

UNIVERSIDADE FEDERAL DE GOIÁS – UFG
CAMPUS CATALÃO – CaC
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO – DCC

Bacharelado em Ciência da Computação

Projeto Final de Curso

Um estudo da tecnologia Web 2.0

Autor: Danilo dos Reis Moreira

Orientador: Márcio de Souza Dias

Danilo dos Reis Moreira

Um estudo da tecnologia Web 2.0

Monografia apresentada ao Curso de
Bacharelado em Ciência da Computação da
Universidade Federal de Goiás Campus Catalão
como requisito parcial para obtenção do título de
Bacharel em Ciência da Computação

Área de Concentração: Tecnologia Emergente

Orientador: Márcio de Souza Dias

Moreira, Danilo dos R.

Um estudo da tecnologia Web 2.0/Márcio de Souza Dias- Catalão - 2009

Número de paginas: 101

Projeto Final de Curso (Bacharelado) Universidade Federal de Goiás, Campus Catalão, Curso de Bacharelado em Ciência da Computação, 2009.

Palavras-Chave: 1. Web 2.0. 2. Plataforma Web. 3. AJAX

Danilo dos Reis Moreira

Um estudo da tecnologia Web 2.0

Monografia apresentada e aprovada em _____ de _____
Pela Banca Examinadora constituída pelos professores.

Márcio de Souza Dias – Presidente da Banca

Ms. Liliane do Nascimento Vale

Acrísio José do Nascimento Júnior

Dedico este trabalho aos meus pais, que sempre me incentivaram a superar desafios e alcançar meus objetivos.

AGRADECIMENTOS

Acima de tudo, a Deus, por me dar forças para superar obstáculos, e me mostrar o caminho certo a seguir.

Aos meus pais, Valdilon Eduardo e Maura Vaz, pelo carinho e educação que me deram, me ensinando a ser pessoa, e pelo zelo de sempre cuidar para que eu tenha um futuro próspero.

A minha irmã Aline Reis, pelas críticas e brincadeiras, e todo companherismo e apoio

A minha namorada Larissa Nassif, pelos incentivos, e principalmente cobranças não deixando eu perder o foco neste trabalho.

Aos colegas da graduação, que sempre estiveram comigo em todos os momentos, sejam de alegria ou tristezas. Em especial, agradeço à Gustavo Batista e Nélio Junior que se tornaram verdadeiros companheiros durante essa jornada.

Aos colegas de curso, que embora não estivessem sempre por perto, apareciam de alguma forma quando eu precisava. Agradeço os bons momentos compartilhados comigo. Em especial Fábio Gomes, Míriam, Andrea, Eustáquio, e todos os outros amigos desta turma.

Ao meu orientador professor Ms. Márcio de Souza Dias, pela paciência e dedicação, pela cobranças e elogios. Por toda a convivência e experiência oferecida durante a elaboração deste trabalho.

Aos outros mestres da graduação, que sempre tinham a acrescentar de alguma forma com suas colocações.

Enfim, a todas as pessoas, mestres e amigos, familiares e colegas, que de qualquer forma, contribuíram para o meu crescimento profissional e pessoal, para minha formação e para minha vida.

Obrigado.

“Um dia, percorrendo o texto de um livro, com a velocidade que a globalização exige, consegui fazer com que os meus olhos capturassem todas as palavras; porém, sobre o conteúdo do qual me recordo tenho somente o registro que segue: ‘você tem a chance de, ao terminar este livro, estar mais velho e mais sábio; porém, se o percorrer apenas com os olhos, estará somente mais velho...’.

Sergio Luiz Tosing

RESUMO

MOREIRA, D. R. Um estudo da tecnologia Web 2.0. Curso de Ciência da Computação, Campus Catalão, UFG, Catalão, Brasil, 2009, 101p.

Desde a criação da *Web*, os sites eram construídos com o intuito de fornecer informação aos usuários. A comunicação seguia um sentido único que ia das empresas para os clientes. O papel destes era o de consumir as informações e utilizar os serviços da forma que lhes eram impostos pelas empresas, sem qualquer autonomia. A *Web 2.0* é a segunda geração de comunidades e serviços baseados na plataforma *Web*, onde a idéia é que o ambiente *on-line* se torne mais dinâmico e que os usuários colaborem para a organização desse conteúdo. Esta monografia busca conceituar o termo *Web 2.0*, além de estudar as técnicas computacionais usadas nas aplicações da *Web 2.0* e as idéias que a envolve. Ao longo deste trabalho serão abordados os principais conceitos, características e críticas desta nova geração. Serão mostrados os serviços, as tecnologias e ferramentas que deram suporte a tal mudança e as conseqüências que elas geraram no desenvolvimento e utilização de aplicativos *Web*.

Palavras-Chaves: Web 2.0, Plataforma Web, AJAX

Sumário

Introdução	1
1 Evolução Histórica	3
1.1 Origem da Internet	3
1.2 A internet no Brasil	8
1.3 A evolução da internet	8
1.3.1 Conteúdo da Internet	9
1.4 A história da modernização da <i>Web</i>	14
2 O conceito web 2.0 - Princípios e Arquitetura	16
2.1 O conceito <i>Web</i> 2.0	16
2.1.1 A Tecnologia - infraestrutura e plataformas de <i>Web</i>	19
2.1.2 Aspectos não tecnológicos - Comunitários e Social	20
2.1.3 Impacto sobre as arquiteturas corporativas	21
2.2 Relações econômicas - Temor e Ambição	21
2.3 Princípios Fundamentais da <i>Web</i> 2.0	22
2.4 Quebra de Paradigmas	23
2.5 Características	24
2.5.1 Participação	24
2.5.2 Cooperação	25
2.5.3 Usabilidade	25
2.5.4 Descentralização de Dados	26
2.5.5 Confiança no conteúdo dos usuários	27
2.5.6 Experiência Rica para o Usuário	27
2.5.7 Personalização	28
2.5.8 Atualizações frequentes	28
2.6 A <i>Web</i> como plataforma	29
2.6.1 O Usuário	29
2.6.2 O Conteúdo	30
2.6.3 Organização	31

2.6.4	Avaliação	31
2.6.5	Distribuição	31
2.6.6	O <i>Design</i>	33
2.6.7	Publicidade e <i>Marketing</i>	34
2.6.8	R.I.A - <i>Rich Internet Applications</i> ou Aplicações de Internet Rica .	35
3	Tecnologias e Serviços	38
3.1	<i>RSS</i> - <i>Real Simple Syndication</i> ou Acesso Fácil as Notícias	38
3.1.1	<i>Resource Description Framework</i> ou Moldura de Descrição de Recursos (RDF)	41
3.2	APIs - <i>Application Program Interfaces</i> ou Interface de Programação de Aplicativos	42
3.2.1	<i>Mashups</i>	42
3.3	<i>Blogs</i>	43
3.4	<i>Wikis</i>	44
3.5	Sites Sociais	45
3.6	JOOMLA!	46
3.7	Wordpress	46
3.8	AJAX	47
3.8.1	JavaScript	49
3.8.2	<i>Cascading Style Sheets</i> ou Folha de Estilo em Cascata (CSS)	50
3.8.3	<i>Document Object Model</i> ou Modelo de Documento Objeto (DOM) .	50
3.8.4	XMLHttpRequest	51
4	Estudos de Caso	53
4.1	Desenvolvimento do <i>WebSite</i> do Curso de Ciência da Computação da Universidade Federal de Goiás, Campus Catalão	53
4.1.1	Plano de Desenvolvimento	54
4.1.2	Distribuição do conteúdo através de RSS	55
4.1.3	Folha de Estilos (CSS)	56
4.1.4	Painel de Administração do <i>Website</i>	57
4.2	<i>Blog</i> Universitário	57
4.2.1	Distribuição do conteúdo através de RSS	58
4.2.2	Participação	59
4.2.3	CSS e Recursos Gráficos	59
4.2.4	Painel de Controle	60
4.3	Avaliação dos Produtos	61
	Conclusão	62

Referências	65
Apêndices	67
A Código Fonte	68
A.1 RSS 2.0 - Site do Departamento de Ciência da Computação	68
A.2 Folha de Estilo (CSS) do Layout do website do DCC	70
A.3 Folha de Estilo (CSS) do Layout do Blog	79

Lista de Figuras

1.1	Rede centralizada [Monteiro, 2005]	4
1.2	Rede descentralizada [Monteiro, 2005]	5
1.3	Taxa de crescimento de domínios do Brasil	9
1.4	<i>Website</i> estático [JavaNoroeste, 2008]	11
1.5	<i>Website</i> dinâmico [JavaNoroeste, 2008]	12
2.1	Modelo com os conceitos e noções da <i>Web</i> 2.0 [O'Reilly, 2005]	18
2.2	Exemplo de uma página personalizada no PageFlakes	28
2.3	A plataforma <i>WEB</i> [Zambonini, 2006]	29
2.4	Editor e compartilhador de imagens online	30
2.5	Identidade Visual do Youtube	32
2.6	A cauda longa [Sena, 2007]	32
2.7	Links Patrocinados - Google	35
2.8	Modelo Síncrono [Crane et al., 2007]	36
2.9	Modelo Assíncrono [Crane et al., 2007]	36
3.1	Simbolos de um RSS [Sampaio, 2007]	38
3.2	Código de indicação do uso de conteúdo RSS	39
3.3	Código de indicação do uso de conteúdo RSS	40
3.4	Código RSS 1.0 [Nottingham, 2005]	41
3.5	- Representação da utilização de APIs [Dohms, 2007]	43
3.6	Visão geral do Sistema Localizador de Congressistas [Dohms, 2007]	44
3.7	Editando um artigo no Wikipédia	45
3.8	Modelo de Aplicações na <i>Web</i> : Clássico e com Ajax [Tosing, 2007]	49
3.9	Instanciamento da classe XMLHttpRequest	52
4.1	<i>WebSite</i> atual do DCC	53
4.2	<i>WebSite</i> do DCC com característica <i>Web</i> 2.0	55
4.3	Visualização do ícone que indica o uso de XML	56
4.4	Página de visualização da RSS e dos agregadores de feeds (marcação)	56
4.5	Painel de Administração do <i>website</i> do DCC	57

4.6	<i>Layout</i> do Blog Universitário	58
4.7	Ícone que indica uso de RSS, azul para geral e vermelho para categorias . .	59
4.8	Formulário pra envio de comentários	59
4.9	Recursos Gráfico do Blog	60
4.10	Editor de texto para envio de <i>post</i>	61

Lista de Tabelas

1.1	Evolução da Internet [JavaNoroeste, 2008]	10
2.1	Comparação entre serviços Web 1.0 e Web 2.0	24
2.2	Descentralização dos dados em diversos setores	26
3.1	Comparação entre RSS1.0 e RSS [Ruby, 2009]	40
3.2	Os elementos-chave do Ajax [Crane et al., 2007]	48

Lista de Abreviações

ARPA - *Advanced Research Projects Agency*
MIT - *Massachussets Institute of Technology*
ARPANET - *Advanced Research Projects Agency Network*
NPL - *Nuclear Physics Laboratory*
BBN - *Bolt, Beranej e Newman*
IMP - *Processador de Interfaces de Mensagens*
UCLA - *University of California at Los Angeles*
SRI - *Stanford Research Institute*
NWG - *Network Working Group*
ICCC - *Conferência Internacional de Comunicação entre Computadores*
WWW - *World Wide Web*
TCP/IP - *Transmission Control Protocol/ Internet Protocol*
RNP - *Rede Nacional de Ensino e Pesquisa*
FAPESP - *Fundo de Amparo à Pesquisa do Estado de São Paulo*
DNS - *Domain Name System*
HTML - *HyperText Markup Language*
SGBD - *Sistema Gerenciador de Banco de Dados*
API - *Interface de Programação de Aplicativos*
REST - *Representational State Transfer*
RSS - *Really Simple Syndication*
SOA - *Arquitetura Voltada a Serviços*
XML - *Extensible Markup Language*
AJAX - *Asynchronous Javascript and XML*
TI - *Tecnologia da Informação*
PDA - *Assistente Pessoal Digital*
RDF - *Resource Description Framework*
PhP - *Hypertext Preprocessor*
SGC - *Sistema de Gerenciamento de Conteúdo*
CMS - *Content Management Systems*
DOM - *Document Objetc Model*

CSS - *Cascading Style Sheets*

W3C - *World Wide Web Consortium*

Introdução

Numa sociedade cada vez mais voltada para as novas tecnologias, a internet passou a ser uma ferramenta indispensável na vida de milhões de pessoas em todo o mundo, no entanto são poucos aqueles que sabem como nasceu este “fenômeno” , quem foram os responsáveis ou mesmo como ela foi evoluindo ao longo do tempo.

Desde a criação da *Web*, os sites eram construídos com o intuito de fornecer informação aos usuários. A comunicação seguia um sentido único que ia das empresas para os clientes. O papel destes era o de consumir as informações e utilizar os serviços da forma que lhes eram impostos pelas empresas, sem qualquer autonomia.

Com o passar do tempo, ocorreu uma popularização da Internet e os usuários passaram a assumir um comportamento de destaque, participando efetivamente do processo de desenvolvimento de aplicações para a *Web*. Esta mudança acarretou numa evolução dos serviços para atender a tal participação. As empresas passaram a ter o usuário como principal objetivo e começaram a usá-lo para contribuir com a produção e organização do conteúdo. Desta forma, ocorreu uma revolução social na *Web*, denominada de *Web 2.0*.

Mas a *Web 2.0* é uma tecnologia que realmente traz grandes evoluções de acessibilidade e conteúdo, ou ela é apenas um jogo de marketing para atingir os usuários?

Este trabalho tem como objetivo, esclarecer todas as dúvidas pertinentes a *Web* em si, desde seu surgimento até os dias atuais, quais são suas perspectivas de futuro, o que representa a *Web 2.0* comparada a *Web* clássica, o que muda a partir de então, enfim como essa tecnologia vem sendo usada, se seus propósitos estão sendo alcançados, além de propor melhorias a possíveis deficiências, estudar a aceitação dos usuário e quais mudanças já foram feitas e ainda podem ser realizadas para alcançar um melhor resultado. Visando alcançar estes objetivos, apresentam-se as metas propostas para a apresentação deste trabalho:

- Estudar e compreender a história da Internet no Brasil e no mundo, desde seu surgimento até os dias atuais, fazendo uma projeção de futuro para a mesma.
- Estudar e compreender a tecnologia *Web 2.0*, bem como seus princípios, tecnologias envolvidos, e objetivos.
- Ser capaz de identificar um *website* enquadrado dentro dessa tecnologia;

- Mostrar vantagens e desvantagens quanto ao uso dessa tecnologia, fazendo uma comparação entre a *Web* clássica e a *Web* 2.0.
- Aplicar os conhecimentos adquiridos ao longo do estudo para o aprendizado de novos tópicos relacionados a *Web*.
- Desenvolver um *blog* para o Campus Catalão onde os professores, diretores ou demais pessoas ligadas ao Campus, poderão postar sua idéias e projetos permitindo ao aluno e a toda comunidade opinar acerca do assunto proposto, gerando assim uma maior interação entre os cursos e as novas tendências de pesquisa no país, ou mesmo sobre problemas da comunidade em geral.
- Desenvolver o *website* do Departamento de Ciência da Computação da Universidade Federal de Goiás, Campus Catalão dentro dos padrões da *Web* 2.0.

O trabalho descrito neste texto está dividido em 4 capítulos, sendo o capítulo 1 um histórico da internet, desde seu nascimento até o surgimento do termo *Web* 2.0.

O capítulo 2 aborda todos os conceitos acerca do termo *Web* 2.0, o que ela propõe de inovador, quais seus princípios, arquiteturas e características, além de apresentar os conceitos a cerca da *Web* como plataforma, quais as mudanças causadas nos diversos elementos da *Web* e quais são as considerações que estão sendo analisadas pelos profissionais.

No capítulo 3 são apresentados os serviços oferecidos na *Web* 2.0, além de um estudo detalhado de AJAX, e das tecnologias envolvidas nele.

O capítulo 4 traz a aplicação de tudo que foi estudado no desenvolvimento de um blog para Universidade Federal de Goiás, Campus Catalão, e de um *website* para o Departamento de Ciência da Computação, bem como as ferramentas utilizadas e todo trabalho necessário para a construção dos mesmos.

Por fim, traz as conclusões obtidas com o desenvolvimento deste trabalho, bem como as dificuldades encontradas para a realização do mesmo e sugestões para trabalhos futuros relacionados ao tema.

Ainda é possível encontrar o código fonte das aplicações desenvolvidas nos apêndices desta monografia.

Capítulo 1

Evolução Histórica

A Internet tem revolucionado o mundo dos computadores e das comunicações como nenhuma outra invenção foi capaz de fazer. Ela é um mecanismo de disseminação da informação e divulgação mundial, além de ser um meio para colaboração e interação entre indivíduos e seus computadores, independentemente de suas localizações geográficas.

A história da Internet é complexa e envolve muitos aspectos - tecnológicos, organizacionais e comunitários, sua influência atinge não somente o campo técnico das comunicações via computadores, mas toda a sociedade, na medida em que usa-se cada vez mais ferramentas online para fazer comércio eletrônico, adquirir informação e operar em comunidade.

1.1 Origem da Internet

Em 1957, durante o auge da guerra fria¹ foi lançado pelos soviéticos o primeiro satélite *Sputnik*, ultrapassando os EUA na corrida para o espaço, então como resposta a esse lançamento, o Departamento de Defesa dos EUA criou a *Advanced Research Projects Agency* (ARPA²), afim de acelerar o progresso tecnológico do país.

Assim, sobe uma ameaça de um bombardeio nuclear por parte da União Soviética, o EUA notou que enfrentava um problema de falhas na comunicação caso esse bombardeio se realizasse, sendo que a tecnologia utilizada era de computadores de grande porte acessíveis por terminais, onde um sistema central guardava todas as informações, e os terminais acessavam-nas através de ligação por cabo, como mostra a figura 1.1 [Monteiro, 2005].

Obviamente, esse tipo de sistema era muito vulnerável, pois bastaria bombardear o sistema central para desativar toda a rede. Então, em 1962, foi publicado pela *Rand*

¹A Guerra Fria foi a designação atribuída ao conflito político-ideológico entre os Estados Unidos (EUA), defensores do capitalismo, e a União Soviética (URSS), defensora do socialismo, compreendendo o período entre o final da Segunda Guerra Mundial (1945) e a extinção da União Soviética (1991).

²ARPA - Agência de Projetos de Pesquisa Avançada

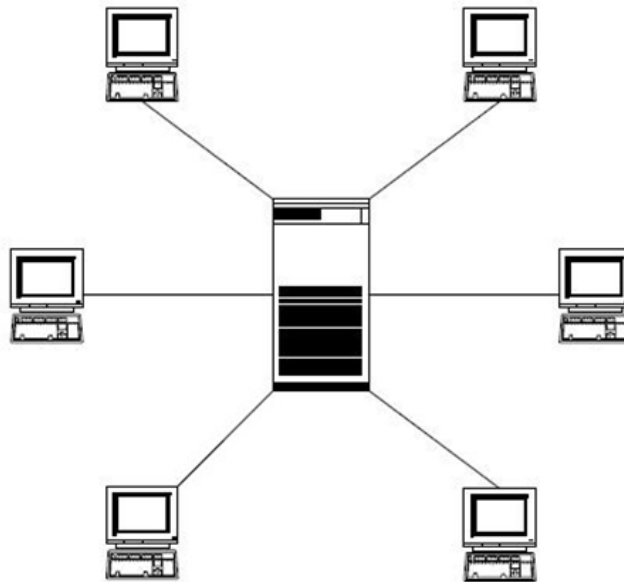


Figura 1.1: Rede centralizada [Monteiro, 2005]

Corporation o relatório “*On distributed Communications Network*”³ escrito por Paul Baran [Monteiro, 2005]. Esse relatório, patrocinado pela Força Aérea Americana, apresentava soluções para proteger os sistemas de comunicações militares.

Baran apresentava alguns conceitos baseados no princípio de não existir autoridade central, ou seja, uma rede descentralizada, figura 1.2, onde todos os nós da rede teriam o mesmo status, com permissões de originar, passar e receber mensagens.

Em agosto de 1962, foram observados os primeiros registros de interações sociais escritos por J.C.R. Licklider, do MIT - *Massachusetts Institute of Technology*, onde se discute o conceito da “*Galatic Network*”⁴.

Licklider previa vários computadores interconectados globalmente, onde todos teriam acesso a dados e programas de qualquer local rapidamente. Em essência, o conceito foi muito parecido com a Internet de hoje. Licklider foi o primeiro gerente do programa de pesquisa de computador da ARPA, começando em outubro de 1962. Enquanto trabalhava neste projeto, ele convenceu seus sucessores Ivan Sutherland, Bob Taylor e Lawrence G. Roberts da importância do conceito de redes computadorizadas. [Monteiro, 2005]

Licklider escreveu vários artigos sobre redes, onde em um deles com co-autoria de Robert lia-se: [Licklider e Robert, 1968], “*In a few years men will be able to communicate more effectively through a machine than face to face... When mind interacts, new ideas emerge.*”, ou seja, “Em poucos anos o homem será capaz de comunicar-se mais efetivamente através de uma máquina do que face a face, quando a mente interage, novas idéias emergem.” já prevendo o surgimento de uma rede de interação entre usuários.

³Redes de Comunicações Distribuídas

⁴Rede Galática

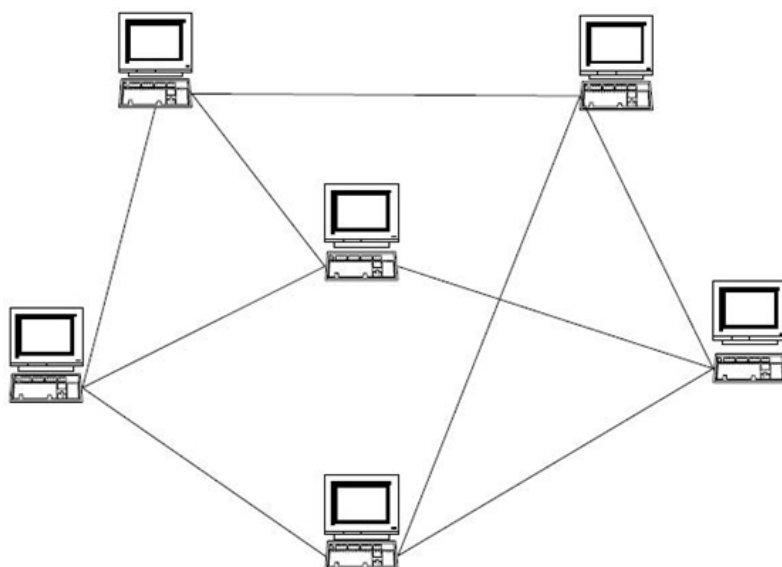


Figura 1.2: Rede descentralizada [Monteiro, 2005]

Leonard Kleinrock, do MIT, publicou o primeiro documento sobre a teoria de trocas de pacotes em 1961 e o primeiro livro sobre o assunto em 1964. [Licklider e Robert, 1968], Ele convenceu Lawrence G. Roberts⁵ da possibilidade teórica das comunicações usando pacotes ao invés de circuitos, ainda usado atualmente, onde as mensagens são divididas em pacotes (*packets*), que são pedaços da mensagem, para serem enviados entre computadores.

Cada um dos pacotes contém o endereço do destinatário, do remetente e a informação a enviar, como se fosse uma correspondência propriamente dita. Quando todos os pacotes chegam ao destino, a mensagem original é reconstruída. Mesmo que, por qualquer motivo, um dos pacotes se perca, poderá ser reenviado de forma a completar a informação original.

A comunicação usando pacotes representou um grande passo para tornar possíveis as redes de computadores. O outro grande passo foi fazer os computadores se comunicarem. Em 1965, Lawrence G. Roberts e Thomas Merrill conectaram um computador TX-2 no Instituto de Tecnologia de Massachussets, nas cidades Cambridge e Lexington, com um Q-32 na Universidade da Califórnia, na cidade de Berkley, com uma linha discada de baixa velocidade, criando assim o primeiro computador de rede do mundo [Licklider e Robert, 1968]. Esse experimento resultou na comprovação de que computadores poderiam trabalhar bem juntos, rodando programas e recuperando dados quando necessário em máquinas remotas, porém constatou também que o circuito do sistema telefônico era totalmente inadequado para o intento. Assim, confirmou-se a convicção de Kleinrock sobre a necessidade de trocas de pacotes [Monteiro, 2005].

No ano de 1966, Lawrence G. Roberts começou a trabalhar na ARPA afim de desen-

⁵Criador da ARPANET

volver os conceitos de rede de computadores, e em 1967 publicou seu plano para uma rede, a ARPANET [Monteiro, 2005].

Na *Proceedings of the Spring Joint Computer Conference*⁶ onde ele apresentou este trabalho, houve também uma apresentação sobre o conceito de redes de pacotes desenvolvido pelos ingleses Donald Davies e Roger Scantlebury, da NPL - *Nuclear Physics Laboratory*. Scantlebury conversou com Roberts sobre o trabalho da NPL e do trabalho de Paul Baran e outros da *RAND Corporation*⁷ [Monteiro, 2005].

Após Lawrence G. Roberts e o grupo da *RAND Corporation* terem refinado a estrutura e especificações para a ARPANET, em agosto de 1968 foi selecionado um grupo, chamado BBN - *Bolt, Beranej e Newman* dirigido por Frank Heart para o desenvolvimento do processador de interfaces de mensagens (IMP) que era um dos componentes-chave do projeto.

Paralelo ao trabalho do grupo BBN nos IMPs, Roberts em conjunto Howard Frank e seu grupo da *Network Analysis Corporation*, desenvolveu a topologia e economia da rede, onde Kleinrock⁸ e sua equipe preparou o sistema de mensuração da rede na UCLA - *University of California at Los Angeles*.

Em setembro de 1969 foi escolhido para ser o primeiro nó (ponta) da ARPANET o Centro de Mensuração de Rede da UCLA, sendo instalado pela BBN o primeiro o IMP e conectando o primeiro servidor de computador. O SRI - *Stanford Research Institute*, foi o segundo nó ou ponta.

SRI passou a manter as tabelas de “*Host Name*” para o mapeamento dos endereços. Um mês depois, quando SRI foi conectado à ARPANET, a primeira mensagem entre servidores foi enviada do laboratório de Kleinrock para o SRI. Posteriormente, dois outros “nós” foram acrescentados, sendo eles na UC Santa Barbara (UCSB) e a Universidade de Utah. Este dois nós incorporavam projetos de aplicações visuais, com Glen Culler e Burton Fried na UCSB investigando métodos de uso de funções matemáticas para restaurar visualizações na rede e Robert Taylor e Ivan Sutherland em Utah investigando métodos de representação em terceira dimensão na rede.

No final de 1969, quatro servidores estavam conectados na ARPANET, onde os “nós” conseguiam transmitir dados através de linhas dedicadas de alta velocidade. Embora permitisse que os seus utilizadores partilhassem os recursos uns dos outros verificou-se que o tráfego principal não era de resultados de computação de longa distância mas sim de notícias e de mensagens pessoais. Os usuários estavam a utilizar a ARPANET para

⁶Conferência de Computação Unificada da Primavera

⁷A *RAND Corporation*, com sede na Califórnia, é uma instituição sem fins lucrativos que realiza pesquisas para contribuir com a tomada de decisões e a implementação de políticas no setor público e privado

⁸Leonard Kleinrock, é um cientista da computação e professor da UCLA - Universidade da Califórnia em Los Angeles

colaborar em projetos, para trocar notas de trabalhos, e eventualmente, conversar sobre assuntos fúteis [Monteiro, 2005].

Durante os anos 70 a rede cresceu bastante. A sua estrutura descentralizada facilitou bastante a sua expansão. Computadores foram rapidamente adicionados à ARPANET e os grupos de trabalho desenvolveram um protocolo servidor a servidor funcionalmente completo e outros softwares de rede [Monteiro, 2005].

Em dezembro de 1971, o *Network Working Group* (NWG) gerenciado por S. Crocker, concluiu o primeiro protocolo servidor a servidor da ARPANET, chamado *Network Control Protocol* (NCP). De 1971 a 1972, os usuários da rede finalmente puderam começar a desenvolver as suas aplicações.

Em outubro de 1972, organizou-se uma grande e bem sucedida demonstração sobre a ARPANET na Conferência Internacional de Comunicação entre Computadores (ICCC). Esta foi a primeira demonstração pública da nova tecnologia de rede para o público. Foi também em 1972 que o correio eletrônico, considerado a primeira aplicação “hot”, foi introduzido [Monteiro, 2005].

Em março de 1972, Ray Tomlinson, da BBN, escreveu o *software* básico de e-mail com as funções de “*send/enviar*” e “*read/ler*”, motivado pela necessidade dos desenvolvedores da ARPANET de ter um fácil mecanismo de coordenação. Em julho, Roberts expandiu a utilidade do e-mail escrevendo o primeiro programa utilitário de e-mail para listar, ler seletivamente, arquivar, encaminhar e responder a mensagens. Dali, o correio eletrônico se tornou a maior aplicação de rede por mais de uma década. Este foi o prenúncio do tipo de atividade que se ve hoje na WWW, ou seja, o enorme crescimento de todos os tipos de aplicações e utilitários agregados pessoa-a-pessoa [Monteiro, 2005].

No início dos anos 80, criou-se os protocolos TCP/IP (*Transmission Control Protocol/Internet Protocol*), que fixaram as normas técnicas para a transmissão de informações através da rede, permitindo identificar os usuários por endereços e nomes de domínio. Daí passou a ser possível a intercomunicação entre distintas redes, nascendo então a Internet [Sartori, 2001].

Até então, a internet era restrita a ambientes acadêmicos e científicos. Os EUA liberaram a rede para uso comercial no ano de 1987, surgindo assim, em 1992 as primeiras empresas de provedor de acesso comercial. Porém, a massificação da internet veio em 1993 com a criação da WWW - *World Wide Web*, que permite a transmissão em linha de páginas *Web*, integrando pela primeira vez textos e gráficos, ou seja, com um simples “clique” de *mouse* era possível saltar de um pedaço da informação para outro, não importando onde ela estivesse [Kaneco et al., 1999].

Ainda em 1993, o estudante de computação Marc Andreessen, da Universidade de Illinois, desenvolveu um *software* chamado *Mosaic*, que permite inserir imagens junto com o texto. Com o sucesso do *software*, empresas e o público em geral tomaram consciência

da Internet e suas possibilidades.

A internet se distanciou muito de seu objetivo inicial, onde era usado somente para assuntos militares e científicos, tornando-se um meio de interesses comerciais, e diminuindo distância entre usuários.

1.2 A internet no Brasil

O Brasil deu início a sua participação na Internet no ano de 1988, por iniciativa de comunidades acadêmicas de São Paulo e do Rio de Janeiro, conectando a rede por três pontos, sendo um em São Paulo, na FAPESP⁹ e outros dois no Rio de Janeiro [Sartori, 2001].

Em 1989, foi criada a RNP - Rede Nacional de Ensino e Pesquisa com objetivo de construir uma infra-estrutura de rede Internet nacional de âmbito acadêmico. A Rede Nacional de Pesquisa, como era chamada em seu início, tinha também a função de disseminar o uso de redes no país [Carvalho, 2006].

Em maio de 1995, teve início a abertura da Internet comercial no país. Neste período, a RNP passou por uma redefinição de seu papel, estendendo seus serviços de acesso a todos os setores da sociedade. Com essa reorientação de foco, a RNP ofereceu um importante apoio à consolidação da Internet comercial no Brasil. Foi criado o Centro de Informações Internet/BR para dar suporte no surgimento de provedores e usuários da rede. Mais de 3.000 questões relativas à Internet foram respondidas em seu primeiro ano de funcionamento. Inúmeras empresas fabricantes de bens de informática, tais como Compaq, Equitel, IBM, Philips etc., passaram a oferecer apoio concreto à RNP, fornecendo equipamentos, *software* e, mesmo, financiando atividades diretas do projeto [Carvalho, 2006].

A RNP promove o desenvolvimento tecnológico de redes no Brasil. Ela apóia a utilização de redes Internet como facilitadoras do progresso da ciência e da educação em geral e incentiva o desenvolvimento de tecnologias na área de redes e aplicações inovadoras.

Na figura 1.3, nota-se quão grande foi a evolução da internet no Brasil desde sua implantação. Os dados fornecidos na imagem são da Registro.br¹⁰, que é o órgão responsável no Brasil pela delegação de DNS (*Domain Name System*) para que os sites possam operar e fazer parte da rede.

1.3 A evolução da internet

A internet passou por um processo de evolução muito rápido, tanto se comparado ao número de usuários conectados quanto ao nível das aplicações disponibilizados pela

⁹Fundo de Amparo à Pesquisa do Estado de São Paulo

¹⁰<http://www.registro.br>

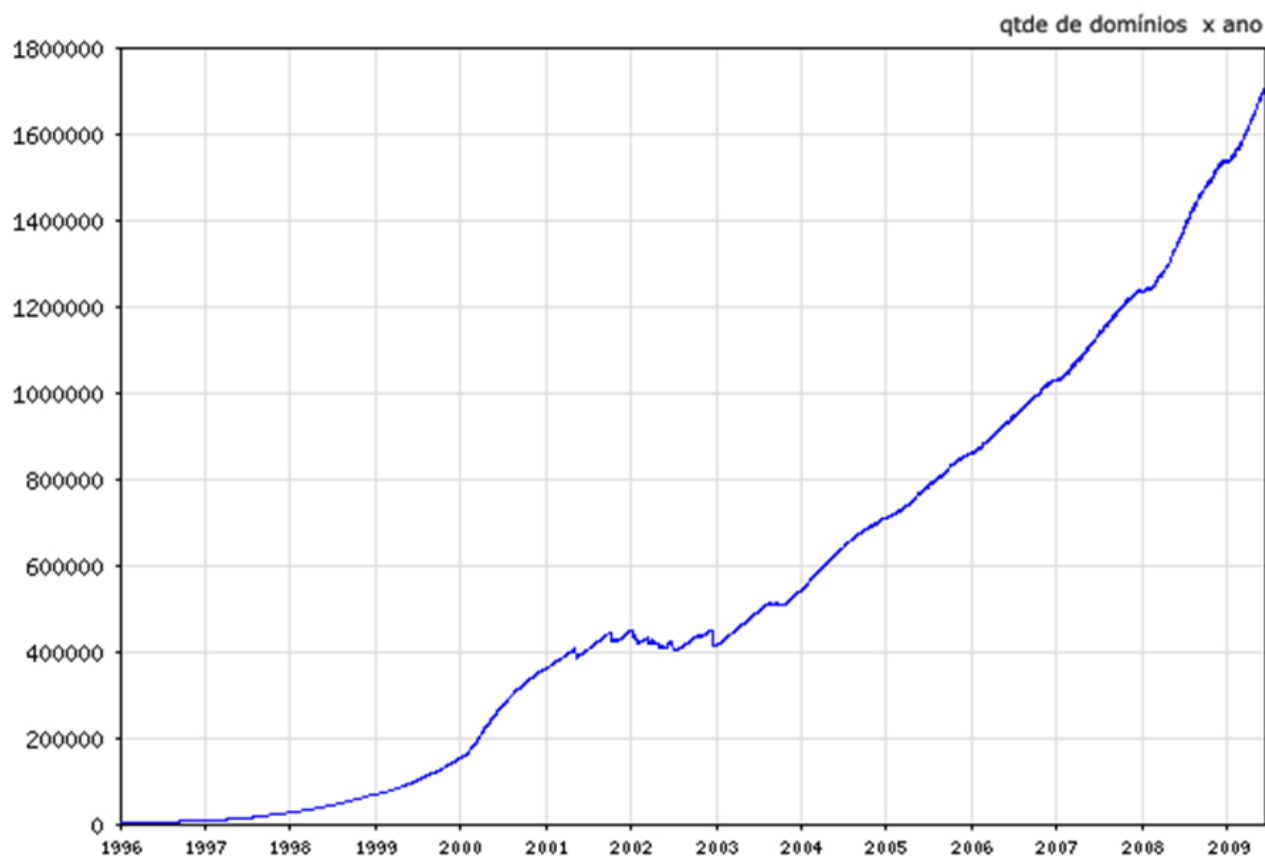


Figura 1.3: Taxa de crescimento de domínios do Brasil

internet.

A tabela 1.1 apresenta cada período da evolução da internet, sendo elas a *Web* Estática, Dinâmica, Transacional e Semântica. Ela descreve a evolução da *Web* em diferentes aspectos: na linha 2, vemos as diferentes linguagens de codificação; na linha 3 as diferentes formas de criação das páginas e o uso destas páginas por pessoas e/ou aplicações; na linha 4 a evolução dos paradigmas; na linha 5 a evolução da forma do usuário pensar; na linha 6 a evolução das aplicações; e na linha 7 exemplos das áreas de domínios comuns e as principais aplicações. Nas colunas 2 a 4, podemos observar as diferentes gerações da *Web*.

Nota-se na tabela, que as linguagens não foram trocadas por novas, e sim acrescidas de novas tecnologias, onde a cada período da evolução uma nova forma de desenvolver um website é criada, surgindo novas técnicas. O pensamento também muda, onde a cada período a *Web* tem um objetivo novo, uma nova forma de oferecer serviços.

1.3.1 Conteúdo da Internet

Conteúdo *Web* é tudo o que aparece em uma página *Web*: texto, gráficos, campos de formulários, hiperlinks para outras páginas, botões de navegação, menus, etc.

	Estática	Dinâmica	Transacional	Semântica
Código	HTML, CGI, Perl, ...	+ SGBDRs, JSP, APS, Java, ...	+ XML, S2EE, .NET, ...	+ RDF, OWL, ...
Criação	Criação manual por pessoas. Uso por pessoas.	<i>Templates</i> criados por pessoas. Uso por pessoas.	+ Criação por aplicações baseadas em esquemas fixos. Uso por aplicações e Pessoas.	Criação por aplicações baseadas em modelos. Uso por aplicações, dispositivos e pessoas
Paradigma	Informação, um grande jornal.	Jornal se transforma em catálogo.	Catálogo se transforma em plataforma transacional.	Plataformas se conectam.
Pensamento	Navegar	Recuperar/ Atualizar	Interagir	Interoperar
Ex. de Áreas	<i>Marketing</i>	Vendas	Serviços	Integração
Principais Aplicações	Navegador	Busca / Gerenciamento de conteúdo / Servidores de Aplicações <i>Web</i>	Portais / Processos de integração / Serviços <i>Web</i>	Agentes Pessoais / Aplicações IP / <i>Advisors</i> / Ferramentas Cognitivas

Tabela 1.1: Evolução da Internet [JavaNoroeste, 2008]

Nos primórdios da Internet, o conteúdo das páginas *Web* era quase que inteiramente formado por código HTML estático e documentos de texto. Um documento estático é simplesmente qualquer página *Web* que seja salva em disco e repassada através de um *browser* sem sofrer quaisquer mudanças. Ainda hoje, diversas das páginas são estáticas, embora a grande maioria já tenha evoluído para *Web* dinâmica ou semântica. [Monteiro, 2005]

Em um *website* estático, todo o conteúdo das páginas é colocado manualmente por desenvolvedores *Web* profissionais. Esse processo é conhecido como construção da página em tempo de projeto, porque as páginas são totalmente construídas enquanto o site está sendo desenvolvido. Sites desse tipo são desenvolvidos e mantidos por profissionais experientes. Em geral, seu custo inicial é menor, porém todas as futuras mudanças têm que ser realizadas por profissionais *Web*. Portanto, um *website* estático pode ser mais caro para mantê-lo atualizado, especialmente quando são necessárias alterações frequentes nas páginas [Primo, 2007].

A figura 1.4 mostra como se dá a *Web* estática, que consistem de páginas HTML (*Hypertext Markup Language*) as quais não mudam a menos que o webmaster modifique as marcações (*tags*) diretamente dentro da página. *Websites* estáticos geralmente possuem problemas de conteúdo devido a grande parte da informação não ser atualizada. O problema de falta de atualização pode levar um usuário a não retornar para uma página.

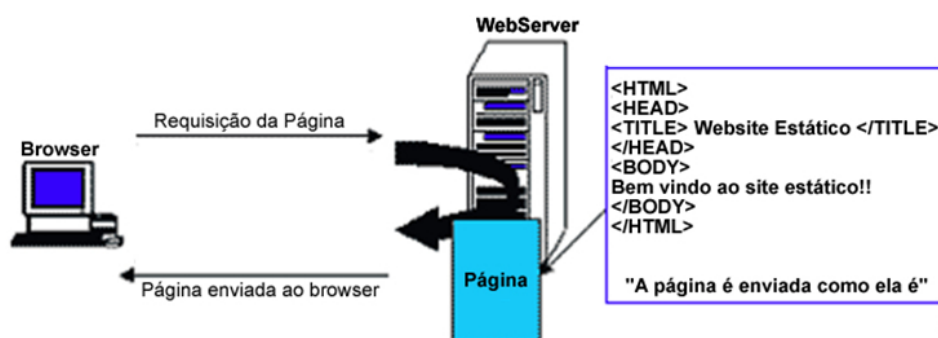


Figura 1.4: *Website* estático [JavaNoroeste, 2008]

As páginas de um *website* dinâmico são construídas “*on the fly*”, ou seja, no momento em que as mesmas são requisitadas por um browser. Sites dinâmicos, embora também sejam desenvolvidos por profissionais, podem ser mantidos diretamente pelo usuário cliente. Seus custos iniciais são mais altos, bem como o tempo de desenvolvimento, mas não é necessário pagar um profissional para fazer as freqüentes alterações necessárias nas páginas [JavaNoroeste, 2008].

Uma página *Web* dinâmica é qualquer página que tenha seu conteúdo alterado por um programa ou *script* no momento em que a mesma é requisitada. Linguagens de programação comumente utilizadas para a construção de páginas dinâmicas são o VBScript, PHP e JSP, além de Javascript.

Na figura 1.5 mostra-se como se dá a construção “*on the fly*”, pois ao ser acessado o conteúdo, ele é exibido dinamicamente através de banco de dados ou pelo próprio servidor.

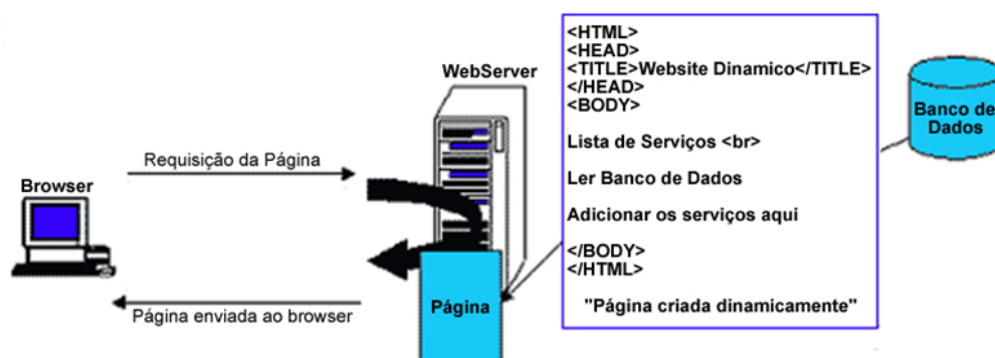


Figura 1.5: *Website* dinâmico [JavaNoroeste, 2008]

Exemplos comuns de página dinâmica são aquelas que apresentam um contador de acessos ou a data e hora corrente. Outro exemplo são as páginas de resultados retornadas quando um usuário submete um formulário *Web*. O formulário em si pode ser estático ou dinâmico, mas a página de resultados deve obrigatoriamente ser dinâmica, a menos que o formulário seja apenas algum exercício e não tenha finalidade.

Pressupõe-se que quando se introduzem dados em um formulário e este é processado, os resultados devem ser variáveis. Com frequência os formulários são a principal interface para vários tipos de consultas e controlam diretamente quais dados serão apresentados como resultado da sua submissão. Mesmo se o formulário simplesmente coleta informações para enviar um email ou somente salva dados em um banco, espera-se que uma página de resultado seja apresentada indicando se a operação foi ou não bem sucedida.

Outros exemplos de páginas dinâmicas são aquelas que permitem a um usuário configurar preferências com relação ao tipo de informação apresentada. A maioria dos mecanismos de busca atuais apresentam anúncios baseados no conteúdo que o usuário está pesquisando, onde os produtos anunciados de alguma forma se relacionam às palavras que o usuário está pesquisando.

Finalmente, alguns sites são inteiramente dinâmicos porque todo o seu conteúdo é armazenado em um banco de dados e montado em uma página *Web* “*on the fly*”. Tipicamente, sites desse tipo visam reduzir o trabalho de manutenção das páginas, com usuários que não conhecem a linguagem HTML digitando o conteúdo em banco de dados via uma interface amigável. Quando o conteúdo é recuperado ele é misturado com um *template* que fornece elementos comuns ao site como um todo e recebe adicionalmente componentes que controlam o estilo da apresentação da página.

Enfim, *websites* dinâmicos consistem de páginas HTML que são criadas no servidor *Web* antes de serem enviadas ao usuário. A maioria dos *websites* dinâmicos usa um sistema

gerenciador de banco de dados (SGBD) relacional para criar o conteúdo dinâmico. Sites dinâmicos são em geral mais caros para desenvolver, porém mais baratos para manter. As informações em geral estão atualizadas e podem ser otimizadas para a apresentação para cada usuário específico.

Um site pode incluir uma combinação de conteúdo estático e dinâmico. A informação que não muda frequentemente é melhor manipulada estaticamente, enquanto a informação que sofre alterações frequentes é melhor manipulada dinamicamente. Essa abordagem mista permite que os *webmasters*¹¹ concentrem a maior parte de seu tempo em adicionar conteúdo e menos tempo em manutenção.

De uma forma geral, os usuários terão uma experiência mais agradável e interativa em um *website* que contenha algum conteúdo dinâmico. Tal fato deve-se a possibilidade que o usuário tem de solicitar informações específicas para os seus interesses.

A *Web Transacional* representa a era de serviços *Web*, onde se faz uso de entidades estáticas e dinâmicas, afim de causar uma maior dinamização das ações e proporcionar ao usuário uma maior interação se comparada somente a web estática ou somente a dinâmica [Tavares et al., 2008].

Em um servidor *Web Transacional* trabalha-se dentro de 3 entidades básicas, para proporcionar facilidade no uso do serviço *Web* disponibilizado, que são produtos, clientes e sessões. A primeira entidade compreende os objetos das transações, havendo dois tipos de dados a respeito deles: estático e dinâmico. Dados estáticos modelam informações que não são afetadas pelos serviços providos pelo servidor, como a descrição de um livro, ou as características de um equipamento eletrônico. Dados dinâmicos, por outro lado, modelam informações que são alteradas pelas operações executadas durante o provimento dos serviços, como o estoque dos produtos, ou a identidade do comprador [Tavares et al., 2008].

A segunda entidade representa os clientes, que também demandam o armazenamento de dados estáticos e dinâmicos. Neste caso, informações como nome e endereço são estáticas, enquanto outros atributos, como o conteúdo da cesta de compras, são dinâmicos. A identificação de dados estáticos e dinâmicos é crucial para a distribuição dos serviços em servidores agrupados. Enquanto dados estáticos podem ser facilmente replicados nos servidores agrupados, acessos a dados dinâmicos devem ser controlados de forma a evitar inconsistências [Tavares et al., 2008].

A terceira entidade representa a relação entre as duas primeiras entidades, isto é, a interação entre clientes e produtos. Essa interação é sintetizada pela sessão de usuário no contexto de servidores de comércio eletrônico, a qual é construída à medida que o servidor responde às requisições do usuário. Uma sessão de servidor de comércio eletrônico combina referências a dados tanto estáticos quanto dinâmicos de clientes e produtos que estejam

¹¹WebMasters são os desenvolvedor de sites

associados a sua interação [Tavares et al., 2008].

A *Web Semântica* surge como uma possível solução para a estruturação semântica dos dados na *Web*, viabilizando o processamento da informação por parte das máquinas [Oliveira, 2008].

A *Web Semântica* visa incorporar semântica às informações. Isso proporcionará não somente aos usuários entenderem as informações como também as máquinas. Ela pretende fornecer estruturas e dar significado semântico ao conteúdo das páginas *Web*, criando um ambiente onde agentes de *softwares* e usuários possam trabalhar de forma cooperativa.

Neste novo contexto, segundo [Berners-Lee et al., 2001], a *Web* será capaz de representar associações entre coisas que em princípio poderiam não estar relacionadas. Os computadores necessitam ter acesso a coleções estruturadas de informações (dados e metadados) e de conjuntos de regras de inferência que ajudem no processo de dedução automática para que seja administrado o raciocínio automatizado, ou seja, a representação do conhecimento.

1.4 A história da modernização da *Web*

No final da década 90, a Internet comemorava momentos de grande euforia com as ações “PontoCom”. Esses papéis atingiam altos patamares, onde valiam mais pelo “potencial de futuro” que tinham do que pelo lucro que as empresas geravam no presente. Algumas ações do setor na bolsa de valores aqui no Brasil chegaram a apresentar valorização de 300% em um único mês. Tudo por conta do otimismo que levava os especuladores a esperar que se repetisse no Brasil o que acontecia nos EUA (com papéis de empresas como a AOL, *Amazon* e *Yahoo!* atingindo alta de 80% ao ano na *Nasdaq*¹² contra apenas 22% da *DowJones*¹³) [PUC-RIO, 2008].

O ano de 2000 chegou, e com ele, o “Estouro da Bolha da Internet”. A *Nasdaq* apresentou então seu pior resultado, com ações de algumas empresas perdendo mais de 90% do valor. Os prejuízos chegaram à casa dos trilhões de dólares e a “Nova Economia” perdia o seu “*Glamour*”. Várias empresas fecharam suas portas e termos como Internet e PontoCom passaram a ser associados à medo, crise e demissão [PUC-RIO, 2008].

Entretanto negócios bem-estruturados, com idéias sólidas e viáveis, conseguiram sobreviver. Assim, no ano de 2004 em uma conferência de *brainstorming*¹⁴ entre as empresas *O’Reilly Media* e *Live International Dale Dougherty*, onde Tim O’Reilly, um dos pioneiros na *Web*, e fundadores da *O’Reilly Media*, notou que, ao contrário de uma decadência, a

¹²Bolsa de Valores do EUA

¹³Bolsa de Valores do EUA

¹⁴O *brainstorming* é uma técnica de dinâmica de grupo desenvolvida para explorar a potencialidade criativa do indivíduo, colocando-a a serviço de seus objetivos

Web estava mais importante do que nunca, apresentando instigantes aplicações novas e sites eclodindo com surpreendente regularidade, surgindo então o conceito *Web 2.0* [PUC-RIO, 2008].

Neste mesmo ano, o termo “*Web 2.0*” veio à tona com mais de 9,5 milhões de citações no *Google*¹⁵, porém, ainda com certa resistência por parte de alguns desenvolvedores, que menosprezavam a expressão *Web 2.0*, como sendo apenas um jogo de marketing para salvar as empresas do estouro da bolha.[Primo, 2007].

Contudo, a massificação da *Web 2.0* se tornou evidente, com a publicação em 2005 do relatório “*Web 2.0 - Principles e Best Practices*”¹⁶, que é fruto da conferência de 2004, onde Tim O’Reilly relata as melhores práticas e consequências no mundo empresarial da aplicação dos princípios da *Web 2.0*. Com este relatório as empresas puderam se familiarizar com o que propõe a tecnologia *Web 2.0*, entendendo conceitos, e práticas aplicáveis a *Web 2.0*, que apesar do termo sugerir uma nova versão da WWW, não se refere a uma atualização de qualquer especificação técnica, mas sim a uma mudança na relação entre o usuário e a aplicação *Web*, que se tornará mais simples, porém mais completa e atrativa ao usuário [O’Reilly, 2005].

A *Web 2.0* não se refere somente na utilização de novas interfaces de usuários, ela visa melhorar a criatividade, o compartilhamento de informações, e principalmente a colaboração entre usuários, mantendo as aplicações como “betas perpétuos”, ou seja, a aplicação seria uma versão beta para sempre, por estar em constante atualização, provinda da arquitetura de participação proposta, onde os próprios usuários compartilham as informações e deixam o conteúdo mais completo [Primo, 2007].

A *Web 2.0* propõe uma experiência de uso semelhante à de aplicativos para *desktop*, frequentemente fazendo uso de uma combinação de tecnologias surgidas no final da década de 1990, que incluem *Web services APIs* (1998), *AJAX* (1998), *Web syndication* (1997), entre outras. Estas tecnologias aumentaram a velocidade e a facilidade de uso de aplicativos *Web*, sendo responsáveis por um aumento significativo no conteúdo, seja ele colaborativo ou meramente expositivo, existente na Internet. E também permitiu que usuários que antes não tinham conhecimento necessário para publicar algo na internet, agora consigam com facilidade publicar qualquer conteúdo, devido à semelhança entre os aplicativos *Web* e os *softwares* para *desktop*.

¹⁵<http://www.google.com.br>

¹⁶Tradução: *Web 2.0 - Princípios e Melhores Práticas*

Capítulo 2

O conceito web 2.0 - Princípios e Arquitetura

A *Web 2.0* não é apenas um conjunto de tecnologias, mas também de atributos que têm uma dimensão social: novos modelos corporativos, conteúdos desenvolvidos por usuários e metadados gerados por usuários, processos corporativos mais abertos e transparentes, simplicidade de *design* e recursos, produtos e processos descentralizados e participativos [Primo, 2007].

Ao contribuir para permitir a inovação descentralizada, a *Web 2.0* catalisa mudanças rápidas, impulsionadas por usuários/consumidores que poderão acelerar o crescimento para as empresas que a explorarem. As comunidades da internet proporcionam interações novas e mais ricas entre funcionários, parceiros e clientes corporativos que poderão apoiar ou ameaçar a empresa, dependendo de como essas interações forem administradas. Finalmente, a arquitetura *Web* oferece um modelo tecnológico adaptável e requer uma infraestrutura menos cara para proporcionar esses benefícios.

2.1 O conceito *Web 2.0*

O grande impacto da *Web 2.0* é fascinante e perturbador. Não há uma definição simples e direta para a *Web 2.0*. Embora sua origem seja atribuída a Tim O'Reilly, fundador e presidente da O'Reilly Media, que conceitualiza o termo *Web 2.0* como “*mudança para uma internet como plataforma, e um entendimento das regras para obter sucesso nesta nova plataforma. Entre outras, a regra mais importante é desenvolver aplicativos que aproveitem os efeitos de rede para se tornarem melhores quanto mais são usados pelas pessoas, aproveitando a inteligência coletiva*” [O'Reilly, 2005], o termo continua confundindo a maioria das pessoas.

Com o intuito de esclarecer e apresentar os diversos aspectos desta nova forma de

desenvolvimento para a Web o site Web 2.0 BR¹ fez uma coletânea de definições dadas por profissionais envolvidas no cenário da Web no Brasil. Abaixo segue algumas das definições:

“Melhor aproveitamento da inteligência coletiva e do poder de processamento da máquina cliente. Poder às pessoas.”

Marco Gomes

“A Web 2.0 representa a transição para um novo paradigma onde a colaboração ganha força suficiente para concorrer com os meios tradicionais de geração de conteúdo.”

Renato Shirakashi

“Web 2.0 é um buzz word que define conteúdo gerado pelo usuário e com foco no compartilhamento de informações. Tudo regado a AJAX.”

Nando Vieira

“Web 2.0 é como chamamos, depois de uma profunda análise histórica da Web, um conjunto de práticas que ao longo dos anos provaram dar resultado.”

Gilberto Alves Jr

“A Web 2.0 aponta para uma mídia popular, independente de grandes corporações, recriada pelos seus próprios usuários.”

Frederick van Amstel

“Ajax, redes sociais, CGM: as definições mais comuns pra Web 2.0, ou um jeito para se voltar a falar de internet? Para mim nada mudou, tudo evoluiu.”

Michel Lent

“Web 2.0 é buzzword, é fato que a internet está sofrendo transformações, mas precisamos rotulá-la para que essas mudanças tenham validade? Pra maioria da população mundial, que ainda está offline, essa é a Web 1.0.”

Edney Souza

“Sinaliza uma fase na Web onde se pratica a liberdade de falar e ser ouvido. É uma consequência natural do desenvolvimento da internet.”

Vicente Tardin

¹<http://web2.0br.com.br/>

“Web 2.0 usa a Web como plataforma de socialização e interação entre usuários graças ao compartilhamento e criação conjunta de conteúdo”

Guilherme Felitti

Assim, a Web 2.0 tem essencialmente a ver com ambientes propícios à criação e manutenção de redes sociais (abertas ou fechadas, públicas ou privadas). Este espírito estende-se para além das paredes de um determinado site, sendo que cada vez mais se observa o estabelecimento de ligações entre vários sites, com o objetivo de proporcionar funcionalidades adicionais aos membros das respectivas comunidades. A Web 2.0 não possui uma fronteira determinada, mas sim, um centro gravitacional, onde orbitam inúmeros conceitos e recomendações. A figura 2.1 representa este modelo proposto.

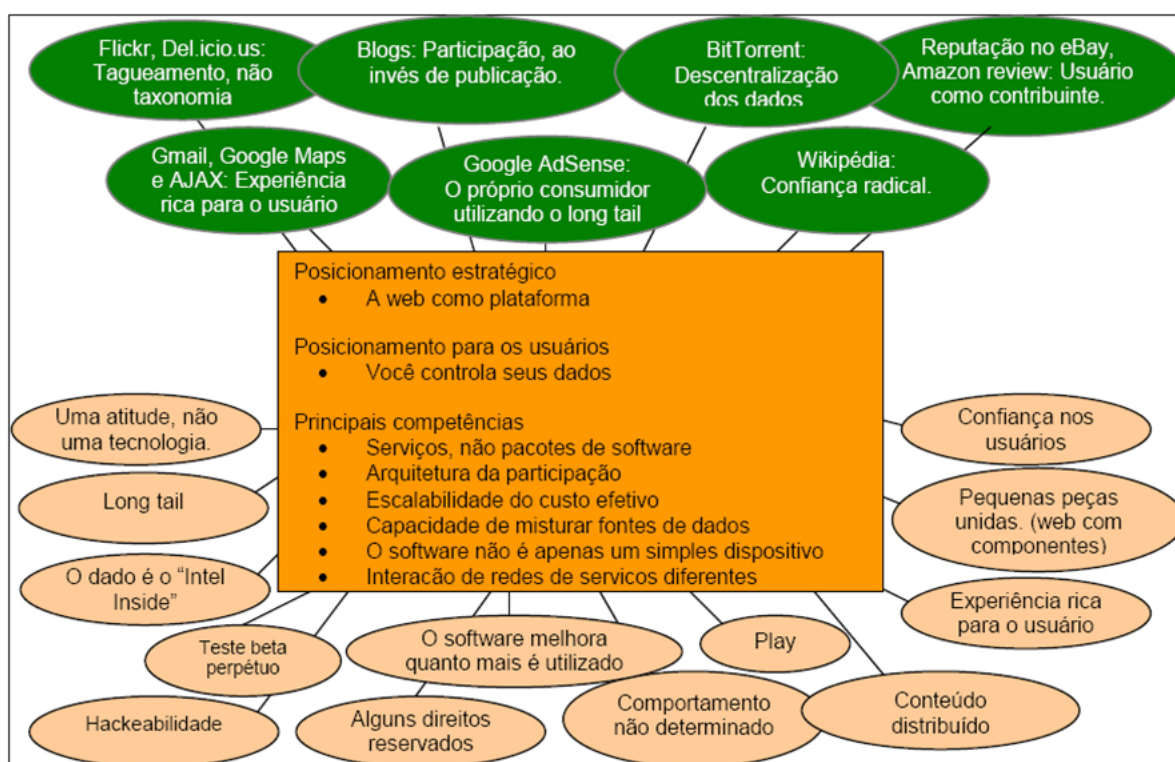


Figura 2.1: Modelo com os conceitos e noções da Web 2.0 [O'Reilly, 2005]

É devido a este objetivo de abertura e transparência que a Web 2.0 se caracteriza também, em grande parte, pelo caráter gratuito (da maioria) dos sites e ferramentas, e também pela criação e disponibilização de API's (*Application Programming Interface* - Interface de Programação de Aplicativos) que permitem a comunicação com outros sites. Estes têm, ultimamente, resultado na criação de múltiplos plugins, desenvolvidos essencialmente pela comunidade de utilizadores, e que permitem estender a funcionalidade básica de um determinado site (ou aplicação).

O fenômeno da Web 2.0 pode ser representado em três pontos-chave básicos que lidam com questões tecnológicas, questões de pessoal e questões corporativas. A visão de cada

um da *Web* 2.0 depende em grande parte do interesse e dos antecedentes de cada pessoa [Smith, 2008].

Os três pontos básicos da *Web* 2.0 são:

- Tecnologia e arquitetura - consiste da infraestrutura da *Web* e o conceito de plataformas de internet. Os exemplos de tecnologias específicas incluem Ajax, *Representational State Transfer* (REST) e *Really Simple Syndication* (RSS). Os tecnólogos tendem a gravitar em direção à essa visão.
- Comunitário e social - enfatiza a dinâmica em torno das redes sociais, comunidades e outros modelos de publicação/partilha de conteúdos pessoais, *wikis* e outros modelos de conteúdos colaborativos. A maioria das pessoas tende a gravitar para essa visão; assim, há um grande foco na "arquitetura da participação" da *Web* 2.0.
- Empresas e processos - modelos corporativos de serviços baseados na internet e em *mashup*². Os exemplos incluem publicidade, modelos de assinatura - *software* como serviço (SaaS) - e uma longa lista de dimensões econômicas, que os homens de negócios tendem a enfatizar muito mais.

Nenhum dos aspectos depende dos outros. Por exemplo: *wikis*, *blogs* e *postcards* não requerem o uso de Ajax ou de publicidade. A publicidade na *Web* não requer REST ou modelos colaborativos.

2.1.1 A Tecnologia - infraestrutura e plataformas de *Web*

A infraestrutura da *Web* 2.0 baseia-se na arquitetura tecnológica e no modelo de desenvolvimento já usado na *Web*, mas representa um novo paradigma para construir, implementar e executar aplicações. A arquitetura voltada para a *Web* é um subconjunto de Arquitetura Voltada a Serviços (SOA), que oferece um modelo globalmente conectado, centrado em redes (e não centrado em dispositivos) e ampliável. Ela se caracteriza por tecnologias como Ajax e o uso de interfaces de programação de aplicações de última geração - serviços de internet, o XML³ e REST⁴ - para expor elementos de um site de internet/aplicação usando uma arquitetura voltada para SaaS e aplicações compostas entregues como *mashups*.

E, nesse aspecto, Tecnologia da Informação(TI) e *Web* estão associadas, e alimentam e amplificam uma à outra. Os fenômenos dirigidos ao consumo agem como impulsionadores fundamentais da *Web*. Os motivos para isso são que uma das principais forças por trás da customização da TI é a internet e a adoção, pelas empresas, de tecnologias *Web*, como

²Site de internet ou aplicação que combina conteúdos de mais de uma fonte. [Sampaio, 2007]

³Extensible Markup Language

⁴Representational State Transfer

serviços de mensagens instantâneas, mecanismos de busca e tecnologias de colaboração desenvolvidas para a indústria de jogos. A integração de telemática, telefones celulares, PDAs, *game boxes* e outras tecnologias é mais ampla do que a internet, e há aspectos da internet que não são dirigidos aos consumidores.

As plataformas da internet emergirão como implementações específicas de cada revendedor. O conceito de “plataforma” é usado em excesso e muitas vezes de forma confusa. Porém, ele oferece um ponto de referência, isto é, um sinal de “você está aqui”, no labirinto da TI.

Como as aplicações entregues aos usuários, sejam eles consumidores ou empresas, exploram melhor os recursos oferecidos pela *Web*, é necessária uma nova maneira de ver o conceito de plataforma para descrever o que está acontecendo. A reivindicação da *Web 2.0* de ser “a internet como plataforma” é melhor descrita como “a *Web* como infraestrutura”. Estão surgindo plataformas que são construídas com base em uma infraestrutura da Internet. Nos referimos a elas como “plataformas da internet”. Algumas são novas, como o Google, enquanto outras são evoluções de tipos anteriores de plataformas, como os sistemas da *Microsoft*. Essas estão emergindo como plataformas amplas, de uso geral. Embora o primeiro efeito seja o uso feito por consumidores, elas serão alavancadas também para o uso corporativo.

Aplicações compostas e plataformas para processos corporativos serão construídas com base em muitas coisas, incluindo aplicações corporativas voltadas para serviços e plataformas da internet.

2.1.2 Aspectos não tecnológicos - Comunitários e Social

Embora seja simples e fácil acrescentar tecnologias específicas, como Ajax e RSS, a produtos, plataformas e aplicações, é mais difícil acrescentar uma dimensão social, como conteúdos desenvolvidos por usuários, ou um novo tipo de modelo corporativo, se esse tiver de ser arquitetado na aplicação.

Acrescentar esses novos aspectos requer repensar o *design* do sistema e possivelmente seu público alvo. Apresenta, assim, maiores desafios do que uma reformulação de *layout* da aplicação usando Ajax, porque isso poderá significar mudanças significativas no desenvolvimento do processo, no modelo corporativo e até mesmo na missão do sistema em construção.

Como essas implicações têm um alcance maior, o Gartner⁵ prevê que a adoção pelas empresas se dará em um nível bem inferior do que a simples incorporação de tecnologia. Porém, ignorar os aspectos não tecnológicos da *Web 2.0* significa que muitas empresas ignorarão também alguns dos benefícios corporativos positivos. Além disso, é importante

⁵Gartner Group é uma empresa de consultoria fundada em 1979 por Gideon Gartner.

não apenas a adoção pelas empresas, mas também é importante o modo como a empresa “se conecta” aos usos corporativos e de consumo direto [Smith, 2008].

2.1.3 Impacto sobre as arquiteturas corporativas

A *Web 2.0* é consistente com uma abordagem que o Gartner denominou computação de classe global. A computação de classe global é uma abordagem para o *design* e a arquitetura de sistemas que amplia os processos computacionais para além da empresa e em direção à cultura do consumidor, do trabalhador móvel, e dos parceiros corporativos. A abordagem de classe global explora as características da computação baseada na internet e emprega aplicações e serviços que são mais flexíveis e inclusivos, mais simples e menos caras do que aquelas projetadas para uso corporativo [Smith, 2008].

Fundamental para a abordagem de classe global é uma ênfase na interoperabilidade por meio de padrões baseados na internet. A arquitetura da *Web* (especialmente porque está evoluindo para a *Web 2.0*) pode ser pensada como a base para a próxima geração de tecnologias de classe global. De forma mais geral, os conceitos da computação de classe global alimentarão a *Web 2.0* e refletirão a cultura dos consumidores.

Em contraste, a computação de classe corporativa costuma ser associada a projetos mais complexos, a uma maior disponibilidade, a sistemas mais escaláveis e mais confiáveis, e equipes maiores de desenvolvedores e gerenciadores de sistemas. Geralmente, essa abordagem envolve um pensamento “ao avesso”, na qual a maioria dos aspectos da computação corporativa (sistemas, aplicações e processos) reflete a cultura das empresa e apóia predominantemente os esforços internos.

2.2 Relações econômicas - Temor e Ambição

Além da tecnologia e da arquitetura, a *Web 2.0* promete afetar outros aspectos das estratégias corporativas. Grande parte da excitação em torno da *Web 1.0* não se referiu à tecnologia, mas muito mais às questões corporativas, com temor e ambição.

Com as amplas implicações econômicas, a atualização de conteúdos, a participação em massa, a inteligência coletiva, os *mashups* e a funcionalidade que podem agora ser acessados em aparelhos pela internet, poderá surgir uma nova concepção criativa corporativa.

O conceito de empresa virtual poderá ressurgir, e o retorno dos temores de que a concorrência baseada na internet reformulará as regras para as indústrias pode não estar distante.

Grande parte do que as pessoas chamam de *Web 2.0* não é inteiramente novo. Muitos dos conceitos e tecnologias já existem há algum tempo. O RSS, por exemplo, é essencialmente o mesmo que o RDF, um formato popularizado pela Netscape durante a era

da *Web* 1.0. O Ajax é essencialmente o JavaScript, o DHTML e o XML, tecnologias que também já existiam. Certamente, colaboração e publicidade não são coisas novas. Os *mashups* possuem uma impressionante similaridade com as “aplicações compostas” derivadas do termo SOA, que é o Software como Serviço. O que é novo é o modo como alguns deles são usados e em quais combinações [Primo, 2007].

A aplicação mais poderosa da *Web* 2.0 pode ser obtida com a exploração de novos modelos corporativos que alavanquem os aspectos tecnológicos e comunitários da *Web* 2.0, porém, essa não é a única maneira de gerar valor. Cada um dos três pontos básicos pode ser analisado individualmente, assim como em várias combinações, para gerar valor. Por exemplo, os modelos comunitários de internet podem ser explorados para os esforços internos de colaboração sem mudar o modelo corporativo básico da empresa. Além disso, as tecnologias *Web* podem ser aplicadas como uma maneira de desenvolver um projeto de SOA flexível e adaptativo, sem mudar o modelo corporativo ou explorar as comunidades da internet. Finalmente, uma empresa poderá explorar o modelo corporativo da PontoCom com maior ênfase na publicidade baseada na intenção e na exploração de programas derivados sem usar as tecnologias voltadas para a *Web* 2.0 e nem explorar as comunidades da internet.

2.3 Princípios Fundamentais da *Web* 2.0

Se na primeira geração de sites, ou *Web* 1.0, os sites eram trabalhados como unidades isoladas, passa-se agora para uma estrutura integrada de funcionalidades e conteúdo [Primo, 2007]. Assim, pode ser ressaltado 5 princípios fundamentais dentro da *Web* 2.0, que são: simplicidade, foco no conteúdo, colaboração, compartilhamento e por fim *Web* como plataforma [Brandt, 2006].

Na *Web* 2.0 o foco principal é no conteúdo, “*content is the kin*⁶”. Seja texto, imagem, áudio, vídeo, etc. Ela só não pode ser estática, deve possuir um conteúdo participativo e democrático, onde há sempre uma realimentação de informação [O’Reilly, 2005].

A colaboração do usuário é também um ponto principal na *Web* 2.0, onde o usuário que produz o conteúdo, deixando de ser apenas consumidor de informação e passando a ser o agente construtor, formando assim uma comunidade, garantindo a credibilidade das informações publicadas, e a disseminação do conteúdo.

Quando se tem foco no conteúdo, e colaboração total dos usuários, automaticamente surge o compartilhamento entre os usuários do sistema, onde há uma grande exposição de conteúdo produzido, opiniões, informações pessoais, habilidades, preferência pessoais, dentre outros.

⁶Conteúdo é a Chave.

A *Web* como plataforma vem selar esses princípios básicos da *Web 2.0*, fazendo da internet uma espécie de rede peer-to-peer (par-a-par), que antes só se usava com algum software instalado no computador. Na *Web 2.0*, as informações são armazenadas na própria *Web*, e não mais no computador pessoal do usuário, como por exemplo, o Gmail⁷, que permite que o usuário acesse seus emails de qualquer lugar, diferente do *Outlook*, que tem estar instalado no computador, e as mensagens ficam disponíveis somente nesse computador específico.

2.4 Quebra de Paradigmas

A *Web 1.0* pode ser vista como o momento em que as pessoas começaram a utilizar e conhecer a Internet. Houve uma grande motivação de várias companhias em apresentarem seus produtos e muitos serviços surgiram.

Em 2001, uma crise afetou a *Web* e a maioria das companhias perdeu destaque e popularidade. A partir deste fato, surgiu a necessidade de mudança no modelo de desenvolvimento. Algumas empresas, que sobreviveram a tal crise, encontraram uma nova forma de criação, um novo modelo de desenvolvimento, nomeado de *Web 2.0*.

O processo de evolução tecnológica levou a *Web* a se adaptar às necessidades que surgiram ao longo do tempo. Levar informação às pessoas não era mais o suficiente para obter sucesso. Foi preciso permitir a participação, envolver as pessoas no processo de criação. Muitas mudanças podem ser percebidas, porém a maior delas é a preocupação em criar serviços preocupando-se com a participação. A era *Web 2.0* trata a rede como um ambiente social. A *Web 1.0* foi criada por cientistas; *Web 2.0* foi criada através da Sociedade [Zambonini, 2006].

Na *Web 2.0* os documentos perderam o foco, o diferencial das empresas está na manipulação e distribuição dos dados. Gerenciar banco de dados é a nova competência central das companhias *Web 2.0* [O'Reilly, 2005]. Na geração anterior, os documentos eram proprietários, com distribuição restrita, quando não proibida. A *Web 2.0* trabalha com dados abertos, com distribuição livre e desejável, permite e estimula a cooperação do usuário com o intuito de melhorar os serviços.

O desenvolvimento e a manutenção de *websites* na *Web 1.0* eram feitos por profissionais, seguindo requisitos determinados pelos clientes. Na maioria dos casos, as páginas eram estáticas e toda alteração demandava tempo e alto custo. Nos poucos *websites* dinâmicos, a introdução de dados, era feita através de sistemas de administração de conteúdo, ou seja, de forma centralizada. Na *Web 2.0*, a construção das páginas é, muitas vezes, feita pelos próprios utilizadores, através de serviços *online*. Eles também participam ativamente da

⁷www.gmail.com

Web 1.0	Web 2.0
Ofoto	Flickr
Mp3.com	Napster, iTunes
Enciclopédia Britânica online	Wikipedia
<i>websites</i> pessoais	Blogs
especulação de domínios	otimização de ferramentas de busca
Visita a páginas	Custo por click
<i>screen scraping</i>	web services
Publicação	Participação
sistema de gerenciamento de conteúdo	wikis
diretórios (taxonomia)	tags ("folksonomy")

Tabela 2.1: Comparação entre serviços Web 1.0 e Web 2.0

manutenção destes serviços.

A tabela 2.1 apresenta alguns aplicativos e princípios da *Web 2.0* comparados aos da *Web 1.0*. Os serviços têm a participação, cooperação e descentralização dos dados como filosofia.

2.5 Características

Pelo modelo proposto por O'REILLY (figura 2.1) são citadas ferramentas, que serão detalhadas na sessão 3.4, e indicações que caracterizam a *Web 2.0*. Entre as indicações podem ser destacadas: participação, cooperação, usabilidade, descentralização dos dados, confiança no conteúdo dos usuários, experiência rica para o usuário, personalização e atualizações frequentes.

2.5.1 Participação

Desde o surgimento da *Web*, os *websites* eram construídos com o intuito de informar. Os usuários se comportavam como consumidores do conteúdo que trafegava na rede. Uma das maiores revoluções causadas pela *Web 2.0* foi a efetiva participação dos usuários na criação do conteúdo. A *Web* ganhou caráter social, uma rede formada de pessoas participativas e influentes no processo de geração, distribuição e organização da informação. O conceito predominante para o sucesso das empresas na era *Web 2.0* é a Inteligência Coletiva. A coletividade está sendo explorada para enriquecer o conteúdo da própria rede.

O conceito predominante para o sucesso das empresas na era *Web 2.0* é a Inteligência Coletiva. A coletividade está sendo explorada para enriquecer o conteúdo da própria rede.

Explorar a Inteligência Coletiva significa usar da maior fonte de informação, o usuário, para gerar o conteúdo. Deixar as pessoas interagirem, permitir que possam disseminar seus conhecimentos.

Com o passar do tempo, surgiram várias ferramentas que permitem aos utilizadores comuns publicarem informações sem grandes conhecimentos técnicos. O crescimento dos *wikis* e *blogs* é um bom exemplo do enorme crescimento da tendência de participação.

Outro aspecto importante é o conceito de comunidade. Ele já existia desde o início da *Web* nos fóruns e nas conhecidas "salas ou listas de discussões". Porém, nos últimos anos, o crescimento foi considerável e caracterizou uma verdadeira revolução social na *Web*. Através de aplicações simples e intuitivas os membros das comunidades são atraídos a participar e trocar experiências sobre temas diversos e com pessoas diferentes. Os sites sociais, como o *Orkut*⁸ e *Octopop*⁹, são exemplos do conceito de comunidades.

2.5.2 Cooperação

"Aplicativos *Web 2.0* são construídos a partir de uma rede cooperativa de serviços de dados." [O'Reilly, 2005]. O controle de dados na *Web 2.0* não é uma prática desejável. Em um momento de notória colaboração, a preocupação está na reutilização e desenvolvimento coletivo dos aplicativos. O conceito de *software* livre e código aberto se estendem pela *Web*.

Ferramentas como os *microformats*¹⁰, *mashups* e APIs permitem que soluções já desenvolvidas sejam facilmente utilizadas e personalizadas para problemas específicos. O *GoogleMaps*¹¹ lidera o *ranking* dos *mashups* mais utilizados. Muitos aplicativos que desejam mostrar imagens geográficas, marcar coordenadas, calcular distâncias, e outras funcionalidades, encontram neste serviço uma maneira simples e eficiente de atender suas necessidades. Além disso, elas podem incrementar este serviço e desenvolver aplicações inovadoras.

2.5.3 Usabilidade

A usabilidade permite que o usuário tenha acesso à informação, executando com sucesso as tarefas desejadas [Nielsen, 2000]. Os aplicativos *Web 2.0* têm a missão de

⁸<http://www.orkut.com>

⁹<http://br.octopop.com/>

¹⁰A idéia de *microformats* é usar padrões para solucionar problemas comuns. Serviços como o geo ou hCalendar podem ser usados quando necessita-se apontar coordenadas geográficas ou escrever datas, respectivamente. [Ribeiro, 2008]

¹¹Disponível em <http://maps.google.com/>

Serviços	Descentralização
Software livre	Indústria de software
MP3/i-música	Indústria fonográfica
Ipod/podcasting	Indústria de rádio
VoIP	Indústria de telefonia
RedeMesh	Indústria de telecomunicações
Laptop a baixo custo	Indústria de hardware
Wikipedia	Indústria de conteúdo
Portais colaborativos, blogs e redes sociais	Indústria da informação

Tabela 2.2: Descentralização dos dados em diversos setores

facilitar a participação do usuário. Os serviços estão sendo acessados por pessoas com conhecimentos diversos e devem atender de forma eficiente a todos. Os sistemas intuitivos e que facilitam a navegação estão, finalmente, ganhando atenção. Os profissionais vêm percebendo que é preciso projetar antes de criar para que os serviços não fiquem obsoletos.

O teste de usabilidade ganhou uma peculiaridade no contexto da *Web 2.0* [Nielsen, 2000]. Um monitoramento constante é feito considerando as ações do usuário e a partir disso são feitas alterações e melhorias nos sistemas. Os fatores mais influentes são a grande massa de usuários e a sua participação efetiva. As respostas aos testes tornam-se mais eficientes, pois o retorno às questões é maior e espontâneo.

2.5.4 Descentralização de Dados

Fontes únicas de funcionamento estão em declínio na *Web 2.0*. “O PC não é mais o único dispositivo de acesso aos aplicativos de Internet e os aplicativos limitados a um único dispositivo têm menos valor do que os que são conectados.” [O’Reilly, 2005]. A utilização e a conexão de vários dispositivos possuem uma forte tendência de crescimento e são suportadas por muitas tecnologias inovadoras. Existe um grande entusiasmo dos estudiosos em relação a este fator. Eles acreditam que este seja o futuro das aplicações *Web*.

A descentralização, entretanto não se restringe a serviços sendo executados em vários dispositivos. Trata-se da descentralização no domínio dos dados e no controle sobre eles. A tabela 2.2 apresenta alguns serviços que caracterizam a descentralização em diversos setores.

Algumas companhias, pioneiras no movimento P2P, tiveram uma posição radical no processo de descentralização da Internet. O cliente também é um servidor e os dados, ou

fragmentos deles, podem ser fornecidos de qualquer lugar e de maneira transparente ao usuário.

2.5.5 Confiança no conteúdo dos usuários

A participação do usuário na criação do conteúdo afetou em outro fator: a confiança. O controle da informação não está mais centralizado em uma equipe de desenvolvimento, mas sim disperso entre os inúmeros colaboradores da rede.

O monitoramento das informações acompanha a ética do *software* livre. Eric Raymond¹² diz que “com um número suficiente de olhos, todos os *bugs* tornam-se visíveis”. Existe um comprometimento na relação entre aplicativo e usuário, se uma informação estiver incorreta ou a tecnologia falhar, as pessoas serão notificadas através de avaliações, comentários, enfim pela própria comunicação facilitada.

Na *Wikipedia*, por exemplo, o usuário pode contribuir com postagens sobre quaisquer temas, bem como editar o conteúdo inserido por alguém. Esse poder é controlado pelos próprios visitantes do site, são eles que avaliam, corrigem e denunciam uma informação equivocada. A *Amazon*¹³ encontrou um diferencial ao permitir que os usuários avaliassem, sugerissem e criticassem os produtos a venda. Com esta filosofia as informações equivocadas são rapidamente denunciadas e corrigidas.

2.5.6 Experiência Rica para o Usuário

A plataforma *Web* foi idealizada sem muitas considerações quanto a performance avançada na execução de aplicativos. Praticamente todas as aplicações para a *Web* carregam novamente as páginas inteiras, a cada requisição feita, o que dificulta a experiência do usuário e reduz o potencial das aplicações. Isso está mudando com o uso de algumas tecnologias e com novas formas de programação. As páginas estão se tornando mais leves e agradáveis.

Os aplicativos *Web* estão adquirindo um formato muito próximo ao do *desktop*. As páginas não são mais recarregadas a cada clique, as informações são processadas de forma assíncrona. Os serviços estão unindo a rapidez e facilidade de uso do *desktop* ao grande alcance e compartilhamento de dados permitidos na *Web*. A utilização do AJAX e de aplicações RIA¹⁴ possibilitou uma grande evolução no desempenho dos sistemas. São tecnologias distintas com o mesmo objetivo: melhorar a experiência do usuário.

RIA são aplicações que contêm características e funcionalidades de uma aplicação

¹²Eric Steven Raymond é um famoso hacker americano, um ícone no movimento do Open Source e do software livre

¹³Disponível em <http://www.amazon.com/>

¹⁴Abreviação de *Rich Internet Applications* ou Aplicações Ricas para Internet [Crane et al., 2007]

desktop tradicional. Tipicamente uma aplicação RIA transfere a necessidade de processamento do cliente, numa arquitetura cliente-servidor, para o navegador mas, mantém o processamento mais pesado no servidor de aplicação.

2.5.7 Personalização

Com o aumento do número de fontes de informação, o usuário ganhou o poder de escolher o conteúdo que deseja ter acesso e a forma com que ele será apresentado.

A visualização do conteúdo é feita de acordo com o que o usuário deseja. Em sites como *PageFlakes*¹⁵ (figura 2.2) o usuário tem o poder de escolher entre diversos temas, como esportes, notícias, fotos para que sejam exibidos e ainda optar pela forma com que serão organizados na página. O resultado é uma mistura de conteúdo acessível de forma simples e rápida.

O *browser* também não é mais a única forma de acesso a informações. O usuário pode receber as notícias ou a previsão do tempo, por exemplo, no desktop através de aplicativos especializados, como o *FeedDemon*¹⁶. No capítulo 3 serão abordadas as tecnologias como RSS, XML e RDF que possibilitam a personalização e distribuição do conteúdo.



Figura 2.2: Exemplo de uma página personalizada no PageFlakes

2.5.8 Atualizações frequentes

“Quando dispositivos e programas estão conectados à internet, os aplicativos não são mais artefatos de *software*, são serviços em andamento” [O’Reilly, 2005].

Os *softwares* são lançados no mercado como um produto em aberto que pode ser incrementado em curto período de tempo. A filosofia segue o lema do código aberto,

¹⁵Disponível em <http://www.pageflakes.com/>

¹⁶Programa com uma interface semelhante ao *Outlook* que lê notícias em formato RSS, além de comunicar-se com blogs para adicionar notas.

“lançar logo, lançar sempre”. O produto não deve ser liberado aos usuários com *bugs* ou instabilidades, este não é o objetivo. As funções que estiverem disponíveis devem funcionar perfeitamente. Porém, novos recursos serão lançados constantemente e junto com os existentes ficam em contínuo monitoramento para análise do comportamento do usuário. Não existe mais um prazo para o lançamento de um *software* ou de novas funcionalidades. Isso é feito continuamente como reação às respostas obtidas dos usuários.

2.6 A Web como plataforma

O que é uma plataforma? Talvez não seja o melhor, mas o mais popular exemplo é o *Windows*. De forma bem simples e superficial, a plataforma é o lugar onde os programas existem. Ela dá os elementos que os programas precisam para existir. Quem diz o que deve ser impresso é o programa, quem manda a informação para a impressora é o *Windows*, a plataforma. A internet é assim: um lugar, um ambiente, a plataforma [Jr, 2005].

Na figura 2.3, tem-se a representação da plataforma *Web 2.0*. Ela reúne todas as tecnologias e serviços, juntos com a filosofia e modelos apresentados pela *Web 2.0*. Assim, com as filosofias da *Web 2.0* descritas no capítulo anterior percebe-se uma série de mudanças na maneira de projetar. O comportamento dos usuários, a elaboração do conteúdo, o *design* das páginas, a maneira de anunciar e comercializar o produto foram elementos que sofreram o impacto da *Web 2.0*.

Neste capítulo serão abordadas as mudanças causadas nos diversos elementos da *Web* e as considerações que estão sendo analisadas pelos profissionais.



Figura 2.3: A plataforma *WEB* [Zambonini, 2006]

2.6.1 O Usuário

O usuário é o principal objetivo a ser alcançando na *Web 2.0*. Todos os serviços são desenvolvidos com a pretensão de envolvê-los, fazê-los participar e contribuir no desen-

volvimento de todo o conteúdo da *Web*. O usuário, por sua vez, mudou a postura que tinha diante dos serviços. Eles perderam o medo de participar, contribuir e opinar.

Segundo [O'Reilly, 2005], a grande parte dos usuários não irá adicionar valor a um aplicativo sem que tenham algum benefício. Eles serão atraídos a contribuir quando perceberem que o serviço pode ser útil aos seus interesses. Um bom exemplo deste comportamento é o FLICKR¹⁷. A companhia conseguiu uma forte contribuição dos usuários oferecendo a oportunidade de criar álbuns pessoais. Eles não têm o intuito de fornecer imagens ao serviço, isso acontece como consequência do uso, como na figura 2.4 onde mostra cerca de 7871 *uploads* de imagens somente no último minuto referente ao momento em que a página foi acessada.

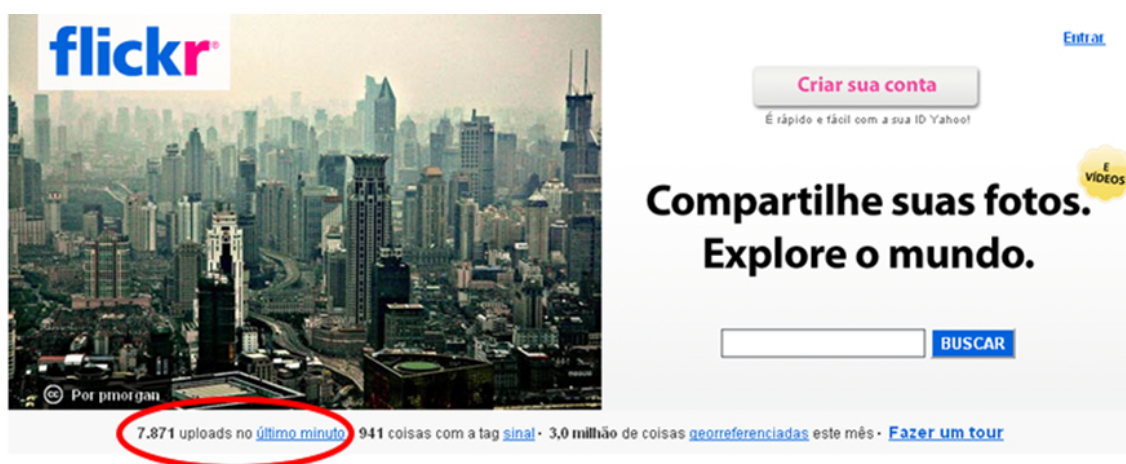


Figura 2.4: Editor e compartilhador de imagens online

Os usuários estão contribuindo também economicamente no processo de desenvolvimento. Com a participação efetiva, eles mantêm os sites e os sistemas sempre atualizados, sem que para isso seja preciso uma equipe especializada e um custo elevado.

Muitos *websites* só sobrevivem devido à participação do usuário, como no caso dos *wikis*, que têm seus conteúdos atualizados pelos próprios usuários, como veremos na sessão 4.4.

2.6.2 O Conteúdo

Neste momento, de grande valorização da inteligência coletiva, as características ao redor do conteúdo sofrem mudanças significativas. A preocupação não está mais em publicar o conteúdo e sim em criar alternativas que ajudem a organizá-lo, selecioná-lo e distribuí-lo.

¹⁷<http://www.flickr.com/>

2.6.3 Organização

A organização do conteúdo está sendo feita de maneira livre, sem que obrigatoriamente sejam seguidas hierarquias impostas. Esta organização é chamada de *folksonomia*, ou seja, uma forma de marcação utilizando palavras chaves, chamadas *tags*. O usuário associa livremente uma palavra para identificar a informação. Desta forma, múltiplas combinações de *tags* podem ser usadas para encontrar um objeto. Esta é a principal filosofia dos sites *Del.icio.us*¹⁸, *Flickr* e *Youtube*¹⁹. Neles os usuários podem guardar os links favoritos, as imagens ou vídeos, respectivamente, associando a eles qualquer palavra.

A busca é muito favorecida nos casos acima, pois apresenta diversas combinações criadas pelos usuários, que dificilmente seria gerada com uma estrutura pré-determinada.

2.6.4 Avaliação

O cuidado com a avaliação do conteúdo também cresceu muito. Como já citado anteriormente, o grande número de usuários participando é um auxílio na avaliação do conteúdo.

Porém, as companhias perceberam que é preciso criar artifícios simples e eficazes para que os usuários denunciem abusos e erros. A maioria dos aplicativos *Web 2.0* coloca em destaque essas opções e alguns ainda recompensam, com pontos positivos ou status diferenciados para aqueles que colaboram.

2.6.5 Distribuição

Em decorrência do grande crescimento de dados na *Web*, os leitores se viram com a necessidade de selecionar aquilo que lhes interessa. A visualização do conteúdo também ganhou um diferencial. Ela ocorre da forma que o usuário deseja. Ele escolhe o que quer ver e a forma com que as informações serão apresentadas. Considerando essa característica, muitos *websites* permitem a personalização, inclusive estimulam a distribuição do conteúdo livremente e ainda se preocupam em projetar seu bom uso em diferentes dispositivos.

Segundo [O'Reilly, 2005], “A proteção à propriedade intelectual limita a reutilização e impede a experimentação”. Por isso, não é aconselhável restringir a apropriação coletiva. As companhias estão se preocupando mais em divulgar o conteúdo, mantendo uma identidade visual do que controlá-lo. Esta é a postura do *Youtube*. Um vídeo pode aparecer em qualquer site que será identificado como postados no *Youtube*, como mostra a marcação vermelha na figura 2.5.

¹⁸Disponível em <http://del.icio.us>

¹⁹Disponível em <http://www.youtube.com>



Figura 2.5: Identidade Visual do Youtube

Um conceito importante em relação à distribuição dos dados na *Web* é a *Long Tail*. O termo, traduzido como cauda longa, foi criado por Chris Anderson²⁰ para designar o poder coletivo de pequenos sites, que constituem a maior parte do conteúdo da rede. A figura 2.6 apresenta uma análise da distribuição do conteúdo. Percebe-se que as grandes corporações ocupam o topo, mas um pequeno espaço, enquanto os nichos são pequenos e inúmeros, formando essa "cauda longa" [Sena, 2007].

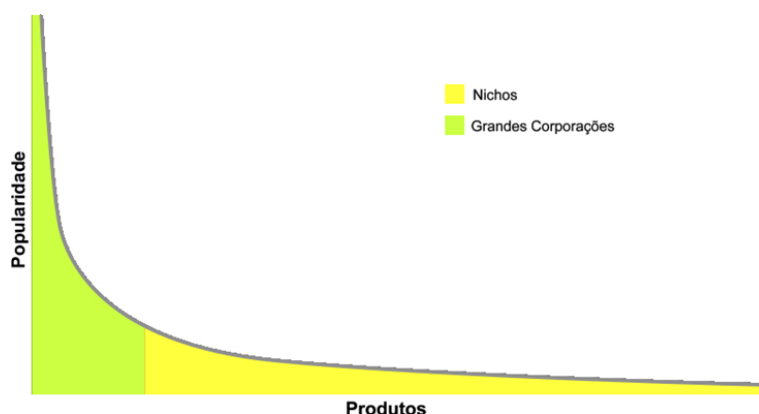


Figura 2.6: A cauda longa [Sena, 2007]

As grandes companhias estão ganhando destaque à medida que aprendem a incentivar a tendência de agregar e desenvolver serviços para os clientes menores. Aumenta também a preocupação com algoritmos de gerenciamento de dados que visam atingir a rede em toda a sua extensão. A regra básica da Cauda Longa é disponibilizar o máximo possível, de todas as formas, e ajudar as pessoas a encontrar seu produto ou serviço [Sena, 2007].

²⁰Editor-chefe da revista americana Wired e autor do livro "A Cauda Longa" (Campus/Elsevier), em 2006 [Sena, 2007]

2.6.6 O *Design*

O desenvolvimento de *layouts* propõe a facilidade e simplicidade do uso dos *websites* e sistemas *Web*.

A pretensão do *designer* é estimular o usuário a participar da *Web* e permitir a ele autonomia, facilidade e essencialmente simplicidade na sua busca por informação e na colaboração no desenvolvimento desta.

Um site na Web 2.0 não pode ser lento, nem difícil de se usar. Por causa disso foram criadas novas tecnologias (como o AJAX, por exemplo), e novas formas de interatividade do usuário com os sites. Sites com AJAX promovem uma experiência muito mais rica para o usuário. Este parece ser um caminho sem volta. A tendência é que todos os sites da internet se transformem em sites com AJAX ou alguma tecnologia parecida [Jr, 2005];

Algumas tendências podem ser percebidas no desenvolvimento de *layouts* e todas elas visam tornar a visita aos *websites* uma atividade agradável e satisfatória.

Layout Simples

A simplicidade facilita ao usuário encontrar o que deseja. O projeto da página deve se preocupar em não perder o foco do usuário nas informações. O *Layout* não pode deixar a atenção do usuário ser atraída por animações ou por elementos gráficos mal distribuídos na página.

Há algum tempo, o uso de três ou mais colunas era muito comum para organizar o conteúdo das páginas. Atualmente, esta organização usa, na maioria dos projetos, uma ou duas colunas. Reduzir os elementos na página e o volume de informações são as grandes vantagens desta simplificação [Tosing, 2007].

Projete o conteúdo e não a página

Esta característica reflete a preocupação de chamar a atenção do usuário para o conteúdo da página. O uso de cores fortes ou elementos em 3D são aceitos neste contexto se usados para dar destaque ao conteúdo e nunca para desviar o foco.

Uso de espaço em branco

O uso de espaço em branco foi muito criticado há alguns anos atrás, por que entendiam que era um desperdício de espaço na página. Mas, hoje, o projeto se preocupa em organizar bem o conteúdo e para isso os espaço em branco são um excelente recurso. Eles possibilitam uma separação bem definida dos tópicos que facilita muito a navegação [Tosing, 2007].

Textos grandes

Esta tendência deve ser bem analisada, pois o uso de fontes grandes não é bem aceito em todo o contexto da página. Ele deve ser aplicado para o conteúdo mais importante, que merece maior destaque. O texto mais importante deve ser maior que o texto normal. [TOSING, 2007]

O uso de textos grandes também facilita o acesso de uma parcela dos visitantes que possuem alguma deficiência visual. Torna a informação principal fácil de absorver.

Cores fortes e contrastantes

As cores fortes ganham destaque quando unidas à neutralidade do fundo da página. Elas atraem o olhar do usuário e ajudam a dividir as seções claramente, contribuindo para a usabilidade das páginas.

Outras padronizações, como o uso de imagens refletidas e gradientes, estão sendo muito difundidas entre os profissionais de *design*. Porém, neste trabalho, foram abordadas as tendências que influenciam na acessibilidade e usabilidade.

Quando ocorre uma tentativa de padronização do *design*, deve-se ter uma cautela muito grande, pois um dos princípios da *Web 2.0*, a personalização, pode ser afetado.

Utilizando-se de tantos padrões pode ocorrer uma redução na criatividade dos profissionais e os sites ficarem todos com a mesma característica visual.

2.6.7 Publicidade e *Marketing*

O modo das empresas se comunicarem com seu público alvo também sofreu uma mudança de impacto com a *Web 2.0*. A propaganda não é mais um processo de mão única, onde a empresa apresenta o produto e o consumidor se limita a recebê-la. A participação dele é ativa neste processo. Por exemplo, o usuário auxilia na publicidade de um livro ou CD quando posta um comentário positivo sobre o produto em uma loja virtual. Os serviços devem ser elaborados para permitir ao consumidor a participação direta na propaganda.

Os grandes aplicativos de sucesso da *Web 2.0* não necessitaram de grandes investimentos em *marketing*. Eles exploraram o chamado *marketing* viral. O *marketing* viral e a publicidade viral referem-se a técnicas de *marketing* que tentam explorar redes sociais pré-existentes para produzir aumentos exponenciais em conhecimento de marca, com processos similares a extensão de uma epidemia, onde a empresa divulga seu produto a algumas pessoas e elas se encarregam de propagar a informação aos amigos, que por sua vez informam aos seus amigos e assim cria-se uma ampla rede de divulgação [Maia, 2008].

Um comportamento que vem gerando bons resultados em relação à propaganda é a

utilização de links patrocinados, como mostra na área marcada na figura 2.7. A divulgação dos produtos e serviços é feita seguindo o interesse do usuário. O *Google* é um excelente exemplo.

Os links são mostrados de forma discreta, sem poluição visual, de acordo com a busca que o usuário faz. Desta forma, o usuário é facilmente atraído, pois se trata de um assunto do seu interesse e ainda evita o inconveniente de desviar a sua atenção.

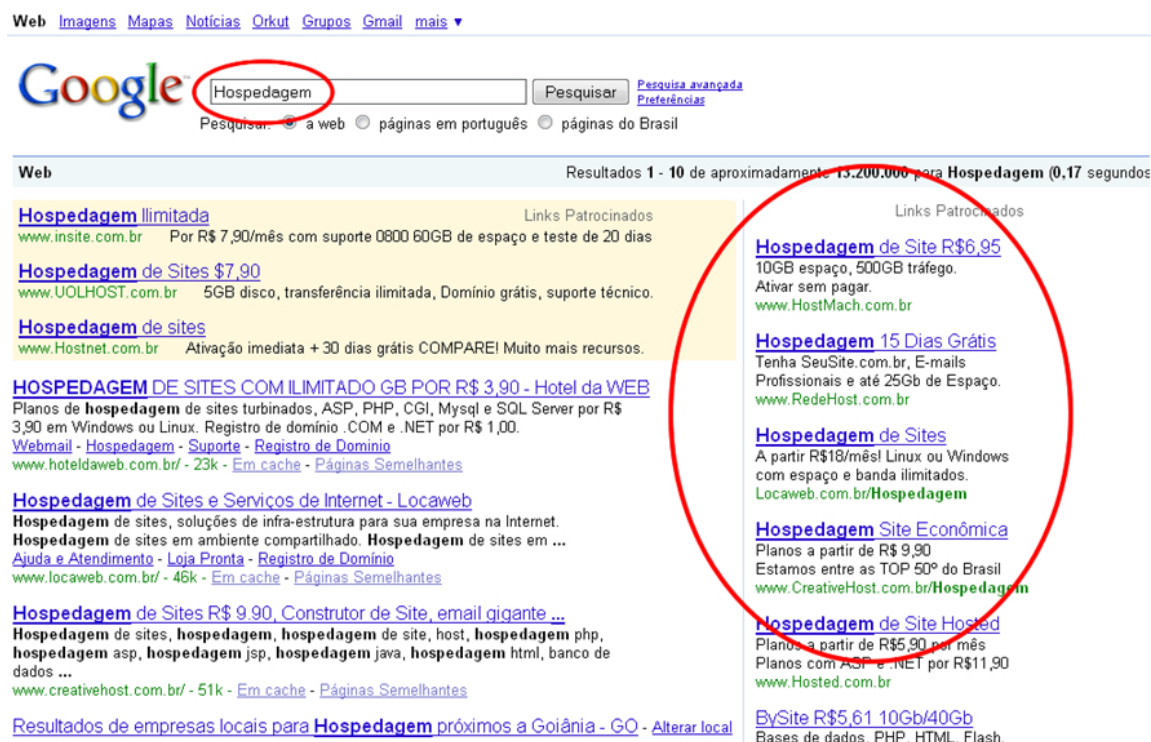


Figura 2.7: Links Patrocinados - Google

2.6.8 R.I.A - *Rich Internet Applications* ou Aplicações de Internet Rica

Uma consequência da *Web* como plataforma é que há menos ênfase no *software* (como um pacote: licenciado e distribuído), e muito mais sobre um pedido de um sistema, sendo que muitos dos sistemas *Web 2.0* permanecem em uma espécie de “beta perpétuo”, ou seja, sempre passando por avaliação de usuário e realimentação de informação.

Assim, cria-se uma melhor experiência de usuário, traduzida pelas Aplicações de Internet Rica (R.I.A.), que são aplicações *Web* que tem características e funcionalidades de *softwares* tradicionais do tipo aplicativo, incorporando a usabilidade das aplicações em *desktop* para *Web* [Fontes, 2006].

A *Web* foi originalmente projetada para navegação por documentos HTML, no modelo de “clique, aguarde e recarregue”, ou seja, a página é transferida do servidor para o cliente

e recarregada a cada evento, envio de dados ou navegação. Essa modelagem é dita modelo de comunicação síncrona.

Na figura 2.8 temos a representação dessa comunicação síncrona, onde a transmissão ocorre de acordo com a atividade do usuário, assim o servidor somente faz processamentos a partir do clique por parte do usuário.

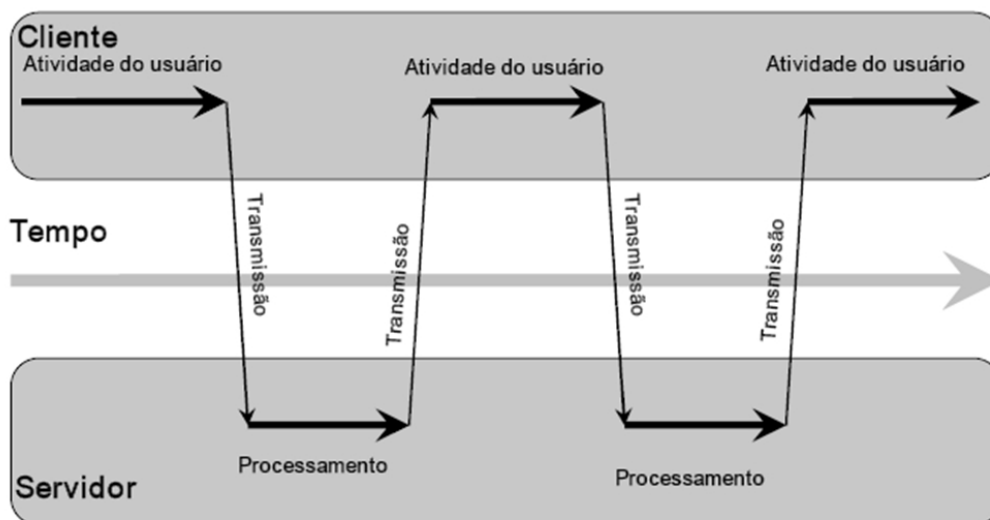


Figura 2.8: Modelo Síncrono [Crane et al., 2007]

Aplicações de Internet Rica (R.I.A) são aplicações *Web* que tem características e funcionalidades de *softwares* tradicionais do tipo aplicativo.

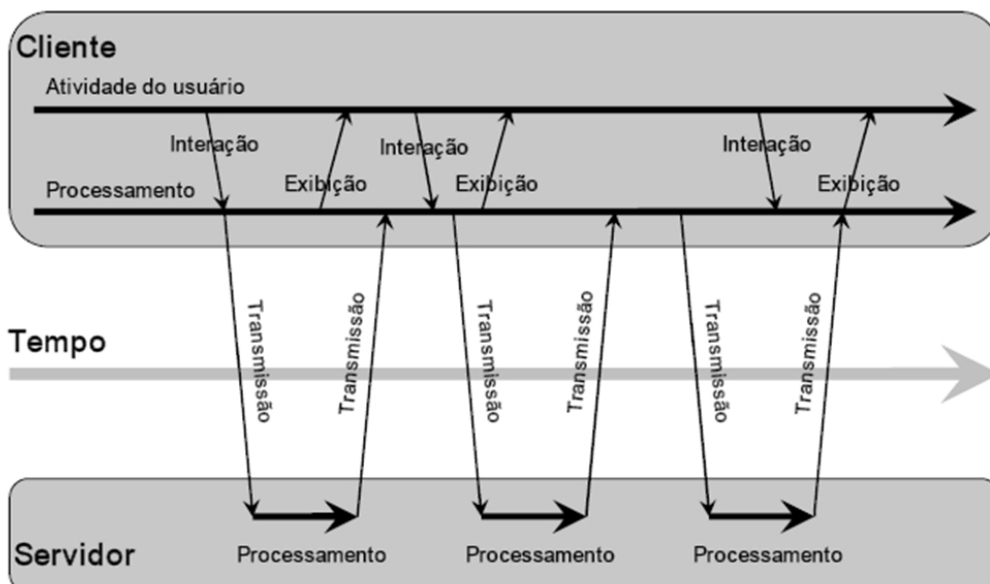


Figura 2.9: Modelo Assíncrono [Crane et al., 2007]

R.I.A típicos transferem todo o processamento da interface para o navegador da internet, porém mantém a maior parte dos dados (como por exemplo, o estado do programa,

dados do banco) no servidor de aplicação. O AJAX (*Asynchronous Javascript And XML* - JavaScript e XML Assíncronos) é a técnica com a habilidade tecnológica para rodar códigos no lado do cliente, porém existem algumas outras alternativas baseadas na tecnologia Flash.

No R.I.A o modelo de comunicação é assíncrono, desta forma, uma ação na interface realizada pelo usuário como o clique em um botão ou no link não necessita esperar por uma resposta do servidor, fazendo com que o usuário espere pelo processamento.

Talvez o mais comum é que estas aplicações, a partir de uma solicitação, antecipe uma futura necessidade de alguns dados, e estes são carregados no cliente antes de o usuário solicite-os, de modo a acelerar uma posterior resposta, como representado na figura 2.9, onde existe uma interação do lado do cliente, prevendo suas ações e já trazendo as informações do servidor para o lado do cliente.

O R.I.A permite também que o fluxo de dados na rede também possa ser significativamente reduzido, levando os usuários a ter a sensação de estarem utilizando uma aplicação *desktop*.

Capítulo 3

Tecnologias e Serviços

Como descrito no capítulo anterior, a *Web* como plataforma, envolve uma série de tecnologias, serviços e filosofias. Neste capítulo serão estudados os serviços e tecnologias envolvidos na plataforma *Web*.

3.1 *RSS - Real Simple Syndication* ou Acesso Fácil as Notícias

Hoje em dia é comum encontrar sites que possuem pequenas figuras (geralmente na cor laranja) acompanhadas de dizeres como “RSS”, “XML” ou “Atom”, como mostram a figura 3.1. Esta figura, representa o logo das siglas RSS, ATOM, e XML, e também o símbolo universal desse padrão



Figura 3.1: Símbolos de um RSS [Sampaio, 2007]

Graças a esse recurso, sites e blogs podem divulgar conteúdo novo de maneira rápida e precisa. Quanto ao usuário, este pode ser informado de cada novidade de seus sites preferidos.

A sigla RSS tem mais de um significado. Alguns a chamam de RDF *Site Summary*, outros a denominam RSS (*Really Simple Syndication*). Há ainda os que a definem como *Rich Site Summary*. [Sampaio, 2007]

RSS é um recurso desenvolvido em XML que permite aos responsáveis por sites e *blogs* divulgarem notícias ou novidades destes. Para isso, o link e o resumo daquela notícia (ou

a notícia na íntegra) é armazenado em um arquivo de extensão .xml, .rss ou .rdf. Esse arquivo é conhecido como feed, feed RSS.

O interessado em obter as notícias ou as novidades deve incluir o link do *feed* do site que deseja acompanhar em um programa leitor de RSS. Esse *software*, ou serviço, se for um site, tem a função de ler o conteúdo dos *feeds* que indexa e mostrá-lo em sua interface.

Uma vez que os feeds estejam disponíveis, os computadores fazem uma busca regular e programada por eles a fim de encontrar atualizações nas publicações dos *websites*. Esta busca é feita através de um programa leitor de RSS, chamado frequentemente de agregador, ou seja, um programa que interpreta o código XML do RSS. Eles têm a missão de fornecer ao usuário uma relação atualizada de todo o conteúdo que ele escolheu visualizar. Desta forma, o leitor não precisa visitar os *websites* e *blogs* para receber as novidades. [Sampaio, 2007]

A RSS significa que o *browser* não é o único meio de se ver uma página [O'Reilly, 2005]. Existem agregadores como o *Bloglines*¹ que é totalmente baseado na rede, aplicativos *desktops* e ainda os que permitem aos usuários acessar através de dispositivos portáteis. Grandes portais de notícias, como o jornal O Globo² possui uma seção chamada Central RSS de onde pode-se baixar vários tipos de *Web feeds*.

A figura 3.2, mostra o código usado para divulgar que a página *Web* disponibiliza um conteúdo RSS.

```
1 <a href="xml/rss.xml">
2   
3   Esta página fornece conteúdo RSS
4 </a>
```

Figura 3.2: Código de indicação do uso de conteúdo RSS

Alguns sistemas operacionais já permitem adicionar feeds em uma barra lateral na área de trabalho, como no caso do *Windows Vista*, e do *Windows Seven*. Existem também *softwares* leitores de RSS, como por exemplo, o *FeedReader*, que permite que você adicione os feeds de vários sites, permitindo assim, a visualização de conteúdos de seu interesse em um local só, sem que seja necessário abrir as páginas onde se encontra cada conteúdo.

Existem versões diferentes de RSS. As mais usadas são RSS 1.0 e RSS 2.0. Cada uma tem vantagens e desvantagens. A RSS 2.0 é conhecida pela simplicidade, enquanto que RSS 1.0 possui especificações mais minuciosas.

As versões RSS 1.0 e RSS 2.0 são classificadas como informais, ou seja, elas não são publicadas por um órgão normatizador, mas sim por um pequeno grupo de pessoas. Com o objetivo de padronizar e tornar estáveis as regras para a publicação de conteúdo, foi

¹Disponível em <http://www.bloglines.com>

²Disponível em <http://oglobo.globo.com/rss/>



Figura 3.3: Código de indicação do uso de conteúdo RSS

RSS 1.0	RSS 2.0
<i>RDF Site Summary</i>	<i>Really Simple Syndication</i>
Desenvolvida por um grupo de desenvolvedores através da continuação da versão RSS 0.9	Foi originariamente documentada pela Netscape e aperfeiçoada pela Userland. Baseada na versão RSS 0.91
Utiliza <i>namespaces</i> XML de acordo com o padrão W3C	Utiliza <i>namespaces</i> XML de acordo com o padrão W3C
Detalhamento da especificação	Simplicidade
Útil para ser integrado a sistemas de <i>Web Semântica</i>	Fácil para codificação manual

Tabela 3.1: Comparação entre RSS1.0 e RSS [Ruby, 2009]

criado, pela *Internet Engineering Task Force*³ (IETF), o Atom. Apesar do nome diferente, as funções são as mesmas do RSS. A tabela 3.1 faz uma comparação entre as versões. [Ruby, 2009]

A figura 3.4 mostra um exemplo de código do RSS 1.0

Na primeira linha tem-se a declaração XML, informando a versão utilizada. O elemento `rdf:RDF` mostra que o conteúdo segue este padrão e `xmlns:rdf` indica o namespace do RDF, seguido da url que aponta a localização do arquivo contendo a sintaxe oficial de RDF.

³Comunidade internacional geradora de documentos e normas para a Internet


```

1  <?xml version="1.0"?>
2  <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3  xmlns="http://purl.org/rss/1.0/"
4  xmlns:dc="http://purl.org/dc/elements/1.1/">
5  <channel rdf:about="http://exemplo.com/news.rss">
6  <title>EXEMPLO</title>
7  <link>http://exemplo.com/ </link>
8  <description>Exemplo de RSS 1.0</description>
9  <items>
10 <rdf:Seq>
11 <rdf:li resource="http://exemplo.com/2002/09/01/" />
12 <rdf:li resource="http://exemplo.com/2002/09/02/" />
13 </rdf:Seq>
14 </items>
15 </channel>
16 <item rdf:about="http://exemplo.com/2002/09/01/">
17 <title></title>
18
19 <link></link>
20 <description> </description>
21 <dc:date></dc:date>
22 </item>
23 </rdf:RDF>

```

Figura 3.4: Código RSS 1.0 [Nottingham, 2005]

3.1.1 *Resource Description Framework* ou Moldura de Descrição de Recursos (RDF)

É uma estrutura declarativa baseada em uma marcação *namespace* provida pela XML para estabelecer relações entre itens da *Web* [Condy, 2006].

A RDF foi criada para representar claramente informações de recursos da *Web* e principalmente de forma que possa ser processado por máquinas. A RDF não é um conceito usado apenas no contexto de RSS, como muitos pensam. Entre as inúmeras possibilidades de uso, a RDF pode ser usada para descrever propriedades para itens de compra, cronogramas de eventos, bibliotecas eletrônicas e informações sobre páginas da *Web*, em geral.

O modelo básico de RDF é definido através de três tipos de objetos: o recurso, as propriedades e a declaração. O recurso representa tudo que é descrito em RDF. As propriedades são as características que descrevem o recurso e a declaração é o conjunto formado pelo recurso juntamente com as suas propriedades e o valor destas.

3.2 APIs - *Application Program Interfaces* ou Interface de Programação de Aplicativos

API é a sigla para *Application Programming Interface* (Interface de Programação de Aplicativos), que não passa de um conjunto de rotinas e padrões de um software para utilizar suas funcionalidades. [Sampaio, 2007] Ou seja, são códigos que possibilitam usar recursos de um aplicativo da web 2.0 em outro.

O uso de APIs tornou-se cada vez mais rotineiro com a web 2.0. Caso um desenvolvedor precise criar uma ferramenta que se integre com outros aplicativos, é fundamental conhecer o API desta para poder ter acesso aos seus dados.

Os API são muito utilizados em mashups, como veremos na próxima seção.

3.2.1 *Mashups*

Mashup é um *website* ou aplicação que combina conteúdo de mais de uma fonte em uma experiência integrada. Ele é uma derivação do conceito de aplicação composta, que é uma aplicação que consome informação disponibilizada através de *Web Services* [Sampaio, 2007]. Uma aplicação *Mashup* reúne conteúdo de fontes diversas, utilizando geralmente APIs.

O desenvolvimento de *mashups* também utiliza RSS, que são fornecidos pelos *websites* para divulgar a informação. Por exemplo, através de um RSS é possível disponibilizar a previsão do tempo que poderia ser utilizada em um sistema de agricultura, indicando possibilidade de chuvas prolongadas ou tempestades. Várias inovações podem surgir por meio de combinações de dados vindos de fontes distintas.

O *Programmable Web*⁴, site especializado em *mashups*, lista 214 serviços online que podem ser integrados livremente entre si, possibilitando, teoricamente, a criação de mais de 45 mil aplicações online. Entretanto, na prática, muitas integrações podem não ter aplicação específica alguma [Felitti, 2007].

O Google é uma companhia que disponibiliza muitas APIs. Através delas é possível gerar inúmeros serviços, tais como criar um sistema de busca que utiliza a *cache* do Google, mostrar imagens de hotéis em um sistema de reservas por meio do Google Maps ou gerenciar datas de eventos em um sistema de marcação de festas usando APIs do Google *Calendar*.

A figura 3.5 apresenta um exemplo de uso de APIs do GoogleMaps. Este sistema foi desenvolvido por Rafael Dohms⁵ para atender aos congressistas do evento *PHP Conference Brasil 2006*. Os participantes indicavam o local de origem no momento do cadastro e a

⁴Disponível em <http://programmableweb.com>

⁵Disponível em <http://www.rafaeldohms.com.br/gmaps/phpconfbrasil2006/phpconfbrasil2006.htm>

partir disso era possível traçar uma estimativa de onde os participantes vinham e quantos quilômetros percorreram.

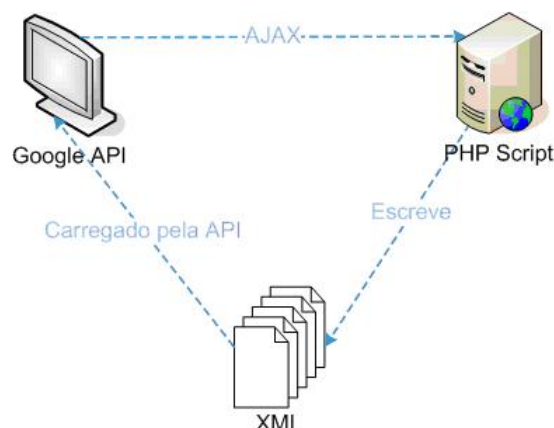


Figura 3.5: - Representação da utilização de APIs [Dohms, 2007]

O primeiro procedimento foi inserir o código, escrito em JavaScript, que contém a chave da API do Google Maps em um arquivo do *website*. Em seguida foi criado um script para interpretar os dados do arquivo XML, que continha a lista com a localização dos participantes. O próximo passo foi estabelecer uma conexão AJAX para o arquivo PHP, que calcula a distância do ponto central ao marcado pelo participante e retorna essa informação. A segunda etapa do sistema se trata de um arquivo PHP responsável por receber os dados do ponto criado e criar um registro no XML deste ponto. A figura 3.6 mostra o resultado o uso dessa API. [Dohms, 2007].

3.3 Blogs

Os *weblogs* ou somente *blogs* é um outro grande exemplo da massificação da tecnologia *Web 2.0* em termos de coletividade entre os usuários. Segundo dados da Technorati⁶, mais de 120 mil *blogs* são criados todos os dias. Embora alguns críticos considerem os *blogs* apenas como diários online, eles são importantes ferramentas de conversação, pois são através dos *blogs* que pequenas redes de amigos ou grupos interessados em um determinado conteúdo específico podem se interagir entre si, ou até mesmo entre outros grupos, causando assim efeitos significativos na rede.

Essa propagação de informação através dos *blogs* faz com que não apenas os grandes portais tenham importância, pois menor que seja o grupo em um *blog*, seus interesses segmentados fazem com os usuários se interajam com outros grupos, modificando de um modelo central de distribuição de conteúdo para uma verdadeira rede de compartilhamento de informação.

⁶<http://www.technorati.com/>

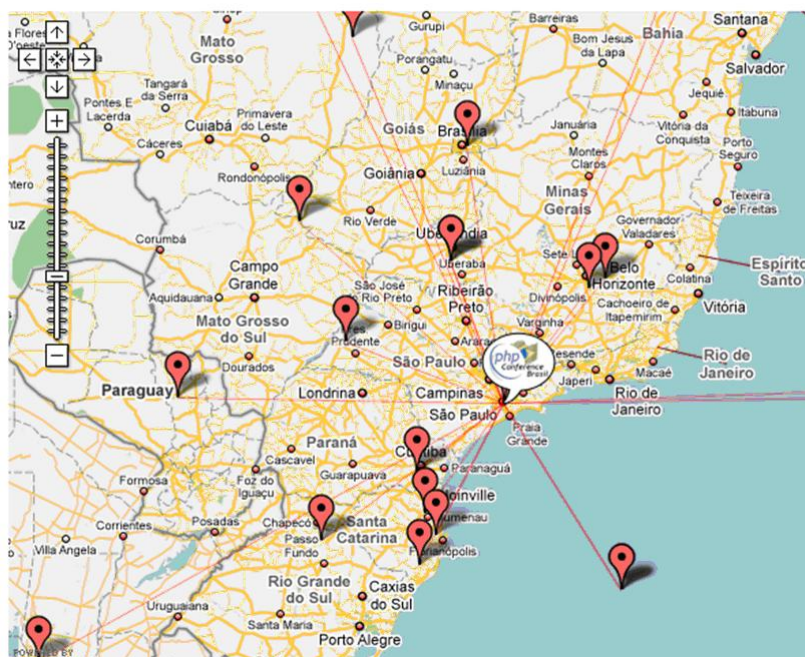


Figura 3.6: Visão geral do Sistema Localizador de Congressistas [Dohms, 2007]

Ferramentas de criação de *blogs*, tais como Wordpress⁷, Blogger⁸, Live Journal⁹ e outras, permitem que usuários sem o conhecimento em linguagens web tenham condições de criar seu próprio blog de acordo com o segmento que lhe interesse.

3.4 Wikis

São sites que permitem a edição da própria página utilizando ferramentas de formatação, sem necessidade de conhecimentos sobre HTML. Essa ferramenta possui uma linguagem mais simples chamada *WikiText*. Porém a utilização deste recurso é opcional, pois toda a formatação pode ser feita através de editores visuais.

No *Wikipédia*¹⁰, por exemplo, que é uma enciclopédia livre, os próprios usuários editam os documentos, publicam artigos, sem necessidade de profissionais específicos para atualização. A *Wikipédia* permite ao usuário a edição do conteúdo da página, por meio de editores visuais, sem que seja necessário conhecimento sobre HTML por parte do usuário, como mostra a figura 3.7.

Os *Wikis* geralmente possuem um conteúdo bem organizado, e de busca muito eficaz,

⁷<http://wordpress.org/>

⁸<http://www.blogger.com/>

⁹<http://www.livejournal.com/>

¹⁰<http://pt.wikipedia.org/>

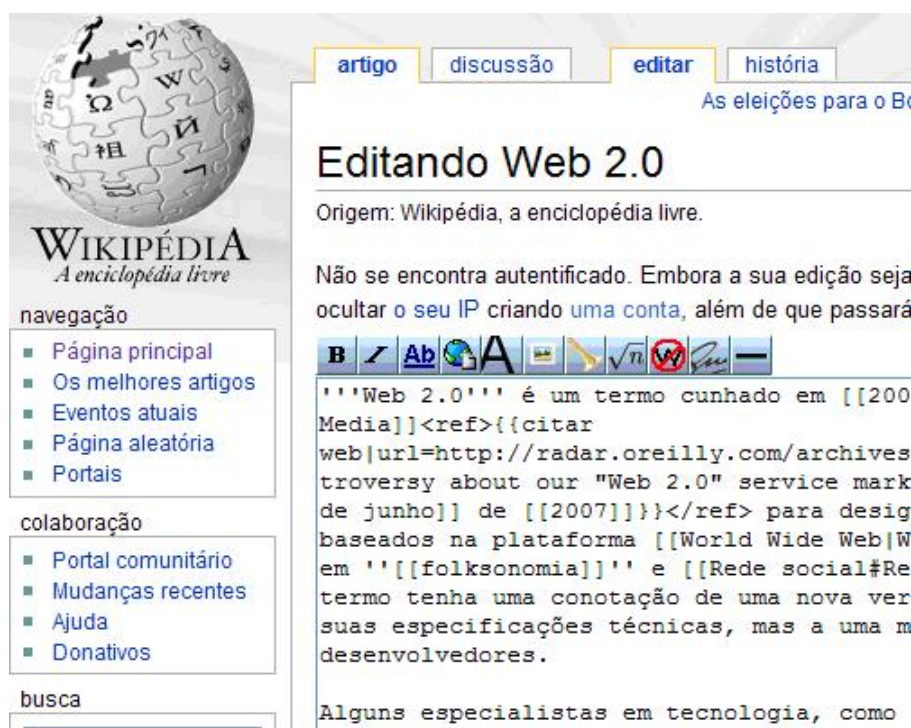


Figura 3.7: Editando um artigo no Wikipédia

assim o relacionamento dos *wikis* com os usuários se torna dinâmico e atrativo, fazendo uma grande rede social de compartilhamento de informações. Além da *Wikipédia*, existem outros tipos de *wikis*, que trabalham da mesma forma, como por exemplo, o *Lis Wiki*¹¹, *Library Success*¹² e outros.

3.5 Sites Sociais

Sites Sociais são aqueles que permitem os usuários criarem um perfil com sua descrição, hábitos, preferências, fotos, etc. Geralmente os usuários buscam nestes sites uma alternativa para conhecer e se comunicar com outras pessoas. A comunicação também é facilitada por meio desta ferramenta.

Os sites sociais estão ultrapassando a fronteira dos computadores. Alguns serviços como o *Twitter*¹³ e Radar já possuem versões para celular. No Brasil o Orkut é o site social mais usado, onde os usuários por meio de seu perfil, participam de comunidade de seu interesse, mandam recados para seus amigos, além de poder garantir sua privacidade como preferir, permitindo acesso de usuários do site somente as sessões que ele autorizar [JavaNoroeste, 2008].

¹¹<http://liswiki.org>

¹²<http://www.libsuccess.org/>

¹³<http://www.twiter.com.br/>

3.6 JOOMLA!

Como descrito nos capítulos anteriores, um *website* que não possua rica funcionalidade ou conteúdo atualizado, dificilmente obterá sucesso dentro da comunidade web.

Um Sistema de Gerenciamento de Conteúdo - SGC, (em inglês *Content Management Systems* - CMS), é um sistema gerenciador de Portais Corporativos. Portais e intranets que integram ferramentas necessárias para criar, gerenciar (editar e inserir) conteúdo em tempo real, sem a necessidade de programação de código, cujo objetivo é estruturar e facilitar a criação, administração, distribuição, publicação e disponibilidade da informação. Sua maior característica é a grande quantidade de funções presentes. [Navita, 2009]

O Joomla! é um poderoso CMS *open source* que permite a criação de *websites* com características *Web 2.0*. Suas duas características chaves, que são fáceis na administração e flexibilidade de modelagem, fazem dele o CMS mais usado pelos programadores web [North, 2008].

Um *website* Joomla! tem diversos elementos que trabalham juntos para produzir uma página *Web*. Os três principais elementos são conteúdo, módulos e o *template*. O conteúdo é o aspecto interno do *website*, os módulos adicionam funcionalidades e o *template* controla como o site é apresentado. [North, 2008]

Em resumo, o Joomla! é um bom sistema de gerenciador de conteúdos de baixo custo que tem a capacidade de produzir *websites* sofisticados, além de ser *open source* (GNU GPL¹⁴).

3.7 Wordpress

Wordpress é uma plataforma de *blogs open-source*, sob licença da GNU GPL, robusta e muito fácil de utilizar. O Wordpress permite a criação de artigos (*posts*), onde depois de publicados os posts aparecem na página principal do *blog*, de forma cronológica. Estes *posts* são depois arquivados tanto por categorias como por data, tornando muito fácil ter acesso à todos os artigos publicados.

Como em todos os *blogs*, o Wordpress está equipado com um sistema de comentários, que permite aos visitantes de *blog* deixar uma opinião sobre o artigo publicado. Isto favorece a interação e o sentido de comunidade, ao invés de uma simples página estática em HTML.

Este *software* é desenvolvido em PHP e suportado por uma base de dados mySQL. A vantagem deste *software* ser *open-source* está na possibilidade de poder alterar suas funcionalidades base por meio da adição de *plugins*, que são pequenos scripts que fornecem funcionalidades adicionais ao blog.

¹⁴GNU General Public License (Licença Pública Geral). [North, 2008]

O wordpress é a plataforma de *blogs* mais utilizada no mundo inteiro, impondo-se devido à sua facilidade de utilização e de adaptação a cada programador. Ele tem uma comunidade de desenvolvedores muito ativa, que produzem dezenas de plugins e temas wordpress diariamente. [Euller, 2009]

3.8 AJAX

A sigla AJAX, *Asynchronous Javascript And XML* ou JavaScript e XML Assíncronos, é utilizada para especificar um conjunto de tecnologias que possibilitam que as aplicações *Web* sejam mais dinâmicas, uma vez que, evitam o processo de carregamento de todo o conteúdo da página a cada nova requisição do usuário. Com o AJAX é possível atualizar apenas a informação solicitada. [Crane et al., 2007]

Desde o surgimento da internet, a interação é feita de forma síncrona, ou seja, para cada requisição em uma página *Web* era necessário o carregamento da página inteira no navegador, mesmo que a solicitação fosse somente de uma pequena parte dela.

Uma aplicação que utiliza o AJAX elimina o processo natural de *start-stop*¹⁵ de uma interação *Web* introduzindo uma camada AJAX intermediária entre o usuário e o servidor. Ao invés de carregar a página o navegador abre a camada AJAX, escrita em JavaScript e geralmente escondida em um frame. Esta camada é responsável pelo que o usuário vê e pela comunicação entre ele e o servidor. Ela permite que a interação entre aplicação e usuário seja feita de forma assíncrona, independente da comunicação com o servidor. [Garret, 2005].

A figura 3.8 apresenta a comparação da interação do servidor com o usuário, entre as aplicações web clássicas, e as que utilizando AJAX, como descritos nos parágrafos anteriores.

A utilização do AJAX se constitui em um recurso para tentar possibilitar às aplicações desenvolvidas para *Web* o mesmo contexto de aplicações desenvolvidos para *Desktop*, assim empregando o AJAX é possível conseguir em algumas aplicações *Web* aspectos como dinamismo e interatividade na interface gráfica, usando o conceito DOM (*Document Object Model* ou Modelo de Documento Objeto), por meio do JavaScript; recuperação assíncrona de dados, utilizando a classe *XMLHttpRequest*, camada de apresentação padronizada, utilizando XHTML e CSS.

O AJAX não é uma nova tecnologia. Em vez disso, é uma coleção de quatro tecnologias que se complementam. A tabela 3.2 resume essas tecnologias e o papel que cada uma tem de desempenhar.

¹⁵Início-Fim - Carregamento da página do início ao fim.

<i>JavaScript</i>	JavaScript é uma linguagem de criação de scripts de uso geral projetada para ser embutida dentro dos aplicativos. O interpretador JavaScript em um navegador <i>Web</i> permite interação programática com muitas das capacidades predefinidas no navegador. Aplicativos Ajax são escritos em JavaScript.
<i>Cascading Style Sheets (CSS)</i>	CSS oferecem uma maneira de definir estilos visuais reutilizáveis para elementos das páginas <i>Web</i> . Elas oferecem uma maneira simples e poderosa de definir e aplicar estilização visual consistentemente. Em um aplicativo Ajax, a estilização de uma interface com o usuário pode ser modificada interativamente por meio de CSS
<i>Document Object Model (DOM)</i>	O DOM representa a estrutura das páginas <i>Web</i> como um conjunto de objetos programáveis que pode ser manipulado com JavaScript. Criar Scripts com o DOM permite que um aplicativo Ajax modifique a interface com o usuário instantaneamente, redesenhando de forma partes da página.
<i>Objeto XMLHttpRequest</i>	O objeto <i>XMLHttpRequest</i> permite que programadores <i>Web</i> recuperem dados do servidor <i>Web</i> como uma atividade em segundo plano. O formato dos dados em geral é XML, mas funciona bem como quaisquer dados baseados em texto. Enquanto o <i>XMLHttpRequest</i> é a ferramenta de uso geral mais flexível para esse trabalho, há outras maneiras de recuperar dados do servidor.

Tabela 3.2: Os elementos-chave do Ajax [Crane et al., 2007]

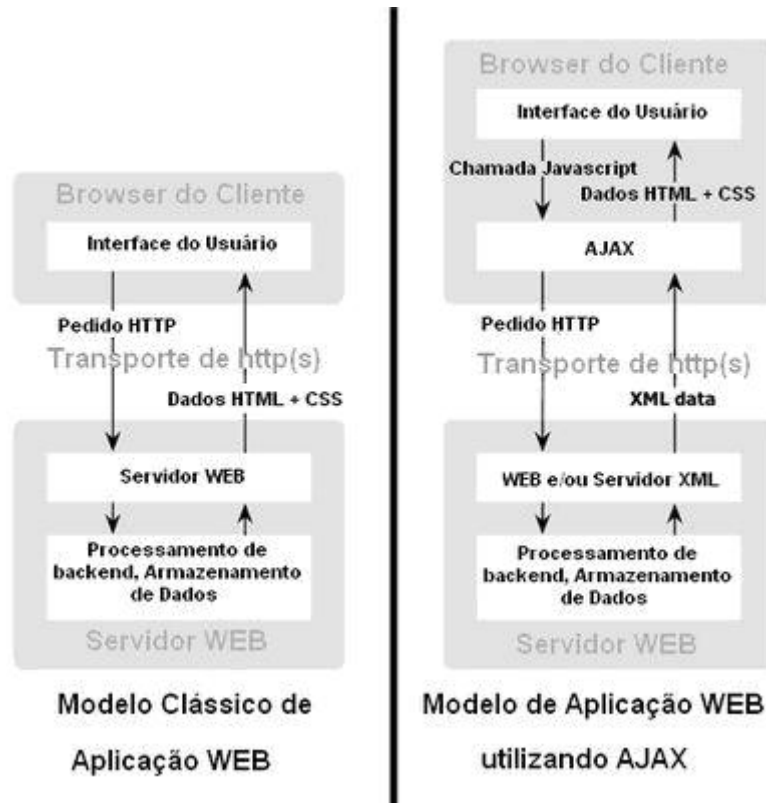


Figura 3.8: Modelo de Aplicações na Web: Clássico e com Ajax [Tosing, 2007]

3.8.1 JavaScript

O papel central no conjunto de ferramentas do Ajax é sem dúvida desempenhado pelo JavaScript. Um aplicativo, Ajax carrega um cliente completo na memória, combinando dados, apresentação e a lógica do programa. O JavaScript é a ferramenta utilizada para implementar essa lógica. O JavaScript é uma linguagem de programação de uso geral de descendência híbrida, com uma leve semelhança com a família das linguagens C. [Crane et al., 2007]

A grosso modo, o JavaScript pode ser caracterizado como uma linguagem de criação de scripts de uso geral, interpretada e fracamente tipada. Fracamente tipada significa que as variáveis não são declaradas especificamente como strings, inteiros ou objetos e que é possível atribuir valores de diferentes tipos à mesma variável. [Negrino e Smith, 2001]

Assim, o JavaScript é basicamente uma linguagem de scripts voltada para executar operações no lado do cliente. Ela permite uma maior interatividade com o usuário, através de validações de campos, mensagens de alerta, manipulação dos eventos de botões e ainda uma mudança dinâmica no estilo da página. A contribuição do JavaScript para o AJAX é muito ampla, pois é por meio deste mecanismo que tornou-se possível atualizar partes de uma página.

Dentro do ambiente do navegador Web, partes da funcionalidade nativa do naveg-

ador, incluindo CSS, DOM e os objetos *XMLHttpRequest* são expostos ao mecanismo JavaScript, permitindo que os desenvolvedores controlem praticamente toda a página. Embora o ambiente JavaScript que encontramos no navegador seja pesadamente preenchido por objetos específicos ao navegador, a linguagem subjacente é simplesmente isso, uma linguagem de programação.

3.8.2 *Cascading Style Sheets* ou Folha de Estilo em Cascata (CSS)

As folhas de estilo em cascata (CSS) são uma parte bem estabelecida do design *Web*, utilizadas tanto em aplicativos *Web* clássicos e também no Ajax. Uma folha de estilo oferece uma maneira centralizada de definir categorias de estilos visuais, que podem ser aplicadas a elementos individuais em uma página concisa.

Além dos elementos fundamentais de estilização, como cores, bordas, imagens de fundo, transparência e tamanho, as folhas de estilo podem definir a maneira como os elementos são organizados em relação um ao outro e também sua interatividade simples com o usuário, permitindo que efeitos visuais bastante poderosos sejam alcançados apenas por meio das folhas de estilo. [Crane et al., 2007]

Em um aplicativo *Web* clássico, as folhas de estilo fornecem uma maneira útil de definir um estilo em um único lugar que poder ser reutilizado em diferentes páginas *Web*. Com o Ajax, não há o pensamento em termos de uma sucessão rápida de páginas, mas as folhas de estilo ainda fornecem um repositório útil de visuais predefinidos que podem ser aplicados dinamicamente a elementos com um mínimo de código.

Com o CSS é possível estruturar em um só documento toda a aparência do site. Desta forma, qualquer mudança pode ser feita rapidamente alterando apenas um arquivo.

3.8.3 *Document Object Model* ou Modelo de Documento Objeto (DOM)

O DOM é uma API que permite aos programadores acessar e modificar o conteúdo de documentos HTML ou XML. Isto proporciona uma representação estrutural de um documento, habilitando o usuário a modificar seu conteúdo e apresentação visual. Essencialmente, isto conecta páginas *Web* à scripts ou linguagens de programação. [Sampaio, 2007].

Todas as propriedades, métodos, e eventos disponíveis para o desenvolvimento *Web* para manipulação a criação de páginas *Web* são organizados dentro dos objetos, ou seja, o documento objeto que representa o documento por si mesmo, a tabela objeto que representa a tabela de elementos HTML, e assim por diante. Estes objetos são acessíveis via linguagens de scripts nos mais recentes navegadores *Web*.

O DOM é normalmente o mais usado em conjunto com o JavaScript. Entretanto, o DOM foi desenhado para ser independente de qualquer linguagem de programação particular, fazendo a representação estrutural do documento disponível em uma simples, API consistente. Embora nosso foco seja no JavaScript por todo este site, implementações do DOM podem ser construídas para qualquer linguagem.

O *World Wide Web Consortium*¹⁶ estabelece um padrão para o DOM, chamado o W3C DOM. Deve, agora que os mais importantes navegadores implementaram isto corretamente, ativar poderosas aplicações multi navegadores.

Com o API DOM é possível criar documentos, navegar pela estrutura de um documento qualquer, incluir, alterar e apagar nós dos documentos XML. Ou seja, na grande maioria das vezes utilizar-se-á a API DOM para manipular uma página *Web* pelo AJAX. [Soares, 2006]

3.8.4 XMLHttpRequest

O *XMLHttpRequest* é uma API, que disponibiliza as funcionalidades HTTP (*Hyper Text Transfer Protocol* - Protocolo de Transferência de Hiper-Texto) permitindo a troca de dados entre cliente e servidor. [Tosing, 2007]

Ele é um objeto JavaScript que permite a conexão assíncrona entre a página e o servidor *Web*. Ele faz requisições ao servidor *Web* sem paralizar o navegador ou recarregar a página atual.

Geralmente, para trocar informações com o servidor é usado o XML. Esta é uma das principais tecnologias do AJAX, já que é por meio dela que a comunicação assíncrona pode ser realizada.

O *XMLHttpRequest* foi introduzido em 2000, principalmente para permitir que o Microsoft Outlook *Web Access* exibisse e-mails sem notificação. Desde então, aplicativos AJAX ganharam popularidade por sua capacidade de trocar dados de maneira assíncrona com um servidor e depois exibir aqueles dados sem ter de recarregar a página da *Web* na qual eles aparecem.[Tosing, 2007]

Conforme o AJAX ganhou popularidade, a adoção do XMLHttpRequest excedeu significativamente as expectativas originais. Inicialmente foi disponibilizada pelo Internet Explorer 5.0, navegador da Microsoft e depois pelos demais navegadores. Porém, enquanto os demais navegadores seguiram as especificações da W3C, o Internet Explorer disponibilizou o recurso na forma de um ActiveX, o qual existe em duas versões diferentes para o objeto XMLHttpRequest questões as seguintes: Microsoft.XMLHTTP e Msxml2.XMLHTTP, sendo esta última a mais recente.

¹⁶<http://www.w3.org/>

A figura 3.9 representa o instanciamento da classe *XMLHttpRequest* para o Internet Explorer e os demais navegadores.

```
1  var XMLHttpRequest;
2
3  try {
4      XMLHttpRequest = new XMLHttpRequest(); // De acordo com a especificação W3C
5  } catch (e) {
6      try {
7          XMLHttpRequest = new ActiveXObject("Msxml2.XMLHTTP");
8      } catch (ee) {
9          try {
10             XMLHttpRequest = new ActiveXObject("Microsoft.XMLHTTP");
11         } catch (eee) {
12             alert("Não consegui criar o objeto XMLHttpRequest: " + eee.description);
13         }
14     }
15 }
```

Figura 3.9: Instanciamento da classe XMLHttpRequest

No trecho de código apresentado pela figura 3.9, na primeira linha tem uma variável é chamada XMLHttpRequest, onde depois a variável é instanciada como sendo objeto da classe *XMLHttpRequest*, conforme especificação estabelecida pela W3C. Em seguida, caso tenha ocorrido algum erro na criação, há a tentativa de instanciar o objeto utilizando o padrão do *browser* Internet Explorer, de acordo com suas versões.

Capítulo 4

Estudos de Caso

Ao longo deste capítulo serão mostrados os principais conceitos e tecnologias da *Web* 2.0 aplicada na construção do *website* do Departamento de Ciência da Computação da Universidade Federal de Goiás, Campus Catalão, e o desenvolvimento de um *Blog* Universitário para o Campus, com o intuito de verificar na prática se as tecnologias da *Web* 2.0 proporcionam ganhos na utilização da mesma.

4.1 Desenvolvimento do *WebSite* do Curso de Ciência da Computação da Universidade Federal de Goiás, Campus Catalão

O objetivo, para este trabalho, foi transformar o *website* atual, figura 4.1, em um serviço mais interativo e permitir a distribuição do conteúdo gerado pelo DCC, usando recursos da *Web* 2.0.

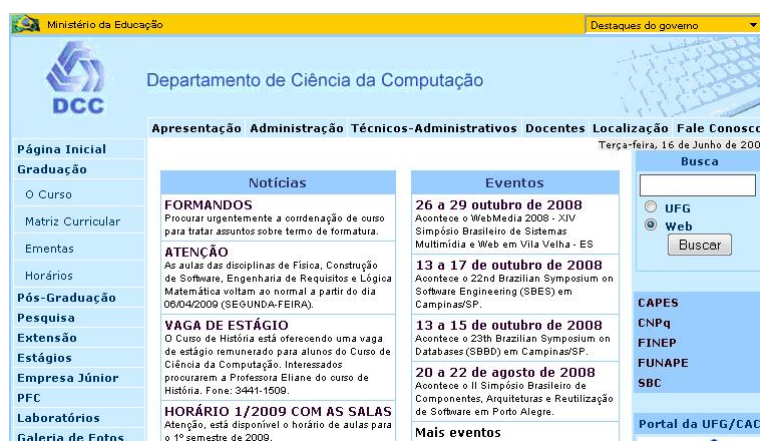


Figura 4.1: *WebSite* atual do DCC

4.1.1 Plano de Desenvolvimento

O Conteúdo

O conteúdo principal do *website* do DCC foi mantido, apenas realizou-se uma reformulação na forma que ele é apresentado, visando atingir principalmente o usuário. Uma das propostas é dar destaque às seções mais acessadas pelos alunos, para torná-las fáceis de serem encontradas, permitindo assim que os mesmos encontrem as informações que procuram com mais rapidez.

Funcionalidades

O desenvolvimento do *website* do DCC em *Web 2.0* traz de inovador, se comparado ao site atual, uma série de recursos de atualização de conteúdo por parte da administração do *website*, e de interação por parte do usuário.

Os administradores do site terão total controle sobre o conteúdo completo da página, podendo alterar informações do curso, horários, professores, tudo via o painel produzido, sem ser necessário entender de programação *Web*, como é no *website* atual, assim a atualização e renovação do conteúdo se torna mais rápida e fácil.

Os usuários passarão a ter opção de imprimir as notícias, ou enviar para qualquer email sobre cada sessão do site, além de inovações como enquetes e pedidos de requerimentos à coordenação do curso.

Layout

Como descritos na sessão 2.6.3.1 desta monografia, a simplicidade facilita ao usuário encontrar o que deseja. Assim, ao projetar a página é interessante preocupar-se em não perder o foco nas informações.

Os menus principais de acessos foram colocados na lateral esquerda da página, e os menus de informação do curso na parte superior, deixando o centro do *layout* somente para as notícias, que apareceram de forma bem distribuída, permitindo ao usuário a visualização dela já quando acessa página.

A figura 4.2, mostra o layout produzido para o desenvolvimento do *website* e como as informações estão distribuídas ao longo da página.

Em comparação ao *website* atual, esse layout prende o usuário especificamente no conteúdo, com cores mais suaves, e sessões destacadas, facilitando assim a visualização rápida do conteúdo pelo usuário.



Figura 4.2: WebSite do DCC com característica Web 2.0

Servidor

Para o desenvolvimento do *website* do DCC não foi necessária nenhuma alteração no servidor atual, que atende todos os requisitos para funcionamento da página. Foi usado um banco de dados MySQL para armazenar os dados e todo o *website* foi produzido com a união de PHP e AJAX.

Como *website* produzido faz uso de XMLHttpRequest, ele obtém um acesso mais rápido a cada sessão clicada pelo usuário, pois sua conexão com o servidor é de forma assíncrona, não necessitando de um novo carregamento da página a cada sessão, como descrito na sessão 3.8.4.

4.1.2 Distribuição do conteúdo através de RSS

Todas as sessões do *website* do DCC serão distribuídas através de RSS, tanto o conteúdo dos menus laterais, como as notícias internas, que são atualizadas com frequência, englobando informações para os alunos, docentes e comunidade em geral. A distribuição deste conteúdo permite que os usuários fiquem informados sobre as atividades do DCC sem mesmo acessar o *website*, como descrito na sessão 3.1.

Para que fosse possível a distribuição desse conteúdo, foi necessário o desenvolvimento de um documento XML, utilizando RSS 2.0. Para indicar a presença do RSS foi utilizado um dos ícones padrão para este serviço, como mostra a marcação na figura 4.3.

O documento em XML é atualizado a cada inserção, alteração ou remoção de notícias e editais. Estas operações são feitas através de um painel de administração do site. Esse



Figura 4.3: Visualização do ícone que indica o uso de XML

procedimento é feito buscando os dados, após a modificação, na base de dados. A partir disso a estrutura é gerada e permanece sempre atualizada.

Por meio da página de visualização da RSS, o usuário pode ver o título e uma breve descrição da notícia ou do edital. Também escolhe o serviço que utilizará para agregar estas informações. A página e algumas opções de agregadores RSS são mostradas na figura 4.4.

No apêndice A1 desta monografia, encontra-se o código XML utilizando RSS 2.0.

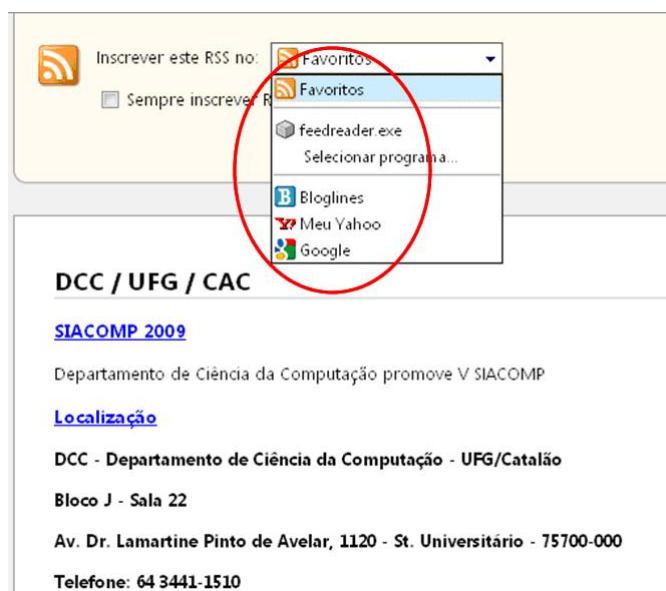


Figura 4.4: Página de visualização da RSS e dos agregadores de feeds (marcação)

4.1.3 Folha de Estilos (CSS)

O layout do site do DCC é formado por uma folha de estilos (CSS), que estipula as cores e imagens que são utilizadas no *layout*, e como cada uma são apresentadas na página.

O uso de CSS permite que o *layout* da página possa ser alterado pelo administrador, como descrito na sessão 3.8.2, sem que tenha que ser feito mudanças na estrutura do conteúdo na página.

O código da folha de estilos do *website* do DCC encontra-se no apêndice A2.

4.1.4 Paine de Administração do Website

O *website* do DCC é composto por um painel de administração, que permite alterar várias funcionalidades na página, além da inserção, edição e remoção de notícias.

Para se ter acesso ao painel de administração é necessário o uso de um login e senha, onde cada usuário do painel se enquadra em um nível de permissão, sendo eles Gerente, Administrador e Super Administrador.

Ao usuário Gerente, é permitido inserir, editar e remover notícias, inserir e remover novos menus na página, inserir e remover categorias, sessões e componentes, e também alterar o conteúdo interno dos menus. O usuário gerente é normalmente o que vai alimentar o site com notícias, arquivo, enquetes, entre outras.

O usuário Administrador possui a permissão para inserir novos usuários, administrar os módulos, que são como o conteúdo é distribuído ao longo da página, ou seja, onde cada sessão ficará localizada, e mais os recursos do usuário Gerente.

Já o usuário Super Administrador, é aquele que detém domínio total sobre o *website*, ele tem permissão de alterar o *layout*, configurações internas, e todas as outras funcionalidades que o *website* possui.

A figura 4.5 mostra a tela principal do Paine de Administração do *website* do DCC.



Figura 4.5: Paine de Administração do *website* do DCC

4.2 Blog Universitário

Desenvolver um *blog* para a Universidade Federal de Goiás tem como objetivo fornecer uma interação entre os cursos, e disponibilizando-o para alunos, professores e comunidade em geral, onde cada usuário possa encontrar assuntos sobre determinados temas relativos aos cursos do Campus Catalão.

Como vistos nos capítulos anteriores, os *blogs* são uma forma de trazer aos usuários um conteúdo de seu interesse permitindo a participação de todos na discussão do mesmo. Para atrair os olhares do usuário, são necessários recursos gráficos interessantes, além de um *layout* atraente e dinâmico.

A figura 4.6 mostra como foi projetado o *layout* do *blog*, com cores e tons mais sutis, a fim de prender o usuário no conteúdo.



Figura 4.6: *Layout* do Blog Universitário

O *blog* foi dividido em categorias, e subcategorias, sendo as categorias as áreas do conhecimento e as subcategorias os cursos a que elas pertencem.

A página inicial do *blog* retorna todos os artigos (*post*) em ordem cronológica, ou seja, toda vez que um novo artigo é inserido ele aparece automaticamente na página inicial, se o usuário quiser visualizar somente os artigos de uma categoria específica, basta ele clicar na categoria, que os artigos serão filtrados.

4.2.1 Distribuição do conteúdo através de RSS

Assim como no *website* do Curso de Ciência da Computação, no *blog* todas as categorias estão distribuídas através de RSS, como descritos na sessão 4.1. O usuário pode se informar sem acessar a página tanto em todas as categorias conjuntos como em uma só específica.

Para indicar o uso de RSS foi utilizado um dos ícones padrão para este serviço. Na figura 4.7 mostra o ícone para o RSS geral e para as categorias específicas.

Assim como na construção do *website* do Departamento de Ciência da Computação, descrito anteriormente, foi necessário o desenvolvimento de um documento XML, uti-



Figura 4.7: Ícone que indica uso de RSS, azul para geral e vermelho para categorias

lizando RSS 2.0, onde o documento em XML é atualizado a cada inserção, alteração ou remoção de notícias e editais. Estas operações são feitas por meio de um painel de administração do site. Esse procedimento é feito buscando os dados, após a modificação, na base de dados. A partir disso a estrutura é gerada e permanece sempre atualizada.

4.2.2 Participação

Com o objetivo de estimular a participação e interação entre os usuários, é disponibilizada no *blog* uma sessão onde os usuários podem deixar seus comentários, citar comentários de outros usuários, ou mesmo dar sugestões ao autor do *post* como mostra a figura 4.8.

Figura 4.8: Formulário pra envio de comentários

4.2.3 CSS e Recursos Gráficos

A *Web 2.0* proporciona ao usuário uma interação intensa com a página. Como descrito na sessão 4.8.2, para o *blog* universitário foi criado um CSS que permite tanto ao administrador do *blog* mudar o *layout*, cores e ajuste na página, quanto ao usuário, que pode aumentar o tamanho da fonte, alargar e encolher a página, sendo necessário apenas clicar nos recursos, como mostra a figura 4.9.

O código da folha de estilos do *blog* encontra-se no Apêndice A3 desta monografia.

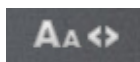


Figura 4.9: Recursos Gráfico do Blog

4.2.4 Painel de Controle

O *blog* é um tipo de *website* onde várias pessoas podem estar administrando-o, com envio de artigos, avaliação de comentários, etc. Foi desenvolvido um painel de administração que permite, através de um usuário e senha, e de acordo com seu nível de permissão, fazer determinadas ações dentro do blog.

Assim como no *website* do Departamento de Ciência da Computação, foi criado categorias de usuários, onde cada um tem seu nível de acesso dentro do painel, sendo eles, o usuário colaborador, autor, editor, e administrador.

O usuário colaborador tem permissão de criar e editar *post* criados por ele, porém ele não tem permissão de publicá-los até que o mesmo seja revisado por um usuário autor, editor ou administrador.

O usuário autor tem permissão para publicar *post*, dentro das categorias existentes no blog, podendo autorizar a publicação de *post* escritos por usuários colaboradores ou mesmo editá-los. O usuário editor tem permissão para criar novas categorias dentro do *blog*, ou mesmo excluir alguma existente, além das mesmas permissões de um usuário autor. O editor normalmente é o responsável por evitar qual problema dentro do *blog*, com envio de artigos indevidos, categorias não necessárias, entre outros. O usuário administrador é o que possui acesso a todo *blog*, podendo criar usuários, alterar o *layout* da página, e fazer todas as alterações dentro do *blog*.

Envio de Artigos

O sistema para enviar *post* para o *blog* é simples e de fácil manuseio, até mesmo por usuários que não possuem conhecimentos em linguagens *Web*. O usuário entra com o texto que será publicado em um editor de texto, similar aos editores *desktops*, como mostra a figura 4.10.

Além de digitar textos, o usuário pode também trabalhar com mídias digitais, tais como imagens, vídeos, entre outros, incluindo junto ao artigo que será publicado.

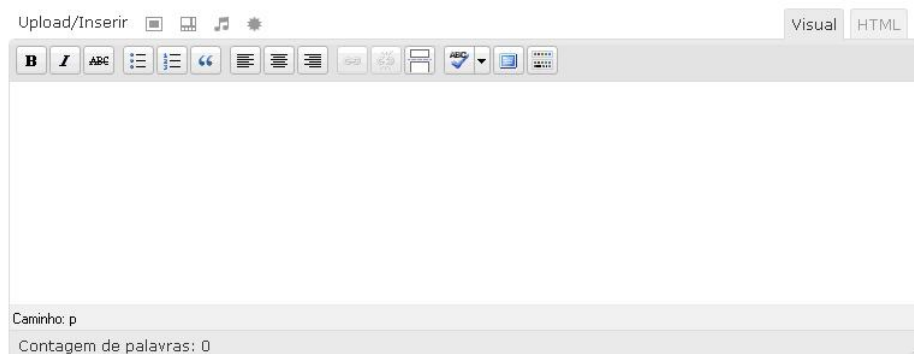


Figura 4.10: Editor de texto para envio de *post*

4.3 Avaliação dos Produtos

Todos os dois estudos de casos apresentados se encaixam nos padrões *Web 2.0*, e foram implementados usando os recursos Joomla! e Wordpress com PHP, MySQL e AJAX.

O uso de RSS permitiu que fosse distribuído um conteúdo antes acessado apenas por meio do site. A folha de estilos (CSS) tornou a atualização de *layout* e definição de estilos um tanto quanto simples e rápida para modificação.

A reestruturação permitiu uma navegação mais simples e rápida. Assim pode-se dizer que objetivo proposto foi alcançado com sucesso quanto ao desenvolvimento do *website* do DCC e o *Blog* Universitário dentro dos padrões, avaliando e conhecendo todos os conceitos *Web 2.0*.

O uso das tecnologias *Web 2.0* trouxe para o Departamento de Ciência da Computação um ganho em qualidade, onde o novo *website* traz para as informações de forma concisa, e permite ao usuário uma interação maior com o conteúdo.

O *blog*, com as tecnologias *Web 2.0*, traz aos professores e aos alunos um nova forma de se atualizar e publicar materiais de pesquisa, discutir e expressar opiniões, bem como a aproximação entre os cursos da Universidade Federal de Goiás.

Conclusão

Foram apresentadas nesta monografia algumas definições para o termo *Web 2.0*, as principais características, os serviços e tecnologias envolvidas. As particularidades no comportamento do usuário, do conteúdo, *design* e da publicidade também foram abordados. A visualização prática dos conceitos e da evolução tecnológica da *Web 2.0* foi analisada através de exemplos, ao longo da monografia, e detalhado com o estudo de caso, onde foi desenvolvido um *blog* para Universidade Federal de Goiás, Campus Catalão, e um *website* em *Web 2.0* para o Departamento de Ciência da Computação.

Através dos exemplos mostrados e do estudo de caso pode-se concluir que os conceitos e tecnologias causaram uma grande revolução nos serviços *Web*. A maior mudança ocorrida nos aplicativos é a participação do usuário. As inovações surgiram com o intuito de possibilitar e estimular o crescimento desta participação. Os utilizadores dos serviços passaram a criar e administrar o conteúdo, deixando para as empresas apenas a função de gerar tecnologias e ferramentas para publicação e distribuição do conteúdo.

O ponto principal da *Web 2.0* é o envolvimento de pessoas no processo de desenvolvimento. Ocorreu uma migração para a *Web* social, voltada para a colaboração, participação e descentralização. Desta forma, a *Web* passou de uma ferramenta para uma plataforma, que dá suporte ao desenvolvimento de inúmeros serviços e integração dos mesmos.

A principal vantagem da *Web 2.0* é maior facilidade de criação de conteúdos *online*. Juntamente com esta vantagem, tem-se uma melhor categorização de conteúdos, o que leva a ser muito mais fácil encontrar a informação que se procura. Por outro lado, isso leva também aquele que é o grande aspecto da *Web 2.0*: a partilha de conteúdos, onde as ferramentas que permitem a criação do conteúdo permitem também a partilha imediata, o que reduz o tempo e as aplicações usadas para tal.

Algumas críticas são feitas ao termo indicando que todo o movimento ao redor da *Web 2.0* foi um golpe de *marketing*. Porém, o sucesso dos aplicativos que representam esta evolução contrasta com tais comentários. Ainda há muito para ser feito no sentido de possibilitar acesso irrestrito a todos e desenvolver mecanismos que permitam participação e colaboração dos usuários. A grande parte das companhias ainda desenvolve serviços classificados como *Web 1.0*, no entanto, a evolução está ocorrendo e já mostra indícios de que atingirá mais adeptos e se tornará uma realidade em poucos anos.

Dado o fato de que todas estas novas ferramentas funcionam à base de perfis pessoais, é muito mais comum a criação de contas em diferentes sites para poder utilizar suas funcionalidades. Ao mesmo tempo têm-se novos sites sociais e a maior facilidade de encontrar informação *online*. Um problema muito específico que foi criado com as chamadas redes sociais que expõe os usuários a todos que porventura utiliza tais redes, tendo como consequência facilidades para criminosos, para que os mesmos possam encontrar e aliciar vítimas *online*, ou até mesmo espalhar certas ideologias e influenciar possíveis seguidores, sejam elas políticas, religiosas, etc.

Para elaboração e execução deste trabalho, foi necessária uma grande pesquisa bibliográfica quanto ao real conceito *Web 2.0*, visto que os profissionais da área divergem muito quanto a sua definição, bem como o que ela propõe. Esta foi a principal dificuldade encontrada, pois por ser um tecnologia emergente, a quantidade de artigos científicos e livros sobre o tema são escassos, e geralmente apresentam idéias divergentes entre si.

Outra dificuldade encontrada foi no desenvolvimento do blog e do *website* para o Departamento de Ciência da Computação, visto que o AJAX é uma união de tecnologias que envolvem bastante estudo e dedicação para adquirir um bom conhecimento para desenvolver a página e o *blog* sem apresentarem nenhum tipo de *bugs*, e também ter a capacidade de tratar qualquer tipo de erro por parte do usuário sem afetar o conteúdo dos aplicativos.

O *website* do DCC, como trabalhos futuros, pode ser melhorado produzindo um sistema de notas e disciplinas em Ajax, com login e senha, onde o aluno pode acessar seu desempenho nas disciplinas, além também de um desenvolvimento de um sistema como ferramenta do *website* do DCC nos moldes do sistema Moodle¹, onde os docentes do curso podem deixar exercícios para os alunos, avaliações, respectivas notas alcançadas, etc.

Ainda como trabalhos futuros, apesar da *Web 2.0* ainda não ter se tornado um conceito concreto e utilizado por todos, já existem algumas discussões sobre tecnologias e ferramentas para a próxima geração da *Web*: a *Web 3.0* ou *Web* semântica. Esta fase seria o momento de aprender a interpretar, entender e obter conclusões a partir do conteúdo disponível na *Web*. Este ideal já é discutido há alguns anos, ainda na *Web 1.0*, e com o movimento da *Web 2.0* ele pode estar próximo de ser alcançado. Porém, o processo deverá ser evolutivo e natural, sem desconsiderar os avanços alcançados até o momento.

Ainda no escopo da *Web 2.0*, há muito para progredir e discutir sobre as mudanças ocorridas. Com a produção de conteúdo em larga escala, será necessário, em breve, o estudo de tecnologias para suportar tal densidade e formas de organizar e armazenar estas informações. Além disso, a participação do usuário muda o contexto de inúmeros setores, tais como a comunicação na *Web*, a maneira de obter lucro, a posse dos dados,

¹Moodle é um sistema de administração de atividades educacionais

entre outros. Os conceitos da *Web 2.0* e as suas conseqüências na sociedade ainda precisam ser amplamente discutidos e aperfeiçoados.

Referências

- Berners-Lee, T., Hendler, J., e Lassila, O. (2001). *The Semantic Web*. Disponível em: <http://www.sciam.com/article.cfm?id=the-semantic-web>. Acesso 18/06/2008.
- Brandt, B. (2006). *Web 2.0*. Disponível em http://bergbrandt.com.br/v1/arquivos/artigos/20061006_web2.pdf. Acesso em 04/05/2009.
- Carvalho, M. (2006). *A trajetória da Internet no Brasil: Do surgimento das redes de computadores à instituição dos mecanismos de governança*. Disponível em <http://www.nethistory.info/Resources/Internet-BR-Dissertacao-Mestrado-MSavio-v1.2.pdf>. Acesso em 26/08/2009.
- Condy, A. (2006). *Semântica: Do XHTML aos Padrões RDF*. Disponível em <http://www.mxstudio.com.br/views/tutorial.php?act=view&cid=14&aid=836>. Acesso em 06/05/2009.
- Crane, D., Pascarello, E., e James, D. (2007). *Ajax em Ação*. Editora Prenhall.
- Dohms, R. (2007). *Google Maps API: Um exemplo prático e comentado*. Disponível em <http://www.rafaeldohms.com.br/2006/12/06/google-maps-api-um-exemplo-pratico-ecomentado/pt/>. Acesso em 29/03/2009.
- Euller, G. (2009). *Primeiros Passos com Wordpress*. <http://blog.euler.eti.br/wordpress/wordpress-primeiros-passos>. Acesso em 13/06/2009.
- Felitti, G. (2007). *Mashups: Entenda a combinação de conteúdo digital em ascensão na web*. Disponível em http://idgnow.uol.com.br/internet/2007/05/24/idgnoticia.2007-05-24.3179902089/IDGNoticiaPrint_view/. Acesso em 28/03/2009.
- Fontes, A. (2006). *Web 2.0 e Ajax*. Disponível em <http://www.sejug.org/docs/ajax/Web2.0-Ajax.pdf>. Acesso em 30/04/2008.
- Garret, J. (2005). *Ajax: A New Approach to Web Applications*. Disponível em <http://www.adaptivepath.com/publications/essays/archives/000385.php>. Acesso em 03/04/2009.
- JavaNoroeste (2008). *Ajax e Web 2.0*. Disponível em http://www.javanoroeste.com.br/2006/jales/apresentacao_jales.pdf. Acesso em 29/04/2008.

- Jr, G. A. (2005). *Web 2.0: a nova internet é uma plataforma*. Disponível em <http://webinsider.uol.com.br/print.php?id=2656>. Acesso em 01/06/2009.
- Kaneco, P., Mendes, A., Souza, A., e Crubellate, J. (1999). *O Comércio eletrônico como estratégia de marketing e algumas considerações sobre o seu uso por uma empresa Maringaense*. P. Kaneco and A. Mendes and A. Souza and J. Crubellate. Acesso em 12/06/2008.
- Licklider, J. e Robert, W. (1968). *The Computer as a Communication Device*. Disponível em <http://gatekeeper.dec.com/pub/DEC/SRC/publications/taylor/licklider-taylor.pdf>. Acesso em 13/06/2008.
- Maia, M. (2008). *Internet Marketing: Web 2.0 E Divulgação Viral*. Disponível em <http://www.artigonal.com/marketing-na-internet-artigos/internet-marketing-web-20-e-divulgacao-viral-642601.html>. Acesso em 16/04/2009.
- Monteiro, J. (2005). *Uma história da internet*. Disponível em: <http://www.forevernet.pt/mjm/docs/Internet.pdf>. Acesso em 11/06/2008.
- Navita, P. (2009). *O que é um CMS?* Disponível em <http://www.navita.com.br/portal/newsroom/definicoes/cms.html>. Acesso em 13/05/2009.
- Negrino, T. e Smith, D. (2001). *JavaScript para World Wide Web*. Editora Campus, Rio de Janeiro, RJ, Brasil.
- Nielsen, J. (2000). *Projetando Websites: Designing Web Usability. Tradução de Ana Gibson*. Editora Elsevier.
- North, B. M. (2008). *Joomla! Guia do Operador*. Editora AltaBooks, Rio de Janeiro, RJ, Brasil.
- Nottingham, M. (2005). *RSS Tutorial*. Disponível em <http://www.mnot.net/rss/tutorial/>. Acesso em 06/05/2009.
- Oliveira, R. (2008). *Web semântica: novo desafio para os profissionais da informação*. Disponível em <http://www.sibi.ufrj.br/snbu/snbu2002/oralpdf/124.a.pdf>. Acesso em 23/09/2009.
- O'Reilly, T. (2005). *Web 2.0 - Principles and Best Practices*. Disponível em http://oreilly.com/catalog/web2report/chapter/web20_report_excerpt.pdf. Acesso em 28/04/2008.

- Primo, A. (2007). *O aspecto relacional das interações na Web 2.0*. Disponível em <http://www6.ufrgs.br/limc/PDFs/web2.pdf>. Acesso em 13/03/2008.
- PUC-RIO (2008). *Web 2.0: uma visão geral da Internet Pós-Bolha.com*. Disponível em http://mtristao.com/blog/wp-content/uploads/2008/08/0610433_2008_cap_2.pdf. Acesso em 23/09/2008.
- Ribeiro, G. (2008). *Microformats Cheat Sheet*. Disponível em <http://blog.gustavoribeiro.net/microformats-cheat-sheet>. Acesso em 02/06/2009.
- Ruby, S. (2009). *Rss2.0 And Atom 1.0 Compared*. [Rhttp://www.intertwingly.net/wiki/pie/Rss20AndAtom10Compared?action=print](http://www.intertwingly.net/wiki/pie/Rss20AndAtom10Compared?action=print). Acesso em 11/06/2009.
- Sampaio, C. (2007). *Web 2.0 e Marshups: Reinventando a Internet*. Editora Brasport, Rio de Janeiro, RJ, Brasil.
- Sartori, R. (2001). *Utilização da Internet pelas empresas de Maringá sob a Ótica de Web Sites*. Disponível em http://www.abepro.org.br/biblioteca/ENEGEP1999_A0351.PDF. Acesso em 12/06/2008.
- Sena, C. (2007). *A Cauda Longa*. Disponível em <http://blog.carlossena.com/resenhas/resenha-a-cauda-longa/>. Acesso em 01/06/2009.
- Smith, D. (2008). *Preparado para a web 2.0?* Disponível em <http://info.abril.com.br/corporate/gartner/preparado-para-a-web-20.shtml?2>. Acesso em 15/04/2009.
- Soares, W. (2006). *AJAX (Asynchronous JavaScript And XML): guia prático para Windows*. Editora Érica.
- Tavares, T., Teodoro, G., Nogueira, D., Coutinho, B., Meira, W., e Guedes, J. (2008). *Estratégias de Balanceamento de Carga em Servidores Web Transacionais*. <http://homepages.dcc.ufmg.br/dorgival/artigos/sbrc2004a.pdf>. Acesso em 19/09/2008.
- Tosing, S. (2007). *Php com AJAX na Web 2.0*. Editora Ciência Moderna.
- Zambonini, D. (2006). *Why you should left Web 2.0 into your hearts*. Disponível em http://oreillynet.com/xml/blog/2006/08/why_you_should_let_web_20_into.html. Acesso em 05/05/2009.

Apêndice A

Código Fonte

A.1 RSS 2.0 - Site do Departamento de Ciência da Computação

```
1  <?php
2
3  defined('JPATH_BASE') or die();
4
5  class JDocumentRendererRSS extends JDocumentRenderer
6  {
7
8      var $_mime = "application/rss+xml";
9
10
11     function render()
12     {
13         $now    =& JFactory::getDate();
14         $data    =& $this->_doc;
15
16         $uri =& JFactory::getURI();
17         $url = $uri->toString(array('scheme', 'user', 'pass', 'host', 'port'));
18
19         $feed = "<rss version=\"2.0\">\n";
20         $feed.= "    <channel>\n";
21         $feed.= "        <title>".$data->title."</title>\n";
22         $feed.= "        <description>".$data->description."</description>\n";
23         $feed.= "        <link>".$url.$data->link."</link>\n";
24         $feed.= "        <lastBuildDate>".htmlspecialchars($now->toRFC822(), ENT_COMPAT, 'UTF-8').
25             "</lastBuildDate>\n";
26         $feed.= "        <generator>".$data->getGenerator()."</generator>\n";
27
28         if ($data->image!=null)
29         {
30             $feed.= "            <image>\n";
31             $feed.= "                <url>".$data->image->url."</url>\n";
32             $feed.= "                <title>".htmlspecialchars($data->image->title, ENT_COMPAT,
33                 'UTF-8')."</title>\n";
34             $feed.= "                <link>".$data->image->link."</link>\n";
35             if ($data->image->width != "") {
36                 $feed.= "                <width>".$data->image->width."</width>\n";
37             }
38             if ($data->image->height!="") {
39                 $feed.= "                <height>".$data->image->height."</height>\n";
40             }
41             if ($data->image->description!="") {
42                 $feed.= "                <description><![CDATA[\"".$data->image->description.\"]]>
43                     </description>\n";
44             }
45             $feed.= "            </image>\n";
46         }
47     }
48 }
```

```

47         if ($data->language!="") {
48             $feed.= "                <language>". $data->language. "</language>\n";
49         }
50         if ($data->copyright!="") {
51             $feed.= "                <copyright>". htmlspecialchars($data->copyright, ENT_COMPAT,
52                                     'UTF-8'). "</copyright>\n";
53         }
54         if ($data->editor!="") {
55             $feed.= "                <managingEditor>". htmlspecialchars($data->editor, ENT_COMPAT,
56                                     'UTF-8'). "</managingEditor>\n";
57         }
58         if ($data->webmaster!="") {
59             $feed.= "                <webMaster>". htmlspecialchars($data->webmaster, ENT_COMPAT,
60                                     'UTF-8'). "</webMaster>\n";
61         }
62         if ($data->pubDate!="") {
63             $pubDate =& JFactory::getDate($data->pubDate);
64             $feed.= "                <pubDate>". htmlspecialchars($pubDate->toRFC822(), ENT_COMPAT,
65                                     'UTF-8'). "</pubDate>\n";
66         }
67         if ($data->category!="") {
68             $feed.= "                <category>". htmlspecialchars($data->category, ENT_COMPAT,
69                                     'UTF-8'). "</category>\n";
70         }
71         if ($data->docs!="") {
72             $feed.= "                <docs>". htmlspecialchars($data->docs, ENT_COMPAT,
73                                     'UTF-8'). "</docs>\n";
74         }
75         if ($data->ttl!="") {
76             $feed.= "                <ttl>". htmlspecialchars($data->ttl, ENT_COMPAT,
77                                     'UTF-8'). "</ttl>\n";
78         }
79         if ($data->rating!="") {
80             $feed.= "                <rating>". htmlspecialchars($data->rating, ENT_COMPAT,
81                                     'UTF-8'). "</rating>\n";
82         }
83         if ($data->skipHours!="") {
84             $feed.= "                <skipHours>". htmlspecialchars($data->skipHours, ENT_COMPAT,
85                                     'UTF-8'). "</skipHours>\n";
86         }
87         if ($data->skipDays!="") {
88             $feed.= "                <skipDays>". htmlspecialchars($data->skipDays, ENT_COMPAT,
89                                     'UTF-8'). "</skipDays>\n";
90         }
91     }
92     for ($i=0; $i<count($data->items); $i++)
93     {
94         if ((strpos($data->items[$i]->link, 'http://') === false) and (strpos($data->items[$i]->link,
95                                     'https://') === false)) {
96             $data->items[$i]->link = $url.$data->items[$i]->link;
97         }
98         $feed.= "                <item>\n";
99         $feed.= "                <title>". htmlspecialchars(strip_tags($data->items[$i]->title),
100                                     ENT_COMPAT, 'UTF-8'). "</title>\n";
101         $feed.= "                <link>". $data->items[$i]->link. "</link>\n";
102         $feed.= "                <description><![CDATA[".$this->_relToAbs($data->items[$i]->description.
103                                     ".")></description>\n";
104
105         if ($data->items[$i]->author!="") {
106             $feed.= "                <author>". htmlspecialchars($data->items[$i]->author, ENT_COMPAT,
107                                     'UTF-8'). "</author>\n";
108         }
109
110         if ($data->items[$i]->category!="") {
111             $feed.= "                <category>". htmlspecialchars($data->items[$i]->category,
112                                     ENT_COMPAT, 'UTF-8'). "</category>\n";
113         }
114         if ($data->items[$i]->comments!="") {
115             $feed.= "                <comments>". htmlspecialchars($data->items[$i]->comments,
116                                     ENT_COMPAT, 'UTF-8'). "</comments>\n";
117         }
118         if ($data->items[$i]->date!="") {
119             $itemDate =& JFactory::getDate($data->items[$i]->date);
120             $feed.= "                <pubDate>". htmlspecialchars($itemDate->toRFC822(),

```

```

121                                     ENT_COMPAT, 'UTF-8')."</pubDate>\n";
122     }
123     if ($data->items[$i]->guid!="") {
124         $feed.= "                <guid>".htmlspecialchars($data->items[$i]->guid,
125                                     ENT_COMPAT, 'UTF-8')."</guid>\n";
126     }
127     if ($data->items[$i]->enclosure != NULL)
128     {
129         $feed.= "                                <enclosure url=\"";
130         $feed.= $data->items[$i]->enclosure->url;
131         $feed.= "\" length=\"";
132         $feed.= $data->items[$i]->enclosure->length;
133         $feed.= "\" type=\"";
134         $feed.= $data->items[$i]->enclosure->type;
135         $feed.= "\"/>\n";
136     }
137
138     $feed.= "                </item>\n";
139 }
140 $feed.= "        </channel>\n";
141 $feed.= "</rss>\n";
142 return $feed;
143 }
144
145 function _relToAbs($text)
146 {
147     $base = JURI::base();
148     $text = preg_replace("/(href|src)=\"(?!http|ftp|https)([^\"]*)\"/", "$1=\"$base\$2\"", $text);
149
150     return $text;
151 }
152 }
153 ?>

```

A.2 Folha de Estilo (CSS) do Layout do website do DCC

```

1  @import url('clearing.css');
2
3  html, body {
4      margin: 0;
5      padding: 0;
6      background: #fff;
7      font-family: Tahoma, Arial, Helvetica, sans-serif;
8      font-size: 11px;
9      line-height: 1.5;
10     color: #828282;
11 }
12 table, ul, a, li, img, p, form, hr, dl, dt, dd, div, h1, h2, h3, h4, h5, h6, label {margin:0; padding:0}
13 h1, h2, h3, h4, h5, h6 { font-size: 14px; color: #595959; font-weight: bold; text-indent: 0px; font-family:"Trebuchet MS"; }
14 img { border: none;}
15
16 #top_box { width: 100%; height: 57px; background: url(..images/top_bg.gif) top left repeat-x;}
17 #top { width: 982px;}
18 #top_l { float: left; width: 800px; height: 57px; }
19 #menu { height: 27px; padding: 30px 6px 0; float: left;}
20 #top_r { float: right; width: 182px; height: 57px; }
21 #logo { height: 49px; width: 156px; background: url(..images/logo.png) top right no-repeat;}
22 #header_box { width: 100%; height: 211px; background:url(..images/header_bg.jpg) top left repeat-x;}
23 #header { width: 982px; height: 211px; background:url(..images/header.jpg) top left no-repeat; }
24 #head_mod_out { padding: 71px 0 0 10px; height: 140px; overflow: hidden; }
25 #head_mod { float: left; width: 250px; height: 140px; background: url(..images/head_mod.png) top left no-repeat;}
26 #head_mod_in { padding: 25px 40px 20px 12px; overflow: hidden;}
27 #head_bot_box {width: 100%; height: 38px; background:url(..images/header_b.gif) top left repeat-x;}
28 #head_bot {width: 982px;}
29 #search_box { float: left; width: 290px; height: 38px; background:url(..images/header_bot_stripe.png) center right no-repeat;}
30 #search { float: right; padding: 11px 35px 0 0; color: #595959; font-family:"Trebuchet MS", Arial, sans-serif; font-size: 11px;}
31 #date_box { float: right; width: 250px; height: 38px; background:url(..images/header_bot_stripe.png) center left no-repeat;}
32 #date { padding: 12px 15px 0 15px; color: #595959; text-align: right;}

```

```

33
34 #container { width: 982px; text-align:left; }
35 #left_col { float: left; width: 251px;}
36 #left_col-noleft {display: none;}
37 #left_col_in { width: 240px; padding-left: 11px;}
38 #left_mod { width: 240px; overflow: hidden;}
39 #right_col { float: right; width: 731px;}
40 #right_col-noleft { width: 982px;}
41 #content { float: left; width: 480px;}
42 #content-noleft { float: left; width: 731px;}
43 #content-noright { width: 731px;}
44 #content-noleft-noright { float: left; width: 782px;}
45 #content_in { width: 460px; padding: 0 10px;}
46 #content_in-noleft, #content_in-noright { width: 711px; padding: 0 10px;}
47 #content_in-noleft-noright { width: 962px; padding: 0 10px;}
48 #pathway_box { width: 460px; height: 30px;background: url(../images/path_bg.gif) left bottom repeat-x;}
49 #pathway_box-noleft, #pathway_box-noright { width: 711px; height: 30px;background: url(../images/path_bg.gif) left bot
50 #pathway_box-noleft-noright { width: 962px; height: 30px;background: url(../images/path_bg.gif) left bottom repeat-x;}
51 #path { padding: 8px 11px 0; text-align: left; color: #4a4a4a; }
52 #mainbody_in { width: 460px; overflow: hidden;}
53 #mainbody_in-noleft, #mainbody_in-noright { width: 711px; padding: 10px 0; overflow: hidden;}
54 #mainbody_in-noleft-noright { width: 962px; padding: 10px 0; overflow: hidden;}
55 #mainbody {padding: 10px 0;min-height: 300px;}
56 #banner {width: 460px; overflow: hidden;}
57 #banner-noleft, #banner-noright {width: 711px; overflow: hidden;}
58 #banner-noleft-noright {width: 962px; overflow: hidden;}
59 #right_box { float: right; width: 251px;}
60 #right_mod { width: 240px; padding-left: 11px; overflow: hidden;}
61 #footer { width: 100%; height: 44px; margin-top: 15px; background: url(../images/foot.gif) top left repeat-x;}
62 #footer_in { width: 982px; color: #FFFFFF;}
63 #copyright_box { float: right; width: 491px;}
64 #copyright { padding-top: 15px; text-align: right;}
65 #license_box { float: left; width: 491px;}
66 #license { padding-top: 5px; text-align: left; float: left;}
67
68
69 #container a:link, #container a:visited {color:#828282; text-decoration: none;}
70 #container a:hover {color:#828282; text-decoration: underline;}
71 #container ul{
72     margin: 0;
73     padding: 0;
74     list-style: none;
75 }
76 #container li{
77     line-height: 1.5;
78     padding-left:15px;
79     margin-left:3px;
80     background: url(../images/submenu.png) no-repeat;
81     background-position:0px 5px;
82     text-align: left;
83 }
84 #container ol li{ background: none; }
85 #footer a:link, #footer a:visited {color:#828282; text-decoration: none;}
86 #footer a:hover {color:#828282; text-decoration: underline;}
87 #path a:link, #path a:visited {color:#4a4a4a; text-decoration: none;}
88 #path a:hover {color:#4a4a4a; text-decoration: underline;}
89 #path img { margin: 0 5px;}
90
91 /*****
92 #left_mod ul#mainlevel li,#left_mod ul.menu li {
93 width: 100%;
94 list-style:none;
95 padding:0 0 0 0;
96 margin:0; border: none;
97 }
98 #left_mod a.mainlevel:link, #left_mod a.mainlevel:visited,
99 #left_mod ul.menu li a:link, #left_mod ul.menu li a:visited {
100     overflow: hidden;
101     display: block;
102     font-family: "Trebuchet MS", Tahoma, Helvetica, sans-serif;
103     font-weight: bold;
104     text-align: left;
105     color: #707070;
106     font-size: 11px;

```

```

107         line-height:20px;
108         background: url(../images/submenu.png) left no-repeat;
109         background-position: 0 7px;
110         margin:0px;
111         padding-left:20px;
112         text-decoration: none;
113         text-transform: none;
114     }
115     #left_mod a.mainlevel:hover,
116     #left_mod ul.menu li a:hover { text-decoration: underline;}
117     #left_mod a.mainlevel#active_menu,
118     #left_mod ul.menu li a#active_menu , #left_mod ul.menu li#current a{ text-decoration: underline;}
119     #left_mod a.sublevel:link, #left_mod a.sublevel:visited {
120         overflow: hidden;
121         display: block;
122         font-family: "Trebuchet MS", Tahoma, Helvetica, sans-serif;
123         font-weight: bold;
124         text-align: left;
125         color: #707070;
126         font-size: 11px;
127         line-height:20px;
128         background: url(../images/sublevel.png) left no-repeat;
129         background-position: 0 7px;
130         margin:0px;
131         padding-left:25px;
132         text-decoration: none;
133     }
134     #left_mod ul.menu li#current li a:link, #left_mod ul.menu li#current li a:visited,
135     #left_mod ul.menu li ul li a:link, #left_mod ul.menu li ul li a:visited {
136         overflow: hidden;
137         display: block;
138         font-family: "Trebuchet MS", Tahoma, Helvetica, sans-serif;
139         font-weight: bold;
140         text-align: left;
141         color: #707070;
142         font-size: 11px;
143         line-height:20px;
144         background: none;
145         background-position: 0 7px;
146         margin:0px;
147         padding-left:25px;
148         text-decoration: none;
149     }
150     #left_mod a.sublevel:hover, #left_mod ul.menu li ul li a:hover {text-decoration: underline;}
151     #left_mod a.sublevel#active_menu{ text-decoration: underline;}
152
153     #left_mod ul, #sidebar-2 ul
154     {
155         list-style-type: none;
156         margin: 0px 0px 0px 0px;
157         padding: 0px 0px 0px 0px;
158     }
159     #left_mod ul li
160     {
161         list-style: none;
162         border-top: solid 1px #d7e7ec;
163         margin: 0px 0px 0px 0;
164         padding: 0px 0px 0px 9px; line-height: 14px;
165     }
166     #sidebar-2 ul li
167     {
168         list-style: none;
169         border-top: solid 0px #d7e7ec;
170         margin: 0px 0px 0px 0;
171         padding: 0px 0px 0px 9px; line-height: 14px;
172     }
173     #left_mod ul li a:link,#sidebar-2 ul li a:link,#left_mod ul li a:visited, #sidebar-2 ul li a:visited
174     {
175         display: block;
176         padding: 4px 2px 4px 10px;
177         text-decoration: none;
178         background: url(../images/default/bg_alink.png) left no-repeat;
179         color: #006383;
180         font-family: Arial, Helvetica, sans-serif;
181         font-weight: normal;
182         vertical-align: top;

```



```

181         line-height: 14px;
182     }
183     #left_mod ul li a:hover, #sidebar-2 ul li a:hover
184     { text-decoration: underline;}
185
186     #left_mod ul li ul, #sidebar-2 ul li ul
187     {
188         list-style-type: none;
189         margin: 0px 0px 0px 0px;
190         padding: 0px 0px 0px 0px;
191     }
192     #left_mod ul li ul li, #sidebar-2 ul li ul li
193     {
194         list-style: none;
195         border-bottom: solid 0px #d7e7ec;
196         margin: 0px 0px 0px 0px;
197         padding: 0px 0px 0px 1px; line-height: 1.3em;
198     }
199     #left_mod ul li ul li a:link, #sidebar-2 ul li ul li a:link, #left_mod ul li ul li a:visited, #sidebar-2 ul li ul li a:visited
200     {
201         display: block;
202         padding: 2px 2px 2px 10px;
203         text-decoration: none;
204         background: none;
205         color: #006383;
206         font-size: 0.9em;
207         line-height: 1.3em;
208         font-family: Arial, Helvetica, sans-serif;
209         font-weight: bold;
210         vertical-align: top;
211     }
212     #left_mod ul li ul li a:hover, #sidebar-2 ul li ul li a:hover
213     { text-decoration: underline;}
214     /*****
215     #left_mod div.module a:link, #left_mod div.module a:visited {color:#707070; text-decoration: none;}
216     #left_mod div.module a:hover {color:#707070; text-decoration: underline;}
217     #left_mod div.module h3{
218         background: none;
219         font-family: "Trebuchet MS", Arial, sans-serif;
220         color:#666666;
221         text-indent: 0px;
222         font-size: 15px;
223         height: 20px;
224         line-height: 20px;
225         font-weight: bold;
226         margin-bottom: 7px;
227         text-align: left;
228     }
229     #left_mod div.module{
230         width: 240px;
231         background: url(..images/mod_in.gif) left top repeat-y;
232         margin: 10px 0;
233     }
234     #left_mod div.module div{
235         width: 240px;
236         background: url(..images/mod_t.gif) top left no-repeat;
237     }
238     #left_mod div.module div div{
239         width: 240px;
240         background: url(..images/mod_b.gif) bottom left no-repeat;
241     }
242     #left_mod div.module div div div{
243         width: 216px;
244         padding: 20px 22px 12px 12px;
245         color:#707070;
246         font-size: 10px;
247         text-align: left;
248     }
249     #left_mod div.module div div div div{
250         padding: 0px;
251         margin: 0px;
252         background: none;
253     }
254     #left_mod div.module div div div table div, #right_mod div.module div div div table div { min-height: 17px;}
255     #left_mod div.module div div div div syndicate div { min-height: 17px;}

```

```

255
256 /*****
257 #left_mod div.module_menu a:link, #left_mod div.module_menu a:visited {color:#707070; text-decoration: none;}
258 #left_mod div.module_menu a:hover {color:#707070; text-decoration: underline;}
259 #left_mod div.module_menu h3{
260     background: none;
261     font-family: "Trebuchet MS", Arial, sans-serif;
262     color:#666666;
263     text-indent: 0px;
264     font-size: 15px;
265     height: 20px;
266     line-height: 20px;
267     font-weight: bold;
268     margin-bottom: 7px;
269     text-align: left;
270 }
271 #left_mod div.module_menu{
272     width: 240px;
273     background: url(..images/mod_in.gif) left top repeat-y;
274     margin: 10px 0;
275 }
276 #left_mod div.module_menu div{
277     width: 240px;
278     background: url(..images/mod_t.gif) top left no-repeat;
279 }
280 #left_mod div.module_menu div div{
281     width: 240px;
282     background: url(..images/mod_b.gif) bottom left no-repeat;
283 }
284 #left_mod div.module_menu div div div{
285     width: 216px;
286     padding: 20px 22px 12px 12px;
287     color:#707070;
288     font-size: 10px;
289     text-align: left;
290 }
291 #left_mod div.module_menu div div div div{
292     padding: 0px;
293     margin: 0px;
294     background: none;
295 }
296 /*****
297 #right_mod div.module a:link,#right_mod div.module a:visited {color:#707070; text-decoration: none;}
298 #right_mod div.module a:hover {color:#707070; text-decoration: underline;}
299 #right_mod div.module h3{
300     background: none;
301     font-family: "Trebuchet MS", Arial, sans-serif;
302     color:#666666;
303     text-indent: 0px;
304     font-size: 15px;
305     height: 20px;
306     line-height: 20px;
307     font-weight: bold;
308     margin-bottom: 7px;
309     text-align: left;
310 }
311 #right_mod div.module{
312     width: 240px;
313     background: url(..images/mod_in.gif) left top repeat-y;
314     margin: 10px 0;
315 }
316 #right_mod div.module div{
317     width: 240px;
318     background: url(..images/mod_t.gif) top left no-repeat;
319 }
320 #right_mod div.module div div{
321     width: 240px;
322     background: url(..images/mod_b.gif) bottom left no-repeat;
323 }
324 #right_mod div.module div div div{
325     width: 216px;
326     min-height: 100px;
327     padding: 20px 22px 12px 12px;
328     color:#707070;

```

```

329         font-size: 10px;
330         text-align: left;
331     }
332     #right_mod div.module div div div div{
333         padding: 0px;
334         margin: 0px;
335         background: none;
336     }
337     /*****
338     #right_mod div.module-adv h3{
339         background: none;
340         font-family: "Trebuchet MS", Arial, sans-serif;
341         color:#666666;
342         text-indent: 0px;
343         font-size: 15px;
344         height: 20px;
345         line-height: 20px;
346         font-weight: bold;
347         margin-bottom: 7px;
348         text-align: left;
349     }
350     #right_mod div.module-adv{
351         width: 240px;
352         background: none;
353         margin: 10px 0;
354     }
355     #right_mod div.module-adv div{
356         width: 240px;
357         background: none;
358     }
359     #right_mod div.module-adv div div{
360         width: 240px;
361         background: none;
362     }
363     #right_mod div.module-adv div div div{
364         width: 240px;
365         padding: 0;
366         color:#707070;
367         font-size: 10px;
368         text-align: left;
369     }
370     #right_mod div.module-adv div div div div{
371         padding: 0px;
372         margin: 0px;
373         background: none;
374     }
375     /*****
376     #head_mod div.moduletable {
377         overflow:hidden;
378         height: 105px; width: 198px;
379         padding:0; margin: 0px;
380         background:none;
381         font-family: Tahoma, Helvetica, sans-serif;
382         color: #4e4e4e;
383         font-size: 11px;
384         text-align: left;
385     }
386     #head_mod div.moduletable h3 {
387         font-family: "Trebuchet MS", Tahoma, Helvetica, sans-serif;
388         line-height: 16px;
389         margin:0;
390         color: #595959;
391         font-size: 14px;
392         font-weight: bold;
393         background:none;
394         text-align:left;
395     }
396     #head_mod a:link, #head_mod a:visited {color:#828282; text-decoration: none;}
397     #head_mod a:hover {color:#828282; text-decoration: underline;}
398     /*****
399     fieldset { border:none;}
400     #search .inputbox {
401         color: #595959;
402         border: none;

```

```

403         background: url(..../images/search_input.gif) top right no-repeat;
404         padding: 2px 2px 0;
405         margin: 0;
406         margin-left: 15px;
407         width: 146px;
408         height: 14px;
409         font-size: 9px;
410         line-height: 1.0;
411         text-align: left;
412         vertical-align: top;
413     }
414     #search .button{
415         line-height: 1.0;
416         background: url(..../images/search_but.gif) top left no-repeat;
417         height: 16px;
418         width: 13px;
419         border: none;
420         padding: 0; margin-left: 6px;
421     }
422     .button {
423         color: #fff;
424         background: url(..../images/button_bg.gif) center left repeat-x;
425         line-height: 12px;
426         font-size: 10px;
427         font-weight: bold;
428         border: none;
429         text-align: center;
430         margin: 0; padding: 2px 5px;
431     }
432     .inputbox {
433         background: #FFFFFF;
434         font-size: 9px;
435         color: #595959;
436         border: solid 1px #a7a7a7;
437         margin: 2px 0;
438         padding: 2px;
439     }
440     .checkbox {
441         background: #FFFFFF;
442         color: #595959;
443         border: solid 1px #a7a7a7;
444         margin: 2px 0;
445     }
446     .contentpane .inputbox { display: block; }
447     .contentheading, .componentheader, .componentheading {
448         width: 98%;
449         font-family: "Trebuchet MS", Tahoma, Helvetica, sans-serif;
450         line-height: 20px;
451         margin-bottom: 5px;
452         color: #0085d1;
453         font-size: 14px;
454         font-weight: bold;
455         background: none;
456         text-indent: 0px;
457         text-align: left;
458     }
459     .contentheading td {
460         width: 98%;
461         font-family: "Trebuchet MS", Tahoma, Helvetica, sans-serif;
462         line-height: 20px;
463         margin-bottom: 5px;
464         color: #0085d1;
465         font-size: 14px;
466         font-weight: bold;
467         background: none;
468         text-indent: 0px;
469         text-align: left;
470     }
471     table.contentpaneopen {
472         padding: 0px; width: 98%;
473         border-collapse: collapse;
474         border-spacing: 0px;
475         margin: 0px;
476     }

```

```

477 table.contentpaneopen {
478     width:98%;
479 }
480 .contentpane {
481     width:98%;
482 }
483 .contentpaneopen td img {
484     margin: 2px 3px;
485 }
486 .blog_more {
487     padding-left: 15px;
488 }
489 table .poll {
490     width:90%;
491     text-align:left;
492 }
493 table .poll-pink {
494     width:90%;
495     text-align:left
496 }
497 table .poll-green {
498     width:90%;
499     text-align:left
500 }
501 .poll thead {
502     color:#595959;
503 }
504 .pollstableborder{
505     text-align:left;
506 }
507 table.contenttoc {
508     padding: 10px;
509 }
510 table.contenttoc th {
511     text-align: center;
512     padding: 2px 10px;
513     background:#b4b4b4;
514     color: #4a4a4a;
515     text-align:left;
516 }
517 table.contenttoc td {
518     text-align: left;
519     padding: 2px 10px;
520     background:#e4e4e4;
521     color: #707070;
522     font-size: 10px;
523 }
524 td.buttonheading {
525     margin: 0px;
526     padding: 0 2px;
527 }
528 p { padding-bottom: 10px;}
529 #mainbody .pagenav {
530     text-align: center;
531     font-size: 10px;
532     font-weight: normal;
533     color: #707070;
534     padding: 0px 5px;
535     line-height: 20px;
536     margin: 1px;
537 }
538 #mainbody .pagenavbar{
539     margin-right: 10px;
540     float: right;
541     font-size: 10px;
542     font-weight: normal;
543 }
544 #mainbody .pagenavbar a:link, #mainbody .pagenavbar a:visited, #mainbody .pagenavbar a:hover{
545     margin-right: 10px;
546     font-size: 10px;
547     font-weight: normal;
548     color: #707070;
549 }
550 #mainbody .pagenav_next, #mainbody .pagenav_prev {

```

```

551         background:none;
552         color: #707070;
553         padding: 5px 2px;
554         font-size: 10px;
555         font-weight: normal;
556         margin: 0px;
557         text-align: center;
558     }
559     #mainbody .pagenav_next a:link, #mainbody .pagenav_prev a:link,
560     #mainbody .pagenav_next a:visited, #mainbody .pagenav_prev a:visited,
561     #mainbody .pagenav_next a:hover, #mainbody .pagenav_prev a:hover {
562         color: #707070;
563         padding: 5px 5px;
564         text-decoration: none;
565         font-size: 10px;
566         font-weight: normal;
567         text-align: center;
568     }
569     #mainbody .back_button{
570         font-size: 10px;
571         line-height: 20px;
572         font-weight: normal;
573         float: left;
574         height: 20px;
575         background: none;
576         padding: 2px 7px;
577         margin:0;
578         color: #707070;
579     }
580     #mainbody .back_button a:link, #mainbody .back_button a:visited, #mainbody .back_button a:hover {
581         color: #707070;
582         font-size: 10px;
583         text-decoration: none;
584     }
585     .readon {
586         float: right;
587         color: #707070;
588         font-size: 10px;
589         font-weight: normal;
590     }
591     div.pagenavcounter {
592         padding-bottom: 10px;
593         text-decoration: underline;
594     }
595     .small {
596         color: #828282;
597         font-size: 11px;
598         line-height:1.0;
599         padding-top: 10px;
600     }
601     .smalldark {
602         color: #828282;
603         font-size: 11px;
604         line-height:1.0;
605         text-align: left;
606     }
607     .createdate {
608         color: #828282;/* kolor daty */
609         font-size: 11px;
610         padding-bottom: 20px;
611         text-align: left;
612         line-height:1.0;
613     }
614     .modifydate {
615         color: #828282;/* kolor daty modyfikacji */
616         height: 20px;
617         font-size: 11px;
618         padding-top: 10px;
619         padding-bottom: 5px;
620     }
621     div.mosimage {
622         border: none;
623     }
624     .mosimage{

```

```

625     border: none;
626     margin: 5px;
627     padding: 5px;
628 }
629 .mosimage_caption {
630     margin-top: 2px;
631     background: none;
632     padding: 10px;
633     color: #828282;
634     border-top: none;
635 }
636 .sectiontableheader {font-weight: bold;}
637 /* kolor rzędów w spisach i tabelach parzystych*/
638 .sectiontableentry1 td, .sectiontableentry1-pink, .sectiontableentry1-green {
639     text-align: left;
640     padding: 4px 0;
641 }
642 /* kolor rzędów w spisach i tabelach nieparzystych */
643 .sectiontableentry2 td, .sectiontableentry2-pink, .sectiontableentry2-green {
644     text-align: left;
645     padding: 4px 0;
646     background: #e4e4e4;
647 }
648 .related { font-weight: bold; font-size: 13px;}
649 fieldset a:link, fieldset a:visited, fieldset a:hover { line-height: 25px;}
650
651 /*formatowanie strony do wydruku*/
652 body.contentpane {background:#fff; color:#000; padding: 15px; font-size: 11px; text-align: left;}
653 body.contentpane h1, body.contentpane h2, body.contentpane h3 {background:#fff; color:#000;font-size: 12px;}
654 body.contentpane a, body.contentpane span.small, body.contentpane table.contentpaneopen td, body.contentpane table td
655 {background:#fff; color:#000; font-size: 11px;}
656 body.contentpane .componentheading, body.contentpane .contentheading {background:#000; color:#fff; font-weight: bold; }
657 body.p {margin-bottom: 5px;}
658 body.contentpane .button{color:black;border:1px solid black; background: #fff;}
659 body.contentpane .inputbox {background: #FFFFFF; color: #000; border: solid 1px #000;}
660 body.contentpane td{padding:3px; background:#fff; color:#000;font-size: 11px;}

```

A.3 Folha de Estilo (CSS) do Layout do Blog

```

1
2     margin: 0;
3     padding: 0;
4 }
5
6 html, body{
7     height:100%;
8 }
9
10 html{
11     font-size: 100%;
12 }
13
14 body{
15     font-family: "Lucida Grande", "Lucida Sans Unicode", Arial, Helvetica, Sans, FreeSans, Jamrul, Garuda, Kalimati;
16     font-size: 0.8em;
17     font-style: normal;
18     color: #000;
19     background: #fff url(images/main-bg.gif) repeat-x left 200px;
20     min-height:100%;
21     height:auto !important; /* ie6 ignores it */
22     height:100%; /* fix */
23     position: relative;
24 }
25
26
27 /* general elements */
28 a{
29     color: #2d83d5;
30     text-decoration: underline;
31 }
32

```

```

33  a:hover{
34      color: #fff;
35      background: #2d83d5;
36      text-decoration: none;
37  }
38
39  img{
40      border: 0;
41  }
42
43  table{
44      margin: .5em 0 1em;
45      border-bottom: 3px solid #d9d9d9;
46      border-left: 1px solid #d9d9d9;
47  }
48
49  table td, table th{
50      text-align: left;
51      padding: .2em .4em;
52      border-top: 1px solid #d9d9d9;
53      border-right: 1px solid #d9d9d9;
54  }
55
56  table th{
57      background: #ebebeb;
58      border-bottom: 2px solid #d9d9d9;
59  }
60
61  table tr.odd{
62      background: #ebebeb;
63  }
64
65  table tr:hover{
66      background: #d9d9d9;
67  }
68
69  p{
70      line-height: 140%;
71      margin: .4em 0 1em;
72  }
73
74  form p{
75      margin: 0;
76  }
77
78  input, textarea, select{
79      font-family: "Lucida Grande", "Lucida Sans Unicode", Arial, Helvetica, Sans, FreeSans, Jamrul, Garuda, Kalimati;
80      font-size: 100%;
81      margin: .2em 0;
82      background: transparent url(images/input-bg.png) no-repeat left top;
83  }
84
85  input, textarea{
86      padding: .2em .4em;
87  }
88
89  input.radio, input.checkbox{
90      background: none;
91      padding: 2px;
92  }
93
94
95  form label{
96      font-weight: bold;
97  }
98
99  fieldset{
100     padding: .8em;
101     border: 1px solid #E7E8E6;
102 }
103
104 legend{
105     padding: 2px 15px 10px 15px;
106     color: #8b8b89;

```



```

107     text-transform: uppercase;
108     font-style: italic;
109     font-size: 115%;
110 }
111
112 small{ font-size: 86%; }
113
114 pre{
115     overflow: auto;
116     background: #eee;
117     padding: .4em;
118 }
119
120 /* lists */
121
122 ul, ol{
123     margin: .4em 0 1em;
124     line-height: 150%;
125 }
126
127
128 ul li, ol li{
129     list-style-position:outside;
130     margin-left: 2em;
131 }
132
133 dl{
134     padding: .3em 0 .8em;
135 }
136
137 dt {
138     float: left;
139     clear: left;
140     width: 9em;
141     text-align: right;
142     font-weight: bold;
143     text-decoration: underline;
144 }
145
146 dd {
147     margin: 0 0 0 10em;
148     padding: 0 0 0.5em 0;
149 }
150
151 blockquote{
152     background: transparent url(images/blockquote.png) no-repeat left top;
153     margin: 10px 65px 10px 10px;
154     padding: 20px 20px 10px 10px;
155     font-style: italic;
156     color: #939494;
157 }
158
159 blockquote p{
160     padding: 8px;
161     margin: 2px;
162     border: 1px dotted #939494;
163 }
164
165 /* headings */
166
167 h1{
168     font-size: 250%;
169     margin: .6em 0 .3em;
170     font-weight: bold;
171     letter-spacing:-0.06em;
172 }
173
174 h2{
175     font-family: "Palatino Linotype", Georgia, "Tahoma", "Century Schoolbook L", Arial, Helvetica;
176     font-size: 200%;
177     font-weight: normal;
178     margin: .6em 0 .3em;
179 }
180

```

```

181 h3{
182     font-family: "Lucida Grande", "Lucida Sans Unicode", Arial, Helvetica, Sans, FreeSans, Jamrul, Garuda, Kalimati;
183     font-size: 150%;
184     font-weight: bold;
185     margin: .2em 0 .2em;
186     letter-spacing:-0.06em;
187 }
188
189 h4{
190     font-family: "Lucida Grande", "Lucida Sans Unicode", Arial, Helvetica, Sans, FreeSans, Jamrul, Garuda, Kalimati;
191     font-size: 125%;
192     font-weight: bold;
193     margin: .4em 0 .2em;
194 }
195
196 h5{
197     font-family: "Lucida Grande", "Lucida Sans Unicode", Arial, Helvetica, Sans, FreeSans, Jamrul, Garuda, Kalimati;
198     font-size: 115%;
199     font-weight: normal;
200     margin: .4em 0 .2em;
201     text-transform: uppercase;
202     font-style: italic;
203 }
204
205 h6{
206     font-family: "Lucida Grande", "Lucida Sans Unicode", Arial, Helvetica, Sans, FreeSans, Jamrul, Garuda, Kalimati;
207     font-size: 110%;
208     font-weight: bold;
209     margin: .4em 0 .2em;
210 }
211
212
213 /*** BASIC LAYOUT ***/
214
215 /* used only for the header background images (center & repeat) */
216 #page-wrap1{ background: transparent url(images/header-bg.jpg) repeat-x left top; }
217 #page-wrap2{
218     background: transparent url(images/header-center.jpg) no-repeat center top;
219     height: 200px;
220     width: 100%;
221     display:block;
222 }
223
224 /* actual layout width: default is 960px (for 1024x768+); 100% = fluid width */
225 #page{
226     width: 960px;
227     min-width: 780px;
228     margin:0 auto;
229 }
230
231 /* content area (sidebars + main) */
232 #main-wrap {
233     position:relative;           /* IE7 overflow bugfix */
234     clear:both;
235     float:left;
236     width:100%;
237 }
238
239 #mid-wrap, #side-wrap {
240     float:left;
241     width:100%;
242     position:relative;
243 }
244
245 #mid, #sidebar, #sidebar2 {
246     float:left;
247     position:relative;
248     padding:0;
249     overflow: visible;
250 }
251
252
253 /* start ie 6/7 position relative + z-index bug-fix
254

```

```

255 #side-wrap, #mid-wrap{
256     float: none;
257     position: static;
258 }
259 #sidebar, #sidebar2{
260     position: static;
261     float: right;
262 }
263 #mid{
264     position: static;
265     float: left;
266 }
267
268 end ie 6/7 position relative + z-index bug-fix */
269
270
271 /* no sidebar? */
272
273 #page #main-wrap #mid-wrap { right:100%; }
274 #page #main-wrap #side-wrap { right:0; }
275
276 #page #main-wrap #mid {
277     width:100%;                /* main column width */
278     left:100%;                /* same */
279 }
280
281
282 /* if sidebar exists... */
283
284 #page.with-sidebar #main-wrap #mid-wrap { right:0%; /* width of the 2nd column, 0 because it doesnt exist */ }
285 #page.with-sidebar #main-wrap #side-wrap { right:66%; /* width of the main column */ }
286
287 #page.with-sidebar #main-wrap #mid {
288     width:66%;                /* width of main column */
289     left:66%;                /* same */
290 }
291
292 #page.with-sidebar #main-wrap #sidebar {
293     width:34%;                /* width of the sidebar */
294     left:66%;
295 }
296
297
298 /* if 2nd sidebar exists... */
299
300 #page.with-sidebar2 #main-wrap #mid-wrap { right:25%; }
301 #page.with-sidebar2 #main-wrap #side-wrap { right:25%; }
302
303 #page.with-sidebar2 #main-wrap #mid {
304     width:50%;                /* width of main column */
305     left:50%;
306 }
307
308 #page.with-sidebar2 #main-wrap #sidebar {
309     width:30%;                /* width of the sidebar (narrower than on 2 column pages) */
310     left:50%;
311 }
312
313 #page.with-sidebar2 #main-wrap #sidebar2 {
314     width:20%;                /* width of 2nd sidebar */
315     left:50%;
316 }
317
318
319 #sidebar2-wrap{
320     margin-top: 230px;
321     padding-left: 10px;
322 }
323
324
325 #mid-content{
326     margin: 40px 0 10px;
327 }
328

```

```

329 #sidebar-wrap1{
330     background: transparent url(images/side-start.png) no-repeat left top;
331     padding-left: 28px;
332     margin-top: 36px;
333     height: 100%;
334 }
335
336 #sidebar-wrap2{
337     background: transparent url(images/side-bg.png) no-repeat right top;
338     padding-right: 30px;
339     padding-top: 40px;
340     height: 100%;
341     min-height: 400px;
342 }
343
344
345 /* clear sidebar/main sections, assuming sidebar is on the right */
346 div.clearcontent{
347     clear: left;
348 }
349
350
351
352 /*** HEADER AREA ***/
353
354 /* logo position */
355 a#logo{
356     top: 80px;
357     position: absolute;
358 }
359
360 a#logo:hover{
361     background: none;
362 }
363
364 #header{
365     height: 200px;
366     z-index: 2; /* fix for stupid ie6/7 bug */
367     position: relative;
368 }
369
370 #header h1{
371     font-family: "Trajan Pro", "Lucida Sans Unicode", Arial, Helvetica, Sans, FreeSans, Jamrul, Garuda, Kalimati;
372     font-weight: bold;
373     position: absolute;
374     color: #fff;
375     top: 85px;
376     font-size: 3.4em;
377     letter-spacing: -0.06em;
378     margin: 0;
379     padding: 0;
380 }
381
382 #header h1 a{
383     color: #fff;
384     text-decoration: none;
385 }
386
387 #header h1 a:hover{
388     background: none;
389     color: #2d83d5;
390 }
391
392 /* top navigation */
393 #topnav{
394     color: #a2a3a6;
395     padding-top: 20px;
396 }
397
398 #topnav a{
399     font-size: 110%;
400     text-transform: uppercase;
401     text-decoration: none;
402     color: #afb0b3;

```

```

403     padding: 0 8px;
404 }
405
406 #topnav a:hover{
407     text-decoration: none;
408     background: none;
409     color: #fff;
410 }
411
412
413 /* header tabs */
414 #tabs{
415     font-size:14px;
416     width: 100%;
417     line-height:normal;
418     position: absolute;
419     bottom: 0;
420     left: 0;
421     z-index: 0;
422 }
423
424 #tabs ul{
425     list-style:none;
426 }
427
428 #tabs li{
429     display:inline;
430     float: left;
431     position: relative;
432     margin: 0;
433     padding: 0;
434 }
435
436
437 #tabs a {
438     float:left;
439     background:url(images/tab-left.png) no-repeat left top;
440     color:#0470a0;
441     padding-left:9px;
442     margin-right:3px;
443     text-decoration:none;
444     color:#cfcfcf;
445     cursor: pointer; /* IE 7 bug-fix */
446 }
447
448 #tabs a span{
449     float:left;
450     display:block;
451     background:url(images/tab-right.png) no-repeat right top;
452     height: 29px;
453     padding-right: 10px;
454 }
455
456 #tabs a span span{
457     display:block;
458     background:none;
459     height: auto;
460     padding: 4px 4px 0 4px;
461 }
462
463 /* Commented Backslash Hack hides rule from IE5-Mac */
464 #tabs a span {float:none;}
465 /* End IE5-Mac hack */
466 #tabs a:hover span {
467     color:#cfcfcf;
468 }
469
470 #tabs a:hover, #tabs li:hover a{
471     background-position:0% -29px;
472     color:#fff;
473 }
474 #tabs a:hover span, #tabs li:hover a span{
475     background-position:100% -29px;
476     color:#fff;

```

```

477 }
478
479 #tabs a.active, #tabs a.active:hover,
480 #tabs li.current_page_item a, #tabs li.current_page_item a:hover,
481 #tabs li.current_page_ancestor a, #tabs li.current_page_ancestor a:hover,
482 #tabs li.current-cat a, #tabs li.current-cat a:hover,
483 #tabs li.current-cat-parent a, #tabs li.current-cat-parent a:hover{
484     background-position:0% -58px;
485 }
486
487 #tabs a.active span, #tabs a.active:hover span,
488 #tabs li.current_page_item a span, #tabs li.current_page_item a:hover span,
489 #tabs li.current_page_ancestor a span, #tabs li.current_page_ancestor a:hover span,
490 #tabs li.current-cat a span, #tabs li.current-cat a:hover span,
491 #tabs li.current-cat-parent a span, #tabs li.current-cat-parent a:hover span{
492     background-position:100% -58px;
493     color:#000;
494 }
495
496 /* sub-menus */
497
498 #tabs ul ul {
499     background:#eeeeef url(images/tab-subnav.png) repeat-x left top;
500     position: absolute;
501     display: none;
502     width:200px;
503     top: 29px;
504     left: 0;
505     margin: 0;
506     padding: 4px 0;
507     line-height: 100%;
508     z-index: 99;
509     border-left: 1px solid #999;
510     border-right: 1px solid #999;
511     border-bottom: 1px solid #999;
512     -moz-border-radius-bottomleft: 4px;
513     -moz-border-radius-bottomright: 4px;
514 }
515
516 #tabs ul ul ul{
517     border: 1px solid #999;
518     -moz-border-radius: 4px;
519 }
520
521 #tabs ul a.active ul,
522 #tabs ul li.current_page_item ul,
523 #tabs ul li.current_page_ancestor ul,
524 #tabs ul li.current-cat ul,
525 #tabs ul li.current-cat-parent ul{
526     background: #fff;
527 }
528
529 #tabs ul.active ul ul,
530 #tabs ul li.current_page_item ul ul,
531 #tabs ul li.current_page_ancestor ul ul,
532 #tabs ul li.current-cat ul ul,
533 #tabs ul li.current-cat-parent ul ul{
534     border: 1px solid #666666 !important;
535 }
536
537 #tabs ul li ul a{
538     width:180px;
539     height:auto;
540     float:left;
541     background: none;
542     margin: 0;
543     padding: 4px 10px;
544     color: #000;
545     font-weight: normal;
546     line-height: 125%;
547 }
548
549 #tabs li:hover ul a span{ color: #000; }
550

```

```

551 #tabs ul li ul a span, #tabs ul li ul a span span{
552     margin: 0;
553     padding: 0;
554     height: auto;
555     float: none;
556 }
557
558 #tabs ul ul a: hover, #tabs ul ul a: hover,
559 #tabs ul a. active ul a: hover,
560 #tabs ul li. current_ page_ item ul a: hover,
561 #tabs ul li. current_ page_ ancestor ul a: hover,
562 #tabs ul li. current- cat ul a: hover,
563 #tabs ul li. current- cat- parent ul a: hover{
564     background: #09c6eb url(images/tab-subnav-active.gif) repeat-x left top;
565 }
566
567 #tabs ul ul a: hover, #tabs ul ul a: hover span,
568 #tabs ul a. active ul a: hover span,
569 #tabs ul li. current_ page_ item ul a: hover span,
570 #tabs ul li. current_ page_ ancestor ul a: hover span,
571 #tabs ul ul li. current_ page_ parent a: hover span,
572 #tabs ul ul li. current_ page_ item a: hover span,
573 #tabs ul ul li. current_ page_ parent li. current_ page_ item a: hover span,
574 #tabs ul ul li. current- cat a: hover span,
575 #tabs ul ul li. current- cat- parent li. current_ page_ item a: hover span{
576     color: #fff;
577 }
578
579 #tabs ul li ul a span, #tabs ul li ul a span{
580     background: none;
581 }
582
583 #tabs ul ul ul{
584     top: auto;
585 }
586
587 #tabs ul li ul ul {
588     left: 195px;
589     top: 4px;
590 }
591
592 #tabs ul li: hover ul ul, #tabs ul li: hover ul ul ul, #tabs ul li: hover ul ul ul ul{
593     display: none;
594 }
595 #tabs ul li: hover ul, #tabs ul li li: hover ul, #tabs ul li li li: hover ul, #tabs ul li li li li: hover ul{
596     display: block;
597 }
598
599 /* active submenus */
600 #tabs ul ul li. current_ page_ parent a span,
601 #tabs ul ul li. current_ page_ item a span,
602 #tabs ul ul li. current_ page_ parent li. current_ page_ item a span,
603 #tabs ul ul li. current- cat a span,
604 #tabs ul ul li. current- cat- parent li. current_ page_ item a span{
605     color: #2d83d5;
606 }
607
608 #tabs ul ul li. current_ page_ item li a span, #tabs ul ul li. current_ page_ parent li a span,
609 #tabs ul ul li. current- cat li a span, #tabs ul ul li. current- cat- parent li a span{
610     color: #000;
611 }
612
613 /* IE 7 bug-fixes */
614 #tabs ul li: hover { z-index: 100; }
615 #tabs ul * li: hover { visibility: inherit; }
616
617
618 /*** MAIN ***/
619
620 /* sort news links (pligg) */
621 #sortlinks{
622     display: block;
623     margin: 5px 0 20px;
624 }

```

```

625
626
627 /* secondary tabs */
628
629 #mid ul.menu {
630     font-size:14px;
631     height:35px;
632     line-height:normal;
633     left: 0;
634     padding-left: 12px;
635     background:transparent url(images/mtab-bg.png) repeat-x left bottom;
636     margin: 10px 0 20px;
637     list-style:none;
638 }
639
640 #mid ul.menu li {
641     display:inline;
642     float:left;
643     margin: 0;
644     padding: 0;
645 }
646
647 #mid ul.menu a {
648     float:left;
649     background:url(images/mtab-left.png) no-repeat left -35px;
650     color:#0470a0;
651     padding-left:6px;
652     margin-right:3px;
653     text-decoration:none;
654     text-transform: uppercase;
655     color:#8c8b8b;
656     cursor: pointer; /* IE 7 bug-fix */
657 }
658
659 #mid ul.menu a span{
660     float:left;
661     display:block;
662     background:url(images/mtab-right.png) no-repeat right -35px;
663     height: 35px;
664     padding-right: 6px;
665 }
666
667 #mid ul.menu a span span{
668     display:block;
669     background:none;
670     height: auto;
671     padding: 7px 10px 0 10px;
672 }
673
674 /* Commented Backslash Hack hides rule from IE5-Mac */
675 #mid ul.menu a span {float:none;}
676 /* End IE5-Mac hack */
677 #mid ul.menu a:hover span {
678     color:#cfcfcf;
679 }
680
681 #mid ul.menu a:hover{
682     background-position:0% -70px;
683     color:#fff;
684 }
685 #mid ul.menu a:hover span{
686     background-position:100% -70px;
687     color:#fff;
688 }
689
690 #mid ul.menu a.active, #mid ul.menu a.current{
691     background-position:0% 0px;
692     color:#ed2124;
693 }
694 #mid ul.menu a.active span, #mid ul.menu a.current span {
695     background-position:100% 0px;
696     color:#ed2124;
697 }
698

```



```

699 #mid #tab-2, #mid #tab-3, #mid #tab-4, #mid #tab-5, #mid #tab-6, #mid #tab-7, #mid #tab-8, #mid #tab-9{
700     display: none;
701 }
702
703 .story, .post, .attachment{
704     background: transparent url(images/div-h2.gif) repeat-x left 8px;
705     padding-top: .6em;
706     margin: 0 0 2.8em;
707     display: block;
708     position: relative;
709     min-width: 0; /* IE7 peakaboo fix */
710 }
711
712
713 #mid .story h2, #mid .post h2, #mid .attachment h2{
714     margin: 0;
715 }
716
717 .story h2 a, .post h2 a, .story h3 a, .post h3 a{
718     color: #000;
719     text-decoration: none;
720 }
721
722 .story h2 a:hover, .post h2 a:hover, .story h3 a:hover, .post h3 a:hover{
723     background: none;
724     color: #ed1f24;
725 }
726
727 /* voting (only for Pligg) */
728
729 .story .vote{
730     width: 70px;
731     height: 89px;
732     background: transparent url(images/vote-up.png) no-repeat left top;
733     float:left;
734     display: block;
735     position: relative;
736 }
737
738 .story .vote span.votenum{
739     text-align: center;
740     width: 70px;
741     display: block;
742     margin-top: 10px;
743     position: absolute;
744     top: 0;
745     left: 0;
746 }
747
748 .story .vote a{
749     color: #fff;
750     text-decoration: none;
751 }
752
753 .story .vote span.votenum a{
754     font-size: 30px;
755 }
756
757 .story .vote span.votenum a:hover{
758     background: none;
759     text-decoration: underline;
760 }
761
762 .story .vote span.subtext{
763     background: transparent url(images/vote-text.png) no-repeat left top;
764     text-align: center;
765     width: 70px;
766     height: 29px;
767     position: absolute;
768     top: 62px;
769     left: 0;
770 }
771
772 /* story header */

```

```

773
774 .story .postheader, .post .postheader, .attachment .postheader{
775     display: block;
776     margin-left: 74px;
777     position: relative;
778 }
779
780 .story p.postcategory, .post p.postcategory, .attachment p.postcategory{
781     color: #e5e5e5;
782 }
783
784 .story .postinfo, .post .postinfo, .attachment .postinfo{
785     background: #e2e3e3 url(images/postinfo-bg.gif) repeat-x left top;
786 }
787
788 .story .postinfo p, .post .postinfo p, .attachment .postinfo p{
789     padding: .4em .8em;
790 }
791
792 /* story body */
793
794 .story .postbody, .post .postbody, .attachment .postbody{
795     clear: left;
796     display: block;
797 }
798
799 .story p.postcontrols, .post p.postcontrols, .attachment p.postcontrols{
800     color: #e5e5e5;
801     float: right;
802     margin: 0;
803 }
804
805 .story p.tags, .post .tags, .attachment p.postcontrols{
806     float: left;
807     display: block;
808     padding-left: 19px;
809     margin: 0;
810     background:transparent url(images/post-icons.png) no-repeat 0 top;
811 }
812
813 .story p.postcontrols a, .story p.postcategory a,
814 .post p.postcontrols a, .post p.postcategory a,
815 .attachment p.postcontrols a, .attachment p.postcategory a{
816     padding: 0 2px;
817 }
818
819 /* comments link */
820
821 a.comments, .story p.postcontrols a.comments, .post p.postcontrols a.comments{
822     background:transparent url(images/post-icons.png) no-repeat 2px bottom;
823     padding-left: 19px;
824     color:#ed1f24;
825 }
826
827 a.comments:hover, .story p.postcontrols a.comments:hover, .post p.postcontrols a.comments:hover{
828     background: #ed1f24;
829     color: #fff;
830 }
831
832 a.no.comments, .post p.postcontrols a.no.comments{
833     color: #2d83d5;
834 }
835
836 a.no.comments:hover, .post p.postcontrols a.no.comments:hover{
837     background: #2d83d5;
838     color: #fff;
839 }
840
841
842 /* comments */
843
844 ol#comments, ol#comments li,
845 ol#trackbacks, ol#trackbacks li{
846     list-style-type: none;

```

```

847     padding: 0;
848     margin: 0;
849 }
850
851 ol#comments ul.children, ol#trackbacks ul.children{
852     margin: 0;
853     padding: 0;
854 }
855
856 li.comment{
857     background: transparent url(images/comment-div-bg.png) repeat-y left bottom;
858     list-style-position: outside; /* ie 7 bugfix */
859     display: block;
860     position: relative;
861     min-height: 110px;
862 }
863
864 li.comment .wrap{
865     background: transparent url(images/comment-div-start.png) no-repeat left top;
866     min-height:110px;
867     height:auto !important; /* ie6 ignores it */
868     height:110px; /* fix */
869 }
870
871 li.comment.with-avatars{ background-position: 96px bottom; }
872 li.comment.with-avatars .wrap{ background-position: 96px top; }
873
874 /* gravatar */
875 li.comment .avatar{
876     float: left;
877     width: auto !important;
878     padding-top:6px;
879     padding-left: 8px;
880 }
881
882 li.comment .avatar img{
883     padding: 3px;
884     margin: 4px 10px;
885     border: 1px solid #E7E8E6;
886     border-radius: 4px;
887     -moz-border-radius: 4px;
888 }
889
890 li.comment .details{ display: block; margin-left: 12px; }
891 li.comment.with-avatars .details{ margin-left: 108px;}
892
893 li.comment .details .head{
894     background: #f7f8f8 url(images/postinfo-bg.gif) repeat-x left bottom;
895     padding: 2px 10px;
896     margin: 0;
897     display: block;
898     color: #8f9090;
899 }
900
901 li.comment .details.admincomment .head{
902     background: #59b2e4 url(images/postinfo-bg-admin.gif) repeat-x left bottom;
903     color: #fff;
904 }
905
906 li.comment .details.admincomment .head a{ color: #fff; }
907 li.comment .details.admincomment .head a:hover{ background: none; color: #fcb722; }
908
909
910 /* comment rate links, pligg only */
911
912 span.rate{
913     right: 8px;
914     position: absolute;
915 }
916
917 span.rate a.karma{
918     font-family: "Arial Black", Helvetica, Sans, FreeSans, Jamrul, Garuda, Kalimati;
919     font-weight: bold;
920     font-size: 110%;

```

```

921     text-decoration: none;
922     padding: 0;
923 }
924
925 span.rate a.karma.negative{ color:#ed2124; }
926 span.rate a.karma.positive{ color:#6fb23c; }
927
928 span.rate a.karma:hover{
929     background: none;
930 }
931
932 span.rate a.ratelink{
933     padding: 4px;
934     text-decoration: none;
935 }
936
937 span.rate a.ratelink.up{ background: transparent url(images/thumb-up.gif) no-repeat left center; }
938 span.rate a.ratelink.down{ background: transparent url(images/thumb-down.gif) no-repeat left center; }
939
940 li.comment .details .text{
941     background: transparent url(images/comment-bg.png) repeat-x left top;
942     padding: .3em .8em;
943     margin-top: 3px;
944     min-height: 64px;
945     font-size: 90%;
946     color: #4e5359;
947 }
948
949 li.comment .details.admincomment .text{
950     background: #f0f0f0 url(images/comment-bg-admin.png) repeat-x left top;
951 }
952
953 div.act{
954     position: absolute;
955     display: none;
956     right: 0;
957     top: 0;
958     margin-top: 50px;
959     font-size: 110%;
960     text-transform: uppercase;
961 }
962
963 span.button a{
964     background: transparent url(images/button-bg.png) no-repeat right top;
965     display: block;
966     height: 31px;
967     float: left;
968     margin-right: 3px;
969     cursor: pointer;
970 }
971
972 span.button a span{
973     float: left;
974     height: 31px;
975     display: block;
976     padding: 0 10px 0 25px;
977 }
978
979 span.quote a span{ background: transparent url(images/button-quote.png) no-repeat left top; }
980 span.reply a span{ background: transparent url(images/button-reply.png) no-repeat left top; }
981 span.submit a span{ background: transparent url(images/button-submit.png) no-repeat left top; }
982
983 span.button a span span{
984     background: none;
985     display: block;
986     padding: 6px 10px 0px 10px;
987     height: auto;
988 }
989
990 span.button a:hover{ background-position: right bottom; }
991 span.button a:hover span{ background-position: left bottom; }
992
993
994 /* sidebars */

```

```

995
996 #sidebar h2.title, #sidebar2 h2.title{
997     font-family: "Lucida Grande", "Lucida Sans Unicode", Arial, Helvetica, Sans, FreeSans, Jamrul, Garuda, Kalimati;
998     font-weight: normal;
999     text-transform: uppercase;
1000     font-size: 130%;
1001 }
1002
1003 #sidebar ul, #sidebar2 ul{
1004     margin: 0;
1005     padding: 0;
1006 }
1007
1008 #sidebar ul li, #sidebar2 ul li{
1009     list-style-type: none;
1010     line-height: 140%;
1011     margin: 0;
1012     padding: 0;
1013     display: block;
1014     list-style-position: outside; /* ie 7 bugfix */
1015 }
1016
1017 /* sidebar menu */
1018
1019 #sidebar ul.nav{
1020     margin: 1em 0 2em;
1021     padding: 0;
1022 }
1023
1024 #sidebar ul.nav li{
1025     list-style-type: none;
1026     background: transparent url(images/div-h.gif) repeat-x left bottom;
1027     padding: 2px 0 3px 0;
1028     margin: 0;
1029     position: relative;
1030 }
1031
1032 /* no bg on the last menu item */
1033 #sidebar ul.nav li.last{
1034     background: none
1035 }
1036
1037
1038 #sidebar ul.nav ul{
1039     margin: 0px 0 12px;
1040 }
1041
1042 #sidebar ul.nav ul ul{
1043     margin: 0px 0 8px; /* lower bottom-padding for 2+ level menu */
1044 }
1045
1046 /* 2nd level menu */
1047 #sidebar ul.nav li li{
1048     background: none;
1049     padding: 0;
1050     margin-left: 25px;
1051 }
1052
1053 /* 3rd ... n levels */
1054 #sidebar ul.nav li li li{
1055     margin-left: 10px;
1056 }
1057
1058 #sidebar ul.nav li a{
1059     background: none;
1060     display: block;
1061     text-decoration: none;
1062     text-transform: uppercase;
1063     font-size: 110%;
1064     padding: 3px 27px 3px 8px;
1065     color: #000;
1066     cursor: pointer;
1067 }
1068

```

```

1069 #sidebar ul.nav li a:hover{
1070     background: #efefee url(images/side-nav-bg.png) repeat-y right top;
1071     color: #ed2124;
1072 }
1073
1074 #sidebar ul.nav li a span{
1075     background: transparent url(images/b1.png) no-repeat left 3px;
1076     padding-left: 17px;
1077     display: block;
1078 }
1079
1080 #sidebar ul.nav li a:hover span{
1081     background-position: left -97px;
1082     color: #ed2124;
1083 }
1084
1085 #sidebar ul.nav li li a{
1086     text-transform: none;
1087     font-size: 100%;
1088     padding: 0;
1089 }
1090
1091 #sidebar ul.nav li li a:hover{
1092     background: none
1093 }
1094
1095 #sidebar ul.nav li li a span{
1096     background: none;
1097     padding: 0;
1098 }
1099
1100 #sidebar ul.nav li a.rss{
1101     background: transparent url(images/side-nav-rss.gif) no-repeat left top;
1102     display: none;
1103     width: 26px;
1104     position: absolute;
1105     height: 13px;
1106     right: 0;
1107     padding: 0 4px 0 0;
1108     margin-top: 2px;
1109     top: 0;
1110     text-decoration: none;
1111     cursor: pointer;
1112     z-index: 5;
1113 }
1114
1115 #sidebar ul.nav li li a.rss{
1116     visibility: hidden;
1117 }
1118
1119 #sidebar ul.nav li a.rss:hover{
1120     background-position: 0px -13px;
1121 }
1122
1123
1124 /* sidebar content box */
1125
1126 #sidebar .box, #sidebar2 .box{
1127     margin: 16px 0 2px;
1128 }
1129
1130 #sidebar .box h2.title, #sidebar2 .box h2.title{
1131     background: transparent url(images/side-box-top.png) no-repeat left bottom;
1132     padding: 4px 0 4px 20px;
1133     color: #8d8b8b;
1134     margin: 20px 0 0 0;
1135 }
1136
1137 #sidebar .box ul, #sidebar2 .box ul{
1138     background: transparent url(images/side-box-bg.png) repeat-y left bottom;
1139     padding: 10px 20px;
1140     display: block;
1141 }
1142

```

```

1143 #sidebar .box ul ul, #sidebar2 .box ul ul{ background: none; }
1144 #sidebar .box ul li, #sidebar2 .box ul li{ line-height: 150%; }
1145
1146 #sidebar .box caption, #sidebar2 .box caption{
1147     text-align: left;
1148     font-weight: bold;
1149 }
1150
1151
1152 /* sidebar search */
1153
1154 #searchtab{
1155     background: transparent url(images/search-bg.png) no-repeat left top;
1156     height: 40px;
1157     position: relative;
1158 }
1159
1160 #searchtab .inside{
1161     background: transparent url(images/search-go.png) no-repeat right top;
1162     height: 40px;
1163 }
1164
1165 #searchtab input{
1166     border: 0;
1167     background: none;
1168     font-size: 12px;
1169     padding: 0px;
1170 }
1171
1172 #searchtab input.searchfield, #searchtab input#s{
1173     position: absolute;
1174     top: 6px;
1175     left: 46px;
1176     color: #949496;
1177     padding: 0px;
1178     margin: 0;
1179     width: 50%;
1180 }
1181
1182 #searchtab input.searchfield:focus, #searchtab input#s:focus{ color: #000; }
1183
1184 #searchtab input.searchbutton, #searchtab input#searchsubmit{
1185     position: absolute;
1186     right: 37px;
1187     top: 6px;
1188     color: #e5e5e5;
1189     text-transform: uppercase;
1190     padding: 0px;
1191     margin: 0;
1192 }
1193
1194 #searchtab label{ display: none; }
1195 #searchtab input.searchbutton:hover{ color: #fff; }
1196
1197
1198
1199 /*** FOOTER ***/
1200
1201 #footer{
1202     background: transparent url(images/div-h2.gif) repeat-x left top;
1203     padding: 20px 0;
1204     margin-top: 2em;
1205     width: 100%;
1206
1207     text-align: center;
1208 }
1209
1210 #footer a#toplink{
1211     text-decoration: none;
1212     padding: 0 .6em;
1213 }
1214
1215
1216 /*** top right page controls ***/

```

```

1217
1218 #layoutcontrol{
1219     position: absolute;
1220     top: 10px;
1221     right: 20px;
1222     display: block;
1223     width: 74px;
1224     height: 16px;
1225     z-index: 1000;
1226 }
1227
1228 #layoutcontrol a{
1229     float: left;
1230     display: block;
1231     width: 16px;
1232     height: 16px;
1233     margin-right: 3px;
1234     background: transparent url(images/pagecontrols.png) no-repeat left top;
1235 }
1236
1237 #layoutcontrol a.setLiquid{ background-position: -50px top; }
1238 #layoutcontrol a.setFont{ background-position: left top; width: 25px; }
1239
1240 #layoutcontrol a.setLiquid:hover{ background-position: right top; }
1241 #layoutcontrol a.setFont:hover{ background-position: -25px top; }
1242 #layoutcontrol a span{ display: none; }
1243
1244
1245 /* misc... */
1246
1247 .left{ float: left; }
1248 .right{ float: right; }
1249 div.clear{ clear: both;height:1px; }
1250
1251 /* clearfix */
1252 .clearfix:after{
1253     content: ".";
1254     display: block;
1255     height: 0;
1256     clear: both;
1257     visibility: hidden;
1258 }
1259
1260 .clearfix{
1261     min-width: 0;                /* trigger hasLayout for IE7 */
1262     display: inline-block;
1263     /* \*/ display: block;      /* Hide from IE Mac */
1264 }
1265
1266 * html .clearfix{
1267     /* \*/ height: 1%;          /* Hide from IE Mac */
1268 }
1269
1270 .error, .red{ color:#ed1f24; }
1271
1272
1273 /*** WORDPRESS SPECIFIC STYLES ***/
1274
1275 .widget{
1276     margin: 1em .6em 1.8em .6em;
1277     min-width: 0; /* IE7 peakaboo fix */
1278     padding: 0;
1279 }
1280
1281 .widget ul ul{
1282     margin: 0 0 0 .8em;
1283     padding: 0;
1284 }
1285
1286 #sidebar .widget h2.title,
1287 #sidebar2 .widget h2.title{
1288     background: transparent url(images/div-h2.gif) repeat-x left bottom;
1289     color: #8b8b89;
1290     padding-bottom: 2px;

```



```

1291     font-size: 125%;
1292     margin-bottom: .4em;
1293 }
1294
1295 #sidebar li.linkcat h2.title,
1296 #sidebar2 li.linkcat h2.title{
1297     background: transparent url(images/div-h2.gif) repeat-x left bottom;
1298     color: #8b8b89;
1299     padding-bottom: 2px;
1300     text-align: right;
1301     font-style: italic;
1302     font-size: 115%;
1303 }
1304
1305 /* lists with arrow bullets on some of the widgets */
1306 #sidebar li.linkcat li, #sidebar2 li.linkcat li,
1307 .widget.widget_links ul li, .widget.widget_meta ul li, .widget.widget_categories ul li, .widget.widget_archive ul li,
1308     background: transparent url(images/b3.gif) no-repeat left 6px;
1309 }
1310
1311 #sidebar li.linkcat a, #sidebar2 li.linkcat a,
1312 .widget.widget_links ul li a, .widget.widget_meta ul li a, .widget.widget_categories ul li a, .widget.widget_archive ul li a,
1313     color: #8b8b89;
1314     text-decoration: none;
1315     margin-left: 10px;
1316     display: inline-block;
1317 }
1318
1319 .widget_tag_cloud a{
1320     vertical-align: middle;
1321     text-decoration: none;
1322     padding: 0 0.2em;
1323     letter-spacing: -0.04em;
1324 }
1325
1326
1327 #sidebar li.linkcat a:hover, #sidebar2 li.linkcat a:hover,
1328 .widget.widget_links ul li a:hover, .widget.widget_meta ul li a:hover, .widget.widget_categories ul li a:hover, .widget.widget_archive ul li a:hover,
1329     color: #ed1f24;
1330     background: none;
1331     text-decoration: underline;
1332 }
1333
1334 /* recent comments widget */
1335 .widget ul#recentcomments li{
1336     background: transparent url(images/post-icons.png) no-repeat left -141px;
1337     padding-left: 18px;
1338     display: block;
1339     font-style: italic;
1340     margin-bottom: 5px;
1341 }
1342
1343 .widget ul#recentcomments li a.url{
1344     border-bottom: 1px dotted #8b8b89;
1345     color: #000;
1346     text-decoration: none;
1347     font-style: normal;
1348 }
1349
1350 .widget ul#recentcomments li a.url:hover{
1351     border-bottom: 1px solid #8b8b89;
1352     background: none;
1353 }
1354
1355 .widget#tag_cloud{ text-align: justify; }
1356 .widget#tag_cloud a{ line-height: 125%; }
1357
1358
1359 /* calendar widget */
1360 .widget_calendar h2{
1361     display: none;
1362 }
1363
1364 .widget_calendar table#wp-calendar{

```

```

1365     width: 100%;
1366     border: 1px solid #ccc;
1367 }
1368
1369 .widget_calendar table#wp-calendar th, .widget_calendar table#wp-calendar td{
1370     text-align: center;
1371 }
1372
1373 .widget_calendar table#wp-calendar thead th{
1374     font-weight: bold;
1375     background: #ccc;
1376     color: #fff;
1377     border: 0;
1378 }
1379
1380 .widget_calendar table#wp-calendar tbody td{
1381     border: 0;
1382     background: #eee;
1383     color: #555;
1384 }
1385
1386 .widget_calendar table#wp-calendar tbody td.pd{
1387     background: #fff;
1388 }
1389
1390 .widget_calendar table#wp-calendar tfoot td{
1391     border: 0;
1392     background: #fff;
1393 }
1394
1395 .widget_calendar table#wp-calendar caption{
1396     text-align: right;
1397     font-style: italic;
1398     color: #aaa;
1399 }
1400
1401
1402 .post .postheader{
1403     display: block;
1404     margin-left:0;
1405 }
1406
1407 .post .postbody img, .attachment img, .navigation .alignleft img, .navigation .alignright img{
1408     padding: 3px;
1409     margin: 4px 10px;
1410     border: 1px solid #E7E8E6;
1411     border-radius: 4px;
1412     -moz-border-radius: 4px;
1413 }
1414
1415 .post img.alignleft, .attachment img.alignleft{ margin-left: 0; }
1416 .post img.alignright, .attachment img.alignright{ margin-right: 0; }
1417 .post img.aligncenter, .attachment img.aligncenter{ margin: 0 auto; text-align: center; }
1418
1419
1420 .attachment-entry{
1421     text-align: center;
1422 }
1423
1424 .attachment div.caption{
1425     background:#F7F7F7;
1426     border: 1px solid #E7E8E6;
1427     border-radius: 4px;
1428     -moz-border-radius: 4px;
1429     padding-top:5px;
1430     margin:4px 10px;
1431     text-align:center;
1432     color: #8d8b8b;
1433     min-width: 0; /* ie7 peakaboo fix */
1434 }
1435
1436 /* no border on smilies */
1437 img.wp-smiley, .post .postbody img.wp-smiley, .attachment img.wp-smiley{
1438     padding: 0;

```

```

1439     margin: 0;
1440     border: 0;
1441 }
1442
1443 .wp-caption {
1444     background:#F7F7F7;
1445     border: 1px solid #E7E8E6;
1446     border-radius: 4px;
1447     -moz-border-radius: 4px;
1448     padding-top:5px;
1449     margin:4px 10px;
1450     text-align:center;
1451     color: #8d8b8b;
1452 }
1453
1454 .wp-caption img, .post .postbody .wp-caption img, .attachment .wp-caption img{
1455     margin: 0;
1456     border: 0;
1457     padding: 0;
1458 }
1459
1460 .wp-caption.alignleft{ margin-left: 0; }
1461 .wp-caption.alignright{ margin-right: 0; }
1462
1463
1464 .navigation{
1465     font-size: 80%;
1466     color: #c4c3c1;
1467     margin: 0;
1468     padding: 0;
1469 }
1470
1471 .navigation a{
1472     text-decoration: none;
1473     color: #d2d0cf;
1474 }
1475
1476 .navigation a:hover{
1477     background: none;
1478     color:#ed1f24;
1479 }
1480
1481 .postmetadata{
1482     color: #c4c3c1;
1483     margin: 1em 0;
1484 }
1485
1486 .postmetadata a{
1487     color: #929292;
1488 }
1489
1490 .postmetadata a:hover{
1491     background: none;
1492     color:#ed1f24;
1493 }
1494
1495 span.editlink{
1496     position: absolute;
1497     right: 10px;
1498     top: 5px;
1499 }
1500
1501 span.editlink.page{
1502     position: relative;
1503     top: 20px;
1504 }
1505
1506 span.editlink a{
1507     display: block;
1508     height: 16px;
1509     background: transparent url(images/edit.png) no-repeat left top;
1510     padding-left: 17px;
1511     color: #000;
1512     text-decoration: none;

```

```

1513 }
1514
1515 span.editlink a:hover{
1516     background-position: left bottom;
1517     background-color: none;
1518     color:#ed1f24;
1519 }
1520
1521 span.advedit a{
1522     background: transparent url(images/edit.png) no-repeat left top;
1523     display: block;
1524     width: 16px;
1525     height: 16px;
1526 }
1527
1528 #footer a.rss, a.rsssubscribe{
1529     background: transparent url(images/rss.png) no-repeat left 50%;
1530     padding-left: 18px;
1531 }
1532
1533 #footer a.rss:hover, a.rsssubscribe:hover{
1534     background: #2d83d5 url(images/rss.png) no-repeat left 50%;
1535 }
1536
1537 a.rsssubscribe{
1538     margin: 6px 0 0 10px;
1539     display: block;
1540 }
1541
1542
1543 #commentform{
1544     display: block;
1545     background: transparent url(images/div-h2.gif) repeat-x left 0px;
1546     padding: 6px 0 0 0;
1547     margin: 20px 0;
1548 }
1549
1550 #commentform textarea{
1551     margin-bottom: 6px;
1552 }
1553
1554 #commentform #commentdata label{
1555     background: #FFFF99;
1556 }
1557
1558
1559 /* pagination */
1560
1561 .commentnavi{
1562     margin: 10px 0;
1563     display: block;
1564 }
1565
1566 .commentnavi .page-numbers, #pagenavi a, #pagenavi span.pages, #pagenavi span.extend{
1567     padding: 2px 6px 2px 6px !important;
1568     border: 1px solid #ccc !important;
1569     background: #fff !important;
1570     margin:0;
1571 }
1572
1573 #pagenavi a, #pagenavi span.pages, #pagenavi span.extend{
1574     margin: 0 2px;
1575 }
1576
1577
1578 #pagenavi span.current{
1579     background: #ccc !important;
1580     color: #fff !important;
1581     padding: 2px 6px 2px 6px !important;
1582     border: 1px solid #ccc !important;
1583 }
1584
1585 .commentnavi a{
1586     color: #4a9dce;

```

```

1587     text-decoration: none;
1588     position: relative;
1589 }
1590
1591 .commentnavi a:hover{ color: #00619d; }
1592 #comment .page-numbers.current, #pagenavi span.pages{ color: #ccc !important; }
1593
1594 #pagenavi{
1595     display: block;
1596     padding-bottom: 10px;
1597     font-size: 100% !important;
1598 }
1599
1600 /* threaded comments? */
1601 ol#comments li.comment.depth-2,
1602 ol#comments li.comment.depth-3,
1603 ol#comments li.comment.depth-4,
1604 ol#comments li.comment.depth-5,
1605 ol#comments li.comment.depth-6,
1606 ol#comments li.comment.depth-7,
1607 ol#comments li.comment.depth-8,
1608 ol#comments li.comment.depth-9{ margin-left: 14px; }
1609
1610 /* threaded comments, with avatars */
1611 ol#comments li.comment.with-avatars.depth-2,
1612 ol#comments li.comment.with-avatars.depth-3,
1613 ol#comments li.comment.with-avatars.depth-4,
1614 ol#comments li.comment.with-avatars.depth-5,
1615 ol#comments li.comment.with-avatars.depth-6,
1616 ol#comments li.comment.with-avatars.depth-7,
1617 ol#comments li.comment.with-avatars.depth-8,
1618 ol#comments li.comment.with-avatars.depth-9{ margin-left: 96px; }
1619
1620
1621 .aligncenter {
1622     display: block;
1623     margin-left: auto;
1624     margin-right: auto;
1625 }
1626
1627 .alignleft { float: left; }
1628 .alignright { float: right; }
1629
1630 li.infotext{
1631     text-align: center;
1632     padding: 2px 12px 0 6px;
1633 }
1634
1635 img.wp-smiley { vertical-align: -20%; }
1636
1637
1638 .profile .avatar img{
1639     padding: 3px;
1640     margin: 0 6px 6px 0;
1641     border: 1px solid #E7E8E6;
1642     border-radius: 4px;
1643     -moz-border-radius: 4px;
1644 }
1645
1646 .profile .info{
1647     margin-left: 148px;
1648 }
1649
1650 p.im{
1651     margin: .1em 0 .2em;
1652     padding: 1px 0;
1653 }
1654

```