

Construindo Ambientes de Educação Baseada na Web Através de Web Services Educacionais

Roseli P. Hansen¹, Sérgio Crespo, S. C. Pinto²

¹Centro Universitário La Salle – UNILASALLE
Av. Victor Barreto, 2228 – 92010-000 Canoas, RS

²Universidade do Vale do Rio dos Sinos – UNISINOS
Programa Interdisciplinar de Pós-Graduação em Computação Aplicada – PIPCA
Av. Unisinos, 950 – 93022-000 São Leopoldo, RS

rhansen@brturbo.com, crespo@exatas.unisinos.br

Resumo. *Este trabalho faz um overview da tecnologia de Web Services e do metadado LOM. Apresenta um mecanismo que permite a integração de Web Services e recursos educacionais. Esta integração é feita através de uma linguagem de composição chamada GlueScript.*

Palavras-Chave: Web Services, Metadados Educacionais, Educação a Distância, Linguagem de composição

Abstract. *This work makes an overview of the technology of Web Services and of the LOM metadata. It introduces a mechanism that allows Web Services's integration with educational resources. This integration is made through of a language of composition called GlueScript.*

Key words: Web Services, Educational Metadata, Web-Based Education, Composition Language

1. Introdução

A educação baseada na Web utiliza a WWW como meio para a publicação de material didático, aplicação de tutoriais, aplicação de provas e testes além da comunicação com estudantes. Também compreende o processo de uso da Web como veículo de comunicação para a apresentação de aulas à distância (conferência multimídia). As tecnologias de educação baseada na Web, estão em pleno uso, utilizando uma série de ferramentas que auxiliam os processos de cooperação, coordenação e comunicação [Lucena 2000, Crespo et. al. 1998].

Com o crescimento da Internet surgem novas tecnologias para disponibilização de serviços. Juntamente com as novas tecnologias surge a necessidade de tornar estes serviços interoperáveis e reutilizáveis.

Os *Web Services* apresentam uma estrutura arquitetural que permite a comunicação entre aplicações. Um serviço pode ser invocado remotamente ou ser utilizado para compor um novo serviço juntamente com outros serviços. A tecnologia de *Web Services* está baseada em *eXtensible Markup Language* (XML) [Bray 2000], e permite invocar ou reutilizar um serviço sem a necessidade de conhecer a plataforma ou linguagem de programação usada na sua construção.

Por outro lado, os sistemas de *e-learning* estão direcionando seus esforços na tentativa de obter a interoperabilidade dos recursos entre diferentes sistemas educacionais. Para tornar esta interoperabilidade possível, torna-se necessário a padronização dos metadados [Crespo et. al. 1998]. É apresentado neste trabalho o padrão de metadado *Learning Object Metadata* (LOM) [IEEE LTSC 2002].

As arquiteturas de *e-learning* existentes têm uma definição bastante precisa quanto a descrição de recursos educacionais. Já a arquitetura de *Web Services* apresenta uma estrutura bem definida para permitir busca, acesso e principalmente reuso de serviços na WEB.

Foi desenvolvida uma arquitetura de integração, que utiliza a linguagem de composição chamada GlueScript, para permitir que recursos disponibilizados por sistemas educacionais possam ser reutilizados juntamente com serviços acessados através da arquitetura de *Web Services*.

Este artigo apresenta além das tecnologias de *Web Services* e LOM-XML, a arquitetura de integração, a linguagem de composição e um estudo de caso utilizando a GlueScript.

Na próxima seção será apresentado um overview de *Web Services* para melhor compreensão desta tecnologia.

2. Um Overview de Web Services

Web Services são aplicações modulares que podem ser descritas, publicadas e invocadas sobre uma rede, geralmente a Web. Ou seja, é uma interface que descreve uma coleção de operações que são acessíveis pela rede através de mensagens em formato XML padronizadas. Permitem uma integração de serviços de maneira rápida e eficiente [Hansen et. al. 2002, Kreger 2001].

Um *Web Service* é um componente de software independente de implementação e plataforma. É descrito utilizando uma linguagem de descrição de serviços, publicado em um registro e descoberto através de um mecanismo padrão. Pode também ser invocado a partir de uma *Application Program Interface* (API) através da rede e ser composto juntamente com outros serviços.

2.1. Arquitetura de Web Services

Na arquitetura *Web Service* a descrição de um serviço cobre todos os detalhes necessários para que haja interação entre serviços, incluindo o formato das mensagens, protocolos de transporte e localização.

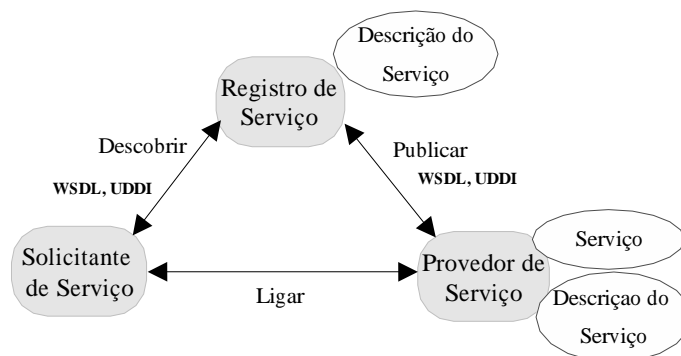


Figura 1: *Web Services* Papéis, Operações e Artefatos [Kreger 2001]

A arquitetura de *Web Services* apresentada na Figura 1, está baseada nas interações entre um **Provedor de Serviço** para disponibilizar um serviço, um **Solicitante de Serviço** que faz uso do serviço e um **Registro de Serviço** onde os provedores publicam as descrições dos serviços. A arquitetura de *Web Services* é composta por padrões XML assim como: a *Web Services Description Language* (WSDL) para descrever os serviços [Cristensen et. al. 2001]; o *Simple Object Access Protocol* (SOAP) utilizado para publicar, localizar e invocar um *Web Service* em um registro UDDI [Clements 2001, Box et. al. 2000, Gudgin et. al. 2001]; o *Universal Description, Discovery and Integration* (UDDI), um registro que é acessado por clientes para localizar os serviços de que necessitem [Ehnebuske et. al. 2001, ARIBA INC 2000, Rajaram 2001].

Até o momento foi apresentada a tecnologia de *Web Services* bem como os padrões utilizados. Porém torna-se relevante conceituar brevemente metadados na próxima seção.

3. O que são metadados?

Os metadados são “dados sobre dados” ou “informações sobre informações”. São um conjunto de palavras, frases ou sentenças que resumem ou descrevem o conteúdo de um *site* WEB, uma página WEB individual ou um recurso computacional com o objetivo de beneficiar o trabalho de agentes de busca [BECTa 2001, Babu 2001].

Os metadados são descritos em formato XML, formando uma estrutura de dados que descreve as características de um recurso. Desta forma, podem ser utilizados para descrever conteúdos e estruturas além de fornecer informação sobre acessibilidade, organização e relacionamento entre os dados [Robson 2000].

Os sistemas de *e-learning* podem obter benefícios com uso de metadados, porque estes propiciam o desenvolvimento de aplicações interoperáveis. A troca de recursos de aprendizagem entre os sistemas pode ser feita, mediante uma descrição formal das suas interfaces.

Alguns padrões de metadados para *e-learning* estão emergindo, dentre eles pode-se citar: o Dublin Core [Dublin Core 2002] e o IEEE-LOM [IEEE LTSC 2002].

Um dos principais contribuidores para a padronização dos metadados educacionais é o IEEE-*Learning Technology Standards Committee* (LSTC). Em 1998 os projetos IMS [IMS 2002] e ARIADNE [ARIADNE 2002], submeteram uma proposta de padronização que resultou no documento base IEEE-LOM, que é apresentado brevemente na próxima seção.

3.1. Learning Object Metadata (LOM)

Um ambiente de *e-learning* pode ser constituído com recursos totalmente educacionais, mas também pode conter serviços que não são de um domínio específico como por exemplo um *Webmail*.

Desta forma, a integração entre *Web Services* e metadados educacionais em ambientes de *e-learning* poderá ser útil, na medida que uma nova aplicação poderá ser composta a partir de um *Web Service*, de um recurso educacional ou por ambos. Para tanto, LOM-XML torna-se uma importante ferramenta.

LOM especifica um esquema conceitual de dados o qual define a estrutura de uma instância de um metadado para objetos de aprendizagem. Neste padrão, um objeto de aprendizagem é definido como uma entidade que pode ter um formato digital ou um não. Este padrão pode ser utilizado em sistemas de aprendizado, educação e treinamento. Uma instância de um metadado descreve características relevantes do objeto de aprendizagem enfocando a sua semântica, uso e pressupostos técnicos e pedagógicos de como deve ser melhor utilizado [BECTa 2001, IEEE LTSC 2002].

LOM é caracterizado por permitir a definição de blocos independentes, compartilhamento e troca de conteúdo educacional. Estes blocos podem conter referências para outros objetos e podem ser combinados ou seqüenciados para formar grandes unidades educacionais [IEEE LTSC 2002].

A preocupação dos desenvolvedores de ambientes de *e-learning* está centrada na interoperabilidade entre os recursos de diferentes sistemas através do uso de metadados educacionais padrões. Os metadados padrões podem ser descritos no padrão XML, permitindo que estes sejam facilmente manipulados.

Para obter uma interoperabilidade das tecnologias de *WebServices* e LOM, devemos entender como as arquiteturas de ambos estão projetadas.

A seguir, será apresentada uma proposta de arquitetura que visa promover a interoperabilidade entre diferentes sistemas de *e-learning*, além de um alto grau de reutilização por meio da integração de *Web Services* e LOM-XML. A interoperabilidade, composição e reuso em ponto grande será atingida pela definição da arquitetura e da linguagem GlueScript.

4. Arquitetura de Integração

Nas seções anteriores foram apresentados aspectos relacionados a tecnologia de *Web Services*. A arquitetura de *Web Services* não está voltada a nenhum domínio específico de aplicações. Observa-se porém que seu uso está intensificado na área de comércio eletrônico.

Através dos metadados educacionais torna-se possível descrever de maneira bastante detalhada qualquer tipo de recurso educacional.

A arquitetura de Integração é composta pela arquitetura de *Web Services* apresentada na Seção 2.1 e por uma arquitetura genérica extraída a partir de estudos realizados em algumas arquiteturas de metadados educacionais, dentre elas ARIADNE [ARIADNE 2002], Artesia-TEAMS [Artesia-TEAMS 2002] e GESTALT [Foster et. al. 2000].

Esta arquitetura genérica é composta por um **Cliente** o qual busca por um recurso educacional através de um **Broker**. Este consulta um repositório de metadados educacionais em formato LOM-XML. O **Broker** referencia o **Provedor de Conteúdo** que é responsável por acessar o repositório de recursos educacionais e fazer a entrega do recurso solicitado ao cliente. Isto pode ser visto na Figura (2).

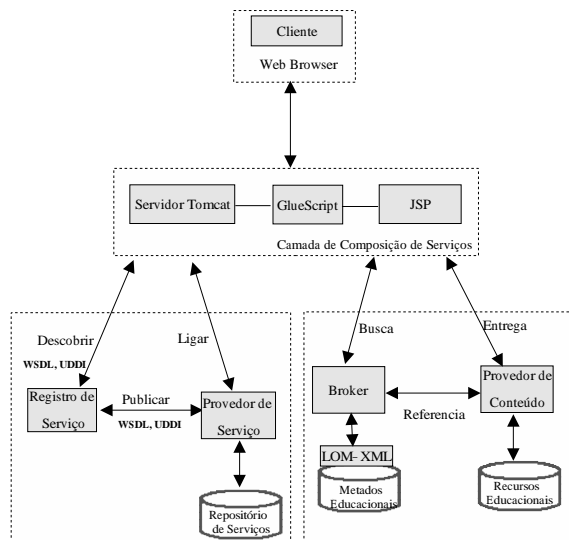


Figura 2: Protótipo da Arquitetura da DSL GlueScript [Hansen 2003]

4.1. Arquitetura da *Domain Specific Language* (DSL) GlueScript

A DSL GlueScript baseia-se na arquitetura da Figura 2, para realização de suas operações.

Esta arquitetura é composta pela estrutura padrão de *Web services* e uma arquitetura genérica utilizada com metadados educacionais [Hansen 2003]. A arquitetura proposta apresenta uma **Camada de Composição de Serviços**, a qual permite que seja feita busca de serviços utilizando a estrutura de *Web Services* existente, bem como fazer uso de recursos educacionais especificados por LOM-XML.

Esta nova camada permite que serviços sejam encontrados tanto em ambientes *Web Services* tradicionais como em ambientes específicos de educação. Torna-se possível manipular estes serviços para a construção de novas aplicações para ambientes de *e-learning*.

4.1.1. Camada de Composição

A camada de composição é composta por um servidor Web Tomcat, pela linguagem JSP e pela GlueScript.

As *JSP Tags Libraries* permitem estender JSP [Schachor et. al. 2001, Silva 2002]. A linguagem de composição GlueScript é uma extensão de JSP, sendo utilizada através de uma *Tag Library*.

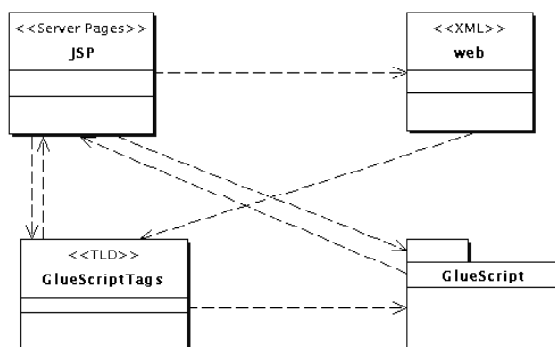


Figura 3: Diagrama UML da GlueScript

A Figura 3 apresenta um diagrama UML do relacionamento da *Tag Library* da GlueScript e JSP. O desenvolvedor Web poderá fazer uso dos recursos já disponíveis em JSP e fazer uso das *Tags* da GlueScript. Quando o servidor for compilar os códigos JSP, e encontrar *Tags* que não forem padrão de JSP, o servidor através de um arquivo web.xml irá localizar o arquivo que armazena as *Tags* da GlueScript. Um arquivo GlueScriptTags irá referenciar as *Tags* e os parâmetros possíveis de serem utilizados.

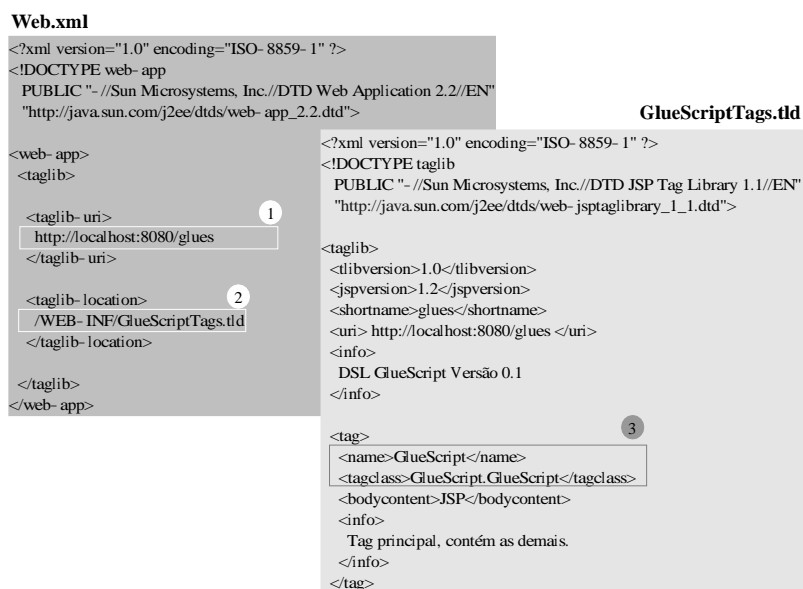


Figura 4: Relacionamento dos Arquivos web.xml e GlueScriptTags.tld

O relacionamento entre os arquivos Web.xml e GlueScriptTags.tld é apresentado na Figura 4 onde:

1. Define o endereço da localização de todos os arquivos relacionados para que possam ser encontrados pelo Tomcat.
2. Define a localização do arquivo descritor das *tags* (GlueScriptTags.tld). Desta forma o servidor pode localizar o descritor de *tags*, o qual apresenta a relação de todas as *tags* desenvolvidas, os parâmetros (quando houver) e o nome da classe Java relacionada.
3. Apresenta a definição de uma *tag* (GlueScript) e sua classe Java correspondente no pacote GlueScript. Esta é a *tag* principal do conjunto de *tags*. Em virtude do arquivo descritor de *tags* ser de um tamanho considerável, ele apresenta apenas a *tag* principal (GlueScript).

Foram desenvolvidas um total de oito *tags*, que são descritas sinteticamente como segue:

- **GlueScript**: é necessária para a utilização de todas as outras *tags* desenvolvidas. Esta *tag* é responsável por armazenar os parâmetros que são retornados pelas demais *tags* da linguagem. É a *tag* que determina o início e o final da utilização das *tags* da GlueScript.
- **Tags LocateWS e LocateLOM**: estas *tags* fazem a localização de *Web Services* e recursos educacionais em registros. Através do endereço de um registro UDDI ou de um registro de metadados LOM é possível saber onde estão os documentos de descrição de *Web Services* e recursos educacionais.
- **Tags ParsingWS e ParsingLOM**: permitem analisar os documentos de descrição de *Web Services* e recursos educacionais para obtenção de informações como a localização, o tipo de recurso ou serviço dentre outras. São utilizadas em conjunto com as *tags* LocateWS e LocateLOM.
- **Tags AllocateWS e AllocateLOM**: estas *tags* permitem a utilização de *Web Services* e recursos educacionais
- **Tag Pipe**: esta *tag* tem como objetivo permitir que o resultado da invocação de um *Web Service* ou um recurso educacional possa servir de entrada para outro.

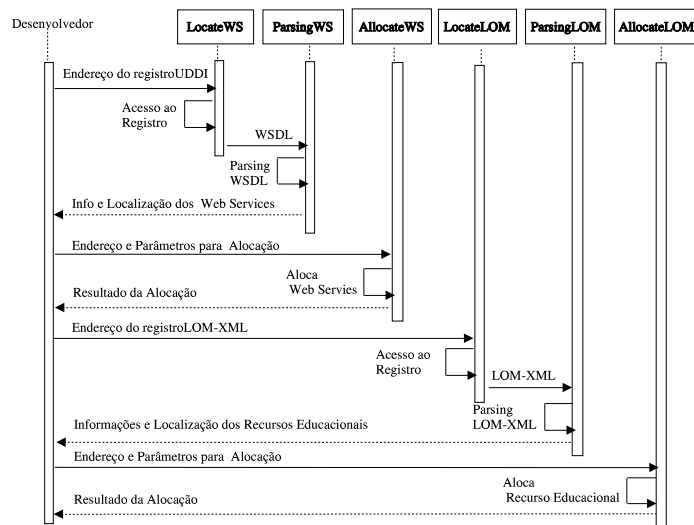


Figura 5: Diagrama de Sequência das Tags da GlueScript

O relacionamento entre as tags é apresentado na Figura 5. A tag **LocateWS** é utilizada para localizar os *Web Services* disponíveis em um registro UDDI. Esta tag é utilizada em conjunto com a tag **ParsingWS** que realiza o *parsing* do registro para retornar os nomes dos *Web Services* e a localização do documento WSDL destes. Com a localização do documento WSDL é possível saber a localização do *Web Service* e quais métodos são necessários para sua utilização, bem como os parâmetros que devem ser passados. Através da tag **AllocateWS** é possível então fazer uso do *Web Service*. A tag **LocateLOM** é utilizada para localizar recursos educacionais através de um endereço de um repositório de LOM-XML. É utilizada em conjunto com a tag **ParsingLOM** para extrair as informações de localização e como um recurso educacional pode ser utilizado. Estas informações são retornadas e então através da tag **AllocateLOM** é possível fazer uso deste recurso.

As tags foram implementadas em Java utilizando o pacote *Java Web Services Development Pack* (JWSDP) [Amrstrong et. al. 2002], que apresenta um conjunto de APIs Java para XML o que facilita o uso do protocolo SOAP para o acesso aos registros UDDI, documentos WSDL e LOM-XML.

A GlueScript permite através dos seus comandos algumas funcionalidades como localizar serviços ou recursos educacionais, através de documentos WSDL e do metadado LOM-XML. Permite que estes recursos possam ser acessados e reutilizados em forma de composição para a criação de ambientes educacionais para Web.

Para facilitar o uso da GlueScript foi desenvolvido um assistente que permite ao desenvolvedor realizar a busca e acesso a *Web Services* e recursos educacionais de maneira mais rápida.

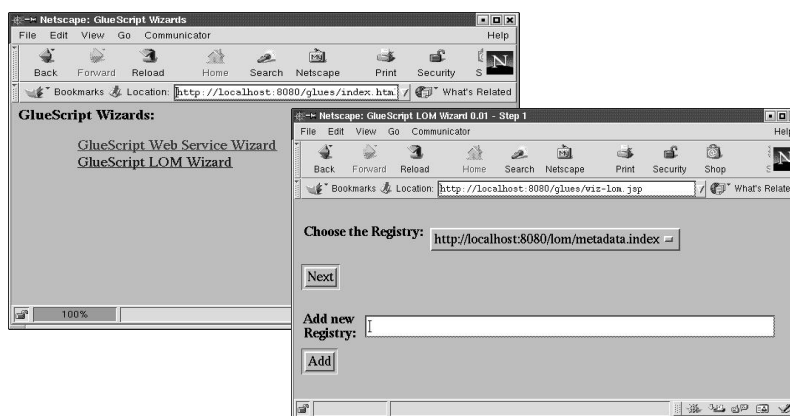


Figura 6: Assistente GlueScript - Acesso aos Registros de Lom-XML

O assistente utiliza as *tags* desenvolvidas e gera um código básico para utilização na construção de páginas Web. A Figura 6 apresenta a seqüência de passos para localizar e fazer uso de Recursos Educacionais. Através da escolha do registro de metadados educacionais é possível verificar os recursos educacionais descritos em LOM-XML que está disponíveis.

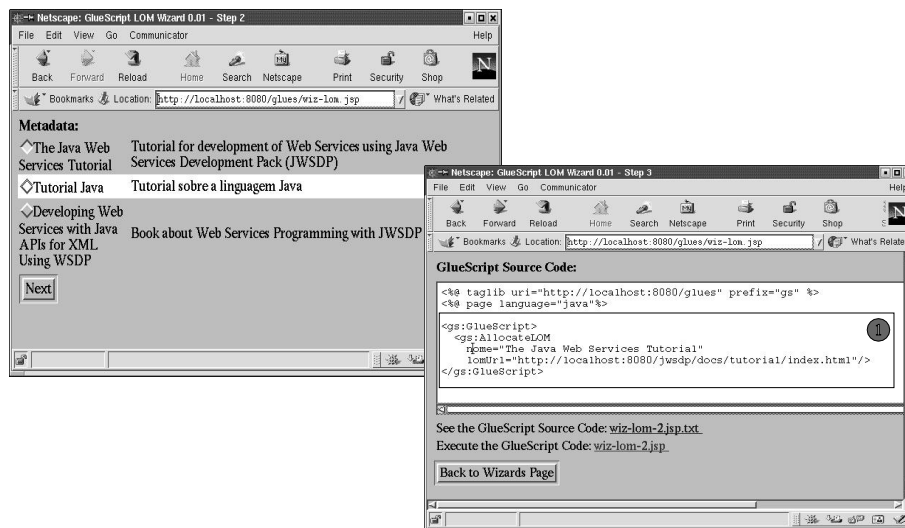


Figura 7: Assistente GlueScript - Escolha do Recurso e Geração do Código

A Figura 7 mostra que o assistente permite que se escolha o recurso educacional que se deseja fazer uso. O assistente gera então um código básico para construir uma página Web (1). Este código pode ser salvo ou acessado através do assistente. A partir do código gerado, o desenvolvedor necessita apenas construir a interface do ambiente Web.

O Assistente também permite a localização e acesso a *Web Services* de forma análoga a LOM-XML. Através da escolha de um registro UDDI é possível acessar os documentos WSDL que descrevem os *Web Services* que estão disponíveis no registro. Os métodos necessários para acessar o *Web Service* são disponibilizados pelo assistente ao desenvolvedor. O assistente gera então o código básico para acesso ao *Web Service* escolhido.

Utilizando as *tags* desenvolvidas foi realizado um estudo de caso, com a finalidade de verificar as funcionalidades da GlueScript.

5. Estudo de Caso

Com o objetivo de avaliar as funcionalidades das *tags* da GlueScript, foi realizado um estudo de caso utilizando composição de *Web Services* e recursos educacionais.

Foi desenvolvido um curso para Web, apresentando uma dinâmica de trabalho com momentos síncronos (*chat*) e assíncronos (acesso a página, *e-mail*). Foi utilizado o assistente da GlueScript para facilitar a localização de *Web Services* e recursos educacionais que estivessem disponíveis na Web. O uso do assistente permitiu que além da localização, fosse possível gerar um código básico com as *tags* e os parâmetros necessários para a alocação dos *Web Services* e dos recursos utilizados. A partir deste código básico foi necessário apenas o desenvolvimento das interfaces para os usuários que foram implementadas em JSP.

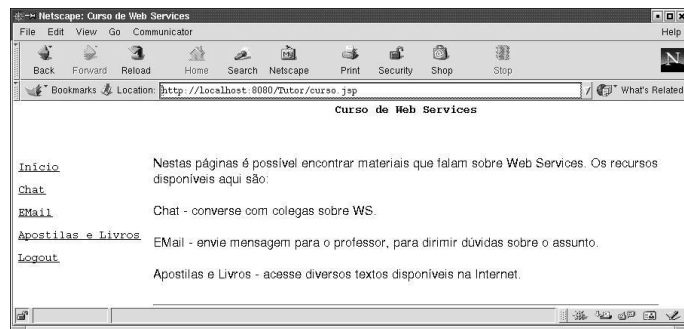


Figura 8: Página Principal do Curso

A Figura 8 apresenta a página principal do curso. A partir da página principal o aluno pode escolher entre enviar um *e-mail* para o professor, acessar o *chat* e trocar idéias a fim superar possíveis dúvidas, além de ter acesso a tutoriais e livros.

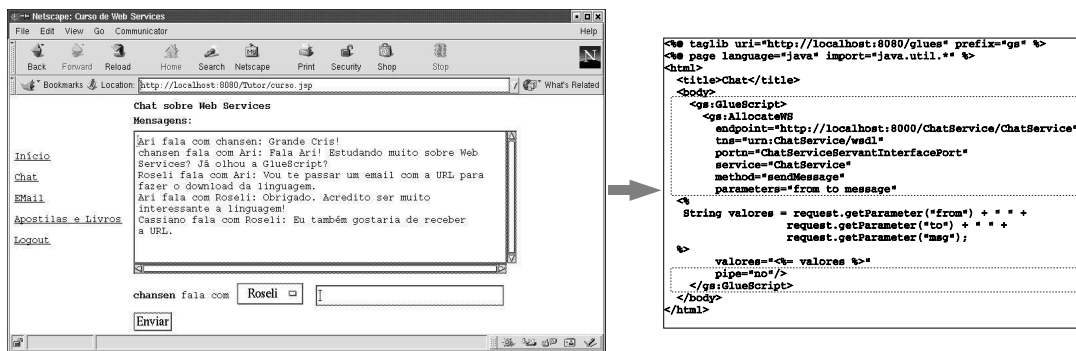


Figura 9: Página de Chat do Curso

Através do link *Chat*, o aluno acessa uma página conforme a Figura 9, na qual pode conversar sobre *Web Services* com demais interessados no assunto, que tenham acesso ao curso. O código destacado com a *tag AllocateWS*, foi gerado pelo assistente da GlueScript.

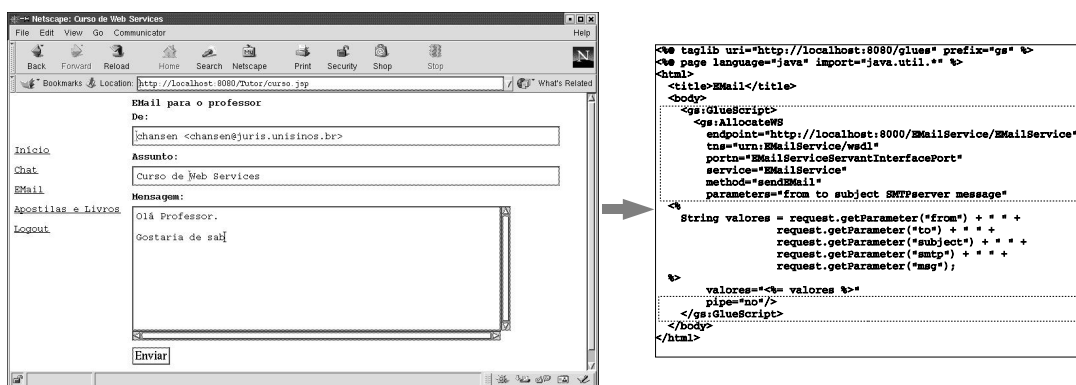


Figura 10: Página de E-mail do Curso

Ao acessar o link *Email* o aluno pode enviar *e-mails* para o professor do curso, de forma a obter informações adicionais ou sanar dúvidas que possa ter. A Figura 10 apresenta a página que permite o envio de mensagens e o código necessário para uso do *Web Service* de *e-mail*. O fragmento de código destacado foi gerado pelo assistente da GlueScript.

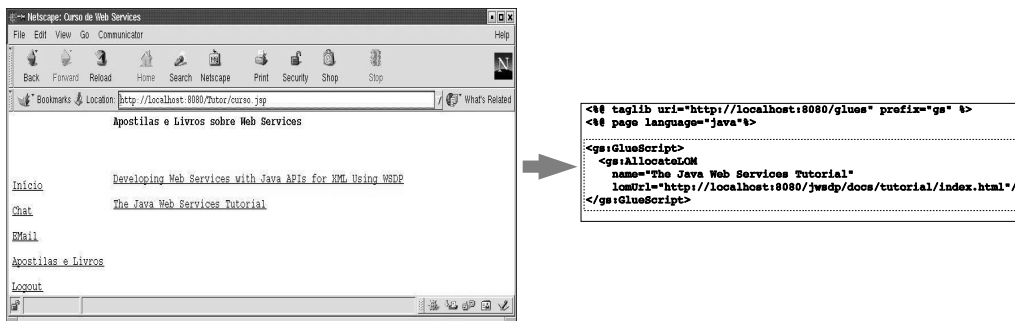


Figura 11: Página de Apostilas e Livros do Curso

O curso disponibiliza também uma página com *links* de apostilas e tutoriais sobre *Web Services*. Na Figura 11 são apresentados dois *links* de um livro e um tutorial sobre *Web Services*. O *link* do tutorial *The Java Web Services Tutorial*, apresenta o código conforme a Figura 11 que faz uso da tag *AllocateLOM* gerado pelo assistente da GlueScript.

6. Conclusões

A criação de ambientes para a Web dispense conhecimentos específicos de linguagens com suporte a aplicações Web, além de tempo na construção de aplicações necessárias. Como forma de amenizar este problema, este trabalho sugere uma arquitetura e uma linguagem de composição baseada em *Web Services* para a criação destes ambientes.

A utilização da tecnologia baseada em *Web Services* permite que serviços possam ser disponibilizados na Web de forma padronizada. Esta padronização permite a localização e acesso a serviços de forma bastante eficiente, já que as tecnologias empregadas estão baseadas em XML. As mensagens SOAP são transmitidas em formato XML, o que facilita a manipulação de dados de entrada e saída. O uso do protocolo SOAP facilita o acesso a estes serviços de forma que este é encapsulado dentro do protocolo HTTP.

Através do uso das *tags* da GlueScript é possível localizar e acessar *Web Services* e recursos educacionais de forma dinâmica. Também torna a construção de ambientes Web mais simplificada pois um desenvolvedor pode fazer reuso de *Web Services* e recursos educacionais já existentes na Web.

O assistente da GlueScript permite geração de código básico para acessar *Web Services* e recursos educacionais de forma que um desenvolvedor precisa apenas fazer as interfaces para uso dos mesmos.

Ao utilizar a GlueScript é possível também fazer uso da linguagem JSP para dar mais poder de processamento a linguagem. O uso de JSP em conjunto com a GlueScript permite que o desenvolvedor possa acrescentar funcionalidades à sua página.

O estudo de caso foi realizado utilizando-se as *tags* da GlueScript onde foi possível observar as facilidades para a localização e acesso de *Web Services* e recursos educacionais baseados em LOM-XML. Isto foi possível utilizando o assistente que mostrou-se uma ferramenta bastante útil na geração de código básico para acesso aos *Web Services* e recursos educacionais. Não foi possível comparar a GlueScript, por não haverem trabalhos correlatos.

Referências

- Amrstrong, E.; Bodoff, S.; Carson, D.; Fisher, M.; Green, D. and Haase, K. *The Java Web Services Tutorial*. <http://java.sun.com/webservices/downloads/webservicestutorial.html>. Último acesso - Abril 2002.
- ARIADNE *Educational Metadata Recommendation*, Version 3.2. February 2002. <http://ariadne.unil.ch/>. Último acesso - Maio 2002.
- ARIBA, INC., International Business Machines Corporation and Microsoft Corporation. *UDDI Data Structures Reference*. September, 2000.

- Artesia-TEAMS. http://www.artesia.com/teams_architecture.html. Último acesso - Maio 2002.
- Babu, S. C. *e-Learning Standards*. 2001.
- BECTa - British Educational Communications and Technology agency. *Metadata in Education*. August 2001.
- Box, D; Ehnebuske, D.; Kakivaya, G.; Layman, A.; Mendelsohn, N; Nielsen, h. F.; Thatte, S.; Winer, D. *SOAP: Simple Object Access Protocol*, April 2000.
- Bray, T.; Paoli, J.; Sperberg-McQueen, C. M. and Maler, E. eXtensível Markup Language (XML) 1.0. Second Edition. <http://www.w3.org/TR/2000/REC-xml-20001006>. Último acesso - October 2000.
- Clements, T. *Overview of SOAP Web Services - Technical Overviews*. Sun Microsystems, August 2001.
- Christensen, E.; Curbera, F.; Meredith, G. and Weerawarana, S. *Web Services Description Language (WSDL)*, March 2001.
- Crespo, S.; Fontoura, M. F. and Lucena, C. J. *A Web-based Educational Environments Comparison using a Conceptual Model compatible with the EDUCOM/IMS Platform*. Brazilian Symposium on Education and Computer Science (SBIE'98), Fortaleza, Brazil, 1998 (in Portuguese).
- Dublin Core Metadata Element Set, Version 1.1: Reference Description*. July 1999. <http://dublincore.org/documents/1999/07/02/dces/>. Último acesso - Maio 2002.
- Ehnebuske, D.; Rogers, D.; Riegen, C. V. *UDDI Version 2.0 Data Structure Reference*. June 2001.
- Foster, P.; Kraner, M.; Graziano, A.; Romano, S. P. *GESTALT - Getting Educational Systems Talking Across Leading-edge Technologies*. April 2000.
- Gudgin, M.; Hadley, M. Moreau, J.; Nielsen, H. F. *SOAP Version 1.2*. W3C, July 2001.
- Hansen, R. P. *GlueScript: Uma linguagem Específica de Domínio para composição de Web Services*. Dissertação de Mestrado - Programa Interdisciplinar de Pós-Graduação -PIPCA - Universidade do Vale do Rio dos Sinos - Fevereiro de 2003.
- Hansen, R. P.; Santos, C.T.; Pinto, S. C. C. S.; Lanius, G. L and Massen, F. *Web Services: An Architectural Overview*. First International Seminar on Advanced Research in E-Business - EBR 2002. PUC-RIO. November 2002.
- IEEE LTSC. *Draft Standard for Learning Object Metadata*. March 2002. <http://ltsc.ieee.org/>. Último acesso - Maio 2002.
- IMS Global Learning Consortium, Inc. *IMS Learning Resource MetaData Information Model*. September 2001. <http://www.imsproject.org/>. Último acesso - Maio 2002.
- IMS Global Learning Consortium, INC. <http://www.imsproject.org/>. Último acesso - Maio 2002.
- Kreger, H. *Web Services Conceptual Architecture*. IBM Software Group, May 2001.
- Lucena, C.; Fuks, H. *A Educação na Era da Internet*. Clube do Futuro. 2000 156 páginas.
- Rajaram A. *Overview of UDDI. Web Services - Tecnical Overviews*. Sun Microsystems, August 2001.
- Robson, R. *Report on Learning Technology Standards*. ED-Media 2000.
- Schachor, G.; Chace, A. and Rydin M. *JSP Tag Libraries*. Manning Publications Co. 2001. 623 pages.
- Silva, W. L. S. *JSP and Tag Libraries for Web Development*. New Riders. 2002. 442 pages.