

ATOMICIDADE

SISTEMAS DISTRIBUÍDOS  
E  
TOLERÂNCIA A FALHAS

Carlos Santos Pinto, m1392

# ATOMICIDADE

## SUMÁRIO

- Atomicidade
- Transacções atómicas
- Sistemas Distribuídos
- Sistemas Concorrentes
- Sistemas Tolerantes a Falhas

# ATOMICIDADE

## Operação atômica

é indivisível, não tendo etapas intermédias visíveis pois todas aparecem como sendo uma só (é chamada sequência atômica).

## Transacções atômicas

são paradigmas que permitem que sequências arbitrárias de operações de dados sejam transformadas em operações atômicas.

# ATOMICIDADE

Tem 2 directivas

começar transacção

acabar transacção

Tem 2 resultados

terminar com sucesso

não terminar

# ATOMICIDADE

Tem 2 componentes

mecanismo de recuperação: quando a transacção aborta permite recuperar o estado anterior do sistema

invisibilidade intermédia: garante que os resultados intermédios não se tornam visíveis a não ser que a transacção seja bem sucedida

# ATOMICIDADE

## Armazenagem duradoura

para que a recuperação seja possível é necessário um dispositivo (físico ou lógico) de armazenagem onde os dados possam ser guardados de uma forma segura sem poderem ser perdidos

## *Log*

estrutura de dados onde são guardados os valores dos dados a actualizar e actualizados

# ATOMICIDADE

## Atenção a falhas

o resultado da actualização dos dados é guardado no *log*, assim como uma entrada certificando que a transacção teve sucesso. Só assim, a transacção pode ser confirmada ao utilizador

adicionalmente, uma transacção não pode ter sucesso até que todas as actualizações tenham sido gravadas

# ATOMICIDADE

## Resultados possíveis

se o nodo pendura o log passa ao estado de recuperação da situação inicial

se a transacção aborta o log passa ao estado de recuperação da situação inicial

se a transacção tem sucesso a actualização dos dados é deixada inalterável.



# ATOMICIDADE

## Sistemas Distribuídos

num sistema distribuído, uma transacção pode aceder aos dados em diferentes nodos

a atomicidade manifesta-se através da concordância de todos os nodos envolvidos acerca do sucesso ou aborto de uma transacção. Esta concordância é assegurada pelo *Distributed Atomic Commitment Protocol*

# ATOMICIDADE

Na ausência de falhas

1º fase

um nodo é designado “coordenador” e envia a todos os outros uma mensagem *prepare*

recebe *notok* no caso de algum detectar algum erro que impeça o sucesso e *ok* se estão prontos

# ATOMICIDADE

## 2ª Fase

quando recebe ok de todos, o coordenador responde *commit* a todos os nodos

se receber pelo menos um *notok* responde abort a todos os nodos

em qualquer caso, o coordenador espera um *ack* para assegurar uma rápida disseminação do resultado da transacção, retransmitindo a decisão mesmo se alguns *ack's* estiverem perdidos.

# ATOMICIDADE

Devido a esta estrutura em duas fases, o protocolo que assegura este tipo de transacções é conhecido por *Two-Phase Atomic Commitment Protocol*

tem como desvantagem o facto de poder bloquear se o coordenador falhar em alguns passos do protocolo, porque os restantes processos podem ser incapazes de aceder à decisão certa (*commit / abort*), se esta tiver sido emitida.

# ATOMICIDADE

## Sistemas Concorrentes

garante, juntamente com os restantes componentes da sigla ACID ( atomicidade, **consistência, isolamento e durabilidade**) o correcto funcionamento e a integridade dos processos, por exemplo, em bases de dados

# ATOMICIDADE

## Sistemas Tolerantes a Falhas

fornece mecanismos que permitem assegurar que eventuais execuções de um programa levam não apenas a resultados confiáveis, mas também garantem a consistência dos dados mesmo sob condições de falhas. Dados manipulados pelo sistema só podem ser levados de um estado consistente a outro.