

CENTRO UNIVERSITÁRIO DE BRASÍLIA - UniCEUB

Curso de Graduação em Ciências da Computação

Alunos:

Enzo Alves Miranda Leite

RA: 21909300

Mateus de Brito Bacelar

RA: 21905387

RESUMO

FERRAMENTA DE BACKUP TRANSPARENTE BASEADA EM P2P

Brasília

2020

RESUMO

O artigo ‘Ferramenta de Backup Transparente Baseada em P2P’ mostra um protótipo de uma ferramenta de backup e seus módulos e funcionamento , que seja transparente ao usuário e faz proveito da conectividade presente entre os mesmos, e que surgiu devido ao avanço da tecnologia e a necessidade de uma rotina de backup de arquivos devido a sua quantidade crescente.

Palavras-Chave: sistemas distribuídos; Peer-to-Peer; backup; JXTA; XML; middleware

Abstract: The article ‘Ferramenta de Backup Transparente Baseada em P2P’ show a prototype of a backup tool and its modules and functioning , let it be transparent for the user and make advantage gives connectivity present among them, and that appeared due to advance of technology and the need for a file backup routine due to its increasing amount.

Keywords: distributed systems; Peer-to-Peer; backup; JXTA; XML; middleware.

1. INTRODUÇÃO

Devido ao avanço tecnológico, a crescente quantidade de arquivos multimídia e em formato digital, a necessidade de uma cópia de segurança às vezes se torna vital, entretanto diversos usuários negligenciam a rotina de *backup* de arquivos, mesmo sabendo que computadores e aparelhos eletrônicos podem apresentar defeitos por diversos motivos, que podem causar a perda desses arquivos.

Além disso, tendo em mente que as redes locais estão cada vez mais comuns, tendo mais de um dispositivo conectado na mesma rede, surgiu a ideia do desenvolvimento de uma ferramenta de backup multiplataforma baseada na tecnologia Peer-to-Peer, a qual é apresentada no artigo em que este resumo se baseia, abordando seus componentes.

2. SISTEMAS DISTRIBUÍDOS

computadores interligados em rede se comunicam e coordenam suas ações através de mensagens. Portanto, a construção desse tipo de sistema se deve principalmente pela necessidade de compartilhamento de recursos que possam ser utilizado por servidores ou por outros clientes.

Sistemas distribuídos funcionam em computadores conectados em rede, independentemente da localização física dos seus membros.

2.1. COMPARTILHAMENTO DE RECURSO

Compartilhamento é o simples compartilhar dados, seja fisicamente como uma impressora, ou digitalmente como dados do trabalho. O acesso a arquivos compartilhados é feito através de serviços de sistemas de arquivos. Nos sistemas distribuídos os recursos são fisicamente encapsulados provendo acesso através de um mecanismo de comunicação.

Durante o desenvolvimento de sistemas distribuídos um dos desafios encontrados é a heterogeneidade de computadores, redes, sistemas operacionais, linguagens de programação e formas de representação de dados.

Portanto deve ser levado em consideração os tipos de dados representados, uma vez que esta ferramenta irá ser operada em modelo multiplataforma.

3. SISTEMA PEER-TO-PEER

P2P é um tipo de sistema que se refere a uma rede de dispositivos(nós) conectados usando sistemas de comunicação, possibilitando a colaboração uns com os outros sem existir uma coordenação central. A topologia de rede utilizada é a topologia descentralizada que mais se aproxima de ser verdadeiramente P2P, onde não há uma autoridade central, apenas dispositivos individuais capazes de se conectar e comunicar com quaisquer outros dispositivos conectados na rede.

4. MIDDLEWARE

É preciso dividir a pilha de software, versão única do sistema, em duas partes para que o sistema distribuído suporte uma variedade de computadores e redes. A parte mais alta é a aplicação e a parte inferior é o middleware, responsável por interagir com as camadas mais baixas de redes e computadores e prover a transparência necessária, e para isso o framework escolhido foi o JXTA.

protocolos do desenvolvedor de aplicações, provendo uma variedade de facilidades para os programadores e liberando-os da necessidade de conhecimentos específicos sobre cada plataforma onde o sistema será executado.

5. JXTA

O *middleware* JXTA é um conjunto de protocolos abertos e generalizados que possibilitam quaisquer dispositivos conectados na rede, a comunicação e a colaboração. Podemos dizer que o JXTA é um framework com um conjunto de padrões que suportam aplicações P2P, sem especificar uma linguagem específica para implementação. Os seis protocolos que trabalham juntos no JXTA e que possibilitam a interação na rede e a existência de um nó em uma rede P2P descentralizada são:

- Protocolo de consulta de Nó
 - Consulta a qualquer número de nós e recebimento de respostas
- Protocolo de descoberta de Nó
 - Propagar e descobrir conteúdo
- Protocolo de Informação
 - Obter informações do status do nó
- Protocolo de Ligação
 - Comunicação entre nós
- Protocolo de Roteamento
 - Encontrar rota entre nós
- Protocolo Rendezvous
 - Propagar mensagens na rede

5.1. ANÚNCIO

O anúncio pode ser descrito como uma ferramenta principal usada pelos protocolos JXTA que permite a criação de um nó, grupos de nós, e de informações de configurações disponíveis na rede. É um arquivo que pode ser compartilhado entre nós através de um formato comum, o XML, que apresenta uma representação hierárquica, ou seja facilmente expansível de informações necessárias para os nós operarem em uma rede JXTA.

5.2. ARQUITETURA JXTA

A arquitetura JXTA é dividida em três camadas, o núcleo, camada mais baixa da arquitetura e responsável por estabelecer a comunicação, conexão e gerenciamento. Onde um nó recém criado entra num grupo de protocolos que fornece todas as

fornecer funcionalidades de mais alto nível. Onde se faz o uso pesado de recursos de ligação fornecidos pelo núcleo, podendo ser utilizados como componentes em sistemas P2P. E por último temos a camada de aplicação, onde o desenvolvedor cria os sistemas P2P. Esta camada é responsável por auxiliar o desenvolvimento de diversos tipos de aplicações. Por fim podemos dizer que a linha divisória entre as camadas não é rígida, portanto se um nó fornece uma funcionalidade outro nó deveria enxergar ela como um serviço que pode ser utilizado pelo outro, como uma aplicação completa.

5.3. SERVIÇO DE GERENCIAMENTO DE CONTEÚDO

O Serviço Gerenciador de Conteúdo (CMS), é responsável por habilitar para um nó o compartilhamento de arquivos. O CMS utiliza anúncios para fornecer informações relacionadas ao conteúdo que será compartilhado pelo nó, e usa JXTA pipes para a transferência do mesmo. Os pipes são utilizados para receber requisições, consultas e conteúdo de outros nós.

6. FERRAMENTA DE BACKUP

A ferramenta desenvolvida é dividida em dois módulos, um de configuração, e outro de backup, que fará uso das configurações geradas pelo primeiro.

6.1. MÓDULO DE CONFIGURAÇÃO

Este módulo gera o arquivo XML que é usado pelo módulo backup, este arquivo conterá informações relevantes para a execução do módulo de backup. O arquivo XML apresenta os seguintes atributos:

- <nomepeer>: Nome que identifica o nó local, utiliza-se como o intuito de indicar aos nós remotos o dono dos arquivos publicados na rede.
- <pastacompartilhada>: É o caminho que da pasta do local onde estão os arquivos para backup. O módulo de backup sempre procurará os arquivos a serem publicados nesta pasta.
- <repositorio>: Indica o local, irão guardados os arquivos recebidos de nós remotos na rede.
- <pastasistema>: Indica onde serão armazenados os arquivos com informações de conexão da ferramenta.
- <pastapeer>: Indica onde serão armazenados os arquivos com informações relacionadas ao nó local.

seja compatível com a estrutura do sistema de arquivos do sistema operacional onde ele será executado. Ou seja, o padrão de nomenclatura do sistema de arquivos não fica vinculado ao programa.

6.2. MÓDULO DE BACKUP

O módulo de backup é iniciado com a leitura do arquivo XML criado no módulo de configuração. Primeiro ele verifica se há arquivos anteriores nas pastas definidas em <pastasistema> e <pastapeer>, os apaga e recria arquivos de informação de conexão e de informações locais.

O servidor presente em cada computador junto com o nó local recebe tanto conexões do nó local quanto de nós remotos. Após o estabelecimento da conexão do nó com o servidor local, é iniciada a busca de arquivos na pasta definida para publicação (<pastacompartilhada>), que serão publicados pelo CMS do JXTA.

Cada publicação de arquivos realizada é feita pelo CMS, onde é armazenado informações sobre os arquivos publicados. O controle da integridade do arquivo é feito usando uma assinatura hash, o algoritmo MD5. E para as origens do arquivo é utilizado o nome definido em <nomepeer>.

É definido um tempo limite de 1 hora pra a republicação dos arquivos na pasta de backup. Após serem publicados os arquivos locais é realizado uma requisição de arquivos dos nós remotos, sendo realizado a análise dos anúncios para verificar a existência de tais arquivos. Então é realizado uma verificação de existência de arquivos remotos na rede para certificar que eles existem no <repositorio> local, e por fim é utilizado a assinatura MD5 para verificar a versão do arquivo, podendo realizar ou não uma nova requisição.

O efetivo controle da transmissão do arquivo das requisições de download é realizado usando threads. Cada thread é gerado para cada arquivo, respeitando o numero máximo de threads permitido, neste caso cinco, quaisquer requisições posteriores são alocadas em uma fila.

O controle da integridade do arquivo é feito utilizando-se uma assinatura hash. O método escolhido é o algoritmo MD5. Já para as informações pertinentes à origem do arquivo, é utilizado o nome do nó local, definido em <nomepeer>. Este nome é utilizado para que os nós remotos identifiquem o dono do arquivo.

Ao longo do artigo resumido foi concluído que a ferramenta JXTA é um framework com uma quantidade de recursos grande para o desenvolvimento de sistemas P2P. Entretanto uma das maiores dificuldades foi a falta de material atual sobre o JXTA, e se chegou a uma previsão de que em trabalhos futuros poderia ser implementado um módulo de recuperação de arquivos em outro nó, além de uma camada de segurança.

8. REFERÊNCIA

Google Sala de Aula. Google.com. Disponível em:

<<https://classroom.google.com/u/1/c/Mzc1MDk5NDgzNzg5/a/Mzc3MTYzMTQ5MjUz/details>>. Acesso em: 18 Aug. 2021.