
O Aprimoramento do Sign WebMessage como Base para o Desenvolvimento da SWService: uma Biblioteca para a Escrita da Libras na Internet Baseada em Web Services

Artigo Completo

Vinícius Costa de Souza e Sérgio Crespo Coelho da Silva Pinto

Programa Interdisciplinar de Pós-Graduação em Computação Aplicada (PIPICA)

Universidade do Vale do Rio dos Sinos (UNISINOS)

Av. Unisinos, nº 950 - bairro: Cristo Rei - São Leopoldo/RS - Brasil

CEP.: 93.022-000 - Caixa Postal: 275

Telefone: +55 (51) 591-1122 - Fax: +55 (51) 590-8305

e-mail: [vinicius, crespo]@exatas.unisinos.br

Resumo

Um dos grandes problemas enfrentados pelos surdos é não poder se expressar através da escrita de sua própria língua, a língua de sinais - LS. Em função disso, sua comunicação se estabelece quase que exclusivamente de forma presencial, de forma que os surdos não podem usufruir totalmente das novas tecnologias como, por exemplo, o correio eletrônico e a Internet. Por isso, este artigo apresenta um aprimoramento do Sign WebMessage, uma ferramenta para comunicação em LS na Internet, como base para o desenvolvimento de uma camada de software chamada SWService - SignWriting Web Service. Tal camada utilizará a tecnologia de Web Services com o objetivo de fornecer os recursos necessários para que softwares baseados na Web possam utilizar o sistema para escrita das línguas de sinais, SignWriting, de forma rápida e eficiente, sem a necessidade de implementar ou instalar localmente.

O Aprimoramento do Sign WebMessage como Base para o Desenvolvimento da SWService: uma Biblioteca para a Escrita da Libras na Internet Baseada em Web Services

1. Introdução

A popularização do computador e sua utilização em diversas áreas é fato inquestionável, assim como o uso da informática na educação e o crescente desenvolvimento de softwares educacionais. Entretanto, muitas pessoas, pelos mais variados motivos, não têm acesso às novas tecnologias da informação. Vivemos em um processo de grandes mudanças sócio-culturais, onde a difusão de novas ferramentas que possibilitam transferência de conhecimento se torna extremamente importante, a fim de preparar as pessoas para o exercício da cidadania e para qualificação profissional.

Um dos problemas existentes é o fato de a maioria dos softwares serem desenvolvidos sem levar em consideração a grande diversidade de usuários que os mesmos possam ter, o que vem a limitar a utilização dessas ferramentas por portadores de necessidades especiais [2]. Segundo o último censo demográfico realizado pelo IBGE em 2002, cerca de 14,5% da população brasileira apresenta algum tipo de deficiência, sendo que 3,38% dos brasileiros possuem algum nível de deficiência auditiva [19].

Uma das grandes dificuldades dos surdos é comunicar-se na forma escrita, visto que não existe, ainda hoje, uma forma de registro escrito das línguas de sinais. Entretanto, nos últimos anos, o SignWriting [22], um sistema para escrita de línguas de sinais, vem despertando interesse de lingüistas, pesquisadores, professores e surdos de vários países [12].

Devido a esse avanço, já existem alguns softwares como o Sign WebMessage [17, 18], o SWEdit [25] e o SignWriter [20, 21], entre outros, que fazem uso do SignWriting. Porém, a tarefa para disponibilizar a escrita em SignWriting ainda é bastante complexa, dispendiosa e demanda muito tempo e recursos. Não existe uma solução padrão confiável e documentada que possa ser facilmente reutilizada de forma a diminuir o tempo e a complexidade para o desenvolvimento de novos softwares que utilizem a escrita de línguas de sinais.

Por isso, este trabalho propõe o desenvolvimento de uma camada de software chamada *SWService*, cujo objetivo principal é facilitar o desenvolvimento de novas ferramentas computacionais para surdos, que permitam a escrita em língua de sinais.

2. Língua de Sinais

A comunicação é uma necessidade humana, e as linguagens oral e escrita são as formas mais comuns de comunicação [15]. A língua utilizada por um indivíduo depende do grupo em que está inserido. Para os ouvintes, a comunicação se estabelece em termos oral-auditivos. No entanto, para os surdos se estabelece em termos gestual-visuais, em que gestual significa o conjunto de elementos lingüísticos manuais, corporais e faciais necessários para a articulação e a significação cultural do sinal [16].

As línguas de sinais são utilizadas pela maioria das pessoas surdas. No Brasil, existem duas línguas de sinais: a língua brasileira de sinais Kaapor – LSKB, utilizada pelos índios da tribo Kaapor, cuja maioria são surdos, e a Língua Brasileira de Sinais - Libras, utilizada nos centros urbanos. A Língua Portuguesa é considerada uma segunda língua para os surdos [3].

As línguas de sinais são dotadas de toda a complexidade e utilidade encontradas nas línguas orais e, assim como elas, possuem gramáticas próprias, com regras específicas em seus níveis lingüísticos fonológico, morfológico e sintático. Entretanto, somente a partir de 24 de

abril de 2002 a Libras foi reconhecida como meio legal de comunicação e expressão das comunidades surdas do Brasil, de acordo com a lei Nº. 10.436.

2.1. Sistema de escrita visual direta de sinais SignWriting

As línguas de sinais, ao contrário das línguas orais, não desenvolveram uma representação escrita que tenha sido amplamente divulgada e aceita pelas comunidades surdas [11]. Como consequência, os usuários dessas línguas não estão capacitados a lerem e produzirem textos em suas línguas, tendo que recorrer para isso à representação gráfica de uma língua oral, o que é muito difícil para eles pois não dispõem da audição para aprendê-las. Entretanto, nos últimos anos o sistema SignWriting vem sendo difundido e utilizado por várias pessoas em todo o mundo [22].

O SignWriting é um sistema de representação gráfica das línguas de sinais que permite, através de símbolos visuais representar as configurações das mãos, seus movimentos, as expressões faciais e os deslocamentos corporais. Este sistema tem um caráter gráfico-esquemático intuitivo que funciona como um sistema de escrita alfabética, em que as unidades básicas representam unidades gestuais fundamentais, suas propriedades e relações [22].

O sistema SignWriting é definido por três estruturas básicas: posição de mão, movimentos e contato. Além destas, existem símbolos para expressões faciais, pontos de articulação, dentre outros [22]. A figura 1 apresenta ilustrações de alguns sinais seguidas de sua representação em SignWriting.



Figura 1. Exemplos de sinais em SignWriting

3. SignWriting Markup Language – SWML

A SWML é uma linguagem padrão *Extensible Markup Language* (XML), pois provê um formato de troca de dados flexível e independente de software que pode ser facilmente analisado sintaticamente. Foi desenvolvida por Costa e Dimuro [6] com o objetivo de tornar possível e viável a realização de operações de troca, armazenamento e processamento de textos escritos em SignWriting por diversas aplicações.

A estrutura dos arquivos SWML descreve, por intermédio de suas *tags* e respectivos atributos, todas as características necessárias para que qualquer software, apto a processar o padrão XML, reproduza com fidelidade os sinais criados com base nos símbolos do sistema SignWriting.

Basicamente, um arquivo SWML descreve quais símbolos compõem cada sinal, o posicionamento de cada símbolo no espaço e as transformações as quais os símbolos devem ser submetidos. Para cada sinal existe um elemento denominado <signbox>, o qual concentra um grupo de símbolos <symb> que juntos compõem um sinal. Para cada símbolo <symb>, em um sinal <signbox>, existe um conjunto de informações definidas pelos elementos *category*, *group*, *symbnum*, *variation*, *fill* e *rotation* que identificam o símbolo base e a quais transformações ele deve ser submetido para compor determinado sinal.

Além disso, existe um grupo de atributos *x*, *y*, *x-flop*, *y-flop* e *colour* identificando as coordenadas de cada símbolo, se o símbolo possui rotação em x ou em y e sua cor,

respectivamente. Desta forma, as características de qualquer sinal podem ser extraídas de um arquivo SWML e a imagem do sinal pode ser reconstruída sempre que necessário [7].

4. Web Services

Web Services são componentes de software que independem de implementação ou de plataforma e podem ser descritos, publicados e invocados sobre uma rede, geralmente a Web, por meio de mensagens padrão XML [14]. Com o uso de *Web Services* é possível estabelecer integração entre aplicações baseadas na Internet, de forma padronizada, utilizando para isso padrões abertos incluindo XML, *Simple Object Access Protocol* (SOAP), *Web Services Description Language* (WSDL) e *Universal Description, Discovery e Integration specification* (UDDI). As mensagens são estruturadas com XML, o SOAP encarrega-se de transferir as mensagens, o WSDL descreve os serviços disponíveis e o UDDI os lista [5].

A arquitetura *Web Services*, apresentada na figura 2, está baseada em um Provedor de Serviços, um Solicitante de Serviços e um Registro de Serviços. O provedor de serviços é responsável por disponibilizar os serviços e armazenar sua descrição, através da WSDL, a qual contém detalhes de interface, operações dos serviços e mensagens de entrada e saída para as operações. Após o recebimento da descrição de um serviço, o provedor publica a descrição do mesmo, em WSDL, em um Registro de serviços. O solicitante de serviços é uma aplicação que invoca ou inicializa uma interação com o serviço, a qual pode ser um navegador Web ou outra aplicação qualquer como um outro *Web Service*. O registro de serviços é o local onde os servidores publicam seus serviços e onde os solicitantes procuram por serviços [8].

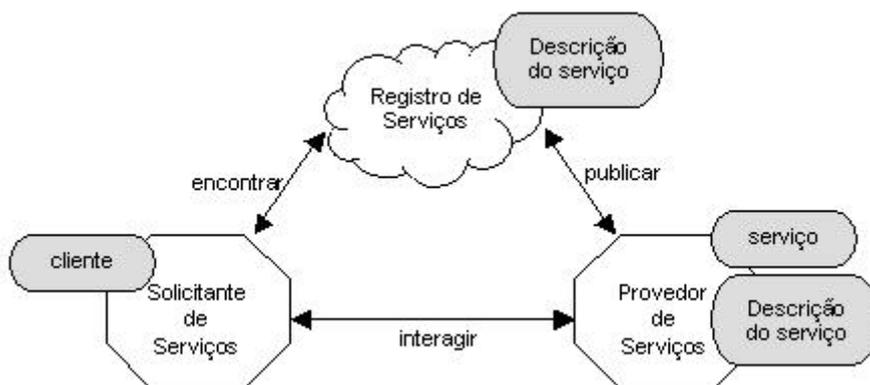


Figura 2. Arquitetura dos *Web Services* [8]

5. Sign WebMessage

O *Sign WebMessage* (<http://www.inf.unisinos.br/swm>) é um software para comunicação assíncrona na Web, através do qual pode-se interagir utilizando tanto a escrita da língua portuguesa quanto a da escrita da Libras. Nas mensagens, os sinais podem ser visualizados em *SignWriting* e, opcionalmente, seus significados em português, o que proporciona uma forma de aprendizagem de ambas as línguas [17]. Quanto ao desenvolvimento do protótipo, foi utilizado PHP para programação, banco de dados PostgreSQL, servidor web Apache e sistema operacional Red Hat Linux, pois tratam-se de ferramentas não proprietárias e de ampla utilização na Internet [18].

A figura 3 apresenta a interface do Sign WebMessage.

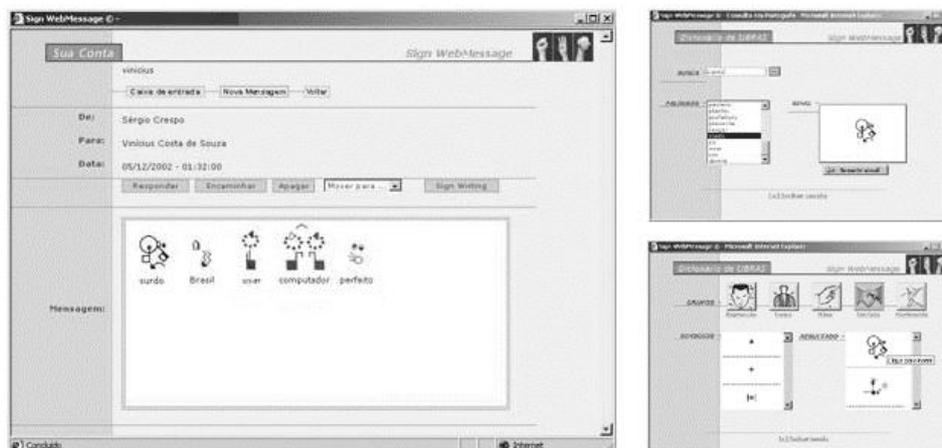


Figura 3. Sign WebMessage

6. Estudo de Softwares

Diversas iniciativas estão sendo desenvolvidas para difundir as línguas de sinais, inclusive através do desenvolvimento de software, como forma de apoiar a inclusão digital e social das pessoas surdas. Porém, uma das dificuldades para o cumprimento deste objetivo ocorre no momento de se registrar as línguas de sinais na forma escrita e, por isso, existem poucos ambientes que as utilizam como idioma principal em suas funcionalidades e interfaces.

Assim, foi realizado um estudo de algumas ferramentas computacionais direcionadas ao público surdo e que utilizam o sistema de escrita SignWriting a fim de se identificar suas características e diferenças. Os sistemas estudados foram os editores de texto para escrita em língua de sinais SignWriter [20, 23], SignEd [4] e SWedit [25]; o sistema SignSim [4], para tradução entre português e Libras; o SignBank [21] que é um banco de dados relacional para criação de dicionários em SignWriting; o SignTalk [4], que é uma ferramenta de chat que permite comunicação síncrona em Libras ou português e o Sign WebMessage [17, 18].

É possível perceber pontos em comum nos sistemas estudados, bem como características que não estão presentes em todos eles, ou que se apresentam em alguns de formas diferentes. A tabela 1 apresenta uma análise comparativa entre os softwares.

Tabela 1. Estudo comparativo entre softwares que utilizam o SignWriting

Software / Recurso	SignWriter	SignEd SignSim SignTalk	SWedit	SignBank	Sign WebMessage
Cria sinais	x	x	x	x	
Edita sinais	x	x	x	x	
Possui dicionário de sinais	x	x	x	x	x
Consulta o dicionário a partir da língua de sinais		x		x	x
Consulta o dicionário a partir da língua oral	x	x	x	x	x
Importa e exporta sinais em SWML			x		

7. Aprimoramento do Sign WebMessage

Como visto no estudo apresentado na sessão anterior, existem alguns recursos, essenciais para a manipulação das línguas de sinais, que ainda não estão disponíveis no Sign

WebMessage, software no qual a biblioteca SWService será baseada. Tais recursos são os necessários para criação e edição de sinais e para importação e exportação de arquivos no formato SWML, conforme pode ser observado na tabela 1. Assim, implementou-se essas novas funcionalidades no Sign WebMessage de forma que para o desenvolvimento da SWService já se tenha experiência sobre as mesmas, garantindo assim maior qualidade da biblioteca.

A seguir, é apresentada uma descrição de como as funcionalidades foram implementadas. A implementação foi realizada em duas etapas. Primeiramente desenvolveu-se um módulo através do qual os usuários podem criar novos sinais e editar sinais já existentes, os quais são armazenados em arquivos SWML. A segunda etapa foi desenvolver um módulo que permita ao Sign WebMessage importar um arquivo qualquer padrão SWML e analisá-lo sintaticamente de forma a reproduzir fielmente os sinais especificados no mesmo.

7.1 Criação e edição de sinais no Sign WebMessage

A figura 4 apresenta o módulo para criação de sinais, onde o usuário deve primeiramente selecionar os símbolos que irão compor o sinal, os quais são exibidos na área de construção. Após, o usuário pode movê-los (à esquerda, à direita, para cima ou para baixo) ou até mesmo apagá-los, em caso de erro, até compor o sinal desejado. Para possibilitar a manipulação dos símbolos (imagens GIF) em um ambiente Web, foram desenvolvidas funções em JavaScript que são chamadas cada vez que ocorre um evento *onClick* sobre os botões da interface. Essas funções manipulam as seguintes propriedades da página HTML: *left layer*, *top layer*, *visibility layer*, *border image* e *form field*

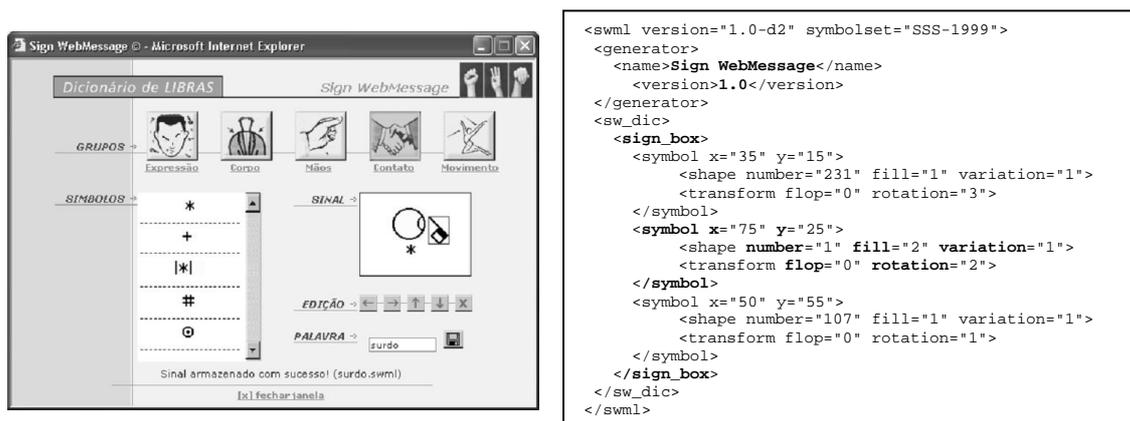


Figura 4. Criação de sinal e arquivo SWML correspondente

Na implementação destes recursos foram utilizados, também, *layers* que possibilitam a criação de áreas dentro de páginas web com largura e altura pré-definidas e em uma posição exata, através de coordenadas dos eixos x e y. Os *layers* podem conter textos, imagens, formulários ou qualquer outro objeto utilizado em HTML, até mesmo outros *layers*. Além disso, fornecem a possibilidade de um controle fino sobre o seu posicionamento em relação à página e possuem as seguintes propriedades: *left* e *top* (coordenadas x e y, respectivamente), *z-index* que permite definir a sobreposição dos *layers* e *visibility* que define se um *layer* é visível ou não.

Para realizar a análise do arquivo SWML utilizou-se a *Simple API for XML – SAX*, uma API leve e de fácil utilização, disponível no PHP, que permite analisar sintaticamente arquivos padrão XML. Outro recurso utilizado foi a biblioteca *gd*, desenvolvida em C e atualmente com interface para PHP, a qual provê funções para criação e manipulação de imagens. A figura 5 apresenta o algoritmo, implementado em PHP, para análise do arquivo SWML e a forma de exibição do sinal, através dos *layers*.

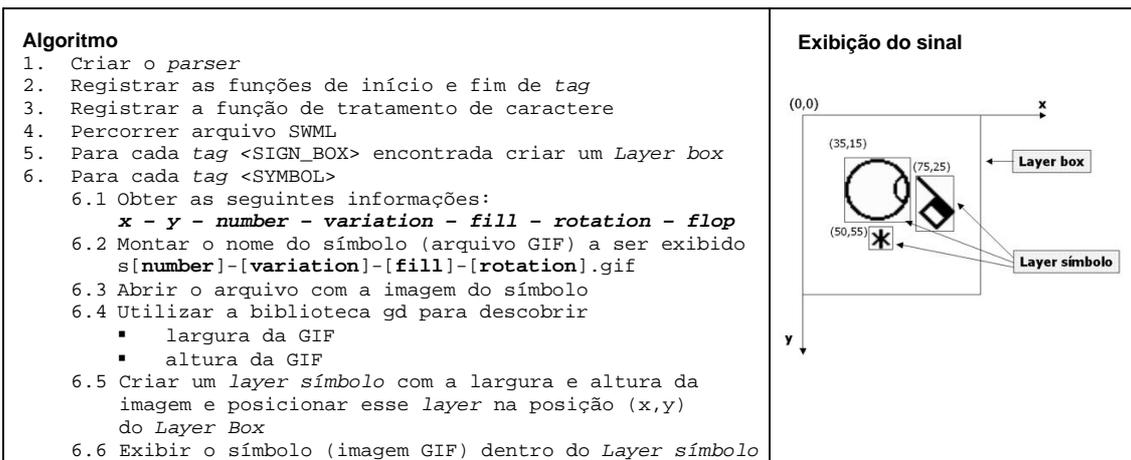


Figura 5. Algoritmo para análise do SWML e exibição do sinal através de layers

7.2 Redução dos símbolos do SignWriting

Ainda como resultado do estudo dos softwares apresentados na sessão 6, constatou-se que uma das dificuldades na utilização das ferramentas se dá pela enorme quantidade de símbolos existentes no sistema de escrita SignWriting, atualmente estimados em 16.000 símbolos. Com tantos símbolos, torna-se difícil disponibilizá-los nas interfaces de modo que os usuários tenham acesso aos símbolos desejados de forma rápida e eficiente. Além disso, como existem muitos símbolos parecidos, as interfaces acabam ficando visualmente poluídas, como por exemplo no SWEdit apresentado na figura 6, o que causa confusão e dificuldade para os usuários.

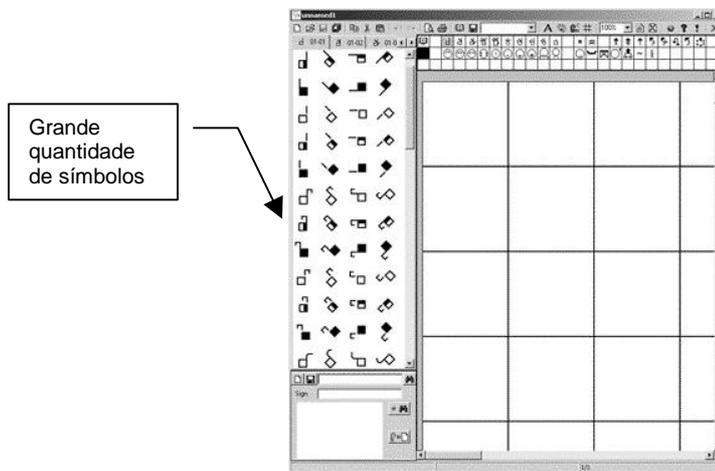


Figura 6. Interface do SWEdit

Por isso, foi realizado um estudo sobre os símbolos do sistema SignWriting [24], no qual identificou-se aproximadamente 250 símbolos base dos quais todos os outros derivam. Assim, com objetivo de minimizar o problema da grande quantidade de símbolos, também no desenvolvimento da SWSservice, realizou-se um estudo preliminar com os símbolos derivados do primeiro símbolo base, os símbolos pertencentes ao grupo de mão número um. Para este grupo de mão, assim como para todos os outros, existem 95 variações.

Com o estudo, pode ser observado que as variações do símbolo base são decorrentes de rotações da mão do sinalizador em torno do próprio eixo (palma, lado e dorso), mão na horizontal ou na vertical, oito rotações com as possibilidades de direcionamento da mão e

espelhamento destas. A tabela 2 apresenta as variações iniciais que combinadas dão origem às 95 variações existentes para este símbolo base.

Tabela 2. Símbolo base do grupo de mão 1 e suas variações iniciais

Símbolo base	Rotação em torno do próprio eixo			Orientação da mão	
	Palma	Lado	Dorso	Vertical	Horizontal
					

Com base nesta análise, foram desenvolvidas funções no Sign WebMessage que permitam aos usuários partir do símbolo base e chegar em qualquer uma das suas 95 variações possíveis. Assim, é possível diminuir consideravelmente o número de símbolos na interface e tornar o processo de criação e edição de sinais mais rápido e eficiente. Para isso, foram incluídos mais quatro botões na área de edição dos símbolos: para rotação nas oito direções possíveis (1), para espelhamento (2), para mão na horizontal ou vertical (3) e para rotação em torno do próprio eixo (palma, lado e dorso) (4). A nova interface para criação e edição dos sinais é apresentada na figura 7.

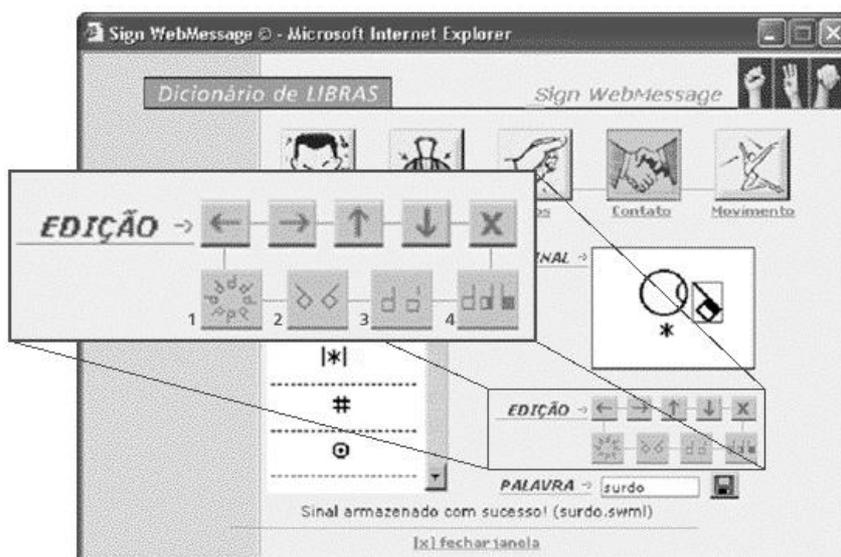


Figura 7. Nova interface para criação e edição de sinais do Sign WebMessage

Dessa forma, para se chegar no símbolo de mão destacado na figura 7, o usuário deverá inserir o símbolo base do grupo 1 de mão, clicar uma vez no botão para rotação em torno do próprio eixo (botão 4) e uma vez no botão para mudança de direção da mão (botão 1).

8. SWService

Como já mencionado, a SWService trata-se de uma camada de software que utilizará a tecnologia de Web Services com o objetivo de fornecer os recursos necessários para que softwares baseados na Web possam utilizar o sistema SignWriting, de forma rápida e eficiente, sem a necessidade de implementar ou instalar localmente.

A figura 8 apresenta a arquitetura que possibilitará a integração e interoperabilidade entre a SWService, as aplicações Web e os usuários finais. Como ilustrado na arquitetura proposta, os usuários poderão utilizar aplicações Web (chats, webmails, fóruns, dentre outros) as quais estarão fazendo uso dos serviços oferecidos pela biblioteca SWService (criação, edição

e pesquisa de sinais e criação, edição e leitura de textos escritos em língua de sinais) sem a necessidade de implementação destes recursos.

A interoperabilidade será garantida pelo uso da linguagem SWML, através da qual as características dos sinais são transmitidas em formato padrão XML. Dessa forma, as aplicações Web que utilizarem os serviços da SWSservice poderão ser desenvolvidas em qualquer linguagem e plataforma, independentemente da linguagem e plataforma utilizadas na implementação da SWSservice.

Além disso, a arquitetura prevê a disponibilização de um dicionário público bilíngüe (língua de sinais – língua oral), o qual estará disponível para consultas e será compartilhado por todas as aplicações que fizerem uso da biblioteca. Além do dicionário público, é prevista a possibilidade de criação de dicionários pessoais nas aplicações, os quais armazenarão os sinais criados pelos próprios usuários.

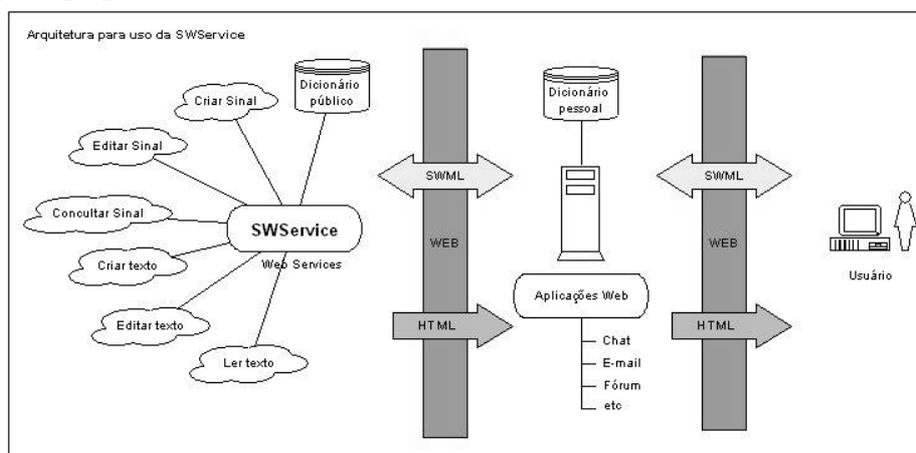


Figura 8. Arquitetura de uso da biblioteca de serviços SWSservice

9. Considerações finais

Através dessa experiência, pôde-se constatar que através da linguagem SWML foi possível melhorar a forma de armazenamento e processamento de sinais no Sign WebMessage, pois foi possível criar um novo sinal, em ambiente exclusivamente web, e gerar um arquivo SWML com as características do mesmo. Posteriormente, conseguiu-se reproduzir este sinal com base apenas na análise do arquivo SWML o que garantirá a interoperabilidade desejada na SWSservice. Além disso, pode-se confirmar que é possível reduzir consideravelmente o número de símbolos do sistema SignWriting nas interfaces e, assim, tornar mais ágil e fácil o uso de softwares que possibilitam a escrita em SignWriting.

Assim, pode-se concluir que através da arquitetura proposta, será possível disponibilizar uma camada de software, que forneça os recursos necessários para utilização das línguas de sinais em ambientes web, para que outras aplicações possam fazer uso de forma rápida, eficiente e independente de linguagem ou plataforma devido ao uso da tecnologia de *Web Services*.

Pretende-se dar continuidade ao trabalho através de um estudo sobre as plataformas para *Web Services*, traçando um paralelo entre elas, a fim de escolher a que melhor atenderá as necessidades da biblioteca SWSservice. Posteriormente, pretende-se implementar a biblioteca utilizando-se a plataforma escolhida, descrever os serviços através da linguagem WSDL e tornar os serviços públicos e disponíveis para que possam ser utilizados por quaisquer outras aplicações. Após a implantação e disponibilização da SWSservice, pretende-se avaliar o potencial da biblioteca através de um estudo de caso, onde a mesma será utilizada no desenvolvimento de uma ferramenta de Fórum que possibilitará a escrita das mensagens em Libras ou português.

10. Referencias bibliográficas

1. Booth, David *et. al.* Web Service Architecture. W3C Working Draft. August 2003. Disponível em <http://www.w3.org/TR/2003/WD-ws-arch-20030808>
2. Campos, Márcia de Borba e Silveira, Milene Selbach. Tecnologias para a Educação Especial. In: IV Congresso Ibero-Americano de Informática na Educação - RIBIE, Brasília 1998.
3. Campos, Márcia Borba *et. al.* SIGNSIM: uma ferramenta para auxílio à aprendizagem da língua brasileira de sinais. In: V Congresso Ibero-Americano de informática na Educação – RIBIE – Chile, 2000.
4. Campos, Márcia de Borba. Ambiente Telemático de interação e comunicação para suporte à educação bilíngüe de surdos. Tese de doutorado apresentada ao Programa de Pós-Graduação em Informática na Educação da UFRGS. Porto Alegre, 2001.
5. Chung, Jen-Yao *et. al.* Web Services Computing: Advancing Software Interoperability. Publishe by the IEEE Computer Society. October 2003.
6. Costa, Antônio Rocha e Dimuro, G. P. “A SignWriting-Based Approach to Sign Language Processing”, In: GW2001 – Gesture Workshop 2001. City Unversity, Londres, 2001.
7. Costa, Antônio Rocha e Dimuro, G. P. “Supporting Deaf Sign Languages in Written Form on the Web”, In: 10th World Wide Web Conference, Hong Kong, Home Page of Web and Society Track, <http://www10.org/cdrom/posters/frame.html.Processing>”, Written Form on the Web. 2001
8. Ferris, Christopher and Farrell, Joel. What Are Web Services? Communications of the ACM. Vol.46 n°. 6. June 2003.
9. Hansen, Roseli. Glue Script: Uma Linguagem Específica de Domínio para Composição de *Web Services*. Dissertação de Mestrado. Programa Interdisciplinar de Pós-Graduação em Computação Aplicada - PIPCA. Universidade do Vale do Rio dos Sinos – Unisinos. São Leopoldo, fevereiro de 2003.
10. Hansen, Roseli e Pinto, Sérgio Crespo C. S. Construindo Ambientes de Educação Baseada na Web Através de Web Services Educacionais In: XIV Simpósio Brasileiro de Informática na Educação – SBIE. Rio de Janeiro, novembro de 2003
11. Macedo, Daniela Remião. *Sign Dic*: Um ambiente multimídia para a criação e consulta de dicionários bilíngües de línguas de sinais e línguas orais. Dissertação de mestrado - PUCRS. Porto Alegre, 1999
12. Macedo, Daniela R. e Costa, Antônio C. R. *Sign Dic*: um ambiente para geração de dicionários bilíngües Língua de Sinais – Língua Oral Língua Oral – Língua de Sinais. In: V Congresso de Educação Bilíngüe para Surdos. UFRGS, Porto Alegre, 1999.
13. Newcomer, Eirc. Understanding Web Services. Independent Technology Guides. Dadid Chappell, Series Editor. 2002
14. Pinto, Sérgio Crespo C. S *et. al.* *Web Services: An Architectural Overview*. In: First International Seminar on Advanced Research in E-Business – ERB. PUC-RIO. November 2002.
15. Santarosa, Lucila Maria. Telemática y la inclusión virtual y social de personas con necesidades especiales: un espacio posible en la Internet. in: V Congresso Ibero-Americano de Informática na Educação- RIBIE: Viñadelmar, 2000.

-
16. Skliar, Carlos. A SURDEZ, um olhar sobre as diferenças. Editora Mediação, 2ª Edição. Porto Alegre, 2001.
 17. Souza, Vinícius Costa e Pinto, Sérgio Crespo C. da Silva. *Sign WebMessage*: um ambiente para comunicação via *Web* baseado na escrita de LIBRAS. In: III Congresso Ibero-americano de Informática na Educação Especial. Fortaleza, agosto de 2002.
 18. Souza, Vinícius Costa e Pinto, Sérgio Crespo C. S. *Sign WebMessage*: uma ferramenta para comunicação via web através da Língua Brasileira de Sinais – Libras. In: XIV Simpósio Brasileiro de Informática na Educação – SBIE 2003. Rio de Janeiro, novembro de 2003.
 19. Souza, Vinícius Costa, Aguiar, Márcia Rafaeli e Pinto, Sérgio Crespo C. S. Desafios e Resultados de uma experiência na Inclusão Digital de Surdos. In: XIV Simpósio Brasileiro de Informática na Educação. Rio de Janeiro, novembro de 2003.
 20. Sutton, Valerie. SignWriter Computer Program - Instruction Manual (version 4.4). La Jolla, Deaf Action Committee for SignWriting, 2001.
 21. Sutton, Valerie. SignBank 2002 – The SignWriting Online Dictionary Center for Sutton Movement Writing, Inc. A US nonprofit educational organization. La Jolla, 2002.
 22. Sutton, Valerie. Lessons in SignWriting – Textbook and Workbook, Third Edition, La Jolla, Deaf Action Committee for SignWriting, 2003.
 23. Sutton, Valerie. SignWriter Java – Sign Language Processor. La Jolla, Deaf Action Committee for SignWriting, 2003.
 24. Sutton, Valerie. Sutton’s Sign-Symbol-Sequence. La Jolla, Deaf Action Committee for SignWriting, 2004.
 25. Torchelsen, Rafael P. e Costa, Antônio Carlos R. Editor para Língua de Sinais Escritas em SignWriting. In: XIII Simpósio Brasileiro de Informática na Educação – SBIE 2002. São Leopoldo, novembro de 2002.