

## Framework para o Ensino a Distância baseado em Componentes de Software

### Framework for Distance Education based in Software Components

Iolanda Cláudia Sanches Catarino\*  
 Antonio Francisco do Prado\*\*  
 Raphael Marcílio de Souza Neto\*\*\*

\* Mestre em Ciência da Computação (UFSCar).  
 Docente da Universidade Norte do Paraná  
 (UNOPAR)

e-mail: <iolanda@unopar.br>

\*\* Doutor em Informática (PUC/RJ). Docente do  
 Dep. de Computação da Universidade Federal  
 de São Carlos (UFSCar)

e-mail: <prado@dc.ufscar.br>

\*\*\*Mestrando em Ciência da Computação  
 (UFSCar).

e-mail: <raphael.marcilio@dc.ufscar.br>

#### Resumo

A educação a distância envolve o processo de ensino e aprendizagem mediado por recursos didáticos e tecnológicos, sistematicamente organizados, possibilitando a auto-aprendizagem sem limitações de tempo e lugar. Atualmente, as tecnologias da informação, como a Internet e a WWW, com todos os seus mecanismos de informação, comunicação e cooperação, estão proporcionando a construção do conhecimento em grande escala e a custos mais acessíveis. Muitas instituições de ensino estão adotando um ambiente de desenvolvimento de cursos para oferecer educação a distância via Web. Esses ambientes integram recursos e serviços didáticos, administrativos e suporte aos cursos. A partir do estudo desses ambientes foi desenvolvido um *Framework* para Ensino a Distância que suporta a construção de ambientes para elaboração e gerenciamento de cursos na Web, a partir da reutilização de seus componentes. Este artigo apresenta a metodologia de desenvolvimento do *Framework* e a sua arquitetura com os componentes reusáveis.

**Palavras-chave:** framework, componentes, Tecnologia da Informação, web, Educação a Distância.

#### Abstract

*Distance education comprehends the teaching process and learning through didactic and technological resources, systematically organized, making possible learning by oneself with no time or place limitations. Nowadays, information technology like Internet and WWW with their mechanisms of information, communication and cooperation are ensuring a positive environment for the development of courses accessible via WEB. These ambiances integrate didactic resource and services to provide a base for these courses. Taking them as a base it was developed a Framework for Distance Education that makes feasible to construct the appropriate ambiances to elaborate and manage such courses in the WEB. This article presents the methodology for developing such Framework and its architecture with reusable components.*

**Key words:** frameworks, components, Information Technology, web, Distance Education.

#### Introdução

A Educação a Distância (EAD) é uma modalidade de educação, com a mediação de recursos didáticos sistematicamente organizados, apresentados em diferentes suportes de informação, utilizados isoladamente ou combinados e veiculados pelos diversos meios de comunicação (LDB, 1996).

A EAD é uma alternativa de educação que tem se desenvolvido com o objetivo de atender um grande contingente de pessoas em busca de ensino, treinamento ágil e atualização constante. Para muitos, essa forma de educação representa a oportunidade de novos conhecimentos, sem limitações de tempo e lugar.

Com o avanço das tecnologias de comunicação e a expansão das redes de computadores, novas tecnologias, como a Internet e a *World Wide Web* (WWW), estão sendo utilizadas para realização de cursos a distância. A Internet com todos os mecanismos de informação e comunicação proporciona grandes oportunidades para a educação, à medida que facilita a troca e

a cooperação entre pessoas distantes, assim como possibilita que cada um se torne fornecedor de informação e, ao mesmo tempo, usuário da informação (LUCENA; SALVADOR, 1999).

Muitas instituições de ensino estão criando, aperfeiçoando ou implantando programas de EAD que utilizam recursos da Internet. Entretanto, elaborar cursos via web exige muito planejamento e empenho do professor. Para facilitar aos professores e alunos no processo de ensino e aprendizagem via Web, ambientes para criação e administração de cursos a distância estão sendo desenvolvidos com ferramentas interativas que integram recursos e serviços didáticos e de comunicação.

Hoje no mercado existem vários ambientes para desenvolvimento de cursos via Web destacando-se:

- UniverSite* (1998] da empresa latino-americana MHW;
- WebCT* (1997) da *University of British Columbia* no Canadá;
- AulaNet* (1998) do Laboratório de Engenharia de Software (LES) da Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio);

- d) *Web Course in a Box* (1997) da *University of Commonwealth* na Virginia;
- e) *Learning Space* (1998) da *Lotus Education*, comercializado pela IBM;
- f) *Classnet* (1997) da *Iowa State University Computation Center*.

A partir do estudo desses ambientes foi desenvolvido um *Framework* para Ensino a Distância que suporta a construção de sistemas, denominados ambientes, para a elaboração e gerenciamento de cursos, a partir da reutilização de seus componentes. Este artigo apresenta a metodologia de desenvolvimento do *Framework* e a sua arquitetura com seus componentes reusáveis.

As principais tecnologias empregadas na construção do *Framework* são (SANCHES, 2000):

- a) *Catalysis* (D'SOUZA; WILLS, 1998) como método de desenvolvimento de software Orientado a Objetos do *Framework*;
- b) a ferramenta CASE *Rational Rose* (RATIONAL SOFTWARE CORPORATION, 2000) para documentar a especificação de requisitos do *Framework*;
- c) a linguagem Orientada a Objetos Java (SUN MICROSYSTEMS, 2000), usada para implementação do *Framework*, por ser multiplataforma e ter facilidades para comunicação com a Web, e principalmente por incluir a tecnologia *Enterprise JavaBeans* (EJB) (VALESKEY, 1999; RAJ, 2000). A tecnologia EJB é usada para:
  - definir as interfaces de acesso aos componentes do *Framework*;
  - criar os componentes que implementam o *Framework*; e
  - gerenciar a persistência dos objetos construídos com os componentes.

Segue-se uma apresentação da evolução da EAD, das características de alguns ambientes de desenvolvimento de cursos, da metodologia adotada na construção do *Framework* e da arquitetura do *Framework*.

## 1 A Evolução da Educação a Distância

A EAD está fortalecendo e facilitando o aprendizado de estudantes em todo o mundo, proporcionando o compartilhamento de experiências e a transformação da informação em conhecimento (SANCHES, 2000).

A EAD não é um fenômeno novo, mas sim um modo de ensino e aprendizagem utilizado por muitos indivíduos durante pelo menos os últimos cem anos (MOORE; KEARSLEY, 1996). Fatores econômicos, sociais e políticos propiciaram o nascimento e posterior desenvolvimento da EAD.

A história da EAD apresenta registros de experimentos, sucessos e fracassos. Os principais centros de desenvolvimento de EAD começaram na Europa (França, Espanha e Inglaterra) em meados do século XX e se expandiram aos demais continentes. No Brasil, o início da EAD está associado ao rádio. Nunes (1992) destaca a fundação de Rádio Sociedade do Rio de Janeiro em 1923 como o marco inicial no Brasil. Em 1939 surgiu o Instituto Rádio Monitor e em 1941 o

Instituto Universal Brasileiro, destinado aos estudantes sem oportunidade de acesso às instituições educacionais tradicionais, especialmente estudantes com limitações físicas ou residentes em áreas distantes.

A EAD envolve o processo de ensino e aprendizagem com o auxílio de mecanismos e tecnologias para sua implementação e utilização. A principal característica da EAD, que a distingue do ensino presencial, é a separação física do professor e do aluno no espaço e ou tempo (PIMENTAL, 1998).

Em função, principalmente, da tecnologia de comunicação, a evolução da EAD pode ser dividida em três gerações (KEEGAN, 1990):

- a) *geração textual (até década de 1960)*: baseada na auto-aprendizagem com suporte apenas em textos impressos;
- b) *geração analógica (décadas de 1960 a 1980)*: baseada na auto-aprendizagem com suporte em textos impressos, complementados com recursos tecnológicos de multimídia;
- c) *geração digital (a partir de 1980)*: baseada na auto-aprendizagem com suporte, quase que exclusivamente, em recursos tecnológicos.

Holmberg (1985), no entanto, também classifica a evolução da EAD em outras gerações:

- d) *primeira geração*: a de autodidatas, tendo por base tecnologia impressa;
- e) *segunda geração*: a de opções multimídia, utilizando tecnologias de auto-aprendizado, materiais impressos, áudio, vídeo-cassetes e computadores pessoais, combinados de várias maneiras, dependendo das exigências do curso e dos alunos;
- f) *terceira geração*: a de tecnologias interativas, pelas quais os alunos estão ligados entre si em tempo real ou não, possibilitando a simulação de uma sala de aula, utilizando ferramentas de áudio, computadores e videoconferência. Os cursos baseados na *Internet* são um exemplo dessa terceira geração.

Moore e Kearsley (1996), por sua vez, identificam três gerações para a EAD:

- a) *até 1970*: o estudo foi feito através de correspondência, sendo os principais meios de comunicação os materiais impressos, geralmente um guia de estudo, com tarefas ou outros exercícios enviados pelo correio;
- b) *a partir de 1970*: o estudo foi baseado no design e implementação de sistemas de cursos a distância, utilizando material impresso, transmissões por televisão aberta, rádio e fitas de áudio e vídeo, com interação por telefone, satélite e TV a cabo;
- c) *a partir de 1990*: o estudo está fundamentado em redes de conferência por computador e estações de trabalho multimídia.

Nos últimos anos, os educadores testemunharam o rápido desenvolvimento das redes de computadores, melhoramento no poder de processamento dos computadores pessoais e avanços na tecnologia de armazenamento magnético. Esses avanços tornaram o computa-

dor uma ferramenta para EAD, fornecendo meios novos e interativos para superar o tempo e a distância.

O avanço tecnológico dos meios de comunicação contribuiu significativamente para que as instituições educacionais se aperfeiçoassem e desenvolvessem programas de EAD. Dentre os meios de comunicação utilizados, destaca-se a Internet, uma alavanca tecnológica para o ensino a distância que reúne diferentes formas de comunicar idéias e transmitir informações, com uma grande interatividade capaz de centralizar estudantes geograficamente distantes.

A explosão da Internet deve-se principalmente ao sucesso obtido pela WWW, ou simplesmente Web. A Web integra as informações disponíveis na Internet, de diferentes plataformas. Atualmente, várias tecnologias são utilizadas na EAD via Internet, como: correio eletrônico, listas de discussão, *chat*, serviços de teleconferência e ambientes de desenvolvimento de cursos que permitem a integração, comunicação, cooperação e o compartilhamento de informações entre as pessoas.

## 2 Ambientes de Desenvolvimento de Cursos

Os ambientes de desenvolvimento de cursos têm o objetivo de facilitar e agilizar o trabalho do professor em organizar, gerenciar e disponibilizar cursos na Internet. O processo de criação de um curso na Web constitui basicamente duas etapas:

- a) a criação e disponibilização do curso, caracterizado pela elaboração e apresentação do material didático no ambiente Web;
- b) gerenciamento do curso, caracterizado pela realização e manutenção do curso.

Os ambientes, normalmente, mantêm recursos e serviços de comunicação, segurança e acesso a banco de dados, agrupando-os em um único pacote, como exemplo, os ambientes WebCT, AulaNet e *Learning Space*.

O ambiente WebCT foi desenvolvido no Departamento de Ciência da Computação da *University of British Columbia*. Em 1999, o WebCT foi adquirido pela *Universal Learning Technology*, uma empresa de desenvolvimento de plataformas de ensino e aprendizagem baseados na Web. Esse ambiente pode ser usado para criação de cursos on-line ou para publicação de materiais que complementam os cursos presenciais existentes. O ambiente fornece mecanismos que permitem ao professor configurar a organização e aparência das páginas do curso e disponibiliza um conjunto de ferramentas educacionais e administrativas que ajudam o professor no processo de elaboração e gerenciamento do curso.

No WebCT existem quatro classes de usuários:

- a) *administrador*: é o responsável pela criação, inicialização e remoção do curso, e pelo controle de senhas dos usuários. O administrador não inclui o conteúdo do curso, apenas cria o curso e repassa a responsabilidade para o designer;
- b) *designer*: cada curso tem um designer responsável pelo controle do conteúdo e gerenciamento das

atividades do curso. Normalmente, a tarefa do designer é realizada por professores/monitores;

- c) *monitor*: é o responsável pela correção de testes e atualização das notas dos alunos;
- d) *estudante*: são as pessoas cadastradas no curso.

O AulaNet é um ambiente desenvolvido no Laboratório de Engenharia de Software (LES) do Departamento de Informática da Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio).

O AulaNet oferece um conjunto de recursos e serviços que são classificados em três mecanismos, de acordo com suas funcionalidades:

- a) *mecanismos de comunicação*: correspondem aos recursos para o estabelecimento da comunicação entre professores e alunos e entre os alunos;
- b) *mecanismos de coordenação*: correspondem aos recursos para o agendamento das tarefas e para avaliação do aluno;
- c) *mecanismos de cooperação*: correspondem ao material pedagógico utilizado durante a aplicação do curso e que deve ser previamente selecionado pelo autor do curso.

O AulaNet permite várias formas de interação nos cursos através das ferramentas de correio eletrônico, lista de discussão, sistemas de videoconferência e *chat* para aulas e debates ao vivo pela rede. O ambiente considera três atores envolvidos no processo de elaboração e manutenção do curso:

- a) *autor*: é o responsável pela criação e conteúdo do curso, podendo assumir também a função de professor;
- b) *administrador*: é o responsável pelas funções operacionais e gerenciais do ambiente;
- c) *aluno*: é o usuário final, representando o público alvo do curso.

O *Learning Space* é um ambiente desenvolvido pela *Lotus Development Corporation*, uma subsidiária da IBM. O ambiente contém uma ferramenta central e cinco módulos integrados de banco de dados. A ferramenta central auxilia no gerenciamento, na administração e na configuração da aparência do curso. Os módulos, Agenda, Centro de Mídia, Sala de Curso, Perfil dos Participantes e Gerenciador de Avaliação permitem que os usuários realizem atividades, debates, discussões e exercícios de aprendizado.

Esses três ambientes apresentados possuem o propósito de facilitar a tarefa do professor em elaborar e gerenciar um curso a distância na Web. Esses ambientes dispõem de características em comum, e de recursos e serviços que não requerem nenhum conhecimento técnico por parte do autor do curso ou por parte do aluno.

## 3 A Metodologia de Desenvolvimento do Framework para Ensino a Distância

O *Framework* para Ensino a Distância suporta a construção de aplicações para a elaboração e gerenciamento de cursos via Web. O *Framework* visa facilitar a

criação de aplicações no domínio de EAD, principalmente, pelo princípio de reutilização de componentes, aumentando a produtividade do processo de desenvolvimento de software, com menor custo e tempo, e maior flexibilidade de manutenção.

O *Framework* integra tecnologias atuais da Engenharia de Software, na sua especificação e implementação. Para validar o *Framework* foi construído um protótipo cuja evolução segue o modelo espiral do ciclo de vida de sistemas (PRESSMAN, 1995). Estudos de caso estão sendo utilizados para testar o protótipo e fornecer “*feed-back*” para corrigir os erros em todas as fases do ciclo de vida do *Framework*.

No projeto de desenvolvimento do *Framework* para Ensino a Distância foi adotado o método *Catalysis* devido à sua forte ênfase na produção de *frameworks* e componentes. O método *Catalysis* começou em 1991 como uma formalização do OMT e teve influências dos métodos *Fusion* (COLEMAN et al., 1994) e da *Unified Modeling Language* (UML) (FOWLER; SCOTT, 1997; BOOCH; RUMBAUGH; JACOBSON, 1999).

O método *Catalysis* utiliza a notação UML que é independente do processo de desenvolvimento, não se limitando à execução seqüencial das atividades do ciclo de vida. Realiza as atividades de forma interativa e incremental, resultando protótipos que evoluem durante as fases: Concepção, Elaboração, Construção e Transição, numa nova versão do sistema (SANCHES, 2000).

A Figura 1 (SANCHES, 2000) apresenta as atividades para construção do *Framework*: *Identificar Regras de Negócio (RN) do Domínio do Problema*, *Especificar Componentes*, *Projetar Componentes* e *Implementar e Testar Componentes*. Estas atividades se repetem ao

longo do ciclo de vida do *Framework* desde a sua Concepção até a sua Transição.

Na atividade *Identificar Regras de Negócio do Domínio do Problema* é dada ênfase na identificação dos requisitos do sistema no domínio do problema, especificando “o que” o sistema deve fazer para solucionar o problema. Identificam-se os tipos de objetos e ações, agrupando-os em diferentes visões por áreas de negócio. Na atividade de *Especificar Componentes* é dada ênfase na identificação, comportamento e responsabilidades dos componentes, e nas atividades *Projetar Componentes* e *Implementar e Testar Componentes* é dada ênfase na implementação dos requisitos especificados para os componentes do sistema, preocupando-se com a distribuição física destes.

Na atividade *Identificar Regras de Negócio do Domínio do Problema*, baseado no levantamento dos requisitos do *Framework* e, seguindo exemplos de *frameworks* de negócio existentes, o Engenheiro de Software usa a ferramenta CASE *Rational Rose* para especificar as regras de negócio do domínio do problema. Seguindo as técnicas do método *Catalysis*, desta atividade, obtém-se as regras de negócio especificadas em modelos que definem os atores com seus domínios e iterações. Entre as várias técnicas de *Catalysis* para especificar o *Domínio do Problema*, é utilizado o **Diagrama de Colaboração** em vários níveis de abstração. Essa atividade é executada com maior ênfase na fase de Concepção do *Framework*, na qual definem-se os requisitos funcionais e a delimitação do escopo do *Framework*.

Na atividade *Especificar Componentes*, baseado nas regras de negócio especificadas e seguindo o método

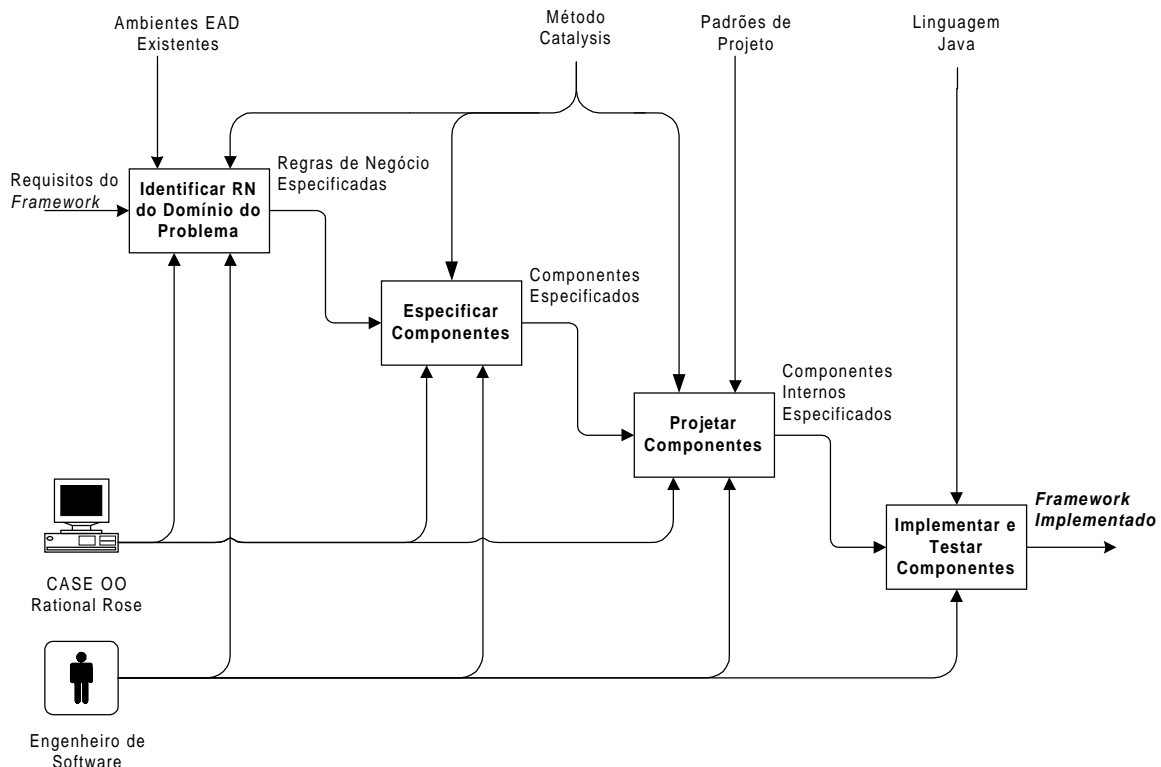


Figura 1 – Atividades das Fases do Ciclo de Vida da Construção do *Framework*.

*Catalysis*, o Engenheiro de Software usa a ferramenta CASE *Rational Rose* para especificar os tipos, agrupando-os em pacotes e definindo os componentes do *Framework*. Seguindo as técnicas de *Catalysis*, desta atividade, obtêm-se os componentes especificados em modelos que definem os pacotes e suas dependências. Entre as técnicas de *Catalysis* para *Especificar os Componentes*, são utilizados o **Modelo do Framework**, o **Modelo de Tipos** e o **Modelo da Aplicação do Framework**. Esta atividade é executada com maior ênfase na fase de Elaboração do *Framework*, na qual define-se o comportamento funcional dos requisitos do sistema.

Na atividade *Projetar Componentes*, baseado nos componentes especificados e, seguindo o método *Catalysis* e Padrões de Projeto, o Engenheiro de Software usa a ferramenta CASE *Rational Rose* para especificar as iterações dos componentes e definir os componentes EJB do *Framework*. Seguindo as técnicas de *Catalysis*, desta atividade, obtêm-se os componentes internos especificados em modelos que definem a sua distribuição física. Entre as técnicas de *Catalysis* para *Projetar os Componentes*, é utilizado o **Modelo de Classes do Sistema** e os **Diagramas de Interações**. Esta atividade é executada com maior ênfase na fase de Construção do *Framework*, na qual define-se a arquitetura executável do sistema.

Na atividade *Implementar e Testar Componentes*, baseado na especificação interna dos componentes EJB, o Engenheiro de Software implementa o *Framework* utilizando a linguagem Java com as tecnologias EJB, RMI, *Servlets*. Através de testes, o Engenheiro de Software valida as funções de acordo com o levantamento dos requisitos de negócio. Esta atividade é executada com maior ênfase na fase de Transição do *Framework*, na qual é disponibilizado o sistema para produção.

Para melhor compreensão do Processo de Desenvol-

vimento adotado apresenta-se a Figura 2 (SANCHES, 2000) que mostra as atividades, relacionadas com as fases do ciclo de vida do *Framework*, ao longo do tempo.

#### 4 Arquitetura do *Framework* para Ensino a Distância

*Frameworks* permitem a geração de sistemas de softwares a partir dos princípios da Orientação a Objetos. Classes abstratas e concretas são reutilizadas ou estendidas, para sistemas específicos no domínio do *Framework*.

O *Framework* para Ensino a Distância suporta os requisitos necessários em uma aplicação de construção e gerenciamento de cursos via Web. Dentre os requisitos destacam-se os recursos e serviços didáticos, administrativos e suporte aos cursos.

O *Framework* utiliza a linguagem Java com EJB, *Servlets* e RMI para a sua implementação. As aplicações construídas a partir do *Framework* têm uma arquitetura *MultiTiered*. A arquitetura *MultiTiered* é utilizada no desenvolvimento de sistemas distribuídos para garantir uma separação e reutilização de suas funcionalidades e facilitar a manutenção. Uma arquitetura *MultiTiered* é composta de três camadas como mostra a Figura 3 (SANCHES, 2000).

Na camada 1 tem-se o Cliente, que pode ser um *browser* Web, um aplicativo Java ou algum dispositivo de armazenamento. Na camada 2 tem-se o Servidor *Middle-Tier* que provê os serviços aos Clientes, acessando dados do sistema, disponíveis na camada 3, e retornando-os para o Cliente que os requisitou. A camada 3 armazena os dados usando Sistema de Arquivos ou Banco de Dados.

A Figura 4 mostra a arquitetura *MultiTiered* integrando o *Framework* com diferentes tecnologias: um *Browser*, *Protocolo HTTP*, *Servidor J2EE*, o *SGBD* e o

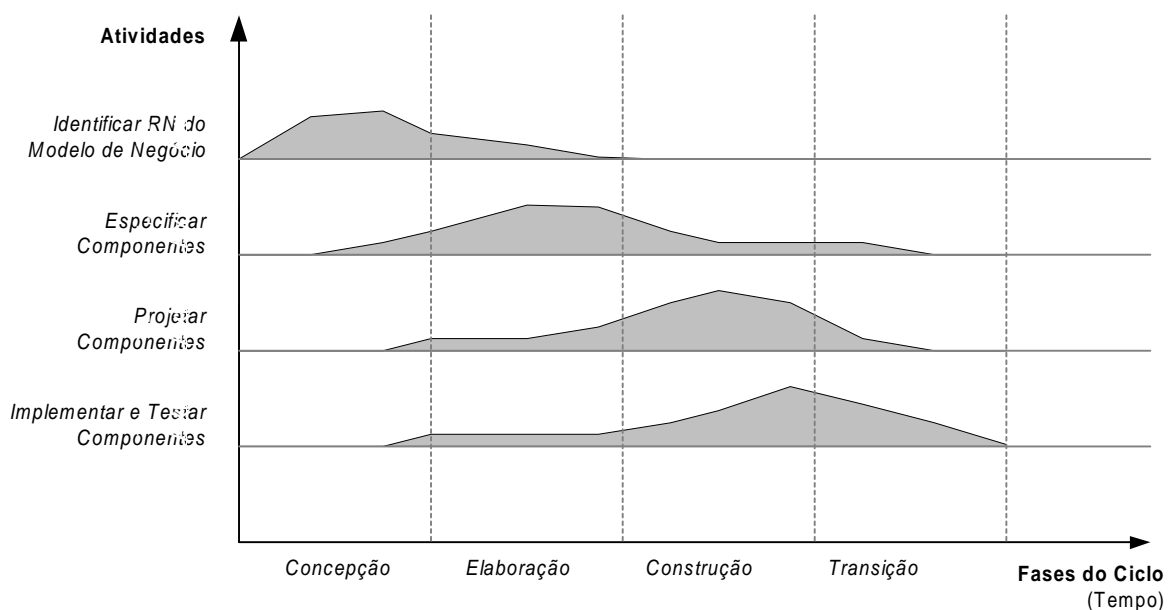


Figura 2 – Atividades e Fases da Construção do *Framework*.

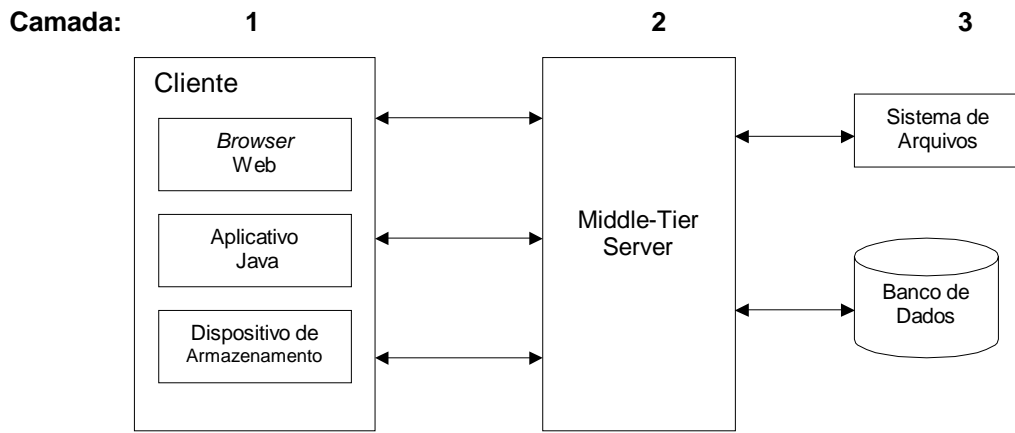


Figura 3 – Arquitetura MultiTiered.

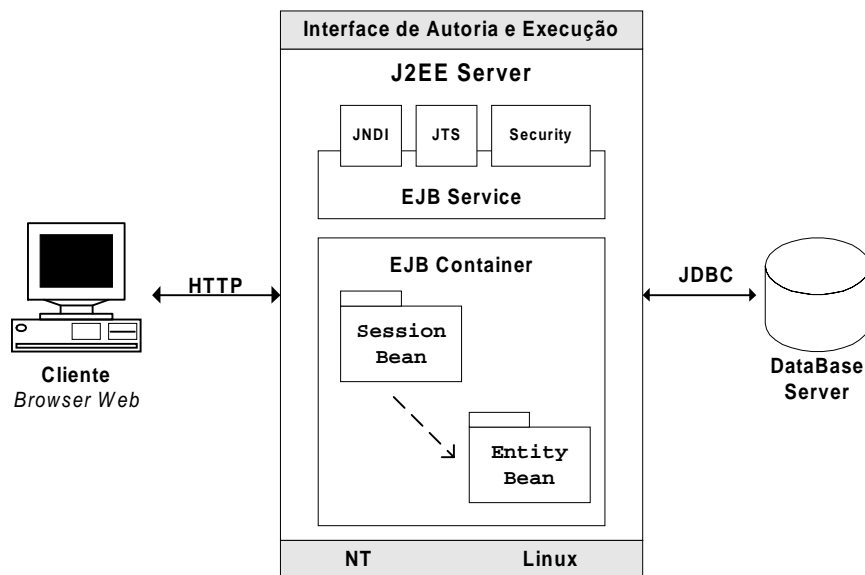


Figura 4 – Arquitetura MultiTiered do Framework.

*Sistema Operacional.* Para facilitar a reutilização, o Framework mantém duas Interfaces para o usuário:

- interface de autoria:* que disponibiliza recursos e serviços para a criação e gerenciamento do curso. É específica para usuários do tipo Coordenador e Professor do curso;
- interface de execução:* que disponibiliza recursos e serviços para realização do curso. É específica para usuários do tipo Aluno.

Aplicações, utilizando o Framework, são executados na Web em um Browser como Netscape ou Internet Explorer. O Servidor Web faz a comunicação do Cliente com o sistema pelo protocolo HTTP.

Na Figura 4, tem-se o Servidor J2EE que fornece serviços de: localização dos componentes, através JNDI que procura a Interface Home do bean para fazer o contato inicial com o Cliente; os serviços de autenticação dos usuários; os serviços de acesso ao Browser com Servlets; e os serviços de gerenciamento às chamadas aos métodos do EJB feita pelo Cliente.

As aplicações EAD, construídas a partir do Framework, usam a arquitetura distribuída J2EE. Esta arquitetura apresenta dois tipos de componentes beans reusáveis: Session ou Entity. O componente bean do tipo Session tem o propósito de realizar uma tarefa para um cliente, sendo criado e destruído pelo cliente. Cada componente Session bean permite o acesso ao Servidor EJB, de apenas um Cliente, tendo o tempo de vida de acordo com a sessão do cliente. Os componentes Entity beans representam objetos em depósitos de dados persistentes. Cada componente Entity bean é associado a uma linha de uma tabela de um Banco de Dados Relacional. A chave primária de uma tabela identifica cada componente Entity bean. Os componentes Entity beans permitem o acesso compartilhado por muitos Clientes, permanecendo ativo por tempo indeterminado.

Cada aplicação EJB possui um Deployment Description que descreve os atributos de transação, autorização de acesso, e outras características de gerenciamento e acesso ao bean. O Servidor J2EE lê esse arquivo em tempo de execução e age de acordo com as informações contidas nele (VALESKEY, 1999).

No servidor EJB, o *Framework* é reutilizado pelas aplicações através dos componentes *Session* e *Entity beans*. O pacote *Session Bean* mantém os componentes que realizam a tarefa para um cliente, como *Gerenciar Acesso*, *Realizar Matrícula*, *Realizar Aula*, *Realizar Avaliação*, *Consultar Avaliação*, *Controlar Agenda* e outros. O pacote *Entity Bean* mantém os componentes que representam os objetos em depósitos de dados, como: *Pessoa*, *Curso*, *Aula*, *Conteúdo da Aula*, *Matrícula*, *Agenda*, *Avaliação*, *Questão da Avaliação* e outros.

Cada *Enterprise Bean* é empacotado num arquivo (< arq >. jar) que inclui os arquivos (< arq >.class) correspondentes às *Interfaces Home* e *Remote* e ao *Enterprise Bean*.

As aplicações EAD construídas a partir do *Framework* podem ser executadas nas plataformas *Windows NT* e *Linux*, utilizando um Banco de Dados Relacional como, por exemplo, o Oracle.

A reutilização dos componentes do *Framework* é feita pelo estabelecimento de um relacionamento entre o componente do *Framework* com a aplicação. O

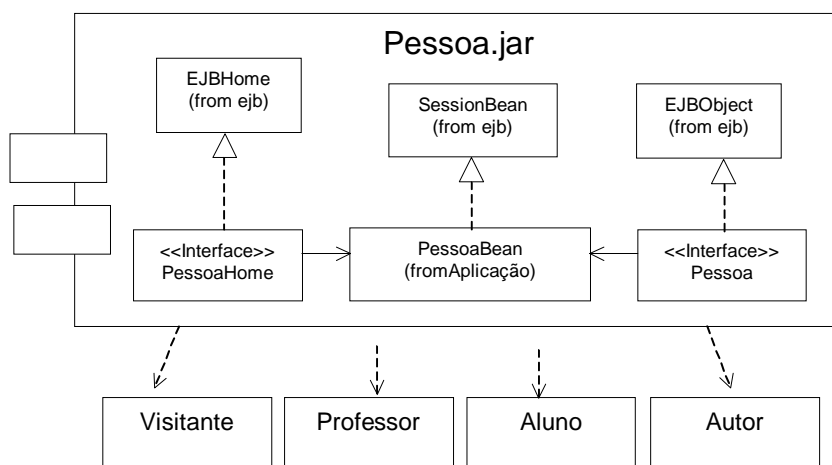


Figura 5 – Componente *bean* do tipo *Entity*.

componente *Pessoa.jar*, por exemplo, suporta a criação de um ator que interage com a aplicação. A Figura 5 mostra um componente *bean* do tipo *Entity*, para uma aplicação que reutiliza o componente *Pessoa.jar* do *Framework* para cadastrar um tipo de ator como: Professor, Aluno, Visitante e Autor.

## Conclusão

O processo de ensino e aprendizado a distância engloba o compartilhamento de informações e a colaboração entre pessoas de diferentes níveis de aprendizado, tendo o professor como um facilitador, guiando e orientando o processo educacional com o auxílio das tecnologias da informação.

Com a explosão da Internet as fronteiras para a EAD se expandiram, possibilitando reunir num só meio de comunicação as vantagens dos diferentes modos de comunicar informações e idéias. Assim, para viabilizar a EAD via Web, são necessários ambientes apropriados para o desenvolvimento e gerenciamento de cursos. Vários ambientes como o AulaNet, WebCT e outros estão sendo desenvolvidos e adotados por instituições educacionais.

Baseado nessa difusão de ambientes e ferramentas para EAD, e na expansão de instituições e empresas que estão adotando a EAD, este artigo apresentou o método de desenvolvimento e a arquitetura de um *framework* para o domínio de EAD que disponha os prin-

cipais recursos para criar e gerenciar cursos na Web. Utilizou-se *Catalysis* como método de desenvolvimento e a tecnologia EJB como plataforma de implementação dos seus componentes. A reutilização de componentes de softwares agiliza o processo de desenvolvimento e assegura funcionalidades já prontas e testadas, facilitando a posterior manutenção da aplicação.

## Referências

- AULANET. *Projeto AulaNet – Pontifícia Universidade Católica do Rio de Janeiro*. Disponível em: <<http://www.les.inf.puc-rio.br/aulanet>>. Acesso em: 30 set. 1998.
- BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. *The Unified Modeling Language – User Guide*. Reading (MA): Addison Wesley, 1999.
- CLASSNET. *Iowa State University Computation Center*. Disponível em: <<http://classnet.cc.iastate.edu>>. Acesso em: 15 abr. 1997.
- COLEMAN, D. et al. *Object-Oriented Development – The Fusion Method*. Englewood Cliffs (NJ): Prentice Hall, 1994.
- D'SOUZA, D.; WILLS, A. C. *Objects, Components and Frameworks with UML – The Catalysis Approach*. Reading (MA): Addison Wesley, 1998.
- FOWLER, M.; SCOTT, K. *UML Distilled. Applying the*

*Standard Object Modeling Language*. [Reading (MA)]: Addison Wesley, 1997.

HOLMBERG, Börje. *Educación a Distancia: Situación y Perspectivas*. Buenos Aires: Editorial Kapeluz, 1985.

KEEGAN, D. *Foundations of Distance Education*. 2<sup>nd</sup> ed. London: Routledge, 1990.

LDB - Leis de Diretrizes e Bases. Art 80 da Lei nº 9.394/96 (20/12/96). Disponível em: <<http://www.intelecto.net/ead/ldb.htm>>. Acesso em: 20 jun. 2001.

LEARNING SPACE. *A Large Scale Implementation of Lotus Learning Space at Staffordshire University*. Disponível em: <[http://www.staffs.ac.uk/ucisa\\_presentation/index.htm](http://www.staffs.ac.uk/ucisa_presentation/index.htm)>. Acesso em: 30 set. 1998.

LUCENA, M.; SALVADOR, V. Learn@Web: um ambiente integrado para aprendizagem cooperativa. In: WORKSHOP DE INFORMÁTICA NA ESCOLA (WIE'99), 5.; CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO (SBC'99), 19., jul. 1999. *Anais...* Rio de Janeiro: Sociedade Brasileira de Computação, 1999. p. 743-758.

MOORE, M. G.; KEARSLEY, G. *Introduction to Distance Education. White paper*. Disponível em: <<http://www.distance-educator.com/intro.htm>>. Acesso em: 2 nov. 1996.

NUNES, I. B. Pequena introdução a educação a distância. *Educação a Distância*, Brasília, n. 1, jun. 1992.

PIMENTAL, N. M.. *Primeira e segunda geração em educação a distância: material impresso, vídeo e teleconferência*. Florianópolis, 1998. Material do Curso

de Capacitação a Distância: Introdução a Educação a Distância – Laboratório de Ensino a Distância, Universidade Federal de Santa Catarina.

PRESSMAN, R. S. *Engenharia de Software*. São Paulo: Makron Books, 1995.

RAJ, G. S. *Enterprise JavaBeans*. Disponível em: <<http://www.execpc.com/~gopalan/java/ejb.html>>. Acesso em: 27 maio 2000.

RATIONAL SOFTWARE CORPORATION. *UML Notation Guide*. Disponível em: <<http://www.rational.com/uml/html/notation>>. Acesso em: 15 ago. 2000.

SANCHES, I. C. *Framework para Ensino a Distância via Web*. 2000. Monografia de Qualificação (Mestrado em Ciência da Computação) – UFSCar, São Carlos (SP).

SUN MICROSYSTEMS. *Tutorias Java*. Disponível em: <<http://java.sun.com/docs/books/tutorial>>. Acesso em: 15 ago. 2000.

UNIVERSITE. *Modern High-Tech Web*. Disponível em: <<http://www.mhv.com.br/universite>>. Acesso em: 30 set. 1998.

VALESKEY, T. *Enterprise JavaBeans – Developing Component-Based Distributed Application*. [Reading (MA)]: Addison-Wesley, 1999.

WebCT. *British Columbia University*. Disponível em: <<http://www.wcb.vcu.edu>>. Acesso em: 15 abr. 1997.

WEB COURSE IN A BOX. *Virginia Commonwealth University*. Disponível em: <<http://www.wcb.vcu.edu/wcb/menus/start.html>>. Acesso em: 15 abr. 1997.