

INTEGRAÇÃO DE BANCOS DE DADOS EM UM AMBIENTE DE ENSINO-APRENDIZAGEM BASEADO NA WEB

Fábio de Jesus Lima Gomes*

RESUMO

A utilização do computador na educação vem demonstrando ser um grande auxílio no processo ensino-aprendizagem. Uma das formas de sua utilização é o software educacional: um software para auxiliar estudantes no aprendizado de um determinado conteúdo. Com a grande disseminação do uso da World Wide Web (*www*), e com o seu poder de alcançar pessoas nos mais diversos lugares do mundo, a *www* tornou-se um grande recurso para distribuição da informação e para vários outros fins. Dentro desta perspectiva, pode-se utilizar a *www* com fins educacionais, de diversas formas, sendo uma delas a utilização de software educacional. A disponibilização e integração de dados surgem como tendência de serviços oferecidos pela *www*. O objetivo de um sistema de integração de dados é oferecer aos usuários uma interface uniforme de acesso a diferentes fontes de dados, de forma que os usuários definam consultas especificando o que desejam saber e o sistema determine onde a informação pode ser encontrada e, em seguida, apresente as respostas para as consultas do usuário. O módulo de integração tem por objetivo atender aos requisitos de cooperação e colaboração entre os estudantes e o professor em um ambiente de ensino-aprendizagem, baseado na Web, fazendo que vários bancos de dados integrados

* Fábio de Jesus Lima Gomes - Mestre pela UFPE, Professor da Faculdade Santo Agostinho

sejam vistos por usuários globais como um único banco de dados.

PALAVRAS-CHAVE: Sistemas tutores inteligentes, Banco de dados. java, Educação a distância. Objetos distribuídos.

Abstract

The usage of the computer in the education has been showing to be a great aid in the teaching-learning process. One of the forms of this usage is the educational software, software to help students in the learning of determinate contents. Because of the great use dissemination of the World Wide Web ($_{www}$), and its power to reach people in the most different places of the world, the $_{www}$ became a great resource for distribution of information and many other purposes. Inside those perspectives, it is able to use the $_{www}$ with educational purposes, in several forms, being one of those the utilization of educational software. The disponibilization and integration of data appear as the tendency of services offered by $_{www}$. The objective of a integration data system is to offer users with a uniform interface for accessing to different data source, so the users define queries specifying what they want to know and the system determine where the information can be found and, afterwards, shows the reply to the user's queries. The quoted module has the objective of attend the requires of cooperation and collaboration between the students and the professor in a teaching-learning environment based on the web, making the various integrated databases to being viewed by global users as a unique database, supplying the wanted transparencies of distributed databases.

Key-words:

Intelligent Tutorial Systems. Database. Java. Long-distance Education. Distributed Objects.

1 Sistemas Tutores Inteligentes

A informática destaca-se entre as ferramentas utilizadas na atualidade para a educação, representando novas maneiras de se atingir um dos objetivos principais do processo ensino/aprendizagem que é tornar a

compreensão e absorção do conhecimento mais fácil e consistente. Com o uso das tecnologias de informação, pode-se conseguir uma maior atenção dos aprendizes, formas diferentes de se transmitir a teoria e simular a prática, focalizando todos os sentidos conhecidos para o assunto abordado.

O propósito não é substituir os professores, mas auxiliar a aprendizagem. Os sistemas educacionais são desenvolvidos na tentativa de conseguir resultados não atingidos, utilizando-se apenas as técnicas pedagógicas, como também permitir a realização de treinamentos e estudos sem a presença ou proximidade obrigatória do mestre humano.

Nesta perspectiva, foram desenvolvidos os Sistemas Tutores Inteligentes (STIs), sistemas voltados à educação com o auxílio do computador e que utilizam técnicas de Inteligência Artificial (IA) para flexibilizar o ensino, permitindo uma melhor interação entre o sistema e os usuários.

Os STIs são sistemas nos quais a IA desempenha um papel de relevo, não só por permitir maior flexibilidade no ensino por computador, mas também por possibilitar a participação ativa do aluno e do sistema, gerando um ambiente cooperativo para o ensino e a aprendizagem (de ambos os agentes, aluno e sistema). É preciso destacar a importância da interação cooperativa, onde ambos os agentes (tutor/aluno) interagem visando a troca de conhecimento.

Os Sistemas Tutores Inteligentes possibilitam um ensino auxiliado por computador flexível e inteligente na forma de apresentação do conteúdo, na avaliação de qual material será exposto ao aluno dependendo do seu nível de aprendizado, na sua forma de comunicação onde predominam o entendimento do aluno e a satisfação deste no uso de um sistema agradável e atraente.

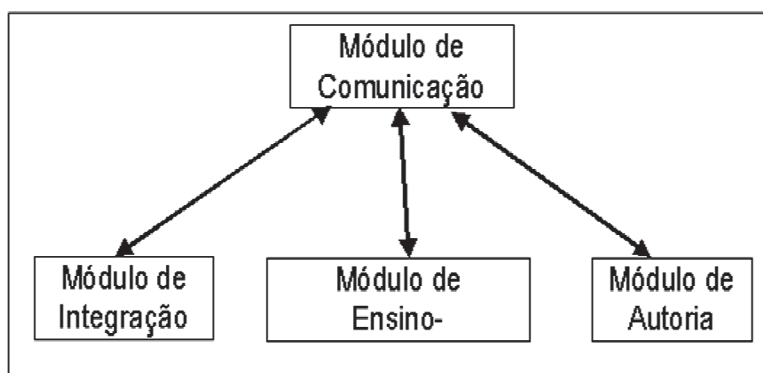
Um sistema tutor inteligente usado na *WWW* oferece como grande vantagem a possibilidade de poder ser utilizado por diversos usuários, em diversos lugares.

No intuito de atender os requisitos descritos acima, foi proposto um ambiente inteligente de ensino-aprendizagem, baseado na *Web* chamado WILE (*Web based Intelligent teaching Learning Environment*),

que é uma extensão de um projeto em andamento na Universidade Federal de Pernambuco, chamado Sistema de Ensino Inteligente (SEI).

Todas as características do SEI são incorporadas pelo WILE, através do módulo de ensino-aprendizagem. O ambiente é baseado em um sistema tutor inteligente (STI), que é um recurso educacional importante no processo de ensino-aprendizagem permitindo tornar a arquitetura mais adaptável e portátil. Esse ambiente é fundamentalmente cooperativo e colaborativo, pois sua arquitetura permite que cada participante tenha acesso ao curso e às pesquisas individuais dos demais participantes, criando uma maior integração e permitindo a troca de informações. WILE permite a construção e o acúmulo de informação através das bases de conhecimento do próprio ambiente e das bases de dados individuais de cada aluno. Seus módulos constituintes são: módulo de ensino-aprendizagem, módulo de autoria, módulo de integração e módulo de comunicação, e são mostrados na figura 1.

FIG 1. ORGANIZAÇÃO DOS MÓDULOS



O módulo de ensino-aprendizagem é um sistema tutor inteligente (corresponde ao SEI).

O módulo de autoria dentro do WILE tem por objetivo permitir que um autor sem conhecimento especializado de informática, como por exemplo, programação, construa a base de conhecimento do domínio

específico.

O módulo de comunicação agrega ferramentas para dar suporte à estrutura organizacional do curso tais como: biblioteca, lista de discussão, mural e agenda de aulas, dentre outros, como também os requisitos de facilidade de uso.

O módulo de integração endereça as características de distribuição e autonomia, importantes para atender aos requisitos de cooperação e colaboração entre estudantes e professores.

Desta forma, este ambiente ameniza o impacto do uso da tecnologia tanto para o aprendiz, que tem disponível um módulo de ensino e acompanhamento do aprendizado, quanto para o autor, que tem disponível um módulo que permite a construção de seu curso sem que o mesmo tenha que ter conhecimentos especializados de informática.

O WILE ainda está em desenvolvimento e, após sua conclusão, necessita de testes exaustivos, avaliações e ajustes, com a intenção de melhorar a sua eficiência e o seu potencial de reutilização, através do refinamento da estrutura da sociedade de agentes e seus mecanismos de raciocínio.

2 Java como linguagem de programação para a Web.

A linguagem de programação Java foi desenvolvida pela Sun Microsystems no início da década de 90, quando o objetivo principal era o uso de uma linguagem de programação, que permitisse a integração total de sistemas de computação com equipamentos de consumo como caixas de comutação de TV a cabo.

Uma das suas maiores vantagens é a portabilidade, ou seja, um programa escrito em Java pode ser executado em várias plataformas, como Windows NT, Unix ou Solaris. Essa característica torna essa linguagem de programação ideal para a WWW, pois um programa Java pode ser executado em qualquer plataforma sem necessidade de alterações no código-fonte.

Além desta importante funcionalidade, Java também se caracteriza por ser orientada a objetos, robusta, segura, simples e multithreading.

Com relação a ser uma linguagem particularmente adequada a

www, Java se popularizou com pequenos aplicativos chamados *applets*. Estes pequenos aplicativos são carregados embutidos em páginas HTML e rodam geralmente dentro dos browsers que acessam a Internet. Um novo conjunto de aplicação, os *servlets*, estão cada vez mais popularizando a linguagem. *Servlets* são aplicativos que podem ser acoplados em diversos tipos de servidores para expandir suas funcionalidades.

As aplicações atuais da *Web* tendem a requerer cada vez mais conteúdo dinâmico. Uma das formas mais seguras de armazenamento deste conteúdo é através de Sistemas de Gerenciamento de Banco de Dados (SGBD). O principal problema encontrado nas aplicações tradicionais de banco de dados, entretanto, é que os códigos dos programas ficam dependentes das características dos bancos de dados que são utilizados. Se a aplicação necessita se comunicar com outros bancos de dados, será necessária a alteração dos programas que acessam esses bancos de dados.

A linguagem Java apresentou como solução uma API, chamada JDBC (*Java Database Connectivity*), que é um conjunto de classes e interfaces escritas na própria linguagem para facilitar o envio de comandos SQL (*Structured Query Language*) para sistemas de bancos de dados relacionais.

A vantagem de usar a tecnologia JDBC está em construir aplicações que podem acessar várias fontes de dados heterogêneos, podendo executar essa aplicação em qualquer plataforma que possua uma máquina virtual Java. Ainda, com referência à conectividade com banco de dados, devido ao seu ciclo de vida, o *servlet* permite manter abertas as conexões ao banco de dados. Uma conexão existente pode cortar vários segundos de um tempo de resposta.

Outra vantagem de *servlets* sobre outras tecnologias é que, juntamente com a API JDBC, se torna independente do banco de dados. Um *servlet* escrito para acessar um banco de dados Sybase pode, com a modificação de duas linhas ou uma alteração em um arquivo de propriedades, começar a acessar um banco de dados Oracle.

JDBC está sendo utilizada para acessar todas as bases de dados, quer a base de dados localizada no servidor, quer as bases de dados cooperantes. A funcionalidade básica dessa API é prover basi-

camente: conexão com bancos de dados, enviar comandos SQL e processar resultados.

O acesso ao conteúdo instrucional do STI é feito através do protocolo HTTP com chamadas aos *servlets*, e recebendo como respostas páginas HTML. Os *servlets* são responsáveis pela conexão com o banco de dados do servidor através de um *driver* JDBC.

3 Objetos Distribuídos

Devido a fatores como o barateamento do processamento, a maturidade de tecnologias utilizadas em redes de computadores, a necessidade de maior flexibilidade e escalabilidade na infra-estrutura computacional, bem como a crescente necessidade de comunicação entre organizações, a atual infra-estrutura de informação predominante é composta por nodos processadores heterogêneos interconectados para a troca de informação.

O paradigma de orientação a objetos tem causado mudanças significativas na análise, projeto e desenvolvimento de sistemas. O desenvolvimento de software orientado a objetos tem se acentuado com a promessa de aumento de modularidade, reusabilidade e conseqüente facilidade de construção. Apesar das vantagens, a orientação a objetos possui ainda áreas ainda não totalmente exploradas, especialmente com sistemas legados.

O aumento da capacidade de processamento das estações de trabalho acelerou o interesse em adaptar as aplicações de grande porte para estações de trabalho, distribuindo funcionalidades entre esses nodos. Através dos conceitos definidos pela pesquisa em sistemas distribuídos migra-se de uma infra-estrutura de plataformas homogêneas para outra geograficamente aberta e composta por máquinas heterogêneas. Este fato contribuiu para uma crescente necessidade de desenvolvimento de software para ambientes distribuídos.

Neste contexto, as aplicações, conseqüentemente, passam a ser compostas por módulos comunicantes dispostos em máquinas remotas. A interação entre módulos distribuídos pode acontecer mediante a utilização de vários tipos de mecanismos. Alguns deles exigem que o desenvolvedor de aplicações trate problemas de locali-

zação e comunicação entre os módulos. As duas áreas de pesquisa envolvidas são: orientação a objetos e sistemas distribuídos.

A utilização da orientação a objetos agrega inúmeras características ao processo de desenvolvimento dos objetos distribuídos, das quais merecem destaque o encapsulamento, comportamento, polimorfismo, herança e identidade de objetos. Já os sistemas distribuídos agregam novas funcionalidades como interoperabilidade, escalabilidade, disponibilidade e desempenho.

Objetos distribuídos são componentes de software criados com os conceitos da orientação a objetos, podendo estar distribuídos através de uma rede heterogênea, mas visto pelas aplicações como se fossem componentes locais, fornecendo com isso vários tipos de transparências desejadas em sistemas distribuídos.

A tecnologia de objetos distribuídos pode trazer uma série de benefícios às aplicações via Internet. Dentre eles, destacam-se:

- Transparência;
- Balanceamento de carga;
- Processamento Distribuído;
- Independência de plataforma e Interoperabilidade;
- Escalabilidade;
- Tolerância a falhas;
- Extensibilidade e Flexibilidade;

Dentre os padrões para sistemas distribuídos com suporte à criação de objetos distribuídos, pode-se citar: DCOM da Microsoft, CORBA do OMG (*Object Management Group*) e Java/RMI, dentre os quais, foi escolhido Java/RMI para implementação do módulo de integração.

Java/RMI é um sistema de invocação remota de métodos que estende os conceitos de RPC (*Remote Procedure Call*) para dar suporte a objetos distribuídos Java.

O Java/RMI não precisa lidar com questões relacionadas com heterogeneidade, já que, tanto as classes do cliente como do servidor, são escritas em Java e executam em Máquinas Virtuais Java, o que torna a rede uma coleção de máquinas (virtuais) homogêneas e elimina a necessidade de linguagens adicionais.

No módulo de integração, Java/RMI está sendo utilizada para comunicação entre objetos distribuídos (remotos) no que diz respeito

aos acessos aos bancos de dados cooperantes. A estratégia principal é manter vários servidores RMI rodando, permitindo com isso, a possibilidade de rodar solicitações de conexões com os bancos de dados cooperantes em paralelo. Isso somente é possível pela característica *multithreading* da linguagem Java, em especial a característica *multithreading* dos *servlets*.

4 Bancos de Dados e Web

A evolução da tecnologia da informação e comunicação está promovendo uma transformação social e econômica de tal complexidade, que suas conseqüências podem ser comparadas ao que ocorreu com a introdução da prensa tipográfica. Os computadores pessoais, os laptops, a Internet, os telefones celulares, o comércio eletrônico estão efetivamente mudando a estrutura da vida das pessoas, criando um mundo gerado pela velocidade, conectividade, intangibilidade e criatividade.

A Internet é o exemplo mais claro da velocidade das transformações que é vivenciada. Criada em 1969, com o nome de Arpanet, como parte de um programa de pesquisa do Departamento de Defesa Americano, na Universidade da Califórnia, teve uma evolução relativamente lenta enquanto ficou restrita ao papel de elemento facilitador da comunicação no meio acadêmico.

Com a evolução da tecnologia de informação e das telecomunicações, houve uma ampliação do seu papel, constituindo-se hoje em um fantástico repositório de informações, onde a participação é absolutamente livre e voluntária. Qualquer pessoa ou organização pode colocar ou acessar informações na Internet. Esta facilidade de participação, aliada ao fato de estar associada a um mercado potencial de milhões de usuários, fez com que o número de *sites*, redes, *hosts* e domínios tivesse um crescimento extraordinário nos últimos anos.

No campo da informação, vive-se hoje o fenômeno da hiperinformação. Seja em papel, vídeo, filme, formato eletrônico ou áudio, por exemplo, o homem está cada vez mais exposto ao excesso de informação. Isto pode implicar no paradoxo de que dispor de informação em excesso e incontrolável, é tão ruim quanto não ter

nenhuma informação.

Nos últimos anos, a Internet e, em particular, a *World Wide Web* (www) ou simplesmente Web, tem se tornado a mídia preferida para transmissão, disseminação de informações e acesso a dados.

A Web passou da filosofia de troca de arquivos e comunicação pessoal para publicação de grandes bases de dados com inúmeras novas aplicações distribuídas e multi-plataforma, como: comércio eletrônico, bibliotecas digitais e educação à distância.

Paralelamente, os Sistemas Gerenciadores de Banco de Dados (SGBDs) apresentaram uma evolução nos últimos anos, que permitiu torná-los o componente central de modernos ambientes de computação. No entanto, os SGBDs surgiram e evoluíram para atender às necessidades características de seu ambiente, sem a preocupação com uma futura e eventual integração com outros sistemas.

Além disso, o volume de informações disponível na Web vem aumentando consideravelmente e a tecnologia de Banco de Dados representa uma parte crítica desse crescimento, não só como fonte para armazenamento e gerenciamento de objetos Web, como também para o desenvolvimento de novas aplicações sob Sistemas de Banco de Dados na Web.

5 Módulo de Integração de Banco de Dados em um Ambiente de Ensino / Aprendizagem Baseado na Web.

Na arquitetura proposta por Souza e Campos para um ambiente de ensino-aprendizagem baseado na Web, além das bases de conhecimento, cada estudante tem sua própria base de dados, chamada de cooperante, e constituída por suas pesquisas. A utilização de bases de dados cooperantes traz algumas vantagens:

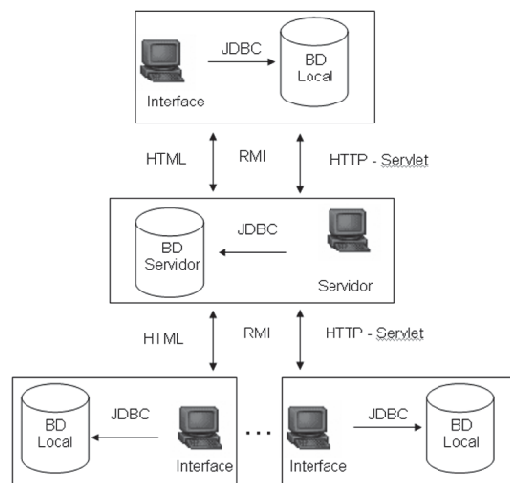
- Compartilhamento de dados;
- Confiabilidade e disponibilidade;
- Aceleração de processamento de consultas;
- Independência de localização;
- Escalabilidade;
- Autonomia e Descentralização.

Essa base de dados local e cooperante pode ser compartilhada

com o grupo, permitindo que os participantes tenham acesso às pesquisas individuais de cada participante, criando uma maior integração e troca de informações, permitindo a construção e o acúmulo de informação através das bases de conhecimento do próprio ambiente, como também, através das bases de dados individuais de cada aluno, contribuindo assim para um ambiente efetivamente cooperativo.

Sua arquitetura, mostrada na figura 2, permite que qualquer usuário do sistema visualize, através do *browser*, as informações disponíveis de seu grupo de estudos. O tipo de informação que o usuário pode recuperar é limitado ao conjunto de funcionalidades disponibilizadas pelas páginas HTML, que são acessos à base de dados local de um estudante pelo tópico ou pela lição para recuperação das pesquisas realizadas pelo estudante.

FIGURA 2: ARQUITETURA DO MÓDULO DE INTEGRAÇÃO



Bases Cooperantes

O módulo de integração implementa um conjunto de consultas pré-definidas, que permite o acesso ao banco de dados de outros estudantes pertencentes ao curso. As solicitações são: consulta sobre uma única base de dados, específica para um tópico ou para toda a lição, ou mais de uma base de dados para um determinado tópico ou lição. Sua função é recuperar dados das fontes de dados

escolhidas submetendo os pedidos e realizar o processamento necessário para integrar dados de diversas fontes. O caso mais simples é quando é feita a consulta sobre uma única fonte de dados.

REFERÊNCIAS BIBLIOGRÁFICAS

SOUZA, F. F. ; CAMPOS, V.V.S. **WILE – Web based Intelligent teaching-Learning Environment. Proceedings of the ACIS International Conference on Computer Science, Software Engineering, Information Technology, e-Business and Applications – CSITCA '02.** Foz do Iguaçu, 2002, p. 385-390.

TEDESCO, P. **Dissertação de Mestrado: SEI - Sistema de Ensino Inteligente.** Departamento de Informática da UFPE. 1997. Disponível em: Java, <<http://www.javasoft.com>>. Acesso em: 09/2003.

HOSTMANN, Cay S; CORNELL Gary: **Core Java 2: fundamentos.** São Paulo, Makron Books, 2001. v. 2.

CASTRO, Elizabeth. **HTML: programação para World Wide Web.** Ed. Peachipit Press. 1997.
Disponível em: <<http://java.sun.com/products/servlet>>. Acesso em: 09/2003.

HOSTMANN Cay S.; CORNELL Gary. **Core Java 2: recursos avançados.** São Paulo, Makron Books, 2001. v. 2.

HUNTER, Jason. Java servlet programming. O'Reilly & Associates, United States of America, CA, 1998. Disponível em: <http://java.sun.com/jdbc/>. Acesso em: 09/2003.

ELMASRI, Ramez; NAVATHE Shamkant B. **Fundamentals of database systems.** 3. ed. Addison-Wesley, 2000.

WALLNAU, K.; WEIDREMAN, N.; NORTHOP, L. **Distributed object technology with CORBA and Java:** Key Concepts and Implications. Technical Report CMU SEI-97-tr-004. Pennsylvania: Carnegie Mellon University, 1997.

NELSON, B. J. **Remote procedure call. PhD thesis.** Department of Computer Science. Carnegie-Mellon University, Pittsburgh, 1981.