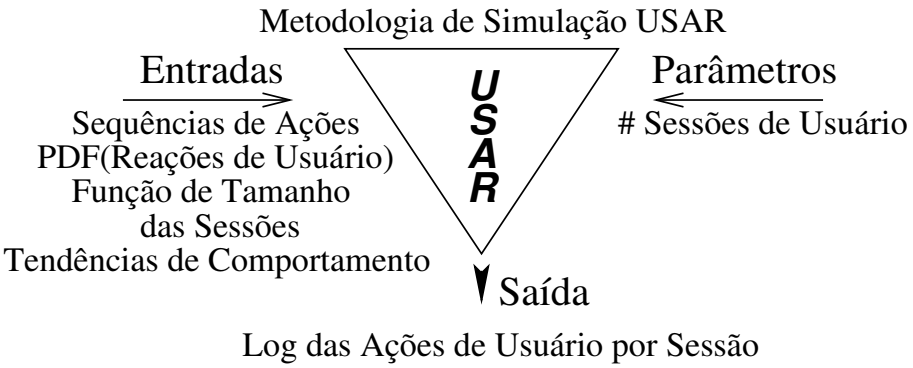


Figura 3.4: O simulador *USAR*

Com a utilização desse simulador torna-se possível equiparar as características do *log* gerado às características do *log* real, a fim de se verificar a precisão do processo de simulação do comportamento a partir da caracterização e modelagem do comportamento dos usuários.

Geração de Cargas de Trabalho Reativas

Como já foi descrito no Capítulo 2 (na Seção 2.2), os geradores de carga atuais, tais como SURGE [Barford and Crovella, 1998] ou httperf [Mosberger and Jin, 1998], não permitem a simulação das reações de usuários a variações no tempo de resposta do servidor. Portanto, a geração de *logs* sintéticos que se preocupam com a latência permitiria a melhoria de geradores de carga, proporcionando a geração de cargas de trabalho *Web* que consideram o impacto real da qualidade de serviço oferecida sobre as diversas ações dos usuários.

Essa inovação pode trazer um grande ganho na qualidade da carga de trabalho gerada, reduzindo a distância entre os modelos tradicionais de geração de carga e a carga de trabalho real. Com isso, o objetivo principal da estratégia de simulação da metodologia *USAR* é ser o ponto de partida para a construção de um gerador de carga que através da

simulação do comportamento dos usuários mimetize as reações de usuários reais, baseado na relação entre *IAT* e latência das ações de usuário.

Sumário

Esse capítulo apresentou a metodologia *USAR*, que é dividida em duas partes principais: caracterização de cargas de trabalho *Web*, com foco na modelagem do comportamento dos usuários; e simulação desse comportamento, realizada através da simulação de reações dos usuários às variações de latência observadas.

A metodologia *USAR* divide a caracterização de carga em níveis, definidos de acordo com as diferentes formas de abstrair a interação do usuário com o servidor *Web*. A análise realizada através dos níveis de *Requisição*, *Ação*, *Sessão* e *Usuário* permite a caracterização completa da carga de trabalho, definindo aspectos importantes como o processo de chegada de requisições, a frequência de execução das funções e a distribuição de tamanho das sessões. Além disso, a principal contribuição da metodologia está na modelagem do comportamento dos usuários, que é realizada de forma detalhada baseada na relação entre *IAT* e latência das requisições, definindo tendências de comportamento a partir da análise das sequências de ações dos usuários.

Outra inovação importante apresentada é a metodologia de simulação *USAR*. Ela permite replicar o comportamento modelado gerando ações de usuário que apresentem um aspecto reativo em relação à latência observada para a requisição anterior, e reflitam o impacto real que a qualidade do serviço oferecido pelo *site* tem sobre os seus usuários.

A seguir, o Capítulo 4 irá apresentar um estudo de caso utilizando o *log* do servidor *proxy-cache* da UFMG, mostrando a aplicação da metodologia *USAR* para a caracterização da carga de trabalho e a simulação do comportamento dos usuários.

Capítulo 4

Estudo de Caso

Este capítulo apresenta um estudo de caso que demonstra a eficiência e aplicabilidade da metodologia *USAR*, tanto para caracterizar cargas de trabalho de servidores *Web*, quanto para modelar e simular o comportamento dos usuários. O foco desse estudo de caso está nas inovações da metodologia, mais especificamente a caracterização do nível de *Usuário*, por isso algumas análises consideradas irrelevantes para a modelagem do comportamento dos usuários não serão realizadas. Contudo serão avaliados aspectos básicos de caracterização e explicados os principais resultados relacionados aos outros níveis - *Sessão*, *Ação* e *Requisição*.

Na Seção 4.1 serão apresentados os dados utilizados nesse estudo de caso, relativos ao servidor *proxy-cache* da UFMG. As duas seções seguintes mostram a aplicação da metodologia *USAR*: a Seção 4.2 descreve o processo de caracterização da carga de trabalho e a modelagem do comportamento dos usuários e a Seção 4.3 apresenta a simulação do comportamento modelado.

4.1 Dados para a Caracterização

Os servidores *Web* podem ser configurados para armazenar informação sobre todas as requisições de seus clientes. Os arquivos que contêm essas informações são chamados de

arquivos de *log*, ou simplesmente *log*. Para se caracterizar a carga de um servidor, utiliza-se o *log* de acesso do mesmo, que contém informação sobre as requisições processadas pelo servidor. Cada linha desse arquivo contém informação sobre uma única requisição para um documento.

A caracterização aqui apresentada foi realizada sobre um *log* de dois meses de dados do servidor *proxy-cache Squid* da Universidade Federal de Minas Gerais (UFMG). O *log* contém uma entrada para cada requisição feita por um usuário e consiste das seguintes informações:

- *Timestamp*: momento em que o *socket* do cliente foi fechado. Formato UTC (segundos desde 1 de Janeiro de 1970), com fração de milisegundos;
- *Tempo Transcorrido*: para conexões HTTP persistentes, é o tempo entre a leitura do primeiro *byte* da requisição e a escrita do último *byte* da resposta. Para o usuário pode ser considerado como a latência para obtenção do recurso;
- *Endereço Cliente*: endereço IP do cliente requisitante;
- *Log Tag e Código HTTP*: status do objeto no *cache* (*hit, miss*, etc) e código da resposta, tirado do cabeçalho HTTP;
- *Tamanho*: quantidade de *bytes* transferidos para o cliente;
- *Método da Requisição*: método da requisição HTTP;
- *URL*: *URL* requisitada;
- *Identificação do Usuário*: sempre '-' para o *log* analisado (não foi coletado);
- *Dados da Hierarquia*: status e *host* para *caches* que estejam em uma hierarquia;
- *Tipo de Conteúdo*: campo *Content-type* obtido na resposta HTTP.

Um exemplo de uma linha do *log* de acesso do *Squid* é:

```
1074011245.011   4896 150.164.10.196 TCP_MISS/200 17225 GET
http://www.ufmg.br/index.html - DIRECT/- text/html
```

4.2 *USAR* - Caracterização de Cargas de Trabalho *Web*

Nessa seção será apresentada a aplicação da metodologia de caracterização *USAR* sobre o *log* utilizado nesse estudo de caso. Serão mostrados aspectos de caracterização do nível de *Requisição* ao nível de *Usuário*, com um enfoque maior à caracterização realizada no nível de *Usuário*, onde é feita a modelagem do comportamento dos usuários.

Na caracterização aqui apresentada foram usados somente quatro semanas de *log*, visto que uma análise inicial mostrou que a quantidade de informação obtida com essas quatro semanas é similar à informação obtida com o período de dois meses. A Tabela 4.1 apresenta informação geral sobre o *log*, que contém mais de 2 milhões de requisições por semana e uma quantidade significativa de *bytes* transferidos, objetos e IPs únicos, e sessões de usuário.

	Semana 1	**Semana 2**	**Semana 3**	**Semana 4**
#Requisições	2439146	2099018	2148717	2079594
MegaBytes	34408.62	19862.51	16805.13	20743.45
#IPs únicos	488	485	497	507
#Sessões	6952	6979	7369	7756
#Objetos únicos	482186	413122	428665	396996

Tabela 4.1: Informação geral sobre a carga de trabalho

As próximas seções apresentam a caracterização hierárquica em cada um dos quatro níveis.

4.2.1 Caracterização do Nível de *Requisição*

Na caracterização do nível de *Requisição* o foco está na requisição, sem levar em consideração a ação, sessão e usuário associado. O *log* L contém os acessos de usuários de uma das

maiores universidades federais no Brasil e possui um número considerável de requisições por dia.

Durante a caracterização desse nível foram analisadas em torno de 9 milhões de requisições feitas por quase 500 endereços IP assinalados estaticamente, gerando um tráfego de quase 90 *Gigabytes*, como mostrado na Tabela 4.1. A Tabela 4.2 apresenta os resultados acerca da utilização de protocolos. Como esperado, prevalece o acesso através do protocolo HTTP, que representa aproximadamente 98% das requisições, sendo o restante dos acessos via protocolo HTTPS(SSL) e FTP. Nota-se que para os acessos utilizando *SSL* não há HIT - requisições atendidas pelo servidor *proxy-cache* sem necessidade de submissão ao servidor final - entretanto, a taxa de HIT é bem significativa, representando em torno de 48% das requisições.

	#Requisições			**MegaBytes**		
Protocolo	**Quant.**	**Porcentagem**	**%Hit**	**Quant.**	**Porcentagem**	**%Hit**
HTTP	8600202	98%	48%	89038.03	97%	8%
SSL	165334	2%	-	927.28	1%	-
FTP	939	0%	2%	1854.40	2%	11%

Tabela 4.2: Resumo da utilização de protocolos

A Tabela 4.3 apresenta a frequência de acesso dos tipos de requisições mais solicitados. Nota-se que mais de 65% das requisições estão associadas a imagens, o que demonstra o grande impacto desse tipo de requisição para o servidores *Web*. Ao analisar-se a quantidade de *bytes* requisitados, a contribuição das imagens cai para 12% e os filmes passam a representar 48% dos dados transferidos. As requisições que representam consultas em máquinas de busca também representam uma parcela significativa (13%). Já as requisições por páginas HTML, arquivos texto, executáveis e outros tipos menos solicitados somam 22% de todos os acessos.

A análise dos dados ao nível de *Requisição* mostra que dimensões tradicionais de caracterização, tais como tamanho dos objetos, popularidade de objetos, e distribuição de chegada de requisições, se comparam a caracterizações anteriores do mesmo tipo de tráfego

Tipo de Requisição	#Requisições			MegaBytes		
	Quant.	**Porcent.**	**%Hit**	**Quant.**	**Porcent.**	**%Hit**
Imagem	6294924	65%	61%	10612.12	12%	35%
Consulta	1304332	13%	-	8063.36	9%	-
Diretório	402140	4%	11%	2605.76	3%	7%
HTML	381656	4%	25%	1718.40	2%	11%
Filme	6648	0%	14%	44018.99	48%	1%
Outros	1577965	18%	52%	24801.08	26%	9%

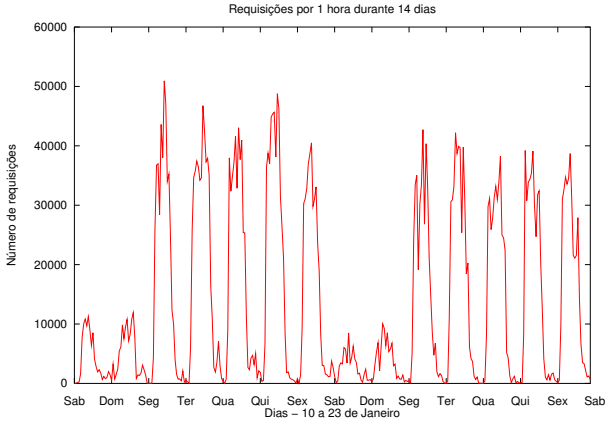
Tabela 4.3: Frequência de acesso por tipo de requisição

[Barford and Crovella, 1998, Menascé et al., 2003]. A Figura 4.1 mostra três gráficos que esboçam o processo de chegada de requisições em três escalas de tempo diferentes: uma hora, cinco minutos e cinco segundos. A análise visual dos gráficos revela, aparentemente, uma dependência forte que mostra longas sequências de aumento e diminuição no volume de requisições, evidenciada, especialmente, em escalas de tempo intermediárias. Apesar de ser de extrema importância para o processo de caracterização da carga, a análise do processo de chegada de requisições de maneira mais detalhada (utilizando VTP, por exemplo) não será realizada, pois não é relevante para a modelagem do comportamento dos usuários.

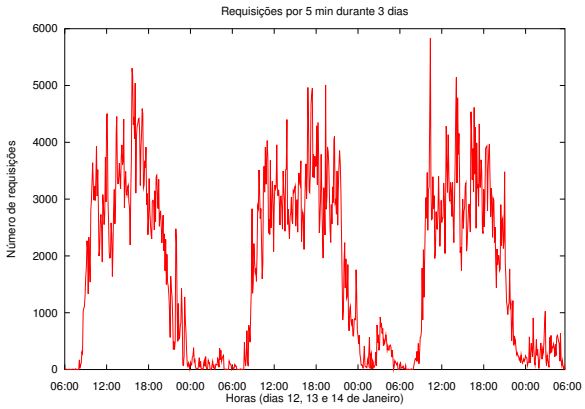
A Figura 4.2 apresenta dois gráficos que mostram as distribuições de probabilidade de popularidade de objetos e de tamanho dos objetos. Em ambos os casos, pode-se ver claramente que as duas distribuições são bastante inclinadas, como observado em outras caracterizações de tráfego *Web*.

4.2.2 Caracterização do Nível de *Ação*

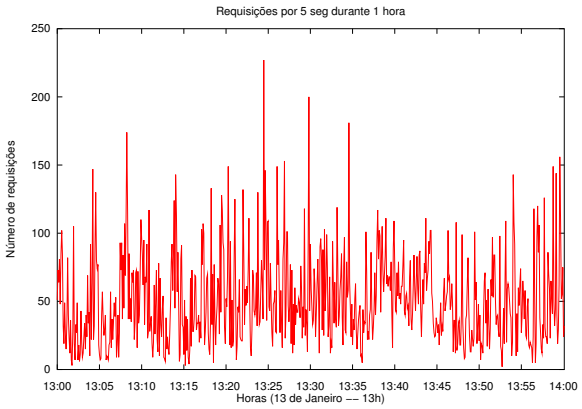
Nesta seção, o *log* de acesso será analisado a partir das ações executadas pelos usuários. Como foi explicado na seção 3.1, uma simples ação do usuário pode gerar várias requisições.



(a) Chegadas por hora durante duas semanas



(b) Chegadas a cada 5min durante 3 dias

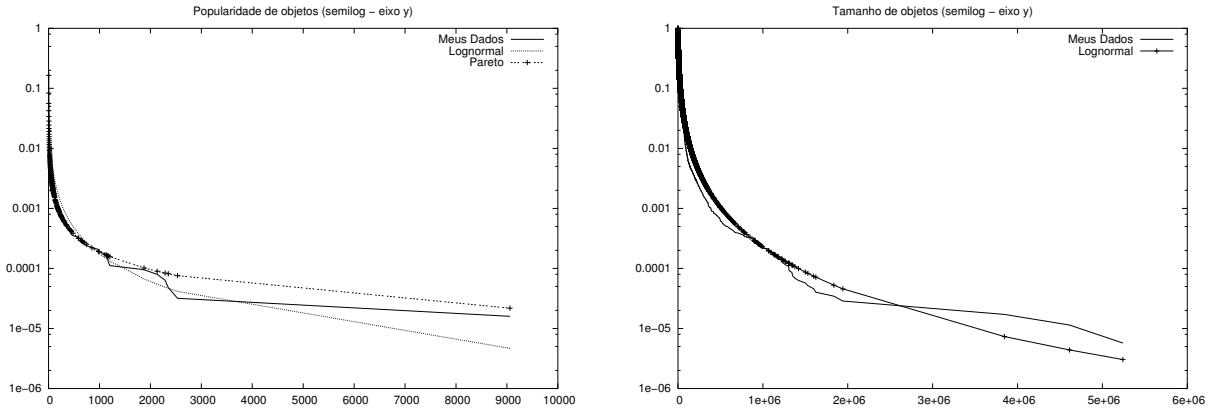


(c) Chegadas a cada 5s durante uma hora

Figura 4.1: Número total de requisições chegando ao *proxy-cache* em diferentes escalas de tempo

Com isso, a caracterização do nível de *Ação* demanda a geração de um *log* temporário com as requisições relevantes, que estão diretamente relacionadas às ações. Com o intuito de capturar o comportamento dos usuários ao interagir com o servidor e perceber variações no tempo de resposta às suas requisições, foram realizadas ainda duas operações de "limpeza" do *log*, que permitiram analisar mais precisamente as reações dos usuários às diversas latências observadas:

1. Retirada das requisições que foram atendidas pelo servidor *proxy-cache* (HITS). Os HITS são respondidos com muita rapidez e não permitem que o usuário observe aumento no tempo de resposta, não sendo úteis na análise que se deseja realizar nesse estudo de caso.



(a) Popularidade de objetos (b) Tamanho de objetos

Figura 4.2: Caracterização do nível de *Requisição*

2. Retirada das requisições a objetos que não sejam HTML. Objetos como imagem ou vídeo, por exemplo, demoram um certo tempo para serem retornados. Com isso o usuário, que já está esperando um tempo de resposta maior, normalmente realiza outras ações no *site* enquanto espera. Com a retirada desses objetos, a análise se torna mais focada nas ações que realmente afetam o comportamento dos usuários, quando estes percebem variações não esperadas no tempo de resposta do servidor.

Após a filtragem do *log*, a caracterização consiste na identificação da natureza das ações, que estão estritamente associadas às funções providas pela aplicação, e no estudo da intensidade da carga que essas ações impõem ao servidor, dentre outras análises relevantes.

Uma vez que a análise é baseada em um *log* de um servidor *proxy-cache*, não é possível determinar a natureza da ação do usuário simplesmente analisando a URL e os outros dados que compõem o *log*, pois não possuímos informação sobre o serviço sendo provido. Com isso, no contexto desse estudo de caso, os tipos de ação e as funções associadas a cada ação não são tão relevantes como seriam para outros domínios de aplicações, como por exemplo um serviço de *E-business* onde seja possível identificar funções como “compra”, “pagamento”, ou “navegação”.

A Tabela 4.4 mostra que 48,15% das requisições que chegam ao servidor *proxy-cache* são requisições relativas a ações dos usuários. Com a filtragem realizada para análise do

Quantidade de Ações (total)	4221426
Quantidade de Ações (sem HIT)	2889570
Quantidade de Ações (objetos HTML)	223695
Porcentagem de Ações (total)	48,15%
Porcentagem de Ações (sem HIT)	32,96%
Porcentagem de Ações (objetos HTML)	2,55%
MBytes transmitidos (total)	78.142,90
MBytes transmitidos (sem HIT)	71.477,94
MBytes transmitidos (objetos HTML)	2.774,86

Tabela 4.4: Informação quantitativa do *log* do servidor *proxy-cache* no nível de *Ação*

comportamento reduziu-se para 2,55% a quantidade de requisições a serem analisadas, o que resulta em 223695 ações de usuário considerando somente as requisições a objetos HTML. Além disso, pode-se observar que os resultados da execução de ações transferiram 78.142,90MB de dados, aproximadamente 85,1% do que foi transferido através do servidor *proxy-cache*. Isso mostra que as ações em geral têm grande impacto na intensidade da carga transmitida. Desses dados, 2.774,86MB (3,02%) correspondem aos objetos HTML, o que é esperado uma vez que os objetos HTML são geralmente objetos pequenos.

Finalizando a caracterização do nível de *Ação*, foi modelada a distribuição de probabilidade das latências das ações e o resultado se mostrou compatível com uma distribuição *lognormal*, como pode ser visto no gráfico da Figura 4.3. Esta compatibilidade demonstra a grande variabilidade das latências observadas, o que instiga a investigação da correlação entre o tempo de resposta do servidor e a reação do usuário.

4.2.3 Caracterização do Nível de *Sessão*

A análise apresentada nesta seção é feita baseada na seção do usuário. Como descrito na seção 3.1, sessão é o nome dado à visita de um usuário a um *web site*. Para esse estudo de caso, será considerada sessão um conjunto de ações executadas por um usuário, submetidas ao servidor *proxy-cache*, durante um determinado período de tempo.

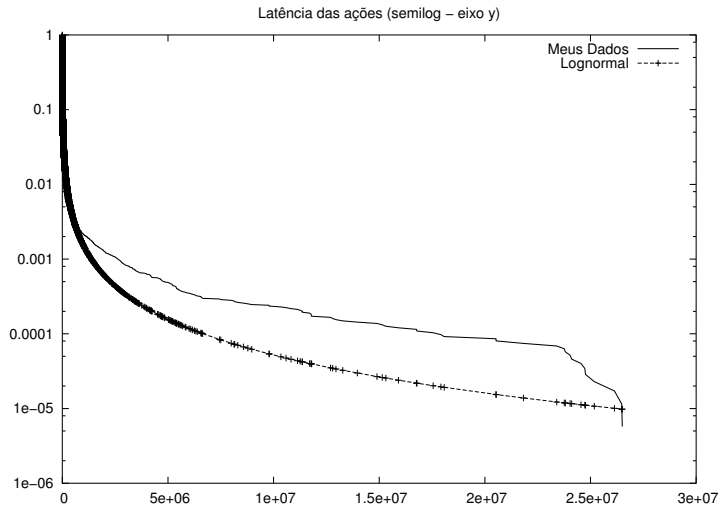


Figura 4.3: Latência das ações

Para a realização desse estudo, foi necessário o processamento do *log* de acesso, que fora filtrado para a análise realizada na seção 4.2.2, de forma que ele contivesse as requisições separadas por sessão de usuário.

O primeiro passo desse processamento é a identificação das requisições de cada usuário individualmente. Para *logs* de acesso que possuem identificadores, como *cookies* ou simplesmente identificadores de sessão, essa tarefa é trivial. Entretanto, o *log* do servidor *proxy-cache* não possui esses facilitadores. Com isso, a separação dos usuários foi feita pelo endereço IP.

Realizada a identificação dos usuários e suas requisições, a próxima etapa é a divisão do *log* em sessões. Para isso, é necessário identificar os limites de uma dada sessão. Foi definido que uma sessão é delimitada pelo período de inatividade do usuário. Em outras palavras, se um usuário não faz nenhuma requisição ao servidor por um período maior que um *threshold* τ, sua sessão é considerada finalizada. Alguns servidores forçam esse *threshold* e fecham as sessões inativas de forma a economizar os recursos gastos por elas. Como o servidor *proxy-cache* não fornece tal informação, é preciso estimar o τ do *log*. Esse valor, portanto, tem influência direta no número de sessões que são tratadas pelo servidor.

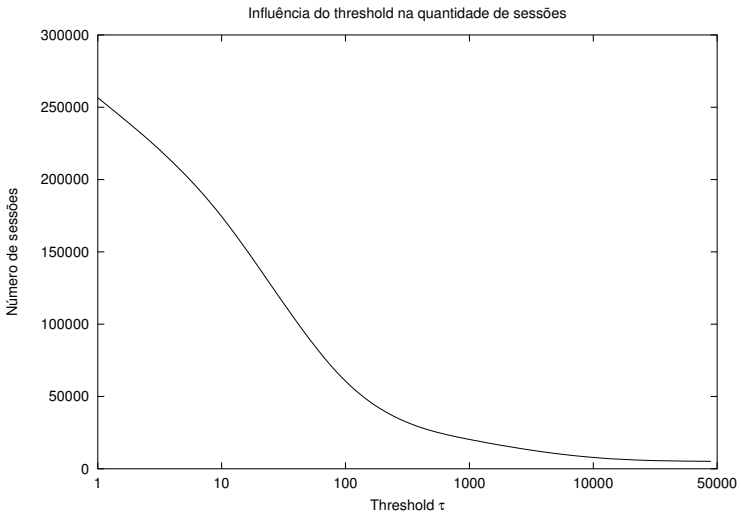


Figura 4.4: Influência do *threshold* τ no número total de sessões do *log*

A Figura 4.4 mostra o efeito do valor de τ no número total de sessões existentes no servidor *proxy-cache Squid* da UFMG. Pode-se observar que, quando o τ aumenta de 1 para 100 segundos, o número de sessões iniciadas decresce rapidamente. A partir de 1.000 segundos, o decaimento passa a ser muito menor. Isso é uma indicação de que a maioria das sessões possuem períodos de inatividade menores que 1.000 segundos. Com isso, decidiu-se utilizar o valor mais comumente usado para o τ, que é de 1.800 segundos, ou 30 minutos.

Definido o *threshold* a caracterização do nível de *Sessão* passa para a análise numérica das sessões de usuário presentes no *log* de acesso. A Tabela 4.5 mostra dados interessantes sobre essa análise.

Threshold τ	1.800s
Quantidade de sessões	14.744
Tamanho médio das sessões	15 ações
Duração média das sesões	1.212s
Média de sessões por usuário	28
Endereços IP únicos	523

Tabela 4.5: Informação quantitativa do *log* do servidor *proxy-cache* no nível de *Sessão*

Considerando somente as requisições aos 75.721 objetos únicos HTML e usando um

threshold τ de 1.800 segundos para a duração da sessão, foram identificadas 14.744 sessões de usuário associadas a 523 endereços IP únicos, resultando em um número médio de 28 sessões por usuário. O tamanho médio de uma sessão para o *threshold* escolhido é de 15 ações, sendo que a menor sessão possui 1 ação, e a maior possui 1105 ações. A duração média de uma sessão foi de 1212 segundos. A menor e a maior duração de uma sessão foram 0 segundos - o usuário que faz uma única requisição - e 15,3 horas, respectivamente.

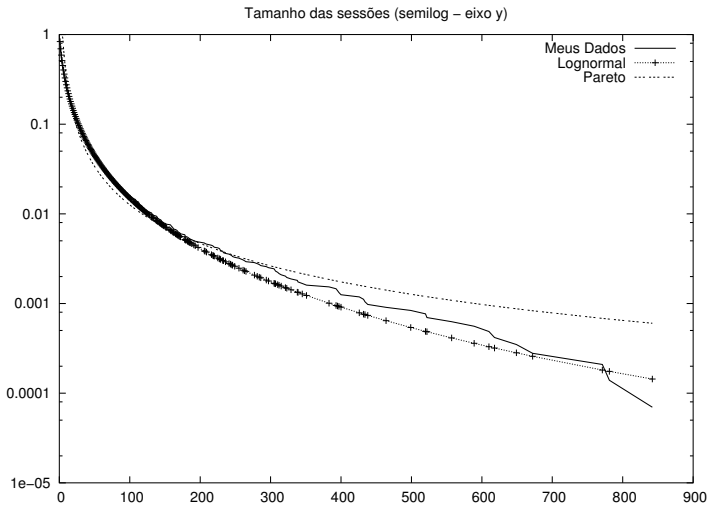


Figura 4.5: Distribuição do tamanho das sessões

Além dos dados apresentados na Tabela 4.5, foi analisada a distribuição do tamanho da sessão (vide Figura 4.5), em função da quantidade de ações. Comparando-se a distribuição observada com distribuições tradicionais de cauda pesada, como *lognormal* e Pareto, nota-se a curva está entre as duas aproximando-se mais de *lognormal*. Esse comportamento de cauda pesada existe, provavelmente, pela utilização de vários usuários em um mesmo endereço IP, ocasionando sessões muito grandes. Entretanto, esse não é o comportamento comum, uma vez que foi observado que a maioria das sessões (quase 90%) possui no máximo 26 ações.

Por ser considerada irrelevante para a modelagem do comportamento dos usuários, não foi realizada a análise de início de sessões em diferentes escalas de tempo.

Por fim é importante ressaltar que a significativa variação na distribuição das requisições pelas sessões e o tamanho médio de sessão observado (em torno de 15 ações), mostram a viabilidade da utilização do *log* do servidor *proxy-cache* para esse trabalho, uma vez que sessões pequenas não provêem informação suficiente para se modelar o comportamento dos usuários.

4.2.4 Caracterização do Nível de *Usuário*

Essa seção descreve a caracterização do nível de *Usuário* realizada para o estudo de caso de acordo a metodologia proposta. O *log* temporário gerado na seção 4.2.3 agrupando as sessões de cada usuário será utilizado também nesse nível, onde serão modeladas as características de comportamento dos usuários que submeteram requisições ao servidor *proxy-cache Squid* da UFMG.

Análise de *IAT* e Latência das Ações dos Usuários

Como citado na seção 3.2, primeiramente é realizado um estudo dos dados pelas seguintes perspectivas:*IATs* entre requisições consecutivas, latência associada às requisições dos usuários, e correlação entre *IAT* e latência.

Foram geradas as funções de distribuição cumulativa e de probabilidade (CDF e PDF) da razão entre *IAT* e latência. E foi feito o mesmo usando a diferença dessas duas medidas. De acordo com a metodologia essas funções podem ser usadas para se discretizar o conjunto de valores em classes, mas nesse estudo de caso esta técnica não mostrou um bom resultado, não permitindo a distinção de classes bem definidas. Com isso, foi decidido utilizar uma outra técnica de discretização provida pela metodologia e descrita a seguir.

As medidas de *IAT* e latência foram discretizadas usando funções que as correlacionam, mais especificamente as métricas de razão (RAZ) e diferença (DIF). Essas métricas foram definidas como:

$$DIF(k) = I(k, k+1) - L(k),\ \forall\ k \in Lu;$$

$$RAZ(k) = \begin{cases} I(k,k+1)/L(k) & , \ DIF(k) > 0 \\ L(k)/I(k,k+1) & , \ DIF(k) < 0 \\ 1 & , \ DIF(k) = 0 \end{cases} ;$$

onde k é uma requisição do usuário, $I(k,k+1)$ é o IAT entre a requisição k e $k+1$, e $L(k)$ é a latência associada à requisição k.

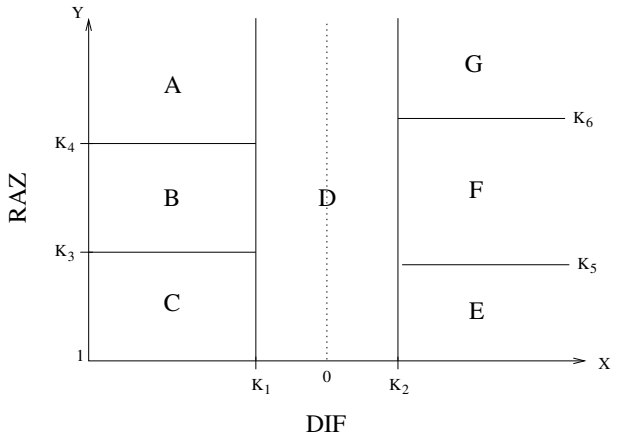


Figura 4.6: Modelo de discretização

A Figura 4.6 descreve o modelo de discretização baseado nas duas funções. O eixo x está ssociado à função DIF e o eixo y à função RAZ. O modelo define sete classes de ação de usuário (A a G), usando dois valores limite para cada eixo. As constantes k_1 e k_2 dividem x em lados positivo e negativo de acordo com a função DIF, definindo uma zona próxima a zero, onde não se pode dizer muito a respeito do comportamento do usuário. Essa zona compreende valores de IAT e latência muito próximos entre si, e pode representar situações como: usuários que requisitam um objeto e já pedem um outro pouco antes do primeiro chegar; e usuários que requisitam um objeto e não analisam a resposta recebida, pois pedem outro objeto imediatamente após a chegada da resposta à primeira requisição. Como mostrado na Figura 4.6, foi definido um intervalo D entre k_1 e k_2, que compreende todos os valores do eixo y, motivado pelo fato que os valores da função RAZ dentro desse intervalo não têm influência significativa.

Continuando a análise do modelo de discretização, as constantes k_3 e k_4 dividem o eixo y em três zonas distintas, de acordo com a função RAZ que quantifica a correlação entre

IAT e latência. Essas constantes definem três classes de ação - A, B e C - para valores de DIF menores que k_1. Essas classes representam comportamentos onde os usuários não esperam pela resposta às suas requisições e pedem um outro objeto. A mesma estratégia é aplicada às ações que possuem valores de DIF maiores que k_2. Dessa forma, são definidas três classes de ação - E, F e G - limitadas por outras duas constantes: k_5 e k_6. Essas classes que representam comportamentos onde os usuários esperam pela resposta às suas requisições antes de requisitarem outros objetos.

Para exemplificar os comportamentos expressados por cada classe podemos citar a classe E, que representa ações onde o usuário requisita um novo objeto um pequeno período de tempo após receber o objeto previamente requisitado. Por outro lado, a classe C representa usuários que não receberam o objeto dentro de tempo previsto e requisitaram outro, mas esperaram um tempo significativo, considerando-se a latência do objeto.

Com o intuito de definir quais valores são ideais para cada constante, foi esboçado um histograma de pontos e foram utilizados os valores que permitiram uma boa distribuição das ações pelas diversas classes. Com isso tem-se $k_1 = -0,1$ e $k_2 = 1$. Os valores de k_3 e k_4 são 2 e 4, respectivamente, para a discretização das classes A, B e C. E para se delimitar as classes E, F e G foram escolhidos os valores 4 para a constante k_5 e 8 para k_6.

Seguindo a metodologia e utilizando a estratégia de discretização, as sessões de usuário foram transformadas em sequências de classes de ação de usuário. Esta transformação consiste em um mapeamento direto um a um da aplicação das funções RAZ e DIF para cada requisição da sessão, expressando uma classe de ação de usuário. Como resultado, foi definido, para cada ação, um par $u(DIF((k), RAZ(k))$ da requisição do usuário, onde k é a requisição corrente na sessão. Este par corresponde a um ponto no modelo de discretização, definindo a classe para cada uma das ações. A Tabela 4.6 apresenta a frequência de ocorrência das classes de ação.

	Classes de Ação de Usuário						
	A	**B**	**C**	**D**	**E**	**F**	**G**
Frequência (# de ocorrências)	2509	2362	3933	6108	13487	14398	117788
Porcentagem (% das ocorrências)	1.56	1.47	2.45	3.80	8.40	8.97	73.35

Tabela 4.6: Distribuição das ações de usuário em classes

Análise do Padrão de Acesso dos Usuários

Para modelar o padrão de acesso dos usuários, em função das ações executadas, será utilizado o CBMG (*Customer Behavior Model Graph*) [Menascé et al., 2000], um grafo de transição de estados que é usado para descrever o comportamento de um grupo de usuários que exibe padrões de navegação similares. Esse grafo possui um nó para cada estado possível (representado pelas classes de ação dos usuários, definidas pela discretização) e as transições entre eles. A probabilidade de mudar de estado é atribuída a cada transição. Tipos de usuários diferentes podem ser caracterizados por diferentes CBMGs em termos de probabilidade de transições. Do CBMG pode-se derivar, por exemplo, o número médio de visitas a cada estado e a semântica do padrão de acesso.

No caso do servidor *proxy-cache Squid* da UFMG, as sessões de cada usuário, formadas por sequências de classes de ação (A - G), foram transformadas em CBMGs e foram agrupadas. Para isso foi utilizada uma ferramenta construída em *Java* para mineração de dados [Garner, 1995], que implementa regressão, regras de associação e técnicas de clusterização. Mais especificamente, foram utilizados os algoritmos *K-Means* (KM) e *Expectation Maximization* (EM) [Kearns et al., 1997] para fazer a clusterização.

Por fim, foram utilizados os resultados obtidos com o algoritmo *Expectation Maximization*, que mostrou melhores conclusões relacionadas ao padrão de acesso dos usuários. Para um número de grupos (*clusters*) igual a 6, se destacam os *clusters* 1, que agrupa 54% das sessões, 3 e 4, com 13% cada um. A Tabela 4.7 apresenta a distribuição das sessões de usuário pelos grupos.

A seguir serão apresentados e discutidos os dados relativos aos CBMGs dos três *clusters*

	Cluster **ID**					
	1	**2**	**3**	**4**	**5**	**6**
Ocorrência de sessões	54%	03%	13%	13%	11%	07%

Tabela 4.7: Distribuição das sessões de usuário em *clusters* (CBMG)

com maior concentração de sessões de usuários:

- *Cluster* **1**: a tabela 4.8 mostra o CBMG relativo ao *cluster* 1.

	Classe de ação								
	A	B	C	D	E	F	G	Fim	**Total**
A	0,000	0,000	0,000	0,000	0,000	0,000	0,004	0,000	0,004
B	0,000	0,000	0,000	0,000	0,000	0,000	0,004	0,000	0,004
C	0,000	0,000	0,000	0,000	0,000	0,000	0,005	0,000	0,005
D	0,000	0,000	0,000	0,003	0,000	0,001	0,039	0,003	0,046
E	0,000	0,000	0,000	0,000	0,001	0,002	0,015	0,001	0,019
F	0,000	0,000	0,000	0,000	0,002	0,003	0,026	0,005	0,036
G	0,004	0,004	0,005	0,025	0,016	0,028	0,369	0,212	0,663
Início	0,000	0,000	0,000	0,017	0,000	0,004	0,200	0,000	0,221
Total	0,004	0,004	0,005	0,045	0,019	0,038	0,662	0,221	

Tabela 4.8: Matriz do CBMG relativo ao padrão de acesso do *cluster* 1

A análise do CBMG relativo ao *cluster* 1 mostra uma tendência muito forte à permanência no estado representado pela ação G (transição $G \rightarrow G$). Visto pelo lado do servidor, esses usuários possuem uma tendência paciente, pois esperam um tempo razoável, após o recebimento da resposta, antes de fazerem uma nova requisição.

Esse comportamento pode ser explicado pela origem do *log* caracterizado. Além de ser o *log* de uma universidade que possui acesso rápido à rede, o fator principal é que o comportamento comum aos usuários que acessam a *Internet* através da universidade é o de pesquisadores, que buscam documentos com um propósito definido e analisam os objetos recebidos antes de fazerem novas requisições.

- *Cluster* **3**: a tabela 4.9 mostra o CBMG relativo ao *cluster* 3.

	Classe de ação								
	A	B	C	D	E	F	G	Fim	**Total**
A	0,001	0,000	0,000	0,001	0,001	0,000	0,007	0,001	0,011
B	0,000	0,000	0,000	0,000	0,001	0,000	0,006	0,000	0,007
C	0,000	0,000	0,000	0,006	0,000	0,002	0,002	0,003	0,013
D	0,003	0,002	0,003	0,031	0,006	0,008	0,054	0,114	0,221
E	0,001	0,001	0,001	0,009	0,005	0,003	0,039	0,007	0,066
F	0,001	0,001	0,001	0,011	0,001	0,001	0,045	0,002	0,063
G	0,005	0,004	0,007	0,069	0,027	0,025	0,238	0,059	0,434
Início	0,001	0,000	0,001	0,095	0,024	0,025	0,040	0,000	0,186
Total	0,012	0,008	0,013	0,222	0,065	0,064	0,431	0,186	

Tabela 4.9: Matriz do CBMG relativo ao padrão de acesso do *cluster* 3

No CBMG relativo ao *cluster* 3 nota-se uma distribuição um pouco maior das transições, sendo destacadas as participações dos estados representados pelas ações G e D. A transição que prevaleceu sobre as outras foi a permanência no estado G (transição $G \rightarrow G$). Visto pelo lado do servidor, esses usuários possuem uma tendência paciente, mas mostrando certa variação no seu padrão de acesso, representada pela ocorrência um pouco marcante de transições para o estado D.

Esse *cluster* agrupa sessões de usuário que apresentam comportamento com tendência a paciente, mas que em determinadas situações executam ações sem analisar o objeto recebido como resposta à requisição anterior.

- *Cluster* **4**: a tabela 4.10 mostra o CBMG relativo ao *cluster* 4.

  O CBMG relativo ao *cluster* 4 mostra a distribuição mais homogênea das transições dentre todos os estados, podendo ser observada uma pequena prevalência para as transições que envolvem os estados representados pelas ações G e D. Com isso, nenhuma característica marcante no padrão de acesso dos usuários pode ser identificada. A análise desse *cluster* revela um outro aspecto interessante que é a presença de transições entre todos os estados, mostrando a variação das reações dos usuários às diversas latências observadas.

	Classe de ação								
	A	B	C	D	E	F	G	Fim	**Total**
A	0,006	0,004	0,005	0,012	0,005	0,006	0,027	0,025	0,090
B	0,004	0,004	0,004	0,011	0,006	0,005	0,023	0,005	0,062
C	0,004	0,004	0,007	0,006	0,006	0,006	0,021	0,007	0,061
D	0,008	0,009	0,006	0,024	0,005	0,008	0,052	0,008	0,120
E	0,009	0,006	0,006	0,006	0,009	0,006	0,024	0,007	0,073
F	0,008	0,006	0,008	0,007	0,008	0,007	0,029	0,007	0,080
G	0,028	0,019	0,019	0,043	0,029	0,030	0,202	0,042	0,412
Início	0,022	0,011	0,006	0,011	0,006	0,011	0,034	0,000	0,101
Total	0,089	0,063	0,061	0,120	0,074	0,079	0,412	0,101	

Tabela 4.10: Matriz do CBMG relativo ao padrão de acesso do *cluster* 4

Analisando-se de maneira geral, a discretização das ações dos usuários, baseada na relação entre IAT e latência, e o seu agrupamento em classes de ação permitiram a análise dos padrões de acesso dos usuários, com a utilização do CBMG, e a identificação de algumas características marcantes do comportamento dos usuários. Por exemplo, foi possível perceber que a maioria das ações realizadas mostraram uma certa tendência paciente dos usuários, visto que foram executadas após o recebimento da resposta à ação anterior.

Identificação de Tendências de Comportamento dos Usuários

A análise dos padrões de navegação mostrou algumas tendências de comportamento dos usuários. Entretanto a caracterização do comportamento requer uma análise mais profunda da influência que o tempo de resposta do servidor tem sobre as futuras ações dos usuários. Com isso, decidiu-se criar uma escala de paciência (vide Figura 4.7) e determinar tendências mais bem definidas de perfil de acesso dos diversos usuários, como será mostrado a seguir.

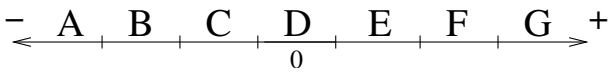


Figura 4.7: Tendências de comportamento de usuário - escala de paciência

Considerando a escala apresentada na Figura 4.7, foram definidos seis tendências de comportamento de usuário, também chamadas de perfis de usuário, cada uma represen-

tando uma direção do comportamento do usuário na escala de paciência, com as seguintes características:

Impaciente: A sequência de ações está no lado negativo da escala, incluindo zero (e.g., $C \rightarrow A \rightarrow B$, ou $A \rightarrow C \rightarrow B$, ou $C \rightarrow D \rightarrow B$, ou $C \rightarrow C \rightarrow C$, ou $A \rightarrow A \rightarrow A$). Representa uma variação no comportamento do usuário, mantendo a tendência à impaciência.

Paciente: A sequência de ações está no lado positivo da escala, incluindo zero (e.g., $E \rightarrow G \rightarrow F$, ou $G \rightarrow E \rightarrow F$, ou $F \rightarrow D \rightarrow G$, ou $E \rightarrow E \rightarrow E$, ou $G \rightarrow G \rightarrow G$). Representa uma variação no comportamento do usuário, mantendo a tendência à paciência.

Contínuo: A sequência de ações mostra pequena variação, ficando numa classe fixa em zero (e.g., $D \rightarrow D \rightarrow D$). Representa uma tendência fixa, situações onde a latência do objeto requisitado e o IAT são muito próximos. Este é um comportamento típico de um *software* robô ou de usuários cuja tendência de interação não está muito bem definida.

Tendente a Impaciente: A sequência de ações representa uma tendência à impaciência (e.g., $G \rightarrow D \rightarrow A$, ou $C \rightarrow B \rightarrow A$, ou $G \rightarrow F \rightarrow E$, ou $G \rightarrow A \rightarrow C$, ou $E \rightarrow B \rightarrow C$, ou $F \rightarrow G \rightarrow A$, ou $E \rightarrow G \rightarrow B$).

Tendente a Paciente: A sequência de ações representa tendências pacientes, onde normalmente há uma movimentação para a direita na escala (e.g., $A \rightarrow D \rightarrow G$, ou $A \rightarrow B \rightarrow C$, ou $E \rightarrow F \rightarrow G$, ou $B \rightarrow G \rightarrow F$, ou $C \rightarrow F \rightarrow E$, ou $B \rightarrow A \rightarrow E$, ou $C \rightarrow B \rightarrow F$).

Inconstante: Sequência de ações que movem do lado negativo (positivo) para o lado positivo (negativo) e retornam para o lado negativo (positivo) ou zero - e sequências que movem do lado negativo (positivo) ou zero para o lado positivo (negativo) e

retornam para o lado negativo (positivo). (e.g., $C \rightarrow E \rightarrow B$, ou $A \rightarrow G \rightarrow D$, ou $F \rightarrow C \rightarrow D$, ou $G \rightarrow B \rightarrow E$, ou $D \rightarrow G \rightarrow C$, ou $D \rightarrow B \rightarrow G$).

Finalizando o processo de discretização e definição das classes de ação de usuário, o *log* de sessões de usuário é traduzido para uma sequência de perfis (tendências de comportamento) de usuário. A nova representação das sessões consiste de uma sequência de ações, onde cada tendência representa uma variação na escala de paciência.

Definição dos Tipos de Comportamento dos Usuários

Após o mapeamento das tendências de comportamento dentro das sessões de usuário, o próximo passo definido pela metodologia é a análise dessas tendências e definição dos diversos tipos de comportamento dos usuários. Para tal, é realizado o agrupamento das sessões de modo a identificar sessões similares. Foram utilizados os algoritmos *K-Means* (KM) e *Expectation Maximization* (EM) [Kearns et al., 1997] para a clusterização.

Por fim, foram utilizados os resultados obtidos com o algoritmo *K-Means*, que mostrou conclusões interessantes relacionadas ao comportamento do usuário. Adotando dentre os clusters a soma dos erros quadrados foi identificado 7 como a melhor configuração para o número de clusters. A distribuição das seções pelos 7 clusters pode ser vista na Tabela 4.11. Já a Tabela 4.12 apresenta a distribuição das ações de usuário pelos diversos perfis, para cada cluster.

	Cluster **ID**						
	1	**2**	**3**	**4**	**5**	**6**	**7**
Quantidade de Sessões	50%	17%	3%	4%	11%	11%	4%

Tabela 4.11: Distribuição das sessões de usuário nos *clusters*

Analisando a Tabela 4.11 nota-se que o cluster mais popular é o de número 1, que agrupa metade das sessões dos usuários. O cluster de número 2 tem 17%, seguido pelos clusters 5 e 6, ambos com 11% das sessões. Os 11% restantes são divididos pelos clusters de número 4 (4%), 7 (4%) e 3 (3%).

Cluster ID	Perfil de Usuário (%)					
	Impaciente	**Paciente**	**Contínuo**	**Tend Imp**	**Tend Pac**	**Inconst**
1	0.01	99.34	0.01	0.21	0.37	0.06
2	0.57	67.69	0.86	14.92	10.14	5.81
3	31.41	0.48	33.77	0.88	6.17	27.28
4	0.2	0.6	0.0	97.63	1.48	0.09
5	1.28	36.28	0.56	41.65	13.86	6.37
6	0.27	50.04	0.14	6.96	40.16	3.44
7	0.0	0.03	0.13	1.22	98.11	0.24

Tabela 4.12: *Clusters* - distribuição das ações de usuário nos perfis

Analisando-se os clusters em relação à distribuição das ações dos usuários pelos perfis, eles podem ser descritos como:

Cluster 1 : Praticamente todos os usuários agrupados nesse cluster apresentam perfil *Paciente* durante as suas sessões.

Cluster 2 : O cluster 2 apresenta ocorrência significativa de dois perfis (*Tendente a Paciente* e *Tendente a Impaciente*), um pouco de *Inconstante*, e, na maioria das sessões, os usuários mostraram perfil *Paciente*.

Cluster 3 : O cluster 3 representa as sessões que possuem ocorrência balanceada de três perfis (*Impaciente*, *Contínuo* e *Inconstante*).

Cluster 4 : Quase todos os usuários agrupados nesse cluster apresentam perfil *Tendente a Impaciente* durante as suas sessões.

Cluster 5 : Os perfis *Paciente* e *Tendente a Impaciente* foram identificados nesse cluster como sendo os mais significativos. Entretanto, há ainda uma quantidade razoável de usuários com perfil *Tendente a Paciente* nesse grupo.

Cluster 6 : Os perfis *Paciente* e *Tendente a Paciente* foram identificados nesse cluster. Este é um grupo de usuários tipicamente pacientes, que mantém uma forte tendência à paciência com alguma variação.

Cluster 7 : Praticamente todos os usuários agrupados nesse cluster apresentam perfil *Tendente a Paciente* durante as suas sessões.

Ainda a partir da análise dos 7 clusters pode-se observar grande predominância do perfil *Paciente*. Entretanto a ocorrência dos perfis *Tendente a Paciente* e *Tendente a Impaciente* também é bastante significativa. Já os perfis *Impaciente*, *Contínuo* e *Inconstante* são significativos apenas no cluster com a menor popularidade.

Com isso, a modelagem do comportamento dos usuários está completa e agora será realizada a sua simulação, como descrito na próxima seção.

4.3 *USAR* - Simulação de Comportamento de Usuário

Nesta seção é apresentada a geração das classes de ação de usuário, como uma estratégia para validar a metodologia de caracterização e simular as características comportamentais dos usuários do servidor *proxy-cache* da Universidade Federal de Minas Gerais (UFMG).

Analisando os resultados obtidos com a caracterização da carga de trabalho (Seção 4.2), foram definidos os dados de entrada listados na Seção 3.3 como sendo:

1. Foram caracterizados 7 tipos de comportamentos de usuário, representados pelos clusters, e 6 tendências de comportamento, os perfis *Impaciente*, *Paciente*, *Contínuo*, *Tendente a Paciente*, *Tendente a Impaciente* e *Inconstante*.

2. A distribuição de probabilidade aplicada aos tipos de comportamentos de usuário usa os valores obtidos com a clusterização (vide Tabela 4.11) para determinar o percentual de ocorrência de cada tipo.

3. A distribuição de probabilidade associada aos perfis de usuário utiliza os valores listados na Tabela 4.12 para determinar, para cada tipo de comportamento, o percentual de ocorrência das ações de usuário relativas aos perfis.

4. A distribuição de tamanho de sessão, de acordo com o número de requisições (cada uma das ações de usuário) por sessão, segue a distribuição *lognormal* (vide Figura 4.5)

com $\sigma = 1,463065$ e $\zeta = 1,430258$. O quadrado mínimo calculado para esta função foi $0,022657$.

5. A distribuição de probabilidade das classes de ação de usuário em sequências usa o percentual de ocorrência de cada sequência no *log* real - considerando-se sequências de tamanho 1, 2 e 3 - para determinar pesos para a futura geração dessas classes durante a simulação. Ainda nesta seção será explicado como as sequências são compostas.

6. A caracterização mostrou ainda que a popularidade de objetos segue a distribuição de *Pareto* com $\alpha = 0,980308$ e $k = 0,158868$. Para garantir a qualidade da compatibilidade com essa distribuição, foi calculado o quadrado mínimo e obtido o valor de $0,000653$ para *Pareto*.

7. Por fim, observa-se que o tamanho dos objetos está diretamente relacionado com a latência observada no servidor e não segue nenhuma distribuição probabilística conhecida (vide Figura 4.2).

O processo de caracterização definiu sete classes de ação de usuário que correlacionam IAT e latência. Para mapear essas classes para os seis perfis de usuário, foi decidido agrupar as classes de ação em sequências. Combinando as sete classes tem-se 343 possíveis sequências de tamanho três, 49 de tamanho dois e 7 formadas por uma única classe (sequências de tamanho dois e um são usadas como a última sequência caso a sessão a ser gerada possua tamanho não múltiplo de 3). Cada sequência possui um peso, calculado de acordo com o seu valor absoluto de ocorrência no *log* real e a porcentagem de sessões onde ela ocorre. Por fim, as sequências são mapeadas para um perfil de usuário seguindo uma tendência, como apresentado na seção 4.2.

Utilizando estes dados e considerando o número de sessões informado como parâmetro, o processo de geração para cada sessão segue os passos listados a seguir:

- De acordo com a distribuição dos tipos de comportamentos de usuário (nesse estudo de caso uma distribuição probabilística) informada como dado de entrada, é escolhido um comportamento para a sessão.

- É calculado o tamanho da sessão (como visto anteriormente, para o estudo de caso aqui apresentado segue uma distribuição *lognormal*). Sabendo-se o tamanho da sessão (em quantidade de requisições), calcula-se também o número de classes de ação e o número de sequências de classes que comporão a sessão.

- Por fim, as sequências de classes de ação de usuário são geradas seguindo a distribuição de perfis de usuário (também uma distribuição probabilística, nesse estudo de caso) que compõem o comportamento previamente escolhido (vide Tabela 4.12). É importante ressaltar que a frequência de ocorrência das sequências no *log* sintético é dado pelo peso calculado para cada uma das sequências.

Ao final do processo de geração de ações e simulação do comportamento dos usuários, tem-se como dado de saída um *log* sintético composto de ações de usuário. Essas ações são agrupadas em sessões de usuário e simulam de forma precisa o comportamento real modelado durante a caracterização, pois consideram a qualidade do serviço provida pelo sistema para simular a interação de usuários com o servidor *proxy-cache*.

Deve-se ressaltar que a geração de um *log* sintético que simule o comportamento real de forma precisa é um desafio uma vez que foi criado um modelo bastante abstrato de um comportamento que é geralmente muito variável. Como será visto a seguir com a exibição dos resultados, a precisão alcançada pela metodologia é definitivamente uma das contribuições desse trabalho.

4.3.1 Resultados

O objetivo da simulação é mostrar a aplicabilidade da metodologia de caracterização para se gerar um *log* sintético que informa qual a ação do usuário em resposta à qualidade do serviço provido. Utilizando um método eficiente para a geração de variáveis aleatórias discretas com distribuições gerais [Walker, 1977], foi simulada de forma minuciosa a distribuição das sessões entre os comportamentos de usuário. Além disto, a simulação produziu resultados precisos em termos da distribuição das sequências pelos seis perfis de usuário observados em cada *cluster* (vide Tabelas 4.11 e 4.12).

Mesmo conseguindo realizar uma simulação precisa do comportamento dos usuários, foi difícil reproduzir, no *log* gerado, exatamente as mesmas sequências de ação que identificam uma tendência no perfil do usuário. Isso se deve ao fato de que a distribuição de tamanho das sessões foi calculada para o *log* como um todo, independente da *clusterização* ou de uma análise mais detalhada dos perfis. No entanto, uma análise mais fina mostra um resultado muito bom relativo à frequência de ocorrência de cada classe de ação de usuário no *log* sintético, quando comparado ao real. A Tabela 4.13 apresenta os resultados para esse estudo de caso.

	Classes de Ação de Usuário (%)						
Log	**A**	**B**	**C**	**D**	**E**	**F**	**G**
Real	1.56	1.47	2.45	3.8	8.4	8.97	73.35
Sintético	1.54	1.44	2.4	3.76	8.38	8.95	73.51

Tabela 4.13: Distribuição das ações de usuário em classes

Os valores da simulação para todas as classes de ação de usuário são muito próximos aos obtidos com a caracterização do *log* real e demonstra a viabilidade da adoção de metodologias bem definidas, baseadas no comportamento do usuário, para a caracterização de cargas de trabalho *Web* e a simulação desse comportamento, considerando-se a reação do usuário a latências variadas.

Finalmente, durante o processo de validação, foram identificadas algumas direções para a continuação e o melhoramento desse trabalho, tal como a análise da correlação entre tamanho de sessão e perfil de usuário, o que pode minimizar a dificuldade da simulação real da distribuição das sequências de ação pelos diversos perfis.

Capítulo 5

Conclusão

A constante evolução dos *web sites* proporciona o surgimento a cada dia de novas aplicações e novos tipos de negócio eletrônico, como *home banking*, leilões pela *Internet* e *e-government*. Isso atrai cada vez mais usuários para a rede. Com isso, o desempenho dessas aplicações e a qualidade dos serviços oferecidos se tornam fatores cruciais para a manutenabilidade dos usuários atuais e a atração de novos clientes.

Entender as características das cargas de trabalho impostas aos servidores é um passo importante para melhorar a qualidade do serviço oferecido ao usuário. Além disso, com esse conhecimento, pode-se estudar as variações da intensidade da carga e os efeitos causados no sistema, que são características fundamentais para a análise de desempenho e o planejamento de capacidade.

Devido à demanda por esse conhecimento muitos estudos têm sido publicados a respeito de caracterização de cargas de trabalho de servidores *Web*. Entretanto, nenhum deles faz uma análise detalhadada do comportamento dos usuários, modelando, por exemplo, as reações dos usuários a variações no tempo de resposta das requisições. Informação que pode ser muito útil para aprimorar o entendimento das cargas de trabalho *Web* e para construir geradores de carga mais realísticos.

Visando acrescentar esse importante aspecto à caracterização de cargas de trabalho *Web*, foi apresentada a metodologia *USAR*, que é uma metodologia para caracterização e

simulação do comportamento dos usuários estruturada de maneira hierárquica em quatro níveis: *Usuário*, *Sessão*, *Ação* e *Requisição*. A metodologia *USAR* se baseia no modelo proposto em [Menascé et al., 2003] e adiciona a caracterização e modelagem do comportamento dos usuários.

A fim de demonstrar a aplicabilidade da metodologia *USAR* foi apresentado um estudo de caso, utilizando o *log* do servidor *proxy-cache Squid* da UFMG. Durante a modelagem do comportamento dos usuários, foi apresentado um modelo de discretização que permitiu classificar as ações dos usuários, baseado na correlação entre IAT e latência das respostas às requisições feitas ao servidor.

A partir da classificação das ações, as sessões de usuário foram transformadas em sequências de ações e foi feita a análise do padrão de navegação dentro dessas sessões, mostrando uma forte tendência dos usuários a esperar a resposta a uma requisição antes de solicitar um novo objeto. A modelagem continuou analisando as sequências de ações e identificando tendências de comportamento, também chamadas de perfis de usuário. Essas tendências foram identificadas a partir das diversas ações realizadas pelos usuários no servidor em reação a um tempo de resposta observado e permitiram a definição de seis perfis de usuário - *Impaciente*, *Paciente*, *Contínuo*, *Tendente a Impaciente*, *Tendente a Paciente* e *Inconstante*.

Com a definição dos perfis de usuário foi possível agrupar as sessões de acordo com diferentes tipos de comportamento observados, sendo identificados 7 *clusters* distintos, representando grupos de usuários que possuem características semelhantes de interação com o sistema. Para o estudo de caso apresentado notou-se, analisando esses 7 grupos, grande predominância do perfil *Paciente*, com ocorrência significativa dos perfis *Tendente a Paciente* e *Tendente a Impaciente*.

Fazendo uma analogia entre a caracterização dos usuários de serviços *Internet* e a caracterização do tráfego de carros de uma determinada rua, pode-se fazer uma comparação interessante: os modelos de caracterização anteriores permitiam determinar que na rua passam carros de cores diferentes, a metodologia *USAR* permite determinar quais são os carros azuis, pretos ou vermelhos. Isso torna o processo de caracterização do comportamento dos

usuários mais poderoso e traz inúmeras possibilidades de melhoria de desempenho dos servidores, capacidade de planejamento dos recursos e personalização dos serviços oferecidos na *Web*, aumentando a satisfação dos clientes e o lucro dos *sites* prestadores de serviço.

Por fim, foi realizada simulação do comportamento dos usuários através da geração das reações dos usuários às variações de latência modeladas durante o processo de caracterização. Com isso foi comprovada a aplicabilidade e eficiência da metodologia de simulação *USAR* para se gerar *logs* sintéticos formados por sessões de usuário que são compostas por ações que simulam de forma precisa o comportamento dos usuários observado pelos servidores *Web*.

Como a metodologia *USAR* é uma metodologia genérica que pode ser aplicada a qualquer carga de trabalho *Web*, propõe-se como trabalho futuro a caracterização de outras cargas de trabalho que permitam estudar diferentes características do comportamento dos usuários, como por exemplo a análise da interação dos clientes de uma aplicação de comércio eletrônico baseada nas funções oferecidas por essa aplicação.

Além disso esse trabalho é a base para esforços no desenvolvimento de geradores de cargas sintéticas que se preocupam com a latência, proporcionando a geração de cargas de trabalho *Web* que consideram o impacto real da qualidade de serviço oferecida sobre as diversas ações dos usuários.

Essa inovação pode trazer um grande ganho na qualidade da carga de trabalho gerada, reduzindo a distância entre os modelos tradicionais de geração de carga, baseados em distribuições estatísticas típicas, e a carga de trabalho real.

Bibliografia

[Almeida et al., 2001a] Almeida, J. M., Krueger, J., Eager, D. L., and Vernon, M. K. (2001a). Analysis of educational media server workloads. In *Proceedings of the 11th international workshop on Network and operating systems support for digital audio and video*, pages 21–30. ACM Press.

[Almeida et al., 2001b] Almeida, J. M., Krueger, J., and Vernon, M. K. (2001b). Characterization of user access to streaming media files. In *Proceedings of the 2001 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, pages 340–341. ACM Press.

[Almeida et al., 1996] Almeida, V., Bestavros, A., Crovella, M., and de Oliveira, A. (1996). Characterizing reference locality in the WWW. In *Proceedings of the IEEE Conference on Parallel and Distributed Information Systems (PDIS)*, Miami Beach, FL.

[Arlitt, 1996] Arlitt, M. (1996). A performance study of internet web servers. Phd dissertation, Dept. Computer Science, University of Saskatchewan, Saskatoon, Saskatchewan.

[Arlitt and Williamson, 1996] Arlitt, M. F. and Williamson, C. L. (1996). Web server workload characterization: The search for invariants. In *Measurement and Modeling of Computer Systems*, pages 126–137.

[Arlitt and Williamson, 1997] Arlitt, M. F. and Williamson, C. L. (1997). Internet web servers: workload characterization and performance implications. *IEEE/ACM Transactions Networking*, 5(5):631–645.

[Balachandran et al., 2002] Balachandran, A., Voelker, G., Bahl, P., and Rangan, P. (2002). Characterizing user behavior and network performance in a public wireless lan. In *Proceedings of ACM SIGEMTRICS'02*.

[Barford and Crovella, 1998] Barford, P. and Crovella, M. (1998). Generating representative web workloads for network and server performance evaluation. In *Proceedings of the 1998 ACM SIGMETRICS joint international conference on Measurement and modeling of computer systems*, pages 151–160. ACM Press.

[Busari and Williamson, 2002] Busari, M. and Williamson, C. (2002). Prowgen: a synthetic workload generation tool for simulation evaluation of web proxy caches. *Comput. Networks*, 38(6):779–794.

[Catledge and Pitkow, 1995] Catledge, L. D. and Pitkow, J. E. (1995). Characterizing browsing strategies in the World-Wide Web. *Computer Networks and ISDN Systems*, 27(6):1065–1073.

[Chatterjee et al., 1998] Chatterjee, P., Hoffman, D., and Novak, T. (1998). Modeling the clickstream: Implications for web-based advertising efforts.

[Cherkasova and Gupta, 2002] Cherkasova, L. and Gupta, M. (2002). Characterizing locality, evolution, and life span of accesses in enterprise media server workloads. In *Proceedings of the 12th international workshop on Network and operating systems support for digital audio and video*, pages 33–42. ACM Press.

[Cherkasova and Phaal, 1998] Cherkasova, L. and Phaal, P. (1998). Session based admission control: A mechanism for improving the performance of an overloaded web server. Report HPL-98-119, Hewlett-Packard Laboratories.

[Costa et al., 2004] Costa, C., Cunha, I., Borges, A., Ramos, C., Rocha, M., Almeida, J., and Neto, B. R. (2004). Analyzing client interactivity in streaming media. In *Proceedings of the 13th World Wide Web Conference*.

[Crovella and Bestavros, 1996] Crovella, M. E. and Bestavros, A. (1996). Self-similarity in world wide web traffic: evidence and possible causes. In *Proceedings of the 1996 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, pages 160–169. ACM Press.

[Cunha et al., 1995] Cunha, C., Bestavros, A., and Crovella, M. (1995). Characteristics of World Wide Web Client-based Traces. Technical Report BUCS-TR-1995-010, Boston University, CS Dept, Boston, MA 02215.

[Dilley, 1996] Dilley, J. A. (1996). Web Server Workload Characterization. Report HPL-96-160, Hewlett-Packard Laboratories.

[Franco and Meira, Jr., 2004] Franco, G. and Meira, Jr., W. (2004). Usar: a user behavior characterization model. In *Proceedings of the PhD and MSc Workshop of the 2nd Latin American Web Congress and the 10th Brazilian Symposium on Multimedia and the Web (LA-WebMedia 2004)*, Ribeirão Preto, SP, Brazil. IEEE Computer Society.

[Garner, 1995] Garner, S. (1995). Weka: The waikato environment for knowledge analysis. In *Proceedings of the New Zealand Computer Science Research Students Conference*, pages 57–64.

[Henderson, 2001] Henderson, T. (2001). Latency and user behaviour on a multiplayer game server. In *Proceedings of the Third International COST264 Workshop on Networked Group Communication*, pages 1–13. Springer-Verlag.

[Hlavacs et al., 2000] Hlavacs, H., Hotop, E., and Kotsis, G. (2000). Workload generation by modeling user behavior. In *Proceedings of OPNETWORKS 2000*.

[Hlavacs and Kotsis, 1999] Hlavacs, H. and Kotsis, G. (1999). Modeling user behavior: A layered approach. In *MASCOTS*, pages 218–225.

[Jin and Bestavros, 2001] Jin, S. and Bestavros, A. (2001). Gismo: a generator of internet streaming media objects and workloads. *SIGMETRICS Perform. Eval. Rev.*, 29(3):2–10.

[Kearns et al., 1997] Kearns, M., Mansour, Y., and Ng, A. Y. (1997). An information-theoretic analysis of hard and soft assignment methods for clustering. In *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence*, pages 282–293, Providence, Rhode Island, USA.

[Krishnamurthy and Rolia, 1998] Krishnamurthy, D. and Rolia, J. (1998). Predicting the performance of an e-commerce server: Those mean percentiles. In *Proceedings of the 1st Workshop on Internet Server Performance, ACM SIGMETRICS*.

[Menascé et al., 2000] Menascé, D., Almeida, V., Riedi, R., Ribeiro, F., Fonseca, R., and Meira, Jr., W. (2000). In search of invariants for e-business workloads. In *Proceedings of the 2nd ACM conference on Electronic commerce*, pages 56–65. ACM Press.

[Menascé et al., 2003] Menascé, D. A., Almeida, V. A. F., Riedi, R., Ribeiro, F., Fonseca, R., and Meira, Jr., W. (2003). A hierarchical and multiscale approach to analyze e-business workloads. *Perform. Eval.*, 54(1):33–57.

[Menascé and Almeida, 2000] Menascé, D. and Almeida, V. (2000). *Scaling for E-business: Technologies, Models and Performance and Capacity Planning*. Prentice Hall.

[Menascé et al., 1999] Menascé, D., Almeida, V., Fonseca, R., and Mendes, M. (1999). A methodology for workload characterization for e-commerce servers. In *Proceedings of the 1st ACM Conference in Eletronic Commerce*, pages 119 – 128, Denver, CO.

[Menascé et al., 2000] Menascé, D. A., Almeida, V. A. F., Fonseca, R., and Mendes, M. A. (2000). Business-oriented resource management policies for e-commerce servers. *Performance Evaluation: An International Journal*, 42(2-3):223–239.

[Mosberger and Jin, 1998] Mosberger, D. and Jin, T. (1998). httperf–tool for measuring web server performance. *SIGMETRICS Perform. Eval. Rev.*, 26(3):31–37.

[Paliouras et al., 2000] Paliouras, G., Papatheodorou, C., Karkaletsis, V., and Spyropoulos, C. (2000). Clustering the users of large web sites into communities. In *Proceedings of the 4th European Conference on Research and Advanced Technology for Digital Libraries (ECDL)*, Lisboa, Portugal.

[Pereira et al., 2004a] Pereira, A., Franco, G., Silva, L., and Meira, Jr., W. (2004a). A hierarchical characterization of user behavior. In *Proceedings of the 2nd Latin American Web Congress and the 10th Brazilian Symposium on Multimedia and the Web (LA-WebMedia 2004)*, Ribeirão Preto, SP, Brazil. IEEE Computer Society.

[Pereira et al., 2004b] Pereira, A., Franco, G., Silva, L., Meira, Jr., W., and Santos, W. (2004b). The usar characterization model. In *Proceedings of the IEEE 7th Annual Workshop on Workload Characterization (WWC-7)*, Austin, Texas, USA. IEEE Computer Society.

[Pitkow, 1998] Pitkow, J. E. (1998). Summary of WWW characterizations. *Computer Networks and ISDN Systems*, 30(1–7):551–558.

[SPECweb99, 1999] SPECweb99 (1999). Specweb99 benchmark. http://www.specbench.org/osg/web99/.

[Tang et al., 2003] Tang, W., Fu, Y., Cherkasova, L., and Vahdat, A. (2003). Medisyn: a synthetic streaming media service workload generator. In *Proceedings of the 13th international workshop on Network and operating systems support for digital audio and video*, pages 12–21. ACM Press.

[Veloso et al., 2002] Veloso, E., Almeida, V., Meira, W., Bestavros, A., and Jin, S. (2002). A hierarchical characterization of a live streaming media workload. In *Proceedings of the second ACM SIGCOMM Workshop on Internet measurment*, pages 117–130. ACM Press.

[Walker, 1977] Walker, A. J. (1977). An efficient method for generating discrete random variables with general distributions. *ACM Trans. Math. Softw.*, 3(3):253–256.

[WebBench, 2002] WebBench (2002). Webbench 5.0 benchmark. http://www.veritest.com/benchmarks/webbench/.

[Zaki, 2001] Zaki, M. J. (2001). SPADE: An efficient algorithm for mining frequent sequences. *Machine Learning*, 42(1/2):31–60.

[Zipf, 1949] Zipf, G. (1949). *Human Behavior and the Principle of Least Effort*. Addison-Wesley, Cambridge, MA.