



PÓS-GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO

**Análise de Desempenho do Servidor *Web*
Apache no provedor FlorianoNet**

Por:

Francisco Eudes do Amaral

Dissertação de Mestrado



Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
[http:// ftp.cin.ufpe.br](http://ftp.cin.ufpe.br)
<ftp://ftp.cin.ufpe.br/pub/posgrad>

RECIFE, DEZEMBRO/2002



UNIVERSIDADE FEDERAL DE PERNAMBUCO-UFPE
CENTRO DE INFORMÁTICA - CIn
MESTRADO EM CIÊNCIAS DA COMPUTAÇÃO

Análise de Desempenho do Servidor *Web*

Apache no provedor FlorianoNet

Dissertação apresentada como parte dos requisitos
para obtenção do grau de Mestre em Ciência da
Computação.

Área de Concentração: Redes e Sistemas
Distribuídos

Orientadora: Prof. Marcília Andrade Campos

Co-orientador: Fernando da Fonseca de Souza

RECIFE, DEZEMBRO/2002

Agradecimentos

Agradeço em especial aos professores Marcília A. Campos e Fernando Fonseca, respectivamente, orientadora e co-orientador deste trabalho, pelo apoio na definição do mesmo e estímulo visando meu amadurecimento como aluno e pesquisador.

Aos meus pais , Enilda Teixeira e José Amaral por todas as oportunidades , e principalmente pela educação a min proporcionada.

A Ronildes, minha esposa, e a Ian, meu filho, por terem sabido suportar os longos momentos de ausência.

Aos meus irmãos, sobrinhos e amigos pelo carinho e compreensão.

Aos meus colegas de curso e principalmente a, Júnior, Edenise, Chico Demes, Constantino, Giovani, Ricardo Queiroz e Eduilson pela companhia, sugestões e principalmente pelo apoio nos momentos difíceis.

Ao proprietário do provedor FloianoNet, que gentilmente permitiu a monitoração e acesso aos dados do sistema.

À direção do CEFET-PI pela oportunidade proporcionada.

Aos demais professores do departamento de informática da UFPE que colaboraram diretamente para minha formação acadêmica no mestrado.

Ao professor e amigo Marcos Antônio, que fez um grande esforço ao assumir as turmas nos períodos em que me ausentei da UNED-Florianópolis.

Ao professor Luiz Nascimento e a todos que fazem a UNED-Florianópolis pelo apoio e incentivo.

A professora Giselda e o amigo Júlio pela força durante a estadia em Recife.

E, principalmente, a Deus sem o qual nada disso teria acontecido.

In memoriam

Ao professor Santílio Alves da Costa Filho, mestre e amigo incentivador deste trabalho.

Sumário

Agradecimentos	iii
Sumário	iv
Lista de Acrônimos.....	vii
Índice de Figuras.....	viii
Índice de Gráficos.....	ix
Índice de Símbolos.....	x
Índice de Tabelas	xi
Resumo.....	xii
Abstract	xiii
 Capítulo 1 Introdução	 14
1.1 Introdução.....	15
1.2 Objetivos.....	16
1.3 Estrutura da Dissertação	17
 Capítulo 2 Web e Banco de Dados	 18
2.1 Introdução	19
2.2 A World Wide Web.....	19
2.2.1 URL	20
2.2.2 HTML (<i>Hypertext Markup Language</i>).....	20
2.2.3 O Protocolo HTTP.....	21
2.2.4 A Arquitetura Web.....	21
2.2.5 Scripts Web e Interface CGI	22
2.2.6 SSIs e APIs de Servidores Web.....	23
2.2.6.1 SSIs	23
2.2.6.2. APIs.....	24
2.2.7 Linguagens de Programação na Web.....	24
2.2.7.1 Java	24
2.2.7.2 JavaScript	25
2.2.7.3 PERL.....	25
2.2.8 O Servidor Web Apache	26
2.3 Sistemas Gerenciadores de Banco de Dados (SGBDs).....	27
2.3.1 Arquiteturas de Sistemas de Bancos de Dados	27
2.3.1.1 Sistemas Centralizados	27
2.3.1.2 Sistemas Cliente/Servidor	28
2.3.1.3 Sistemas Paralelos	29
2.3.1.4 Sistemas Distribuídos.....	29
2.3.2 O Sistema Gerenciador de Banco de Dados MySQL.....	29
2.4 Integração Web e Banco de Dados.....	30
2.4.1 Problemas da Integração Web e Banco de Dados	32
2.4.1.1 Sistemas Legados.....	32
2.4.1.2 Problemas Transacionais.....	32
2.4.1.3 Segurança e Acesso ao Banco de Dados.....	33
2.4.1.4 Desempenho	33
2.4.1.5 Desenvolvimento e Portabilidade	34
2.4.1.6 Linguagens de Programação	34

2.5 A Linguagem XML.....	35
2.6 Conclusões.....	35
CAPÍTULO 3 Avaliação de Desempenho de Sistemas Computacionais	37
3.1 Introdução	38
3.2 Erros em Avaliação de Desempenho	38
3.3 Metodologia para Avaliação de Desempenho	39
3.4 Métricas de Avaliação de Desempenho	40
3.5 Técnicas de Avaliação de Desempenho	42
3.5.1 Monitoração/Mensuração	42
3.5.2 Modelagem Analítica	43
3.5.3 Simulação.....	44
3.6 Caracterização da Carga de Trabalho.....	44
3.7 Avaliação de Desempenho de Servidores <i>Web</i>	46
3.8 Conclusões.....	47
CAPÍTULO 4 Conceito e Aplicações de Séries Temporais	48
4.1 Introdução	49
4.2 Séries Temporais.....	49
4.3 Objetivos da Análise de Séries Temporais	51
4.4 Componentes de uma Série de Tempo	51
4.5 Métodos de Estimção da Tendência	53
4.6 Métodos de Suavizamento.....	53
4.6.1 Suavizamento Exponencial Simples.....	54
4.6.1.1 Vantagens e Desvantagens do Método de Suavizamento Exponencial Simples	55
4.7 Aplicações de Séries Temporais	56
4.7.1 Nos Negócios e na Economia	56
4.7.2 Na Avaliação de Desempenho de Sistemas Computacionais	58
4.8 Conclusões	58
CAPÍTULO 5 Avaliação de Desempenho de um Servidor <i>Web</i>	60
5.1 Introdução	61
5.2 O Provedor de Acesso à <i>Internet</i> FlorianoNet.....	62
5.3 Objetivos da Avaliação de Desempenho	63
5.4 Métricas de Interesse	63
5.4.1 Taxa de Processamento.....	63
5.4.2 Tempo de Resposta.....	64
5.5 Carga de Trabalho do Servidor <i>Web</i>	64
5.6 Instrumentação	65
5.6.1 Estratégia de Aquisição dos Perfis de Desempenho.....	66
5.7 Perfil de Desempenho do Servidor WWW Apache.....	67
5.7.1 Taxa de Processamento.....	68
5.7.2 Tempo de Resposta.....	69
5.8 Acesso a <i>Sites</i> e Previsão de Desempenho do Servidor <i>Web</i>	70
5.8.1 Indicadores Estatísticos de Acesso.....	71
5.8.2 Previsão de Desempenho	73

5.9 Análise de Resultados	77
5.10 Conclusões	78
CAPÍTULO 6 Conclusão	80
6.1 Considerações Finais.....	81
6.2 Contribuições	81
6.3 Trabalhos Futuros.....	81
Referências Bibliográficas	82
Anexos	87
Anexo 1 – Código fonte contador Acessos	88
Anexo 2 – Medidas estatísticas de acesso ao servidor Apache	90
Anexo 3 – Dados originais das requisições HTTP submetidas ao servidor <i>Web</i> Apache ...	91
Anexo 4 – Dados originais e ajustados para a taxa diária de processamento do servidor <i>Web</i> Apache em <i>Mbytes</i> por dia.	93

Lista de Acrônimos

WWW	<i>World Wide Web</i>
SGBD	Sistema Gerenciador de Banco de Dados
HTML	<i>Hypertext Markup Language</i>
URL	<i>Uniform Resource Location</i>
SGML	<i>Standard Generalized Markup Language</i>
CGI	<i>Common Gateway Interface</i>
SSI	<i>Server Side Includes</i>
API	<i>Application Programing Interface</i>
HTTP	<i>Hypertext Transfer Protocol</i>
XML	<i>Extensible Markup Language</i>
AES	Alisamento Exponencial Simples

Índice de Figuras

Figura 1 : Arquitetura <i>Web</i> Simplificada	22
Figura 2: Sistema de banco de dados centralizado	28
Figura 3: Sistema de banco de dados cliente/servidor.....	29
Figura 4: Arquitetura de integração <i>Web</i> e banco de dados.....	31

Índice de Gráficos

Gráfico 1: Requisições diárias ao <i>site</i> Jornal Local do Provedor FlorianoNet	50
Gráfico 2: Vendas das fábricas da GM (em milhões de unidades)	57
Gráfico 3: Medidas do servidor: taxa de processamento em requisições/dia	68
Gráfico 4: Medidas do servidor: taxa de processamento em Mbytes/dia	69
Gráfico 5: Dados originais (Acessos) e valores ajustados (Acessos Ajustados).....	73
Gráfico 6 : Dados originais (Requisições) e ajustados (Requisições Ajustadas)	75
Gráfico 7: Dados originais (Tráfego) e com valores ajustados (Tráfego Ajustado).....	76

Índice de Símbolos

Y_i	Valor da série temporal no instante i
T_i	Componente tendência de uma série temporal
S_i	Componente sazonal de uma série temporal
a_i	Termo aleatório de uma série de tempo
μ_i	Parâmetro desconhecido de uma série temporal
$\bar{Y}_i$	Valor de uma série temporal exponencialmente suavizada
$\bar{Y}_{i-1}$	Valor de uma série temporal exponencialmente suavizada(AES), calculada no período de tempo I
k	Constante de suavizamento de um AES
$\hat{Y}_i$	Indica a previsão de todos os valores futuros de uma série temporal no suavizamento exponencial simples
Var	Indica a variância

Índice de Tabelas

Tabela 1: Parâmetros da carga de trabalho.....	45
Tabela 2: Distribuição do número diário de visitantes únicos ao <i>site Jornal Local</i>	50
Tabela 3: Vendas das fábricas da GM (em milhões de unidades).....	57
Tabela 4: Caracterização da carga de trabalho do servidor Apache.....	65
Tabela 5: Número e percentual classificados segundo à extensão	67
Tabela 6: Perfil de desempenho do servidor WWW Apache.....	70
Tabela 7: Sumário estatístico de dados de acesso ao servidor Apache.....	71
Tabela 8: Distribuição do número diário de acessos a <i>sites</i> do servidor Apache.....	72
Tabela 9: Valores de K e S utilizando o AES (<i>site GurguéiNet</i>).....	73
Tabela 10 Valores de S utilizando AES (Requisições ao servidor Apache)	74
Tabela 11 Valores de S utilizando AES (Tráfego no servidor Apache)	77
Tabela 12: Estatístico de acesso a <i>sites</i> do servidor Apache,dados da Tabela 8	90
Tabela 13: Estatístico de requisições ao servidor Apache	90
Tabela 14 : Estatístico do tráfego processado pelo servidor Apache (<i>Mbytes</i>).....	90
Tabela 15 : AES aplicado a valores de acesso ao site GurguéiaNet, dados Tabela 8	90
Tabela 16 : AES aplicado a 208 valores da série de requisições, $K=0,99$	91
Tabela 17 : AES aplicado a 208 valores da série de tráfego, $K=0,99$	93

Resumo

Este trabalho descreve os conceitos básicos do ambiente que integra a *Web* e Sistemas de Banco de Dados no que se refere especificamente a construção de aplicações que permitam acessar banco de dados através da *Web*. É feita uma abordagem sobre a avaliação de desempenho de sistemas computacionais, em especial, dos sistemas de banco de dados e servidores *WWW*. São abordados também conceitos e aplicações de séries temporais, com ênfase nos métodos de alisamento exponencial para previsão de uma série de tempo. Por fim, é realizada uma análise de desempenho de um servidor *WWW* Apache, onde apresenta-se um perfil de execução do sistema, ao mesmo tempo em que faz-se um prognóstico de seu desempenho para um período no futuro, através da previsão das séries formadas pelo número de requisições *HTTP* atendidas diariamente e pela quantidade de *bytes* transmitidos a cada dia.

Palavras Chaves: *Web*, Servidor *WWW*, Banco de Dados, Análise de Desempenho, Séries Temporais.

Abstract

This dissertation describes basic concepts of the environment that integrates the Web and Database Systems concerning with the construction of applications specifically allowing to access database through the Web. It also focuses on computer systems performance evaluation, especially database systems and Web server. Furthermore, this work deals with time series applications, emphasizing methods for forecasting with time series. Finally, it is performed a analysis of a Apache server, showing a profile of how the system is working.

Key words: Web, Server Web, Database, Analyses of Performance, Time Series.

Capítulo 1

Introdução

1.1 Introdução

A *World Wide Web* ou apenas *Web* como é mais conhecida, tem se tornado ultimamente o local de preferência universal para a publicação de dados eletrônicos, sejam eles de caráter técnico, comercial ou científico [ABI99]. Contudo, esse aumento cada vez maior no número de usuários traz várias conseqüências. Como descrito em [MAR99], algumas dessas conseqüências são: perda de velocidade na localização da informação, falta de segurança e diversidade no tipo de plataforma utilizada para a disponibilização dos dados na rede.

Paralelamente à popularização da *Web*, os Sistemas de Gerenciamento de Banco de Dados (SGBDs) apresentaram uma evolução nos últimos anos a ponto de se constituírem em componentes centrais de modernos ambientes de computação. A tecnologia de banco de dados, entre outras questões importantes, dá ênfase à utilização de técnicas de armazenamento, linguagens de consultas e mecanismos para a manutenção de integridade e consistência de dados, as quais, segundo Silberchatz [SIL99] são imprescindíveis em um sistema de Banco de Dados.

A medida que a *Web* foi crescendo foram surgindo novas aplicações específicas para negócios via *Internet*, como comércio eletrônico, por exemplo. Estas aplicações segundo [MEN00], consistem essencialmente em transações, baseadas em dados e informações interativas, ligadas em banco de dados corporativos. Atividades como atendimento ao consumidor, cadastramento de clientes e localização de produtos são exemplos das diversas atividades comerciais que vêm sendo executadas por servidores de comércio eletrônico.

Essa expansão da *Web* trouxe como conseqüência um aumento no volume de dados em seus nós, de forma que se tem buscado soluções em banco de dados para gerenciar melhor esta quantidade crescente de informações. De fato, a tecnologia de banco de dados como aponta [ABI00, ADR98] se apresenta como a melhor solução para organizar toda essa avalanche de dados presentes na *Web*, uma vez que possuem, entre outros, mecanismos eficientes para controle de concorrência e para recuperação de informações de modo a preservar a estrutura dos dados. Isto tem possibilitado uma nova área de pesquisa denominada *Web Banco de Dados* [LIM97].

No entanto, para que seja possível a integração da *Web* aos atuais Sistemas de Banco de Dados, se faz necessário a utilização de um *software* chamado *Gateway* [GUN98]. Além

disso, para se construir aplicações *Web* Banco de Dados, requisitos de integração das duas tecnologias devem ser levados em consideração. Dentre esses requisitos estão alguns componentes da *Web* e dos Sistemas de Banco de Dados.

Alguns dos requisitos citados acima se referem aos Servidores *Web* e aos SGBDs e merecem ser estudados mais detalhadamente. O primeiro por ser um dos componentes determinantes no tempo de resposta de uma transação cliente [MUR96]. O outro por ter funcionalidades que permitem, por exemplo, um melhor gerenciamento do acesso aos dados armazenados. Percebe-se que o fator desempenho nesses sistemas influencia diretamente as aplicações que têm por base a *Web* e os sistemas de Banco de Dados.

1.2 Objetivos

Este trabalho faz um estudo das tecnologias de integração da *Web* e dos Sistemas de Banco de Dados e uma análise de como o desempenho de um servidor *Web* é afetado diante de aplicações dinâmicas que realizam acesso a banco de dados via *Web*.

A *World Wide Web*(WWW) é um sistema de informações distribuído baseado em hipertextos, e atualmente contém uma grande quantidade de informações relevantes em todas as áreas do conhecimento humano [LIM97]. Os Sistemas de Banco de Dados assim como a WWW, também se caracterizam pela troca de informações. Dessa forma têm surgido naturalmente propostas de integração dessas duas tecnologias.

O acesso a banco de dados através da *Web* tem se mostrado a cada dia como uma grande necessidade, especialmente para organizações. Por outro lado, são várias as aplicações *Web* que podem usar banco de dados como um mecanismo mais eficiente para o armazenamento de informações [ADR98]. No entanto, um dos grandes problemas nesse novo ambiente que integra a *Web* aos Sistemas de Banco de Dados se refere ao desempenho das aplicações, como pode ser verificado nos estudos desenvolvidos por [LIM97].

Dessa forma, pode-se enumerar os principais objetivos desse trabalho, os quais são:

- Estudar e documentar o estado da arte desta nova área de pesquisa, denominada *Web* Banco de Dados, que integra a tecnologia de Sistemas de Banco de Dados com a *World Wide Web*;
- Desenvolver um estudo que permita fazer uma análise do desempenho do Servidor *Web* cuja carga de trabalho é formada por requisições estáticas e requisições que incluem acesso a um servidor de banco de dados.

1.3 Estrutura da Dissertação

O restante da dissertação está organizado da seguinte maneira:

- O capítulo 2 apresenta os conceitos dos ambiente que integram a *Web* e Banco de Dados. São enumerados os principais problemas e vantagens do ambiente de integração. Em especial, são apresentadas algumas características do servidor *Web* Apache e do sistema gerenciador de banco de dados MySQL.
- O capítulo 3 aborda questões acerca da avaliação de desempenho de sistemas computacionais. Enfatizam-se as principais métricas aplicadas nos problemas de avaliação de desempenho de servidores *Web* e SGBDs. Apresenta-se também uma discussão das três técnicas de avaliação conhecidas: Monitoramento/Mensuração, Modelagem Analítica e Simulação. Ainda são feitas considerações sobre a caracterização da carga de trabalho submetida a sistemas computacionais.
- O capítulo 4 apresenta os conceitos básicos de séries temporais e dos métodos de alisamento exponencial para previsão de séries temporais. Uma ênfase é dada ao método de alisamento exponencial simples e as aplicações de série temporais na avaliação de desempenho de sistemas computacionais.
- O capítulo 5 apresenta uma avaliação de desempenho do servidor *Web* Apache no provedor de acesso à *Internet* FlorianoNet, de forma a se obter um perfil de desempenho do sistema, especificamente no que diz respeito a taxa de processamento e ao tempo de resposta do servidor a uma requisição de usuário. Com complemento dessa avaliação apresenta-se também um prognóstico para cargas futuras do servidor *Web* através da aplicação do método de alisamento exponencial simples para previsão de séries temporais.
- Finalmente no capítulo 6 são apresentadas as conclusões, contribuições e trabalhos futuros.

Capítulo 2

***Web* e Banco de Dados**

2.1 Introdução

Tanto a tecnologia *Web* quanto a de banco de dados se caracterizam pela troca de informação, e a necessidade de integrá-las parece urgente, principalmente porque a tecnologia de banco de dados [MAR99] se apresenta como uma boa solução para organizar toda essa avalanche de dados presente na *Web*. Essa interação deu origem a uma nova área de pesquisa denominada *Web Banco de Dados*, que vem crescendo muito devido ao desenvolvimento de várias aplicações *Web* baseadas em SGBDs e à necessidade de acesso a banco de dados através da *Web* [LIM97]. Este capítulo descreve as principais características do ambiente que integra *Web* e SGBD. A seção 2.2 apresenta os conceitos básicos do ambiente *Web*, os quais serão utilizados no decorrer deste trabalho. São abordadas ainda características do servidor *Web Apache*. Na seção 2.3 são abordados conceitos dos sistemas gerenciadores de banco de dados, e em especial do sistema MySQL, um SGBD Relacional de código-fonte aberto. Na seção 2.4 enumeramos algumas vantagens e problemas relacionados a integração das duas tecnologias. Finalmente na seção 2.5 apresentamos um resumo do capítulo e as conclusões.

2.2 A *World Wide Web*

A *World Wide Web*, ou simplesmente *Web*, é a estrutura arquitetônica que permite o acesso a documentos vinculados espalhados por milhares de máquinas na *Internet* [TAN97], oferecendo a possibilidade de acesso a informações relevantes nas diversas áreas do conhecimento humano.

O desenvolvimento da *Web* começou há pouco tempo, mas precisamente em 1989. No início da década de 90 muitos dos documentos (arquivos) armazenados na *Internet* eram inacessíveis para a maioria dos usuários. Felizmente muita coisa mudou com criação da *World Wide Web* [LAQ95].

A *Web* é um sistema de informação distribuído baseado em hipertextos, que nada mais são do que documentos formatados de acordo com uma linguagem chamada HTML (*HyperText Markup Language*) [SIL99]. O hipertexto organiza a informação de uma forma que estabelece uma interconexão de textos (*links*). Os documentos HTML, são compostos de textos, especificações de fontes e diversas outras instruções para formatação.

Os textos (documentos) visualizados pelos usuários da *Web* podem ser formatados para serem exibidos ao lado de imagens, o que os tornam mais atrativos além de serem

mostrados como documentos hipertexto. Utilizando um programa chamado navegador ou *browser*, o usuário pode ter acesso a um documento (informação), apenas acessando uma área do texto que possua um *link* associado a esse documento.

2.2.1 URL

Um sistema de hipertexto tem a necessidade de que sejam armazenados ponteiros para documentos [TAN96]. A funcionalidade dos ponteiros é fornecida pelas URL (*Uniform Resource Location* - Localizadores de Recursos Universais). Um exemplo de URL pode ser dado por <http://www.florianonet.com.br/notjreg.htm>. Uma URL é composta pelos seguintes componentes:

- Tipo de protocolo. No exemplo esse protocolo é o HTTP (*HyperText Transfer Protocol*).
- Nome do domínio em que o recurso é armazenado (nome da máquina).
- Nome do diretório (Opcional).
- Nome do arquivo.

No exemplo acima, a URL informa ao cliente Web para fazer uma conexão *Internet* usando o protocolo HTTP (*HyperText Transfer Protocol*), cujo documento encontra-se na máquina *host* “*www.florianonet.com.br*” e cujo documento é “*notjreg.htm*”.

Um ponto que merece destaque é que os Localizadores de Recursos Universais fornecem um único nome global para cada documento que pode ser acessado por um sistema Web, dessa forma eles podem ser usados diretamente pelo usuário para que esse tenha acesso a uma informação de seu interesse, sem ter que navegar por um caminho a partir de localizações pré-definidas.

2.2.2 HTML (*Hypertext Markup Language*)

Para a apresentação e formatação de documentos na *Web* é utilizada a HTML [LAQ95], que é uma linguagem de apresentação de hipertexto de uso geral com base em SGML (*Standard Generalized Markup Language*) [SIL99]. Ela é usada para formatar documentos para serem visualizados na *Web* e é baseada em *tags*, que são marcações responsáveis pela formatação das páginas *Web*.

A interpretação das marcações feitas em HTML, não requer qualquer compilador [MAR99]. Assim, a criação e leitura de páginas HTML pode ser feita no editor de texto preferido do programador, além disso as mudanças podem ser feitas através de *browsers*, como Internet Explorer e Netscape. Uma outra característica importante da linguagem HTML é a possibilidade de que sejam gerados formulários contendo ícones e campos de forma a dinamizar uma interação com o usuário.

2.2.3 O Protocolo HTTP

O protocolo básico utilizado na comunicação entre cliente e servidor na *Web* é o HTTP (*Hypertext Transfer Protocol*), um protocolo da camada de aplicação que usa TCP (*Transport Control Protocol*) como protocolo da camada de transporte. Uma das características principais do HTTP é a transmissão de documentos *Web* sobre a *Internet*.

Atualmente o protocolo HTTP é o mais importante da *Web* [MEIG02], sendo utilizado em duas versões, HTTP 1.0 e HTTP 1.1. Basicamente a comunicação com esse protocolo, uma vez estabelecida uma conexão entre o cliente *Web* e o servidor *Web*, ocorre pelo envio de requisições, identificadas por uma URL, por exemplo, feita pelo usuário ao servidor *Web*. Este por sua vez envia normalmente como resposta o conteúdo do arquivo associado a URL através da rede que interliga o cliente e o servidor.

Uma das principais diferenças relativas à versão do protocolo HTTP está no fato de que a versão 1.1 permite o uso de conexões persistentes de forma que um cliente mantenha aberta sua conexão com o servidor no decorrer de toda a interação [MEI02]. Isto reduz o tempo de resposta das aplicações *Web* e o tráfego na *Internet*, tendo em vista que uma nova requisição por um documento não mais vai exigir uma nova conexão TCP.

2.2.4 A Arquitetura Web

A *World Wide Web* é basicamente um sistema cliente-servidor. Os clientes *Web* são responsáveis por receberem as respostas às suas solicitações ao servidor *Web* e verificar a validade dos dados antes de mostrar o documento para o usuário. Atualmente, os principais clientes *Web* são o Internet Explorer da Microsoft, e o Netscape Navigator da Netscape [SIL01].

Os clientes *Web* auxiliam na visualização das páginas (documentos da *Web*). Opcionalmente, o cliente pode estar acompanhado por aplicativos externos usados na apresentação do documento, ou por partes destes, caso o *browser* sozinho não seja capaz de interpretar algum tipo de dado [LIM97]. Do outro lado do sistema, estão os servidores, dentre eles os servidores *Web*, cuja função é atender às solicitações dos clientes por documentos armazenados no sistema de arquivos da plataforma onde se encontram instalados. Os *softwares* para servidores mais conhecidos são: Apache, Internet Information Server, Oracle Web Server, Omni entre outros [MEN00] .

O CGI (*Common Gateway Interface*) [GUN96], é um conjunto de normas e especificações técnicas que definem como o servidor deve receber requisições de execução de programas, efetuar o processamento desses programas e retornar resultados para o requisitante. A Figura 1 ilustra a arquitetura simplificada da *Web* [Lim97].

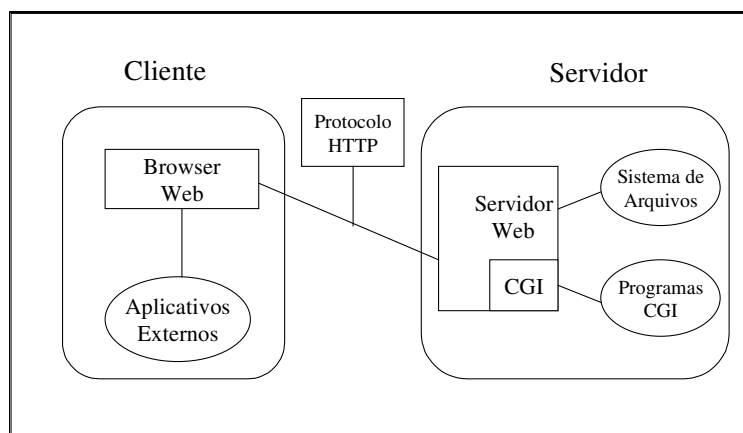


Figura 1 : Arquitetura *Web* Simplificada

2.2.5 Scripts Web e Interface CGI

Um *script Web* é um programa que pode ser executado pelo servidor *Web* em função das solicitações de clientes para proporcionar serviços dinâmicos em ambientes *Web*.

Para [Lim97], o servidor *Web* é responsável por realizar as seguintes operações no momento em que inicia um programa *script*:

- Avaliar a possibilidade de execução do programa *script*.
- Iniciar o programa *script* e garantir que os dados de entrada do cliente *Web* sejam passados para o programa *script*.

- Receber os dados retornados pelo programa *script* e os enviar de volta ao cliente *Web*.
- Se algo errado acontecer com o programa *script*, enviar uma mensagem de volta ao cliente *Web*.
- Encerrar a conexão de rede apropriadamente, quando o programa *script* finalizar sua execução.

Com o objetivo de implementar as ações descritas acima, foi especificada a interface CGI (*Common Gateway Interface*) [GUN96] apud [MAR99]. CGI é uma interface padrão implementada pela maioria dos servidores *Web* existentes, para a execução de programas *scripts* externos ao servidor *Web*.

A grande vantagem de se escrever aplicações *Web* usando a interface CGI é a boa portabilidade de aplicações no servidor. No entanto, toda vez que o servidor *Web* recebe uma requisição CGI ele carrega um novo programa, executa-o, envia a resposta para o cliente e é finalizado, resultando assim num significativo *overhead* do servidor [MEI02].

2.2.6 SSIs e APIs de Servidores *Web*

Outras técnicas que também podem gerar documentos *Web* são as SSIs (*Serves Side Includes*) e as APIs (*Application Programming Interface*). As seções seguintes abordam cada uma delas.

2.2.6.1 SSIs

SSIs são trechos de códigos, também denominados *tags*, inseridos em um documento *Web* e que serão executados pelo servidor *Web* no momento em que um cliente solicita um documento.

Através de códigos SSIs, dados dinâmicos num documento *Web* podem ser obtidos de forma mais simples [LIM97]. No entanto, algumas desvantagens podem ser observadas em aplicações SSIs, tais como: problema de segurança no que diz respeito a acesso aos dados e baixa de desempenho, tendo em vista que o servidor deve examinar todo documento a procura de códigos SSI, o que pode acarretar sobrecarga no processamento de pedidos de cliente *Web*.

2.2.6.2. APIs

É um conjunto de funções que são incorporados ao servidor *Web* podendo substituir completamente a interface CGI para a execução de programas *script* [MEL98]. As principais APIs são NSAPI (Netscape), ISAPI (Microsoft) e APIapache.

Enquanto as aplicações APIs apresentam um excelente desempenho por um lado, já que podem ficar residentes, por outro têm problemas com a portabilidade, pois não há padronização das APIs, o que conseqüentemente faz com que uma aplicação API só funcione junto ao servidor para o qual foi projetada.

2.2.7 Linguagens de Programação na *Web*

Várias são as linguagens que podem ser usadas no ambiente *Web*. No entanto, analisaremos brevemente aqui, apenas aquelas chamadas de “código móvel”, como Java e JavaScript, e a linguagem PERL.

2.2.7.1 Java

A linguagem Java foi criada pela Sun Microsystems. Com o aparecimento da *Web*, seus criadores perceberam-na como uma valiosa ferramenta para ambientes de rede distribuídos como a *Web*. Contudo, como aponta [ARN97], Java é uma poderosa linguagem de programação de propósito geral, adequada à construção de uma série de aplicações que não dependem das características da rede.

Java é uma linguagem orientada a objetos que apresenta algumas semelhanças com C e C++, apesar de não ser compatível com nenhuma das duas. Trata-se portanto de uma nova linguagem. A principal característica de Java é a portabilidade entre plataformas distintas, o que pode ser explicado devido a sua arquitetura [LIM97].

Segundo [TAN97] a idéia de usar Java para as páginas interativas da *Web* se deve ao fato de que uma página da *Web* pode apontar para um pequeno programa em Java, chamado *applet*. Assim o *browser* busca este programa, e o *applet* é instalado e executado de forma segura na máquina-cliente.

Duas razões, pelas quais os usuários querem incluir *applet* Java em suas páginas *Web* são: os *applets* permitem que as páginas se tornem intensivas e além disso possibilitem a inclusão de imagens animadas e sons nas páginas da *Web*.

O sistema Java possui três partes que são: (i) compilador Java para gerar *bytecode*, (ii) *browser* que compreenda *applet* e (iii) interpretador de *bytecode*. Atualmente, diversos *browsers* comerciais oferecem suporte para linguagem Java, entre eles o Netscape_Navigator e o Internet Explorer da Microsoft.

No que se refere a Banco de Dados, a implementação de acesso a SGBD em Java permite realizar tarefas como: criação de um banco de dados vazio, conexão com um banco de dados, criação de tabelas, inserção ou remoção de registros na tabela e consulta de dados [FUN99].

2.2.7.2 JavaScript

A linguagem JavaScript foi criada pela Sun Microsystem e a Netscape, especificamente para o ambiente *Web*. É uma linguagem proveniente de Java, mas diferentemente desta é uma linguagem apenas interpretada e destinada a tarefas mais simples. Segundo[MCC97] “as portas precisam ser inteligentes suficientemente para interpretar o código contido numa página HTML, tanto no cliente, como no servidor, pois o código JavaScript não é compilado, mas sim, interpretado”.

Da mesma forma que Java, JavaScript permite que clientes *Web* individualmente realizem tarefas de processamento, tais como verificações de campos de entrada, aliviando a carga de procura do Servidor *Web*. Como é uma linguagem proprietária, apresenta algumas desvantagens como restrições de plataformas e suporte para a maioria dos *browsers*.

2.2.7.3 PERL

Assim como JavaScript, a linguagem PERL é uma linguagem interpretada, tornando assim a programação mais flexível e sem imposição de regras excessivamente rígidas na composição de *scripts* CGI. PERL, segundo [HER97] é a linguagem mais utilizada para escrever programas CGI, tornando-se assim um padrão de fato para a grande maioria de aplicações na *Web*, dada a sua portabilidade, estabilidade e riqueza de recursos.

2.2.8 O Servidor *Web* Apache

Nesta seção serão descritas algumas características do servidor Apache, o servidor *Web* mais popular do mundo. Este sistema é gratuito e está presente em quase todas as distribuições de Linux. Atualmente, servidor o Apache é utilizado por cerca de 60% dos *sites Web* [APA02].

A popularização da *Web* nos últimos anos tem obrigado as empresas a disponibilizarem conteúdo na mesma. Nesse sentido, a utilização de um servidor *Web* como o Apache é imperativa para qualquer empresa que deseja manter uma presença na rede.

O Apache é executado na forma de um *daemon* chamado **httpd**, que é lançado durante o processo de *boot* da máquina. Um *daemon* mestre cria um conjunto de *daemons* escravos (processos filhos) para atender às requisições de páginas *Web*, visando aumentar o desempenho do sistema. Esses *daemons* são executados sob a identidade de um usuário especial (normalmente **nobody**), para evitar problemas de segurança. Por essa razão, todos os arquivos acessíveis aos *daemons* (páginas html, CGIs, imagens) devem estar com permissões de acesso adequadas para terceiros.

Os principais diretórios usados pelo Apache na distribuição Linux/RedHat [RED02] são:

- **/etc/httpd/conf/httpd.conf** : arquivo de configuração.
- **/var/log/httpd** : arquivos de logs de acessos e erros.
- **/home/httpd** : diretório principal do serviço WWW.
- **/home/httpd/html** : páginas *default* do servidor (index.html).
- **/home/httpd/cgi-bin** : programas executáveis no padrão CGI.
- **/home/httpd/icons** : ícones usados pelo servidor.
- **/usr/lib/apache** : módulos adicionais usados pelo servidor (plug-ins).
- **\$HOME/public_html** : páginas pessoais de cada usuário.

A localização dos diretórios pode ser alterada editando-se o arquivo de configuração.

A configuração *default* do servidor Apache somente permite a execução de CGIs presentes no diretório **/home/httpd/cgi-bin**. Essa configuração deve ser alterada caso se

deseje habilitar a execução de CGIs em outros diretórios, como por exemplo os diretórios HOME dos usuários.

2.3 Sistemas Gerenciadores de Banco de Dados (SGBDs)

O principal objetivo de um SGBD é proporcionar um ambiente conveniente e eficiente para que seja possível a recuperação e o armazenamento das informações do Banco de Dados [SIL99]. De fato, as aplicações que envolvem um grande volume de dados, necessitam que esses dados sejam bem estruturados para efeito de armazenamento, e que mecanismos para a manipulação de informações sejam definidos.

Como dito anteriormente, o crescimento da *Web* trouxe novas aplicações baseadas em banco de dados. Os mecanismos eficientes e padronizados de armazenamento e acesso aos dados por parte dos SGBDs, consistem em alguns dos motivos para o desenvolvimento dessas aplicações. Esses mecanismos podem, por exemplo, dar suporte a transações em ambientes multiusuário e permitir a execução otimizada de consultas, melhorando assim o desempenho das aplicações.

Alguns SGBDs que podem ser utilizados no desenvolvimento de aplicações que disponibilizam acesso a base de dados pela Internet são: Microsoft SQL Server, IBM DB2, Oracle Application Server, Sybase e MySQL [SIL01].

2.3.1 Arquiteturas de Sistemas de Bancos de Dados

Em um Sistema de Bancos de Dados a arquitetura é bastante influenciada por aspectos como rede, paralelismo e distribuição. A seguir apresentamos de forma breve algumas características das arquiteturas de Sistemas de Bancos de Dados.

2.3.1.1 Sistemas Centralizados

Sistemas de Bancos de Dados Centralizados, segundo [ELM94], são aqueles executados sobre um único sistema computacional que não interagem com outros sistemas.

Esses podem ser divididos em sistemas monousuários e sistemas multiusuários. Os primeiros são sistemas de um só usuário como as estações de trabalho e os computadores

pessoais. Os sistemas multiusuários atendem a um grande número de usuários que estão conectados ao sistema por meio de terminais, podendo se caracterizar como sistemas de alto desempenho. A Figura 2 ilustra um sistema computacional centralizado [SIL99] .

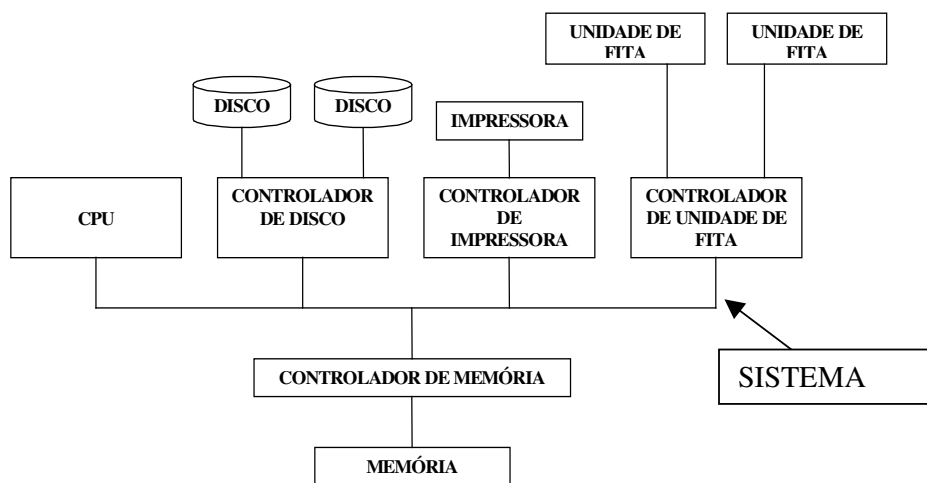


Figura 2: Sistema de banco de dados centralizado

2.3.1.2 Sistemas Cliente/Servidor

Os terminais conectados aos sistemas centralizados estão sendo substituídos cada vez mais por computadores pessoais, dada sua rapidez, poder e preço baixo. Esta tendência, está fazendo com que sistemas centralizados, de certa forma, atuem como sistemas servidores atendendo às solicitações de sistemas clientes.

A arquitetura de um sistema cliente-servidor é definida pela distribuição dos componentes lógicos de um programa de aplicação pelas diferentes plataformas. Para [ELM00] os tipos básicos de arquitetura são:

- Arquitetura de camada única: todos os componentes são centralizados em uma única plataforma.
- Arquitetura de múltiplas camadas: os componentes lógicos de um programa de aplicação são distribuídos por duas ou mais plataformas.

Na arquitetura de múltiplas camadas, o modelo que atualmente é mais utilizado é o de duas camadas. A Figura 3 ilustra representação da estrutura geral de um sistema cliente-servidor.

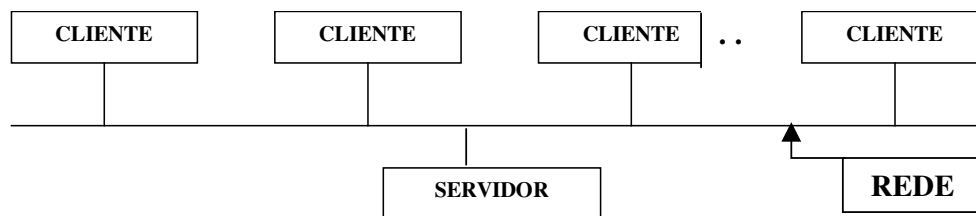


Figura 3: Sistema de banco de dados cliente/servidor

2.3.1.3 Sistemas Paralelos

Os Sistemas de Bancos de Dados Paralelos possuem vários processadores e vários discos conectados por rede de alta velocidade. As arquiteturas de bancos de dados paralelos permitem compartilhamento desses discos, compartilhamento de memória e arquiteturas hierárquicas [SIL99].

2.3.1.4 Sistemas Distribuídos

Um sistema de banco de dados distribuído corresponde a um conjunto de banco de dados parciais e independentes que compartilha um esquema de banco de dados comum e que é responsável pela coordenação de processamento de transações com acesso de dados não-local [OZS99]. A comunicação entre os processadores se dá através das redes de comunicação que tratam do roteamento e das estratégias de comunicação.

2.3.2 O Sistema Gerenciador de Banco de Dados MySQL

O objetivo desta seção é abordar o sistema gerenciador de banco de dados MySQL [MSL02], que integra o conjunto de componentes de *software* do provedor de acesso à *Internet* FlorianoNet. A ênfase dada a este sistema se dá em função da análise desenvolvida no Capítulo 5 deste trabalho, sobre o desempenho de um servidor *Web*, no qual uma parte das requisições submetidas incluem acesso a um servidor de banco de dados MySQL.

O MySQL é de domínio público, exceto na versão Windows ou de uso profissional. É um sistema de nível corporativo, conveniente com o ambiente multiusuário e multitarefa e que conta com um ótimo sistema de segurança. Através do comando GRANT pode-se atribuir privilégios aos usuários, e assim restringir o acesso de um determinado usuário ao banco de dados. É ideal para a *Internet*. Por ser gratuito, é bastante utilizado por universidades e

organizações sem fins lucrativos. É um sistema rápido, confiável e por trabalhar muito bem como servidor de banco de dados baseado na *Internet*, é o banco de dados preferido de muitos provedores de serviços de *Internet* [MAR00].

Pode-se acessar uma base de dados através de PHP, Perl, Java, C++, entre outros. No entanto, PHP é o maior usuário desse gerenciador [DAT01]. Os *scripts Web*, executados pelo servidor *Web* em função das requisições dos clientes que envolvem acesso ao servidor de banco de dados MySQL do provedor FlorianoNet, são desenvolvidos em PHP.

O MySQL pode se adequar a várias arquiteturas [MAR00]. Pode, por exemplo, ser utilizado em uma arquitetura cliente/servidor ou como um banco de dados independente. Outra vantagem desse sistema está no fato de ser totalmente compatível com ANSI SQL 92.

2.4 Integração *Web* e Banco de Dados

Dados eletrônicos ou documentos *Web*, ao invés de serem compostos manualmente, cada vez mais estão sendo gerados automaticamente a partir de banco de dados [ABI00]. Nesse sentido, percebe-se que a *Web* tem proporcionado o desenvolvimento de muitas aplicações baseadas em SGBDs, o que só reforça a necessidade de uma discussão acerca da integração dessas duas tecnologias.

A importância de dispor de interfaces de Banco de Dados para a *World Wide Web* [OZS99] deve-se por duas razões: 1) o crescimento do comércio eletrônico na *Web*, o qual exige que bancos de dados utilizados para o processamento das transações estejam ligados à *Web*; 2) as fontes de dados HTML fixas são limitadas em relação à apresentação dos conteúdos para os usuários.

Algumas vantagens da integração *Web* e Banco de Dados [ABI00 , LIM97], são:

- A *Web* apresenta deficiências nas formas de armazenamento e acesso a dados. Em contrapartida, a tecnologia de Banco de Dados oferece técnicas de armazenamento e linguagem de consulta que possibilitam acesso a grandes conjuntos de dados.
- O vultoso volume de dados gerenciado pelos SGBDs pode ser colocado à disposição de um número imenso de usuários em redes locais ou remotas.
- Os SGBDs têm problema com relação ao desenvolvimento de aplicações dirigidas a usuários, em função da dependência das ferramentas proprietárias distribuídas pelos

fabricantes. A *Web*, entretanto proporciona uma boa interface gráfica para aplicações em banco de dados multiplataforma.

- A *Web* está possibilitando o desenvolvimento de novas aplicações baseadas em SGBDs. Exemplos dessas aplicações são as bibliotecas virtuais e o comércio via *Internet*.

A Figura 4 ilustra uma arquitetura típica do ambiente de integração banco de dados e *Web* [OZS99]. Nela tem-se a presença de um Gateway, um *software* responsável pelo gerenciamento da comunicação e por serviços de aplicação entre o servidor *Web* e o SGBD.

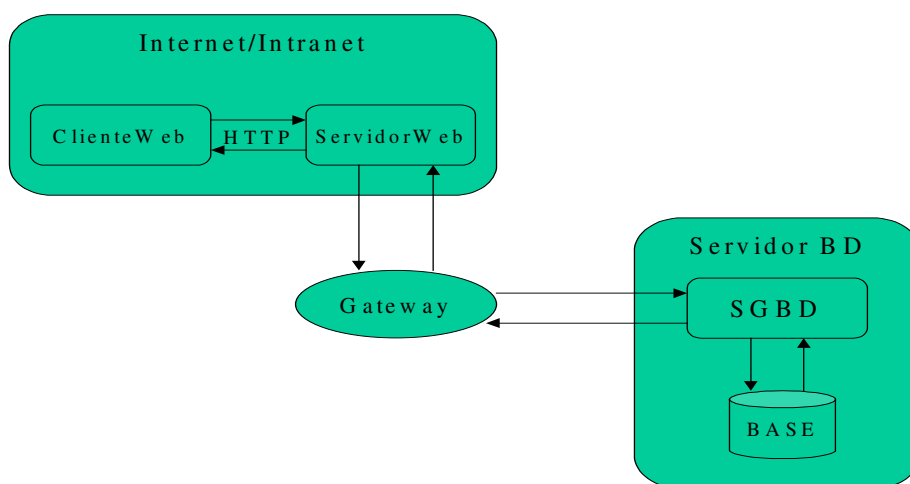


Figura 4: Arquitetura de integração *Web* e banco de dados

De acordo com a Figura 4, o fluxo de dados de uma aplicação no ambiente de integração funciona da seguinte forma :

- O cliente *Web* faz uma solicitação (página) ao servidor *Web* .
- O servidor *Web* dispara processo no *gateway* enviando parâmetros do cliente.
- O *gateway* trata os parâmetros do cliente, formula o comando SQL, abre conexão com o SGBD e aguarda resposta.
- O SGBD executa SQL e retorna resposta.
- O *gateway* repassa os dados para o servidor *Web* em formato compreensível pelo cliente .
- O servidor *Web* retorna os dados para o cliente *Web*.

- O cliente *Web* identifica os formato e exibe os dados.

É importante que se diga que o funcionamento descrito acima não é sempre o mesmo, ou seja, a dinâmica do funcionamento pode variar de arquitetura para arquitetura.

2.4.1 Problemas da Integração *Web* e Banco de Dados

Vários são os problemas técnicos decorrentes do desenvolvimento de aplicações nesse ambiente de integração, dos quais alguns dos mais relevantes são brevemente discutidos a seguir.

2.4.1.1 Sistemas Legados

Sistemas Legados são aqueles desenvolvidos em plataformas do passado, apresentam resistência a modificações e evolução. Os SGBDs de primeira e segunda gerações se caracterizam como um desses sistemas. A maioria dos SGBDs que são comercializados hoje não levam em consideração as necessidades especiais do ambiente *Web*, além dos investimentos que já foram feitos pelos seus proprietários.

2.4.1.2 Problemas Transacionais

Outro problema inerente à integração SGBD e *Web* se refere às transações *Web*, tendo em vista a natureza “sem estado” (*stateless*) dessas transações em decorrência do protocolo HTTP usado na comunicação cliente e servidor *Web*. Numa transação *Web* (LIM97) “o servidor *Web* não armazena qualquer informação sobre a aplicação e/ou usuário da aplicação, ou seja, o servidor *Web* não mantém estado da aplicação entre interações”.

Uma transação para os Sistemas de Gerenciamento de Bancos de Dados é tida, segundo [MEL98] como uma sequência de operações elementares que são tratadas como um elemento indivisível (atômico) no que se refere à sua recuperabilidade. Nas transações *Web* e Banco de Dados pode-se perceber as seguintes características:

- Têm início no cliente *Web*, que pode estar localizado em qualquer parte do planeta.
- Ocorre sob o protocolo HTTP, o que pode provocar uma descontinuidade da sessão.

- São formulados através da interação com o usuário e que poderia levar a tempo “infinito”.
- São orientadas a páginas. Uma página *Web* corresponde a uma transação, mas uma transação pode resultar em várias páginas *Web*.
- São baseadas no modelo navegacional.

2.4.1.3 Segurança e Acesso ao Banco de Dados

Uma outra questão importante com relação ao funcionamento dos SGBDs é a segurança e acesso ao banco de dados. Os mecanismos de segurança e acesso ao Banco de Dados, sendo o usuário desse banco a própria *Web*, podem ser os mesmos já existentes nos SGBDs, como por exemplo o uso de visões .

Outros problemas que são relevantes no que diz respeito à segurança e acesso ao banco de dados de acordo com [LIM97], são:

- Gerenciadores de acesso (senhas de *login*), que traz como consequência a perda das características de acesso individuais de cada usuário.
- O número de acessos simultâneos aos SGBDs é limitado, o que significa dizer que um usuário ou grupo de usuários pode ter uma transação não completada.
- Os problemas com a autorização para a execução de uma aplicação *Web* que esteja consultando ou atualizando um banco de dados, em alguns *gateways* de integração banco de dados e *Web*.
- Não existem mecanismos de recuperação para tratar das paradas e falhas de clientes e servidores *Web*, e os mecanismos dos SGBDs podem não ser suficientes, dependendo da aplicação.

2.4.1.4 Desempenho

O desempenho no que se refere ao ambiente *Web* está relacionado ao tempo de resposta a uma solicitação do usuário, que depende do desempenho dos diversos componentes da *Web*. No caso das transações *Web* Banco de Dados, os três componentes cujos desempenhos interferem diretamente são: o cliente *Web*, o tráfego na rede e o servidor *Web*.

No caso do cliente *Web*, os problemas de desempenho se relacionam à capacidade de exibir os dados enviados pelo servidor *Web*, ao nível de gerenciamento da *cache* local a execução de códigos associados aos dados enviados, *aplets* Java por exemplo.

A questão do tráfego na rede é extremamente complexa, pois depende de várias variáveis. Entre outras destaca-se o tempo necessário para prover acesso remoto, a escolha da melhor rota para conexão com o servidor *Web*, a transmissão dos dados na rede e o modelo probabilístico gerador do tráfego.

No que se refere ao servidor *Web*, o seu desempenho [MUR96] é muito afetado em virtude da carga de pedidos que recebe, da plataforma de hardware e do ambiente de *software* (sistema operacional e subsistema de gerência de arquivos, entre outros).

Outros fatores que podem afetar de alguma forma o desempenho de aplicações banco de dados e *Web* são: o balanceamento de carga, o mínimo de usuários e a utilização de consultas.

2.4.1.5 Desenvolvimento e Portabilidade

Com relação ao desenvolvimento e portabilidade de aplicações *Web* baseadas em bancos de dados, destacam-se os pontos:

- Inexistência de metodologia de desenvolvimento de aplicações *Web* Bancos de Dados, até por ser uma tecnologia recente.
- A falta de padrões estáveis.
- É recomendável a existência de mecanismos de transferência de variáveis de entrada do cliente *Web* para as consultas SQL no servidor do banco de dados.

2.4.1.6 Linguagens de Programação

Linguagens como Java, JavaScript, Vbscript, C, C++, Lua e PERL, têm sido bastante utilizadas para aplicações banco de dados e *Web*. Além disso, os fabricantes de SGBDs, não deixaram de usar suas linguagens proprietárias no desenvolvimento de aplicações *Web* Banco de Dados. Para essas aplicações, no entanto, essa diversidade de linguagens traz grandes problemas no que se refere à complexidade e portabilidade.

2.5 A Linguagem XML

XML (*Extensible Markup Language* - Linguagem Extensível de Marcação) é uma linguagem de marcadores que surgiu como um novo padrão proposto pelo *World Wide Web Consortium* para complementar a HTML para troca de dados na *Web*.

A linguagem XML é um subconjunto de SGML (*Standard Generalized Markup Language*). No entanto, ao contrário de HTML foi projetada de forma exclusiva para descrever o conteúdo, em vez de descrever a apresentação dos dados. Para [ABI00] três grandes aspectos a diferem de HTML:

- Possibilidades de definição de novas marcas por parte do usuário de forma livre.
- Estruturas podem ser aninhadas em qualquer nível.
- Um documento XML pode conter uma descrição opcional de sua gramática.

A convergência das tecnologias *Web* e Banco de Dados, até então de difícil conciliação, está sendo provocada pelo desenvolvimento de XML para dados na *Web* e o intimamente ligado desenvolvimento de dados semi-estruturados pela comunidade de pesquisa em Banco de Dados [ABI00]. A prova disso são as diversas pesquisas que têm apontado no sentido de estender técnicas tradicionais de banco de dados para manipulação de dados semi-estruturados. A maioria destas pesquisas [VID01] tem focalizado no desenvolvimento de modelos de dados e linguagens de consulta para dados semi-estruturados.

XML tem permitido, dentre os seus diversos benefícios, a representação interna de dados armazenados em SGBDs para a publicação na *Web*. Adicionalmente, novos SGBDs têm sido desenvolvidos com o objetivo de gerenciar e armazenar documentos XML em seu formato original.

2.6 Conclusões

Neste capítulo foram discutidas brevemente algumas técnicas utilizadas no desenvolvimento de aplicações *Web* integradas a SGBDs, levando em consideração algumas vantagens e desvantagens com relação ao uso de uma ou outra técnica. Neste sentido, foi feita de certa forma uma comparação entre essas técnicas.

A integração da *World Wide Web* com banco de dados é uma área em desenvolvimento, e muito do que se tem hoje é objeto de intensa pesquisa e experimentos. As diversas vantagens em integrar essas duas tecnologias mostram porque é cada vez maior o número de aplicações *Web* que estão usando os recursos oferecidos pelos atuais sistemas de banco de dados. No entanto, vários problemas técnicos têm dificultado a construção de aplicações capazes de atender às exigências dos modernos sistemas de computação de banco de dados cliente/servidor.

Diante dessas dificuldades, que são conseqüências da falta de um padrão para o ambiente de integração, os desenvolvedores de aplicações são levados a procurar ferramentas proprietárias na busca de soluções para os problemas inerentes a este novo ambiente.

CAPÍTULO 3

Avaliação de Desempenho de Sistemas Computacionais

3.1 Introdução

A avaliação de desempenho de sistemas computacionais é uma arte [JAI91]. Neste sentido, uma avaliação de sucesso não pode ser produzida de forma mecânica. É necessário que se tenha um certo conhecimento do sistema de maneira que isto facilite a seleção da metodologia, a caracterização da carga de trabalho e até mesmo das ferramentas a serem utilizadas.

A definição do problema real, de acordo com o estabelecido pelas ferramentas e técnicas, é de extrema importância, no entanto não se pode esquecer que para um mesmo problema diferentes métricas de avaliação de desempenho podem ser usadas. O interesse neste capítulo é discutir as questões acima. Assim, o capítulo está organizado em 5 seções, das quais esta é a primeira. A seção 3.2 enumera alguns erros comumente cometidos durante o processo de avaliação de desempenho. Na seção 3.3 é apresentada uma metodologia para avaliar o desempenho de sistemas computacionais. Nas seções 3.4 e 3.5 discutem-se algumas métricas e técnicas utilizadas na avaliação de desempenho. Na seção 3.6 são feitas considerações sobre cargas de trabalho submetidas a sistemas computacionais. A seção 3.7 aborda a avaliação de desempenho de servidores *Web* de um modo geral. Finalmente a seção 3.8 apresenta as conclusões.

3.2 Erros em Avaliação de Desempenho

Proceder corretamente na avaliação de desempenho de um sistema de computadores não é uma tarefa fácil. Portanto, são vários os equívocos que se comete. A seguir são listados alguns dos erros mais comuns durante um processo de avaliação de desempenho de sistemas computacionais:

- Não estabelecer objetivos.
- Falta de metodologia (estabelecer objetivos, métricas, etc.).
- Uso de métricas incorretas.
- Falta de caracterização da carga de trabalho.
- Erro nas técnicas de avaliação.

- Não tomar conhecimento de parâmetros, tais como comprimento das transações, taxa de chegada de requisições, etc.

A lista de erros acima contém apenas aqueles julgados importantes para os objetivos deste trabalho. Uma discussão completa dessas questões pode ser encontrada em [JAI91].

3.3 Metodologia para Avaliação de Desempenho

Para se realizar a avaliação de desempenho de um sistema é imprescindível a utilização de um procedimento sistemático, isto é, de uma metodologia. Para [MEI02], uma metodologia para avaliar o desempenho consiste de um conjunto de passos durante o processo de avaliação que, seguidos de forma adequada, produzirão dados e conclusões acerca do desempenho computacional do sistema sob avaliação. Os passos a seguir fazem parte de uma metodologia definida em [JAI91] e [MEI02].

1. Definição de objetivos. Nesta fase inclui-se o entendimento das características do sistema e se define o que deve ser melhorado nele. O conhecimento das características do sistema facilita a especificação dos objetivos. Estes por sua vez, permitem a seleção de técnicas de avaliação mais adequadas e auxiliam nas variáveis que deverão ser analisadas [VIC98]. Exemplos de objetivos:

- melhorar o tempo de resposta para uma requisição de usuário;
- aumentar a taxa de processamento ou *throughput* do sistema;
- verificar se é possível adiar ou evitar a aquisição de *hardware* adicional.

2. Seleção das métricas de desempenho. Uma métrica é um critério usado para avaliar o desempenho. Em se tratando de sistemas computacionais é natural a utilização de uma dimensão computacional para avaliar o seu desempenho. No entanto, sistemas computacionais podem ser avaliados no aspecto comercial como é o caso de servidores de comércio eletrônico, para os quais podem ser utilizadas métricas comerciais [MEI02]. Neste trabalho serão discutidas apenas métricas computacionais.

3. Seleção da técnica de avaliação. A seleção da técnica apropriada depende da disponibilidade de recursos para resolver o problema de desempenho. Algumas vezes não dispomos, por exemplo, das ferramentas mais adequadas para resolver um problema. Outros fatores que influenciam a escolha da técnica é o tempo disponível para avaliação e o grau de precisão que se deseja para os resultados. As técnicas desenvolvidas com o propósito de avaliar o desempenho de um sistema computacional, são denominadas

monitoramento/mensuração, simulação e modelagem analítica e são discutidas na seção 3.5.

4. **Caracterização da carga de trabalho.** A caracterização da carga de trabalho é outra etapa extremamente importante, tendo em vista que nenhum problema de desempenho pode ser de fato solucionado sem que antes a carga de trabalho a ser processada pelo sistema tenha sido especificada. Em sistemas de banco de dados, por exemplo, a carga de trabalho pode ser caracterizada pelo conjunto de consultas submetidas ao sistema. Uma abordagem mais detalhada sobre a carga de trabalho, é feita na seção 3.6.
5. **Aplicação da técnica de avaliação.** Nesta etapa será aplicada a técnica de avaliação, cuja duração e complexidade vai depender da técnica escolhida. Esse é o momento onde se utilizam ferramentas para medir os resultados, por exemplo.
6. **Análise e interpretação de resultados.** A interpretação de resultados por um analista é parte chave para a análise de desempenho. Nesta etapa, é possível que se obtenham vários resultados diferentes para o mesmo problema. Nesse sentido, esta etapa deve ser realizada com bastante cuidado.
7. **Apresentação dos resultados.** É importante também que os resultados sejam apresentados de uma forma clara para que sejam facilmente compreendidos. A apresentação através de formas gráficas é bastante útil para um melhor entendimento dos resultados.

3.4 Métricas de Avaliação de Desempenho

Uma vez que o sistema tenha sido claramente compreendido e os objetivos da avaliação de desempenho definidos, é o momento de identificar as métricas. Anteriormente, neste trabalho foi definida uma métrica como sendo um critério usado para avaliar o desempenho do sistema. Neste caso, um conjunto de critérios ou métricas de desempenho deve ser especificado para que seja possível realizar uma avaliação adequadamente.

A seleção das métricas depende, entre outros fatores, do sistema a ser avaliado e das ferramentas disponíveis para a obtenção dos dados do sistema. Um servidor *Web*, por exemplo, pode ser avaliado pelo *throughput* ou taxa de processamento e pelo tempo de resposta [MEN00].

Um fato que pode auxiliar na identificação das métricas é analisar a lista dos serviços oferecidos pelo sistema. Por exemplo, um banco de dados oferece o serviço de respostas às

consultas. Para cada serviço requisitado existem várias possibilidades de resultados [JAI91]. As métricas, de um modo geral, estarão relacionadas com a disponibilidade dos serviços do sistema, com a velocidade ou com a precisão.

Em um sistema de computação compartilhado, como é o caso de servidores *Web* e servidores de banco de dados, dois tipos de métricas de desempenho devem ser consideradas: gerais e especializadas. Métricas especializadas refletem a utilidade de cada usuário, enquanto métricas globais refletem a utilidade de todo o sistema. Tempo de resposta e taxa de processamento podem ser medidas para cada indivíduo como igualmente de todo o sistema.

A seguir são citadas algumas das métricas de desempenho mais usadas na avaliação de desempenho de sistemas computacionais [FER83, DEI90] apud [VIC98]:

- *Taxa de processamento*: é utilizada para medir a quantidade de trabalho executado pelo sistema num determinado período de tempo;
- *Tempo de resposta*: corresponde ao intervalo de tempo entre o instante entre a entrada de um comando, em um sistema de interação com o usuário e o instante em que as respostas começam a ser visualizadas no terminal;
- *Tempo de reação*: tempo decorrido entre o instante no qual o sistema recebe um comando para ser executado e o instante no qual a CPU inicia a execução desse comando;
- *Disponibilidade*: equivale à taxa de tempo total na qual o sistema está à disposição dos usuários;
- *Capacidade*: corresponde ao valor máximo que a taxa de processamento pode atingir;
- *Nível de multiprogramação*: corresponde à quantidade de programas que podem ser executados simultaneamente em um sistema e que, conseqüentemente, estarão competindo por recursos.

Das métricas descritas acima, duas são bastante utilizadas nos problemas de desempenho de servidores *Web* e sistemas de banco de dados, são elas: taxa de processamento e tempo de resposta. A taxa de processamento no caso dos servidores *Web* representa a taxa de dados que o servidor pode produzir [MUR96]. Esta taxa pode ser medida, por exemplo, pelo número de requisições HTTP atendidas por segundo. O tempo de resposta do servidor é o intervalo de tempo entre o início e o fim de uma requisição HTTP [MEN00]. No capítulo 5

deste trabalho, serão consideradas essas duas métricas para fins da avaliação de desempenho de um servidor *Web Apache*.

É importante salientar que o conhecimento acerca do que significa cada métrica a ser utilizada numa avaliação é importante. Contudo, saber como essas métricas são obtidas tem uma importância ainda maior. Finalmente, vale ressaltar que a habilidade de se utilizar métricas que são relevantes para o sistema, pode levar a conclusões mais sólidas.

3.5 Técnicas de Avaliação de Desempenho

As técnicas de avaliação de desempenho de sistemas podem ser utilizadas para se fazer um planejamento da capacidade futura do sistema ou para se obter soluções para os problemas encontrados quando da execução do sistema, isto é, ajustar o desempenho deste. Essas técnicas podem ser classificadas em técnicas de medição e de previsão [DEI90] apud [VIC98].

Para as técnicas de medição, os valores são obtidos do próprio sistema. Exemplo deste tipo de técnica é **mensuração**. Ao contrário da medição, nas técnicas de previsão como modelagem analítica e simulação, os valores são estimados e sua obtenção ocorre em função de modelos e não da execução real do sistema.

Nas seções seguintes serão discutidas algumas características dessas três técnicas.

3.5.1 Monitoração/Mensuração

Consiste na coleta e na análise de dados de execução de um sistema para obtenção de um perfil de desempenho. Esta técnica é orientada para sistemas já existentes ou quando se tem a intenção de projetar uma melhora para versões posteriores de um sistema.

Monitores podem se implementados em *hardware* ou *software*, ou ainda podem ser híbridos, isto é, incorporam tanto características de monitores de *hardware* como de *software*. Os que são baseados em *hardware* coletam dados de componentes de *hardware* de um sistema distribuído, registrando informações sobre a execução das aplicações do sistema [MEI02]. Monitores baseados em *software* consistem em um conjunto de rotinas inseridas na aplicação com o propósito de coletar dados de desempenho.

A técnica de mensuração apesar de ser a mais dispendiosa é uma das técnicas mais utilizadas na avaliação de desempenho de sistemas computacionais. Muitas são as vantagens quando se utilizar esta técnica. Algumas delas segundo [VIC98] se referem a: utilização do sistema real na avaliação; flexibilidade na determinação dos parâmetros que serão monitorados e na modificação das estatísticas que são coletadas, e disponibilidade de *software* para monitoramento.

Muitos SGBDs permitem a coleta de estatísticas de desempenho. Para o SQL Server da Microsoft, por exemplo, duas ferramentas gráficas fornecem a capacidade de examinar as atividades que estão afetando o desempenho do seu servidor, o SQL Server Profiler e o SQL Performance Monitor [COF99]. O SQL Server Profiler tem a capacidade de fornecer dados em tempo real acerca das atividades que estão acontecendo no servidor, além de monitorar chamadas de procedimento remoto.

O SQL Server Profiler fornece mais informações sobre o desempenho do banco de dados. No entanto o SQL Performance Monitor apresenta o recurso adicional de agendar *jobs* e gerar alertas quando os limites de desempenho são ultrapassados.

Apesar das vantagens descritas acima e de se caracterizar como a técnica na qual os resultados são mais confiáveis e próximos da realidade, o monitoramento possui uma enorme desvantagem que é a dificuldade de se fazer uma análise sobre a grande quantidade de dados coletados. Além disso, muitos dos parâmetros não representam o limite das variáveis encontradas no mundo real. Deste modo, a precisão dos resultados pode ter uma variação muito grande quando se usa esta técnica.

3.5.2 Modelagem Analítica

Modelos analíticos são compreendidos como representações matemáticas do sistema computacional ou de seus componentes. Esses modelos são bastante utilizados na fase de projeto, assim com para o planejamento da capacidade do sistema [JAI91].

Essa técnica leva uma enorme vantagem com relação ao monitoramento e simulação (ver seção 3.5.3) em termos de tempo requerido para a construção do modelo, coleta de dados e obtenção de resultados [DEL02]. Um das causas é a grande quantidade de linguagens para a programação desses modelos, que por sua vez não requerem muito tempo de computação para executarem.

Outra vantagem da modelagem analítica é com relação aos custos. Se compararmos, por exemplo, o monitoramento que necessita de equipamentos reais e instrumentos, além de tempo disponível para medição, esta alternativa é indiscutivelmente mais barata.

Contudo, uma desvantagem dessa abordagem é que quanto mais abstrato for o modelo, mais complexo é o tratamento analítico. As conseqüências disso, são refletidas no grau de precisão da avaliação que deve ser menor em comparação às outras técnicas.

3.5.3 Simulação

A simulação consiste em uma técnica na qual a implementação de programas objetiva imitar a execução dos sistemas reais. Nesse sentido, a exemplo da simulação, a análise de desempenho feita é a do modelo e não do sistema real [MEI02].

O propósito da simulação é fazer um prognóstico de quais seriam as medidas se o sistema simulado fosse medido diretamente. Assim, os resultados de uma avaliação obtidos através desta técnica serão muito dependentes do modelo definido para o sistema.

Essa técnica requer um conhecimento muito detalhado do sistema [DEL02]. O que significa dizer que não é uma tarefa fácil para o avaliador desenvolver uma análise de desempenho por meio dos recursos de simulação. Além disso, os modelos são caros e demandam muito tempo para se construir. Não obstante, a facilidade de mudanças nas configurações é às vezes a razão da escolha da simulação para sistemas dispendiosos.

Uma vantagem segundo [VIC98] da técnica de simulação se refere à possibilidade de se obter respostas para diferentes problemas, inclusive para os de grande complexidade. No entanto, as simulações necessitam validação para demonstrar que os modelos são representações acuradas dos sistemas reais simulados.

3.6 Caracterização da Carga de Trabalho

Entende-se por carga de trabalho submetida a um sistema computacional, o processamento de todas as requisições submetidas ao sistema, pelo usuário, num determinado intervalo de tempo. Em servidores *Web*, por exemplo, equivale ao conjunto de requisições HTTP recebidas pelo sistema durante um dado período de observação.

A carga de trabalho de um sistema computacional é definida em função de parâmetros que permitam o desenvolvimento da avaliação de desempenho deste sistema. Parâmetros como taxa de chegada de requisições HTTP, número de usuários conectados simultaneamente e tamanho médio de documentos requisitados ao sistema são úteis na definição daquilo que denominamos caracterização da carga de trabalho de um sistema ou, mais especificamente, de um servidor *Web*.

A caracterização da carga de trabalho de um sistema é atividade essencial nos problemas de avaliação de desempenho, pois além de permitir um entendimento detalhado da carga submetida ao sistema, representa também o ponto de partida para a geração da carga de trabalho para uma avaliação de desempenho do mesmo[MEI02].

A Tabela 1 apresenta parâmetros que permitem fazer a caracterização da carga de trabalho de um sistema computacional [FER83] apud [VIC98].

Tabela 1: Parâmetros da carga de trabalho

Parâmetro	Descrição
Tempo de CPU	Tempo total de CPU consumido por um programa
Operações de E/S	Número total de operações de E/S executadas
Espaço de memória	Espaço de memória demandado pelo programa
Prioridade externa	Prioridade de escalonamento atribuída ao usuário
Taxa de chegada	Tempo entre a chegada de duas requisições sucessivas para o recurso sendo considerado
Distribuição do tempo entre chegadas	Distribuição dos tempos entre a chegada de duas requisições consecutivas para o recurso sendo considerado
Tempo de espera do usuário	Tempo entre a chegada de uma mensagem do sistema no terminal e o início da entrada do próximo comando do usuário
Intensidade de requisições	Razão entre o tempo de resposta e o tempo de espera do usuário
Intensidade do tráfego	Razão entre o tempo de serviço por transação e tempo de chegada de comandos
Número de usuários	Número de usuários conectados simultaneamente ao sistema

Uma fonte de dados importante para caracterização da carga de trabalho de servidores *Web* são arquivos de registro de acesso ao servidor (arquivos de *log*). A carga de trabalho usada para avaliar o desempenho do servidor *web* Apache descrita no capítulo 5 deste trabalho tem como fonte estes dados.

3.7 Avaliação de Desempenho de Servidores *Web*

Os servidores *Web* são componentes fundamentais de sistemas baseados na *Web* [MEN00]. O crescimento do comércio eletrônico e da telefonia com o uso da *Web*, trouxe a *Internet* para o núcleo das telecomunicações tradicionais, tornando os servidores *Web* elementos chave também para as futuras redes de comunicação, embora não concebidos originariamente para isso. No entanto, os usuários desses sistemas desejam acesso rápido e fácil aos documentos armazenados. Nesse sentido, a atividade de avaliação de desempenho de servidores *Web* é imprescindível.

O desempenho de um servidor *Web* segundo Murta et al.[MUR96] depende de quatro fatores: o ambiente de *software* (sistema operacional, subsistema de gerência de arquivo, sistema gerenciador de banco de dados, entre outros), a plataforma de *hardware*, o *daemon* HTTP e a carga de trabalho. A análise feita aqui será restrita à carga de trabalho, que corresponde ao conjunto de requisições HTTP submetidas ao servidor *Web*.

No que se refere à carga, os diversos trabalhos pesquisados [MIL00, MEI02, MEN98, MUR96], utilizam para fins de avaliação de desempenho, cargas de teste sintética, ou seja, cargas que podem consistir de um subconjunto de componentes do mundo real ou uma mistura de componentes da carga de trabalho real e componentes construídos propositadamente para o teste. Neste aspecto, a avaliação de desempenho realizada no capítulo 5 deste trabalho difere dos mencionados acima, pois consiste de uma carga de teste real (programas originais e dados processados num dado intervalo de tempo).

Outro ponto no qual este estudo se distingue daqueles, é com relação aos dados de execução do sistema. Enquanto os trabalhos pesquisados sobre análise de desempenho tinham como objetivo principal analisar a saturação da capacidade de atendimento de servidores *Web*, pretende-se aqui obter um perfil de desempenho do servidor *Web* e propor uma alternativa no nível de previsão de desempenho, através do uso de séries temporais. Conceitos de série temporais serão discutidos no capítulo 4.

3.8 Conclusões

Neste capítulo, foram abordadas diversas questões sobre avaliação de desempenho de sistemas computacionais de um modo geral. Entre essas questões foi apresentada uma metodologia para uma adequada avaliação de desempenho. Foram abordadas também as diversas métricas e técnicas de avaliação. A escolha de uma métrica ou conjunto de métricas para avaliar o desempenho de um sistema, depende entre outros fatores, dos objetivos a serem atingidos e das características do sistema.

As várias técnicas de avaliação, por sua vez se destinam a resolver problemas específicos de desempenho. Enumeramos vantagens e desvantagens dessa técnica. Contudo a seleção de uma ou outra técnica pode depender apenas de uma questão de diminuição dos custos de avaliação.

Finalmente, discutimos a caracterização da carga de trabalho, uma atividade extremamente importante no processo de avaliação, tendo em vista que dificilmente são resolvidos os problemas de avaliação sem que se tenha um conhecimento da carga de trabalho a ser processada pelo sistema.

CAPÍTULO 4

Conceitos e Aplicações de Séries Temporais

4.1 Introdução

Neste capítulo, são abordadas as séries temporais ou séries de tempo com o intuito de prover embasamento para o entendimento do capítulo 5, no qual conceitos e aplicações de séries temporais são utilizados para auxiliar no processo de avaliação de desempenho de sistemas computacionais. Nesse sentido este capítulo, está organizado da seguinte forma. As seções que vão de 4.2 a 4.5 apresentam os conceitos, os componentes e os objetivos da análise de uma série temporal. Na seção 4.6 são abordados alguns métodos de suavizamento exponencial para previsão de séries de tempo, com ênfase no método de suavizamento exponencial simples. Finalmente, a seção 4.7 apresenta aplicações de séries de tempo, e em especial no suporte às soluções dos problemas de previsão de desempenho de sistemas de computadores.

4.2 Séries Temporais

Chama-se série temporal um conjunto de observações ordenadas no tempo. Estas podem ser contínuas ou discretas. Se os dados são observados de forma contínua, tem-se o que se chama de série temporal contínua. Os eletrocardiogramas são exemplos deste tipo de série. Por outro lado, quando os dados são observados discretamente no tempo, as séries são denominadas discretas. Como exemplos, poderíamos considerar os índices diários da bolsa de São Paulo ou a quantidade anual de chuvas na cidade de Recife.

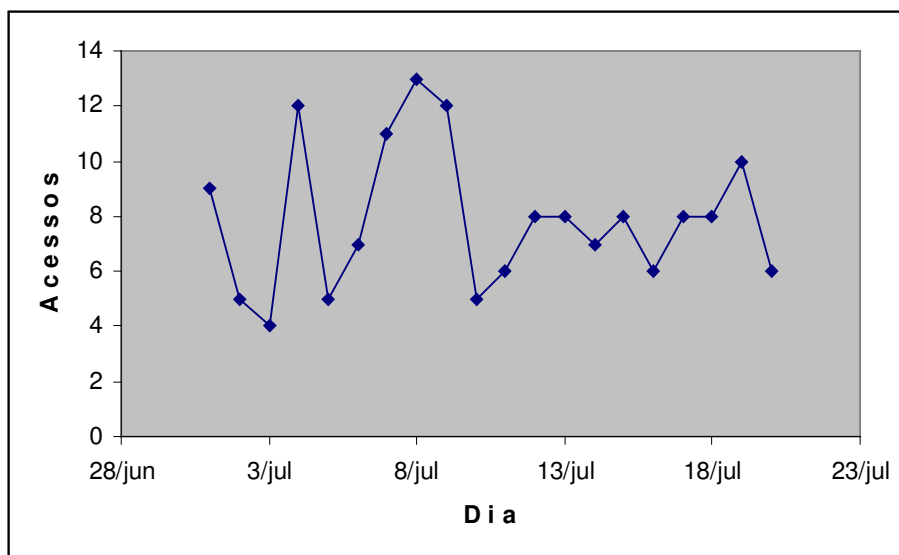
Neste trabalho, trataremos apenas de séries discretas, particularmente do conjunto de observações resultantes do número diário de requisições HTTP feitas a um servidor WWW num período de tempo especificado e da série formada pela quantidade de bytes transmitida por esse servidor no mesmo período.

Para se entender séries temporais, considera-se a Tabela 2, com dados sobre o número diário de visitantes únicos do *site Jornal Local* do Servidor WWW Apache, obtidos através de um contador de acessos. Visitantes únicos são aqueles que visitaram o *site* independentemente da quantidade de sessões que iniciaram.

Tabela 2: Distribuição do número diário de visitantes únicos ao *site Jornal Local*

Dia	Requisições ao Site Jornal Local	Dia	Requisições ao Site Jornal Local
01/07	09	11/07	06
02/07	05	12/07	08
03/07	04	13/07	08
04/07	12	14/07	07
05/07	05	15/07	08
06/07	07	16/07	06
07/07	11	17/07	08
08/07	13	18/07	08
09/07	12	19/07	10
10/07	05	20/07	06

Seja designado por Y_i o número de acessos ao *site* Jornal local no instante i . O conjunto de pontos ilustrado no Gráfico 1 é uma trajetória do processo que está sendo observado; no eixo das abcissas tem-se os dias, e no eixo das ordenadas, o número de acessos. Outras trajetórias poderiam ser obtidas para cada um dos conjuntos de observações de um determinado *site* em particular. O conjunto de todas as possíveis trajetórias que se poderia observar é o que se denomina processo estocástico. Neste caso, uma série temporal seria o conjunto de observações disponíveis para análise, o que na realidade equivale apenas a uma parte de uma trajetória, dentre outras que poderiam ter sido observadas [MOR87]. Pode-se considerar então, que uma série temporal é uma realização de um processo estocástico.

Gráfico 1: Requisições diárias ao *site* Jornal Local do Provedor FlorianoNet

4.3 Objetivos da Análise de Séries Temporais

Na análise de uma série temporal pressupõe-se que os fatores que tiveram influência nos padrões da atividade no passado e no presente continuarão de certa forma, a fazê-lo no futuro. Nesse sentido, pode-se levar em consideração diversos objetivos. No entanto, alguns são básicos [MOR85, LEV00]:

- modelagem do fenômeno sob consideração;
- obtenção de conclusões em termos estatísticos;
- identificar e isolar os fatores que influenciaram os padrões da atividade para fins de previsão, como também para o planejamento e controle gerencial.
- modelar o comportamento da série e neste caso a construção do gráfico, a verificação da existência de tendência, ciclos e variações sazonais, a construção de histogramas e diagramas de dispersão, entre outros, podem ser ferramentas úteis;
- investigar o mecanismo gerador da série temporal; por exemplo, na análise de tráfego autosimilar, pode-se ter interesse em descobrir como o tráfego foi gerado.

Dois dos objetivos acima o terceiro e o quarto são extremamente importantes para as séries analisadas aqui. Um por se referir em termos de previsibilidade de valores futuros (uma vez que é interessante neste trabalho fazer um prognóstico para um dado período no futuro, como discutiremos com detalhes no capítulo 5). O outro por considerar as informações que possam ser obtidas dos gráficos, que se caracterizam como ferramentas bastante úteis para a descrição do comportamento da série observada, bem como na apresentação de um sumário de dados.

4.4 Componentes de uma Série de Tempo

Numa série de tempo Y_i , $i = 1, \dots, n$, pode-se distinguir de maneira geral, as seguintes características: uma tendência, variações sazonais ou estacionais e variações aleatórias [AND71, KEN76]. Dessa forma, a série pode ser escrita como:

$$Y_i = T_i + S_i + a_i, \quad i = 1, \dots, n \quad (4.1)$$

onde T_i é a tendência, S_i é a componente sazonal e a_i é o termo aleatório.

A tendência permite entender o comportamento do movimento da série; se crescente, decrescente, linear, periódico ou exponencial, entre outros. A componente sazonal está relacionada, por exemplo, às épocas do ano, isto é, as observações são registradas mensalmente, semanalmente ou diariamente. E finalmente as variações aleatórias que aparecem na série como consequência de efeitos que não podem ser explicados e não obedecem a qualquer regularidade, tais como: problemas políticos, terremotos, inundações ou até mesmo erro de medida. Dessa forma, fazer uma análise de uma série de tempo é, entre outras coisas, procurar isolar e identificar suas componentes, para fins de previsão [LEV00].

Um dos aspectos de fundamental importância no início da análise de uma série temporal é verificar o comportamento do modelo com relação às suas componentes. O modelo (4.1), é denominado aditivo. O modelo multiplicativo por sua vez é dado pelo produto *tendência* $\times$ *sazonalidade* $\times$ *variações aleatórias*, ou seja,

$$Y_i = T_i \times S_i \times a_i, \quad i = 1, \dots, n \quad (4.2)$$

Há também os modelos mistos, como

$$Y_i = T_i S_i + a_i, \quad i = 1, \dots, n \quad (4.3)$$

Neste trabalho, i será sempre o tempo e restringido às séries de tempo onde $i = 1, \dots, n$.

Contudo, um problema na análise de série de tempo está em modelar de forma conveniente suas componentes T_i , S_i e a_i de forma a se fazer previsões futuras para a série. Uma solução segundo Morettin [MOR85] é representar $f(i) = T_i + S_i$ por uma função suave de tempo, que pode ser uma combinação de polinômios e funções trigonométricas.

4.5 Métodos de Estimação da Tendência

Dentre os métodos de estimação da tendência, existem os chamados métodos de ajuste funcional ou modelagem analítica, e os métodos não paramétricos (técnicas de suavização). Modelagem analítica no sentido de se adequar uma função matemática que melhor represente o fenômeno cuja realização particular é a série de tempo observada. Tradicionalmente são utilizados polinômios, funções exponenciais ou outras funções de tempo.

As técnicas de suavização diferem das anteriores porque a hipótese de suavização permite usar somente observações próximas a um dado ponto no tempo de forma que se possa estimar a tendência naquele ponto, enquanto que para a hipótese de tendência polinomial (em sentido amplo), todas as observações são usadas para estimar o polinômio que representa a tendência sobre todo o intervalo inteiro [AND71, KEN76].

Ao se suavizar uma série de tempo, se quer representar a tendência num ponto como uma média ponderada dos valores próximos do ponto, para toda série, ou seja, um estudo do comportamento local da série. Isto ocorre, quando não for conveniente o ajuste funcional da tendência ao conjunto de dados numéricos obtidos, ou quando não há necessidade de previsão, desejando-se apenas conhecer o valor da tendência em algum ponto.

Serão enfatizados aqui aos métodos de suavização ou suavizamento, ou mais especificamente aos de suavizamento exponencial que serão discutidos na seção 4.6.

4.6 Métodos de Suavizamento

Dentre os métodos de suavizamento podem ser destacados os métodos de suavizamento exponencial. Esses métodos têm sido muito utilizados, devido à simplicidade, à eficiência computacional e também à sua razoável precisão [MOR87]. Uma atenção especial será dada ao método de suavizamento exponencial simples ou alisamento exponencial simples (AES). É importante ressaltar que outros métodos não foram utilizados para fins de aplicação neste trabalho, e que a escolha do AES se deu por conta do fácil entendimento e simplicidade, conseqüentemente uma boa opção para o início do desenvolvimento do trabalho.

Em seguida faz-se uma exposição sobre alguns desses métodos, os quais são muito eficazes quando a série de tempo tem um número pequeno de observações, se deseja fazer previsões em curto prazo e se trabalha com várias séries ao mesmo tempo.

4.6.1 Suavizamento Exponencial Simples

Considere-se, o caso de uma série de tempo $Y_1, \dots, Y_n$, que em determinado instante é composta de nível (parâmetro desconhecido) e um termo aleatório (componente irregular), isto é,

$$Y_i = \mu_i + a_i, \quad i = 1, \dots, n \quad (4.4)$$

onde a_i tem média igual a zero e variância constante e μ_i é um parâmetro desconhecido.

O suavizamento exponencial simples é um método que pode ser utilizado tanto para ajustar uma série temporal, e conseqüentemente ter uma idéia sobre o comportamento geral da série, como para obter previsões de curto prazo (um período no futuro) para séries temporais nas quais seja questionável o tipo de efeito de tendência de longo prazo, caso exista. Para este trabalho, o último contexto é especial, pois, como dito anteriormente objetiva-se fazer prognósticos de valores futuros de séries de tempo.

No suavizamento exponencial simples é empregada uma média ponderada com pesos que decaem exponencialmente tal que, para as observações mais próximas é dado maior peso, e menor peso para as observações passadas. Neste método, cada previsão ou cálculo de ajuste [LEV00] depende de todos os valores previamente observados.

O suavizamento exponencial simples pode ser descrito matematicamente por:

$$\bar{Y}_i = K Y_i + (1 - K) \bar{Y}_{i-1}, \quad \bar{Y}_0 = Y_1, \quad i = 1, \dots, n \quad (4.5)$$

onde $\bar{Y}_i$, é o valor da série exponencialmente suavizada, calculado no período de tempo i , $\bar{Y}_{i-1}$, é o valor da série exponencialmente suavizada, calculada no período de tempo $i-1$, Y_i , é o valor da série de tempo no tempo i e K é a constante de suavizamento ou peso, $0 \leq K \leq 1$.

A equação (4.5) é equivalente a:

$$\bar{Y}_i = K \sum_{w=0}^{i-1} (1-K)^w Y_{i-w} + (1-K)^i \bar{Y}_0, \quad i = 1, \dots, n \quad (4.6)$$

Se for expandida a equação (4.6), tem-se que

$$\bar{Y}_i = K Y_i + K (1-K) Y_{i-1} + K (1-K)^2 Y_{i-2} + \dots \quad (4.7)$$

O último valor exponencialmente suavizado, representa a previsão de todos os valores futuros da série temporal, isto é,

$$\hat{Y}_i(h) = \overline{Y}_i, \quad \forall h > 0 \quad (4.8)$$

ou

$$\hat{Y}_i(h) = K Y_i + (1 - K) \hat{Y}_{i-1}(h+1) \quad (4.9)$$

Visto que tem-se uma nova observação, (4.9) pode ser interpretada como uma equação que permite uma atualização de previsão [Mor87].

Particularmente, se o modelo (4.4) apresentar média global constante, isto é, $\mu_i = \mu$, para todo i , demonstra-se que, para $t \rightarrow \infty$, o cálculo da média e da variância, pode ser dado por

$$E = \left[\hat{Y}_i(h) \right] = \mu \quad (4.10)$$

e

$$\text{Var} = \left[\hat{Y}_i(h) \right] = \frac{K}{2 - K} \sigma_a^2, \quad (4.11)$$

onde σ_a^2 é a variância de uma seqüência de variáveis aleatórias independentes, não correlacionadas de média zero e variância constante (σ_a^2). .

4.6.1.1 Vantagens e Desvantagens do Método de Suavizamento Exponencial Simples

A grande popularidade do método de suavizamento exponencial simples, ocorre justamente por apresentar vantagens como[MOR85]:

- Fácil entendimento.
- Aplicação não dispendiosa.
- Alta flexibilidade devido à variação da constante de suavizamento.
- Possibilidade de armazenar apenas $Y_i, \overline{Y}_i$ e K .

A principal desvantagem consiste na dificuldade em determinar um valor mais apropriado para a constante de suavizamento, a qual tem fundamental importância dentro da série, uma vez que irá afetar os resultados da série suavizada.

4.7 Aplicações de Séries Temporais

Nesta seção, não se pretende mostrar como se aplicam os métodos de previsão de séries temporais para fins de prognóstico de valores futuros de uma ou outra série de tempo a ser considerada. O propósito é, na verdade, enumerar algumas situações onde a análise de séries temporais é interessante, especificamente no que diz respeito ao uso da previsão como auxílio no planejamento de necessidades operacionais futuras.

4.7.1 Nos Negócios e na Economia

A previsão é uma técnica bastante útil nos negócios e na economia, uma vez que gerentes e pessoas responsáveis por analisar e manipular dados econômicos, necessitam de alguma forma de preverem eventos futuros que auxiliem no processo de tomada de decisão. Existem vários métodos de previsão, no entanto a discussão será focada nos métodos de previsão de séries temporais, que são compreendidos como métodos quantitativos.

Como exemplo de séries temporais nos negócios pode-se considerar as vendas anuais das fábricas em todo o mundo, de carros, caminhões e ônibus fabricados pela General Motors Corporation (GM), durante o período de 23 anos, de 1970 a 1992 [LEV00]. A Tabela 3 apresenta os valores para essas vendas (em milhões de unidades). Esta série, assim como todas as analisadas neste trabalho segue o modelo multiplicativo de séries temporais abordado na seção 4.4.

Tabela 3: Vendas das fábricas da GM (em milhões de unidades)

Ano	Vendas das Fábricas	Ano	Vendas das Fábricas
1970	5,3	1982	6,2
1971	7,8	1983	7,8
1972	7,8	1984	8,3
1973	8,7	1985	9,3
1974	6,7	1986	8,6
1975	6,6	1987	7,8
1976	8,6	1988	8,1
1977	9,1	1989	7,9
1978	9,5	1990	7,5
1979	9,0	1991	7,0
1980	7,1	1992	7,2
1981	6,8		

Fonte: Moody's Handbook of Common Stocks, 1980, 1989, 1993.

A construção do gráfico dos dados e a observação de sua tendência ao longo do tempo representa o primeiro passo na análise de séries temporais. No entanto, em casos como esses, avaliar essas tendências pode não ser uma tarefa fácil em função da quantidade de variações de ano para ano. O Gráfico 2 representa a série temporal desses dados.

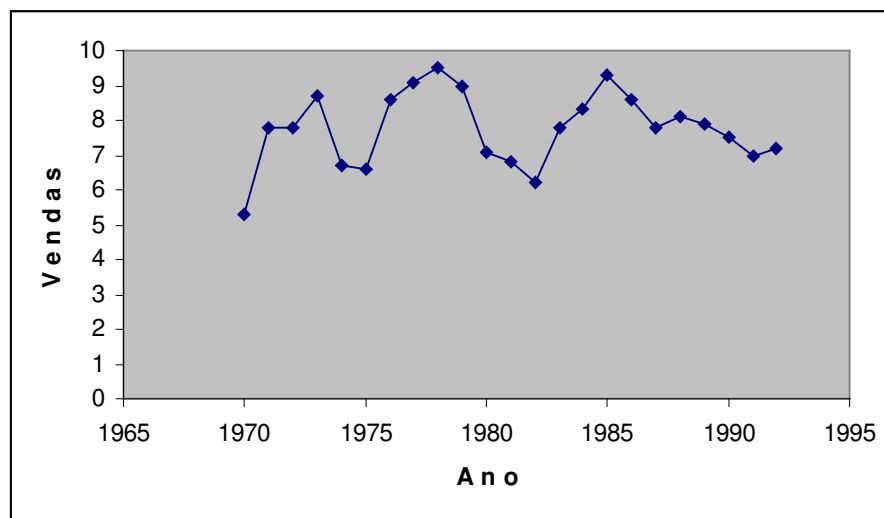


Gráfico 2: Vendas das fábricas da GM (em milhões de unidades)

Tendo em vista isso, pode-se utilizar o método de suavizamento exponencial simples apresentado na seção 4.6.2 para ajustar esta ou até mesmo para fazer um prognóstico do número de unidades vendidas por todas as fábricas da *General Motors Corporation* durante o ano de 1993.

4.7.2 Na Avaliação de Desempenho de Sistemas Computacionais

Os servidores *Web* são componentes fundamentais de sistemas baseados na *Web*. Há dois modos para se investigar o desempenho desses servidores, usando cargas reais ou sintéticas. O crescimento do comércio e da telefonia com o uso da *Web* trouxe a *Internet* para o núcleo das telecomunicações tradicionais, tornando os servidores *Web* elementos chave também para as futuras redes de comunicação, embora não concebidos originariamente para isso.

Os modelos de desempenho ajudam a compreender o comportamento quantitativo desses sistemas servidores, em cenários distintos, possibilitando a previsão do desempenho, e o processo da estimação de medidas de desempenho (comumente taxa de processamento e tempo de resposta), para um determinado conjunto de parâmetros; esses parâmetros podem ser relativos ao próprio sistema (protocolos), aos recursos (latências) e à carga de trabalho. É no aspecto prognóstico, que os métodos de previsão de séries temporais podem contribuir para o processo de avaliação de sistemas computacionais.

Na seção 4.2 foi apresentada uma série formada pelo número diário de visitantes únicos ao *site* **Jornal Local** do Servidor *WWW* do provedor FlorianoNet. Uma análise de séries como as formadas pelos números diários de requisições *HTTP* submetidas ao Servidor *Web* Apache pode ser útil no sentido de auxiliar na avaliação de desempenho do servidor *Web*. Uma outra situação em que é adequada a aplicação de série temporais, envolve as corporações que se utilizam do comércio eletrônico, por exemplo. Neste caso, elas necessitam obter informações acerca do número de usuários que estão visitando ou possam visitar seus *sites Web* num dado período no futuro. Na seção 5.8.1 é feita uma análise mais detalhada de uma série composta pelo número de visitantes diários a um *site Web* cujas requisições *HTTP* são dinâmicas e incluem acesso a um banco de dados.

4.8 Conclusões

Neste capítulo foram abordados conceitos básicos de séries temporais e em particular foi dada uma ênfase à aplicação do método de suavizamento exponencial simples para fins de previsão de séries de tempo formadas por requisições a um *site Web* específico ou à quantidade diária de requisições *HTTP* atendidas com sucesso por um servidor *Web*.

A seguir, no capítulo 5 faz-se uma avaliação do desempenho do servidor *Web* Apache instalado no provedor FlorianoNet, de maneira a obtermos um perfil de desempenho do

sistema. Considerando os dois tipos de séries de tempo exemplificadas logo acima, além das séries formadas pela taxa de processamento diário de *bytes* pelo servidor a fim de se fazer um prognóstico de execução do mesmo.

CAPÍTULO 5

Avaliação de Desempenho de um Servidor

Web

5.1 Introdução

A *Word Wide Web* (WWW), ou simplesmente *Web*, tem apresentado nos últimos anos um enorme crescimento. Segundo dados do Ibope e Ratings.com [IBRC02], o número de usuários residenciais da Web no Brasil cresceu, e em fevereiro de 2002 esta marca já era superior a 13 milhões. Um dos motivos desse crescimento é que a *Web* tem se tornado ultimamente o local de preferência universal para publicação de dados eletrônicos. Contudo, um problema relacionado ao ambiente *Web* diz respeito ao desempenho do servidor WWW, especialmente quando as requisições submetidas a esse sistema são dinâmicas.

Neste capítulo faz-se uma avaliação do desempenho do servidor *Web* Apache, o servidor *Web* mais utilizado no mundo com cerca de 60% dos *sites Web* usados na *Internet* [APA02]. Para fins da avaliação desenvolvida neste trabalho será utilizada uma implementação deste servidor no provedor de acesso à *Internet* FlorianoNet. A metodologia segue as etapas de um processo de avaliação de desempenho descrito em [JAI91] e [MEI02].

Os diversos trabalhos sobre análise de desempenho pesquisados tinham como objetivo principal analisar a saturação da capacidade de atendimento de servidores *Web*, identificando, por exemplo, o número máximo de clientes que o sistema pode processar [MUR96, MEN98, MEI02]. Neste estudo tem-se como interesse obter um perfil de desempenho do servidor *Web* e propor uma alternativa no nível de previsão de desempenho tendo como enfoque, ao contrário dos trabalhos citados acima, o ambiente corporativo real.

Este capítulo está organizado em 10 seções das quais esta é a primeira. A segunda apresenta informações gerais sobre o provedor de acesso à *Internet* FlorianoNet. A seção seguinte estabelece os objetivos da avaliação. As seções 5.4, 5.5 e 5.6 especificam respectivamente, as métricas de interesse na avaliação, a carga de trabalho do servidor e a instrumentação utilizada. Nas seções 5.7 e 5.8 discute-se, nessa ordem, alguns indicadores estatísticos de acesso ao servidor *Web* incluindo o uso de séries de tempo na previsão de desempenho desse servidor. A seção 5.9 traz uma análise dos resultados dos experimentos e a seção 5.10 as conclusões.

5.2 O Provedor de Acesso à *Internet* FlorianoNet

Este provedor está instalado na cidade de Floriano na região sul do Piauí e atende a diversas outras cidades do estado, inclusive a capital Teresina. O FlorianoNet, além de prover acesso à *Internet*, também mantém um banco de dados, no qual são armazenadas informações de seus clientes.

A implementação avaliada do provedor é composta de três componentes de *software*:

- Apache- *software* para servidor *Web* que implementa o protocolo HTTP/1.0 e roda sobre o sistema operacional FreeBSD, em uma máquina Pentium 400 Mhz, com 64 Mbytes de memória e 256 Kbytes de *cache*;
- Aplicação- uma implementação em PHP (*Hypertext Preprocessor*) que provê as várias funcionalidades da lógica da aplicação;
- MySQL- Sistema Gerenciador de Banco de Dados de domínio público e que é utilizado como servidor de dados.

A aplicação é composta de *scripts* em PHP feitos através de comandos do SGBD MySQL. Isto é possível porque PHP possui funções nativas de MySQL. O PHP é uma linguagem interpretada tradicionalmente utilizada para a criação de *scripts* do lado servidor [KAB02]. O MySQL, por sua vez é a plataforma de banco de dados mais extensamente usada para PHP [MAS01]. Para acessar um banco MySQL é preciso um nome de usuário e uma senha que MySQL aceite para o acesso desejado. Com os *scripts* PHP, este acesso é possível após uma conexão com validação de usuário. Feita a conexão, a manipulação do banco de dados ocorre por meio de comandos SQL, como SELECT, INSERT, UPDATE e DELETE, entre outros.

No banco de dados são permitidas operações como inclusão (cadastro de clientes), exclusão e consulta de informações dos diversos clientes. Tais operações são realizadas pela equipe de administração do banco de dados diretamente pela *Web*, caracterizando-se, portanto, em requisições dinâmicas ao servidor *Web* do provedor. Nesse sentido, ele apresenta características de um servidor de comércio eletrônico, embora não possa ser classificado dessa maneira. Contudo, o importante para a análise desse trabalho é que uma parte das requisições HTTP submetidas ao servidor inclui acesso a um servidor de banco de dados.

5.3 Objetivos da Avaliação de Desempenho

Na avaliação de desempenho do servidor *Web* é interessante determinar como o sistema se comporta em função das requisições dos clientes. Nesse sentido, fazem parte dessa avaliação:

- obter um perfil de execução do sistema, através da taxa de processamento ou taxa de serviço e do tempo de resposta do servidor a uma requisição de cliente;
- identificar fatores relacionados à interação do cliente com o servidor que são relevantes para caracterizar a carga de trabalho do servidor *Web*;
- fazer uma previsão do número de requisições que o servidor pode atender diariamente e da taxa de transferência de bytes em Mbytes por dia.

5.4 Métricas de Interesse

Como mencionado anteriormente, as métricas especializadas podem ser utilizadas para quantificar o desempenho de um servidor de comércio eletrônico. Uma vez que o provedor de acesso à *Internet* implementa esse servidor com características próprias de comércio eletrônico, pode-se utilizar métricas especializadas como a taxa de processamento e o tempo de resposta do servidor para uma requisição de cliente a fim de obter um perfil de desempenho de um de seus componentes, no caso o servidor *Web*. A seguir são apresentadas essas duas métricas, as quais serão utilizadas na avaliação de desempenho proposta neste trabalho.

5.4.1 Taxa de Processamento

Sendo a taxa de serviço ou processamento uma medida de produtividade, sua obtenção foi em função da quantidade de trabalho executado pelo servidor *Web* no período de 06/01/2002 a 02/08/2002, totalizando 208 dias. Esta medida permitiu quantificar o número de requisições HTTP atendidas por dia, bem como a taxa total de dados que o servidor produziu, medida em *Mbytes* por dia.

Deve-se ressaltar que a maioria das requisições feitas ao servidor *Web* são estáticas, ou seja, são requisições para a exibição de arquivos estáticos., ou requisições dinâmicas que

exigem do servidor um processamento local devido à execução de aplicações CGI [MUR96]. No entanto, das requisições dinâmicas submetidas a este servidor várias incluem acesso a um Sistema Gerenciador de Banco de Dados MYSQL, o que pode, neste caso, degradar ainda mais o desempenho do servidor *Web*, tendo em vista que isto consome, normalmente, mais recursos e tempo de processamento se comparado à simples exibição da página inicial de um *site*.

5.4.2 Tempo de Resposta

Anteriormente, foi visto que determinar o tempo de resposta percebido pelo usuário em uma requisição HTTP não é uma tarefa trivial, uma vez que se precisa conhecer o tempo de resposta do servidor, além dos atrasos no cliente e na rede que interliga cliente e servidor. Adicionalmente, as requisições dinâmicas a um servidor podem exigir acesso a um banco de dados, o que torna a tarefa ainda mais complexa.

Dada a variedade de *software* que compõe o provedor de acesso à *Internet* FlorianoNet, foi considerado apenas o tempo de execução do servidor *Web*. O tempo de resposta do servidor *Web* corresponde ao intervalo de tempo entre o início e o fim da execução de uma requisição HTTP [MEN00]. No entanto, para efeito da avaliação de desempenho desenvolvida neste trabalho, esse tempo equivale à latência média $\bar{L}_{\text{http}}$, calculada pela expressão seguir, onde *tempo_acumulado* é o tempo decorrido para o atendimento das requisições e *número_total_de_requisições_atendidas* é quantidade de requisições atendidas com sucesso pelo servidor *Web*.

$$\bar{L}_{\text{http}} = \frac{\text{tempo_acumulado}}{\text{número_total_de_requisições_atendidas}}.$$

5.5 Carga de Trabalho do Servidor *Web*

Um dos objetivos deste trabalho é obter um perfil de desempenho do servidor *Web*, a fim de saber, entre outras coisas, se sua capacidade é adequada. Neste sentido, a caracterização da carga de trabalho é de fundamental importância.

A carga de trabalho a ser utilizada para avaliação de desempenho do servidor *Web* do provedor FlorianoNet é resultado de uma análise do arquivo de registros de acesso do servidor (arquivos de *log*). Cada registro de acesso contém as seguintes informações:

- data e hora do acesso;
- identificador de sessão;
- identificador do tipo de requisição ;
- parâmetros de requisição.

A Tabela 4 apresenta a caracterização da carga de trabalho do sistema, que por sua vez consiste de todos os programas originais e dados processados no período de monitoração.

Tabela 4: Caracterização da carga de trabalho do servidor Apache

Características	Carga de Teste Real
Quantidade de arquivos	11166
Tamanho total de arquivos	22332 Kbytes
Tamanho médio dos arquivos	2 Kbytes
Número de requisições	414
HTML ou imagem	90 %
Operação de acesso a Banco de Dados	4 %

A carga de trabalho de servidores *Web* tradicionais pode ser caracterizada pelo processo de chegada de requisições e por características específicas das requisições [MEI02]. Duas das principais características, as que se referem ao tamanho dos arquivos e ao tipo de informação requisitada, são de fundamental importância, considerando-se que parte das requisições ao servidor *Web* do provedor FlorianoNet são dinâmicas.

5.6 Instrumentação

O sistema a ser avaliado neste trabalho se encontra-se em plena atividade, e por se tratar de uma implementação real, a técnica de avaliação de desempenho aplicada é a monitoração (mensuração). Neste caso, se quer saber se o sistema está de fato desempenhando suas funções diante da carga de trabalho ao qual ele está sendo submetido.

Por uma questão de flexibilidade e facilidade de implementação, foi utilizada uma monitoração baseada em *softwares*. Monitores baseados em *softwares* consistem em uma coleção de rotinas que, inseridas na aplicação, permitem coletar dados de desempenho [JAI 91]. Essas rotinas podem ser inseridas automaticamente ou manualmente por um programador.

Para fins da referida avaliação, os dados de desempenho foram coletados a partir dos arquivos de registro de acesso do servidor *Web* Apache. Manualmente foram inseridas algumas rotinas nos códigos de *sites* específicos do servidor com o objetivo de se obter também dados de utilização do sistema, e assim auxiliar na obtenção de um perfil de desempenho.

Apesar da monitoração por *software* ser considerada como uma técnica que possibilita obter resultados que apresentem um grau de confiança maior, é preciso que consideremos na nossa análise o seu caráter intrusivo pois, segundo [JAI91], este tipo de monitoração pode aumentar de forma significativa o tempo de execução de uma aplicação, alterar o seu comportamento de sincronização e produzir tempos diferentes dos medidos em um sistema que não esteja sendo monitorado.

5.6.1 Estratégia de Aquisição dos Perfis de Desempenho

Quando se monitora um sistema para efeito de avaliação de desempenho, obtém-se um perfil de execução do mesmo [MEI02]. Um perfil pode ser global ou local. Tendo em vista que o provedor FlorianoNet possui diversos componentes e que um dos objetivos deste trabalho é avaliar o componente servidor *Web*, deve-se obter um perfil local, de forma que isso possa ser útil para o entendimento de desempenho do sistema como um todo. De outro modo, a geração de um perfil local pode ser utilizada para a construção de um perfil global do sistema. No entanto, há necessidade de que estratégias para coletar estes perfis sejam definidas. As estratégias utilizadas foram a contagem e a análise dos arquivos de registro de acesso. Na sequência, ambas são discutidas com mais detalhes.

Na seção anterior, falava-se de rotinas que foram inseridas manualmente no código da aplicação. Neste procedimento foi utilizada a estratégia de contagem, que em geral é implementada através de código inserido na aplicação. A contagem do evento sumariza o número de ocorrências de um conjunto de eventos durante a execução da aplicação que está sendo monitorada [MEN00]. No nosso caso, o contador acessos [ACS02] inseridos em 6(seis) *sites* do servidor *Web* nos permite conhecer, por exemplo, o número diário de visitantes únicos de cada um destes *sites*. A instrumentação que permite conhecer o número de visitantes únicos em um determinado período, é feita pelo contador Acessos através da inserção de um código escrito na linguagem de programação JavaScript (ver Anexo 1).

A ferramenta *Advanced Log Analyzer* [ALA02] foi utilizada nos experimentos para analisar os arquivos de registros do servidor. Essa ferramenta permitiu a determinação de dados de desempenho como:

- Taxa de serviço, medida pelo número de requisições HTTP atendidas por dia e pela quantidade de bytes transferidos por dia (*Mbytes/dia*).
- Tempo de resposta médio do servidor para uma requisição de usuário.

Além das informações acerca do desempenho do servidor, a ferramenta *Advanced Log Analyzer* também torna disponível dados como páginas mais requisitadas, tipo de arquivos mais requisitados, número de *downloads* por dia, entre outros. A Tabela 5 mostra o número e o percentual de oito dos tipos de arquivo mais requisitados, segundo à extensão, para o *site* GurguéiaNet do provedor FlorianoNet.

Tabela 5: Número e percentual classificados segundo à extensão

Extensão de Arquivo	Número	Percentual
.gif	343370	89,51%
.jpg	17670	4,55%
.php	8338	2,15%
.swf	6933	1,79%
.php3	3275	0,84%
.htm	2727	0,70%
.js	1298	0,33%
.html	480	0,12%

De acordo com a tabela acima, o parâmetro extensão de arquivo dá uma idéia real da carga de trabalho imposta ao servidor (Tabela 4), onde 90% da requisições submetidas são para arquivos do tipo imagem ou HTML e aproximadamente 4% das requisições feitas incluem acesso ao servidor de banco de dados.

5.7 Perfil de Desempenho do Servidor WWW Apache

Com as informações coletadas já podemos construir um perfil de desempenho do servidor WWW apache. Um perfil de desempenho descreve numericamente características

importantes da execução do sistema, de maneira que os resultados de desempenho sejam sintetizados para uma análise posterior.

Os resultados foram obtidos em função de requisições HTTP submetidas ao servidor *Web* através dos clientes, e são reflexos de uma carga de teste real em um determinado período de tempo.

Como mencionado, as métricas utilizadas para avaliar o desempenho do servidor foram a taxa de serviço ou processamento e o tempo médio de resposta do servidor para uma requisição de usuário. Nas duas seções seguintes apresentamos os valores obtidos para essas métricas.

5.7.1 Taxa de Processamento

O Gráfico 3 ilustra os resultados do número de requisições submetidas ao sistema a cada dia. Os dados representam o conjunto de 208 registros diários de acesso ao servidor *Web*, no período de 07/01/2002 a 02/08/2002, totalizando 413216 requisições HTTP. Nesse período registrou-se, como podemos perceber no gráfico, um valor máximo de 4035 requisições em único dia para uma média de 1996 requisições diárias (ver Anexo 2).

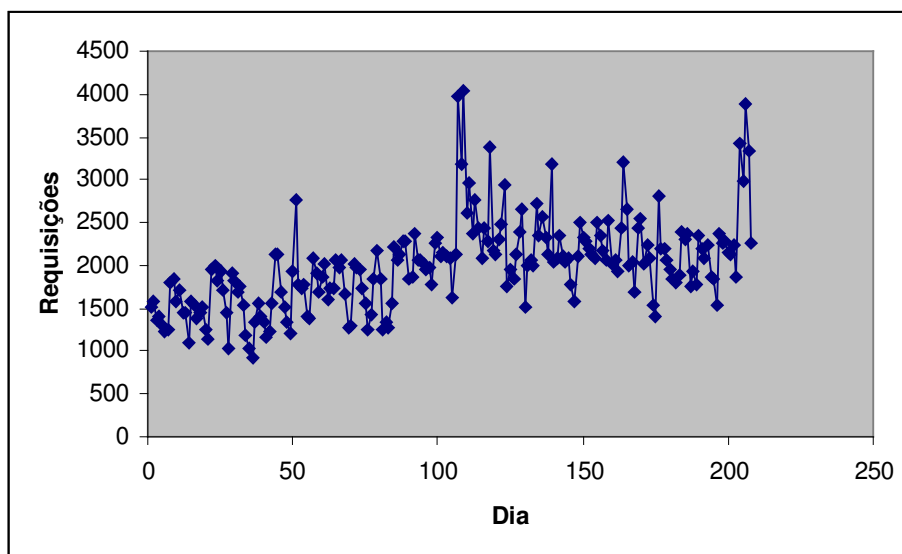


Gráfico 3: Medidas do servidor: taxa de processamento em requisições/dia

A curva para a taxa de processamento mostrada no Gráfico 4 tem comportamento parecido com o da curva do Gráfico 3. Isto é compreensível dado que à medida que aumenta o

número de conexões com o servidor, aumenta também a quantidade de *bytes* transmitidos pelo mesmo. A taxa máxima de dados produzida pelo sistema em um único dia, foi de 122 *Mbytes* de um total de 12610 *Mbytes* ou 12610000 *Kbytes* processados no período, o que nos dá uma média de 60625 *Kbytes* por dia (ver Anexo 2).

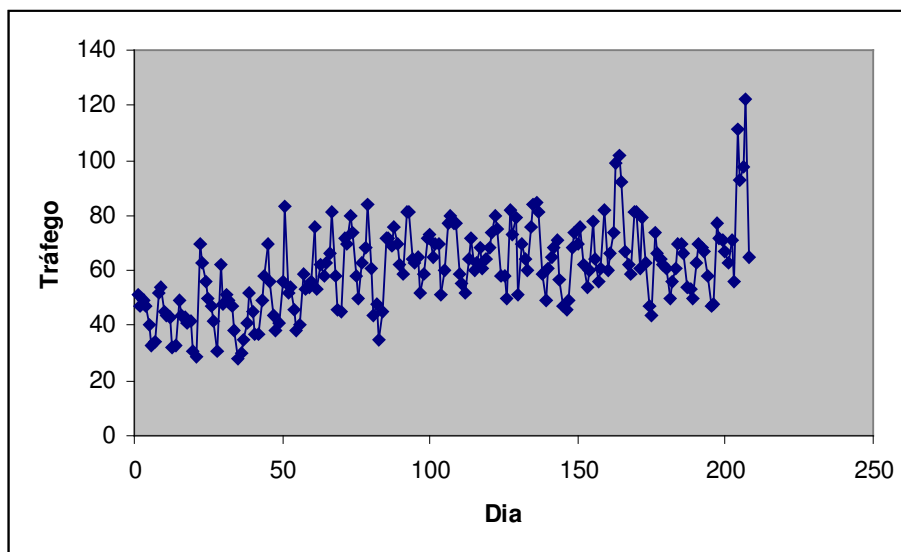


Gráfico 4: Medidas do servidor: taxa de processamento em Mbytes/dia

5.7.2 Tempo de Resposta

Outra métrica de interesse é a latência ou tempo de resposta do servidor. Como dito anteriormente na seção 5.4.2, este tempo corresponde à latência média que é obtida pelo quociente entre o tempo decorrido para atendimento das requisições e o número de requisições atendidas com sucesso.

A ferramenta *Advanced Log Analyzer* nos permite calcular este tempo, que é de aproximadamente 43 segundos. Este valor é calculado em função do número médio de requisições diárias, que por sua vez equivale a menos da metade do número máximo de requisições atendidas pelo servidor em um único dia. Considerando estes dois valores o tempo de resposta médio cai para aproximadamente 21,4 segundos.

A Tabela 6 apresenta valores máximo e médio para a taxa de processamento diária, bem como para o tempo de resposta médio do servidor a uma requisição de usuário. Na seção 5.9 será feita uma análise dos resultados sobre esses dados de execução do sistema.

Tabela 6: Perfil de desempenho do servidor WWW Apache

Servidor Apache	Requisições Estáticas e/ou Dinâmicas
Máximo de requisições	4035
Máximo de bytes transferidos (<i>Mbytes</i>)	122
Média de Requisições/dia	1996
Média de bytes transferidos (<i>Mbytes</i> /dia)	60
Tempo de resposta médio (seg)	43

5.8 Acesso a Sites e Previsão de Desempenho do Servidor Web

Tendo em vista o nosso enfoque na avaliação de desempenho baseada em experimentação (mensuração), nesta seção serão discutidos indicadores estatísticos de acesso ao servidor *Web* Apache, dando ênfase àqueles que são resultados da formação de séries de tempo e que, conseqüentemente, podem nos auxiliar no entendimento do desempenho desse sistema.

Os indicadores podem fornecer informações a uma corporação que se utiliza da *Web* para divulgar ou oferecer seus produtos e/ou serviços. Adicionalmente, as aplicações dinâmicas que acessam banco de dados são extremamente importantes para as corporações que utilizam o comércio eletrônico em *Internet* e *Intranets*.

A previsão consiste em um grau de entendimento de desempenho que pode descrever em detalhes níveis do comportamento de desempenho, como, por exemplo, informar o quanto um sistema pode ser melhorado antes que outro fator se torne limitante. Para [MEI02] o nível de previsão é o mais difícil de ser alcançado, dada a enorme complexidade de se estimar e quantificar os efeitos de mudanças no código e no ambiente de execução no comportamento de um servidor de comércio eletrônico.

A questão anterior é, certamente, a maior desvantagem da técnica de monitoramento para fins de avaliação de desempenho de sistemas computacionais, e será discutida com detalhes na seção 5.8.2.

5.8.1 Indicadores Estatísticos de Acesso

Nos experimentos, monitoramos de forma individualizada 6 (seis) *sites* do servidor *Web* do provedor FlorianoNet [AMA02a]. A Tabela 7 ilustra os resultados das medidas centrais e de variação para três *sites*, no período de 16/06/2002 a 04/07/2002. Os dados foram obtidos através do programa PAJE [BER02].

Tabela 7: Sumário estatístico de dados de acesso ao servidor Apache

Índice	Jornal Local	Museu do Automóvel	Laboratório Rocha
Média	7,0	8,0	14,0
Mediana	7,0	8,0	14,0
Moda	8,0	8,0	10,0
Variância	6,93	5,23	17,76
Desvio Padrão	2,70	2,34	4,32
Coefficiente de variação	0,41	0,29	0,30

As medidas mostram que a média de acessos aos *sites* **Jornal Local** e **Museu do Automóvel** é bastante pequena se compararmos à média de acessos ao *site* **GurguéiaNet** que é de 115 visitantes no mesmo período. Por outro lado, o baixo valor da variância para o conjunto de dados dos *sites* **Jornal Local** e **Museu do Automóvel** revelam que não há muita discrepância no que se refere ao número de usuários que acessam estes *sites* a cada dia. O mesmo não ocorre com o conjunto de dados do *site* **Laboratório Rocha** que apresenta uma variância de 17,76. Uma análise dos dados da Tabela 8 (ver Anexo 2) mostrou que os valores para a média permanecem muito próximos, sendo de aproximadamente 7 para o *site* **Jornal Local** e 123 para o *site* **GurguéiaNet**. Os demais *sites* monitorados apresentaram valores para a variância e desvio padrão muito próximos ao encontrado para o *site* **Jornal Local**, o que significa que para estes *sites* o número de visitantes a cada dia também não é muito variável.

Tabela 8: Distribuição do número diário de acessos a *sites* do servidor Apache

Dia	Gurguéia Net	Bloco Furacão	Revista	Jornal Local
28/07	58	25	11	07
29/07	122	68	13	09
30/07	141	60	16	08
31/07	144	62	16	11
01/08	144	94	20	07
02/08	95	22	13	09
03/08	123	12	08	04
04/08	66	09	12	13
05/08	119	08	16	03
06/08	156	06	16	07
07/08	136	07	24	09
08/08	126	53	19	05
09/08	112	19	16	07
10/08	109	13	08	02
11/08	98	02	11	05
12/08	136	09	11	09
13/08	122	03	15	04
14/08	146	06	15	04
15/08	147	06	14	02
16/08	160	80	09	11

Conjuntos como os registros de observações diárias da Tabela 8 que são formados pelo número de acessos a *sites* do servidor Apache são exemplos das séries temporais discutidas no capítulo 4. Neste caso, podemos utilizar a técnica de suavizamento exponencial simples, também apresentada neste mesmo capítulo, para obter previsões de curto prazo. O Gráfico 5 mostra os valores reais e os obtidos através do modelo multiplicativo também descrito no capítulo 4. O último valor ajustado que corresponde à aproximadamente 160 acessos indica a previsão de todos os valores futuros para a série do *site GurguéiaNet*, utilizando um coeficiente de ajuste $K=0,99$. Este Valor de K foi obtido minimizando a soma de quadrado de resíduos das previsões um passo a frente, de acordo com a equação a seguir,

$$S = \sum_{i=1}^n \left(Y_i - \hat{Y}_{i-1}(1) \right)^2.$$

Em S , Y_i é o valor observado no período de tempo atual i , $\hat{Y}_{i-1}(1)$ é a previsão a um passo e n é o número de observações. A Tabela 9 mostra S para alguns valores de K .

Tabela 9: Valores de K e S utilizando o AES (*site GurguéiNet*)

K	Soma de Quadrados de Ajustamentos (S)
0,75	1086,10
0,80	695,24
0,90	176,42
0,95	44,78
0,99	1,82

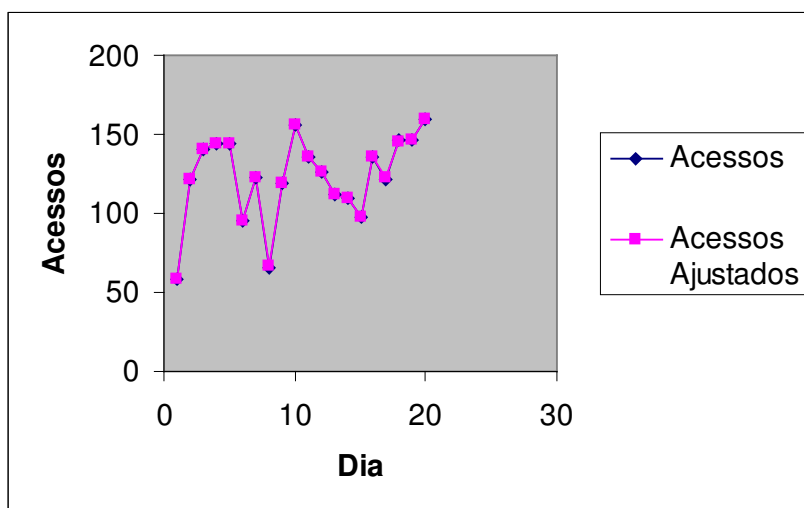


Gráfico 5: Dados originais (Acessos) e valores ajustados (Acessos Ajustados)

Muitas requisições ao *site GurguéiNet* têm caráter dinâmico com acesso ao banco de dados MySQL do provedor FlorinoNet. Nesse sentido, as previsões usando séries temporais são importantes, pois permitem que se faça um prognóstico sobre o número de usuários deste *site*, e como consequência na tomada de decisões para a melhoria do desempenho da aplicação e do desempenho do próprio servidor *Web* como veremos na próxima seção. No Anexo 2 são encontrados os valores reais e exponencialmente ajustados para os dados de acesso ao *site GurguéiNet* (utilizando coeficiente de ajuste $K=0,99$) para os dados da Tabela 8.

5.8.2 Previsão de Desempenho

Um dos objetivos deste trabalho é fazer uma previsão de execução do servidor *Web* Apache usando séries temporais, de maneira que o perfil de desempenho obtido auxilie na

tomada de decisões para o projeto e implementação de uma versão posterior deste sistema. Para isso, é necessário prover informações que permitam, por exemplo, estimar e quantificar aproximadamente a taxa de dados que o servidor deve processar em um determinado período no futuro. Tendo em vista isso, a exemplo do que foi feito para o conjunto de registros diários de acesso ao *site* **GurgúeiaNet**, aplicamos o método de ajuste exponencial simples para fazer previsões de curto prazo (um período no futuro) para a série de tempo obtida do registro diário de todas as requisições HTTP submetidas ao servidor *Web* Apache do provedor **FlorianNet**, no período de 07/01/2002 a 02/08/2002, bem como para a série formada pela quantidade de dados que o servidor processou diariamente no mesmo período, medida em Mbytes. O valor de K , também obtido através de S , para o qual se obtém melhor ajustamento é 0,99, como ilustrado na tabela 10.

Tabela 10 Valores de S utilizando AES (Requisições ao servidor Apache)

K	Soma de Quadrados de Ajustamentos (S)
0,75	2347146
0,80	15277777
0,90	398004
0,95	102016
0,99	4173

No Gráfico 6 temos a representação dos dados originais, ou seja, do número de requisições diárias, e da série exponencialmente ajustada com coeficiente de ajuste $K=0,99$. Os dados originais da série, bem como os valores exponencialmente ajustados são encontrados no Anexo 4.

Conforme foi dito no capítulo 4 sobre séries temporais, um alto valor para o coeficiente K , ou seja, K próximo de 1, é necessário quando objetivamos prever valores futuros para a série [LEV00]. Nesse sentido, a escolha para o valor K acima ocorre em função dos objetivos desse trabalho.

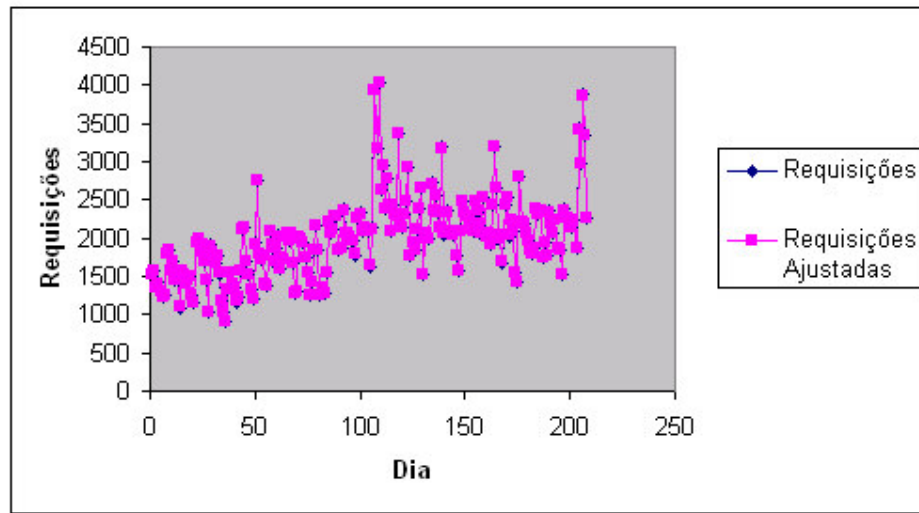


Gráfico 6 : Dados originais (Requisições) e ajustados (Requisições Ajustadas)

Para demonstrar os cálculos para os valores exponencialmente ajustados para fins de previsão, usaremos a equação 4.5 apresentada no capítulo 4. Neste caso, tomamos o valor ajustado em nosso período de tempo atual (período i) como nossa estimativa projetada do valor atual observado da série temporal, no seguinte período de tempo $i + 1$.

Assim, para prever, por exemplo, o número de requisições submetidas ao servidor *Web* Apache durante o dia 03/08/2002, usamos o valor ajustado para o dia 02/08/2002 como sua estimativa. De acordo com a Tabela 14 do Anexo 4, para um coeficiente $K = 0,99$, essa projeção é de aproximadamente 2262 requisições HTTP. Utilizando a equação (4.5), temos:

$$\begin{aligned}\hat{Y}_{208} &= KY_{208} + (1 - K) \hat{Y}_{207}, \\ \hat{Y}_{208} &= (0,99) 2251 + (0,01) 3347, \\ \hat{Y}_{208} &= 2262.\end{aligned}$$

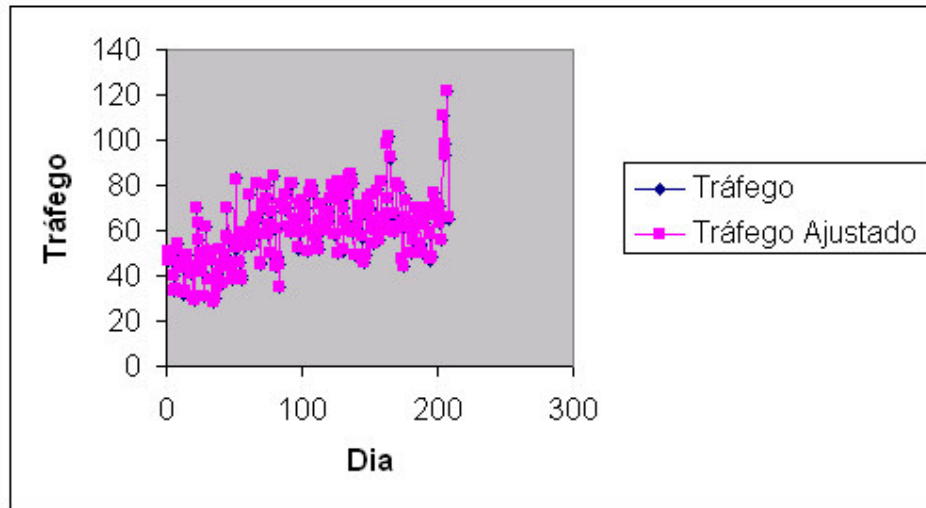


Gráfico 7: Dados originais (Tráfego) e com valores ajustados (Tráfego Ajustado)

De posse dos dados observados para o dia 03/08/2002, podemos obter através da equação (4.8) o valor ajustado para este dia, e utilizar este valor para fazer um prognóstico para o dia 04/08/2002, uma vez que esta equação pode ser interpretada segundo [MOR87] como uma equação de atualização de previsão. Para isso, procedemos da seguinte maneira:

$$\hat{Y}_i(h) = KY_{i+h} + (1 - K) \hat{Y}_{i-1}(h+1), \quad \forall h > 0.$$

Para $h=1$, [MOR87] demonstra que

$$\hat{Y}_i(1) = Ke_i + \hat{Y}_{i-1}(1)$$

onde e_i é o erro de previsão a um passo. Logo,

$$\hat{Y}_{209}(1) = KY_{209} + (1 - K) \hat{Y}_{208}(1),$$

$$\hat{Y}_{209}(1) = (0,99) 3300 + (0,01) 2262,$$

$$\hat{Y}_{209}(1) = 3290,$$

sendo $\hat{Y}_{209}$ é o valor ajustado atual ou nova previsão, Y_{209} é o valor observado atual e $\hat{Y}_{208}$ é o valor ajustado anterior. Este procedimento pode ser utilizado a cada nova observação.

Um raciocínio análogo pode ser utilizado para fazer uma previsão para a taxa de dados a ser processada pelo servidor em um período no futuro. O gráfico 7 contém as curvas para a série de tempo com dados originais e para a série ajustada exponencialmente, com coeficiente de ajuste igual a 0,99. Neste caso, a previsão para o dia 03/08/2002 é de aproximadamente 66

Mbytes. No Anexo 4 encontramos os dados originais e os valores exponencialmente ajustados. A tabela 11 mostra alguns alguns valores para K e para S .

Tabela 11 Valores de S utilizando AES (Tráfego no servidor Apache)

K	Soma de Quadrados de Ajustamentos (S)
0,75	2140,578
0,80	1380,073
0,90	351,935
0,95	89,148
0,99	3,610

5.9 Análise de Resultados

Na seção anterior, encontramos valores para a taxa de processamento do servidor que consistem em um prognóstico para o número de requisições a serem atendidas pelo servidor em um único dia e para a quantidade de dados que ele deve produzir durante um dia.

O conhecimento que temos acerca do perfil de desempenho traçado (ver Tabela 6), como por exemplo, da taxa máxima de processamento em requisições/dia que é de 4035, e da média de requisições/dia que é de 1996, não fornecem informações confiáveis, pois não representam o comportamento típico de acesso ao servidor. Qualquer planejamento de carga futura baseada nestes valores pode significar numa sub-utilização ou sobrecarga do servidor. Por outro lado, os valores obtidos em termos de previsão são de extrema importância, uma vez que na prática serão utilizados pelo administrador do sistema ou equipe responsável, como auxiliares numa avaliação de quais decisões podem ser tomadas para reduzir os custos de execução dos serviços oferecidos, e como consequência no ajuste de desempenho do sistema.

A análise de valores a partir das previsões nos possibilita fazer uma reavaliação das medidas de desempenho do servidor. O tempo de resposta do servidor calculado através da fórmula para a latência média definida na seção 5.4.2, por exemplo, cairia para aproximadamente 38 segundos, caso se confirmasse um valor aproximado de 2262 requisições/dia. Visto que podemos ter acesso a novos dados de utilização do sistema é que a equação (4.8) representa uma equação de atualização de previsão, esse tempo de resposta baixaria para 26,3 segundos para um prognóstico de aproximadamente 3290 requisições/dia.

Do ponto de vista do usuário isto representa uma melhora significativa se compararmos os 43 segundos encontrados quando da obtenção de um perfil de desempenho do sistema. Isto é interessante, pois um baixo tempo de resposta representa um melhor desempenho.

Outro fator importante, e que é de grande interesse das corporações principalmente, se refere às previsões de acesso aos *sites* orientados a dados, isto é, aqueles *sites* nos quais as requisições envolvem acesso a grandes banco de dados. Tais empresas oferecem através dos seus sistemas os chamados serviços de comércio eletrônico. No provedor de acesso à Internet FlorianonNet, por exemplo, é oferecido indistintamente a seus clientes ou futuros clientes um serviço de cadastramento através da *Web*.

5.10 Conclusões

Este capítulo apresenta uma análise de desempenho de um servidor *Web* cuja carga de trabalho é composta por requisições estáticas e requisições dinâmicas a *sites* orientados a dados num percentual de 4%. Através da ferramenta *Advanced Log Analyzer*, foi possível analisar os arquivos de registro de acesso do servidor e capturar dessa forma os dados de utilização do servidor em função das requisições HTTP submetidas.

Os valores máximos obtidos para a taxa de processamento foi de 4035 requisições HTTP por dia e 122 *Mbytes* transmitidos em único dia. Esses valores não são tão baixos se compararmos, por exemplo, um servidor *Web* instalado na National Computer Security Association, um dos *sites*, mais acessados que atende, em média, 24419 conexões por dia [MUR96].

Os valores para o tempo de resposta são relativamente altos, se compararmos a estudos de avaliação de desempenho de servidores *Web* como o desenvolvido por [MEN98], no qual supõe-se, num dado momento, uma carga de trabalho composta de requisições de acesso a banco de dados num total de 20% e 80% de requisições estáticas. O tempo de resposta encontrado nesta avaliação é inferior a 3 segundos. Contudo, os experimentos realizados neste trabalho têm por base uma carga de teste real, com dados e programas originais. Em contrapartida, no estudo referido acima não se usou um ambiente corporativo real. Além do mais, o presente trabalho não tem por objetivo, analisar o aspecto de saturação de um servidor *Web*, e sim obter um perfil de desempenho do mesmo.

Por fim, foi feita uma previsão do desempenho do servidor Web Apache através da aplicação de um método de previsão de séries de tempo, o suavizamento exponencial simples.

Através deste, foi possível fazer um prognóstico de cargas futuras do sistema, de forma que isso possa ser utilizado como mecanismo auxiliar nas tomadas de decisões de redução de custo de uso do sistema e conseqüente ajuste de desempenho.

CAPÍTULO 6

Conclusão

6.1 Considerações Finais

Este trabalho apresentou a avaliação desempenho de um servidor *Web*, no caso o Apache, no provedor FlorianoNeto localizado em Floriano-PI, Brasil. As métricas usadas foram taxa de processamento e tempo de resposta. Para os dados obtidos foram calculados os indicadores estatísticos, média, mediana, moda, variância, desvio padrão e coeficiente de variação. Além destes, os dados foram modelados através dos modelos de suavizamento exponencial de séries temporais, para previsões h passos a frente; estes mostraram-se eficientes no tocante a prover informações gerenciais úteis à política de gerenciamento do servidor.

6.2 Contribuições

A principal contribuição deste trabalho consiste em considerar uma modelagem estatística [AMA02a, AMA02b]:

- **Descritiva**, calculando os indicadores estatísticos usuais (média, mediana, moda, variância e desvio padrão), e
- **Inferencial**, aplicando modelos de séries temporais para analisar dados provenientes do provedor FlorianoNet, com vistas à tomada de decisão sobre sua política de gerenciamento. Convém enfatizar que não foi encontrada na literatura referência que aborde o desempenho de um servidor *Web* através de séries temporais. Portanto, a avaliação de desempenho aqui realizada é inédita na área.

6.3 Trabalhos Futuros

As propostas para continuação deste trabalho são as seguintes:

- Estender o estudo de séries temporais, aqui feito para outros modelos, em particular os não-automáticos.
- Além das variáveis aqui estudadas pesquisar a existência de outras que afetem o desempenho do servidor *Web* Apache implementado no provedor FlorianoNet.
- Aplicar a metodologia de avaliação de desempenho usada neste trabalho para o servidor do FlorianoNet a outros provedores do estado do Piauí.
- Comparar o desempenho dos vários servidores avaliados.

Referências Bibliográficas

-
- [ABI00] Abiteboul, S., Buneman, P., Suciu, D. **Gerenciando Dados na Web**, Editora Campus; Rio de Janeiro, 2000.
- [ACS02] <http://www.acessos.com>
- [ADR01] Adriano, C.M. **Tendência em Gerência de Informação**, 1998. Disponível em [www](http://www.dca.fee.unicamp.br/course/IA368F/1s1998/Monografias/christian.html) por: <http://www.dca.fee.unicamp.br/course/IA368F/1s1998/Monografias/christian.html>. Período da acesso: agosto de 2001.
- [ALA02] <http://www.vknoware.com/ala/>.
- [AMA02a] Amaral, F. E., Campos, M.A e Souza, F. F, **Indicadores Estatísticos de Acesso ao Servidor Apache no Provedor FlorianoNet**. Congresso Nacional de Matemática Aplicada e Computacional, DMC/UERG, 16-19, Setembro 2002, pp 21-21.
- [AMA02b] Amaral, F. E., Campos, M.A e Souza, F. F, **Análise de Desempenho do Servidor Apache no Provedor FlorianoNet**. Encontro Regional de Matemática Aplicada e Computacional, Regional 03, DIMAP/UFRN, 29-31, Agosto 2002, pp87-87.
- [AND71] Anderson, T. W. **The Statistical Analysis of Time Series**. New York, John Wiley and Sons, 1971.
- [APA02] *Apache HTTP Server*, 2002. Disponível por www em <http://www.apache.org/> Período acesso: abril de 2002.
- [ARN97] Arnold, K., Gosling, J. **Programando em Java**. Source, São Paulo, 1997.
- [BER02] Bernardino, C. P. B., Alves, D. J. C., Salvador, E. F., Silva, E. L., Lima, J. S. M., Campos, M. A., Barros, S. C. R. **Programa de Análises em Java para Estatística**. Encontro Regional de Matemática Aplicada e Computacional, Regional 03, DIMAP/UFRN, 29-31, Agosto 2002, pp 31-32.
- [COF99] Coffman, G. **SQL Server 7: The Complete Reference**, McGraw-Hill, 1999.
- [DAM97] Damasceno Jr., A. F. **Intranets e Extranets com Java 1.1**, Editora Érica, São Paulo, 1997.
- [DEI90] Deitel, H. M. **Operating System**. Reading: Addison – Wesley Publishing, 1990.
- [DEL02] System Performance Analysis: Tools, Techniques, and Methodology [Http://www.dell.com.br](http://www.dell.com.br). Período de acesso: julho de 2002.

-
- [ELM02] Elmasri, R; Navathe, S. **Sistemas de Banco de Dados: Fundamentos e aplicações**. LTC, 2002.
- [ELM94] Elmasri, R., Navathe, S. **Fundamentals of Database Systems**, Benjamin-Cummings, 1994.
- [FER83] Ferrari, Dominico, Serrazzi, Giuseppe; Zeiner, Alessandro. **Measurement na Tuning of Computer Systems**. Enlewood Cliffs: Prentice-Hall, 1983.
- [FUN99] Funes, M. **Java e Banco de Dados**, Brasport, Rio de Janeiro, 1999.
- [GUN98] Gundanvaram, S . **CGI Programming on the World Wide Web**. USA: O'Reilly, 1998.
- [HER97] Herrmann, E., **Programação CGI com Perl 5**: Makron Books, 1997.
- [IBRC02] <http://www.ibope.com.br/eratings/>. Período de acesso: março de 2002.
- [JAI91] Jain, R. **The Art of Computer Systems Performance analysis – Techniques for Experimental Design, Measurement, Simulation, and Modeling**. John Wiley & Sons, 1991.
- [KAB02] Kabir, M. J. **Apache Server 2 a Bíblia**, Editora Campus, Rio de Janeiro, 2002.
- [KEN76] Kenn, S. M., **Time Series**, Charles Griff, 1976.
- [LAQ95] Laquey; T e R, Jeanne, C. **O Manual da Internet**. Editora Campus, 1995.
- [LEV00] Levine, D. M., Berenson, M. L., Stephan, D. **Estatística: teoria e aplicações**, 3ª ed., LTC, 2000.
- [LIM97] Lima, I.N. **“O Ambiente Web Banco de Dados : Funcionalidades e Arquiteturas de Integração”**. Dissertação de Mestrado. Departamento de Informática, PUC-Rio , 1997.
- [MAR99] Marcon, A.M., Neves, D. **Aplicações e Banco de Dados para Internet**, Editora Érica , São Paulo , 1999 .
- [MAS00] Maslakowski, M.; Butcher, T. **MySQL in 21 days**. Sams Publishing, 2000.
- [MAY01] Mayer & BUNGE INFORMÁTICA. **Otimizando o Desempenho de Banco de Dado Relacionais**. 1998. Disponível por WWW em <http://www.mbi.com.br/museum/articles/misnet98t1.html>. Período de acesso: agosto de 2001.
- [MCC97] McComb, G. **Java Script Source Book**. Markon Books, 1997.
- [MEI02] Meira Jr, W.; Murta, C. D; Campos, S.V.A; Guedes Neto, D.O. **Sistemas de Comércio Eletrônico – Projeto e Desenvolvimento**. Editora Campus, 2002.
- [MEL98] Melo, R.N., Silva, D.S., Tanaka, A.K. **Banco de Dados em**

- Aplicações Cliente-Servidor** , Infobook , Rio de Janeiro , 1998 .
- [MEN00] Menascé, D., Almeida V., Scaling for e-Business: **Tecnologies, Modelos , Perfomance and Capacity Planning**, Prentice- Hall, 2000.
- [MEN98] Mendes, M.A.**Comparação e Análise de Requisições Dinâmicas em Servidores WWW**. Disponível por WWW em <http://www.dcc.ufmg.br/pos/html/sps98/anais/corelio/corelio.html>. Período de acesso: março de 2002.
- [MOR85] Morettin, P. A, Toloi, C.M.C. **Previsão de Séries Temporais**. Atual Editora.1985
- [MOR87] Morettin, P. A, Toloi, C.M.C. **Métodos Quantitativos – Séries Temporais**. Atual Editora.1987.
- [MSL02] <http://www.mysql.com>. Período de acesso: março de 2002
- [MUR96] Murta, C.D., Almeida, J.M. e Almeida, V.A.F. “**Análise de Desempenho de um Servidor WWW**”, Anais do XXII Seminário Integrado de Software e Hardware, IX CTD, XV CTIC, Recife, PE, 1996, pp.391-402.
- [NHBF02] Performance Tuning <http://www.15seconds.com/issue/971127.htm>.Período de acesso: Março de 2002
- [OZS99] Özsu, M.T., Valduriez, P. **Principles of Distributed Database System**, Prentice-Hall,1999.
- [PMI00] Milani, P. Web Server Response Estimation by Crossing HTTP Time and ICMP Time. **Journal of Computer Resourse Management** Issue 97, Winter 2000. <http://www.org/national/journal.html>.
- [RED02] <http://www.redhat.com>. Período de acesso: agosto de 2002.
- [SAM01] **Database Backed Web Aplication Development Environment for Windows 95/98/ME Platform Using Open-Source Tools**. <http://in-geocities.com/samdarshipadi/apache-php-mysql.htm>. Acesso: agosto de 2001.
- [SIL99] Silberschatz, A., Korth, F., Sudarshan, S. **Sistema de Banco de Dados**, Makron Books , São Paulo , 1999 .
- [SIV01] Silva, L.C., **Banco de Dados para Web: do planejamento à implementação**. Editora Érica, 2001.
- [TAN96] Tanenbaun, Andrew S., **Computer Network**, Prentice Hall, 1996.
- [VIC98] Vicari, S.R. **Proposta de um Sistema de Regras de Sintonia de Performance**

para Aplicações de Banco de Dados. Dissertação de Mestrado. Instituto de Informática, UFRGS, 1998.

- [VID01] Vidal, V.M.P., Lóscio, B.F. and Salgado, A. C. **Using correspondence assertions for specifying the semantics of XML-based mediators**, In Proc. of WIIW 2001 – International Workstation on Information Integration on the *Web* – Technologies an applications, Rio de Janeiro, Brasil, 2001.

Anexos

Anexo 1 – Código fonte contador Acessos

```
<!-- Início Código Acessos -->
```

```
<script language="javascript">
```

```
<!--
```

```
id="NOME_DO_SITE";
```

```
video="NaoDisponivel";
```

```
cores="NaoDisponivel";
```

```
rf=escape(document.referrer)+"";
```

```
if((rf=="undefined") || (rf==""))
```

```
{
```

```
rf="EnderecoDigitado";
```

```
};
```

```
bn=navigator.appName;
```

```
if(bn.substring(0,9)=="Microsoft")
```

```
{
```

```
bn="Internet Explorer";
```

```
};
```

```
if(rf.substring(0,27)=="http%3A//www.acessos.com.br")
```

```
{
```

```
rf="Acessos-Resultado";
```

```
};
```

```
//--></script>
```

```
<script language="javascript1.2">
```

```
<!--
```

```
cores=(bn=="Internet Explorer"?screen.colorDepth:screen.pixelDepth;
```

```
if(cores=="undefined"){
```

```
cores="NaoDisponivel";
```

```
};
```

```
video=screen.width+"x"+screen.height;
```

```
//--></script>
```

```
<script language="javascript">
```

```
<!--
```

```
t1="<a href='http://www.acessos.com.br/cgi-bin/acessos.cgi?relatorio&id="+id;
```

```
t2="" target=_blank>";
```

```
t3="<img src='http://www.acessos.com.br/cgi-bin/acessos.cgi?";
```

```
t4="&ss="+video+"&cores="+cores;
```

```
t5="&rf="+rf+"&id="+id;
```

```
t6="" border='0'";
```

```
t7="alt='Acessos'></a>";
```

```
document.write(t1+t2+t3+t4+t5+t6+t7);
```

```
//--></script>
```

```
<script language="javascript1.2">
```

```
<!--
```



```
document.write("<");document.write("!--");
//--></script>
<noscript>
<img src='http://www.acessos.com.br/cgi-bin/acessos.cgi?' border='0' alt='Acessos'></a>
</noscript>
<script language="javascript1.2">
<!--
document.write("--");document.write(">");
//--></script>

<!-- Fim Código Acessos -->
```

Anexo 2 – Medidas estatísticas de acesso ao servidor Apache

Tabela 12: Estatístico de acesso a *sites* do servidor Apache, dados da Tabela 8

	Média	Variância	Desvio	Mediana	Moda	Mínimo	Máximo	Soma
GurguéiaNet	123	754,4211	27,46673	124,5	122	58	160	2460
Bloco Furacão	28,2	862,4842	29,36	12,5	6	2	94	564
Jornal Local 2	6,8	9,747368	3,122078	7	7	2	13	283
Revistas	14,15	16,45	4,05586	14,5	16	8	24	133

Tabela 13: Estatístico de requisições ao servidor Apache

Medida	Média	Variância	Desvio Padrão	Mediana	Moda	Mínimo	Máximo	soma
Valor	1996	274776,300	524,1911	2006	1414	920	4035	413216

Tabela 14 : Estatístico do tráfego processado pelo servidor Apache (*Mbytes*)

Medida	Média	Variância	Desvio Padrão	Mediana	Moda	Mínimo	Máximo	Soma
Valor	60,25	239,5688	15,478	61	70	28	122	12610

Tabela 15 : AES aplicado a valores de acesso ao site GurguéiaNet, dados Tabela 8

Dia	Acessos	Acessos Ajustados(K=0,75)
28/Jul	58	58,00000
29/Jul	122	121,36000
30/Jul	141	140,80360
31/Jul	144	143,96800
01/Ago	144	143,99970
02/Ago	95	95,49000
03/Ago	123	122,72490
04/Ago	66	66,56725
05/Ago	119	118,47570
06/Ago	156	155,62480
07/Ago	136	136,19620
08/Ago	126	126,10200
09/Ago	112	112,14100
10/Ago	109	109,03140
11/Ago	98	98,110310
12/Ago	136	135,62110
13/Ago	122	122,13620
14/Ago	146	145,76140
15/Ago	147	146,98760
16/Ago	160	159,86990

Anexo 3 – Dados originais das requisições HTTP submetidas ao servidor Web Apache

Tabela 16 : AES aplicado a 208 valores da série de requisições, K=0,99

Dia	Requisições	Requisições Ajustadas	Dia	Requisições	Requisições Ajustadas
07/01/02	1513	1513	06/03/02	1686,307	1741,666
08/01/02	1582	1581,31	07/03/02	1871,133	1840,166
09/01/02	1359	1361,223	08/03/02	2028,411	1982,542
10/01/02	1414	1413,472	09/03/02	1596,364	1689,635
11/01/02	1291	1292,225	10/03/02	1737,574	1726,659
12/01/02	1233	1233,592	11/03/02	1725,126	1725,415
13/01/02	1248	1247,856	12/03/02	2058,631	1977,854
14/01/02	1796	1790,519	13/03/02	1974,846	1974,963
15/01/02	1837	1836,535	14/03/02	2068,058	2045,491
16/01/02	1570	1572,665	15/03/02	1672,991	1763,123
17/01/02	1708	1706,647	16/03/02	1278,98	1397,031
18/01/02	1444	1446,626	17/03/02	1298,8	1323,508
19/01/02	1458	1457,886	18/03/02	2013,778	1846,627
20/01/02	1093	1096,649	19/03/02	1983,308	1948,907
21/01/02	1576	1571,206	20/03/02	1953,303	1951,977
22/01/02	1526	1526,452	21/03/02	1744,113	1794,494
23/01/02	1391	1392,355	22/03/02	1557,881	1615,624
24/01/02	1440	1439,524	23/03/02	1262,979	1348,906
25/01/02	1512	1511,275	24/03/02	1432,29	1412,726
26/01/02	1244	1246,673	25/03/02	1839,883	1736,182
27/01/02	1144	1145,027	26/03/02	2171,649	2065,295
28/01/02	1954	1945,91	27/03/02	1841,336	1894,824
29/01/02	1992	1991,539	28/03/02	1265,813	1418,706
30/01/02	1820	1821,715	29/03/02	1346,188	1364,926
31/01/02	1927	1925,947	30/03/02	1267,792	1291,482
01/02/02	1705	1707,209	31/03/02	1557,078	1492,87
02/02/02	1452	1454,552	01/04/02	2203,471	2030,718
03/02/02	1039	1043,156	02/04/02	2068,365	2057,929
04/02/02	1909	1900,342	03/04/02	2120,474	2105,232
05/02/02	1823	1823,773	04/04/02	2289,295	2244,558
06/02/02	1683	1684,408	05/04/02	2290,983	2279,39
07/02/02	1760	1759,244	06/04/02	1840,55	1946,847
08/02/02	1535	1537,242	07/04/02	1872,675	1891,462
09/02/02	1179	1182,582	08/04/02	2372,947	2256,365
10/02/02	1035	1036,476	09/04/02	2066,099	2111,341
11/02/02	920	921,1648	10/04/02	2063,031	2075,085
12/02/02	1329	1324,922	11/04/02	2021,42	2034,521
13/02/02	1557	1554,679	12/04/02	1961,604	1979,38
14/02/02	1411	1412,437	13/04/02	1964,966	1968,595
15/02/02	1347	1347,654	14/04/02	1782,84	1827,899
16/02/02	1168	1169,797	15/04/02	2255,228	2151,975
17/02/02	1240	1239,298	16/04/02	2323,312	2280,994
18/02/02	1568	1564,713	17/04/02	2099,263	2142,998
19/02/02	2122	2116,427	18/04/02	2145,533	2145,25
20/02/02	2135	2134,814	19/04/02	2112,335	2120,312
21/02/02	1687	1691,478	20/04/02	2092,203	2099,078
22/02/02	1513	1514,785	21/04/02	1636,602	1748,77
23/02/02	1330	1331,848	22/04/02	2123,086	2033,192
24/02/02	1215	1216,168	23/04/02	3946,581	3482,048
25/02/02	1937	1929,792	24/04/02	3183,706	3252,512
26/02/02	2759	2750,708	25/04/02	4026,487	3839,378
27/02/02	1785	1794,657	26/04/02	2636,045	2926,345
28/02/02	1727	1727,677	27/04/02	2955,77	2950,836
01/03/02	1771	1770,567	28/04/02	2385,758	2522,709
02/03/02	1397	1400,736	29/04/02	2771,108	2711,927
03/03/02	1377	1377,237	30/04/02	2429,451	2497,482
04/03/02	2091	2083,862	01/05/02	2089,435	2188,87
05/03/02	1913	1914,709	02/05/02	2436,494	2377,218

Dia	Requisições	Requisições Ajustadas	Dia	Requisições	Requisições Ajustadas
03/05/02	2290	2291,465	30/06/02	1414	1415,365
04/05/02	3381	3370,105	01/07/02	2811	2797,044
05/05/02	2170	2182,001	02/07/02	2198	2203,99
06/05/02	2132	2132,5	03/07/02	2197	2197,07
07/05/02	2301	2299,315	04/07/02	2053	2054,441
08/05/02	2491	2489,083	05/07/02	1961	1961,934
09/05/02	2933	2928,561	06/07/02	1852	1853,099
10/05/02	1767	1778,616	07/07/02	1806	1806,471
11/05/02	1950	1948,286	08/07/02	1877	1876,295
12/05/02	1847	1848,013	09/07/02	2382	2376,943
13/05/02	2120	2117,28	10/07/02	2309	2309,679
14/05/02	2389	2386,283	11/07/02	2367	2366,427
15/05/02	2653	2650,333	12/07/02	1749	1755,174
16/05/02	1514	1525,363	13/07/02	1935	1933,202
17/05/02	2006	2001,194	14/07/02	1769	1770,642
18/05/02	2064	2063,372	15/07/02	2354	2348,166
19/05/02	1999	1999,644	16/07/02	2189	2190,592
20/05/02	2723	2715,766	17/07/02	2076	2077,146
21/05/02	2353	2356,628	18/07/02	2248	2246,291
22/05/02	2558	2555,986	19/07/02	1870	1873,763
23/05/02	2330	2332,26	20/07/02	1839	1839,348
24/05/02	2130	2132,023	21/07/02	1529	1532,103
25/05/02	3191	3180,41	22/07/02	2375	2366,571
26/05/02	2047	2058,334	23/07/02	2258	2259,086
27/05/02	2088	2087,703	24/07/02	2272	2271,871
28/05/02	2350	2347,377	25/07/02	2156	2157,159
29/05/02	2098	2100,494	26/07/02	2134	2134,232
30/05/02	2035	2035,655	27/07/02	2229	2228,052
31/05/02	2087	2086,487	28/07/02	1874	1877,541
01/06/02	1773	1776,135	29/07/02	3435	3419,425
02/06/02	1572	1574,041	30/07/02	2979	2983,404
03/06/02	2104	2098,7	31/07/02	3875	3866,084
04/06/02	2495	2491,037	01/08/02	3342	3347,241
05/06/02	2329	2330,62	02/08/02	2251	2261,962
06/06/02	2287	2287,436			
07/06/02	2206	2206,814			
08/06/02	2122	2122,848			
09/06/02	2086	2086,368			
10/06/02	2494	2489,924			
11/06/02	2347	2348,429			
12/06/02	2175	2176,734			
13/06/02	2053	2054,237			
14/06/02	2533	2528,212			
15/06/02	2012	2017,162			
16/06/02	2070	2069,472			
17/06/02	1928	1929,415			
18/06/02	2440	2434,894			
19/06/02	3202	3194,329			
20/06/02	2652	2657,423			
21/06/02	1992	1998,654			
22/06/02	2044	2043,547			
23/06/02	1681	1684,625			
24/06/02	2437	2429,476			
25/06/02	2542	2540,875			
26/06/02	2026	2031,149			
27/06/02	2231	2229,001			
28/06/02	2091	2092,38			
29/06/02	1545	1550,474			

Anexo 4 – Dados originais e ajustados para a taxa diária de processamento do servidor *Web Apache* em *Mbytes* por dia.

Tabela 17 : AES aplicado a 208 valores da série de tráfego, K=0,99

Dia	Tráfego	Tráfego Ajustado	Dia	Tráfego	Requisições Ajustadas
07/01/02	51	51	06/03/02	53,99058	53,83031
08/01/02	47	47,04	07/03/02	55,97991	55,45758
09/01/02	49	48,9804	08/03/02	75,7998	70,86439
10/01/02	47	47,0198	09/03/02	53,228	57,4661
11/01/02	40	40,0702	10/03/02	61,91228	60,86652
12/01/02	33	33,0707	11/03/02	58,03912	58,71663
13/01/02	34	33,99071	12/03/02	62,95039	61,92916
14/01/02	52	51,81991	13/03/02	65,9695	64,98229
15/01/02	54	53,9782	14/03/02	80,8497	76,99557
16/01/02	45	45,08978	15/03/02	58,2285	62,74889
17/01/02	44	44,0109	16/03/02	46,12228	50,18722
18/01/02	43	43,01011	17/03/02	45,01122	46,29681
19/01/02	32	32,1101	18/03/02	71,73011	65,5742
20/01/02	33	32,9911	19/03/02	70,0173	68,89355
21/01/02	49	48,83991	20/03/02	79,90017	77,22339
22/01/02	44	44,0484	21/03/02	74,059	74,80585
23/01/02	43	43,01048	22/03/02	58,16059	62,20146
24/01/02	41	41,0201	23/03/02	50,08161	53,05037
25/01/02	42	41,9902	24/03/02	62,87082	60,51259
26/01/02	31	31,1099	25/03/02	67,94871	66,12815
27/01/02	29	29,0211	26/03/02	83,83949	79,53204
28/01/02	70	69,59021	27/03/02	61,22839	65,63301
29/01/02	63	63,0659	28/03/02	44,17228	49,40825
06/01/00	56	56,07066	29/03/02	47,96172	48,35206
31/01/02	50	50,06071	30/03/02	35,12962	38,33802
01/02/02	47	47,03061	31/03/02	44,9013	43,3345
02/02/02	42	42,05031	01/04/02	71,72901	64,83363
03/02/02	31	31,1105	02/04/02	71,99729	70,20841
04/02/02	62	61,69111	03/04/02	69,02997	69,3021
05/02/02	48	48,13691	04/04/02	75,9303	74,32553
06/02/02	51	50,97137	05/04/02	70,0593	2279,39
07/02/02	49	49,01971	06/04/02	62,08059	1946,847
08/02/02	47	47,0202	07/04/02	59,03081	1891,462
09/02/02	38	38,0902	08/04/02	80,78031	2256,365
10/02/02	28	28,1009	09/04/02	80,9978	2111,341
11/02/02	30	29,98101	10/04/02	64,16998	2075,085
12/02/02	35	34,94981	11/04/02	63,0117	2034,521
13/02/02	41	40,9395	12/04/02	64,98012	1979,38
14/02/02	52	51,88939	13/04/02	52,1298	1968,595
15/02/02	45	45,06889	14/04/02	58,9313	1827,899
16/02/02	37	37,08069	15/04/02	71,86931	2151,975
17/02/02	37	37,00081	16/04/02	72,98869	2280,994
18/02/02	49	48,88001	17/04/02	65,07989	2142,998
19/02/02	58	57,9088	18/04/02	69,9508	2145,25
20/02/02	70	69,87909	19/04/02	69,99951	2120,312
21/02/02	56	56,13879	20/04/02	51,19	2099,078
22/02/02	44	44,12139	21/04/02	59,9119	1748,77
23/02/02	38	38,06121	22/04/02	76,82912	2033,192
24/02/02	41	40,97061	23/04/02	79,96829	3482,048
25/02/02	56	55,84971	24/04/02	77,02968	3252,512
26/02/02	83	82,7285	25/04/02	77,0003	3839,378
27/02/02	52	52,30728	26/04/02	59,18	2926,345
28/02/02	54	53,98307	27/04/02	55,0418	2950,836
01/03/02	46	46,07983	28/04/02	52,03042	2522,709
02/03/02	38	38,0808	29/04/02	63,8803	2711,927
03/03/02	40	39,98081	30/04/02	71,9188	2497,482
04/03/02	59	58,80981	01/05/02	60,11919	2188,87
05/03/02	53	53,0581	02/05/02	62,97119	2377,218

Día	Tráfego	Tráfego Ajustado	Día	Tráfego	Tráfego Ajustado
03/05/02	2290	67,94971	11/07/02	66	66,03999
04/05/02	3381	61,0695	12/07/02	54	54,1204
05/05/02	2170	63,97069	13/07/02	53	53,0112
06/05/02	2132	67,95971	14/07/02	50	50,03011
07/05/02	2301	73,9396	15/07/02	63	62,8703
08/05/02	2491	79,9394	16/07/02	70	69,9287
09/05/02	2933	75,04939	17/07/02	68	68,01929
10/05/02	1767	58,17049	18/07/02	67	67,01019
11/05/02	1950	58,0017	19/07/02	58	58,0901
12/05/02	1847	50,08002	20/07/02	47	47,1109
13/05/02	2120	81,6808	21/07/02	48	47,99111
14/05/02	2389	73,08681	22/07/02	77	76,70991
15/05/02	2653	78,94087	23/07/02	72	72,0471
16/05/02	1514	51,27941	24/07/02	71	71,01047
17/05/02	2006	69,81279	25/07/02	67	67,0401
18/05/02	2064	64,05813	26/07/02	63	63,0404
19/05/02	1999	60,04058	27/07/02	71	70,9204
20/05/02	2723	75,84041	28/07/02	56	56,1492
21/05/02	2353	83,9184	29/07/02	111	110,4515
22/05/02	2558	84,98918	30/07/02	93	93,17451
23/05/02	2330	81,03989	31/07/02	98	97,95175
24/05/02	2130	59,2204	01/08/02	122	121,7595
25/05/02	3191	49,1022	02/08/02	65	65,5676
26/05/02	2047	60,88102			
27/05/02	2088	64,95881			
28/05/02	2350	67,96959			
29/05/02	2098	70,9697			
30/05/02	2035	57,1397			
31/05/02	2087	47,1014			
01/06/02	1773	46,01101			
02/06/02	1572	48,97011			
03/06/02	2104	67,8097			
04/06/02	2495	73,9381			
05/06/02	2329	70,03938			
06/06/02	2287	75,94039			
07/06/02	2206	62,1394			
08/06/02	2122	54,08139			
09/06/02	2086	59,94081			
10/06/02	2494	77,81941			
11/06/02	2347	64,13819			
12/06/02	2175	56,08138			
13/06/02	2053	60,95081			
14/06/02	2533	81,78951			
15/06/02	2012	60,2179			
16/06/02	2070	65,94218			
17/06/02	1928	73,91942			
18/06/02	2440	98,74919			
19/06/02	3202	101,9675			
20/06/02	2652	92,09967			
21/06/02	1992	67,251			
22/06/02	2044	62,05251			
23/06/02	1681	59,03053			
24/06/02	2437	80,78031			
25/06/02	2542	80,9978			
26/06/02	2026	61,19998			
27/06/02	2231	78,822			
28/06/02	2091	63,15822			
29/06/02	1545	47,16158			
30/06/02	44	44,03162			
01/07/02	74	73,70032			
02/07/02	66	66,077			
03/07/02	64	64,02077			
04/07/02	62	62,02021			
05/07/02	61	61,0102			
06/07/02	50	50,1101			
07/07/02	56	55,9411			
08/07/02	61	60,94941			
09/07/02	70	69,90949			
10/07/02	70	69,99909			