

Um modelo de catálogo distribuído para o Direto

*Marcelo Neres dos Santos
Cristiano Cachapuz e Lima
Universidade da Região da Campanha
{mns, cristiano}@urcamp.tche.br*

Resumo

Este trabalho visa analisar as características do serviço de diretório LDAP e propor um modelo para distribuição do catálogo do *software* DIRETO. Serão apresentados as principais características do DIRETO juntamente com os fundamentos dos serviços de diretório e do LDAP em particular juntamente com a apresentação de um estudo de caso, onde são descritos os passos para a instalação e configuração de servidores LDAP e os resultados de testes realizados com os servidores configurados, principalmente no tocante a replicação de conteúdo entre os servidores.

Palavras-chave: Catálogo distribuído, LDAP, DIRETO, serviços de diretório, replicação, sistemas distribuídos.

Abstract

The goal of this work is analyse the LDAP directory service features and propose a distribution model for the DIRETO adress book. The DIRETO features and the directory services concepts and, in this particular case, LDAP fundamentals will be explained. A estudy case will be presented, the steps to install and configure LDAP servers and the results of the tests realized with the servers configured will be presented, mainly concerned in features about replication between servers.

Key words: Adress book, LDAP, DIRETO, directory servers, replication, distributed systems.

1 Introdução

O uso de redes de computadores, e da Internet de forma especial, potencializa a disponibilização de informações e estimula mudanças nos processos das mais variadas áreas do conhecimento. Para que essa interação ocorra, são utilizados sistemas especializados para suporte à troca de informações eletrônicas e/ou para realização de transações computacionais de forma distribuída. Normalmente, as redes de computadores apresentam características heterogêneas. Essa heterogeneidade ocorre tanto em relação ao *hardware* quanto ao *software*, sendo essa situação válida tanto em ambientes acadêmicos quanto comerciais.

Deste modo, uma das preocupações atuais, tanto em ambientes acadêmicos como empresariais, consiste na disponibilização de informações distribuídas bem como na necessidade de padronizar a forma de comunicação entre os membros dos grupos de trabalho. Para padronizar a comunicação, deve-se obter um produto viável em termos de custos econômicos e ao mesmo tempo com características refinadas, como filtros inteligentes e identificação automática de perfis de usuário e trabalho, que facilitem o uso.

Este trabalho visa a elicitação de requisitos técnicos sobre o LDAP (Lightweight Directory Access Protocol) bem como a elaboração de uma proposta de distribuição e replicação do conteúdo do catálogo do *software* DIRETO.

2 O Projeto DIRETO

A PROCERGS (Companhia de Processamento de Dados do Rio Grande do Sul) e a maioria de seus clientes do Estado do Rio Grande do Sul trabalham desde 1992 com as ferramentas de Correio Eletrônico MEMO e Notes. Alguns problemas e dificuldades dos usuários com o uso destas ferramentas motivaram a criação de um *software* que atendesse as necessidades dos usuários.

A PROCERGS decidiu por desenvolver sua própria solução. A solução, materializada através do produto DIRETO, baseou-se totalmente em protocolos padrão Internet. O DIRETO é um produto modularizado que utiliza os protocolos *vCalendar* para agenda, *SMTP* e *IMAP* para correio e LDAP para catálogo.

O DIRETO está sendo desenvolvido utilizando a linguagem de programação *Java* e também utiliza outros *softwares* de distribuição livre, como o sistema operacional *Linux* e o banco de dados *PostgreSQL*. Tais recursos serão brevemente explanados posteriormente neste trabalho.

Além disso, o DIRETO será um *software* livre, com distribuição e licenças gratuitas, a partir de um *site* construído e mantido pela PROCERGS, seguindo as regras de *software* livre.

Os recursos descritos neste trabalho, por serem considerados de caráter avançado dentro do contexto do projeto, serão implementados em versões futuras do DIRETO.

3 Proposta de utilização de catálogo distribuído no DIRETO

3.1 Catálogo do DIRETO

Obter informações de maneira ágil e precisa está se tornando uma prioridade cada vez maior na atualidade, tanto para empresas quanto para usuários domésticos. Para aumentar a funcionalidade, facilidade de uso e permitir a administração das informações são freqüentemente armazenadas em um banco de dados especial, muitas vezes chamado de *diretório*, que nada mais é do que um catálogo de recursos.

Os serviços de diretórios estão se tornando cada vez mais indispensáveis para a construção de uma arquitetura computacional capaz de suprir as necessidades dos mais variados tipos de usuários de maneira eficaz. A principal função de um serviço de diretório é informar aos usuários onde determinada informação ou recurso está localizado na rede e quais suas características [CAR 1998].

Atualmente, está sendo utilizada no DIRETO uma abordagem na qual as informações são armazenadas no catálogo do *software* de maneira centralizada, em um único servidor. O principal problema desta abordagem, é que, à medida que o volume de informações armazenadas aumenta, o acesso a estas se torna cada vez mais lento.

A adoção de uma abordagem distribuída para o catálogo do DIRETO traz inúmeras vantagens em relação à abordagem atualmente utilizada. Pode-se observar que, a medida em que as informações são adicionadas aos diferentes servidores, que compõem o sistema como um todo, uma ampla malha de serviços de diretório tende a interconectar-se uns com os outros, formando o que muitos chamam de *Serviço de Diretório Global*.

Entretanto, muitas vezes um único servidor é insuficiente para gerenciar as diversas requisições dos usuários do sistema. Torna-se necessário então, utilizar vários servidores simultaneamente, um destes servidores será o *mestre* e os demais os *escravos*. Para utilizar tal abordagem, é necessário utilizar-se de técnicas de replicação de conteúdo, para manter as diversas cópias do catálogo sincronizadas.

3.2 Lightweight Directory Access Protocol

Um diretório é freqüentemente descrito como uma banco de dados, porém ele é um banco de dados especializado, que possui características que o diferem dos bancos de dados relacionais de propósito geral [JOH 1998]. Uma das principais diferenças entre os diretórios e os bancos de dados relacionais, é que os diretórios são muito mais acessados do que alterados. Pelo fato dos diretórios possuírem capacidade de suportar um alto volume de consultas, eles são tipicamente otimizados para leitura. Operações de inserção e alteração de dados geralmente são limitadas aos administradores do sistema ou aos proprietários da informação.

Uma outra importante diferença entre os diretórios e os bancos de dados relacionais é que os diretórios não possuem suporte para transações. Transações são operações atômicas, ou seja, devem ser executadas na sua totalidade ou, caso ocorra algum problema que impeça a sua execução, todas as operações realizadas dentro da transação deverão ser desfeitas.

A última grande diferença entre os diretórios e os bancos de dados é quanto ao método de consulta as informações. A maioria dos bancos de dados suportam a linguagem SQL, que fornece um método poderoso e padronizado de consulta e alteração de informações em um banco de dados. Já os serviços de diretórios utilizam um protocolo simplificado, otimizado para consulta, que pode ser usado em aplicações relativamente simples.

Os diretórios geralmente são acessados utilizando o modelo cliente/servidor de comunicação. Uma aplicação que deseja acessar uma informação em um diretório não acessa o diretório diretamente. A aplicação chama uma função ou API que gera uma mensagem que será enviada a outro processo. Este segundo processo acessa a informação no diretório em favor da aplicação que gerou a mensagem. Os resultado do acesso é então retornado a aplicação que gerou o pedido (figura 1).

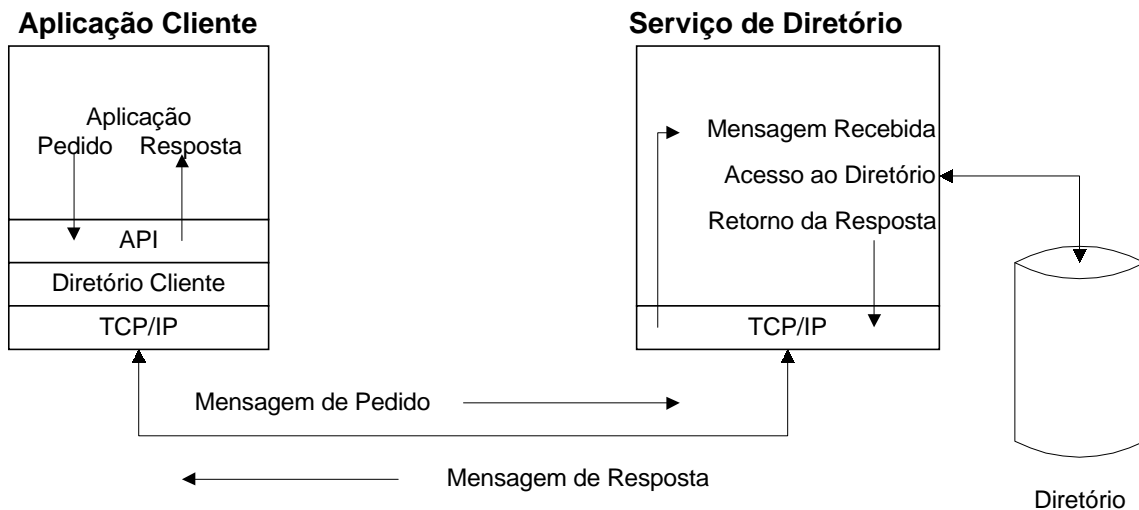


Figura 1 – Interação cliente/servidor de um diretório [JOH 1998]

O pedido é realizado pelo cliente do diretório, o processo que busca a informação no diretório é chamado de servidor de diretório. Normalmente, os servidores fornecem um serviço específico aos clientes. Muitas vezes um servidor pode tornar-se cliente de um outro servidor, para poder obter as informações necessárias para processar um pedido.

Um diretório pode ser centralizado ou distribuído. Se um diretório é centralizado, existe somente um servidor de diretório que fornece acesso ao diretório. Quando um diretório é distribuído, existem pelo menos dois servidores de diretório fornecendo acesso ao diretório.

Quando um diretório é distribuído, a informação armazenada no diretório pode ser particionada ou replicada. Se a informação é particionada, cada servidor armazena um único subconjunto de informações, ou seja, cada entrada no diretório é armazenada por somente um servidor. Se a informação é replicada, uma mesma entrada no diretório pode ser armazenada por mais de um servidor.

Entretanto, torna-se necessário salientar, que na realidade o LDAP não é um serviço de diretório. LDAP apenas define um protocolo de comunicação, ou seja, define o transporte e o formato das mensagens usados pelos clientes para acessar um diretório do tipo X.500. Mesmo assim, muitos usuários freqüentemente utilizam o termo diretório LDAP [JOH 1998], sendo que esta terminologia também será utilizada neste trabalho.

O X.500 define como os diretórios devem ser estruturados. O X.500 é executado sobre a camada OSI e requer uma significativa quantidade de recursos para ser executado enquanto o LDAP é executado diretamente sobre o TCP e fornece mais funcionalidade que o X.500 a um custo bem menor [WEB 2001] [UMC 1996].

Em particular, LDAP vem posicionando-se como um padrão capaz de fornecer um acesso aberto aos serviços de diretório para Internet bem como integrar outros serviços de diretórios e tornar um serviço de diretório global uma verdadeira realidade.

3.2.1 Arquitetura do LDAP

Um diretório armazena e organiza as informações em estruturas de dados chamadas *entradas*. Uma entrada de diretório geralmente descreve um objeto do mundo real, como uma pessoa, uma impressora, etc. Cada entrada possui um nome único, que a identifica, chamado nome distinto (*distinguished name*). Um nome distinto consiste de uma seqüência de partes chamada nome distinto relativo (*relative distinguished name*). As entradas são organizadas em estruturas em árvore, baseadas em seus nomes distintos. Esta árvore de entradas de diretórios é chamada *Árvore de Informações de Diretório (Directory Information Tree)*.

Uma entrada de diretório descreve um objeto genérico. Uma classe de objeto (*object class*) é uma descrição geral de um objeto, ela contém a descrição dos atributos que compõem um determinado tipo de objeto, assim como uma classe de objetos em uma linguagem de programação orientada a objetos. As classes de objetos que um servidor de diretório pode armazenar e os atributos que estes podem conter são denominados *esquema de diretório*. Além disto, um esquema de diretório define quais classes de objeto são opcionais e a sintaxe de cada atributo.

A tabela a seguir mostra alguns tipos de atributos mais comumente encontrados em um diretório LDAP e sua *string* de representação.

Tipo de Atributo	String
Nome (<i>CommonName</i>)	CN
Localidade (<i>LocalityName</i>)	L
Estado (<i>StateOrProvinceName</i>)	ST
Organização (<i>OrganizationName</i>)	O
Departamento (<i>OrganizationalUnitName</i>)	OU
País (<i>CountryName</i>)	C
Endereço (<i>StreetAdress</i>)	STREET
Domínio (<i>domainComponent</i>)	DC
Usuário (<i>userid</i>)	UID

Tabela 1 – Representação dos tipos de atributos [JOH 1998]

3.2.2 Modelos de Configuração do LDAP

Um servidor LDAP pode ser configurado para trabalhar de diversas maneiras. Esta seção descreve brevemente os principais modelos de configuração de um diretório LDAP e como ele interage com os demais componentes de um sistema distribuído em cada uma destas configurações.

- LDAP como um servidor local

Nesta configuração, o servidor LDAP fornece um serviço de diretório somente para um domínio local, não interagindo com outros servidores de diretório de maneira alguma, conforme a figura 2.

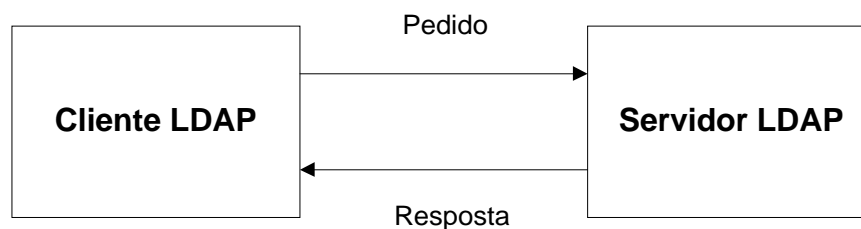


Figura 2 – LDAP como um servidor local

- LDAP como um servidor de diretório distribuído

Muitas vezes, um único servidor LDAP torna-se insuficiente para suportar toda a demanda de requisições dos usuários. Torna-se necessário então, adotar uma abordagem distribuída de servidores LDAP. Nesta abordagem, cada servidor armazenará uma parte da árvore de informações de diretório. No entanto, haverá nos servidores, mecanismos que farão com que estes se referenciem, para que quando um pedido de um usuário não seja atendido por um determinado servidor, este possa passar como resposta da requisição em questão, um outro servidor, até que o pedido seja atendido. Este mecanismo de referência entre servidores se dá através da inserção de uma entrada de referência no diretório de um determinado servidor, esta entrada aponta para um outro servidor, onde a informação provavelmente esteja armazenada.

A figura 3 ilustra esta interação. Um cliente LDAP realiza um pedido ao Servidor 1 (1), este pedido é respondido com uma referência ao Servidor 2 (2). O cliente LDAP então envia novamente o seu pedido ao Servidor 2 (3), que por sua vez, fornece as informações pedidas ao cliente (4).

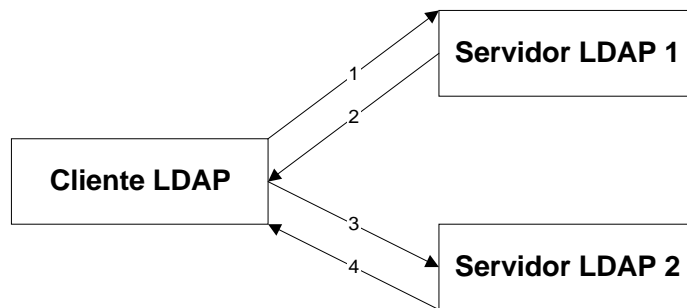


Figura 3 – LDAP como um servidor distribuído

- LDAP como um servidor com o conteúdo replicado

Esta configuração geralmente é utilizada quando uma única cópia da árvore de informações de diretório é insuficiente para atender a demanda de pedidos realizada pelos usuários. O LDAP possui um *daemon* chamado *slurpd*. Em um ambiente onde o conteúdo dos servidores LDAP são replicados, este pequeno programa é encarregado de propagar as informações do servidor LDAP principal (mestre) para um ou mais servidores secundários (escravos). A utilização de uma abordagem replicada aumenta em muito a confiabilidade e a disponibilidade das informações em um serviço de diretório. A figura 4 exemplifica como se dariam as interações entre os componentes do sistema quando da utilização de um servidor com o conteúdo replicado.

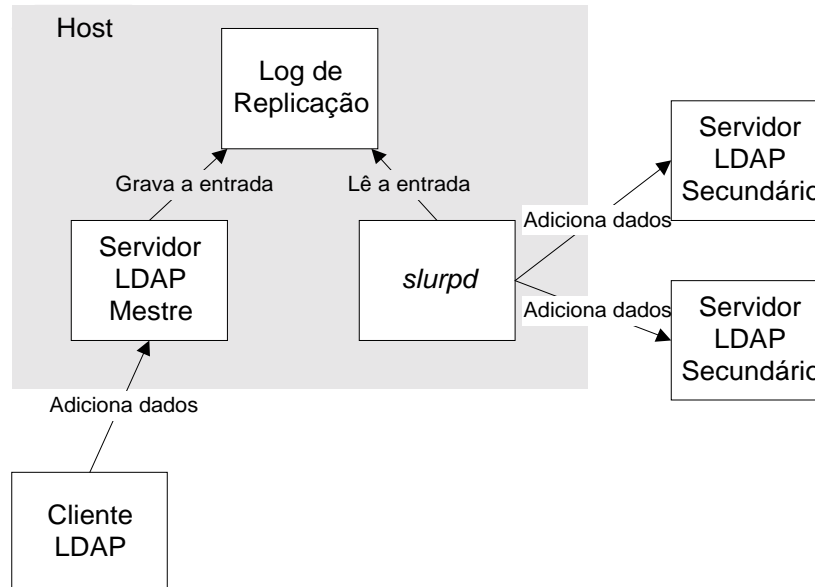


Figura 4 – LDAP como um servidor com o conteúdo replicado

3.3 Proposta de um catálogo distribuído utilizando LDAP

O DIRETO utiliza o LDAP como ferramenta para a implementação do seu catálogo. Atualmente o *software* DIRETO possui seu catálogo implementado de maneira centralizada. Projetar um catálogo que possua flexibilidade suficiente para acomodar possíveis mudanças nas organizações

que se utilizarão do mesmo, é uma tarefa bastante complexa. Um bom projeto, ajudará a reduzir custos e a ganhar tempo à medida que as organizações e o próprio catálogo aumentam e modificam-se.

Quando se projeta um serviço deste tipo, o projeto como um todo pode ser dividido em vários projetos menores, como por exemplo: levantamento do conteúdo do catálogo, criação das estratégias de controle de acesso, planejamento da rede, etc. Devido a existência de um catálogo centralizado para o DIRETO, muitas destas tarefas já foram realizadas. Esta proposta visa somente a apresentação das características de replicação de um servidor LDAP em um ambiente distribuído. A utilização de uma estratégia de replicação torna-se essencial no momento que se deseja criar uma infraestrutura capaz de suportar disponibilidade, escalonamento e gerenciabilidade das informações.

A PROCERGS, juntamente com as demais entidades que compõem a equipe de desenvolvimento, entende que pela vasta amplitude geográfica que os órgãos que utilizarão o DIRETO abrangem, bem como pelas diversas diferenças de infra-estrutura, é de extrema relevância a utilização de uma estrutura utilizando servidores LDAP replicados. A utilização desta abordagem baseia-se nos seguintes fatos:

3.3.1 Aumento da disponibilidade das informações

Em um único servidor, estarão armazenadas as informações de todo o sistema, o que acarretará um menor tráfego nas redes externas as organizações, que atualmente constituem-se no grande “gargalo” no acesso as informações do catálogo. Não haverá mais a necessidade de se acessar uma rede externa para se obter a informação desejada. Isto também diminuirá o custo de implantação do DIRETO, visto que, muitos órgãos ainda utilizam acesso discado para conectarem-se a uma Intranet ou a Internet. O acesso a rede externa será necessário somente no momento da sincronização dos servidores LDAP que compõem o sistema ou no momento da inserção dos dados no servidor principal.

3.3.2 Aumento do desempenho do sistema

O fato das informações serem acessadas quase na sua totalidade na rede local, aumenta muito a performance do sistema. Não haverá a necessidade de conectar-se nas redes externas a uma determinada organização. Estas redes geralmente apresentam uma largura de banda menor do que uma rede local, o que diminui em muito a performance do sistema.

3.3.3 Aumento na confiabilidade do sistema

Caso um servidor falhe, mesmo sendo o local, saberemos que as informações utilizadas pelos usuários do DIRETO poderão ser encontradas em outro servidor, já que o conteúdo estará replicado. Quando o servidor que falhou voltar a funcionar, o LDAP automaticamente fará a sincronização do conteúdo, passando o sistema a operar normalmente.

3.4 Funcionamento da replicação em LDAP

A replicação constitui-se em um dos principais métodos para aumento de performance, disponibilidade e confiabilidade em LDAP.

A disponibilidade das informações no catálogo do DIRETO é quase que indispensável, dado o tipo de aplicação em que se constitui este *software*. Esta disponibilidade muitas vezes é obtida a partir da eliminação de pontos de falha no sistema ou através da redução do seu impacto. Para isso, é necessária a utilização de *hardware* e *software* redundante para difundir os riscos.

Uma abordagem bastante simples para aumentar a disponibilidade do catálogo do DIRETO é a utilização de um servidor LDAP principal e um ou mais servidores secundários. Através da replicação dos dados entre os servidores, os pontos de falha no sistema estarão eliminados tanto para *hardware* quanto para *software*. Esta solução com um servidor principal e um ou mais servidores secundários

normalmente fornece alta disponibilidade para as consultas ao conteúdo do catálogo. geralmente, pedidos de gravação no catálogo são direcionados somente ao servidor principal, visto que ocorreram em um número bastante menor.

A figura 5 ilustra um mecanismo para gerenciar o redirecionamento de um cliente se um servidor falhar. Este redirecionamento pode ser realizado automaticamente com uma técnica de balanceamento de carga utilizando-se um roteador designado para isto. Este roteador redireciona o pedido do cliente para um dos servidores, baseado em critérios configuráveis. É importante que os pedidos subsequentes de um mesmo cliente sejam encaminhados ao mesmo servidor.

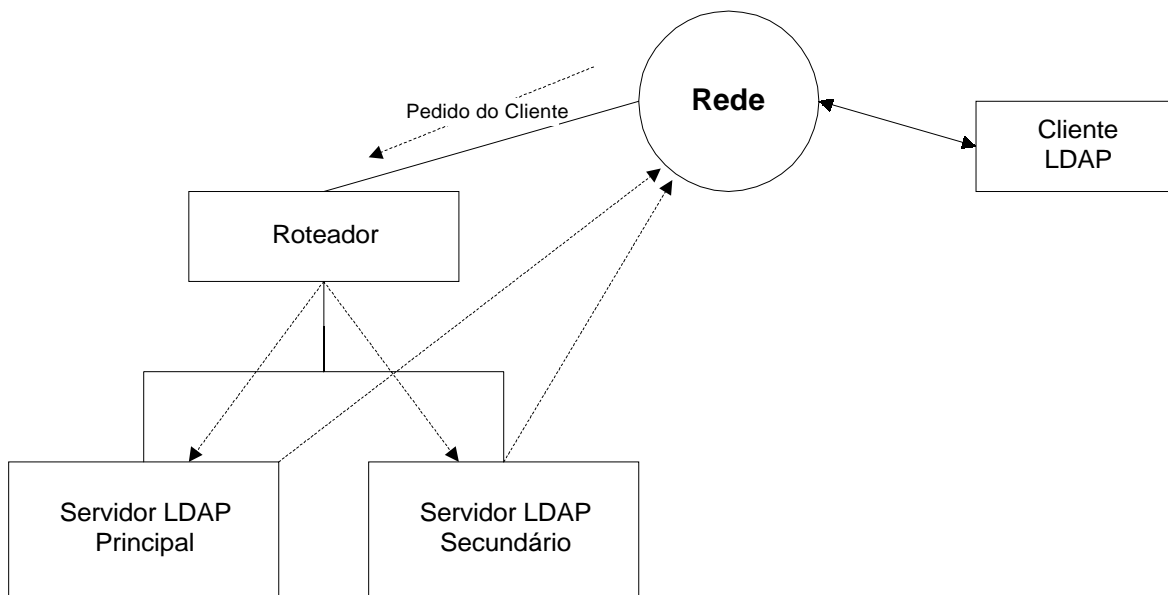


Figura 5 – Balanceamento de carga em um servidor LDAP replicado [JOH 1998]

Em aplicações onde não há a necessidade de uma forte consistência das informações pode-se utilizar um esquema com múltiplos servidores primários, ou seja, onde cada servidor pode ter seus dados alterados. Tal abordagem aumenta em muito a performance para gravações no catálogo, porém, a consistência dos dados fica bastante comprometida.

Um cliente LDAP pode direcionar suas operações de gravação e leitura para uma única réplica, entretanto, este cliente não terá assegurada a consistência das informações, visto que uma outra operação poderá estar ocorrendo ao mesmo tempo em outra réplica do catálogo.

Propõe-se entretanto, que se introduza no módulo de catálogo do DIRETO, determinadas regras, que façam com que todas as alterações tendam a serem executadas em uma única réplica do catálogo. Isto asseguraria que a aplicação em questão realizasse alterações em série, fornecendo então uma alta consistência aos dados.

Na presente proposta, considera-se também a largura de banda e sua confiabilidade. Em muitos casos, pode ser necessária a distribuição de uma réplica em uma LAN com uma conexão bastante lenta ao servidor LDAP principal. Neste caso, o servidor primário de um cliente em particular pode ser o servidor que está dentro de sua própria LAN, enquanto o servidor secundário será então o servidor central (principal), acessado através de uma WAN. A figura 6 exemplifica tal iteração.

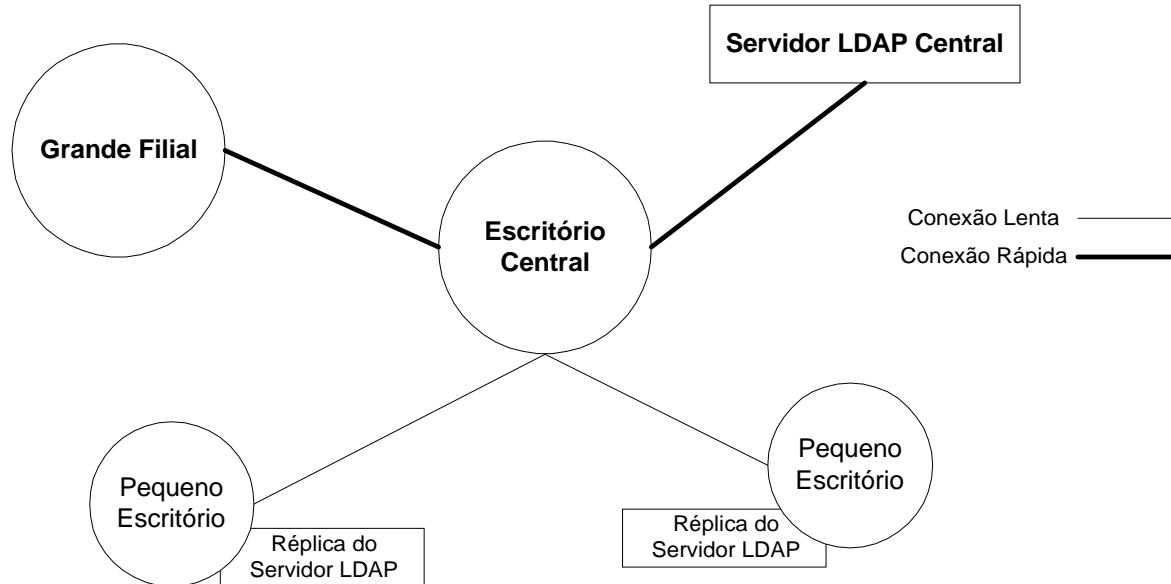


Figura 6 – Servidor LDAP replicado em um ambiente com diferentes larguras de banda

Na figura 6, uma organização possui um escritório central, uma grande filial com uma largura de banda suficientemente grande para comportar o acesso ao catálogo central e dois pequenos escritórios, que possuem uma infraestrutura de rede bastante simples. Nestes dois escritórios existem réplicas do catálogo que são consideradas como servidor primário para as aplicações destes escritórios, visto que a largura de banda destas unidades não é satisfatória para uma iteração direta com o catálogo central.

Na medida em que mais e mais aplicações utilizam o catálogo, a necessidade por escalonamento também aumenta. A realização desse escalonamento em LDAP normalmente se dá através do aumento da capacidade de *hardware* (*upgrade*) ou, seguindo o exemplo do aumento da disponibilidade das informações, através da replicação dos dados.

3.4.1 Funcionamento do *slurpd*

Conforme citado anteriormente, em determinados ambientes, um único servidor LDAP pode tornar-se insuficiente para gerenciar uma quantidade muito grande de acesso ao catálogo. Torna-se necessário então a utilização de mais de um servidor LDAP para atender a demanda dos usuários do sistema. Esta organização mestre/escravo, como é conhecida, fornece uma maneira simples e eficaz para aumentar a capacidade, disponibilidade e confiabilidade do sistema.

O *slurpd* é um programa que faz parte do LDAP e fornece a capacidade para um servidor LDAP propagar as alterações em seus dados para outros servidores LDAP, implementando o esquema mestre/escravo de replicação. É necessário que o *slurpd* seja executado no mesmo *host* que o servidor LDAP principal [UMC 1996].

Para um melhor entendimento do funcionamento do *slurpd* é apresentado a seguir um cenário contendo os passos da replicação de dados no LDAP.

Cenário contendo exemplo de replicação (adaptado de [UMC 1996]):

1. Um cliente LDAP conecta-se a um servidor LDAP secundário.
2. O cliente LDAP submete uma operação de alteração ao servidor secundário.
3. O servidor LDAP secundário retorna uma referência ao cliente LDAP fazendo com que este redirecione seu pedido ao servidor LDAP principal.

4. O servidor LDAP principal realiza a operação de alteração nos dados, altera o *log* de replicação e retorna um código ao cliente informando o sucesso da operação.
5. O processo *slurpd* observa que uma nova entrada foi adicionada ao *log* de replicação. O *slurpd* então lê a entrada no *log* de replicação e a envia aos servidores secundários.
6. O servidor secundário altera os dados e retorna um código ao *slurpd* informando o sucesso da operação.

Nota-se que, utilizando a configuração proposta, se o cliente LDAP submetesse o seu pedido ao servidor principal, o passo 3 passaria a não existir.

Um outro ponto importante a observar neste cenário é o *log* de replicação. Quando um servidor LDAP é configurado para ter seu conteúdo replicado para um ou mais servidores secundários, as alterações realizadas no diretório são gravadas em um *log* de replicação, que é um arquivo com um formato especial contendo as entradas de diretório que foram alteradas no servidor em questão. Quando o *slurpd* entra em funcionamento, o *log* de replicação é lido e as alterações são encaminhadas aos servidores secundários.

O *log* de replicação obedece o formato LDIF, que é tipicamente utilizado para importar e exportar informações de um servidor LDAP. Um arquivo no formato LDIF consiste de uma série de registros separados por linhas em branco. Um registro é formado por uma seqüência de linhas descrevendo uma entrada no LDAP ou uma seqüência de linhas descrevendo um conjunto de alterações a serem realizadas em uma entrada. Entretanto, um único arquivo LDIF não pode definir ao mesmo tempo um conjunto de entradas e um conjunto de alterações.

A estratégia de se criar um catálogo replicado e distribuído para o DIRETO parece ser bastante promissora, especialmente pelo fato dos possíveis usuários estarem geograficamente dispersos. Entretanto, durante a realização deste trabalho, inúmeras dificuldades foram encontradas. No capítulo a seguir, são relatados experimentos que foram realizados a fim de se mensurar as reais vantagens, desvantagens e implicações que a implantação de tal proposta trará ao projeto.

4 Estudo de caso: o LDAP na URCAMP

Como estudo de caso para testar a viabilidade da implementação de um catálogo distribuído para o DIRETO, optou-se pela realização de experiências com as ferramentas necessárias à criação do catálogo replicado e distribuído. Este capítulo traz um relato das experiências realizadas com as ferramentas que envolvem a possível criação de um catálogo distribuído para o DIRETO. Será abordado o ambiente onde foi instalado o servidor LDAP, incluindo configuração de *hardware* e *software*. Será explicado também como se deu o funcionamento do LDAP nos servidores e estações utilizadas neste estudo.

4.1 Ambiente utilizado

Para a realização deste estudo de caso foi utilizada a estrutura física do Laboratório de Computação 1 da URCAMP. Foram utilizados microcomputadores compatíveis com o padrão IBM/PC. Os computadores utilizados possuíam o sistema operacional *Linux* previamente instalado e com as interfaces de rede instaladas e configuradas. A distribuição do *Linux* utilizada foi o *Conectiva Linux 6.0*, visto que esta é a distribuição de maior utilização no Brasil atualmente e também devido a sua compatibilidade com o *hardware* utilizado e sua facilidade de instalação e configuração.

Em relação ao LDAP, existem atualmente inúmeras distribuições de LDAP no mercado, entretanto, foi utilizado o pacote *OpenLDAP*¹, que encontra-se disponível a partir do seu *site* oficial seguindo os conceitos da *GNU General Public License*. A versão do *OpenLDAP* utilizada foi a 2.0.9, visto que esta era a versão estável mais atual deste software no momento da realização dos testes. Atualmente o *OpenLDAP* encontra-se na versão 2.0.11.

4.2 Funcionamento do *OpenLDAP*

Após a instalação e configuração do *OpenLDAP* nos servidores e estações, foram realizadas algumas experiências a fim de se testar a funcionalidade, desempenho e facilidade de uso do *OpenLDAP*. Para a realização destas experiências foram utilizados os utilitários que compõem o pacote de *software OpenLDAP*. Com estes utilitários é possível realizar as operações básicas sobre as informações de um diretório.

Em relação aos testes realizados, o *OpenLDAP* apresentou-se bastante versátil. A inclusão dos dados no diretório pode ser realizada através da linha de comando em um terminal do *Linux* ou através da criação de um arquivo LDAP contendo as entradas de diretório a serem adicionadas.

Apesar dos testes serem realizados com um volume bastante pequeno de dados em relação a uma aplicação real, o desempenho do servidor LDAP foi bastante satisfatório. Infelizmente não foi encontrada nenhuma ferramenta que realizasse testes de performance em servidores LDAP, isto possibilitaria uma avaliação mais precisa do desempenho do sistema em uma situação de uso mais parecida com a real, onde a demanda por informações pode ser bastante grande. Um outro fato a se destacar é a estabilidade da distribuição de servidor LDAP utilizada. Durante todo o período de testes não houve nenhum momento em que o servidor se comportasse de maneira anormal ou simplesmente travasse.

A utilização do *OpenLDAP* como servidor de diretório para o catálogo do DIRETO é de extrema relevância para o sucesso deste *software*. Apesar das poucas experiências realizadas, este produto mostrou-se bastante robusto, obteve um bom desempenho e foi desenvolvido sob os mesmos conceitos de *software* livre utilizados para o desenvolvimento do DIRETO. Isto permite que possíveis ajustes necessários à otimização do catálogo do DIRETO possam ser realizados de maneira bastante eficaz, visto que este é um *software* aberto e possui toda a documentação necessária para sua utilização disponível em seu *site* oficial.

5 Considerações Finais

Este trabalho apresentou os principais conceitos que envolvem a implementação de um catálogo distribuído para o software de comunicação DIRETO. Foram explicadas as necessidades que motivaram a criação do DIRETO bem como as demandas que este software procura atender. Uma breve explicação sobre as ferramentas utilizadas na implementação e no ambiente de execução do DIRETO foi apresentada.

O estudo do serviço de diretório LDAP foi o ponto central deste trabalho. Entende-se que para se obter sucesso na implementação de um catálogo distribuído no DIRETO é de fundamental importância o domínio desta tecnologia. Durante a realização deste trabalho muito tempo foi dedicado ao estudo desta ferramenta, despertando assim o interesse em um domínio de conhecimento até então pouco conhecido pelo autor. O estudo do serviço de diretório LDAP compreendeu a análise do

¹ <http://www.openldap.org>

funcionamento desta tecnologia, estudo da arquitetura de um servidor LDAP, formas de funcionamento de um servidor LDAP e uma atenção especial foi dada aos aspectos que envolvem a replicação de dados entre servidores LDAP.

Com base no estudo realizado, foram apresentadas alternativas para implantação de servidores LDAP com o conteúdo replicado e distribuído no ambiente de execução do DIRETO.

Com o objetivo de tornar este trabalho um pouco mais prático, optou-se por apresentar como estudo de caso os passos para instalação e configuração do LDAP juntamente com um relato dos testes realizados com esta ferramenta.

Atualmente o LDAP vem consolidando-se como um padrão capaz de atender a demanda por um protocolo aberto de acesso à informações armazenadas em um serviço de diretório. Isto faz com que muitas empresas importantes na área da computação implementem suporte ao LDAP em seus produtos.

Com uma vasta quantidade de empresas adicionando suporte a LDAP em seus produtos, o LDAP tende a tornar-se o elo de ligação entre estes diversos produtos. Proporcionando uma maneira eficiente de obtenção de informações entre plataformas diferentes e muitas vezes incompatíveis.

Referências Bibliográficas

- [CAR 1998] CARUSO, B., MOHAN, R., **An LDAP engine for TSpaces**. 1998. Disponível em: <<http://www.cse.scu.edu/projects/1998-99/project4/prelim.pdf>>. Acesso em: 15 jun. 2001.
- [GRE 2001] GREATBRIDGE, **Great Bridge – The Open Source Database Experts**. 2001. Disponível em: <<http://www.greatbridge.com/postgresql/features.php>>. Acesso em: 19 jun. 2001.
- [HEU 2000] HEUSER, Carlos A. **Projeto de Banco de Dados**. 3. ed. Porto Alegre: Sagra Luzzato, 2000. 204p.
- [JAV 2001] SUN Microsystems, **What is the Java™ Plataforma?**. 2001. Disponível em: <<http://java.sun.com/nav/whatis/?frontpage-shortcuts>>. Acesso em 15 jun. 2001.
- [JOH 1998] JOHNER, H. et. al., **Understanding LDAP**. 1998. Disponível em: <<http://www.redbooks.ibm.com/pubs/pdfs/redbooks/sg244986.pdf>>. Acesso em: 24 jan. 2001.
- [POS 2001] POSTGRESQL, **PostgreSQL**. 2001. Disponível em: <<http://pgsql.dbexperts.com.br/>>. Acesso em 19 jun. 2001.
- [SUN 2001] SUN Microsystems, **The Source for Java technology**. 2001. Disponível em: <<http://java.sun.com/>>. Acesso em: 12 mar. 2001.
- [UMC 1996] UNIVERSITY OF MICHIGAN, **The SLAPD and SLURPD Administrator's Guide**. 1996. Disponível em: <<http://www.umich.edu/~dirsvcs/ldap/doc/guides/slaped/toc.html>>. Acesso em: 18 mar. 2001.
- [WEB 2001] WEBOPEDIA, **X.500 - WebOpedia Definition and Links**. 2001. Disponível em: <http://www.webopedia.com/TERM/X/X_500.html>. Acesso em: 20 mar. 2001.