

Afonso Augusto Guimarães Bione

**Solução distribuída de baixo custo para utilização de
Ambientes Virtuais de Aprendizagem**

Recife

2013



Universidade Federal Rural de Pernambuco

Unidade Acadêmica de Educação a Distância e Tecnologia

Pró-Reitoria de Pesquisa e Pós-Graduação

Programa de Pós-Graduação em Tecnologia e Gestão em Educação a Distância

Solução distribuída de baixo custo para utilização de Ambientes Virtuais de Aprendizagem

Dissertação apresentada ao Programa de Pós-Graduação em Tecnologia e Gestão em Educação a Distância como exigência parcial à obtenção do título de Mestre em Tecnologia e Gestão em Educação a Distância.

Área de Concentração: Ferramentas Tecnológicas para Educação a Distância.

Orientador: Prof. Dr. Rodrigo de Souza
DEINFO/UFRPE

Recife
2013

Universidade Federal Rural de Pernambuco
Unidade Acadêmica de Educação a Distância e Tecnologia
Pró-Reitoria de Pesquisa e Pós-Graduação
Programa de Pós-Graduação em Tecnologia e Gestão em Educação a Distância

**Solução distribuída de baixo custo para utilização de
Ambientes Virtuais de Aprendizagem**

Afonso Augusto Guimarães Bione

Dissertação julgada adequada para a obtenção do título de Mestre em Tecnologia e Gestão em Educação a Distância, defendida e aprovada por unanimidade em 23/08/2013 pela Banca Examinadora.

Orientador:

Prof. Dr. Rodrigo de Souza
DEINFO/UFRPE

Banca Examinadora:

Profa. Dra. Juliana Regueira Basto Diniz
Membro Interno - EADTec/UFRPE

Profa. Dra. Sônia Virgínia Alves França
Membro Interno - EADTec/UFRPE

Prof. Dr. Fernando Antônio Aires Lins
Membro Externo - DEINFO/UFRPE

A Deus, a Ana Carolina, minha amada esposa, a minha mãe, Dra. Iza Bione, à Dra. Maria José, a Nilza e Carol, a meus irmãos, familiares e amigos.

AGRADECIMENTOS

Agradeço a todos os que, de alguma forma, contribuíram para a realização desta dissertação de Mestrado.

Ao meu orientador, Prof. Dr. Rodrigo de Souza, por ter acreditado desde o início na realização desta dissertação, mesmo quando ainda era uma ideia que despertava curiosidade e receio entre alguns professores.

Ao meu co-orientador, Prof. Dr. Prof. Francisco Luiz dos Santos, pela confiança em mim depositada, pelas valiosas contribuições e constantes discussões visando ao aprimoramento deste trabalho e pelos ensinamentos de vida aprendidos nesses anos de convivência.

À banca examinadora, Profa. Dra. Juliana Regueira Basto Diniz, Profa. Dra. Sônia Virgínia Alves França e Prof. Dr. Fernando Antonio Aires Lins, pelas valiosas contribuições prestadas.

A todos os meus amigos, em especial a Professor Renato Moraes, a Walmir Soares e Jorge Luiz. Este último contribuiu bastante para a realização desta pesquisa. A Eric de Oliveira. A memória do meu eterno “veio”, Sr. Mendes, Anita, Jobson e colegas, pelo apoio, carinho e amizade demonstrados.

À minha família, fonte constante de inspiração, amor e confiança.

“Eu acredito na intuição e na inspiração. A imaginação é mais importante que o conhecimento. O conhecimento é limitado, enquanto a imaginação abraça o mundo inteiro, estimulando o progresso, dando à luz à evolução. Ela é, rigorosamente falando, um fator real na pesquisa científica.”

Albert Einstein

RESUMO

Neste trabalho estudamos uma integração do *Moodle*, um software de gerenciamento de sistemas de aprendizado, mais conhecido como Ambiente Virtual de Aprendizagem - AVA na Educação a Distância – EAD, com o Postgres-XC, um banco de dados distribuído, em especial, em relação à formação de um pool de computadores interligados em paralelo, o que garante performance, exigida pelo Moodle, a um baixo custo. A metodologia utilizada foi a instalação de 10 computadores de pequeno porte em paralelo, utilizando o sistema operacional livre Linux, que permite o processamento em paralelo dos servidores. Esse protótipo se constitui numa plataforma de software e hardware livres, ou seja, o sistema Ambiente Virtual de Aprendizagem, o AVA, é gratuito, possibilitando o seu uso sem custos de aquisição de software. A partir desse protótipo, foram realizados vários testes em um projeto, verificando-se que o protótipo não demonstrou falhas de usabilidade, disponibilidade e performance do AVA.

Palavras-chave: Moodle. Ambiente Virtual de Aprendizagem. Postgres-XC. Software Livre. Processamento paralelo.

ABSTRACT

In this dissertation we study an integration of Moodle, a virtual learning environment - VLE in Distance Education – DE, with the distributed database system Postgres-XC, aiming at the construction of a pool of computers connected in parallel to ensure performance, as required by Moodle, at a low cost. The methodology used was the installation of 10 small computers in parallel, all running the free operating system Linux, that allow to create parallel processing servers. As a result, we obtain an open, accessible and cheap platform for running VLE's. This prototype has undergone real-world testing and yielded good results concerning usability, availability and performance of the VLE.

Keywords: Moodle. Virtual Learning Environment. Postgres-XC. Free Software. Parallel Processing.

LISTA DE ILUSTRAÇÕES

Figura 1. Diagrama da proposta do <i>Cluster</i>	14
Figura 2. Diagrama interno do Moodle	33
Figura 3. Diagrama de atividades para a instalação do cluster	38
Figura 4. Topologia do <i>Cluster</i> de computadores	40
Figura 5. Topologia do <i>Cluster</i> na nuvem	40
Figura 6. Componentes do Postgres-XC	42
Figura 7. Tela-padrão do servidor Web - Apache	47
Figura 8. Topologia da Solução	49
Figura 9. Foto do protótipo do <i>cluster</i>	52
Figura 10. Tela inicial do protótipo	53
Figura 11. Gráfico de resposta do Jmeter do protótipo Moodle e PostgreSQL	55
Figura 12. Sumário do Jmeter do protótipo com Moodle e PostgreSQL	56
Figura 13. Gráfico de resposta do Jmeter do protótipo Nuvem	57
Figura 14. Sumário do Jmeter do protótipo Nuvem	57
Figura 15. Gráfico das quantidades de conexões x tempo de resposta.....	58

LISTA DE ABREVIATURAS E SIGLAS

ARPANET - Advanced Research Projects Agency Network
AVA - Ambientes Virtuais de Aprendizagem
BACKUP - Cópia de Dados ou de Sistema
CALC - Computer Assisted Learning Center
CMS - Content Management Systems
CNAS - Cisco Networking Academy Management System
DARPA - Defense Advanced Research Projects Agency
DNS - Domain Name System
EAD - Educação a Distância
FOS - Free and Open Source
GPL - General Public License
GTM - Global Transaction Manager
GTM PROXY - Global Transaction Manager Proxy
HTTP - Hypertext Transfer Protocol
IEEE - Institute of Electrical and Electronics Engineers
LB - *Load Balance*
LMS - Learning Management System
LNI - Learning Network Interactive
LTSC - Learning Tecnologia Standards Committee
MIT - Massachusetts Institute of Technology
MOODLE - Modular Object-Oriented Dynamic Learning Environment
OSS - Open Source Software
PLATO - Programmed Logic for Automatic Teaching Operations
SCORM - Sharable Content Object Reference Model
TCP-IP - Transmission Control Protocol - Internet Protocol
TIC - Tecnologias de Informação e Comunicação
UCLA - University of California, Los Angeles

SUMÁRIO

1. INTRODUÇÃO	13
1.1. Considerações Iniciais	13
1.2. Motivação	13
1.3. Problema	15
1.4. Escolhas	17
1.4.1. Linux	17
1.4.2. Moodle	17
1.4.3. Postgres-XC	18
1.4.4. Pool de Computadores	18
1.5. Revisão Bibliográfica	18
1.5.1. Histórico	18
1.5.2. Tecnologias de Informação e Comunicação	21
1.5.3. Conceitos Básicos / Fundamentos Teóricos	23
1.5.3.1. <i>Cluster</i> de Alto Desempenho	24
1.5.3.2. <i>Cluster</i> de Alta Disponibilidade	24
1.5.3.3. <i>Clusters</i> de Balanceamento de Carga	25
1.5.3.4. Computação Paralela	26
1.5.3.5. Topologias de Rede....	27
2. REFERENCIAL TEÓRICO	28
2.1. Modelos Pedagógicos	28
2.2. Ambiente Virtual de Aprendizagem - AVA	30
2.3. Moodle	32
2.3.1. Topologia do Moodle	32
2.4. <i>Cluster</i>	36
3. METODOLOGIA	38
3.1. Preparação do Ambiente de Rede Física e Nuvem	38
3.2. Instalação do <i>Cluster</i>	39
3.3. Instalação na Nuvem	40
3.4. Instalação do Banco de Dados Postgres-XC	41

3.5. Instalação do Servidor Web - Apache no <i>Cluster</i>	46
3.6. Instalação do Balanceamento de Carga	47
3.7. Instalação do Servidor de Alta Disponibilidade	48
3.8. Preparação do Cluster para Área de Dados dos Usuários - NFS	49
3.9. Instalação do Moodle	50
3.10. Preparação do Ambiente Virtual de Aprendizagem	51
4. AVALIAÇÃO	54
4.1. Testes	54
4.1.1. Teste em Ambiente com Moodle e PostgreSQL	55
4.1.2. Teste no <i>Cluster</i> Local com Moodle e Postgres-XC	56
4.1.3. Testes na Nuvem com Moodle e Postgres-XC	56
4.2. Sumário dos Resultados	57
5. CONCLUSÃO	59
REFERÊNCIAS	60

1. INTRODUÇÃO

1.1. Considerações Iniciais

Propomos nesta dissertação uma solução para a construção de uma infraestrutura de rede que visa à possibilidade da sua utilização como uma nuvem de educação com alto poder de processamento e a baixo custo. A escolha do tema deve-se, principalmente, às dificuldades para obter-se uma plataforma escalável, sem representar um grande investimento para manutenção e implementação dessa grande rede.

Para potencializarmos o uso do Ambiente Virtual de Aprendizagem (AVA) Moodle nessa rede, utilizamos uma instalação inédita (segundo nosso conhecimento). Foi instalado, em nosso protótipo, o sistema gerenciador de banco de dados Postgres-XC, um clone do sistema PostgreSQL desenvolvido em 2010 que fornece uma solução aberta de gravação escalável, síncrona e multiprocessada, permitindo utilizar o banco de dados *em paralelo* (conseguido junto aos seus desenvolvedores). O banco de dados era o único componente de nosso sistema que ainda não executava em paralelo, e a principal dificuldade em nossa proposta foi o fato de que desconhecíamos experiências de instalação do Moodle sobre o Postgres-XC. Cabe mencionar que a escolha do Postgres-XC não foi por acaso, já que o Moodle utiliza nativamente o PostgreSQL como banco de dados.

O AVA Moodle, em sua configuração padrão na versão 2.x, consome uma significativa quantidade de recursos, exigindo hardware caro, conforme podemos verificar em estudos divulgados no site <http://www.iteachwithmoodle.com/>.¹ Nossa proposta principal para enfrentar esse problema é montar um *cluster* de baixo custo e que seja eficiente, numa infraestrutura que poderá ser utilizada com a união de computadores modestos, permitindo a instituições de EAD de portes diversos a implementação de uma rede elaborada com grande poder de processamento em razão do paralelismo de nossa solução. Vale salientar que esse modelo também pode ser utilizado mediante virtualização sem alterar o modo e a integração dos sistemas envolvidos. Por exemplo, pode-se usar essa solução em uma rede de computadores na nuvem, a “*cloud computing*”.

1.2. Motivação

Conforme discutimos, a EAD aparece como solução educacional eficaz em um

¹ Conferir por exemplo <http://www.iteachwithmoodle.com/2012/10/12/moodle-performance-testing-how-much-more-horsepower-do-each-new-versions-of-moodle-require/>.

contexto continental e desigual como o Brasil. Ao mesmo tempo, a viabilidade do uso da EAD em larga escala depende de uma infraestrutura combinando robustez, qualidade, escalabilidade e custo acessível. É esse problema que propomos abordar nesta pesquisa.

O Moodle contempla os itens acima levantados, com exceção do baixo custo de implementação, especialmente em situações de grande número de estudantes interconectados simultaneamente e por uma sessão longa, como é o caso do chat. Dessa forma, exige um servidor elaborado, em termos de memória e processador, além de redundância, seja da parte elétrica, seja do armazenamento dos dados, tornando-se dispendiosa a sua implementação.

O uso do sistema operacional Linux, no qual o Moodle pode ser naturalmente instalado, fornece uma primeira solução a esse problema. Além de ser gratuito e de código aberto, o Linux possui recursos adequados para a implementação de *clusters* em paralelo – e o Moodle é adaptado para aproveitar uma arquitetura paralela.

Dessa forma, este trabalho investiga a construção de um *cluster* em paralelo, integrando o Moodle, o Postgres-XC e o Linux, a fim de prover um AVA de alta performance e de baixo custo. A motivação está na possibilidade de se colocar à disposição da sociedade uma ferramenta de *cluster* que dê suporte à criação de uma nuvem de dados educacionais de baixo custo, descentralizados e integrados, conforme podemos observar na figura 1. Na parte superior, percebe-se a primeira camada de aplicação, na qual estão instalados o servidor web e o Moodle, enquanto na parte amarela identifica-se a arquitetura do *cluster* de banco de dados.

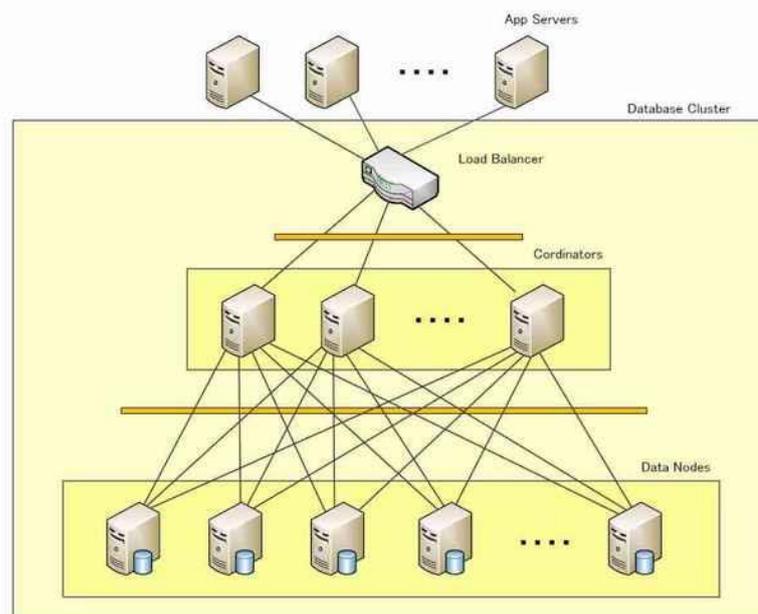


Figura 1. Diagrama da proposta do *Cluster*

Fonte: postgresxc.org

1.3. Problema

Originalmente, a web ou a conhecida WWW, era essencialmente constituída de conteúdos estáticos, rapidamente entregues para alguns poucos usuários que gastavam mais tempo em leitura do que em raros cliques. Agora, existem aplicações de todos os tipos e serviços desde acesso remoto a estoque de uma grande empresa de logística à emissão de segunda via de contas bancárias, exigindo assim que os usuários fiquem “presos” por dezenas de minutos ou horas, com pouco conteúdo para ler entre os cliques e muito do trabalho realizado nos servidores.

Os usuários estão acostumados a receber rapidamente as páginas do servidor web. Eles esperam uma entrega imediata, enquanto inconscientemente infligem grandes cargas de trabalho sobre os servidores a cada clique do mouse. Essa nova dinâmica tem desenvolvido uma necessidade, que é a de alto desempenho e disponibilidade permanente dos servidores da Internet. Conforme Coelho (2010), o indicador de performance pelo qual estamos interessados é o que o utilizador final sente: o tempo de carregamento de uma página. Se esse tempo for rápido, cerca de 1 segundo, o utilizador sente-se confiante e utiliza, sem problemas, a plataforma; se for lento, mais de 10 segundos, pensará antes de cada clique, provavelmente irá abrir várias janelas e sentir que perde tempo para poder utilizar a plataforma. É de fundamental importância manter esse tempo de carregamento de página sempre baixo, evitando degradações no sistema e frustração da experiência de utilização da plataforma pelo estudante.

Uma das vantagens de se usar um software livre é a possibilidade de alteração no código na medida em que se exige alguma intervenção muito específica. Com essa filosofia de construção por módulos independentes, o Moodle transformou-se em um padrão mundial, em que, a cada dia, surgem módulos para praticamente todas as necessidades. Por outro lado, a cada módulo instalado no servidor, mais “pesado” este se torna, e o mesmo pensamento é para a quantidade de usuários conectados simultaneamente. Uma boa instalação do Moodle deverá considerar esses fatores.

A filosofia do Moodle prevê essa situação de crescimento, e a sua arquitetura facilita a sua escalabilidade e flexibilidade para a distribuição de tarefas. Mas com o acréscimo de módulos e usuários simultâneos, a instalação precisa em geral ser reformulada, o que muitas vezes exige um aumento no número de servidores, processadores e memória, acarretando obrigatoriamente um aumento nos custos de operação. Segundo Cristiano (2012), para a quantidade massiva de usuários conectados em usos típicos, o Moodle torna-se uma

plataforma que exige, em geral, aquisição de equipamentos caros e manutenção de hardware de valor alto, a fim de permitir um funcionamento eficiente e sem falhas, principalmente quando se utiliza de serviços que exigem um “estado”², ou seja, a necessidade de saber exatamente o que o usuário remoto está fazendo e qual foi a sua última utilização do sistema, que é o caso do serviço de chat, e quando vários usuários acessam simultaneamente o sistema, necessita criar vários controles de estado individuais, tornando-o muito lento.

Como exemplo, podemos projetar que, a fim de atender a uma demanda total de 25.000 alunos, seria necessária, em uma estrutura tradicional do Moodle, a aquisição de um chassi de servidores Blade com, no mínimo, 960 Gb memória, e, no mínimo 4 (quatro) lâminas, a um custo aproximado de R\$ 400.000,00 reais.

Abordando esse problema, foi testado pela Waiariki Institute of Technology por Benner (2010) a construção de um *cluster* de alta performance e de baixo custo. Tomando-se o nosso exemplo, caso se utilizasse o Moodle sobre um *cluster*, o custo passaria a ser aproximadamente de R\$ 60.000,00 (projeções nossas). Dependendo das instalações pré-existentes, o custo pode ser menor ainda. Tal é nossa proposta: implantamos a plataforma Moodle sobre a solução de Benner em um protótipo utilizado na EADTec da Universidade Federal Rural de Pernambuco.

É preciso mencionar que essa solução já foi discutida na documentação oficial do sistema Moodle; citamos por exemplo o documento disponível em http://docs.moodle.org/22/en/Performance_recommendations (acesso em 23 de agosto de 2013). Todavia, não encontramos, na literatura, nenhuma análise de seu potencial de aplicação, principalmente no que tange à utilização do banco de dados distribuído nos nós da rede, a fim de aumentar a performance e maximizar o uso dos recursos de memória e processador dos outros computadores em detrimento da utilização de um único servidor de banco de dados centralizado e limitado. Dessa forma, mantemos a premissa da nossa inovação que é um estudo da escalabilidade a baixo custo.

Assim, nossa principal contribuição é apresentar uma integração entre o Moodle (AVA) e um banco de dados distribuído (Postgres-XC) que comprove a eficiência dessa arquitetura e chamar a atenção para essa solução, visto que comumente ela não é adotada devido à ausência de literatura científica (segundo nosso conhecimento).

² RFC 2068 - The Hypertext Transfer Protocol (HTTP) is an application-level for distributed, collaborative, hypermedia information systems. It is a generic, stateless, object-oriented protocol which can be used for many tasks, such as name servers and distributed object management systems, through extension of its request methods.

Esta dissertação de mestrado tem como objetivo principal integrar o Moodle, um Ambiente Virtual de Aprendizagem bastante difundido, com o Postgres-XC, um sistema gerenciador de bancos de dados distribuído, em um *cluster* de computadores interligados em uma solução distribuída de baixo custo.

1.4. Escolhas

A fim de sistematizar a discussão feita até este ponto, apresentamos a seguir as principais razões que levaram à escolha da tecnologia empregada em nossa solução.

1.4.1. Linux

O Linux é um software livre, ou *Free Software*. Conforme a definição disponível em <http://www.gnu.org/philosophy/free-sw.pt.html>, e criada pela *Free Software Foundation*³, software livre é o software que pode ser usado, copiado, estudado, modificado e redistribuído sem restrição, portanto sem custo de software. A forma usual de um software ser distribuído livremente é de ser acompanhado por uma licença de software livre (como a GPL ou a BSD), e com a disponibilização do seu código-fonte.

Conforme Barbalacel (2008), podemos afirmar que o Linux atualmente pode ser considerado um sistema operacional de tempo real, fácil na sua utilização e pode ser usado para muitas aplicações que toleram atrasos ocasionais na resposta do sistema. Essa propriedade de performance foi considerada na escolha do Linux.

1.4.2. Moodle

Dentre os principais Ambientes Virtuais de Aprendizagem em uso, podemos destacar os seguintes:

- a) Aulanet (http://www.eduweb.com.br/elearning_tecnologia.asp);
- b) Amadeus (<http://amadeus.cin.ufpe.br/index.html/>);
- c) Eureka (<http://eureka.pucpr.br/entrada/index.php>);
- d) Moodle (<http://moodle.org/login/index.php>);
- e) E-Proinfo (<http://eproinfo.mec.gov.br/>);
- f) WebCT (<https://www.elc.uga.edu/webct/entryPageIns.dowebct>);
- g) *LearningSpace* (<http://openlearn.open.ac.uk/>);
- h) *Blackboard* (<http://www.blackboard.com/>);

³ Consulte em <http://www.fsf.org>.

Desses AVAs, destacamos o *Blackboard*, que um é software proprietário, ou seja, o código-fonte não está disponível para adaptações e deve ser adquirido. Todavia, segundo Machado (2007), o sistema de gestão de aprendizagem Moodle é mais eficaz e eficiente que o sistema de gestão de aprendizagem *Blackboard*. Além desse fato, o código-fonte do Moodle está disponível para adaptações o que, no nosso projeto, é uma prerrogativa importante.

Já o Amadeus também tem código fonte disponível, escrito em Java, e a nossa intenção é testar essa solução em trabalhos posteriores; porém um fator a favor do uso do Moodle em nossa solução foi, além das recomendações de Machado (2007), o fato de *termos conhecimento da ferramenta há 6 anos*.

1.4.3. Postgres-XC

Postgres-XC é um projeto de software aberto para fornecer uma solução escalável de leitura e gravação, síncrona, simétrica, funcionando em um pool de computadores. É um clone do PostgreSQL, o que para o nosso projeto é um fator importante, pois o Moodle já vem preparado para utilizar o PostgreSQL, facilitando a integração.

Seus desenvolvedores disponibilizaram o software para a comunidade no endereço <http://postgresxc.org/>. O Postgres-XC é um conjunto de componentes de dados, fortemente acoplados, que podem ser instalados em um pool de computadores ou máquinas virtuais, utilizando a nuvem de computadores, fato que nos levou à sua escolha.

1.4.4. Pool de Computadores

A ideia principal é a integração do Moodle com um banco de dados distribuído. Para isso, utilizamos não só um *cluster* em um protótipo com 10 equipamentos interligados mas também 4 computadores na nuvem em outro protótipo, comprovando que a solução de integração do Moodle com o Postgres-XC funciona tanto em *cluster* de computadores como em nuvem de computadores.

1.5. Revisão Bibliográfica

1.5.1. Histórico

Os Ambientes Virtuais de Aprendizagem são as principais ferramentas abordadas neste trabalho. A fim de apresentá-las, começamos com um breve histórico da Internet e de como a educação foi influenciada pela rede mundial.

Com origens fundamentais na explosão da revolução científica e tecnológica, logo após a Segunda Grande Guerra e associado ao *boom* de informações produzidas pelos países naquela época, o movimento mundial de globalização, principalmente em termos de mercado, criou condições propícias para o surgimento da Internet. As grandes corporações desenvolviam modelos de industrialização espalhados pelo mundo. As potências mundiais expandiram o seu capital, e as suas tecnologias eram adotadas da mesma forma que no país de origem.

Ao mesmo tempo, a procura por uma comunicação mais eficaz e rápida se tornou uma verdadeira obsessão, pois a globalização era uma realidade, mas a comunicação não avançava na mesma direção.

Com o surgimento dos protocolos de comunicação TCP-IP, HTTP⁴ e outros, o mundo, de fato, se tornou global. A união da tecnologia da informação com a tecnologia de comunicação, por meio dos protocolos de comunicação, possibilitou a conectividade entre as redes locais. Segundo Dantas (2002), uma rede local poderá usar recursos compartilhados de outra rede local, formando-se assim uma comunidade de redes locais. E cada rede se torna então o que chamamos de nó de rede. Através da conexão dessas redes, nasceu a “redes das redes”, a Internet. A Internet foi o resultado de um pensamento visionário de acadêmicos e militares no início dos anos 1960, que viram grande potencial ao permitir que computadores pudessem compartilhar informações sobre a pesquisa e o desenvolvimento em campos científicos e militares.

Nesse mesmo período, amadurecia a necessidade de uma ferramenta computacional que desse suporte à grande globalização da economia (enquanto que o mundo já possuía ferramentas de controle da automação de máquinas e processos, que serviram de marco para a revolução industrial). Para solucionar essa lacuna de dificuldade de comunicação entre países, localidades e empresas, surgiram várias ferramentas, visando facilitar a comunicação, tendo a Internet como principal meio.

Como a Internet estava intrinsecamente ligada à academia, pois nascera dentro das universidades, surgiram também ferramentas educacionais, que se tornaram marco na história da educação, por meio de sistemas de informação. Alguns desses marcos históricos estão relacionados a seguir (mais informações podem ser consultadas em http://docs.moodle.org/25/en/Online_Learning_History):

⁴ Para maiores informações sobre esses protocolos consulte em: Redes de Computadores - 5ª Ed. - 2011, J. Wetherall, David; Tanenbaum, Andrew S.

- **1960:** PLATO - Programmed Logic for Automatic Teaching Operations (lógica programada para operações automáticas de ensino), desenvolvido na Universidade de Illinois em Urbana-Champaign. O sistema permaneceu em funcionamento até meados de 1990. Considerado um marco, pois sua contribuição deixou marcas até hoje, utilizou conceitos de e-mail, lista de discussão entre outras ideias que fazem parte do aprendizado online atualmente;
- **1962:** Richard Buckminster Fuller publica o livro *Automação na Educação*;
- **1969:** Criação da *Advanced Research Projects Agency Network* - ARPANET e logo após a fundação da Internet.
- **1979:** Criação da USENET, uma das redes mais antigas de computadores do mundo, antes mesmo do surgimento da World Wide Web. No início, as ligações entre computadores eram feitas dentro das universidades. Essa conectividade foi estabelecida utilizando o protocolo de comunicação, chamado UUCP entre a Universidade Duke e a UNC por Tom Truscott, Jim Ellis, e Bellovin Steve.
- **1982:** O Computer Assisted Learning Center - CALC foi fundado para promover a entrega de conteúdo educacional por meio do uso de computadores interligados em redes que, alguns anos mais tarde, se tornaria um dos primeiros modelos de aprendizagem em rede;
- **1995:** O CALC Campus Online mudou-se para a Internet como CALCampus.com, sendo o primeiro a desenvolver e implementar o conceito de uma escola totalmente online, por meio da qual a administração do ambiente, sala de aula em tempo real e os materiais didáticos eram disponibilizados no campus QuantumLink;
- **1997:** O *Learning Network Interactive* – LNI foi lançado e instalado em diversas instituições acadêmicas, incluindo Cornell University, Yale Medical School e Universidade de Pittsburgh. LNI foi o primeiro sistema de e-learning de seu tipo para implementação sobre um banco de dados relacional MySQL, o que deu início ao acompanhamento detalhado do estudante;
- **1997:** A WebCT 1.0 foi lançado. É um Ambiente Virtual de Aprendizagem, proprietário que depois viria a ser comprado pela Blackboard;
- **1997:** A Blackboard foi fundada. A Blackboard é um Ambiente Virtual de Aprendizagem, cujo software é proprietário;
- **1998:** Martin Dougiamas, autor do Moodle, começa o trabalho preliminar sobre esse software;
- **1998:** Cisco Networking Academy Management System (CNAMS) é lançado para facilitar a comunicação e o curso de gestão da maior iniciativa de blended learning⁵ de seu tempo, a

⁵ *Blended Learning* é uma modalidade de educação a distância mista, presencial e a distancia.

Cisco Networking Academy. Ele inclui ferramentas para manter listas, livros de classificação, fóruns, bem como uma solução escalável, um robusto engenho de avaliação;

- **1999:** Um ano depois de iniciar os trabalhos, Martin Dourgiamas fez os primeiros ensaios iniciais do Moodle. O papel de Martin Dourgiamas era melhorar a eficácia das ferramentas de educação baseada na Internet, os detalhes e um estudo de caso, incluindo telas do sistema para o protótipo;
- **2001:** Moodle.com e o Moodle 1.0 é lançado;
- **2005:** Lançado o Sakai, um Ambiente Virtual de Aprendizagem, escrito na linguagem Java, de software livre;
- **2006:** Moodle 1.6 é lançado;
- **2006 – 2013:** Criadas plataformas mundiais de aprendizagem na Web, reunindo ofertas de cursos de grandes universidades, como o Coursera e a Khan Academy, levando ao conceito de MOOC – *Massive Open Online Course*.

Essas novas ferramentas trouxeram uma mudança significativa nos métodos anteriores de educação a distância, porque o aluno a distância não está mais isolado do professor e dos colegas de classe. Os alunos podem participar de discussões em grupo, fazer experiências, ouvir palestras, fazer perguntas, participar como voluntários em experimentos, realizar pesquisas bibliográficas, procurar publicações de revistas originais, fazer trabalho em equipes de projeto, realizar estatísticas e análises, como também preparar e apresentar bem o trabalho formatado individual ou em conjunto, preparar apresentações em sessão plenária e, até mesmo, se socializar e bater papo, tudo sem sair de suas casas. Esse foi um afastamento significativo dos métodos tradicionais de educação a distância, porque o aluno a distância não mais permaneceria isolado do professor e dos colegas de classe.

Essas novas ferramentas provocaram uma quebra do paradigma da forma de aprendizado desenvolvido antes dessa revolução educacional via web. Agora a mediação passa a ser o centro entre o aluno, o tutor e o professor.

1.5.2. Tecnologias de Informação e Comunicação

Conforme (ABDI, 2010), Tecnologias de Informação e Comunicação – TIC correspondem a todas as tecnologias que interferem e perpassam os processos informacionais e comunicativos dos seres humanos.

Entendemos as TIC como sendo o meio em que a informação e a comunicação serão transmitidas por meio de diversos tipos de redes e tecnologias tais como, a rede de telefonia, a

Internet, a TV digital, a rádio difusão, as redes sem fio, os satélites, a realidade aumentada, 3D, a realidade virtual, a imersão – técnica de “inclusão total” do aprendiz sob uma determinada ótica, muito utilizado na aprendizagem de idiomas e muitos outros, utilizando-se de hardware e software para essa finalidade.

As TIC podem ser compreendidas como um conjunto de recursos tecnológicos integrados entre si, que proporcionam, por meio das funções de hardware, software e telecomunicações, a automação e comunicação dos processos de negócios, dos serviços públicos, da pesquisa científica e de ensino e aprendizagem.

As TIC vêm contribuindo para a transformação na forma de produzir, distribuir e adquirir matérias-primas para a produção de bens e serviços, além de mudar a percepção de espaço e o tempo no mundo contemporâneo.

Com o advento das TIC, a educação a distância, que nos primórdios, atuava por correspondência, material impresso, agora se apropria de uma nova tecnologia, que possibilitou a criação dos Ambientes Virtuais de Aprendizagem (AVA).

Esse meio eletrônico informatizado com software, hardware, redes sociais, Inteligência Artificial e outros forma a base para a EAD, e o uso correto dessas ferramentas nos leva a novas formas de se fazer a educação chegar, com mais qualidade e eficiência, aos alunos.

As TIC estão revolucionando a sociedade, principalmente quando permitiram, ao encurtar as distâncias, que um antigo vilão da sociedade trabalhadora, desde a época da divisão do trabalho de Adam Smith, a industrialização, mecanização e processos tecnológicos, fosse vencido pelo conhecimento fornecido pela própria tecnologia, a Informação (ALMEIDA E MORAN, 2005).

Ao ligarmos e projetarmos uma informação, temos uma tecnologia de informação; com a mediação, explanação e/ou leitura do professor, a comunicação, e a esse conjunto, temos o meio pelos quais a tecnologia da informação e comunicação é repassada.

Sem dúvida, a Internet está, cada vez mais, sendo utilizada como meio de comunicação e de mídias e, na educação, isso tem se evidenciado de forma bastante significativa nos últimos anos. E com as TIC vieram a inclusão social através de mais uma ferramenta alicerçada na Internet, a Educação a distância - EAD. Evidentemente que já havia educação a distância antes do surgimento da Internet, com a existência dos cursos por correspondência, rádio, TV, etc., entretanto, nesse momento, estamos destacando a possibilidade de levar o conhecimento on-line através das TIC:

“Tecnologia de informação e comunicação (TIC) tornou-se, dentro de um período curto de tempo, um dos blocos básicos de construção da sociedade moderna” (UNESCO, 2002).

Falar de Educação a distância - EAD é falar das TIC e saber que sem o meio capaz de unir a comunicação e a informação, voltaríamos muito no tempo. Na EAD da atualidade, as TIC são fundamentais. É nelas que tudo acontece, não só em termos de inovação como também em relação ao próprio funcionamento da EAD.

As TIC evoluíram no intuito, também, de dar suporte às várias metodologias de ensino e aprendizagem para a EAD, provocando um certo desconforto aos educadores da modalidade presencial, pois elas tornaram mais evidentes certas inconsistências da metodologia praticada.

O desafio na EAD é a percepção do corpo pedagógico sobre como e quais visões vão ser utilizadas em um determinado conteúdo, e para tanto as TIC têm ajudado muito na construção de uma plataforma de aprendizado em que os pedagogos possam imbricar, todas essas visões em um mesmo ambiente. Temos como exemplo de facilitador os Objetos de Aprendizagem, que são recursos pedagógicos capazes de construir conteúdos com várias características, como material didático, ambiente virtual, avaliações, tarefas e atividades, conteúdos esses capazes de trazer benefícios ao processo de ensino e aprendizagem.

A utilização da Educação a Distância, além de estar mudando a forma de pensar a Educação, trouxe um mundo novo, ampliando o universo da educação, pois temos novas possibilidades como a transposição do tempo e da distância, facilitando bastante a educação em um país com dimensões continentais como o nosso.

Com inúmeras ferramentas existentes e o aumento constante da banda larga de comunicação em nosso país, podemos assim utilizar, na infraestrutura da Internet, um Ambiente Virtual de Aprendizagem totalmente configurado, modulado e estruturado, para atender a vários modelos pedagógicos existentes, propiciando uma melhor abordagem ao projeto político-pedagógico do curso a distância.

1.5.3. Conceitos Básicos / Fundamentos Teóricos

Nesse item, abordaremos os conceitos técnicos principais para nosso trabalho referentes a redes de computadores.

1.5.3.1. *Cluster de Alto Desempenho*

Um *cluster* de alto desempenho é direcionado a aplicações bastante exigentes no que diz respeito ao processamento. São sistemas com um significativo número de processadores em geral, oriundos de um grande número de computadores separados, conectados através de uma rede, que podem dedicar algum ou todo seu tempo de processamento para resolver um problema comum.

A crescente disponibilidade de recursos de computação comerciais na nuvem nos últimos anos chamou a atenção da computação de alto desempenho (HPC) e na comunidade de computação científica. Muitos pesquisadores têm examinado o desempenho computacional relativo de ofertas disponíveis no mercado de computação em nuvem e têm obtido grandes resultados, conforme podemos encontrar mais detalhes em (REIS, VIEIRA e TIMM, 2006). Computação na nuvem ou *Cloud Computing* refere-se tanto a aplicações disponíveis na Internet quanto a hardware e software nos datacenters que oferecem esses serviços.

A *Cloud Computing* nesses moldes pode ser pública e privada. *Cloud* pública ou nuvem pública acontece quando utilizamos ou disponibilizamos um serviço em uma estrutura na Internet em que o acesso ao datacenter é feito remotamente, através de conexões utilizando a própria Internet para utilizar esses serviços. *Cloud* privada ou nuvem privada é composta por um datacenter com servidores, nobreaks equipamentos de rede, links etc., cujo acesso às máquinas servidoras pode ser feito também em loco, ou seja, pode haver intervenções presenciais.

1.5.3.2. *Cluster de Alta Disponibilidade*

Clusters de alta disponibilidade são ambientes de computação configurados para fornecer alto grau de disponibilidade do serviço, conhecidos comumente como sistemas de alta disponibilidade. Tais sistemas normalmente têm hardware redundante, ou seja, duplicados e software que permitem ao sistema permanecer disponível, apesar de falhas que possam acontecer. Um sistema de alta disponibilidade bem concebido evita pontos únicos de falha que causem a interrupção do sistema. O ideal é que qualquer componente de hardware ou software que possa falhar tenha um componente redundante do mesmo tipo, conforme (NICOLA, 2000).

O equipamento e o software redundante testa constantemente o ambiente ativo pela rede para que, no caso de uma falha, ele possa assumir automaticamente o lugar do outro serviço. Isso é feito por meio de uma implementação chamada *HeartBeat*, que significa a

pulsação de um token, um sinal enviado para o servidor ativo, e o sistema redundante aguarda uma resposta predefinida. Caso a resposta não venha, então, nesse momento, o servidor redundante corta a comunicação com o servidor ativo e assume o seu lugar. Nesse momento, também o servidor redundante envia um alerta via e-mail para os administradores da rede avisando da ocorrência de falha do sistema principal. Mais detalhes podem ser encontrados em (SCHMIDT, 2006).

Clusters de alta disponibilidade são utilizados em sistema de missão crítica, aqueles que precisam permanecer funcionando, exigindo uma disponibilidade de 99,999%, também conhecido como “five 9s” do tempo de uso, conforme podemos encontrar disponível em: <<http://searchdatacenter.techtarget.com/definition/highavailability>>. Acesso em: 23 ago. 2013. O tempo de inatividade, seja planejada ou não, traz consigo custos consideráveis. Soluções para garantir níveis mais elevados de disponibilidade têm sido tradicionalmente caras, difíceis de implementar e de gerir.

O Linux torna mais simples e menos dispendioso a implementação de níveis mais altos de disponibilidade para importantes aplicações. Existe hardware específico para *Load Balance*, alta disponibilidade e outros que são chamados de *appliances*. No caso do Linux, os serviços de *Load Balance* e alta disponibilidade são configurados via software barateando os custos e tornando possível reduzir o tempo de inatividade planejado e não planejado dos servidores mais detalhes também em:(TARREAU, 2006).

Como benefício dessa arquitetura, podemos ainda ter uma tranquilidade maior quanto à necessidade de intervenções programadas no servidor, como manutenções preventivas.

1.5.3.3. *Clusters* de Balanceamento de Carga

Segundo Dantas (2002), em *clusters* de balanceamento de carga as tarefas de processamento são distribuídas o mais uniformemente possível entre os nós. O balanceamento de carga é um método de redes de computadores para a distribuição de cargas de trabalho em vários computadores ou em um *cluster* de computadores, conectados em rede.

Um balanceamento de carga bem sucedido otimiza o uso dos recursos, maximiza o rendimento, minimiza o tempo de resposta e evita sobrecarga, até mesmo a paralisação do sistema.

Utilizando múltiplos computadores para o balanceamento de carga em vez de um único componente, é possível aumentar a confiabilidade do sistema através da redundância. O balanceamento de carga é normalmente fornecido pelo software ou hardware dedicado, como

um *switch multilayer* ou um processo do servidor de nomes chamado *Domain Name System* – DNS. Mais detalhes em (DANTAS, 2002).

Um método alternativo de balanceamento de carga, que não requer necessariamente um software dedicado ou nó de hardware, é denominado de *round robin* DNS, solução usada no nosso projeto. Nessa técnica, vários endereços IP estão associados a um único nome de domínio, ao qual os usuários do sistema fazem a solicitação via navegador ao endereço único do AVA, e, nesse momento, o servidor de balanceamento de carga verifica o servidor, dentre aqueles que compõe o pool, que está livre naquele momento, na sequência de solicitações, e encaminha a solicitação para este executar, balanceando assim a carga entre os servidores do pool.

1.5.3.4. Computação Paralela

Segundo WHEI-JEN (2007), Computação Paralela é caracterizada pelo uso de várias unidades de processamento ou processadores para executar uma computação de forma mais rápida.

Existem várias formas de computação paralela:

- a) em nível de instrução;
- b) em nível de dados;
- c) em paralelismo de tarefas.

O paralelismo tem sido utilizado por muitos anos, principalmente na computação de alto desempenho, segundo RISTA (2004). Os recentes avanços nas tecnologias de microprocessadores e redes locais de alto desempenho têm criado a possibilidade de utilizar um conjunto de computadores como eficientes ambientes paralelos para execução de um grande número de aplicações.

Atualmente, também existe uma corrida para o paralelismo e a clusterização de servidores como consequência de uma procura por uma tecnologia de sustentabilidade e de meio ambiente, no que tange aos recursos de consumo de energia (e, conseqüentemente, a geração de calor). Por isso, a computação paralela se tornou o paradigma dominante na arquitetura de computadores, principalmente na forma de processadores de múltiplos núcleos.

É uma exigência de aplicações com a necessidade de processamento paralelo, principalmente aquelas que envolvem cálculos complexos de matemática e física, geralmente utilizados em pesquisa dentro das universidades e que utilizam o paralelismo para poder executar aplicativos pesados, ou seja, programas que utilizam mais recursos de memória ou processamento, em tempo hábil para a realização das pesquisas associadas.

1.5.3.5. Topologias de Rede

Segundo Dantas (2002), a topologia de rede é o padrão no qual o meio de comunicação entre os computadores está conectado aos outros componentes de rede, tais como hubs e *switchs*. Essencialmente, refere-se à disposição física ou lógica dos recursos da rede.

Na topologia física, queremos enfatizar a ligação dos computadores fisicamente pelos cabos de rede, por exemplo, ou por qualquer outra estrutura física para compor a rede. Na topologia lógica, queremos enfatizar que os computadores podem estar conectados fisicamente sobre uma determinada topologia física, mas os serviços que compõem a rede podem estar definidos em vários equipamentos criando caminhos diferentes da topologia física; quando isso acontece, a visão da topologia em sua totalidade pode ficar confusa, necessitando de uma documentação específica para a topologia física e para a topologia lógica.

2. REFERENCIAL TEÓRICO

2.1. Modelos Pedagógicos

Segundo Behar e Meirelles (2006), modelos pedagógicos são “reinterpretações” de teorias, como a visão interacionista, com foco na interação dos sujeitos, a visão instrucionista, em que o conhecimento não se constrói, ele é transmitido ou transferido para o aprendiz, e a visão humanista, em que o indivíduo já nasce com o saber e, ao longo da sua caminhada, vai se descobrindo e direcionando o seu aprendizado da maneira que melhor lhe convier. Atualmente os modelos pedagógicos estão se adaptando às novas tecnologias com o intuito de aproveitá-las e proporcionar, com isso, uma maior interação com os alunos, aumentando, assim, a motivação e a sociabilidade entre eles, usando, para isso, ferramentas de redes sociais.

O projeto e desenvolvimento do Moodle é guiado por uma "pedagogia social construcionista". Essa seção tenta explicar esse conceito em termos de quatro grandes conceitos relacionados: o construtivismo, o construcionismo, o construtivismo social, conectado e em separado.

Segundo Behar e Meirelles (2006), de um ponto de vista construtivista, as pessoas constroem ativamente novos conhecimentos à medida que interagem com seus ambientes. Tudo o que se lê, vê, ouve, sente e toca é confrontado com seu conhecimento anterior e, se é viável dentro de seu mundo mental, pode-se formar novo conhecimento para o aprendiz.

Eles ainda detalham as teorias de aprendizagem. Abaixo citamos algumas dessas teorias de acordo com esse ponto de vista: O construcionismo afirma que a aprendizagem é particularmente eficaz, quando se constrói alguma coisa para outros experimentarem. Isso pode ser desde uma frase falada ou uma mensagem na Internet até artefatos mais complexos, como uma pintura, uma casa ou um pacote de software. Por exemplo, pode-se ler esta página várias vezes e ainda esquecer-la amanhã, mas, se alguém tentar explicar essas ideias à outra pessoa com suas próprias palavras ou produzir uma apresentação em slides explicando esses conceitos, então é muito provável que esse alguém tenha uma melhor compreensão, que é mais integrada em suas próprias ideias. É por isso que as pessoas fazem anotações durante as aulas (mesmo que nunca as leiam novamente).

O Construtivismo Social amplia o Construtivismo em ambientes sociais, em que os grupos constroem o conhecimento para o outro, criando em colaboração uma pequena cultura de objetos compartilhados, com significados compartilhados. Quando um está imerso numa

cultura como esta, está aprendendo o tempo todo sobre como ser uma parte dessa cultura em muitos níveis.

Um exemplo muito simples é um objeto como um copo. O objeto pode ser usado para muitas coisas, mas sua forma sugere algum "conhecimento" sobre o transporte de líquidos, e a essa funcionalidade agregam-se muitas funções, dependendo do uso do grupo social e da atividade a ela associada. Um exemplo mais complexo é um curso on-line – não só as "formas" de ferramentas de software indicam certas coisas sobre a forma como cursos on-line devem funcionar, mas as atividades e textos produzidos dentro do grupo como um todo. Essas atividades vão ajudar a moldar o modo como cada pessoa se comporta dentro desse grupo; a isso se denomina Construtivismo Social.

Os comportamentos conectado e separado referem-se às motivações dos indivíduos dentro de uma discussão em grupo, tornando-se uma ferramenta interessante em ambientes colaborativos, conforme podemos observar no próprio site "moodle.org". Algumas das principais premissas nesse tópico são:

- Comportamento separado é quando alguém tenta permanecer 'objetivo' e 'factual' e tende a defender suas próprias ideias usando a lógica para encontrar falhas nas ideias de seus oponentes;
- Comportamento conectado é uma abordagem mais empática, que aceita a subjetividade, tentando ouvir e fazer perguntas em um esforço para entender o ponto de vista do outro;

Comportamento construído ocorre quando uma pessoa é sensível a ambas as abordagens e é capaz de escolher uma delas como apropriada para a situação atual. Em geral, uma quantidade saudável de comportamento conectado dentro de uma comunidade de aprendizagem é um estimulante poderoso para a aprendizagem, não só aproximando as pessoas mas promovendo reflexões mais profundas e reexame das crenças existentes.

Segundo Cole e Foster (2008), o acompanhamento de um curso online e o *feedback* obtido por meio dos resultados das avaliações e pesquisas podem ajudar a análise dessas questões para a concepção e o melhoramento do ensino, baseado nas experiências da aprendizagem a partir do ponto de vista do aprendiz, em vez de apenas publicar e avaliar a informação que se acredita serem necessárias. Também pode ajudar a entender como cada participante de um curso pode ser um professor assim como um aprendiz. O papel de "professor" pode deixar de ser "a fonte do conhecimento" para ser um modelo influenciador, conectando-se com os alunos de uma maneira pessoal que atenda a suas necessidades de aprendizagem e moderando discussões e atividades de forma que, coletivamente, leve os alunos às metas de aprendizagem da classe.

Ainda segundo Cole e Foster (2008), o Moodle não força esse estilo de comportamento, mas é isso que os criadores desse software acreditam que seja o melhor a apoiar.

2.2. Ambiente Virtual de Aprendizagem - AVA

Segundo Ozge Yasar e Tufan Adiguzel (2010), Learning Management System (LMS), também conhecido como Ambiente Virtual de Aprendizagem (AVA), é uma infraestrutura de e-learning com as funções de entrega dos cursos que permitem a colaboração e a avaliação do desempenho do aluno, registrando dados do aluno e gerando relatórios para maximizar a eficácia da organização de aprendizagem.

Enquanto os AVAs executam a maioria dessas funções de entrega de conteúdo, ainda existem algumas limitações da ferramenta que podemos citar: a) ser centrado no curso e menos centrado no aluno, b) ter interação e atividades restritas; c) ser apenas uma solução de aprendizado para prover competências, e d) menos controle para desenvolver habilidades de aprendizagem independentes.

Ainda segundo Ozge Yasar e Tufan Diguzel (2010), com essas fraquezas, os AVAs limitam os alunos a lidar com atividades apenas especificadas e não oferecem praticamente nenhum controle sobre as condições em que as atividades ocorrem. Os AVAs com suas ferramentas-padrão contribuem para o conteúdo mas não para o processo de aprender a aprender, exigindo habilidades de pensamento avançado.

Os AVAs são amplamente encontrados nas atividades diárias de educação, por meio de computador. A maioria deles, basicamente, tentam integrar o ensino colaborativo, assiduidade, gestão de projetos dos alunos e dados individuais de aprendizagem de uma escola ou universidade em um único sistema de computador, que pode servir a todas as necessidades particulares dos envolvidos.

Os AVAs são derivados de ferramentas de gerenciamento de conteúdo (CMS, Content Management Systems). Os sistemas modernos chegaram a cobrir quase todos os aspectos dos procedimentos de ensino: lição de casa, gestão de currículos, gestão de atendimento da classe estudantil e projetos, comunicação incorporada que significa mensagens instantâneas, e-mail e videoconferências, web-conferência e vários testes e ferramentas de quizzing entre outros.

Os AVAs, tanto comerciais como os de código aberto, estão disponíveis para implementar ambientes de aprendizagem baseados na web, mas eles exigem grandes servidores, uma equipe de pessoal de apoio e têm um custo de licenciamento caro.

A complexidade e o custo de realizar a implementação de um AVA, muitas vezes, podem ser desgastantes para uma instituição menor, que, por outro lado, devem oferecer um AVA para competir com outras instituições e fazer parte de uma estrutura quase que obrigatória de educação a distância.

Segundo Eduard Mihailescu (2010), nos últimos vinte anos houve um aumento acentuado na pesquisa em torno de e-learning. Desenvolvimentos recentes na Internet de banda larga e o emprego de FOS (Free and Open Source) proporcionaram um impulso ao processo de aprendizagem a distância e, por meio das funcionalidades dos AVAs, surgiu um aumento de ferramentas e uma abordagem abrangente do ensino aprendizagem, avaliação e gestão de estudante.

No entanto, levantamentos técnicos e feedback do pessoal de gerenciamento de servidor enfatizam que existem inconvenientes e restrições no uso de e-learning, principalmente devido a vários surtos e sobrecarga circunstancial das redes de software e hardware, especialmente ao lidar com AVAs.

Essas plataformas de aprendizagem tipicamente lidam com o ensino, a aprendizagem, avaliação e contabilização do estudante de uma instituição de ensino, ou até mesmo, organizações governamentais. Algumas características básicas que um AVA deve cumprir estão brevemente descritas a seguir:

- a) **administração:** o software de aprendizagem deve permitir aos administradores gerenciar os registros de usuários e perfis, definir os papéis, os currículos, caminhos de certificação, atribuir funções aos tutores, funções de autor de cursos, gerenciar conteúdo;
- b) **adesão coerente de padrões de aprendizagem:** o AVA tem de obedecer aos principais padrões de e-learning, como SCORM - Sharable Content Object Reference Model e os emitidos pelo Institute of Electrical and Electronics Engineers - IEEE e Learning Technology Standards Committee - LTSC;
- c) **arquitetura modular:** embora isso não seja obrigatório para uma plataforma AVA, é desejável a possibilidade de integrar nativamente módulos. O uso crescente de código aberto contribui para esse aspecto, ao permitir ao público em geral a modificação do sistema e a inserção de módulos. Em geral, as modificações do usuário são absorvidas pelo projeto principal, e com isso o software evolui no sentido de incorporar os valores da comunidade de usuários.

2.3. Moodle

O idealizador do software de aprendizagem Moodle, Martin Dougiamas, era graduado em informática e também em educação. Trabalhava na *WebCT*, na Universidade de Perth (Austrália), tendo como base de sua pesquisa ambientes educacionais. Iniciou o desenvolvimento de software, que, mais tarde, se tornaria o mais prático e eficaz sistema para a utilização em Ambiente Virtual de Aprendizagem, educativo e colaborativo online.

Moodle - *Modular Object-Oriented Dynamic Learning Environment* é o acrônimo de Ambiente de Aprendizagem Dinâmico Orientado a Objetos e foi lançado pela primeira vez, em 1999, por Peter Taylor, que construiu o primeiro site, disponibilizando um AVA para as Universidade de Tecnologia Curtin, Perth, Austrália (DOUGIAMAS e TAYLOR, 2003).

Em 1999, lançou-se a primeira versão do Moodle, cuja base pedagógica é a abordagem socioconstrucionista da educação, valendo, entretanto, salientar que o Moodle também dá suporte às demais abordagens pedagógicas. Moodle é um pacote de software para produzir cursos baseados na Internet e em web sites. É um projeto de desenvolvimento global destinado a apoiar um quadro construcionista social da educação.

O Moodle é fornecido gratuitamente como software de código aberto (sob a licença GNU *General Public License*). Basicamente, isso significa que o Moodle é protegido por direitos autorais, embora possa ser modificado e utilizado livremente. Pode-se copiar, usar e modificar o Moodle, desde que haja concordância em: fornecer a fonte para os outros; não modificar ou retirar a licença original e os direitos autorais; aplicar essa mesma licença a qualquer trabalho derivado dele.

O Moodle pode ser instalado em qualquer computador que tenha um contexto web, ou seja, um servidor de HTTP, mais conhecido com servidor web, pois o mesmo é escrito na linguagem de programação *PHP*, um módulo que executa dentro do contexto web, e utiliza um tipo de banco de dados *SQL* (por exemplo, *PostgresSQL*). Pode ser executado em sistemas operacionais Windows e Mac e em muitas distribuições de Linux.

2.3.1. Topologia do Moodle

Quanto à topologia interna do Moodle, apresentamos a figura 2.

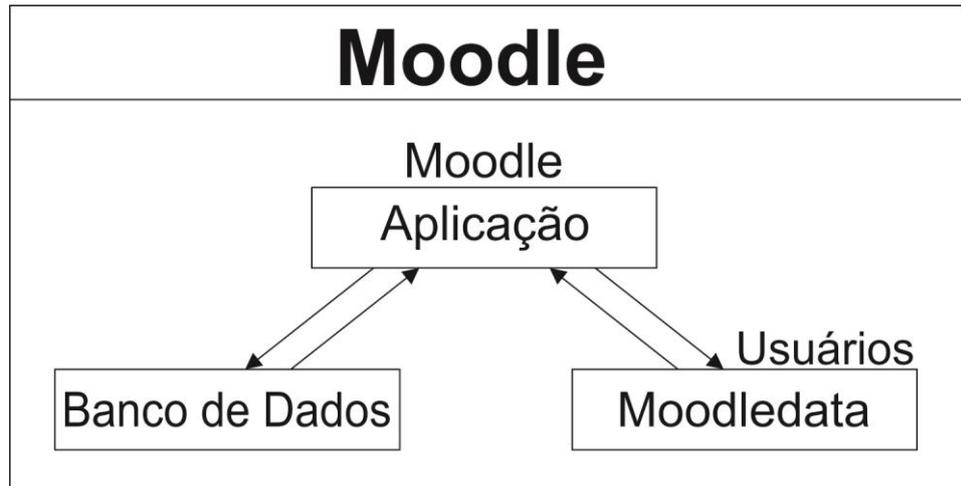


Figura 2. Diagrama interno do Moodle

Topologia lógica diz respeito ao modo como os dados são transmitidos e recebidos dentro da rede. Está ligada intrinsecamente aos protocolos de redes e tem como principal característica ser configurada ou modificada dinamicamente. A topologia lógica de uma rede mostra como estão distribuídos logicamente os servidores por serviços. No caso do Moodle, ele pode ser configurado sob várias, topologias e isso também se constitui em outra versatilidade do Moodle, conforme podemos observar no site oficial do Moodle o moodle.org.

Neste trabalho, conforme anunciado na Introdução, o objetivo principal é a integração do Moodle com um *cluster* de computadores de baixo custo, utilizando o Postgres-XC. Em linhas gerais, nosso experimento consiste de dez computadores funcionando em paralelo, conectados através de placas de rede, controlados pelo software Beowulf. Isso na topologia física; na topologia lógica, o Moodle enxerga um único componente.

A ferramenta Moodle possibilita aos alunos estudarem e aprenderem por meio de um conjunto de mídias projetadas sob uma ótica de modelos pedagógicos, que utilizam a grande rede, maximizando o potencial que essas podem dispor via Internet, como vídeo, *podcasts*, fórum de discussão, *webquests*, atividades, chats, dentre outros. Esses componentes que podem vir acoplados em projetos no formato *Sharable Content Object Reference Model – SCORM*, que é um padrão de empacotamento de conteúdo. Tais pacotes *SCORM* são pedaços independentes de conteúdo, e *JavaScript* como atividades que podem enviar dados para o Moodle, a pontuação dos alunos e locais atuais. O Moodle pode receber cursos, atividades, tarefas, etc. de outros AVAs, e ele utiliza o padrão *SCORM* para essa troca de informações com outros Moodles e/ou outras plataformas de ensino.

O Moodle pode usar pacotes *SCORM* como um tipo de atividade ou como um formato de curso, ou seja, com essa funcionalidade, podemos importar cursos, atividades e tarefas de outros professores, outros cursos e, até mesmo, de outra plataforma, que utilize o mesmo padrão *SCORM*. Quando se tem um grande objeto *SCORM* e deseja utilizá-lo como um curso inteiro, o Moodle importa esse objeto e depois o curso fica disponível no sistema, podendo ser selecionado esse formato de curso para uso na plataforma (COLE e FOSTER, 2008).

Conforme ressaltado por Mihailescu (2009), o Moodle está na ponta da onda do código aberto e lidera entre os softwares não proprietários de sistemas de gestão de aprendizagem no ranking mundial. O Moodle está sendo continuamente aperfeiçoado por vários grupos de pesquisadores e desenvolvedores de todo o mundo. O Moodle foi concebido para ser compatível, flexível e fácil de modificar; desenvolvido na linguagem PHP, que é altamente difundida, usa tecnologias comuns, como bibliotecas compartilhadas, abstração e *Cascading Style Sheets* – CSS, que são estilos de página para definir as interfaces, tornando-se uma ferramenta atrativa para os estudantes (DOUGIAMAS e TAYLOR 2003).

No Brasil, o Moodle é frequentemente usado nos cursos oferecidos pela Unidade Acadêmica de Educação a Distância - EadTec da UFRPE (<http://www.ead.ufrpe.br/>). Também no Núcleo de Educação a Distância da Universidade de Pernambuco NEAD/UPE (<http://www.ead.upe.br/>), além de centenas de universidades espalhadas no Brasil e no mundo.

No que se refere ao seu uso, do ponto de vista do usuário, o *Moodle* consiste numa página web, em que professores disponibilizam recursos e desenvolvem atividades com e para os alunos. Os participantes do ambiente devidamente registrados estão associados a um perfil de usuário, em que perfis e funcionalidades são registrados individualmente ou em grupo. Também cada usuário do ambiente pode usar uma fotografia como identificadora, assim como em redes sociais, facilitando a identificação deste e ajudando na integração com outros alunos. O participante pode comunicar-se com qualquer outro, bastando clicar na imagem da pessoa com a qual se deseja interagir, reforçando a componente social dessa plataforma.

Outra facilidade do Moodle é o fato de que o sucesso da criação da comunidade do "moodle.org" favoreceu a disseminação deste no mundo, e hoje esse software já está traduzido integralmente para mais de 73 línguas diferentes. Este número cresce a cada dia.

O Moodle como sistema de gestão de ensino e aprendizagem apresenta funcionalidades com forte componente de participação, comunicação e colaboração entre formandos, formadores e pares. Segundo BARANAUSKAS (2010), os software sociais têm sido desencadeadores de transformações que estão alterando o modo como as pessoas se relacionam, utilizam e são afetadas pela tecnologia. As oportunidades e os desafios trazidos

por esse tipo de software interativo exigem que os métodos tradicionais de design e a avaliação sejam repensados e que novos conceitos, como valores humanos e técnicos, sejam considerados em sistemas computacionais. O Moodle é um software que tem, na sua fundação, o conceito de software social e coloca esse tipo de sistema no contexto de valores humanos e técnicos.

São oferecidas ferramentas de avaliação específicas de diversas atividades. Vamos citar algumas:

Fórum - é uma ferramenta de discussão por natureza, mas pode ter outro tipo de uso. A dinâmica do fórum é imensa, portanto, o professor capacitado para elaborar um fórum de acordo com a metodologia de ensino e o projeto político-pedagógico do curso ditará o modo como os estudantes farão a reflexão sobre um determinado conteúdo. Os fóruns do Moodle podem ser estruturados de diversas maneiras (discussão geral, uma única discussão, sem respostas, etc.) e podem permitir classificação de cada mensagem (inclusive pelos alunos). As mensagens podem incluir anexos (imagem, pdf, doc, vídeo, áudio, zip).

Trabalho - os trabalhos permitem ao professor classificar e comentar, na página Moodle, materiais submetidos pelos alunos ou atividades *offline*, como apresentações (texto, *powerpoint*, gráficos/desenhos, etc.). As notas são do conhecimento do próprio aluno, e o professor pode exportar os resultados para uma planilha em Excel.

Chat - facilita a comunicação síncrona, através de pequenas mensagens, entre professores e alunos. Pode ser útil como espaço de esclarecimento de dúvidas, mas pode ter outros usos. A sessão de chat pode ser agendada com repetição.

Referendo - pode ser usado de diversas formas, como recolha de opinião ou inscrição numa determinada atividade, sendo oferecida aos alunos uma lista de opções definida pelo professor.

Diálogo - permite a comunicação privada entre dois participantes da disciplina. O professor pode abrir um diálogo com um aluno, o aluno pode abrir um diálogo com o professor, e podem existir diálogos entre dois alunos.

Glossário - possibilita aos participantes da disciplina criar dicionários de termos relacionados com a disciplina, bases de dados documentais ou de arquivos, galerias de imagens ou mesmo *links* que podem ser facilmente pesquisados. Cada entrada permite comentários e avaliação.

Lição - associa a uma lógica de *delivery*, um componente interativo e de avaliação. Consiste num número de páginas ou dispositivos, que podem ter questões intercaladas com classificação e em que o prosseguimento do aluno está dependente das suas respostas. Um conceito baseado na “aprendizagem programada de Skinner”.

Teste - o professor pode construir uma base de dados de perguntas e respostas. Os testes podem ter diferentes formatos de resposta (verdadeiro ou falso, escolha múltipla, resposta curta ou numérica, correspondência, etc.) e é possível escolher perguntas aleatoriamente, corrigir respostas automaticamente e exportar os dados para planilhas eletrônicas.

Questionário - permite construir inquéritos tanto para os participantes de uma página quanto a participantes do Moodle. É possível manter o anonimato dos inquiridos, e os resultados podem ser exportados para uma planilha eletrônica.

Wiki - torna possível a construção de um texto (com elementos multimídia) por vários participantes, em que cada um dá o seu contributo e/ou revê o texto. É possível aceder às várias versões do documento e verificar diferenças entre versões.

2.4. Cluster

Conforme Whei-Jen (2007), o *cluster* é uma técnica que tem sido utilizada por um longo período de tempo para aumentar a capacidade de resistência, desempenho e disponibilidade das soluções implementadas em servidores. Um *cluster* é um conjunto de servidores compostos de duas ou mais máquinas físicas ou máquinas virtuais (também chamado de nós), cada uma com vários subcomponentes registrados para a configuração do *cluster* como recursos.

A funcionalidade dos recursos e os nós são logicamente unidos usando software e hardware adicional. A finalidade de um *cluster* é justamente agrupar os recursos dos diversos computadores, possibilitando que no caso de uma falha do nó, os recursos em outros nós possam assumir a carga de trabalho do nó que falhou, minimizando o tempo de inatividade.

A ideia de agrupamento é apresentar aos usuários uma única máquina, quando, na verdade, o sistema tem vários nós para servir os aplicativos do cliente. Muitos grupos agem em uma configuração ativa / passiva, em que apenas um nó, naquele instante, executa o trabalho.

Algumas soluções de *cluster* são sofisticadas o suficiente, para permitir o balanceamento de carga entre os nós em uma configuração ativo / ativo, maximizando assim o desempenho das aplicações e proporcionando o uso mais econômico dos recursos. Outro recurso importante a fim de contribuir para a alta disponibilidade dos serviços é o balanceamento de carga que distribui os pedidos de informação entre vários servidores.

Sistemas integrados normalmente empregam dois ou mais servidores em uma configuração de balanceamento de carga. Os pedidos são distribuídos para vários servidores

com o intuito de tornar igualitária a carga de trabalho para cada servidor e não sobrecarregar um servidor em particular, fazendo com que ele falhe.

Uma das vantagens de um conjunto equilibrado de carga é que a redundância de múltiplos servidores de aplicações, muitas vezes, reduz ou elimina o tempo de inatividade.

Escalabilidade não é realmente um problema para aplicativos de Intranet, ou seja, a rede interna dentro de uma empresa ou instituição, pois o número de usuários tem poucas chances de aumentar. No entanto, em portais de Internet, a grande rede, a carga aumenta continuamente, principalmente com a disponibilidade de acessos à Internet de banda larga.

O site mantenedor precisa encontrar maneiras de espalhar a carga em vários servidores, seja mediante mecanismos internos, incluídos no servidor de aplicação, ou por meio de componentes externos, ou até mesmo, de redesenho da arquitetura e topologia do sistema. Para essa finalidade, os mantenedores disponibilizam os serviços de balanceamento de carga e, em paralelo, quando detectam sobrecarga no sistema. À medida que o número de servidores cresce, o risco de uma falha aumenta e deve ser tratado, utilizando para isso software ou hardware de alta disponibilidade.

A capacidade para manter o serviço inalterado durante um período e atendendo a um número pré-definido de falhas simultâneas é chamada de alta disponibilidade, serviço configurado, na maioria das vezes, utilizando um computador com as mesmas características do servidor do qual ele será espelho e que ficará em modo de espera até ocorrer alguma falha no sistema principal, assumindo as suas funcionalidades, inclusive se autoconfigurando para a rede. Muitas vezes, é obrigatória com balanceamento de carga, e essa é a razão pela qual as pessoas, muitas vezes, confundem os dois conceitos.

No entanto, de acordo com Stallings (2010), certas técnicas de balanceamento de carga não fornecem alta disponibilidade e, nesse caso, passam a ser perigosas quando do seu uso sem a inserção de um serviço de alta disponibilidade.

Segundo Stallings (2010), um importante e relativamente recente desenvolvimento de design do sistema de computador é *clustering*, uma alternativa para multiprocessamento simétrico, com uma abordagem para o pré-fornecimento de alto desempenho e alta disponibilidade, particularmente atraente para aplicações de servidor. Podemos definir um *cluster* como um grupo de computadores interligados entre si, trabalhando juntos como um recurso de computação unificada, que pode criar a ilusão de ser uma única máquina.

3. METODOLOGIA

Neste capítulo apresentamos uma descrição detalhada da elaboração de nosso protótipo. Usaremos o diagrama de atividades ilustrado na figura 3 para simplificar as várias etapas do processo. Nele observa-se dois tipos de ambientes, do lado esquerdo o ambiente via cluster na rede local e o da direita utilizando o ambiente na nuvem. Pelo diagrama verificamos todas as etapas da instalação do Postgres-XC, servidores de aplicação (Moodle), e servidores de alta disponibilidade (*stand by*), conforme detalharemos a seguir.

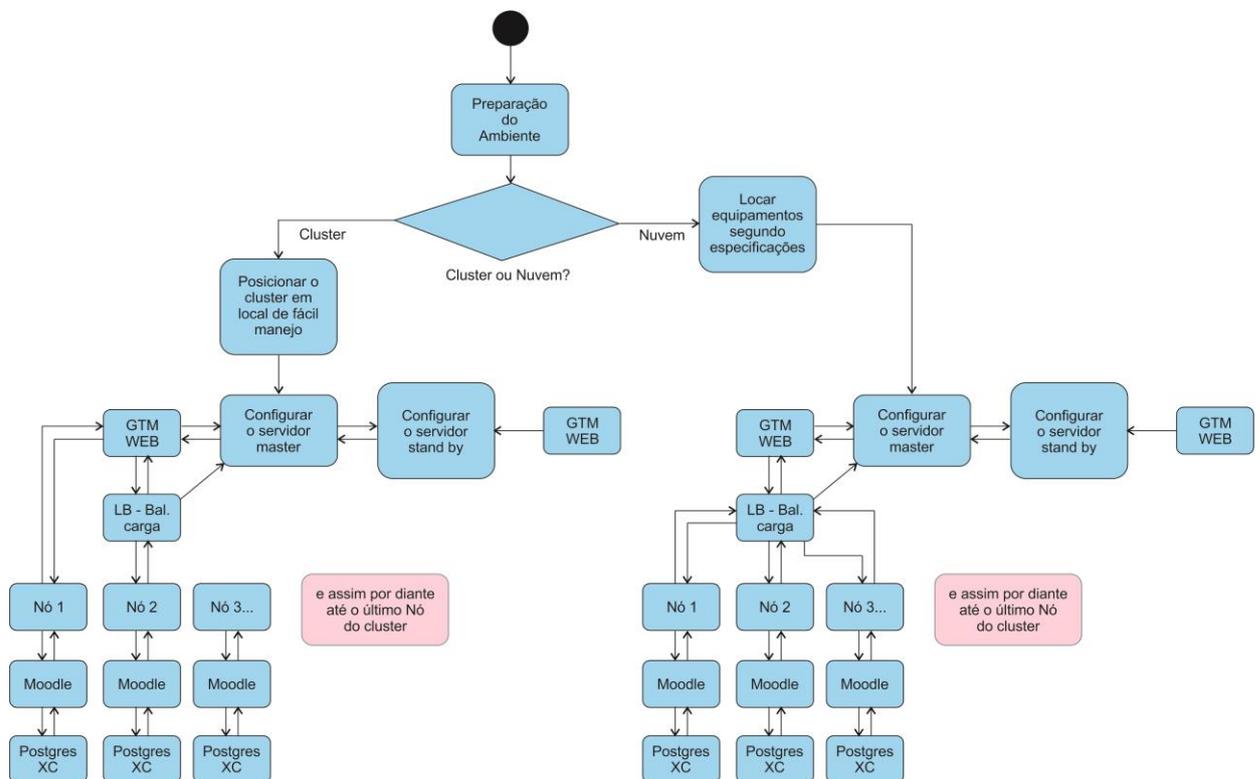


Figura 3. Diagrama de atividades para a instalação do *cluster*

3.1. Preparação do Ambiente de Rede Física e Nuvem

Conforme ilustra a figura 3, antes da instalação do cluster proposto neste trabalho, faz-se necessária uma tomada de decisão: a utilização de uma estrutura existente ou optar por uma instalação na nuvem. No nosso caso, optamos por testar nas duas opções para atestar que essa integração entre o Moodle e o Postgres-XC funcionaria em ambas as plataformas. Para isso, utilizamos, em ambos os ambientes, as seguintes configurações de hardware em cada nó:

- Microcomputador com 2GB de Memória;

- 500 GB de Disco;
- Processador DUAL CORE de 2.6 Ghz;

Utilizaremos esses servidores com diferentes serviços para a composição do nosso *cluster*, e, em nossa nuvem, também essa divisão ocorre por meio de software instalado nos próprios servidores.

3.2. Instalação do *Cluster*

Foi escolhido o sistema operacional Linux Centos 6.3 e instalado em todos os servidores que compõem a solução. Em cada computador, a interface de rede chamada ethernet, que no Linux é mapeada para o dispositivo eth0, deve ser conectada à mesma rede, que compõe o *cluster*, no caso, ao switch ethernet do *cluster* para fazermos uma rede interna entre os computadores. Essa rede é considerada privada, ou seja, todo o tráfego nessa rede está fisicamente separado da rede pública externa (por exemplo, a Internet). Utilizamos endereços internos, o que chamamos de range de IP de intranet; todos os endereços IP's, começando por 192, 172 e 10, são denominados endereços privados.

No computador principal do *cluster*, colocamos duas interfaces de rede, que são obrigatórias, pois uma ficará interligada à Internet e a outra interface conectada com todos os computadores do cluster.

Depois de montar fisicamente o nosso *cluster*, cada nó foi configurado para inicializar sem um teclado, quando ligado. É importante que todos os serviços do *cluster* estejam configurados para funcionarem automaticamente quando o sistema for reiniciado. Podemos ter uma ideia da topologia física na figura 4. Nessa figura, podemos identificar, no nó central, as duas placas de redes, a eth1 e a eth0, sendo a eth1 conectada à rede pública e a eth0 conectada à rede interna do *cluster*. Logo abaixo, vemos os nós do *cluster* configurados na rede interna.

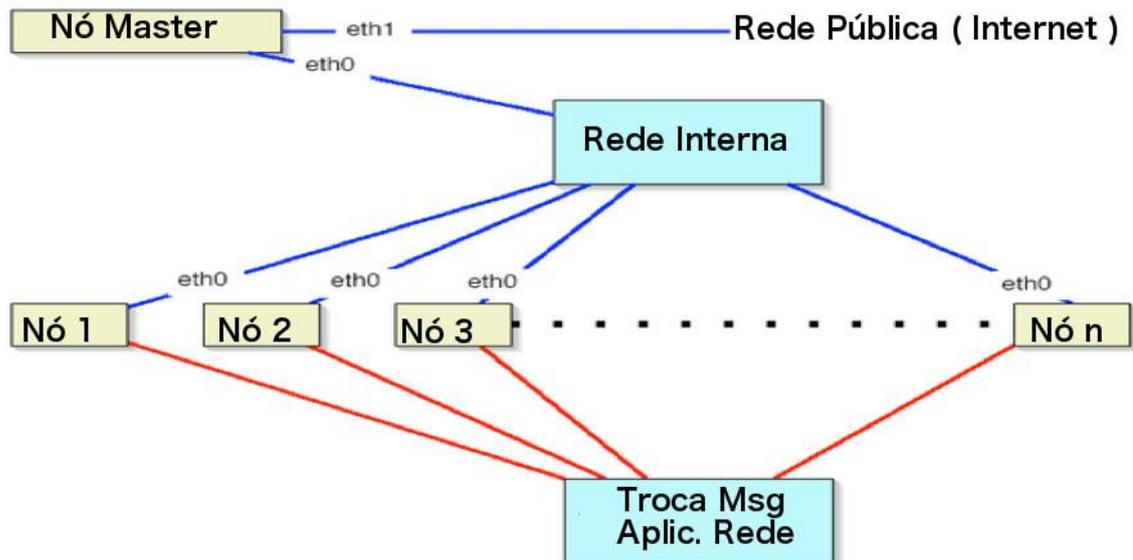


Figura 4. Topologia do Cluster de computadores

Fonte: postgresxc.org

3.3. Instalação na Nuvem

Essa etapa dependeu do aluguel de 4 máquinas na nuvem, sendo eleito o provedor de serviços de hospedagem server4you (<http://server4you.com>) para a instalação do protótipo. Seguimos a mesma linha da instalação do *cluster*, com as mesmas características, mesmo sistema operacional, quantidade de processadores e memória. A diferença principal é que na nuvem, temos IPs válidos da Internet e não necessitamos configurar uma rede interna. Portanto trata-se de uma instalação simples e rápida. Podemos ter uma ideia da topologia física na nuvem na figura 5. Nessa figura, podemos identificar, no nó central, a ligação na eth0, conectada diretamente na Internet. Logo abaixo, vemos os nós do *cluster* configurados todos na Internet.

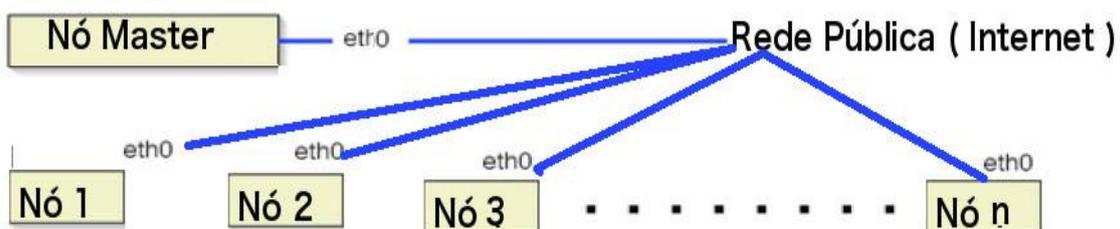


Figura 5. Topologia do *cluster* na nuvem

3.4. Instalação do Banco de Dados Postgres-XC

Conforme já comentamos em nossa descrição do Postgres-XC, esse SGBD é distribuído, ou seja, pode ser instalado em diversas máquinas (físicas ou virtuais), e é escalável, permitindo o atendimento simultâneo de requisições de usuários com eficiência.

Como o Postgres-XC pode ser executado em vários servidores, seus dados são armazenados de forma distribuída. Nos comandos de SQL, o Postgres-XC determina onde os dados são armazenados no cluster. O grande desafio foi instalar o Moodle utilizando essa nova ferramenta, pois o Moodle está homologado para o PostgreSQL e não para o Postgres-XC. Essa implementação, em especial, necessitou de uma pesquisa extensa e detalhada junto ao pessoal do desenvolvimento do software Postgres-XC, através de troca de mensagens eletrônicas.

O Moodle trabalha registrando todos os movimentos do estudante na plataforma. Por exemplo, se o usuário recebeu uma mensagem no fórum e ele apenas leu, o Moodle registra essa movimentação do estudante no banco de dados, assim como se ele postou uma nova mensagem no fórum. Os logs poderão ser analisados pelos professores, tutores e gestores que fazem parte do Ambiente Virtual de Aprendizagem.

O Postgres-XC é composto por vários componentes que têm funções específicas para o *cluster* funcionar em paralelo, e esses componentes podem ser instalados em topologias diversas. Os principais componentes são: o GTM – Global Transaction Manager, que é o gerenciamento global das transações; o GTM_PROXY é um melhorador de *performance* para o GTM; o COORDINATOR é o módulo que faz a comunicação com a aplicação, no nosso caso com o Moodle, e o DATANODE é, de fato, quem armazena os dados.

Nossa topologia do Postgres-XC consistiu em instalar o GTM no computador que chamamos *node0*, ou seja, no computador *master*; em cada nó, instalamos um COORDINATOR (ou seja, *node1*, *node2*, *node3*, *node4*, *node5*, *node6*, *node7*, *node8*, *node9*); o GTM_PROXY foi instalado em cada nó também assim como o DATANODE. Podemos observar o modelo na figura 6. Nela temos, na parte superior, as aplicações do Moodle, trocando mensagens com um coordenador que faz o papel de interface do banco de dados para a aplicação. Já o coordenador troca informações com o GTM – *Global Transaction Manager*, que é o gerenciador de transações, pois todo comando SQL é direcionado para o coordenador que pede uma transação ao GTM após receber o ID da transação; o coordenador executa o comando no datanode que estiver disponível no *cluster* e devolve para a aplicação o resultado da execução do comando.

Ainda na figura 6 podemos observar a existência de apenas um coordenador, embora, em nosso protótipo, tenhamos escolhido utilizar para cada nó do *cluster* de banco de dados um coordenador. Essa topologia do Postgres-XC é bastante flexível. Vale salientar ainda que o catálogo global do banco de dados está diretamente ligado aos coordenadores, pois como o banco de dados Postgres-XC tem transações de *READ/WRITE*, quando uma transação tem efeito no catálogo, essa comunicação é transmitida para os coordenadores.

Componentes Envolvidos na Transação do Postgres-XC

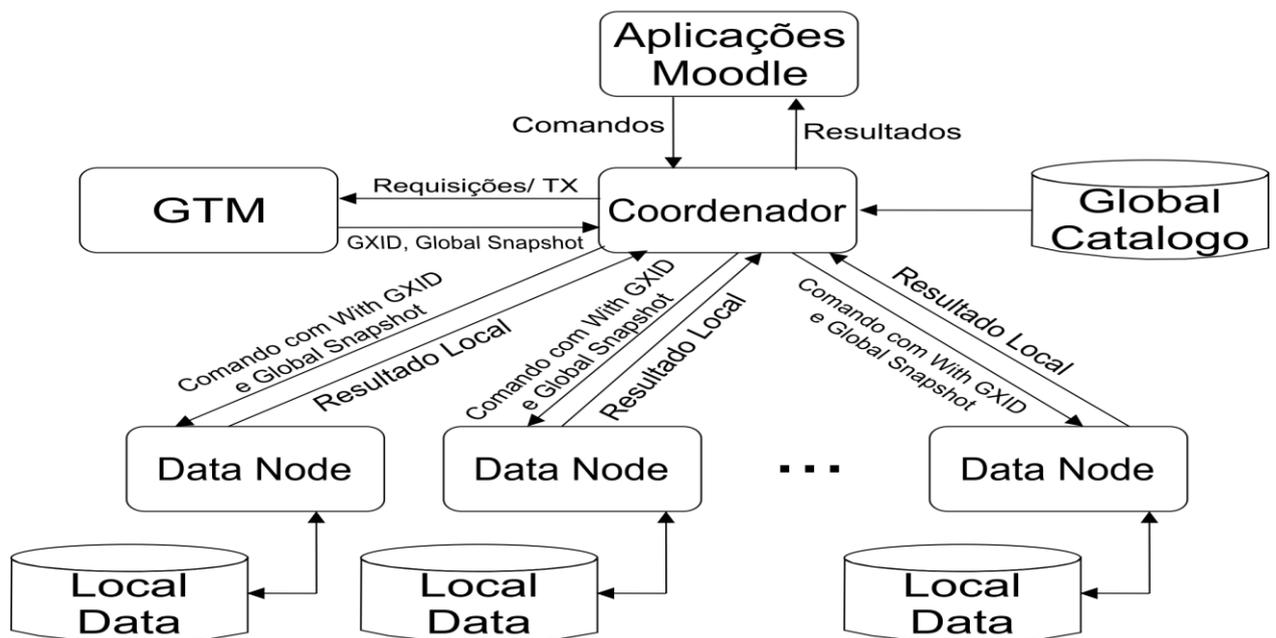


Figura 6. Componentes do Postgres-XC

Fonte: postgresxc.org

Na sequência, apresentamos as etapas de instalação do Postgres-XC conforme feito em nossa pesquisa. Essa descrição é técnica, e nosso objetivo é fornecer um guia permitindo a reprodução da instalação.

A obtenção do software e a preparação para sua instalação são simples:

- Download da versão 1.1 do Postgres-XC em <http://postgresxc.org/>;
- Instalação usual do PostgreSQL, unicamente para criação dos usuários que serão utilizados em nossa instalação Postgres-XC / Moodle:

```
1 # yum install postgresql-server
```

- Descompactação no diretório /usr/local do arquivo pgxc-v1.1.tar.gz:

```
1 # tar -zxvf pgxc-v1.1.tar.gz
```

O primeiro passo da instalação do Postgres-XC consiste em executar o comando abaixo, que criará um arquivo Makefile responsável por controlar a compilação do mesmo:

```
1 # ./configure --prefix=/var/lib/pgsql
2 # make all;make install
```

Em seguida, precisamos configurar nossa instalação do Postgres-XC. Para isso, temos que criar o GTM como segue:

```
1 #initgtm -Z gtm -D /var/lib/pgsql/gtm
2 #cat >> /var/lib/pgsql/gtm/gtm.conf << EOF
3 nodename = 'gtm'
4 listen_addresses = '*'
5 port = 20001
6 EOF
```

A seguir criaremos o GTM_PROXY conforme descrito na sequência de comandos abaixo:

```
1 #initgtm -Z gtm_proxy -D /var/lib/pgsql/gtm_proxy
2 #cat >> /var/lib/pgsql/gtm_proxy/gtm_proxy.conf << EOF
3 nodename = 'gtm_pxy1'
4 listen_addresses = '*'
5 port = 20001
6 gtm_port = 20001
7 gtm_host = 'node03'
8 EOF
```

O próximo passo é a configuração dos COORDENADORES, função importante no que diz respeito ao balanceamento de carga entre os nós.

```
1 #initdb --nodename=coord1 -D /var/lib/pgsql/coordinator1
2 #rm /var/lib/pgsql/coordinator1/postgres.conf
3 #cat >> /var/lib/pgsql/coordinator1/postgresql.conf << EOF
```

```

4 listen_addresses = '*'
5 gtm_port = 20001
6 gtm_host = 'node01'
7 port = 5432
8 pooler_port = 20006
9 EOF
10 #cat >> /var/lib/pgsql/coordinator1/pg_hba.conf << EOF
11 host all all 192.168.0.100/32 trust
12 host all all 192.168.0.101/32 trust
13 host all all 192.168.0.102/32 trust
14 host all all 192.168.0.103/32 trust
15 host all all 192.168.0.109/32 trust
16 EOF

```

Note que, na configuração do arquivo `pg_hba.conf` na linha 10 acima, precisamos definir todos os hosts que fazem parte do nosso cluster. A seguir, necessitamos criar as configurações dos DATANODES. Em cada nó repete-se o processo para todos os nós do *cluster*.

```

1 #Initdb --nodename=datanode1 -D /var/lib/pgsql/dn1
2 cat >> /var/lib/pgsql/dn1/postgresql.conf << EOF
3 listen_addresses = '*'
4 gtm_port = 20001
5 gtm_host = 'node01'
6 port = 20006
7 EOF

```

Aqui começa uma parte importante na integração com o Moodle. Nesse ponto, o SGBD está operacional e necessitamos fazer a instalação do Moodle antes de gerar a dependência no Postgres-*XC* com os outros nós.

No nosso experimento instalamos, em uma rede interna, o cluster em `http://192.168.0.100/eadufrpe` e prosseguimos com o passo a passo da instalação normal do Moodle, utilizando o driver do *PostgreSQL* que acompanha o próprio Moodle.

Após a instalação, faremos um backup do banco de dados recém-criados:

```

1 # su - postgres
2 $ psql -h node01 postgres
3 $ pg_dump bancorecemcriado > backup.sql
4 $ dropdb bancorecemcriado;

```

Nesse ponto, é de fundamental importância editarmos o arquivo gerado de backup.sql e alterarmos todas as ocorrências de UNIQUE INDEX para índices distribuídos nos nós, seguindo o exemplo abaixo:

```
1 CREATE UNIQUE INDEX ON mdl_backidstemp_baciteite_uix
mdl_backup_ids_template
2 USING btree (backupid, itemname, itemid)
```

Ao identificarmos os comandos acima de criação de índices precisamos acrescentar a linha abaixo, pois o Postgres-XC precisa modificar o tipo do índice para ser distribuído na rede:

```
1 ALTER TABLE mdl_backidstemp_baciteite DISTRIBUTE BY HASH
(backupid,)
```

Temos que fazer isso para todos os *CREATE UNIQUE INDEX*. Após concluída essa etapa, está na hora de conectarmos os nós do *cluster* do banco de dados. Em cada nó, temos que executar dentro do banco de dados os seguintes comandos de ligação:

```
1 # su - postgres
2 $ psql -h node01 postgres
3 $ CREATE NODE coord1 WITH (TYPE = 'coordinator', HOST =
'node02', PORT = 5432);
4 $ CREATE NODE datanode1 WITH (TYPE = 'datanode', HOST =
'node01', PORT = 20006);
5 $ CREATE NODE datanode2 WITH (TYPE = 'datanode', HOST =
'node02', PORT = 20006);
6 $ CREATE NODE datanode2 WITH (TYPE = 'datanode', HOST =
'node02', PORT = 20006);
```

Repetimos esses passos para todos os coordenadores e todos os datanodes de nossa instalação. Após essa etapa, criaremos o banco de dados novo, já no *cluster* e importaremos os dados da instalação do Moodle para esse banco. Como segue:

```
1 $ createdb bancorecemcriado;
2 $ psql bancorecemcriado < backup.sql
```

3.5. Instalação do Servidor Web - Apache no *Cluster*

O Projeto Apache HTTP Server é um esforço de desenvolvimento de software colaborativo, que visa à criação de um robusto sistema interpretador do protocolo HTTP, ou seja, interpretador das páginas web. O projeto é gerido em conjunto por um grupo de voluntários localizados em todo o mundo, através da Internet e da Web para se comunicar, planejar e desenvolver o servidor e sua respectiva documentação. Esse projeto faz parte da *Apache Software Foundation*⁶. Além disso, centenas de usuários têm contribuído com ideias, códigos e documentação para o projeto.

A instalação básica do servidor web é muito simples. Primeiro, foi executado, em cada nó do *cluster*, o seguinte comando:

```
1 # yum install httpd mod_ssl
2 # /etc/init.d/httpd start
```

Quando colocamos o servidor no ar, notamos a mensagem de erro do servidor:

```
Starting httpd: httpd: Could not reliably determine the server's
fully qualified domain name, using 127.0.0.1 for ServerName
```

que nos indica que precisamos fazer uma alteração na qualificação do domínio do nosso servidor, no arquivo localizado no `/etc/httpd/conf/httpd.conf`, que, no caso do nosso projeto, será <http://www.progepe.org/69.64.59.112>. Para isso, editamos o arquivo `httpd.conf`:

```
1 # vi /etc/httpd/conf/httpd.conf
```

Notamos que o `ServerName` estava definido por padrão como `127.0.0.1` e colocamos o nosso `ServerName` como sendo `http://www.progepe.org/`. Alteramos para:

```
1 ServerName http://www.progepe.org:80
2 #/etc/init.d/httpd reload
```

Esse procedimento da alteração do `ServerName` é necessário porque, por padrão, o servidor web vem configurado para o endereço `127.0.0.1`, e para ele possa atender aos pedidos via Internet, precisamos definir um endereço válido na Internet. Testamos os servidores web

⁶ Ver www.apache.org

do *cluster* individualmente, digitando o endereço <http://69.64.59.112>, 69.64.59.113 e assim por diante, atestando o aparecimento da tela representada na figura 7 que é a tela padrão do servidor web. Essa tela indica que o servidor web está configurado corretamente.

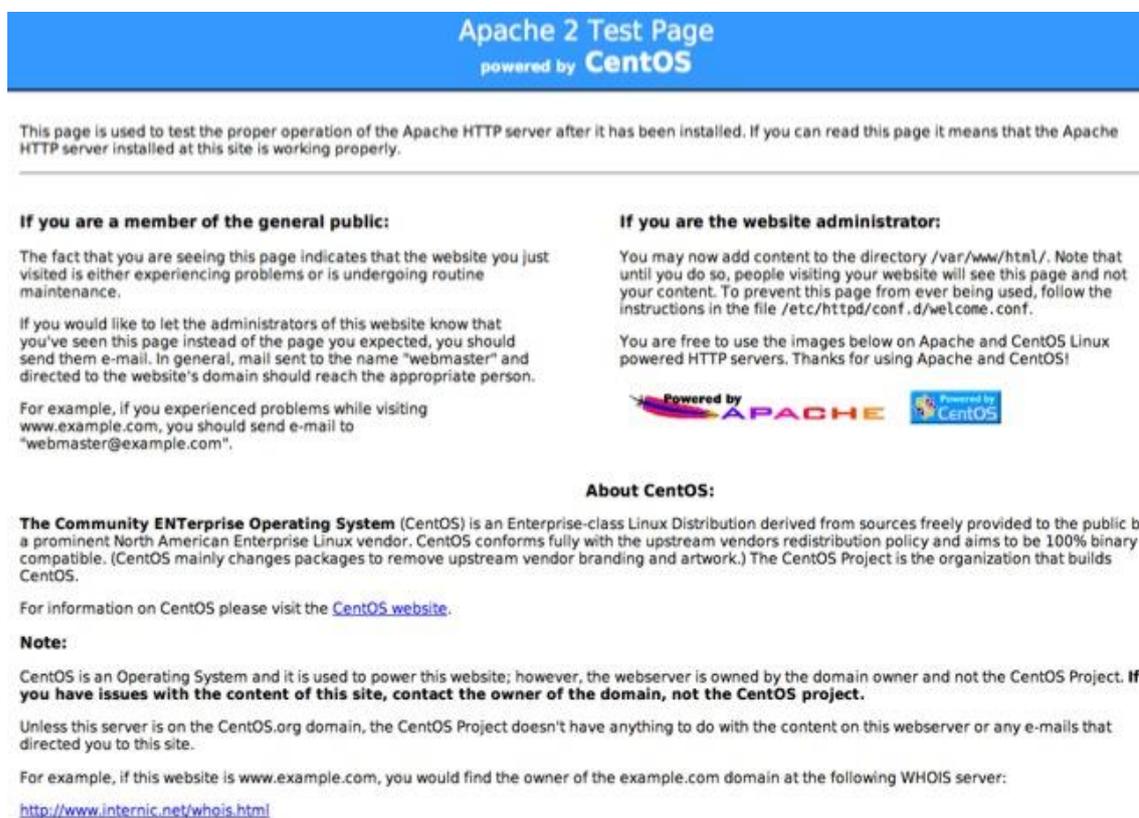


Figura 7. Tela padrão do servidor Web - Apache

3.6. Instalação do Balanceamento de Carga

Foi divulgado o seguinte endereço para os estudantes: <http://www.progepe.org/upevirtual>. O sistema automaticamente redirecionaria a solicitação para o servidor disponível naquele instante, o que é feito de forma transparente para o estudante. Isso torna a instalação do Moodle escalável, ou seja, quando houver a necessidade de aumentar o número de estudantes, basta colocar mais um servidor no pool de servidores.

Isso posto, nossos servidores foram divididos em serviços. Colocamos o domínio em <http://www.progepe.org/>, conforme descrito no item 3.5, como sendo o gerenciador DNS do *cluster*.

Vale salientar, que o serviço web usando o software Apache tem embutido nele uma função de balanceamento de carga que vamos configurar em seguida: primeiro cadastramos todos os 3 (três) servidores no DNS da máquina. Isso foi feito para que o software Moodle

pudesse atender às demandas por um único endereço, <http://www.progepe.org/> e, para tal, configuramos o sistema, seguindo os passos descritos abaixo.

Primeiro, fizemos um upgrade no sistema:

```
1 # yum update
2 # yum groupinstall "Web Server"
```

Colocaremos em seguida a configuração do balanceador de carga no arquivo `/etc/httpd/conf/httpd.conf`:

```
1 #</Proxy>
2 <Proxy balancer://myCluster>
3 BalancerMember http://69.64.59.112/eadufrpe
4 BalancerMember http://69.64.59.113/eadufrpe
5 </Proxy>
6 ProxyPass /upevirtual balancer://myCluster
7 <Location /balancer-manager>
8 SetHandler balancer-manager
9 </Location>
```

Deve-se proceder também assim caso um servidor venha a apresentar problemas físicos; basta identificar o servidor danificado e trocá-lo por outro, e o sistema não para. Isso não vale para o nó principal. Por esse motivo, é que, para o servidor central, aconselhamos o uso de redundância, formando, assim, um sistema de alta disponibilidade.

3.7. Instalação do Servidor de Alta Disponibilidade

Além dessa composição dos 3 servidores com o Moodle, foi feita uma instalação de um outro servidor que permanece em *stand by*, ou seja, aguardando qualquer falha do servidor central para assumir essa posição; a isso denominamos de alta disponibilidade.

Esse servidor se conecta ao servidor central por uma rede interna exclusiva, e o computador reserva escuta constantemente essa rede, a fim de verificar se o servidor central responde. Caso contrário, o servidor reserva se autoconfigura com o endereço IP do servidor central, e o sistema responde pelo servidor reserva, não parando o serviço, conforme está indicado na figura 8. Nessa figura, mostramos uma visão global do sistema. Na parte superior da figura, encontram-se os usuários e, logo em seguida, a estrutura montada para alta disponibilidade e balanceamento de

carga, em que podemos observar o servidor de espera, marcado na figura com o nome de backup. Logo abaixo vem os servidores de aplicações com o Moodle, denominados real server1, real server 2, etc. Em seguida, vê-se nosso cluster de banco de dados Postgres-XC.

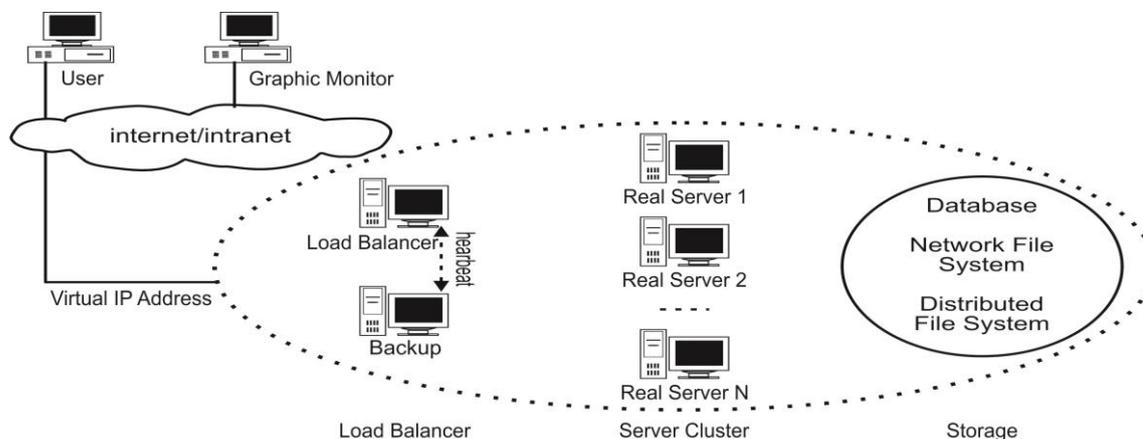


Figura 8. Topologia da Solução

Fonte: postgresxc.org

Esse sistema faz com que seja fácil a conexão de vários outros computadores para aumentar o número de usuários em uma futura expansão.

3.8. Preparação do *Cluster* para Área de Dados dos Usuários - NFS

Na sequência, incluímos um servidor dedicado ao serviço de NFS, ou seja, o servidor de arquivos, pois, conforme a topologia do Moodle, faz-se necessária a criação de um servidor de dados para o armazenamento da pasta moodledata com os dados dos usuários.

Primeiro foi necessário instalarmos a ferramenta de compartilhamento de discos do Linux, o NFS, cuja função será exportar o diretório para o *cluster*. Para isso, instalamos os seguintes serviços: `rpc.portmap` `rpc.mountd`, `rpc.nfsd` `rpc.statd`, `rpc.lockd` and `rpc.rquotad`

```
1 # rpm -q nfs-utils
2 # yum install nfs-utils
3 # service portmap start
4 # service portmap start
```

Após essa configuração, definimos os diretórios que seriam compartilhados pelos outros nós que fazem parte do *cluster*, conforme descrito no arquivo de configuração do NFS, chamado

exports e localizado em /etc/exports:

```
1 /var/www/moodledataeadufrpe eagle863(rw, sync, fsid=0)
```

Abaixo faremos o criação do diretório moodledata, comum a todos os nós, diretório este reservado aos dados de alunos do curso, ou seja: textos, tarefas, documentos e apresentações.

```
1 # mkdir /path/to/moodledata
2 # chmod 0777 /path/to/moodledata
```

3.9. Instalação do Moodle

O Moodle pode ser configurado sob várias topologias, e isso também é outra versatilidade do Moodle. No caso desse *cluster*, ligaremos os 3 computadores em paralelo, por meio das placas de rede e software do Sistema Operacional na topologia física; o Moodle executará como se estivesse em um único componente, na topologia lógica. O Moodle utilizado no nosso protótipo foi a versão do Moodle 2.3.1+ (Build :20120719).

Primeiramente, obtemos o Moodle:

```
1 # git clone -b MOODLE_23_STABLE git://git.moodle.org/moodle.git
2 # chown -R root /path/to/eadufrpe
3 # chmod -R 0755 /path/to/eadufrpe
```

Em seguida precisamos instalar o Moodle em todos os nós, e fazemos isto através de um navegador, em cada estação. O detalhe importante nesse ponto é que precisamos instalar o Moodle, inicialmente com o Postgres-XC sem a conexão com os outros *coordinators*, ou seja, de forma “*stand alone*”, ou local. Primeiramente, entramos no banco como segue:

```
1 #su - postgres
2 #createdb eadufrpe
3 #exit
```

Após este procedimento, acessamos o seguinte endereço no navegador:

```
1 http://69.64.59.112/eadufrpe
```

O procedimento acima irá instalar o Moodle no ambiente, via web na sua versão

padrão em todas as estações no banco de dados local. Nos passos seguintes, faremos as adaptações para o Postgres-XC, para podermos usufruir das funcionalidades de *cluster*.

Após a conclusão da instalação temos o banco PostgreSQL com todas as instalações do Moodle e a base de dados funcionando. Mas precisamos fazer a adaptação para o Postgres-XC. Para isto, faremos o backup das informações do banco PostgreSQL para que possamos fazer as alterações necessárias para o Postgres-XC. Para isso, faremos o backup com o comando *pg_dump* como segue, salvando em um arquivo, em formato texto, o banco de dados no padrão do PostgreSQL:

```
2 $ pg_dump eadufsrpe > eadufsrpe.sql
```

Em seguida, editaremos o arquivo gerado pelo comando *pg_dump* acima, “eadufsrpe.sql”, para fazer as adaptações necessárias para o Postgres-XC. Precisamos neste momento trocar as ocorrências da palavra *DISTRIBUTE BY* pela palavra *REPLICATION* e também para cada termo *CREATE UNIQUE* colocamos a chave do *INDICE* no *DISTRIBUTE BY HASH* (chave), conforme descrito no item 3.4 desta dissertação.

Após essas alterações apagamos o banco antigo e criamos um novo com a nova estrutura utilizando o arquivo alterado anteriormente, no caso o eadufsrpe.sql. Após este procedimento copiamos a instalação para os outros servidores do cluster.

3.10. Preparação do Ambiente Virtual de Aprendizagem

Nesta etapa, já com o cluster em funcionamento, pudemos acessar o ambiente virtual de aprendizagem para construir o protótipo de curso, ou seja, inserir atividades, trabalhos e fóruns.

Vale ressaltar que o protótipo está ligado a uma rede interna entre os componentes do cluster e apenas o servidor Master está interligado com a Internet, fazendo assim a ponte com o mundo externo. Para os alunos é divulgado apenas o endereço eletrônico correspondente ao servidor Master. O ambiente virtual de aprendizagem permanece o mesmo para os alunos, tutores e professores: ninguém é capaz de visualizar a configuração física dos servidores internamente.

Na figura 9 observamos a foto do protótipo do nosso *cluster*. Colocamos em uma sala 9 equipamentos unidos pelas suas CPU's, e o master um pouco mais afastado e isolado para preparação do ambiente de alta disponibilidade com o servidor “stand by”.



Figura 9. Foto do protótipo do *cluster*

A página principal do curso está ilustrada na figura 10, onde para efeito didático apresentamos a marcação “**NODE 0 NODE 0 NODE 0**” para verificar na prática qual servidor está atuando em um dado momento (a fim de testar o balanceamento de carga).

192.168.0.101/moodle/

UFRPE EAD

You are not logged in. (Login)

English (en)

Home

Navigation

Home

Courses

NODE 0 NODE 0 NODE 0

Available courses

No courses in this category

Calendar

August 2013

Sun	Mon	Tue	Wed	Thu	Fri	Sat
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

You are not logged in. (Login)

moodle

Figura 10. Tela inicial do protótipo

4. AVALIAÇÃO

4.1. Testes

Para avaliar o protótipo, fizemos dois testes distintos. O primeiro foi elaborar um curso com um fórum e um chat e testar o funcionamento do protótipo quanto à sincronia entre os nós do sistema de *cluster*; constatamos nessa etapa que o Ambiente Virtual de Aprendizagem se comportou bem. Utilizamos também um recurso de mensagem, e verificamos que a mensagem aparece no terminal da outra máquina imediatamente, sem a necessidade de “refresh” na tela (atualização imediata dos dados vindos do servidor). O sistema se comportou também bem nesse aspecto.

Passamos, então, a um novo teste, a utilização de software para testar automaticamente o sistema. Escolhemos a aplicação Apache JmeterTM ⁷, bastante usada para geração de testes e medições de desempenho em sistemas, que além de possuir uma interface gráfica versátil, é uma aplicação Java de código aberto. Ele foi originalmente projetado para testar aplicações web, mas desde então se expandiu para outras funções de teste. O Jmeter se mostra útil na investigação de problemas de desempenho e na avaliação do impacto de mudanças do código.

O Jmeter trabalha com um plano de testes. A fim de agilizar nosso trabalho, aproveitamos um plano existente na comunidade especializada, localizado em <https://github.com/samhemelryk/moodle-jmeter-perfcomp>, chamado de Moodle *performance comparison tool*, que é uma ferramenta de comparação de desempenho do Moodle.

Em um primeiro momento, executamos o script `perfcomp.init.php`, que criou automaticamente 114 usuários, sendo 5 Managers, 99 Students e 10 Teachers, resultando em uma média de 0.12 segundos, dentro da margem estipulada por COELHO (2010).

Para fins de comprovação nos diversos ambientes testados a seguir, o parâmetro de comparação escolhido foi a vazão (*throughput*), que consiste no número de operações que o sistema é capaz de completar em um dado período de tempo em segundos, conforme a documentação da Apache JmeterTM.

Nos testes dos ambientes de nuvem e de *cluster*, foram utilizados como parâmetro pelo Jmeter:

- número de usuários virtuais (threads): 20
- tempo de inicialização: 2 (segundos)
- contador de iteração: 1000

⁷ Consulte <http://jmeter.apache.org/> para mais detalhes e documentação sobre a ferramenta.

Apresentamos os resultados nas seções seguintes.

4.1.1. Teste em Ambiente com Moodle e PostgreSQL

Após o teste de aproximadamente 20.000 amostras, obtivemos uma média de 4.500 milissegundos de resposta com perda de 1,47% e com uma vazão de 3.1/s. Esse teste consiste da simulação de uma sessão pelo usuário, passando o login e a senha. Nesse ponto, o Moodle grava a informação de acesso do usuário, e depois o simulador executa a saída do sistema, segundo gráficos abaixo compostos pelo gráfico de tempo de resposta e pelo sumário das conexões. No gráfico da figura 11 de tempo de resposta, chamamos a atenção para a média em segundos; no sumário, na figura 12, chamamos a atenção para o campo de erros e para o campo de vazão.

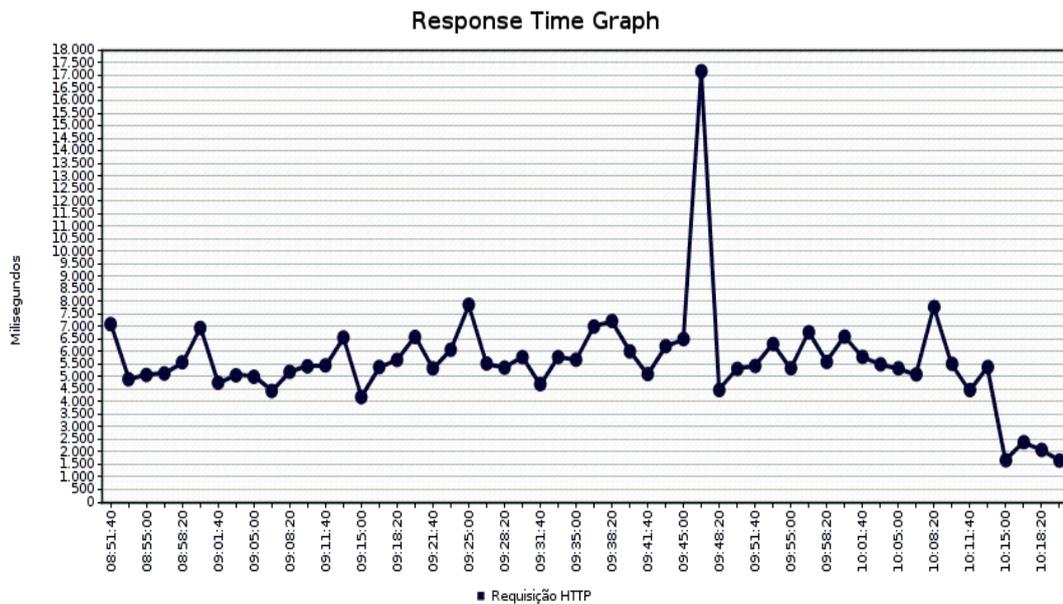


Figura 11. Gráfico de resposta do *Jmeter* do protótipo Moodle e PostgreSQL

Relatório de Sumário

Nome: Relatório de Sumário

Comentários:

Escrever resultados para arquivo / Ler a partir do arquivo

Nome do arquivo: Apenas Logar/Exibir Erros Sucessos

Rótulo	# Amostras	Média	Mín.	Máx.	...	% de Erro	Vazão	KB/s	Média de Bytes
Requisiçã...	19648	5342	883	3076468	...	1,47%	3,1/sec	6,78	2246,1
TOTAL	19648	5342	883	3076468	...	1,47%	3,1/sec	6,78	2246,1

Incluir nome do grupo no rótulo? Salvar Cabeçalho da Tabela

Figura 12. Sumário *do Jmeter* do protótipo com Moodle e PostgreSQL

4.1.2. Teste no *Cluster Local* com Moodle e Postgres-XC

Para fins de utilização de um Ambiente Virtual de Aprendizagem, em cursos a distância, utilizando a plataforma remota da web, concluímos que os testes apresentados utilizando um barramento isolado poderia nos levar a conclusões ótimas de ambiente que dificilmente seriam replicadas em um ambiente remoto. Portanto, sob essas considerações, não utilizamos testes em ambiente de uma rede interna.

4.1.3. Testes na Nuvem com Moodle e Postgres-XC

Após o teste de aproximadamente 20.000 amostras, obtivemos uma média de 1.500ms de resposta com perda de 0,00% e com uma vazão de 10.3/s. Esse teste consiste da simulação de uma sessão pelo usuário passando o login e a senha. Nesse ponto, o Moodle grava a informação de acesso do usuário e depois o simulador executa a saída do sistema, segundo gráficos abaixo compostos pelo gráfico de tempo de resposta e pelo sumário das conexões. No gráfico da figura 13 de tempo de resposta, chamamos a atenção da média em segundos, enquanto no sumário, figura 14, chamamos a atenção para o campo de erros e para o campo de vazão.

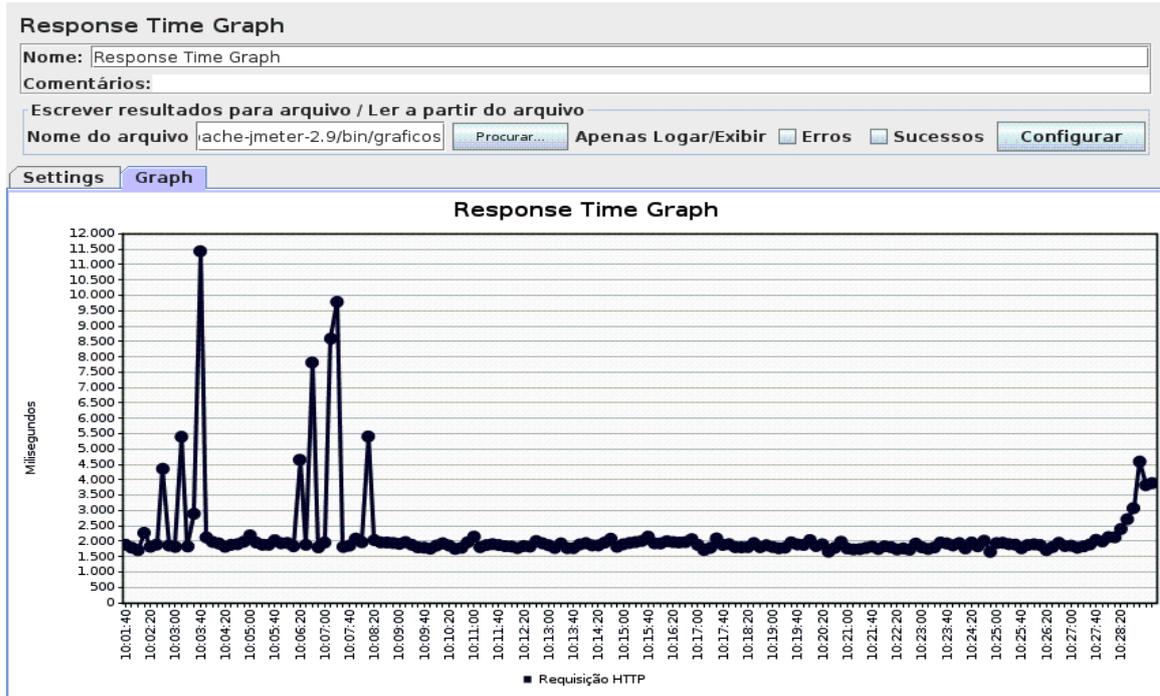


Figura 13. Gráfico de resposta do Jmeter do protótipo Nuvem

The screenshot shows the 'Relatório de Sumário' window in JMeter. It includes a title bar, a name field ('Relatório de Sumário'), a comments field, and a section for saving results to a file ('Nome do arquivo' set to '2.9/bin/afonso.txt'). Below this are 'Settings' and 'Table' tabs. The table displays the following data:

Rótulo	# Amostras	Média	Mín.	Máx.	...	% de Erro	Vazão	KB/s	Média de Bytes
Requisição HTTP	15153	5656	1148	912245	...	0,00%	10,3/sec	261,...	26119,9
TOTAL	15153	5656	1148	912245	...	0,00%	10,3/sec	261,...	26119,9

At the bottom of the window, there are checkboxes for 'Incluir nome do grupo no rótulo?' (unchecked), 'Salvar Dados da Tabela' (button), and 'Salvar Cabeçalho da Tabela' (checked).

Figura 14. Sumário do Jmeter do protótipo Nuvem

4.2. Sumário dos resultados

Foi implementada a solução inédita em laboratório, utilizando a plataforma descrita neste trabalho com um banco de dados paralelo, chamado *Postgres-XC*. Colocou-se em operação um sistema integrado, conforme descrito, obtendo-se sucesso e com um tempo de resposta dentro do estimado. Para a comparação da performance dos servidores, seguimos a orientação de Coelho (2010), que diz que o indicador de performance pelo qual devemos estar interessados é o que o utilizador final sente: o tempo de carregamento de uma página. Se esse

tempo for rápido, cerca de 1 segundo, o utilizador sente-se confiante e utiliza, sem problemas, a plataforma; se for lento, mais de 10 segundos, pensará antes de cada clique, provavelmente irá abrir várias janelas e sentir que perde tempo para poder utilizar a plataforma.

É de importância capital manter esse tempo sempre baixo, evitando a degradação da experiência de utilização da plataforma. Relativamente a esse parâmetro, conseguimos obter o resultado ilustrado na figura 15, em que podemos ver o número de conexões em azul crescente e o tempo de resposta quase constante, em laranja.

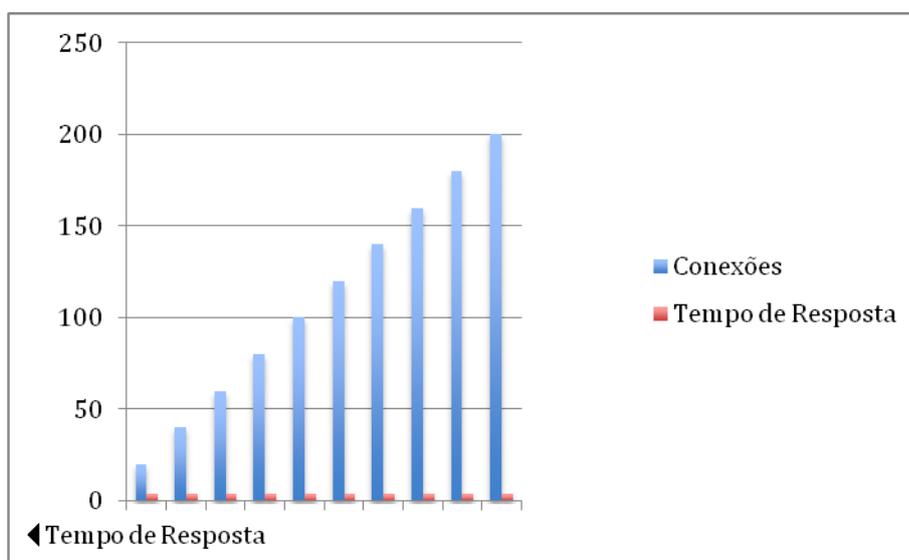


Figura 15. Gráfico das quantidades de conexões x tempo de resposta

Pelas estatísticas de uso do Moodle, verificou-se a viabilidade da solução de um sistema de alta disponibilidade e de baixo custo. O software Moodle foi instalado e operado como se executasse em um único servidor, de forma transparente para os estudantes que, apenas, conheciam um único endereço eletrônico: <http://69.64.59.112/ufupe13>.

Demonstrou-se, também, que, ao se remover um dos computadores, não houve interrupção ou quebra de serviço do sistema, provando-se ser esta uma solução estável e escalável. Porém como a solução de executar o Moodle em paralelo, utilizando Postgres-XC, um banco de dados paralelo, é inédita, novos testes são necessários a fim de validá-la, em particular com turmas de um curso real. Vale salientar ainda que não foi testado um curso completo em operação; testamos a instalação e as funcionalidades do Moodle, utilizando um software de testes chamado Jmeter.

5. CONCLUSÃO

Com este trabalho, concluímos ser, de fato, possível construir uma infraestrutura de baixo custo para atender a uma série de serviços que exigem uma rede complexa de informática. Conforme a exposição do capítulo 2 sobre a motivação para o desenvolvimento dessa infraestrutura por meio de computadores pré-existentes no âmbito da instituição, o objetivo foi fornecer suporte a uma economia de escala, que fosse confiável, estivesse sempre em funcionamento, e, principalmente, fosse de baixo custo. O advento do software livre, ou seja, sem custo inicial de compra de licença, foi crucial para a concretização desse objetivo, como foi demonstrado no capítulo 3, em que detalhamos o funcionamento e a implementação do *cluster* utilizado, com sucesso, nos testes desse protótipo.

Nos testes de estresse do protótipo, foram efetuadas simulações de digitação com a ferramenta Jmeter, para que essas informações fossem acessadas quase que simultaneamente pelos programas, como se fossem alunos reais, exigindo uma infraestrutura de alta performance, confiável, escalável e capaz de manter uma taxa elevada de satisfação dos usuários quanto ao tempo de resposta do Ambiente Virtual de Aprendizagem. Todos os parâmetros acima citados foram avaliados a partir da implementação dos testes feitos por Benner (2010) e replicados em nosso protótipo.

Conclui-se que a abordagem descrita neste documento fornece uma solução viável para fornecer um sistema com melhora da alta disponibilidade; um sistema confiável, com o potencial de fornecer uma solução de alta performance, quando comparado com arquiteturas similares, utilizando o mesmo meio de comunicação, conforme demonstrado no capítulo 4.

Conclui-se, também, que essa infraestrutura pode ser implementada em equipamentos de baixo custo, ou seja, equipamentos existentes com um ou dois anos de uso, ou novos com processadores de modelos antigos de tiragem anteriores, de custo mais baixo para aquisição, reutilizando-se esses equipamentos para a construção dessa infraestrutura. Portanto, nossa hipótese provou o potencial do *Cluster Linux* para prover uma plataforma de AVA, utilizando o Moodle, capaz de ser escalável, disponível e de baixo custo. Ou, ainda, pode-se alugar um conjunto de servidores virtuais na nuvem e prover o mesmo serviço.

Um próximo passo é conseguirmos, junto à comunidade livre que produz o Moodle, inserir essa solução dentro da distribuição oficial do Moodle, tornando, assim, essa plataforma disponível para a comunidade acadêmica. Outra intenção é fazer a mesma integração com o Software de AVA Amadeus junto com Postgres-XC em computação na nuvem.

REFERÊNCIAS

ABDI, A. B. D. D. I. **Tecnologias de informação e comunicação**. Cadernos Temáticos TIC. Brasília, jun. 2010.

ALMEIDA, M. E. B. D.; MORAN, J. M. **Integração das tecnologias na educação**. Brasília: Posigraf, 2005.

BARBALACEL, A., LUCHETTA, A., MANDUCHI, G., MORO M., SOPPELSA A. e TALIERCIO, C. **Performance comparison Of VxWorks, Linux, RTAI and Xenomai in a Hard Real-time Application**. University of Padova. Italy. 2008.

BARANAUSKAS, M.; PEREIRA, R.; SILVA, S. **Softwares: uma visão orientada a valores**. In: SIMPÓSIO BRASILEIRO DE FATORES HUMANOS SISTEMAS COMPUTACIONAIS, 26, 2010, Belo Horizonte.

BEHAR, P. A.; MEIRELLES, S. **The virtual learning environment ROODA: an institutional project of long distance education**. Journal of Science Education and Technology, v. 15, n. 2, p. 159-167, 2006.

BENNER, Gary. et al. **Developing a high performance Linux Cluster at low cost**. Rotorua, NZ: School of Business and Computing, Waiariki Institute of Technology, 2010. Disponível em: <<http://www.citrenz.ac.nz/conferences/2004/benner.pdf>>. Acesso em: 23 ago. 2013.

COLE, J.; FOSTER, H. **Using Moodle: Teaching with the Popular Open Source Course Management**. Editora O'REILLY. Second Edition. 2007.

CRISTIANO, M. A. S. et al. O Moodle como recurso tecnológico para o ensino superior no curso de museologia. **Revista Electrónica de Investigación y Docencia**. Jul. 2012.

COELHO, José Silva. et al. **Implementação de ferramentas específicas ao modelo pedagógico da UAB na plataforma moodle**. Universidade Aberta, 2010. Disponível em: <<http://www.uab.pt>>. Acesso em: 23 ago. 2013.

DANTAS, M.A.R. **Tecnologias de Redes de Comunicação e Computadores**. Rio de Janeiro: Editora Axcel Books do Brasil, ISBN 85-7323-169-6, 2002.

DOUGIAMAS, M. and TAYLOR, P.C. (2003): **Using Learning Communities to Create an Open Source Course Management System (Moodle)**. National Key Center for Science and Mathematics Education, Curtin University of Technology, Australia, World Conference on Educational Multimedia, Hypermedia and Telecommunications (EDMEDIA). 2003.

MACHADO, Michael., TAO, Eric. **Blackboard vs. Moodle: Comparing User Experience of Learning Management Systems**. School of Information Technology and Communication Design, California State University Monterey Bay. 2007.

MIHAILESCU, E. **An Overview of Open Projects in Contemporary E-Learning: A Moodle Case Study, Studies in Computational Intelligence**. Volume 217/2009, Springer Berlin / Heidelberg. 2009.

MIHAILESCU, E. **Developments In Modular Architecture Of Learning Management Systems (LMSs), A Moodle Case Study, The 3rd International Conference on Telecommunications, Electronics and Informatics**. Chisinau, R. Moldova. 2010.

NICOLA, Matthias; JARKE, Matthias. **Performance Modeling of Distributed and Replicated Databases**. VOL. 12, NO. 4, julho / agosto 2000.

REIS, Flávio Menezes; VIEIRA, Alexandre Timm. **Cluster para alta disponibilidade + balanceamento de carga e ferramenta para medição de performance**. RS: Ulbra. 2006. Disponível em: <<http://www.seminfo.com.br/anais/2006/pdf/a02.pdf>>. Acesso em: abr. 2013.

RISTA, C. et al. **OSCAR: Um Gerenciador de Agregado para Ambiente Operacional Linux**. ERAD, Pelotas, 13 janeiro 2004.

SCHMIDT, KLAUS. **High Availability and Disaster Recovery Concepts, Design, Implementation**, 2006, XII, 410 p.

STALLINGS, W. **Computer organization and architecture**, Eighth Edition. Prentice Hall.

2010.

TARREAU, Willy. **Making application scalable with load balancing**. Revision 1.0. September 2006.

WHEIN-JEN, Chen. et al. **High Availability and Scalability Guide for DB2 on Linux, UNIX, and Windows**. RedBooks. First Edition. September 2007.

YASAR, O.; ADIGUZEL,T. **A working successor of learning management systems: SLOODLE**. Procedia Social and Behavioral Sciences , Stanbul, 22 January 2010.

