

O uso da computação distribuída no gerenciamento de informações acadêmicas, integrando *softwares* através de um *middleware* utilizando como laboratório de pesquisa a Universidade Estadual do Piauí.

Ana Patrícia de Sousa¹, João Batista Oliveira Silva², Laiton Garcia dos Santos³, Rubens dos Santos Lopes⁴

Instituto Federal do Piauí (IFPI) – Campus Floriano (PI) – Brasil

¹anapatricia3636@gmail.com; ²joaobatistatads@gmail.com;
³laitongarcia@hotmail.com; ⁴rubens_roc@outlook.com

Abstract. *Through distributed computing techniques, this study develops a platform to support the management of educational institutions. The system performs the management students of information, technical servers, teachers and contractors of the Department of Academic and graduation Control (DCAD) and the Library of the State University of Piauí (UESPI) - Campus Dr. Josefina Demes. In technical sense, this system implements the data distribution by replication techniques and by fragmentation using a middleware securely and transparently via a computer network. The data of the relationship of academic control system CAD are fragmented to the library management system Web Library in order to achieve automatic insertion of information's library users. Through the execution of this project is carried out a process of technology transfer between UESPI and the Federal Institute of Piauí - Floriano Campus (IFPI).*

Resumo. *Através de técnicas de computação distribuída, o presente estudo desenvolve uma plataforma de apoio à gestão de instituições de ensino. O sistema realiza o gerenciamento de informações de discentes, servidores técnicos, docentes e terceirizados do Departamento de Controle Acadêmico e Diplomação (DCAD) e da Biblioteca da Universidade Estadual do Piauí (UESPI) - Campus Dr^a Josefina Demes. De forma técnica, este sistema implementa a distribuição de dados por meio de técnicas de replicação e fragmentação pelo uso de um middleware de forma segura e transparente por meio de uma rede de computadores. Os dados das relações do sistema de controle acadêmico CAD são fragmentados para o Sistema de gestão de bibliotecas Web Library de modo a realizar-se a inserção automática de informações dos usuários da biblioteca. Através da execução deste projeto é realizado um processo de Transferência de Tecnologia entre a UESPI e o Instituto Federal do Piauí - Campus Floriano (IFPI).*

1. Introdução

Diante da crescente demanda pela disponibilidade, confiabilidade e integridade, requisitos imprescindíveis no âmbito do desenvolvimento de *software*, torna-se cada vez

mais necessário que as informações estejam à disposição dos seus usuários no momento que são requisitadas e de forma segura. Os sistemas distribuídos contribuem com isso ao permitir a disponibilidade das informações através do uso de *softwares* de acesso, e gerenciamento de dados adequados interligados por redes de computadores.

As dificuldades de adaptar os sistemas para se comunicarem e compartilharem informações com outros sistemas mais novos e avançados tecnologicamente são grandes e complexas. Assim, torna-se importante o desenvolvimento de sistemas que permitam o acesso e a integração de dados existentes em diferentes fontes.

A utilização de técnicas de programação distribuída visa cumprir metas importantes baseadas em funcionalidades cujo objetivo é prover soluções a fim de proporcionar um melhor uso da capacidade computacional, propiciando vantagens como à interoperabilidade, o desempenho, a escalabilidade, a conectividade, a confiabilidade, a transparência, a tolerância à falhas e segurança em ambientes heterogêneos e homogêneos. Assim, cresce o número de aplicações que demandam acesso de forma integrada, uniforme em tais ambientes (DEITEL, 2007).

Nesse contexto, o presente trabalho realizou a aplicação de procedimentos de computação distribuída para tornar acessíveis em diversas máquinas informações contidas até então no banco de dados central do sistema CAD, utilizando para este fim, a técnica de replicação de dados. Este projeto descreve ainda os processos seguidos com o fim de fragmentar no banco de dados da primeira fase do *software Web Library* (anexo A), através da técnica de fragmentação de dados, informações de alunos armazenadas no banco de dados do *software CAD*. Para a aplicação das técnicas citadas foram utilizadas as melhores práticas de programação de computadores, manipulação de dados e configuração de redes de computadores.

2. Revisão da literatura

2.1 Distribuição de Sistemas

O ambiente computacional atual é composto por Sistemas de Informações autônomos, distribuídos e heterogêneos. Os sistemas são autônomos porque sua construção, manutenção e operação ocorrem de forma independente e sem a preocupação com integração com outros sistemas. “De forma mais simplificada, é possível dizer que um sistema autônomo é um grupo conectado, de um ou mais prefixos IP, executados por operadores de rede que têm uma única e bem definida política de roteamento.” (HAWKINSON e BATES, 1996).

Segundo Colours, Dollimore e Kindberg (2007), um sistema distribuído é aquele no qual os componentes interligados em rede se comunicam e coordenam suas ações por meio da troca de mensagens. Os autores completam que são exemplos de sistemas distribuídos: a internet, uma intranet e a computação móvel e ubíqua.

Tanenbaum e Steen (2007) define um sistema distribuído como um conjunto de computadores autônomos vistos pelo usuário como um sistema único e coerente.

Ainda conforme Tanenbaum e Steen (2007), de modo a suportar máquinas e redes heterogêneas e, ao mesmo tempo, oferecer uma visão de sistema único, os sistemas distribuídos são organizados costumeiramente por meio de uma camada de *software*, situada logicamente entre duas camadas, uma formada por usuários e aplicações e uma camada subjacente formada por sistemas operacionais e facilidades básicas de comunicação. Por isso às vezes os sistemas distribuídos são denominados *middleware*.

Essa abordagem é semelhante à utilizada nos ambientes cliente-servidor de três camadas: uma camada das fontes de dados a serem integradas, uma da aplicação cliente e uma camada intermediária com a função de integração dos dados denominada *middleware*. A utilização de sistemas *middleware* libera os desenvolvedores de aplicações de ter que se envolver com os detalhes de cada fonte de dados e das complexidades dos ambientes computacionais.

O *middleware* foi definido como uma camada de *software* cujo objetivo é mascarar a heterogeneidade e fornecer um modelo de programação conveniente para os programadores de aplicativos. Em particular, ele simplifica as atividades de comunicação de programas aplicativos por meio do suporte de abstração como a invocação a métodos remotos, a comunicação entre um grupo de processos, a notificação de eventos, o particionamento, posicionamento e recuperação de objetos de dados compartilhado entre computadores, a replicação de objetos de dados compartilhados e a transmissão de dados multimídia em tempo real (COLOURIS, DOLLIMORE E KINDBERG, 2007).

O termo *middleware* é aplicado a uma camada de *software* que fornece uma abstração de programação, assim como a ocultação da heterogeneidade das redes, do *hardware*, de sistemas operacionais e linguagens de programação (COLOURIS, DOLLIMORE E KINDBERG, 2007).

A figura 1 mostra a arquitetura de um sistema distribuído organizado como *middleware*.

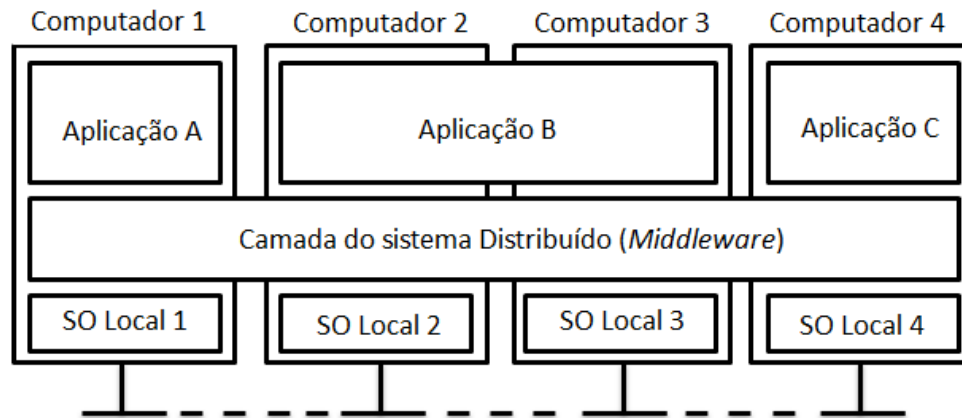


Figura 1- Sistema distribuído organizado como *Middleware* Fonte: Tanenbaum e Steen (2007).

Na figura 1 a aplicação B é distribuída para os computadores 2 e 3. O sistema distribuído oferece formas de comunicação entre os componentes de uma aplicação distribuída e entre diferentes aplicações ocultando, ao mesmo tempo, da melhor forma possível, as distinções entre hardware e sistemas operacionais para cada aplicação, tendo como principal objetivo facilitar o acesso a recursos remotos e seu compartilhamento de maneira controlada e eficiente (TANENBAUM E STEEN, 2007).

2.2 Sistema de Banco de dados distribuídos (SBDD)

Conforme Ozsu e Valduriez (2001) um banco de dados distribuído pode ser conceituado como uma coleção de vários bancos de dados inter-relacionados logicamente, distribuídos por uma rede de computadores.

Numata (2012) por sua vez define um sistema de dados distribuídos como a junção de redes de computadores com os sistemas de banco de dados. O autor completa ainda que o sistema de banco de dados distribuídos é um conjunto de vários bancos de dados logicamente inter-relacionados, separados fisicamente em lugares diferentes, geograficamente dispersos, distribuídos por uma rede de computadores.

A figura 2 apresenta a arquitetura de um ambiente em que se configura a distribuição de dados.

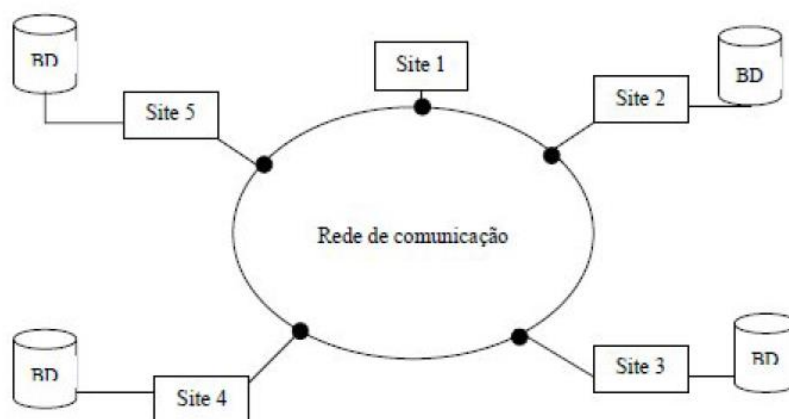


Figura 2 – Ambiente de Sistemas de Banco de Dados Distribuídos

Existem duas alternativas básicas de posicionamento dos dados no SBDD: particionados, onde conforme Ozsu e Valduriez (2001), o banco de dados se divide em várias partições disjuntas posicionadas cada uma em um site distinto; e replicados, estes podem subdividir-se em parcialmente replicados (ou parcialmente duplicados) e totalmente replicados (ou totalmente duplicados). Concordando com Fachin (2012), quando totalmente replicados, cópias do banco de dados inteiro são mantidas em cada site. Quando parcialmente replicado, cada partição do banco de dados é armazenada em vários sites, porém não em todos eles.

A técnica de replicação de dados é aplicada para aprimorar a confiabilidade ou melhorar o desempenho de sistemas distribuídos. Em um sistema de arquivos replicado é possível continuar trabalhando após a queda de uma réplica, por meio da comutação para uma das outras réplicas, configurando assim a confiabilidade do sistema. A replicação para alcançar desempenho como finalidade é aplicada quando um sistema precisa ser ampliado quantitativa e geograficamente. O problema da técnica de replicação é que a existência de muitas cópias pode ocasionar problemas de consistência (TANENBAUM e STEEN, 2007).

Na replicação assíncrona, se um banco é alterado, a alteração será propagada e aplicada para outro(s) banco(s) num segundo passo, sendo que esta poderá ocorrer em segundos, minutos, horas ou até dias depois (BEEHIVE, 2006).

A fragmentação divide-se em duas estratégias fundamentais: horizontal e vertical. Porém, os fragmentos também podem ser aninhados de forma híbrida. Na fragmentação horizontal uma relação é particionada em suas tuplas (linhas), assim, cada fragmento tem um subconjunto das tuplas de uma relação. Na fragmentação vertical

cada fragmento possui um subconjunto dos atributos da relação original, bem como sua chave primária. Também chamada de fragmentação mista ou aninhada, a fragmentação híbrida tem as duas estratégias de particionamento anteriormente citadas aplicadas uma após a outra, resultando em um particionamento estruturado em árvore (OZSU E VALDURIEZ, 2001).

Para O'Brien (2008), são vantagens dos Sistemas de Banco de Dados Distribuídos:

- Transparência na gestão dos dados, tendo como conceito de transparência a segregação do sistema de alto nível e os detalhes da sua implementação de modo que se tenha independência de dados. Tal característica pode ser dividida em transparência de rede, de replicação e de fragmentação.
- Maior confiabilidade nas transações.
- Melhor desempenho em consultas, pois podem ser executadas paralelamente às subconsultas.
- Facilidade de adaptação do sistema ao crescimento do banco de dados.
- Compartilhamento de recursos e informações.
- Disponibilidade para uma maior quantidade de usuários ao mesmo tempo.

A utilização de Sistemas de Banco de Dados Distribuídos tem algumas desvantagens que se relacionam diretamente com a sua complexidade de implementação:

- Necessidade de algoritmos adequados no processamento e otimização de consultas.
- Dificuldade de implementação, manutenção e gerenciamento.
- Dificuldade para evitar que problemas de infraestrutura (máquinas, rede) afetem o sistema.
- Garantir a segurança dos dados em SBDDs exige um maior nível de dificuldade (O'BRIEN, 2008).

Para Numata (2012), alguns fatores como gerenciamento transparente dos dados autonomia, integridade e economia com hardware, devem ser considerados no processo de implementação de Sistemas de Banco de Dados Distribuídos. Para o autor, as dificuldades de implementação de SBDDs estão relacionadas com a complexidade exigida por essa tarefa, logo a necessidade de pessoas qualificadas e da utilização de mecanismos de segurança, geram altos custos no desenvolvimento de banco de dados dessa natureza.

2.3 Redes de computadores

O termo rede de computadores refere-se a um conjunto de computadores autônomos interconectados (TANENBAUM, 1997).

Uma rede pode ser denominada como um conjunto de dispositivos conectados por *links* de comunicação (denominados nós). Um nó pode ser um computador, uma impressora ou qualquer outro dispositivo capaz de enviar e/ou receber dados gerados em outros nós. Atualmente, grande parte das redes usam processamento distribuído para executar uma tarefa entre muitos computadores, tal processo é bem mais eficiente que entregar todo poder de processamento a uma única máquina poderosa e deixá-la responsável por todo gerenciamento da rede (FOROUZAN, 2006).

Ainda conforme Forouzan (2006), as redes de computadores podem ser comparadas conforme alguns critérios. Dentre esses destacam-se pelo nível de importância: a performance, a confiabilidade e a segurança.

O desempenho de uma rede pode ser verificada pelo tempo de trânsito e o tempo de resposta. O primeiro refere-se ao intervalo de tempo que uma mensagem leva para passar de um dispositivo para outro, o segundo é o tempo decorrido entre uma solicitação e uma resposta. A performance de uma rede depende de muitos fatores tais como o número de usuários, o meio de transmissão, a capacidade do hardware conectado à rede e a eficiência do *software* que roda na rede. A confiabilidade de uma rede é medida pela frequência de falhas, o tempo de reconfiguração de *link* após uma falha e a robustez da rede numa catástrofe. A segurança da rede por sua vez, é um critério que tem como fim assegurar a proteção dos dados e das informações que trafegam na rede do acesso não autorizado (FOROUZAN, 2006).

Há um grande mal entendido entre os conceitos de redes de computadores e sistemas distribuídos. A principal diferença entre esses dois temas é que em sistemas distribuídos, a existência das máquinas autônomas é transparente (ou seja, tem seus detalhes ocultados do usuário) (TANENBAUM, 1997).

Tanenbaum (1997) aponta como vantagens das redes de computadores, o compartilhamento de recursos, que tem como objetivo colocar todos os dispositivos, equipamentos e dados de uma rede à disposição dos usuários da mesma; a confiabilidade, pois oferece alternativas de fornecimento de dados; redução de custos, logo é mais eficiente gerenciar várias máquinas de pequeno porte; escalabilidade, que consiste na possibilidade de aumentar gradativamente o desempenho do sistema proporcionalmente à demanda; e meio de comunicação, alternativa eficaz para equipes de trabalho dispersas geograficamente.

A segurança de uma rede engloba diversos aspectos, dentre os quais destacam-se por sua relevância a prevenção de interrupção, seja proposital ou acidental, do funcionamento da rede, prevenção de consulta ou cópia de dados mantidos no servidor, prevenção de modificação ou exclusão de dados, prevenção de acesso aos dados que circulam na rede, autenticação das solicitações de acesso aos dados e demais recursos da rede. Tais aspectos podem ser conseguidos através da implementação de procedimentos de segurança a nível de *hardware* e de *software* (GOUVEIA e MAGALHÃES, 2007).

2.4 Funções de Servidor

Uma função de servidor é um conjunto de programas que, são configurados para deixar o Sistema Operacional pronto para realizar atividades específicas para usuários ou outros computadores em uma rede. Um exemplo de função é o servidor de arquivos, responsável por fornecer tecnologias para gerenciamento de armazenamento, replicação de arquivos, gerenciamento de *namespace* distribuído, pesquisa rápida de arquivos e acesso simplificado aos arquivos pelo cliente (THOMPSON, 2010).

Outra função de servidor é a de Serviços de Domínio *Active Directory* (AD DS), essa função realiza o armazenamento de informações de usuários, computadores e demais dispositivos existentes na rede. O AD DS é responsável por auxiliar os administradores a gerenciar de forma segura essas informações e facilitar o compartilhamento de recursos e a colaboração de usuários. A função Servidor de DNS (*Domain Name System*) oferece um método padrão para a associação de nomes de endereços de internet numéricos, possibilitando a consulta de computadores de rede por usuários utilizando nomes fáceis de lembrar, quando comparados à longa série de números que compõem estes endereços (THOMPSON, 2010).

O servidor fornece ainda funções de Servidor de Aplicativos, Fax, Web, DHCP (*Dynamic Host Configuration Protocol*), Impressão e Documentos, Área de Trabalho

Remota, Acesso e Diretiva de Rede entre outras. Tais serviços podem ser configurados de acordo com as necessidades do projeto.

3. Materiais e Métodos

3.1 Levantamento Bibliográfico

- Foi realizada pesquisa bibliográfica sobre o que são Sistemas Distribuídos suas características e tipos de arquitetura;
- Estudou-se formas de implementar a camada de *software* que estabelece a abstração de programação, o *Middleware*;
- Estudou-se métodos e técnicas do desenvolvimento *web* utilizando ferramentas de Java EE e JPA.
- Foi levantada bibliografia sobre o funcionamento e alternativas de implementação de Banco de Dados Distribuídos seus conceitos, tipos de sistemas e suas arquiteturas;
- Realizou-se estudo sobre Redes de Computadores suas tecnologias, tipos e arquiteturas e a que se adequaria melhor ao projeto;
- Estudou-se os serviços oferecidos por servidores de modo a utilizar os que melhor se adequassem às necessidades do projeto.

3.2 Implementação

3.2.1 Análise

Foi feita a análise dos requisitos para o correto desenvolvimento do projeto baseado nas necessidades dos usuários para os quais estão direcionadas as plataformas.

3.2.2 Banco de dados

O banco de dados foi implementado utilizando o modelo relacional e desenvolvido na Linguagem de Consulta Estruturada (SQL) que representa o banco de dados como uma coleção de relações. Utilizou métodos necessários ao desenvolvimento do banco de dados da aplicação, como o Sistema Gerenciador de Banco de Dados (SGBD), que será o PostgreSQL. Também foram utilizadas técnicas de distribuição de dados, como replicação e fragmentação.

3.2.3 Aplicação

Foi implementada a camada de *software*, *middleware*, para que se obtivesse melhores formas de estabelecer comunicação e compartilhamento de informações da aplicação.

Aplicou-se métodos e técnicas do desenvolvimento *web* para implementar a primeira fase do *software* de gerenciamento de bibliotecas *Web Library*.

3.2.4 Arquitetura da rede

Os computadores foram interligados entre si por meio de uma rede local (*Local Area Network* - LAN) de topologia estrela utilizando um roteador como dispositivo responsável pela interligação.

3.2.5 Configuração do servidor

Para o servidor do projeto, foram configurados Serviços de Domínio *Active Directory*, DNS e Arquivos, para gerenciamento de privilégios de usuários, endereços na rede e cotas de diretórios, entre outras tarefas.

4. Resultados

4.1 Configuração do servidor Windows

Em uma máquina virtual que utiliza o sistema operacional Windows Server 2008 R2 foram configurados um servidor de domínio *Active Directory* configurado também, para realizar serviços de DNS e um servidor de arquivos. De modo a garantir a segurança das informações existentes no banco de dados do CAD, no servidor de domínio foram implementadas políticas de privilégios aos usuários dos computadores da rede. Dois usuários foram criados, um deles terá permissão de acesso apenas a pastas e arquivos pré-definidos, como texto. Documentos de backup, arquivos de mídia e configurações de rede, por exemplo, não poderão ser acessados por este usuário. O segundo usuário, possui privilégios de administrador e tem permissões para acessar e modificar qualquer diretório e tipo de arquivo.

O servidor de arquivos configurado neste trabalho realiza, entre outras funções, o gerenciamento de cotas para cada diretório, ou seja, para cada pasta de determinado usuário poderá ser definido o espaço disponível na mesma. Caso o limite de cotas pré-definidas esteja próximo de ser excedido, relatórios da manipulação incorreta desses diretórios são enviadas ao usuário responsável pelos mesmos.

4.2 Configuração do servidor Linux

No Sistema Operacional Linux Debian 8 foi configurado um serviço de gerenciamento de arquivos utilizando o servidor de arquivos Samba, serviço utilizado apenas em sistemas do tipo Unix. O samba é responsável por fazer o gerenciamento e compartilhamento de arquivos em uma rede.

Por questões de segurança oferecidas por Sistemas Operacionais Linux, no serviço de arquivos configurado no mesmo, está disposto o backup do banco de dados do *Web Library*, que utiliza o MySQL como SGBD.

4.3 Implementação do *middleware*

Para realizar o compartilhamento de dados entre os sistemas CAD e *Web Library*, foram utilizadas as técnicas de replicação assíncrona e fragmentação de dados. A mediação entre os *softwares* foi realizada por meio da abstração *middleware*.

O *middleware* desenvolvido neste trabalho é responsável pela realização da replicação assíncrona entre SGBDs PostgreSQL e fragmentação entre SGBDs PostgreSQL e Oracle. Os procedimentos foram realizados através de uma rede local que utilizou um switch como equipamento de interligação das máquinas. O desenvolvimento foi efetuado na linguagem Java utilizando recursos dos próprios SGBDs para obter as informações a serem replicadas. As tarefas realizadas, para a implementação do *middleware* e sua interação com os bancos de dados, estão organizadas no diagrama de atividade disposto na figura 3.

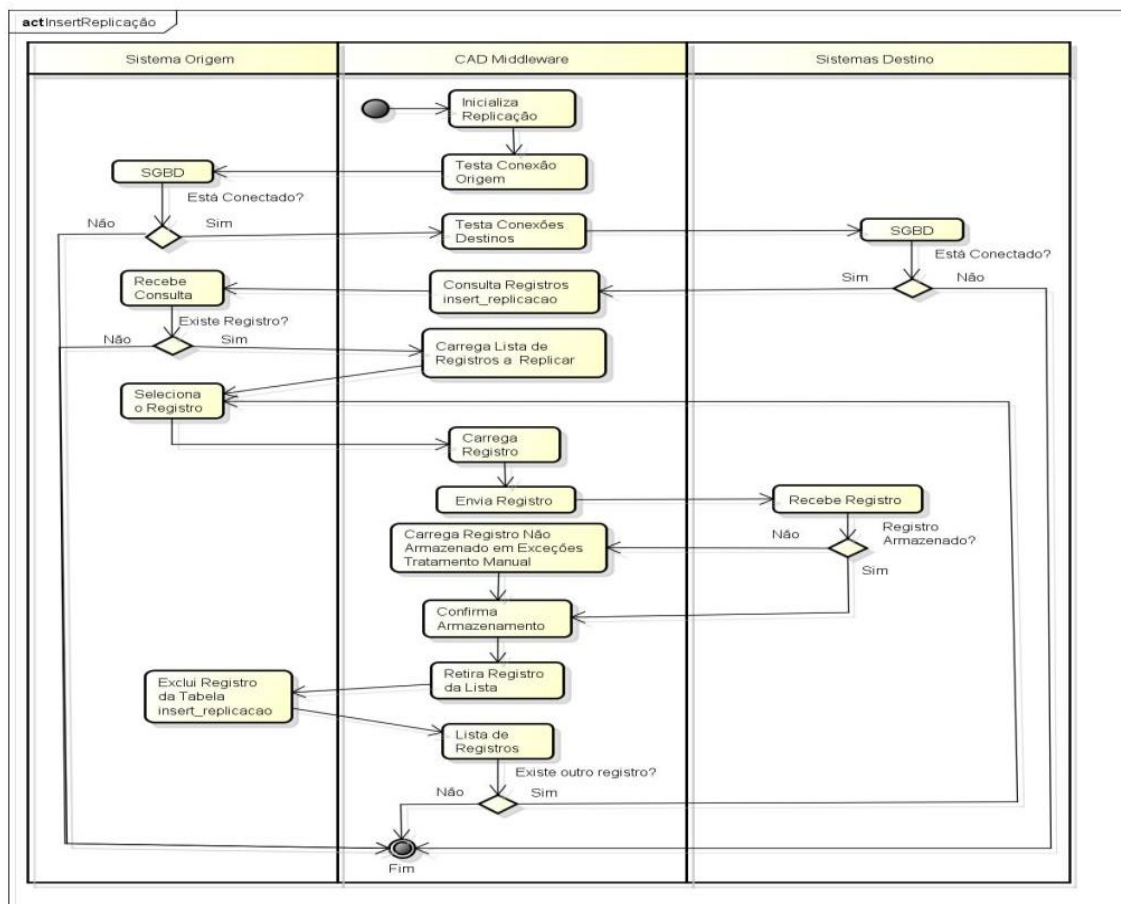


Figura 3: Diagrama de atividade do *middleware*

Seguem as descrições das atividades explicitadas no diagrama da figura 3.

- A classe que realiza a replicação e fragmentação é executada, inicializando os processos para a aplicação de tais técnicas;
- A conexão com o banco de dados local é realizada, caso esta tenha sido feita com sucesso, as conexões remotas com os bancos de dados de destino são testadas. Caso se tenha obtido êxito nas conexões com os bancos de destino, os registros do banco de dados de origem são consultados, se não, o processo é finalizado.
- Caso a consulta retorne algum registro, a lista de dados a serem replicados e fragmentados¹ é carregada e o primeiro registro enviado ao banco de destino. Caso o registro tenha sido armazenado, tal procedimento é confirmado e o registro replicado é removido da lista de registros a replicar.
- Se houver outro registro na lista o processo é retornado para que o mesmo seja selecionado, caso contrário, o processo de execução do *middleware* é finalizado.

¹ As técnicas são aplicadas ao mesmo tempo, porém em bancos de dados distintos.

A sequência de processos explicados anteriormente pode ser vista de forma mais detalhada no diagrama de sequência da figura 4.

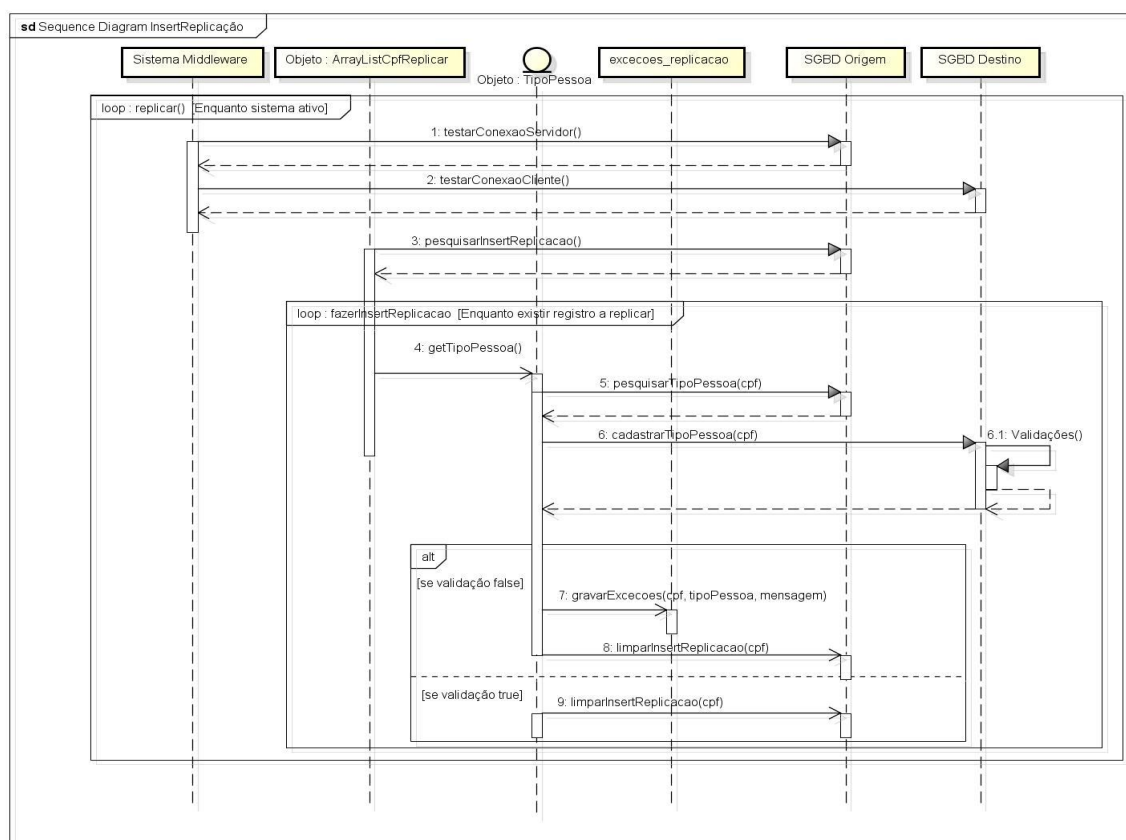


Figura 4: Diagrama de sequência dos processos realizados para a implementação do middleware.

Caso a conexão entre dois determinados nodos não se estabeleça no momento em que deveria ocorrer a replicação dos dados, o aplicativo executa a replicação em uma próxima conexão com o nodo de destino. A ferramenta replica os registros de um determinado banco de dados realizando alterações na estrutura das tabelas, a criação de novas tabelas e sequências não serão replicadas.

O funcionamento deste aplicativo necessita de um arquivo de configuração, o qual é utilizado em duas etapas: na inicialização do SGBD para começar a coletar informações para serem replicadas e na execução do replicador.

As exceções (conflitos) não são automaticamente solucionadas. A exceção é gravada no *log* e resolvida manualmente. O *middleware* bloqueia a replicação de um próximo registro enquanto não solucionar o registro anterior.

Conclusão

Este trabalho utilizou técnicas de computação distribuída como replicação e fragmentação de dados aplicadas por meio da utilização de uma rede de forma transparente ao usuário. Tais usuários são gerenciados por um servidor que oferece diversos tipos de serviços. A utilização dessas técnicas permite concluir que tais ferramentas são de grande utilidade para realizar o compartilhamento de dados acadêmicos a serem utilizados por plataformas com finalidades distintas, havendo homogeneidade ou não de bancos de dados. Assim, no presente trabalho foi realizada a

distribuição de dados entre sistemas acadêmicos por uma rede de computadores oferecendo disponibilidade, integridade, confiabilidade e segurança ao usuário.

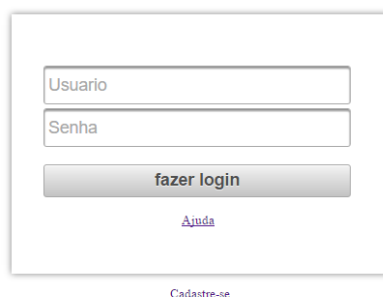
Referências

- BEEHIVE. **Beehive Replicator Agent**. 2006. Disponível em: <<http://www.beehive.com.br/replicator.html>>. Acesso em: 03 jun. 2006.
- COLOURIS, G.; DOLLIMORE, J.; KINDBERG, T. **Sistemas distribuídos: conceitos e projeto**. 4. ed. Porto Alegre: Bookman, 2007.
- DEITEL, Choffnes. **Sistemas Operacionais**. Prentice-Hall. 2007.
- FACHIN, A. **Banco de Dados Distribuídos**. Disponível em: <<http://www.slideshare.net/andrefachin/banco-de-dados-distribuidos-66037>>. Acesso em 21 de Outubro de 2015.
- FOROUZAN, B. A. **Comunicação de Dados e Redes de Computadores**. 3. ed. Porto Alegre: Bookman, 2006.
- GOUVEIA, J.; MAGALHÃES, A. **Redes de Computadores**. 1. ed. Rio de Janeiro: FCA Editora de informática. 2007.
- NUMATA, C. A. **Banco de dados distribuídos**. São Paulo: Faculdade de Tecnologia de São Paulo. 2012.
- OZSU, Tamer M.; VALDURIEZ. P. **Princípios de Sistemas de Banco de Dados Distribuídos**. 2. ed. Vandenberg D. de Souza. Rio de Janeiro: Campus, 2001.
- O'BRIEN, J. A. **Management Information Systems**, 7ª Ed, Mcgraw- Hill/Irwin, 2008.
- TANENBAUM, A. S. **Redes de Computadores**. Insight Serviços de Informática. 3. ed. Rio de Janeiro: Campus, 1997.
- TANENBAUM, A. S. STEEN, M. V. **Sistemas distribuídos: princípios e paradigmas**. 2. ed. São Paulo: Pearson Prentice Hall, 2007.
- THOMPSON, M. A, **Microsoft Windows Server 2008 R2 Fundamentos**. 1. ed. São Paulo: Editora Érica Ltda, 2010.

ANEXO A

A figura 4 apresenta a tela de login do *Web Library*, a tela é única para os diversos tipos de usuários, inclusive para administradores ficando disponíveis para este, funções de gerenciamento de publicações e usuários, realização de empréstimos e devoluções e configurações do sistema. Para os demais usuários, são disponibilizadas funções de administração de conta e renovação de exemplares.

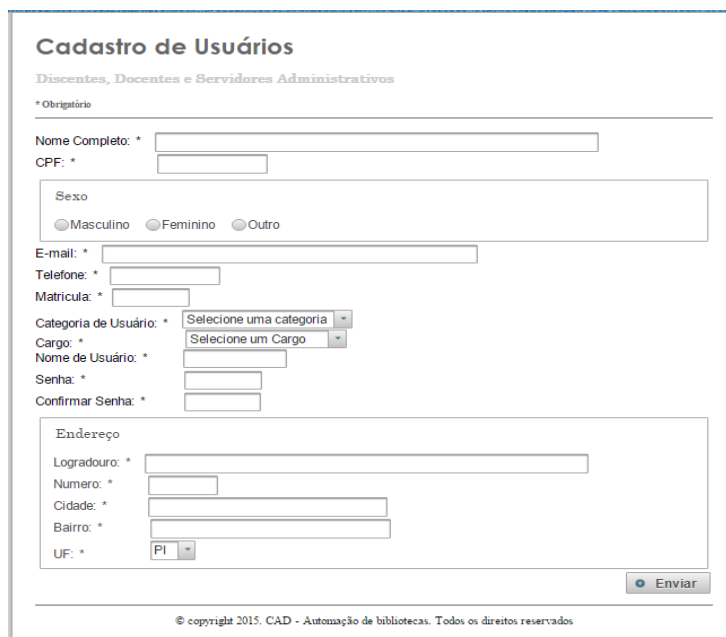
Insira suas informações para acessar o Web Library



A tela de login do Web Library apresenta um formulário com dois campos de entrada: 'Usuario' e 'Senha'. Abaixo desses campos, há um botão cinza com o texto 'fazer login'. Logo abaixo do botão, há um link azul com o texto 'Ajuda'. Na base da tela, há um link azul com o texto 'Cadastre-se'.

Figura 4: Tela de login da plataforma *Web Library*

A tela de cadastro que pode ser acessada através do link “cadastre-se” disposto logo abaixo do botão “fazer login” da tela de login, pode ser observada na figura 5.



O formulário de cadastro da plataforma Web Library, intitulado 'Cadastro de Usuários', é destinado a 'Discentes, Docentes e Servidores Administrativos'. Ele contém campos obrigatórios para: Nome Completo, CPF, Sexo (com opções Masculino, Feminino e Outro), E-mail, Telefone, Matricula, Categoria de Usuário (com uma lista suspensa), Cargo (com uma lista suspensa), Nome de Usuário, Senha e Confirmar Senha. Além disso, há uma seção para o endereço com campos para Logradouro, Numero, Cidade, Bairro e UF (com uma lista suspensa). Um botão 'Enviar' está localizado no canto inferior direito do formulário. No rodapé, há uma linha de copyright: '© copyright 2015. CAD - Automação de bibliotecas. Todos os direitos reservados'.

Figura 4: Formulário de cadastro da plataforma *Web Library*

Caso o usuário acesse o sistema com informações de administrador (Figura 5), uma tela para gerenciamento do sistema é disponibilizada, nela o administrador poderá gerenciar contas de usuários (Figura 6), gerenciar publicações (figura 7) realizar empréstimos e devoluções e configurar o sistema.

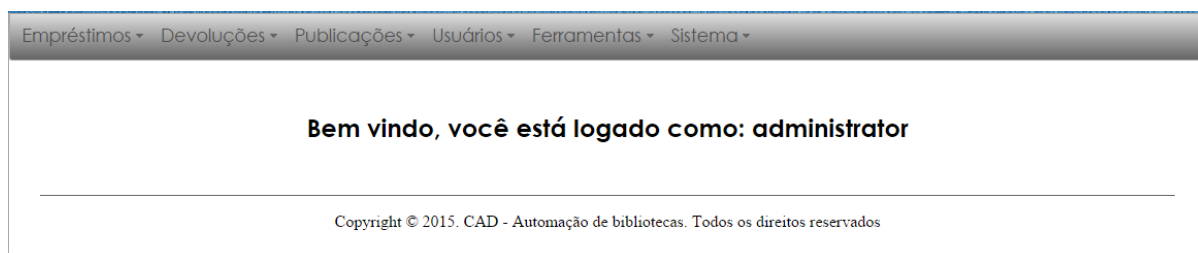


Figura 5. Tela de Administrador



Figura 6. Tela Gerenciar Usuários

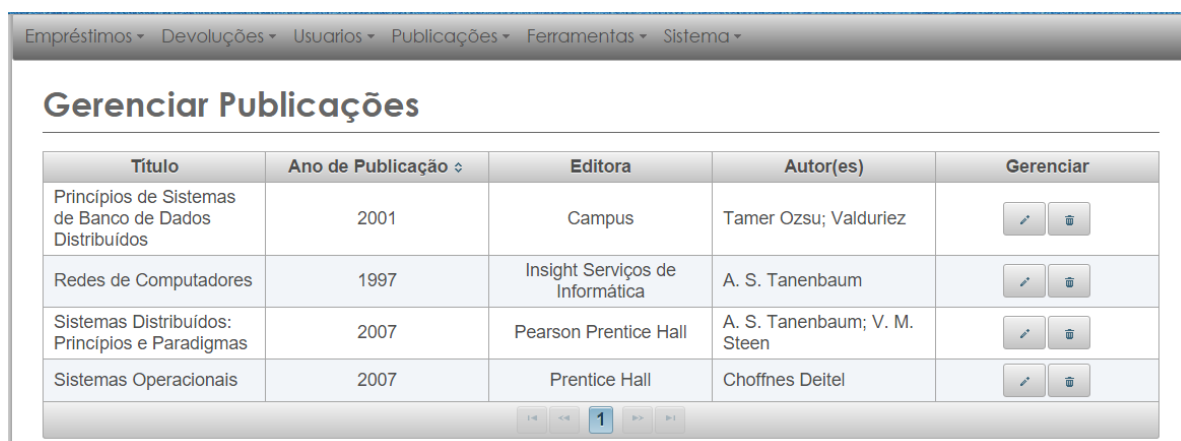


Figura 7. Tela Gerenciar Publicações