



Rafael Savignon Marinho

Ginga-NCL como plugin para navegadores Web

Dissertação de Mestrado

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-Graduação em Informática do Departamento de Informática da PUC-Rio.

Orientador: Prof. Luiz Fernando Gomes Soares

Rio de Janeiro
Outubro de 2010



Rafael Savignon Marinho

Ginga-NCL como plugin para navegadores Web

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-Graduação em Informática do Departamento de Informática do Centro Técnico e Científico da PUC-Rio. Aprovada pela Comissão Examinadora abaixo assinada.

Prof. Luiz Fernando Gomes Soares

Orientador

Departamento de Informática – PUC-Rio

Prof. Marcelo Ferreira Moreno

Departamento de Informática – PUC-Rio

Prof. Sérgio Colcher

Departamento de Informática – PUC-Rio

Prof. Antônio Mauro Barbosa de Oliveira

Departamento de Telecomunicações – UNIFOR

Prof. José Eugenio Leal

Coordenador(a) Setorial do Centro Técnico Científico - PUC-Rio

Rio de Janeiro, 19 de outubro de 2010

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador.

Rafael Savignon Marinho

Recebeu seu título de Bacharel em Informática pela Pontifícia Universidade Católica do Rio de Janeiro em 2007. Atualmente integra o grupo de pesquisadores do Laboratório Telemídia, desenvolvendo pesquisas na área de Sistemas Hiperemídia.

Ficha Catalográfica

Marinho, Rafael Savignon

Ginga-NCL como plugin para navegadores Web/
Rafael Savignon Marinho; orientador: Luiz Fernando
Gomes Soares. - 2010.

98 f.; 29,7 cm

Dissertação (mestrado) – Pontifícia Universidade
Católica do Rio de Janeiro, Rio de Janeiro, 2010.

Incluí referências bibliográficas.

1. Informática - Teses. 2. Sistemas Hiperemídia. 3.
plugin. 4. Nested Context Model. 5. Nested Context
Language. 6. Navegador Web. 7. *middleware* Ginga-NCL .
8. DirectX. I. Soares, Luiz Fernando Gomes. II. Pontifícia
Universidade Católica do Rio de Janeiro. Departamento de
Informática. III. Título.

*A minha família por toda a dedicação, apoio
e carinho ao longo dessa jornada.*

Agradecimentos

Durante os dois anos do curso, muitas pessoas contribuíram direta ou indiretamente para a realização desse trabalho. Sem o apoio dessas pessoas seria inviável obter êxito nesta difícil tarefa. Portanto, dedico-lhes aqui sinceros agradecimentos, em especial:

Ao Professor Luiz Fernando Gomes Soares, meu orientador, pela paciência e dedicação em prol da minha superação.

Ao Professor Marcelo Ferreira Moreno, por todos esses anos de apoio, ensinamentos e conselhos, que certamente serão fundamentais durante toda a minha vida profissional.

Ao Professor Sérgio Colcher, pelos conselhos e por acreditar em mim desde o início.

A todos os amigos do laboratório Telemídia, que lá estão ou por lá passaram: Alexandra, Álvaro, Bruno, Carlos de Salles, Carlos Eduardo, Eduardo, Felipe Nagato, Felipe Nogueira, Felipe Bittencourt, Francisco, Guilherme, Heron, José Geraldo, Luciana, Luciana Rosa, Paula, Márcio Moreno, Rafael Rodrigues, Rodrigo Laiola, Rodrigo Borges, Sérgio Cavendish, Ricardo, Romualdo, Roberto Gerson, Rogério Rodrigues, Rogerio Coelho, Suzana, Vitor e Vinicius.

Ao meu querido irmão, Daniel, sua esposa, Rachel, e minha sobrinha, Maria Clara, pelo carinho e apoio.

A minha namorada, Marcela, pela paciência, apoio, conselhos e carinho nos momentos em que mais precisei.

E finalmente, aos meus pais José Renato e Mercedes, pelo carinho e ensinamentos, sem os quais nada seria possível.

Resumo

Marinho, Rafael Savignon; Soares, Luiz F. G. **Ginga-NCL como plugin para navegadores Web.** Rio de Janeiro, RJ, Brasil, 2010. 98p. Dissertação de Mestrado - Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Nos últimos anos, os usuários da Web (World Wide Web) apresentam uma mudança significativa em seu comportamento, tornando-se, além de consumidores, produtores de conteúdo multimídia. Por outro lado, a produção de conteúdo para a TV Digital brasileira, sobre tudo as aplicações interativas escritas na linguagem declarativa NCL (Nested Context Language), ainda é feita por profissionais alocados em emissoras de TV e em produtoras de conteúdo. Considerando o novo perfil dos usuários da grande rede e o fato de que a NCL é uma linguagem hipermídia cujo emprego não se restringe apenas a aplicações da TV Digital, surge a oportunidade de popularizar o uso da NCL como uma nova forma de especificar conteúdo multimídia também para Web. Mais além, uma vez que a NCL é recomendada pelo ITU-T como referência para serviços IPTV, pode-se vislumbrar uma nova plataforma para o oferecimento de tais serviços na Web. Motivado por esse novo cenário, este trabalho propõe a adaptação do *middleware* declarativo Ginga-NCL, que é a camada de software responsável pela execução das aplicações NCL, ao ambiente da Web. A adaptação proposta busca oferecer o controle de apresentação do conteúdo, o suporte a edição em tempo de exibição e a sincronização entre os objetos de mídia contidos em ambos os domínios (Web e TV Interativa). Em suma, o trabalho discute como uma página Web pode se beneficiar da API interna de exibidores e dos demais recursos oferecidos pelo *middleware* Ginga. Adicionalmente também é proposto o suporte a uma nova plataforma pelo *middleware* a fim de facilitar o processo de adaptação.

Palavras-chave

Middleware Ginga; Ginga-NCL; NCL; Plugin Ginga-NCL; Mozilla Firefox; DirectX; Direct3D; NPAPI; Portabilidade; Integração.

Abstract

Marinho, Rafael Savignon; Soares, Luiz F. G (Advisor). **Ginga-NCL as a Web browser plugin**. Rio de Janeiro, RJ, Brasil, 2010. 98p. **MSc . Dissertation** - Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Over the past few years the Web (World Wide Web) users have been presenting a significant change on their behavior, becoming, beside consumers, multimedia content producers. On the other hand, the Brazilian Digital TV content production, more especially the interactive applications written in NCL (Nested Context Language) are still made by professionals allocated in TV broadcasters and content producers companies. Considering the new Web users profile and the fact that NCL is a hypermedia programming language whose scope is not restricted to Digital TV application the opportunity to popularize the use of NCL arises as a new way to specify multimedia content also in the Web. Moreover, once NCL is recommend by ITU-T as reference to IPTV service, is reasonable to realize a new platform to deploy such services on the Web. Motivated by this new scenario this work proposes the adaptation of Ginga-NCL declarative middleware, which is the software layer in charge to execute of the NCL application, to the Web environment. The proposed adaptation aim to offer the content presentation control, live editing support and synchronization among media objects from both domains (Web and Interactive TV). In short, the presented work discusses how a Web page can be benefit by the internal player API and other features offered by the middleware Ginga. In addition is also proposed a new platform support for the middleware in order to facilitate adaptation process.

Keywords

Middleware Ginga; Ginga-NCL; NCL; Plugin Ginga-NCL; Mozilla Firefox; DirectX; Direct3D; NPAPI; Portability; Integration.

Sumário

1	Introdução	13
1.1.	Motivação	13
1.2.	Objetivos	16
1.3.	Organização do Documento	17
2	Conceitos Básicos	18
2.1.	Nested Context Model	18
2.2.	Nexted Context Language	20
3	Trabalhos Relacionados	26
3.1.	Soluções de Reprodutores Hipermedia para Web	26
3.1.1.	Ambulant / SMIL State	27
3.1.2.	Flash Plugin	29
3.1.3.	Comparação	31
3.2.	Middleware Ginga	33
3.2.1.	Arquitetura	34
3.3.	Mozilla Firefox e Gecko SDK	38
3.3.1.	NPAPI	39
3.3.2.	Modelo de Execução de um Plugin	40
3.3.3.	Modos de Apresentação	42
4	Middleware Ginga-NCL como Plugin para Navegadores Web	44
4.1.	Adaptação do Ginga à Arquitetura de Plugins do Mozilla Firefox	47
4.2.	Integração entre as Máquinas de Apresentação (HTML e NCL)	49
4.2.1.	Objetos de Mídia NCL como Objetos de Mídia HTML	50
4.2.2.	Interface para Controle de Apresentação	58
4.2.3.	Interface para Edição em Tempo de Exibição	59
5	Implementação	64
5.1.	Ginga para Plataforma Windows	65
5.1.1.	Gerente Gráfico	66
5.1.2.	Exibidores	69

5.1.3. Sintonizador	74
5.1.4. Dispositivos de entrada	74
5.2. Middleware Ginga-NCL como Plugin para o Navegador Mozilla Firefox	75
5.2.1. Adaptação à Arquitetura de Plugins	76
5.2.2. Integração entre o Navegador e o Middleware Ginga	77
6 Conclusões e Trabalhos Futuros	81
6.1. Contribuições da Dissertação	82
6.2. Trabalhos Futuros	82
7 Referências	84
Apêndice A : DirectX	89
A.1 Direct3D	89
A.2 DirectShow	93
A.3 DirectInput	98

Lista de figuras

Figura 1 – Máquina de Estados de Eventos NCM. Retirado de (SOARES e RODRIGUES, 2005).	19
Figura 2 – Nós, elos e nós de composição. Retirada de (SOARES NETO, SOARES, <i>et al.</i> , 2010).	20
Figura 3 – Aplicação NCL que apresenta um vídeo em tela cheia.	21
Figura 4 – Aplicação NCL que apresenta outro vídeo ao final do primeiro.	22
Figura 5 – Aplicação NCL que apresenta outros vídeos ao longo do primeiro.	24
Figura 6 – Aplicação NCL que exemplifica o uso de interfaces em nós de contexto.	25
Figura 7 – Modelo de dados comum Retirado de (JANSEN e BULTERMAN, 2009).	27
Figura 8 – Aplicação SMIL que utiliza o SMIL State. Retirado e adaptado de (JANSEN e BULTERMAN, 2009).	29
Figura 9 – Aplicação SMIL embutida em uma página Web.	29
Figura 10 – Documento ActionScript que utiliza a ExternalInterface.	30
Figura 11 – Aplicação SWF embutida em uma página Web.	31
Figura 12 – Arquitetura do <i>middleware</i> Ginga. Retirada de (MORENO, 2010).	35
Figura 13 – Ciclo de execução de um plugin Gecko, segundo a NPAPI.	41
Figura 14 – Exemplo de uma aplicação NCL simples (main.ncl).	48
Figura 15 – Objeto de mídia NCL embutido em um documento HTML.	48
Figura 16 – Apresentação da página Web especificada na Figura 15.	49
Figura 17 – Exemplo especificação de um objeto de mídia NCL.	51
Figura 18 – Definição de elo a partir de uma âncora de conteúdo na aplicação NCL embutida.	53
Figura 19 – Especificação de <i>handlers</i> para envio e recebimento de eventos da âncora de conteúdo.	54
Figura 20 – Visão estrutural do exemplo da Figura 18 e da Figura 19.	55
Figura 21 – Definição de elo a partir âncora de propriedade na aplicação NCL embutida.	56
Figura 22 – Especificação de <i>handlers</i> para envio e recebimento de eventos da âncora de propriedade.	56
Figura 23 – Visão estrutural do exemplo da Figura 21 e da Figura 22.	57

Figura 24 – Especificação do controle de apresentação da aplicação NCL embutida em uma página Web.	58
Figura 25 – Exemplo de uso da interface para obter informações sobre Base Privada e seus documentos.	59
Figura 26 – Exemplo de edição em tempo de exibição local de uma aplicação NCL embutida em uma página Web.	62
Figura 27 – Aplicação <i>main.ncl</i> resultante, após os comandos de edição.	63
Figura 28 – Componentes e módulos afetados pela implementação proposta.	64
Figura 29 – Diagrama da classe DXDeviceScreen.	66
Figura 30 – Abstração de Superfícies e Janelas.	67
Figura 31 – Classe DX2Dsurface representa uma região NCL.	68
Figura 32 – Diagrama de classes da biblioteca DshowGraphLib.	70
Figura 33 – Grafo de filtros para reprodução de áudio.	71
Figura 34 – Grafo de filtros para reprodução de vídeo.	72
Figura 35 – Provedor HTML baseado no Awesomium.	72
Figura 36 – Provedor Texto baseado no GDI.	73
Figura 37 – Renderização do Lua Canvas.	73
Figura 38 – Renderização do FreeImage.	74
Figura 39 – Grafo de Filtros para sintonização ISDB-T.	74
Figura 40 – Classe para dos Dispositivos de entrada.	75
Figura 41 – Classe <i>GingaNCLPluginInstance</i> .	76
Figura 42 – Classe <i>GingaNCLPluginObject</i> associada ao objeto <i>scriptable</i> .	78
Figura 43 – Classe <i>FormatterMediator</i> .	78
Figura 44 – Classe <i>FormatterMediator</i> como Interface externa do <i>plugin</i> .	79
Figura 45 – Diagrama de <i>back buffering</i> . Retirada e adaptada de (MICROSOFT, 2010).	90
Figura 46 – (A) Vértices desconectados e (B) Polígono.	91
Figura 47 – Mapeamento de textura. Retirada de (THORN, 2005, p. 131).	92
Figura 48 – Arquitetura do DirectShow. Retirada e adaptada de (MICROSOFT, 2010).	94
Figura 49 – Grafo de Filtros para reprodução de um arquivo no formato WAV. (a) Filtro Fonte, (b) Filtro de Transformação e (c) Filtro de Renderização. Obtido através de (MONOGRAM, 2009).	95
Figura 50 – Componentes do filtro de renderização VMR. Retirada e adaptada de (MICROSOFT, 2010).	97

Lista de tabelas

Tabela 1 – Tabela Comparativa entre as soluções apresentadas e a solução proposta. 32