Sistemas Computacionais Distribuídos

Prof. Marcos José Santana

SSC-ICMC-USP

São Carlos, 2008

Grupo de Sistemas Distribuídos e Programação Concorrente

Departamento de Sistemas de Computação - SSC

Sistemas Computacionais Distribuídos

Servidores de Arquivos

Conteúdo

- Segurança
- Confiabilidade
- Consistência de dados
- Implementação de transações atômicas

Segurança

- **Controle de acesso**
  - ***Capability-based***
  - ***Identity-based***
  - ***Control list***
  - **Criptografia**

Segurança [*Capability-based*]

- O cliente deve apresentar "***capability***" para poder ter acesso ao arquivo
- Deve haver um bom mecanismo para gerar as "***capabilities***"
- Maioria dos sistemas usa o FID como uma *capability*

Segurança
[*Identity-based + Control list*]

- Neste caso os clientes apresentam algum tipo de prova de sua identidade, junto com o FID
- Existe uma lista de controle de acesso que fornece os direitos de acesso de cada cliente para o arquivo

Segurança [Criptografia]

- Nos sistemas baseados em criptografia, cada arquivo tem sua "chave" e o arquivo é armazenado e recuperado em sua forma codificada
- A chave funciona como uma "*capability*", mas o servidor não precisa validar

Confiabilidade

- Importância
- Servidor compartilhado
- Inclusão de mecanismo para garantir:
  - Um dado grau de confiança
  - Consistência dos dados
  - Recuperação dos dados
  - Robustez
    - Problemas de comunicação
    - Problemas de *crash* do servidor
    - Problemas de *crash* do cliente
    - Problemas nos discos

Confiabilidade

- Problema da concorrência
  - Ambiente multi-usuário
  - Controle de acesso aos dados compartilhados
- Mecanismos adotados
  - Maioria vem de propostas implementadas em outros tipos de sistemas
  - Banco de dados!

Consistência de dados

- Problema tradicional
- Transações atômicas
  - Dados são alterados “automaticamente”
  - Um dado tem um novo estado consistente ou é mantido no seu estado inicial, inalterado
  - Ou tudo, ou nada
- Transações NÃO atômicas
  - Permitem que um estado intermediário, não consistente, seja apresentado ao cliente.

Consistência de dados

Exemplo (tradicional – banco de dados)

- **Contas bancárias A e B com saldos a e b. Transação T transfere x de A para B.**

  begin-trans
  1. read A to get a
  2. read B to get b
  3. write a-x to A
  4. write b+x to B
  end-trans

Consistência de dados

- T é atômica
  - Ou os saldos são: a e b
  - Ou a-x e b+x
- T NÃO é atômica
  - Podem reter: a-x; b
  - Estado inconsistente

Implementação de Transações Atômicas

- Requer mecanismos especiais
- Lampon => Modelo hierárquico para máquinas mais confiáveis
  => Transações atômicas
- Requisitos
  - Operações atômicas (Cj)
  - Recuperação de dados (mecanismo)
  - Controle de concorrência

Implementação de Transações Atômicas

- Mecanismos para a recuperação de dados e para controle de concorrência
  - "*reliable storage*"
- Reliable storage => algum tipo de memória não volátil
- Mecanismos para garantir "recuperação" em uma transação
  - *Shadow-page*
  - *Undo-redo log*
  - *Intentions log*
  - *Tentative versions*

Fim!