

Estabeleça as relações sobre as propriedades das transacções:

Se diversas transacções se executarem em paralelo sobre os mesmos objectos, tudo se passa como se as transacções se executassem em série numa determinada ordem

Para um observador externo, uma transacção ou se executa na totalidade ou não se executa.

Cada transacção deve, a partir de um estado inicial válido e caso se execute completamente, atingir um novo estado válido

Os resultados de uma transacção que confirmou permanecem depois de esta acabar e são supostos sobreviver ao conjunto de faltas expectáveis dos mecanismos de armazenamento

Estabeleça as relações sobre as

Atomicidade

Seriabilidade

Atomicidade

Consistência

Persistência

Considere a seguinte matriz, onde nas colunas estão representados Objectos, nas linhas estão representados Agentes e cada célula representa os direitos de acesso:

| Agentes / Objectos | O1 | O2 | O3 | O4 |
|--------------------|----|-----|----|-----|
| A1 | R | RW | RX | --- |
| A2 | RX | --- | RW | R |

Qual é a Access Control List (ACL) do Obj 2 ?

A1: RW; A2: ---

As Access Control List (ACL) correspondem a uma coluna ou a uma linha desta matriz ?

Uma coluna

Esta matriz é usada para que tipo de Controlo de Acesso ?

Discretionary Access Control (DAC)

Lightshot

| | |
|-------------|-----------------------------------|
| Iniciado em | Segunda, 30 Janeiro 2017, 18:30 |
| Estado | Terminada |
| Completo em | Segunda, 30 Janeiro 2017, 19:30 |
| Tempo gasto | 1 hora |
| Nota | 171,11 num máximo de 200,00 (86%) |

Pergunta 1

Parcialmente correto

Nota: 6,67 em 10,00

Marcar pergunta

No início do protocolo Two-Phase Commit (1ª fase), estabeleça as relações entre as seguintes acções:

No início do protocolo

O Coordenador envia a todos os RM participantes a mensagem Prepare_commit

O Coordenador fica bloqueado e não pode continuar

O Coordenador inicia o protocolo

O Coordenador obtém a decisão de todos os RM participantes e escreve um registo no diário com as respostas recebidas

Neste momento, o Coordenador toma a decisão sobre o desfecho da transacção.

Um participante regista a indicação de Prepare_commit, verifica se pode confirmar a transacção e responde ao Coordenador com "Commit" ou "Abort"

Ação feita por cada participante quando recebe a mensagem Prepare_commit do Coordenador

Pergunta 2

Parcialmente correto

Nota: 7,50 em 10,00

Marcar pergunta

Estabeleça a correspondência entre as seguintes afirmações:

Estabeleça a correspond

Identificação do agente que produziu a informação

Autenticidade

Impossibilidade de um agente negar uma acção que efectivamente cometeu

Não repúdio

A informação está correcta e completa

Integridade

Acesso à informação apenas por agentes autorizados

Segurança

Privacidade

O coordenador

O Coordenador ou um Participante podem falhar durante a execução do protocolo Two-Phase Commit. Isso impacta o funcionamento dos outros nós e quando o nó que falhou recomçar a funcionar, tem que recuperar o ponto onde estava no protocolo e continuar com o mesmo. Estabeleça as relações entre as afirmações seguintes:

| | | |
|---|--|---|
| O Coordenador obtém um timeout quando espera a resposta à mensagem "Prepare_Commit" | O Coordenador decide fazer Abort da transação | ✓ |
| Um Participante falha e quando recupera verifica que estava no estado "Confirmar" | O Participante volta a ficar à espera da mensagem do Coordenador com a decisão sobre a transação | ✓ |
| O Coordenador falha e quando recupera verifica que estava no estado "Confirmar" | O Coordenador reenvia a mensagem "Confirm_Commit" aos participantes | ✓ |
| Um participante tem um timeout quando espera a mensagem "Prepare_Commit" do Coordenador | O Participante volta a ficar à espera da mensagem do Coordenador com a decisão sobre a transação | ✓ |
| | O participante decide fazer Abort da transação | ✗ |

Para que os vários componentes de um Sistema Distribuído possam colaborar, necessitam de aceder a serviços remotos. Indique por ordem crescente de nível de abstracção as formas de acesso e estabeleça a correspondência com as tecnologias respectivas:

| | | | |
|--|---|--------------|---|
| 1. acesso directo ao protocolo de transporte | → | sockets | ✓ |
| 2. invocação programática de funções remotas | → | RPC | ✓ |
| 3. invocação de métodos de objectos remotos | → | RMI | ✓ |
| 4. invocação das operações de um serviço | → | Web Services | ✓ |

Para que vários

RM

Considere 2 agentes A e B em que:

KApriv - Chave privada de A, só conhecida por A

KApub - Chave pública de A, conhecida por toda a gente

KBpriv - Chave privada de B, só conhecida por B

KBpub - Chave pública de B, conhecida por toda a gente

(M)K - Mensagem M cifrada com a chave K

Estabeleça as correspondências correctas quando A envia uma mensagem a B:

| | | | |
|------------------------|----------------------------|---|--------------|
| A → B (M)KBpub | Autenticação de B | ✗ | Privacidade |
| A → B (M)KApriv | Privacidade | ✗ | Autent. De A |
| A → B ((M)KApriv)KBpub | Autenticação e Privacidade | ✓ | |

Considere 2

Estabeleça as relações corretas entre as seguintes afirmações sobre as arquiteturas de sistemas distribuídos:

Estabeleça as relações

| | | |
|--|--|---|
| Os clientes comunicam com outros clientes | Arquitetura peer-to-peer | ✓ |
| O cliente comunica com um servidor que responde a parte do serviço e indica ao cliente outro servidor que este deve contactar para continuar o serviço | Arquitetura de 2 níveis | ✗ |
| O cliente comunica com um servidor que mantém uma cópia local das respostas recentes e lhe devolve imediatamente a resposta se possível | Arquitetura cliente-servidor com proxy | ✓ |
| O cliente comunica com um servidor, que por sua vez comunica com outros servidores para satisfazer o pedido e responde ao cliente | Arquitetura cliente-servidor com servidores recursivos | ✓ |

Arquitetura cliente-servidor com servidores iterativos

A semântica de uma invocação remota indica a forma como são executados os vários passos que a compõem e quais as possíveis acções e consequências em caso de falha. Estabeleça a relação entre os seguintes tipos de invocação e as suas características.

A semântica

| | | |
|--|-------|---|
| A funcionalidade remota pode ou não ser invocada | Maybe | Semântica tipo "Maybe" |
| Se a invocação for de tipo "At-most-once" ou o runtime do RPC detectar duplicações | | As funções remotas não têm que ser idempotentes |
| A funcionalidade remota é invocada, mas não se sabe exactamente quantas vezes | | Semântica tipo "At-least-once" |
| Se a invocação for de tipo "At-least-once" e o runtime do RPC não detectar duplicações | | As funções remotas devem ser idempotentes |
| A funcionalidade remota é invocada no máximo uma vez | | Semântica tipo "At-most-once" |

Os dados que circulam nas mensagens de invocação de serviços remotos são convertidos para um formato genérico antes de ser enviados. Indique para cada um dos paradigmas de invocação remota qual a técnica de conversão de dados utilizada.

Os dados

| | |
|---------------------------------|------------------------------------|
| Remote Procedure Call (SUN RPC) | External Data Representation (XDR) |
| Web Services | Extended Markup Language (XML) |
| Sockets | Network Byte Order |
| Remote Method Invocation (RMI) | Object Serialization |

XDR

Lightshot
Your screenshot is copied to clipboard

Estabeleça as relações correctas entre as seguintes características do Sun RPC e Java RMI:

Estabeleça as relações

| | |
|---|--------------------|
| Permite gerar automaticamente os stubs a partir da definição da interface | Sun RPC e Java RMI |
| Permite a construção de aplicações cliente - servidor | Sun RPC e Java RMI |
| Suporta a instanciação remota de objectos | Java RMI |
| Utilizado na implementação do NFS | Sun RPC |
| Permite criar um cliente em C que comunica com um servidor Java | Afirmção falsa |

Cliente em C

A arquitectura define a estrutura de um sistema distribuído em termos de componentes especificáveis separadamente. O modelo de arquitectura:

A arquitectura

Selecione uma ou mais opções de resposta:

- ☒ a. Define a funcionalidade e localização dos componentes do sistema ✓
- ☐ b. Descreve as classes ou rotinas do servidor
- ☒ c. Identifica as relações e os padrões de comunicação existentes entre os componentes ✓
- ☐ d. Não é necessário se o código estiver bem documentado

Indique as diferenças e semelhanças entre os Sistemas de Gestão de Ficheiros Distribuídos NFS e AFS seleccionando para cada característica o(s) sistema(s) que as implementa(m).

| | | |
|--|-----------------|---|
| Implementa sincronização automática em acessos concorrentes ao mesmo ficheiro | Nem NFS nem AFS | ✓ |
| A implementação obedece ao modelo cliente/servidor | NFS e AFS | ✓ |
| A cache no cliente armazena blocos de disco e é mantida em memória | NFS | ✓ |
| O servidor guarda informação sobre os clientes que têm ficheiros armazenados nas suas caches | AFS | ✓ |
| O servidor não mantém estado sobre os clientes | NFS | ✓ |
| O espaço de nomes é global e igual para todos os clientes | NFS e AFS | ✗ |
| Cada cliente constrói o seu espaço de nomeação através de mount de partições remotas | NFS | ✓ |
| A gestão da coerência da cache dos clientes é baseada em callbacks | AFS | ✓ |
| A cache no cliente armazena ficheiros e é mantida em disco | NFS | ✗ |

Indique as diferenças

AFS

Considere as 4 mensagens do protocolo Needham - Schroeder entre A, B e a autoridade de certificação S:

1. $A \rightarrow S: A, B, Na$
2. $S \rightarrow A: \{Na, B, Kab, (Ts, A, Kab)Kb\}Ka$
3. $A \rightarrow B: \{A, Ta\}Kab, \{Kab, A, Ts\}Kb$
4. $B \rightarrow A: (Ta + 1)Kab$

Considere as 4

Para A enviar a mensagem 3 a B a componente $\{A, Ta\}Kab$, A teve que

Quando B recebe a mensagem 3, acredita que a componente $\{Kab, A, Ts\}Kb$ foi gerada pela autoridade de certificação S porque

Na mensagem 2, A acredita que a mensagem vem da autoridade de certificação S porque:

- Obter a chave Kab da mensagem 2
- Está cifrada com a chave de B
- Está cifrada com a chave de A

Estabeleça as correspondências correctas dos componentes do WSDL:

| | | |
|--|----------|---|
| Descreve a interface do Web Service | portType | ✓ |
| Define o protocolo e com ele o formato das mensagens, de uma forma dependente do protocolo | Binding | ✓ |
| Especifica o endereço do WebService, normalmente um URL | Port | ✓ |
| Definição abstracta dos tipos definidos pelo utilizador | Type | ✓ |
| Descreve o formato abstracto da mensagem | Message | ✓ |

Estabeleça as correspondências correctas

Port

Estabeleça a relação entre os componentes do RPC e a sua funcionalidade:

Pedido: retira os parâmetros da mensagem e coloca-os na pilha para invocar o procedimento do serviço
 Resposta: retira o resultados da pilha e passa-os para um formato independente na mensagem

Pedido: retira os parâmetros da pilha passa-os para um formato independente na mensagem
 Resposta: passa o resultado da mensagem para a pilha de invocação

Assegura o encaminhamento das mensagens e o dispatching destas para o stubs

| | | |
|-------------|---|-------------|
| SOAP | ✗ | Server stub |
| Server stub | ✗ | Client stub |
| RPC runtime | ✓ | |

Considere a seguinte interface SUN RPC:

```
struct calcargs {
    float a;
    float b;
};

program REMOTECALC {
    version CALCVERS {
        float ADD(calcargs) = 2;
        float SUB(calcargs) = 3;
    } = 1;
} = 0x20000100;
```

Considere a seguinte

SUN RPC

Quantos parâmetros de entrada tem cada função ?

1 parâmetro de entrada

Quantas funções remotas são definidas

2 funções remotas

Que utilitário deve ser usado para, com base neste ficheiro, gerar os stubs cliente e servidor ?

rpcgen

2019-----

| | |
|-------------|-----------------------------------|
| Iniciado em | Quinta, 31 Janeiro 2019, 10:33 |
| Estado | Terminada |
| Completo em | Quinta, 31 Janeiro 2019, 20:06 |
| Tempo gasto | 1 hora 32 minutos |
| Nota | 152,33/196,00 |
| Nota | 155,44 num máximo de 200,00 (78%) |

Pergunta 1
 Respondida
 Nota: 10,00 em 10,00
 Marcar pergunta

Estabeleça a correspondência entre as definições e os requisitos dos Sistemas Distribuídos apresentados.

| | |
|---|-----------------|
| A capacidade de permitir a coerência de vários fluxos de execução realizando acessos simultâneos a recursos partilhados | Concorrência |
| A capacidade para suportar a sua extensão, modificação e interacção com outros sistemas | Abertura |
| A capacidade em responder de forma linear a aumentos significativos da carga, número de utilizadores ou área de abrangência | Escalabilidade |
| A capacidade para acomodar a variedade e a diferença | Heterogeneidade |
| A capacidade de esconder do utilizador e das aplicações a natureza distribuída do sistema | Transparência |

Pergunta 2
 Respondida
 Nota: 10,00 em 10,00
 Marcar pergunta

O acesso a serviços remotos pode ser feito utilizando várias tecnologias que utilizam níveis de abstracção relativamente aos protocolos de transporte tradicionais. Estabeleça as correspondências entre os níveis seguintes e as tecnologias que utilizam.

| | |
|--|--------------|
| Acesso directo ao protocolo de transporte | Sockets |
| Invocação de serviços remotos de forma independente de qualquer plataforma | Web Services |
| Invocação programática de funções remotas | Sun RPC |
| Invocação de métodos de objectos remotos | Java RMI |

Pergunta 3

Respondida

Nota: 10.00 em 10.00

⚑ Marcar pergunta

Para que os vários componentes de um Sistema Distribuído possam colaborar, necessitam de aceder a serviços remotos. Indique as principais formas de acesso que existem, ordenando-as por níveis de abstracção crescente e fazendo corresponder as tecnologias respectivas.

- | | | |
|--|---|--------------|
| 1. acesso directo ao protocolo de transporte | → | sockets |
| 2. invocação programática de funções remotas | → | RPC |
| 3. invocação de métodos de objectos remotos | → | RMI |
| 4. invocação das operações de um serviço | → | Web Services |

Pergunta 4

Respondida

Nota: 10.00 em 10.00

⚑ Marcar pergunta

Estabeleça as relações corretas entre as seguintes afirmações sobre as arquiteturas de sistemas distribuídos:

O cliente comunica com um servidor que responde a parte do serviço e indica ao cliente outro servidor que este deve contactar para continuar o serviço

Arquitetura cliente-servidor com servidores iterativos

O cliente comunica com um servidor, que por sua vez comunica com outros servidores para satisfazer o pedido e responde ao cliente

Arquitetura cliente-servidor com servidores recursivos

O cliente comunica com um servidor que mantém uma cópia local das respostas recentes e lhe devolve imediatamente a resposta se ela existir na cópia local

Arquitetura cliente-servidor com proxy

Os clientes comunicam com outros clientes

Arquitetura peer-to-peer

Pergunta 5

Respondida

Nota: 7.50 em 10.00

⚑ Marcar pergunta

Os dados que circulam nas mensagens de invocação de serviços remotos são convertidos para um formato genérico antes de ser enviados. Indique para cada um dos paradigmas de invocação remota qual a técnica de conversão de dados utilizada.

- | | |
|---------------------------------|---|
| Sockets | Web Service Description Language (WSDL) |
| Web Services | Extended Markup Language (XML) |
| Remote Procedure Call (SUN RPC) | External Data Representation (XDR) |
| Remote Method Invocation (RMI) | Object Serialization |

Pergunta 6

Respondida

Nota: 0.00 em 10.00

⚑ Marcar pergunta

Indique quais das características abaixo indicadas correspondem a vantagens de uma arquitetura cliente - servidor.

Selecione uma ou mais opções de resposta:

- ☒ a. Os dados sensíveis estão guardados no servidor
- ☐ b. O código dos clientes e dos servidores tem que ser escrito pela mesma organização
- ☐ c. Pode-se aumentar a disponibilidade do sistema com a replicação de servidores
- ☐ d. A complexidade da implementação do serviço está no servidor, o cliente é mais simples
- ☒ e. Os clientes comunicam diretamente com outros clientes

simples

Pergunta 7

Respondida

Nota: 10.00 em 10.00

⚑ Marcar pergunta

Estabeleça a relação entre os componentes do RPC e a sua funcionalidade:

Assegura o encaminhamento das mensagens e o dispatching destas para o stubs

RPC runtime

Pedido: retira os parâmetros da pilha passa-os para um formato independente na mensagem
Resposta: passa o resultado da mensagem para a pilha de invocação

Client stub

Pedido: retira os parâmetros da mensagem e coloca-os na pilha para invocar o procedimento do serviço
Resposta: retira o resultados da pilha e passa-os para um formato independente na mensagem

Server stub

Pergunta 12

Respondida

Nota: 10.00 em 10.00

🚩 Marcar pergunta

Estabeleça as relações correctas entre as seguintes afirmações relacionadas com o serviço e os servidores de nomes:

As associações nome - objecto variam pouco ao longo do tempo

Cache de associações nome - objecto pode ser mantidas nos servidores ▼

Se não conseguir resolver o nome, devolve ao cliente para este continuar a resolução do nome

Resolução iterativa do nome ▼

Se não conseguir resolver o nome, encaminha a resolução para outro servidor

Servidor de nomes recursivo ▼

cn=Paulo Guedes, ou=Corpo Docente, ou=Dpto de Informática, o=ULHT, c=Portugal

Exemplo de um nome no X.500 ▼

Cria uma associação entre um nome e um objecto

Bind ▼

Faz a resolução do nome

Lookup ▼

Pergunta 13

Respondida

Nota: 6.67 em 10.00

🚩 Marcar pergunta

Considere a seguinte matriz, onde nas colunas estão representados Objectos, nas linhas estão representados Agentes e cada célula representa os direitos de acesso:

| Agentes / Objectos | O1 | O2 | O3 | O4 |
|--------------------|----|-----|----|-----|
| A1 | R | RW | RX | --- |
| A2 | RX | --- | RW | R |

Esta matriz é usada para que tipo de Controlo de Acesso ?

Discretionary Access Control (DAC) ▼

As Access Control List (ACL) correspondem a uma coluna ou a uma linha desta matriz ?

Uma linha ▼

Qual é a Access Control List (ACL) do Obj 2 ?

A1: RW; A2: --- ▼

Pergunta 14

Respondida

Nota: 8.00 em 10.00

🚩 Marcar pergunta

Estabeleça a correspondência entre as seguintes afirmações:

1. A informação está correcta e completa ▼ -> Integridade ▼
2. Acesso à informação apenas por agentes autorizados ▼ -> Privacidade ▼
3. Impossibilidade de um agente negar uma acção que efectivamente cometeu ▼ -> Não Repúdio ▼
4. Identificação do agente que produziu a informação ▼ -> Autenticidade ▼

Pergunta 15

Respondida

Nota: 2.00 em 8.00

🚩 Marcar pergunta

Considere 2 agentes A e B em que:

KApriv - Chave privada de A, só conhecida por A

KApub - Chave pública de A, conhecida por toda a gente

KBpriv - Chave privada de B, só conhecida por B

KBpub - Chave pública de B, conhecida por toda a gente

(M)K - Mensagem M cifrada com a chave K

Estabeleça as correspondências correctas quando A envia uma mensagem a B:

1. Autenticação ▼ -> A ->B (M)KApriv ▼
2. Autenticação e Privacidade ▼ -> A ->B [(M)KApriv]KBpub ▼
3. Privacidade ▼ -> A ->B (M)KBpub ▼

Pergunta 16

Respondida

Nota: 10.00 em 10.00

Marcar pergunta

Considere as 4 mensagens do protocolo Needham - Schroeder entre A, B e a autoridade de certificação S:

- $A \rightarrow S: A, B, Na$
- $S \rightarrow A: (Na, B, Kab, (Ts, A, Kab)Kb)Ka$
- $A \rightarrow B: (A, Ta)Kab, (Kab, A, Ts)Kb$
- $B \rightarrow A: (Ta + 1)Kab$

Na mensagem 2, A acredita que a mensagem vem da autoridade de certificação S porque:

Quando B recebe a mensagem 3, acredita que a componente $(Kab, A, Ts)Kb$ foi gerada pela autoridade de certificação S porque:

Para A enviar a B a mensagem 3 com a componente $(A, Ta)Kab$, A teve que:

Está cifrada com a chave de A

Está cifrada com a chave de S

Obter a chave Kab da mensagem 2

Pergunta 17

Respondida

Nota: 10.00 em 10.00

Marcar pergunta

Estabeleça as relações sobre as propriedades das transacções:

Os resultados de uma transacção que confirmou permanecem depois de esta acabar e são supostos sobreviver ao conjunto de falhas expectáveis dos mecanismos de armazenamento

Se diversas transacções se executarem em paralelo sobre os mesmos objectos, tudo se passa como se as transacções se executassem em série numa determinada ordem

Cada transacção deve, a partir de um estado inicial válido e caso se execute completamente, atingir um novo estado válido

Para um observador externo, uma transacção ou se executa na totalidade ou não se executa.

Peristência

Seribilidade

Consistência

Atomicidade

Pergunta 18

Respondida

Nota: 6.67 em 10.00

Marcar pergunta

No início do protocolo Two-Phase Commit (1ª fase), estabeleça as relações entre as seguintes acções:

O Coordenador envia a todos os RM participantes a mensagem Prepare_commit

O Coordenador obtém a decisão de todos os RM participantes e escreve um registo no diário com as respostas recebidas

Um participante regista a indicação de Prepare_commit, verifica se pode confirmar a transacção e responde ao Coordenador com "Commit" ou "Abort"

Início do protocolo na perspetiva do Coordenador

Neste momento, o Coordenador toma a decisão sobre o desfecho da transacção.

O Coordenador fica bloqueado e não pode continuar

Pergunta 19

Respondida

Nota: 7.50 em 10.00

Marcar pergunta

O Coordenador ou um Participante podem falhar durante a execução do protocolo Two-Phase Commit. Isso impacta o funcionamento dos outros nós e quando o nó que falhou recomear a funcionar, tem que recuperar o ponto onde estava no protocolo e continuar com o mesmo. Estabeleça as relações entre as afirmações seguintes:

Um participante tem um timeout no estado inicial quando espera a mensagem "Prepare_Commit" do Coordenador

O Coordenador falha e quando recupera verifica que estava no estado "Confirmar", tinha recebido "Commit" de todos os participantes mas poderia ainda não lhes ter enviado a sua decisão.

Um Participante falha e quando recupera verifica que estava no estado "Preparado", já depois de ter enviado a sua decisão ao Coordenador

O Coordenador obtém um timeout quando espera a resposta à mensagem "Prepare_Commit"

O Participante decide fazer Abort da transacção

O Coordenador reenvia a mensagem "Confirm_Commit" aos participantes

O Participante volta a ficar à espera da mensagem do Coordenador com a decisão sobre a transacção

O Participante decide fazer Abort da transacção

Pergunta 20

Respondida

Nota: 10.00 em 10.00

Marcar pergunta

Estabeleça as relações correctas relativamente à cache do NFS.

O servidor reutiliza a sua cache de blocos de disco

O cliente detecta que a sua cache pode estar desactualizada

O servidor sabe que clientes têm o ficheiro em cache

NFS

Sim

Sim

Não

Outras:

Pergunta 11

Por responder

Nota: 10.00

Marcar pergunta

Estabeleça a correspondência entre as seguintes afirmações:

Acesso à informação apenas por agentes autorizados

Impossibilidade de um agente negar uma acção que efectivamente cometeu

A informação está correcta e completa

Identificação do agente que produziu a informação

Privacidade

Não repúdio

Integridade

Autenticidade

Pergunta 13

Por responder

Nota: 10,00

🚩 Marcar pergunta

Estabeleça as relações correctas entre as seguintes afirmações relacionadas com o serviço e os servidores de nomes:

| | | |
|--|---|---|
| Lookup | Faz a resolução de um nome | ↕ |
| Bind | Cria a associação entre um nome e um objecto | ↕ |
| Servidor de nomes recursivo | Se não conseguir resolver o nome, encaminha a resolução para outro servidor | ↕ |
| As associações nome - objecto podem ser mantidas em cache nos servidores | Porque as relações nome - objecto variam pouco ao longo do tempo | ↕ |

Pergunta 17

Por responder

Nota: 10,00

🚩 Marcar pergunta

O Coordenador ou um Participante podem falhar durante a execução do protocolo Two-Phase Commit. Isso impacta o funcionamento dos outros nós e quando o nó que falhou recomeçar a funcionar, tem que recuperar o ponto onde estava no protocolo e continuar com o mesmo. Estabeleça as relações entre as afirmações seguintes:

| | | |
|--|--|---|
| Um participante tem um timeout no estado inicial quando espera a mensagem "Prepare_Commit" do Coordenador | O Participante decide fazer Abort da transação | ↕ |
| O Coordenador falha e quando recupera verifica que estava no estado "Confirmar", tinha recebido "Commit" de todos os participantes mas poderia ainda não lhes ter enviado a sua decisão. | O Coordenador reenvia a mensagem "Confirm_Abort" aos participantes | ↕ |
| O Coordenador obtém um timeout quando espera a resposta à mensagem "Prepare_Commit" | O Coordenador decide fazer Abort da transação | ↕ |
| Um Participante falha e quando recupera verifica que estava no estado "Preparado", depois de ter enviado o seu voto de Commit ao Coordenador | O Participante volta a ficar à espera da mensagem do Coordenador com a decisão sobre a transação | ↕ |