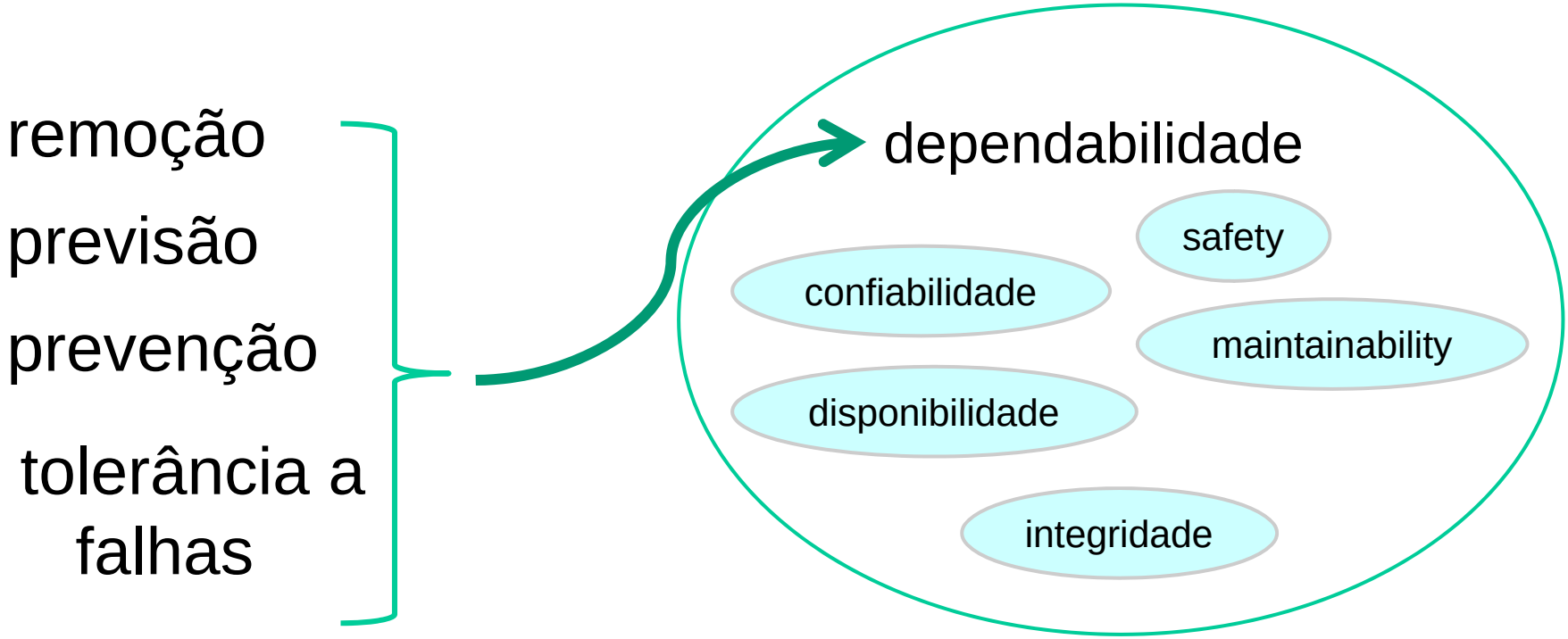
Técnicas de Tolerância a Falhas

Taisy Silva Weber
UFRGS

Meios para alcançar dependabilidade



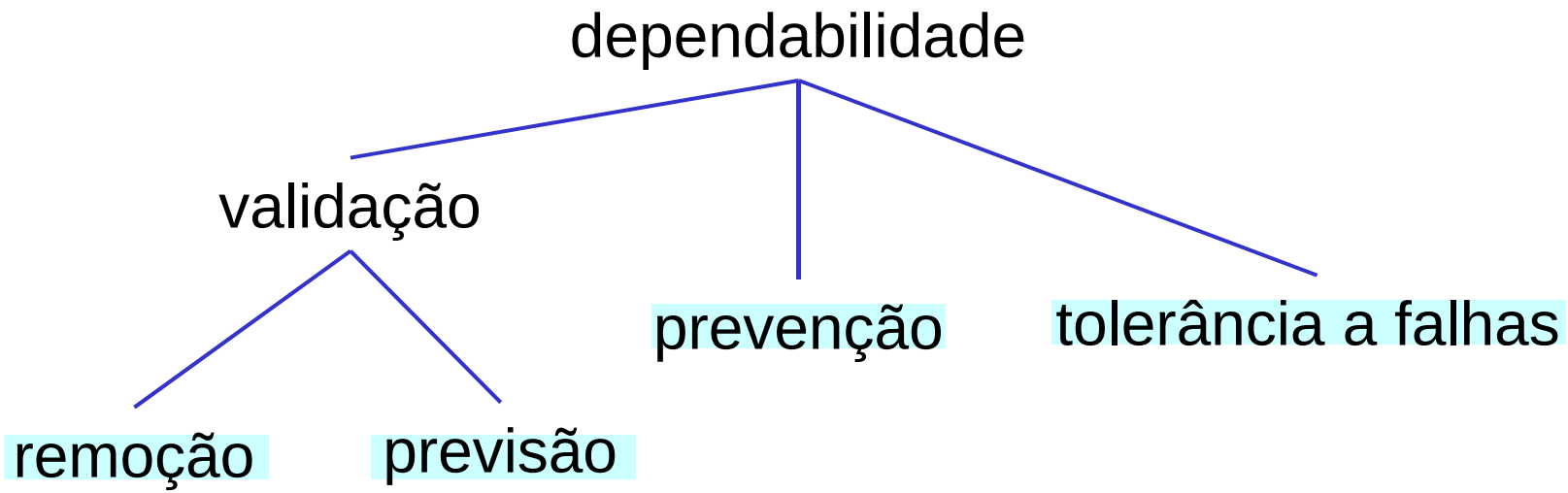
dependabilidade depende de decisões de projeto:

para alcançar dependabilidade (e seus atributos) são necessária técnicas de projeto adequadas

Dependabilidade sem TF?

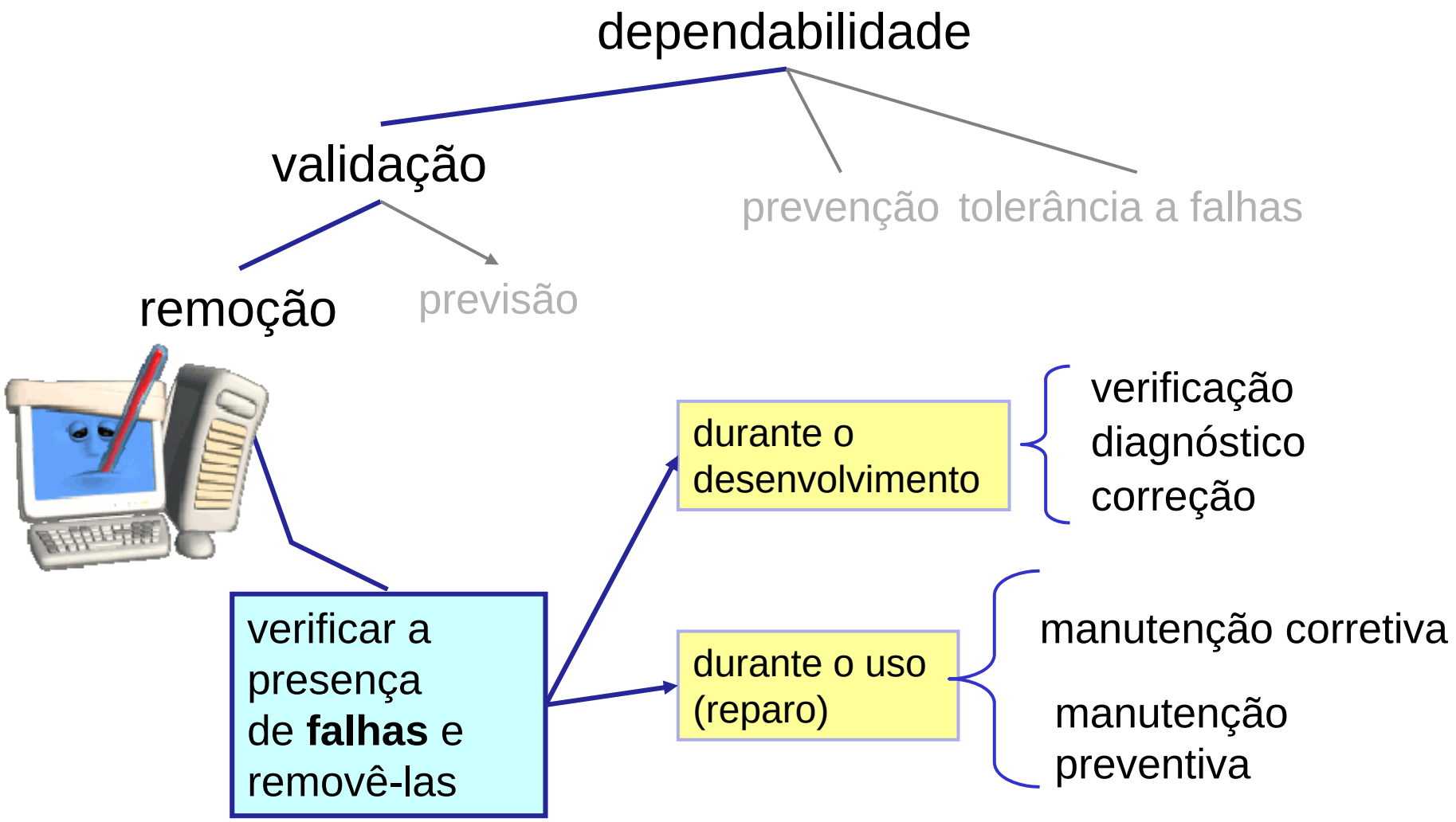
- dependabilidade pode ser alcançada sem TF
  - bons componentes podem levar a uma boa confiabilidade dos sistema
  - bons processos de produção e teste resultam em aumento de dependabilidade
  - manutenção frequente aumenta a qualidade

Meios



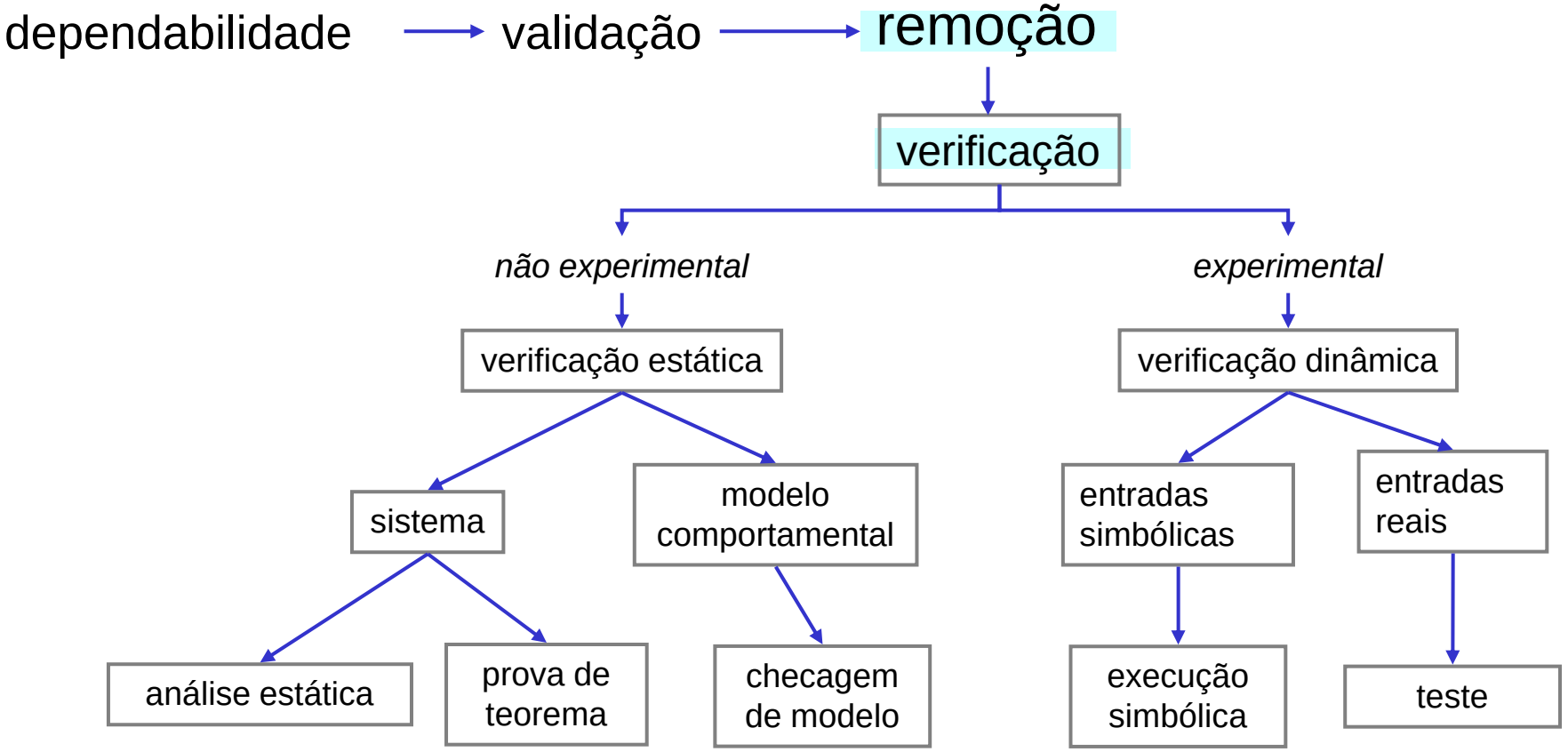
Avizienis - **meios** para alcançar dependabilidade: **remoção, previsão, prevenção** e **tolerância a falhas**

Meios: validação

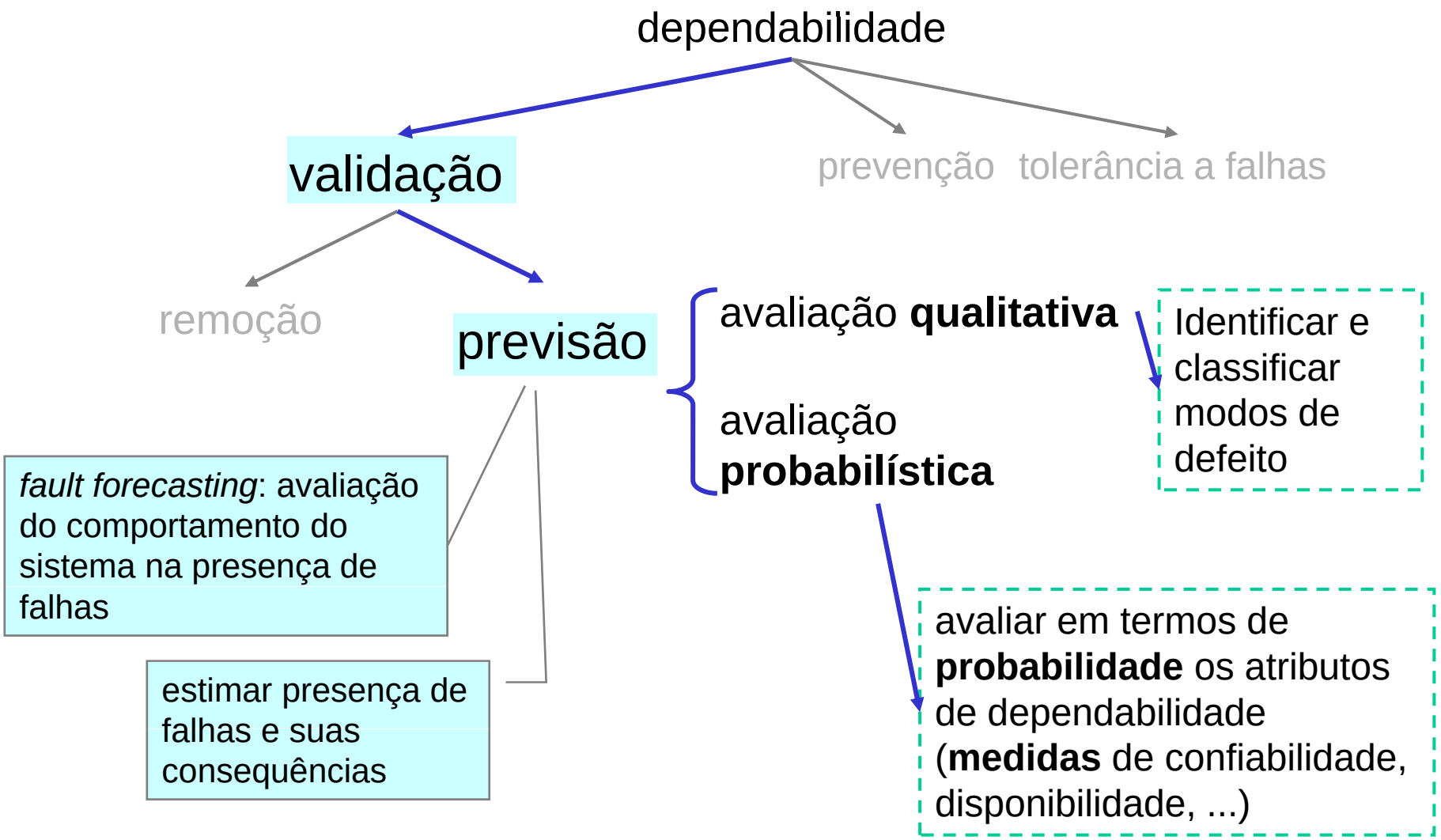


Meios: remoção > verificação

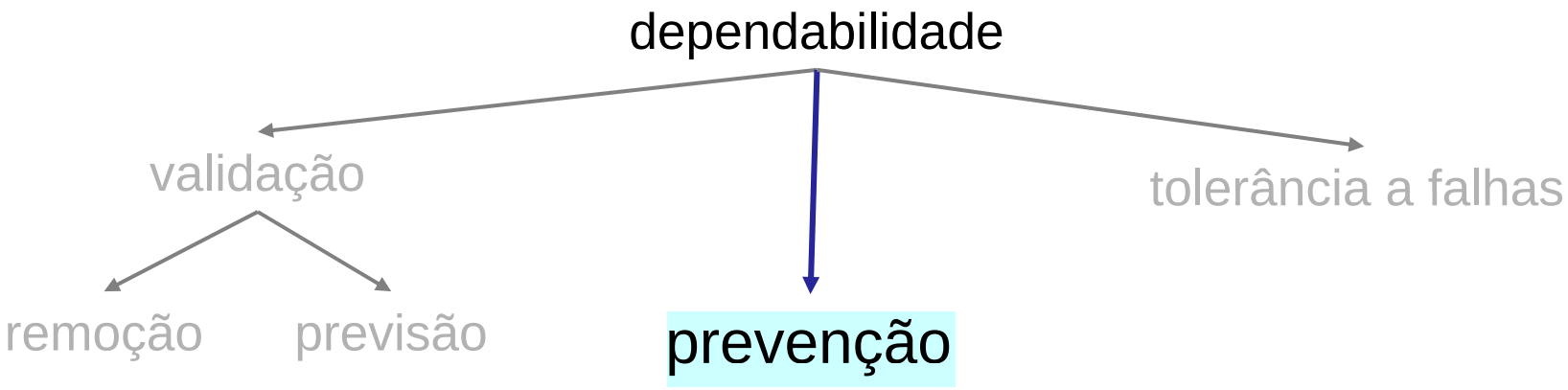
durante do desenvolvimento



Meios: previsão



Meios: prevenção

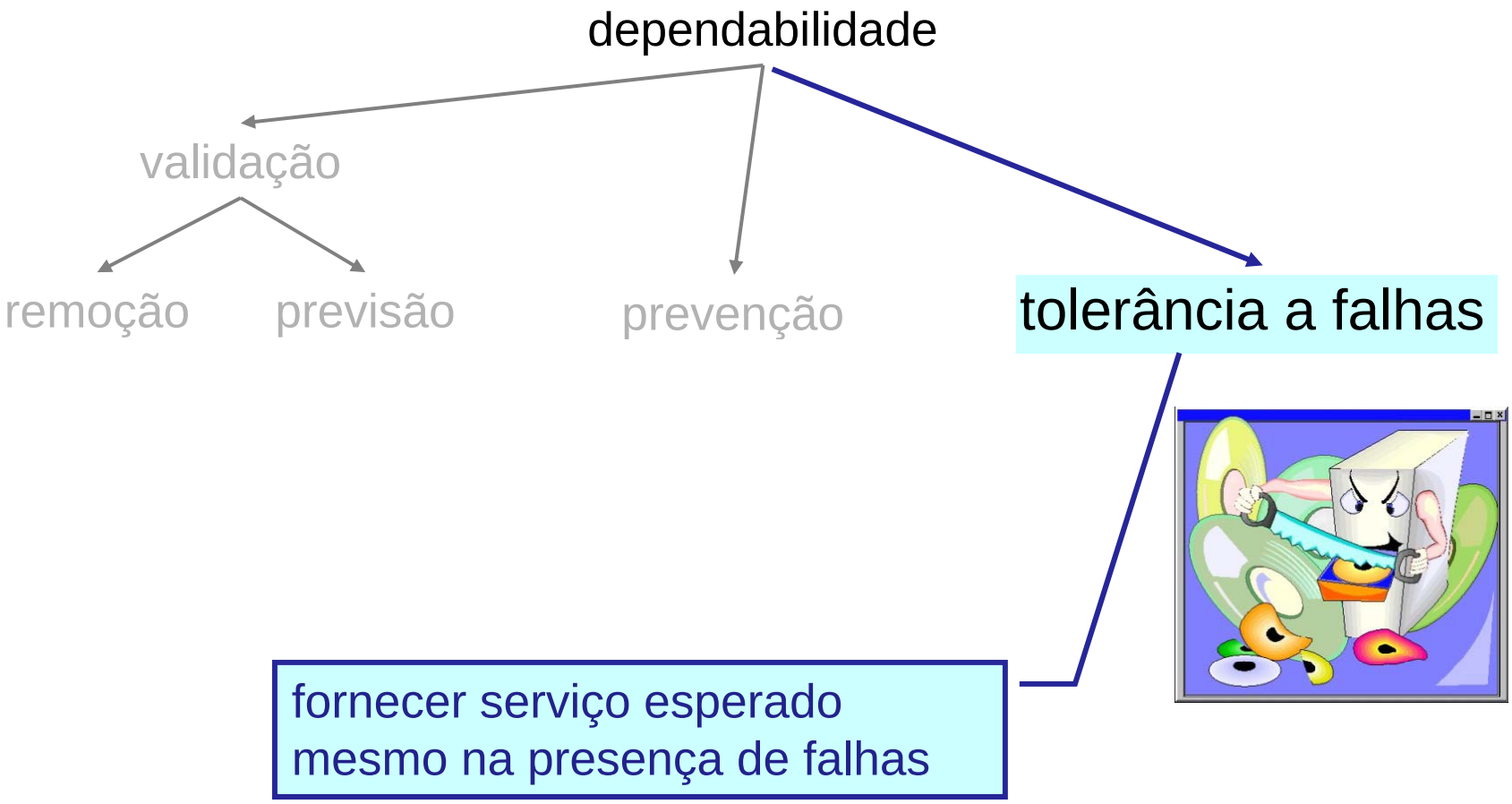


evitar introduzir falhas durante projeto e construção do sistema

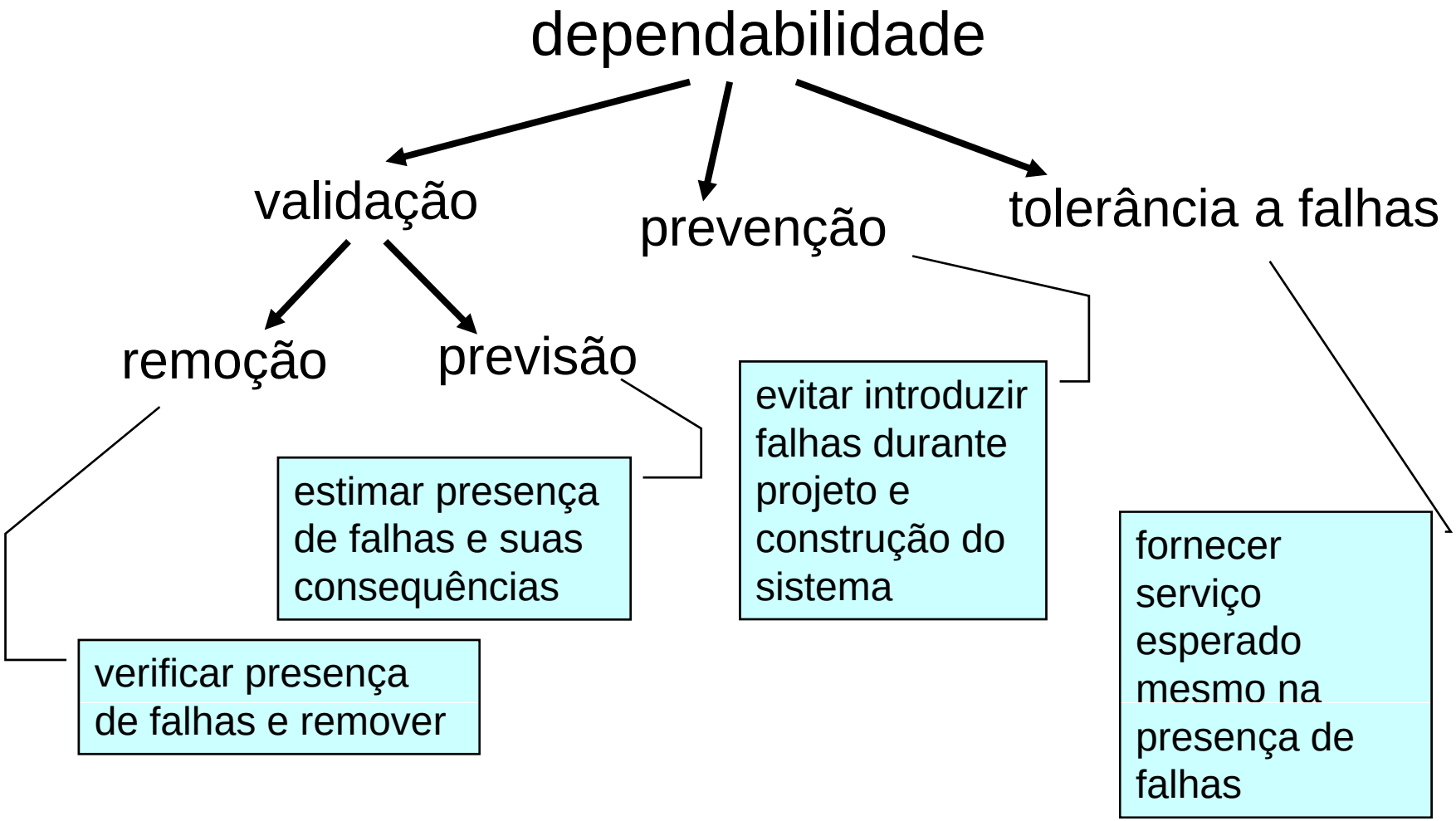
seleção de metodologias de projeto e tecnologia apropriada

não resolve problemas críticos de confiabilidade

Meios: tolerância a falhas



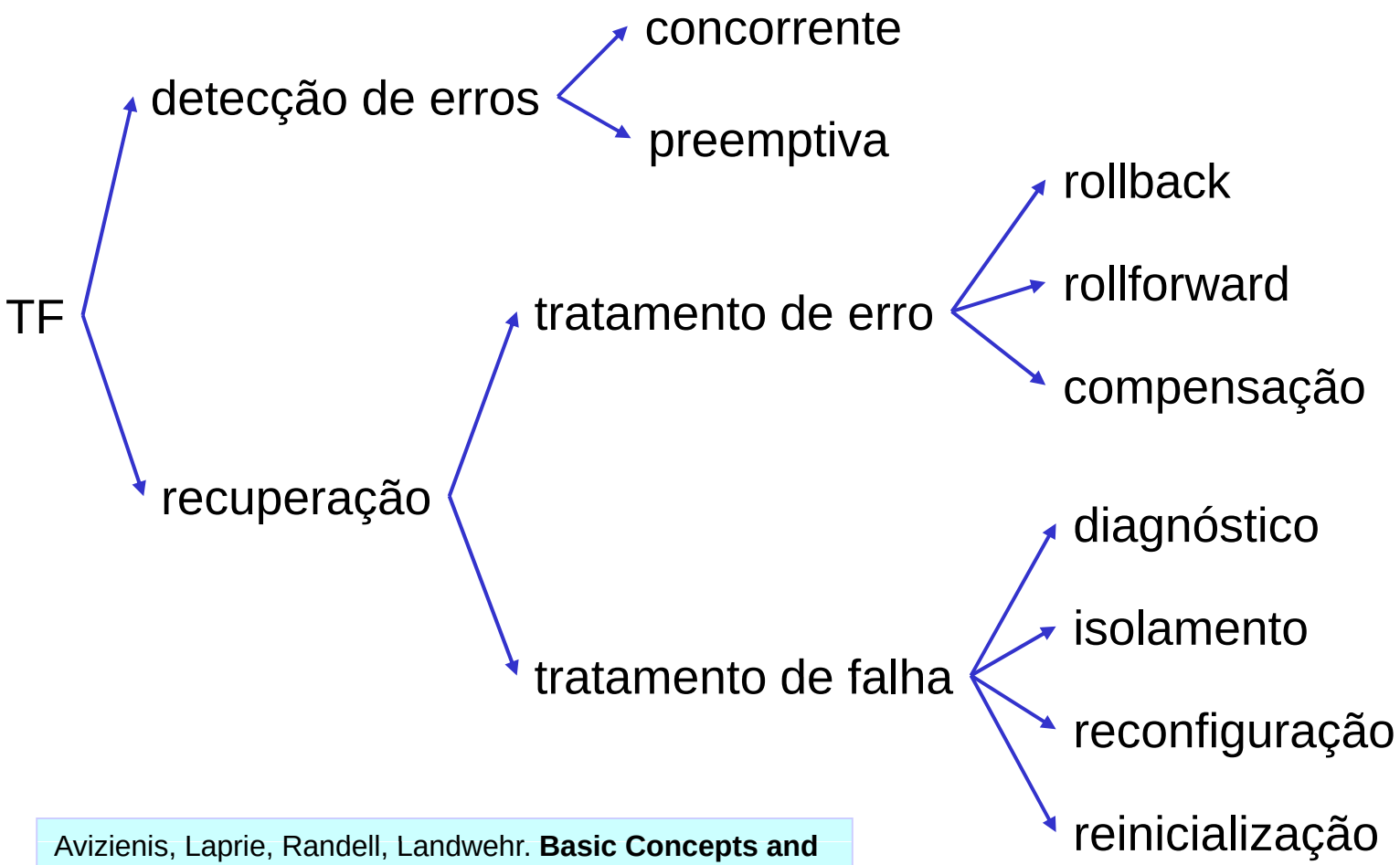
Meios


dependabilidade
validação
prevenção
tolerância a falhas
remoção
previsão
evitar introduzir falhas durante projeto e construção do sistema
estimar presença de falhas e suas consequências
fornecer serviço esperado mesmo na presença de falhas
verificar presença de falhas e remover

Técnicas de TF

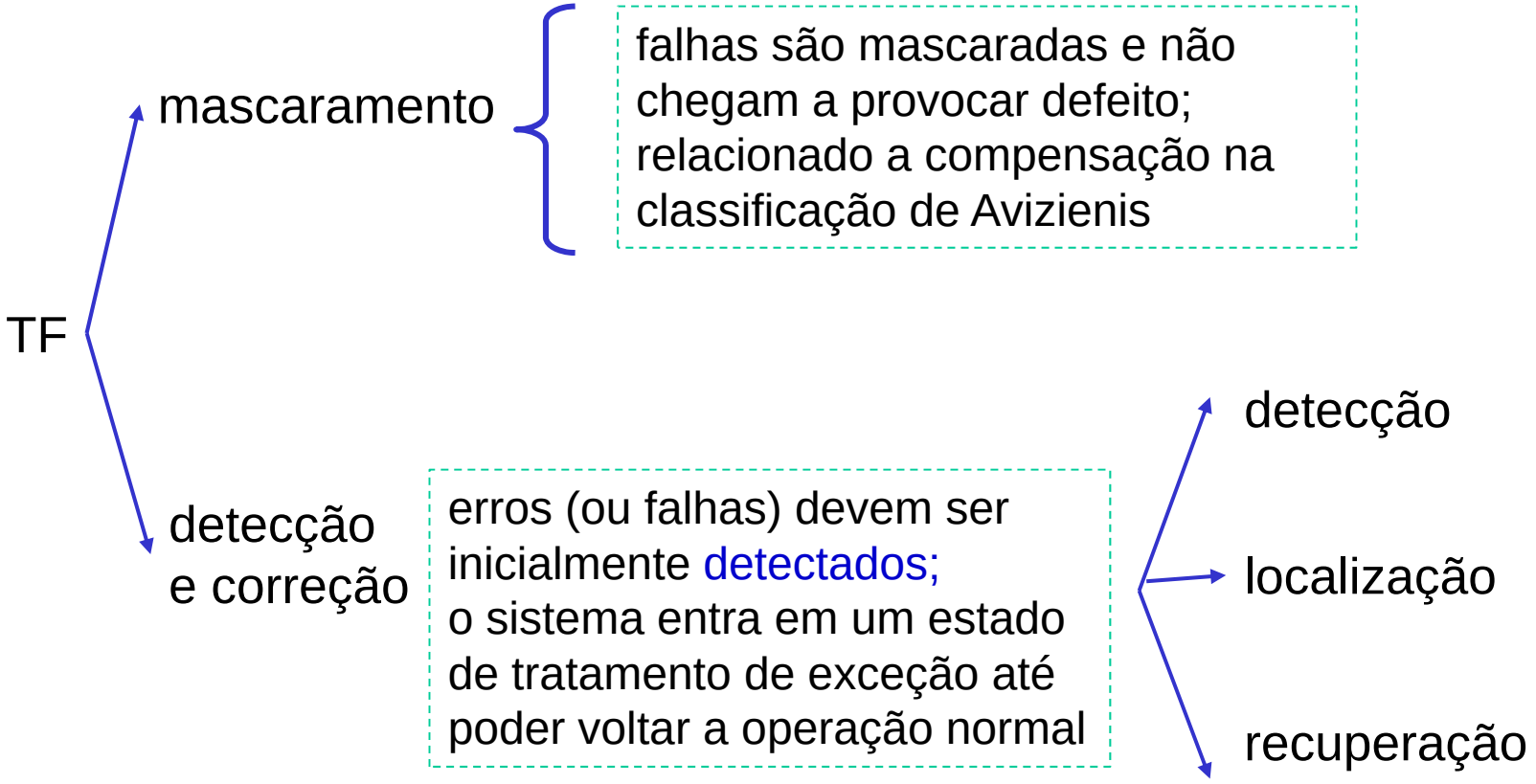
- prevenção e remoção de falhas não são suficientes:
  - quando o sistema exige alta confiabilidade,
  - ou alta disponibilidade
- técnicas de TF exigem
  - componentes adicionais
  - algoritmos especiais

Técnicas de TF



Avizienis, Laprie, Randell, Landwehr. **Basic Concepts and Taxonomy of Dependable and Secure Computing.** IEEE Trans. on dep. and secure comp. 2004

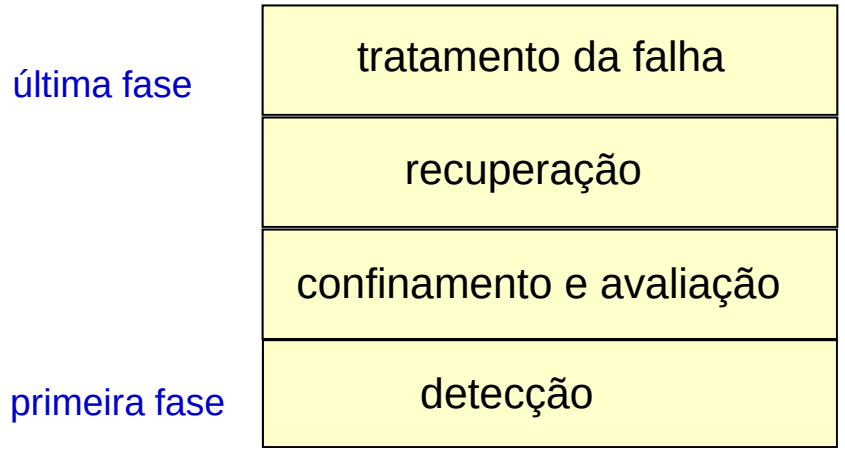
Técnicas de TF : outra classificação



técnicas alternativas, mutuamente exclusivas

mais classificações

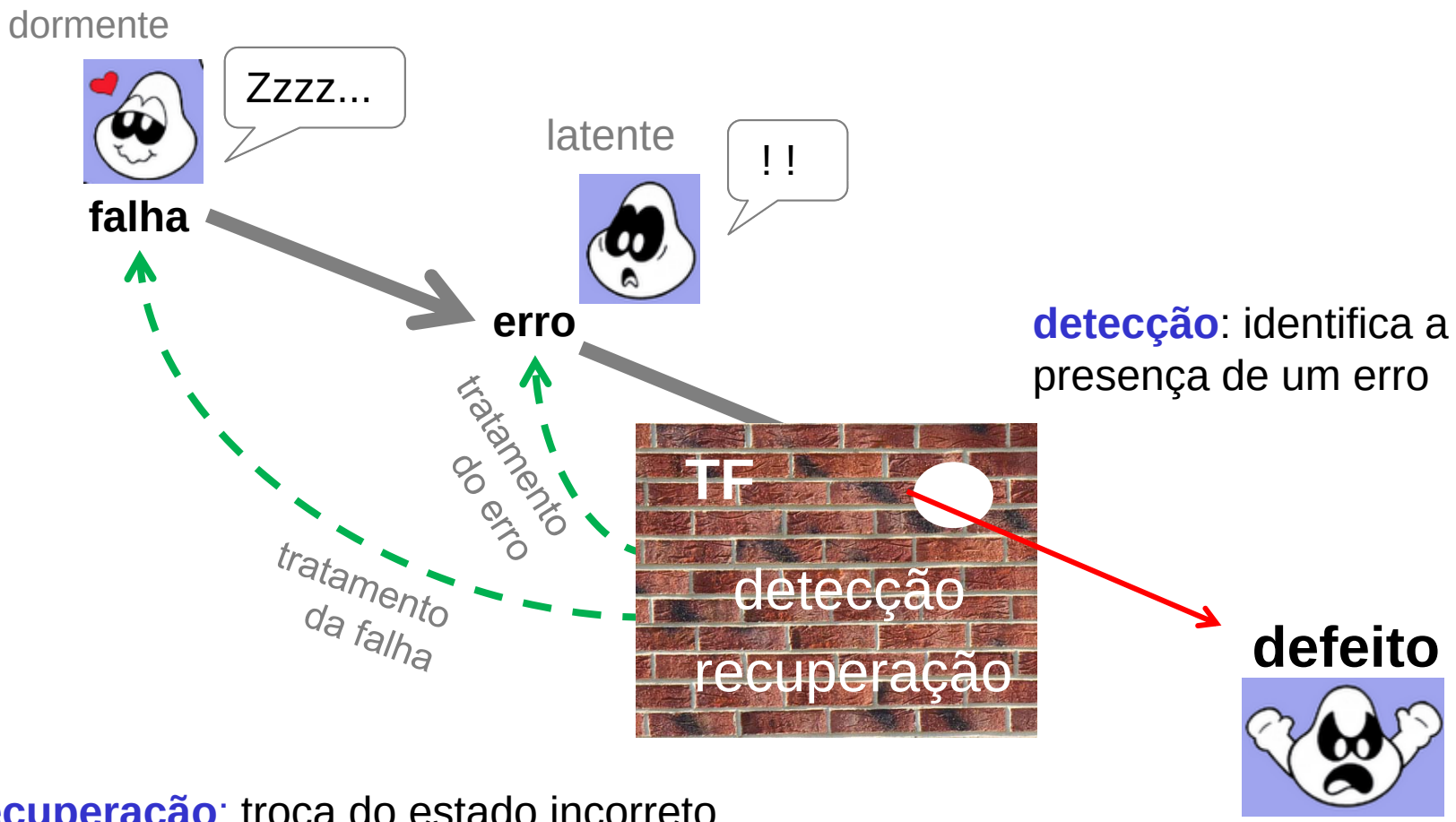
- 4 fases (*Anderson & Lee*):



- múltiplas fases (*Nelson*)
  - mascaramento, detecção, confinamento, diagnóstico, reparo, configuração, recuperação

Nelson, V. **Fault Tolerant Computing: Fundamental Concepts.** IEEE Computer, 1990

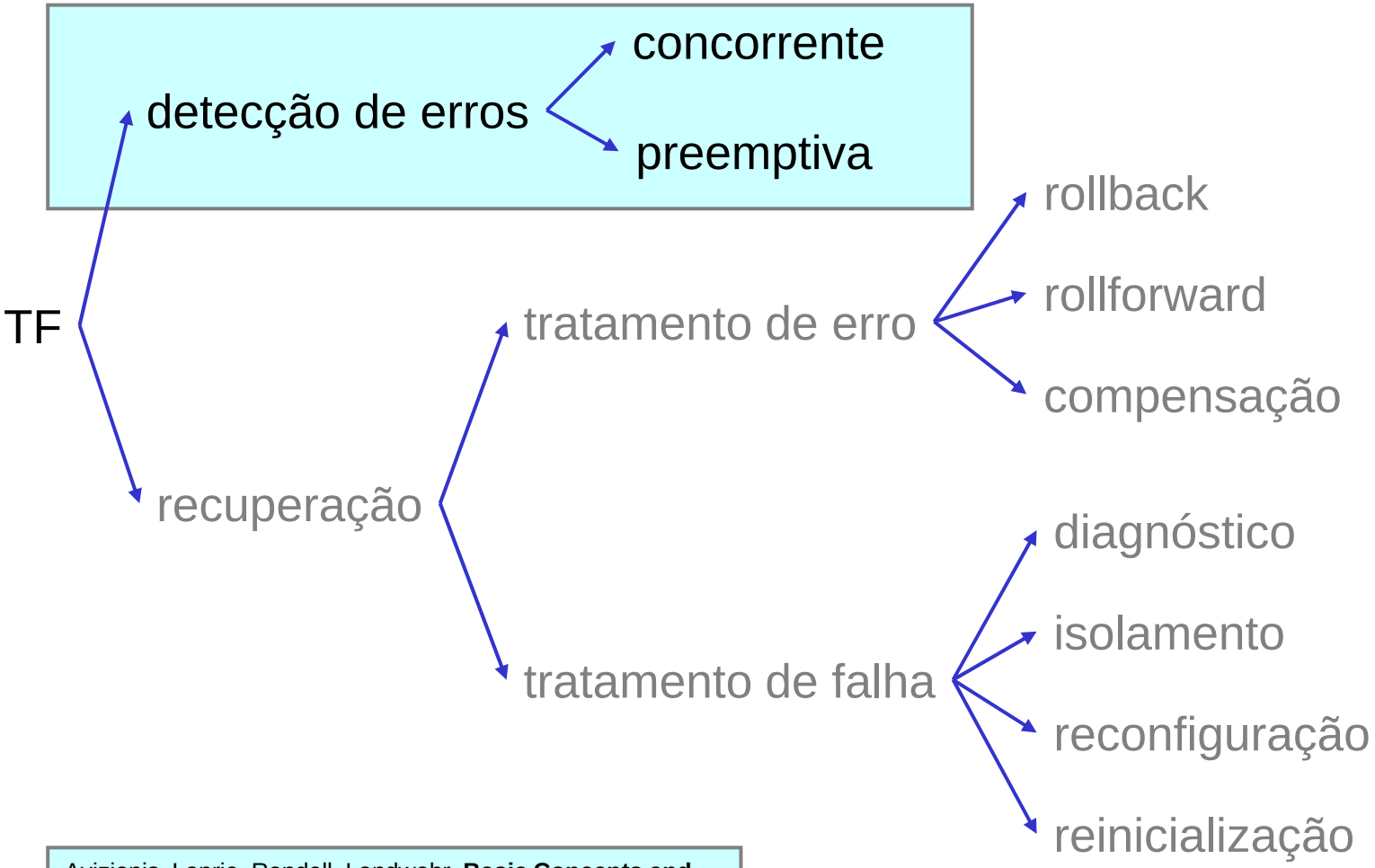
Técnicas de TF



detecção: identifica a presença de um erro

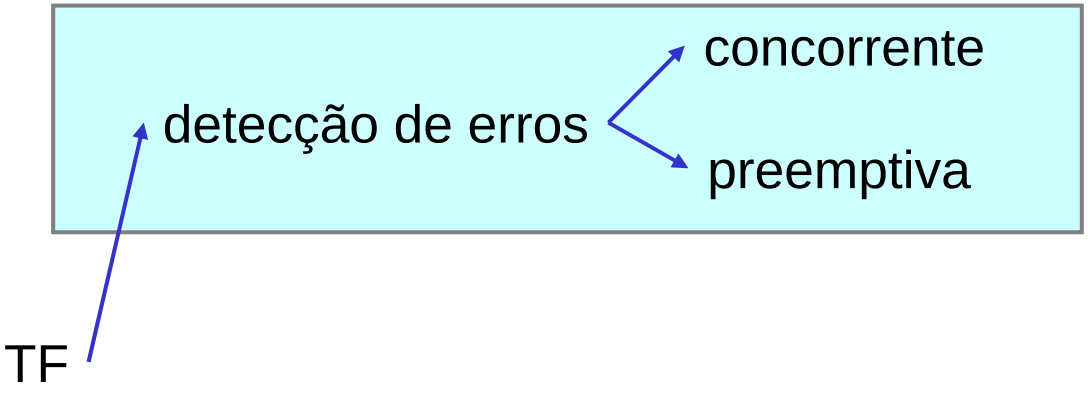
recuperação: troca do estado incorreto para um estado livre dos erros detectados e de falhas que possam ser ativadas

Técnicas de TF: Avizienis



Avizienis, Laprie, Randell, Landwehr. **Basic Concepts and Taxonomy of Dependable and Secure Computing.** IEEE Trans. on dep. and secure comp. 2004

Técnicas de TF: Avizienis



concorrente: durante período normal de operação

preemptiva: com serviço normal suspenso

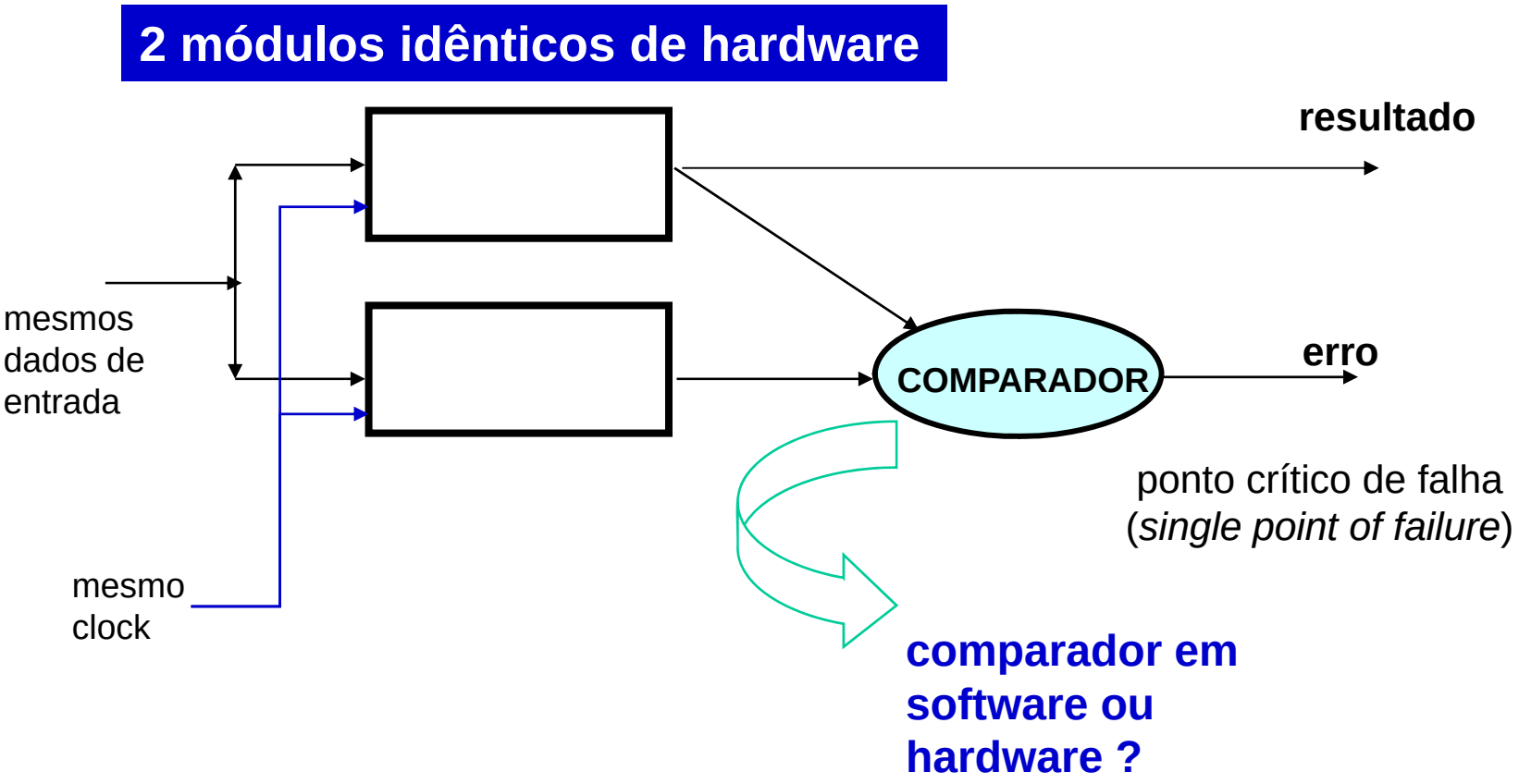
Avizienis, Laprie, Randell, Landwehr. **Basic Concepts and Taxonomy of Dependable and Secure Computing.** IEEE Trans. on dep. and secure comp. 2004

Técnicas de detecção de erros

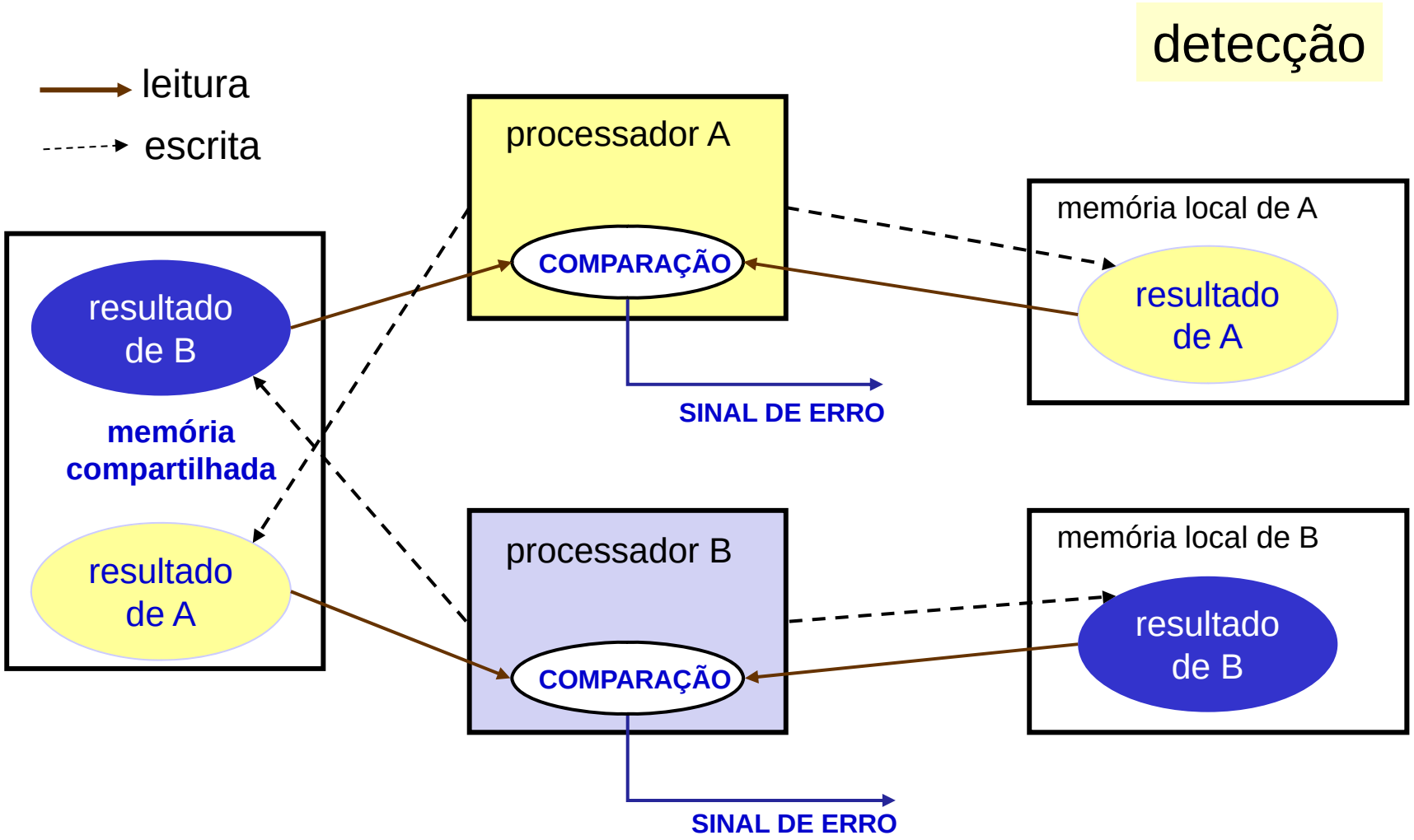
- duplicação e comparação
- codificação
- testes:
  - testes de limites de tempo
    - *time-out*, cão de guarda (*watchdog timers*)
  - testes reversos
  - teste de limites ou compatibilidade
  - testes de consistência
- diagnóstico
  - pode ser concorrente à operação normal ou preemptivo

Duplicação e comparação

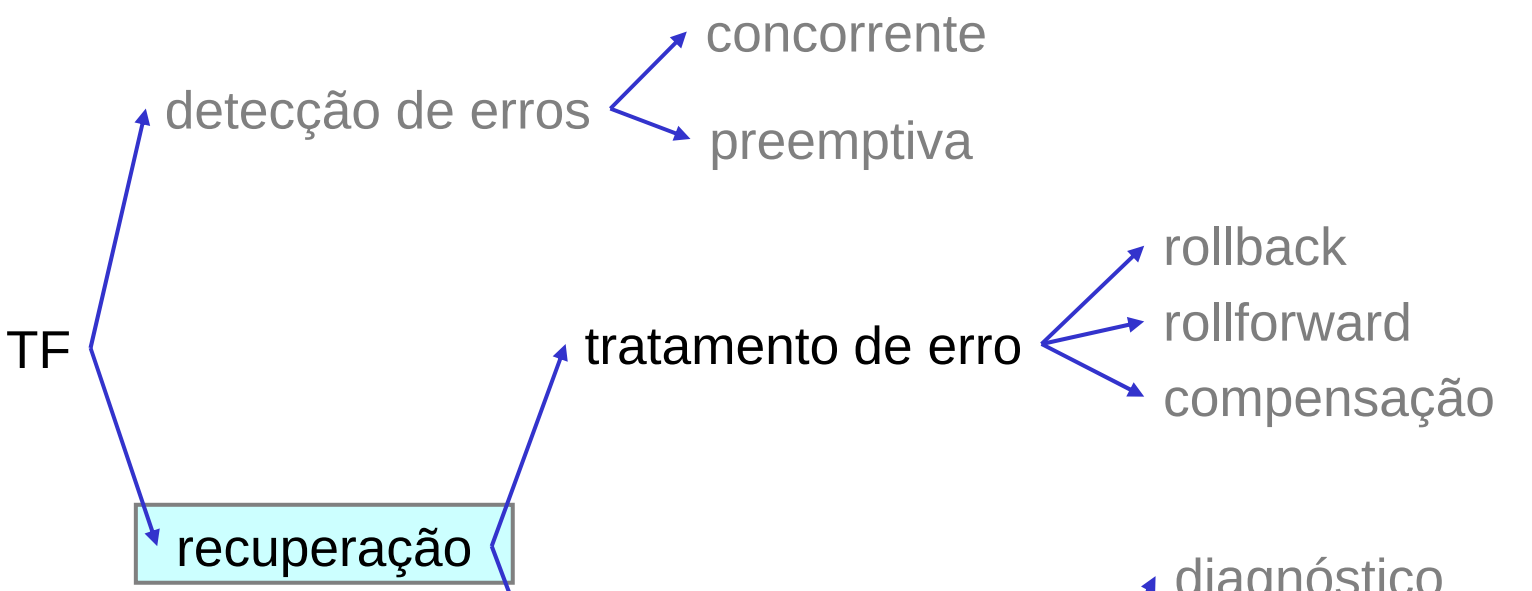
detecção



Duplicação e comparação



Técnicas de TF

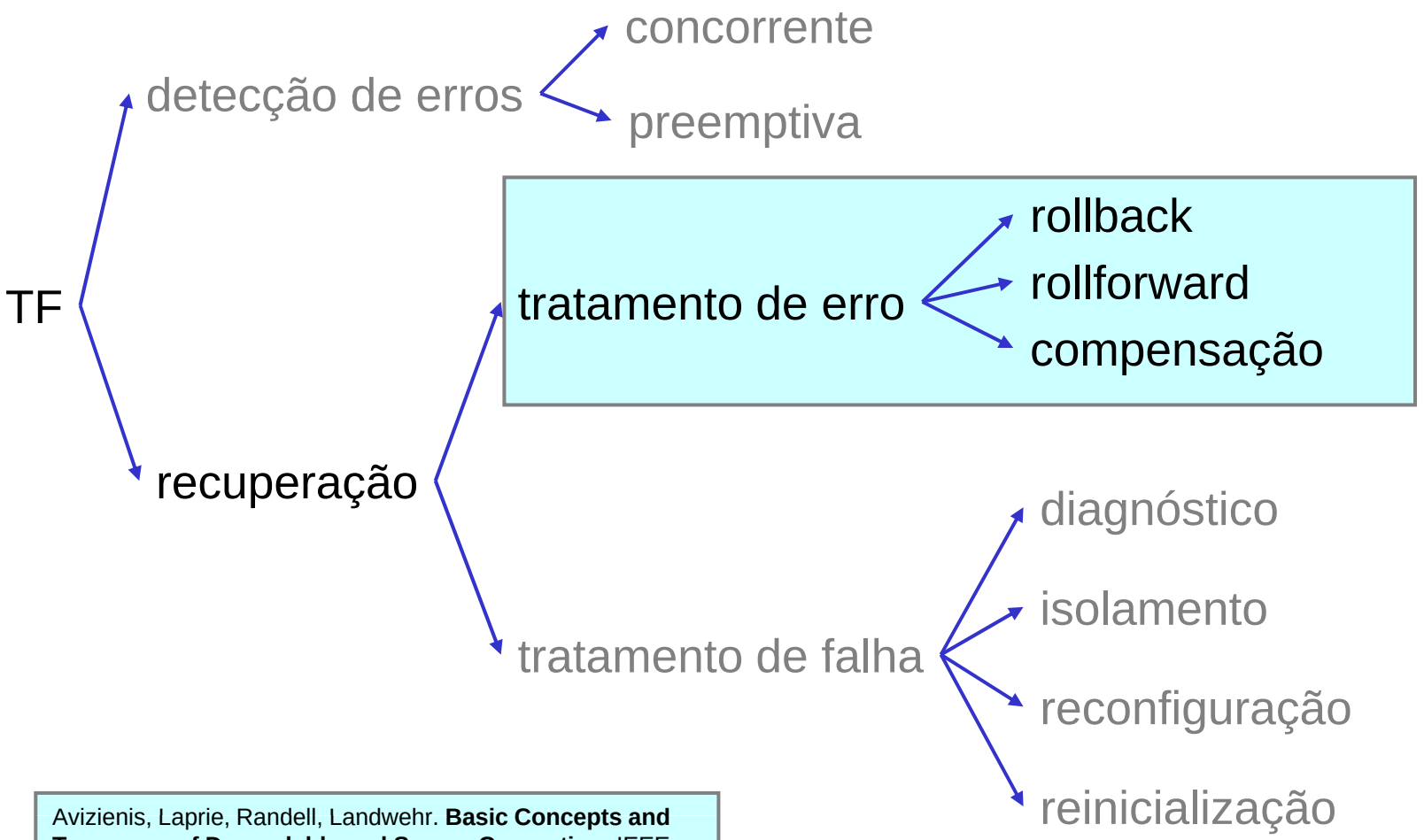


Avizienis, Laprie, Randell, Landwehr. **Basic Concepts and Taxonomy of Dependable and Secure Computing.** IEEE Trans. on dep. and secure comp. 2004

Recuperação

- objetivo
  - troca do estado atual incorreto (que contém um ou mais erros e possivelmente falhas) para um estado livre dos erros detectados e de falhas que possam ser novamente ativadas
  - ocorre sempre após detecção
- tipos
  - tratamento de erros
    - elimina erros do estado do sistema
  - tratamento de falhas
    - previne reativação de falhas

Técnicas de TF

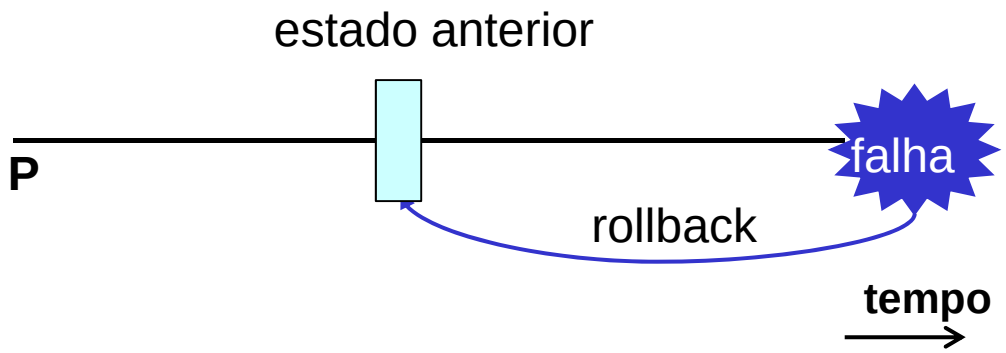


Avizienis, Laprie, Randell, Landwehr. **Basic Concepts and Taxonomy of Dependable and Secure Computing.** IEEE Trans. on dep. and secure comp. 2004

Tratamento de erros

- recuperação por retorno

condução a **estado anterior**

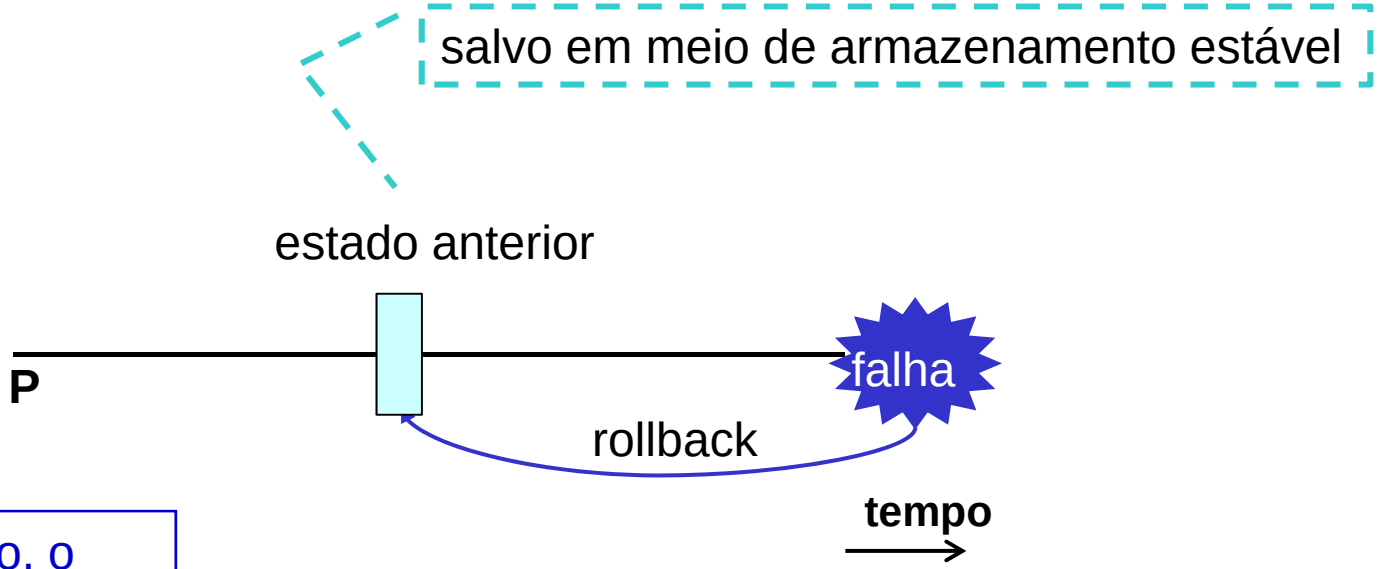


- recuperação por avanço

condução a **novo estado**

Retorno

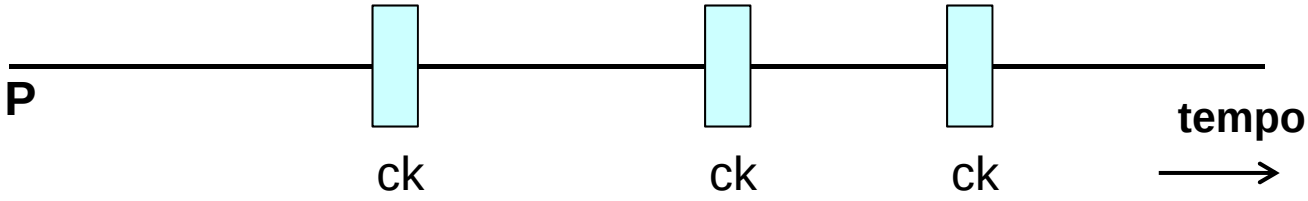
- condução a estado anterior consistente
  - implica no salvamento do estado anterior livre de erros
  - alto custo mas de aplicação genérica



em último caso, o estado anterior pode ser o estado inicial

Retorno: exemplo

- salvamento de todo o estado do sistema periodicamente (checkpoints)



  - simples em um único processo isolado
    - backup e log de operações
  - complexa em processamento distribuído
    - sem restrições a comunicação pode provocar efeito dominó

Avanço

- recuperação por avanço
  - condução a novo estado consistente ainda não ocorrido desde a última manifestação de erro
    - eficiente, mas específica a cada sistema
    - danos devem ser previstos acuradamente

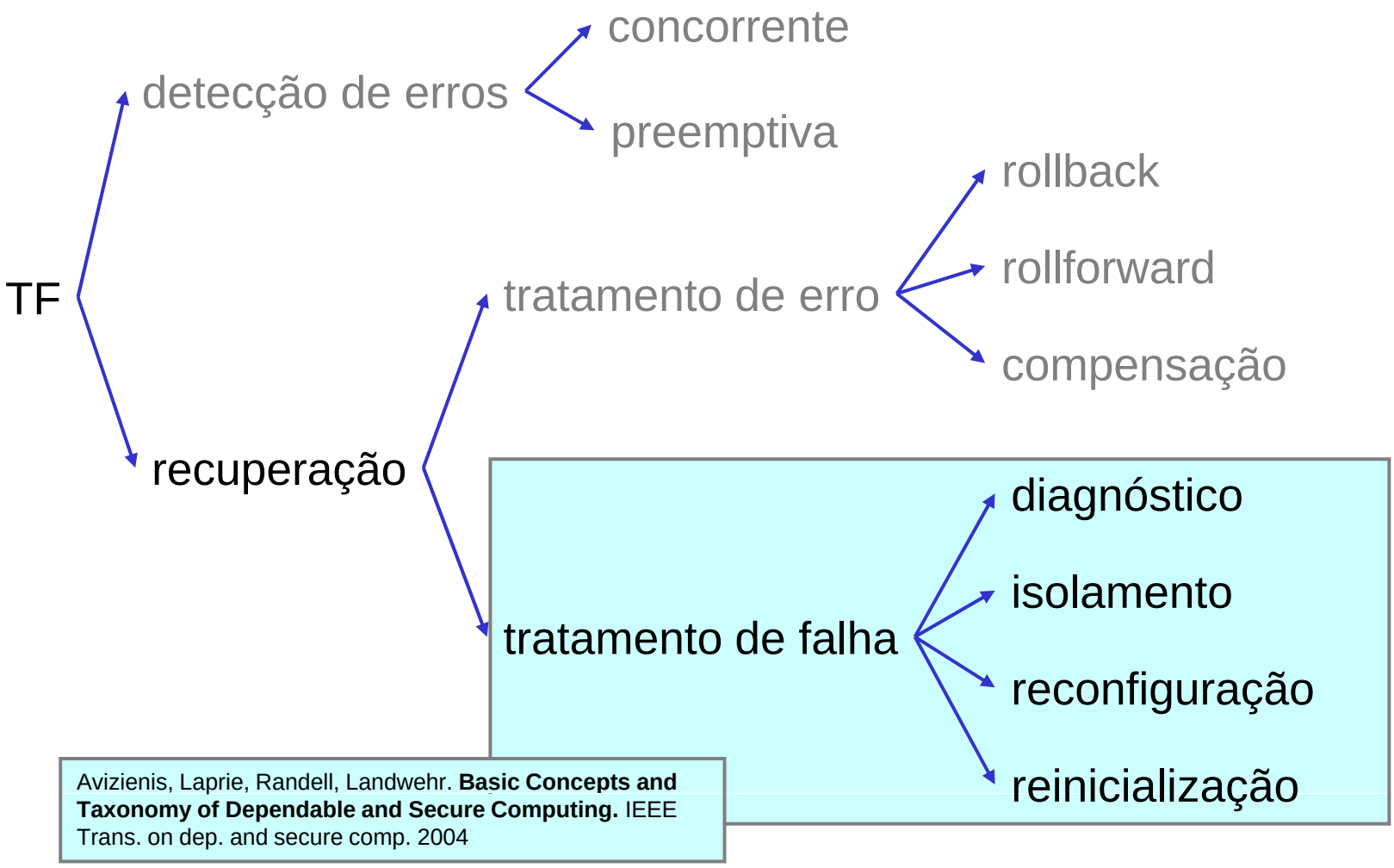
- os dois tipos de recuperação (avanço e retorno) não são mutuamente excludentes

mais usada em sistemas de **tempo real**, onde o retorno para um estado anterior (no tempo) seja inviável

Compensação

- o estado errôneo contém redundância suficiente para permitir que o erro seja mascarado
  - aplicação sistemática de compensação leva a mascaramento de falhas
    - mascaramento pode levar a perda progressiva da redundância de proteção (caso ocorram falhas permanentes e não envolva detecção de erros)
  - exemplos de mascaramento:
    - ECC
    - TMR (serão vistos posteriormente)

Técnicas de TF

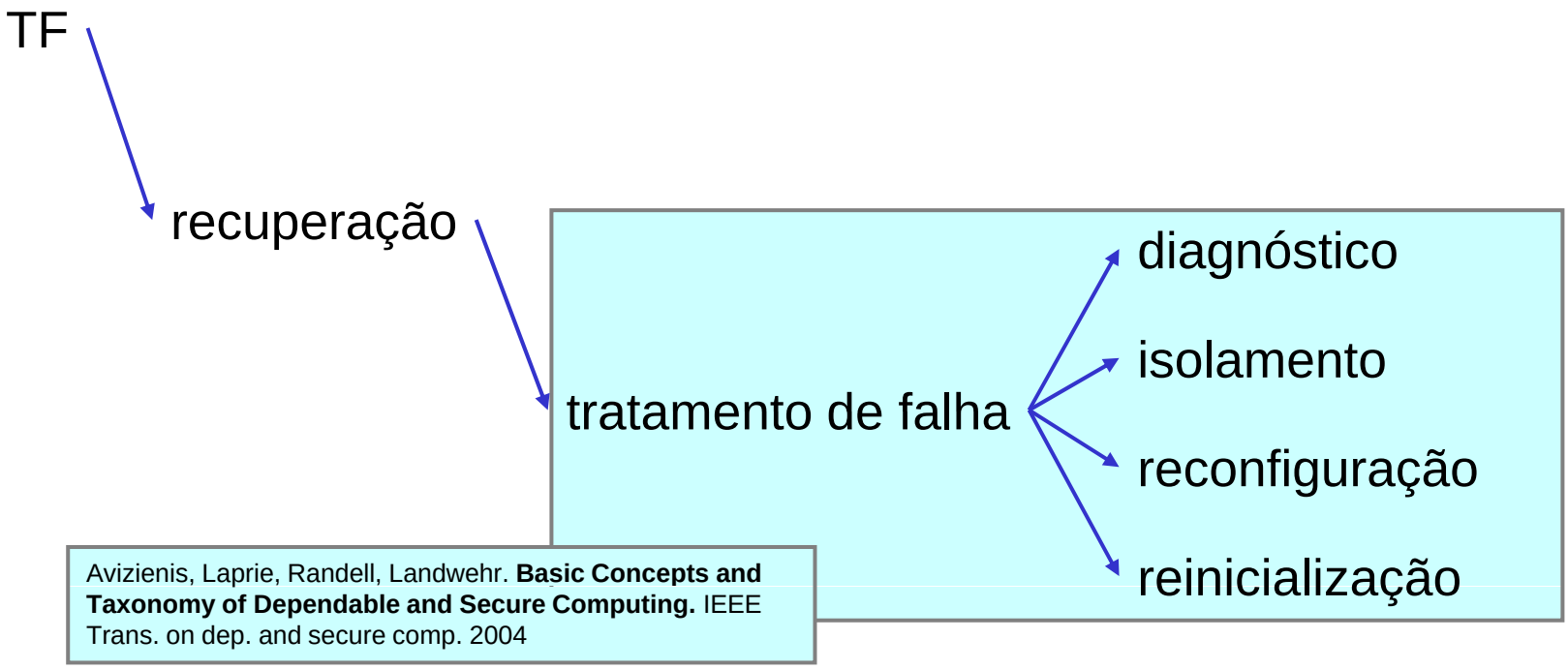


Avizienis, Laprie, Randell, Landwehr. **Basic Concepts and Taxonomy of Dependable and Secure Computing.** IEEE Trans. on dep. and secure comp. 2004

Tratamento de falhas

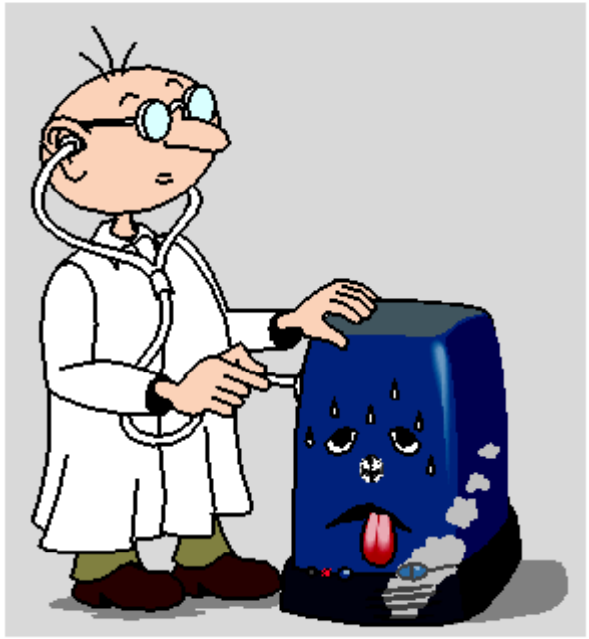
tratamento da falha:

previne reativação de falhas;
geralmente seguida de
manutenção corretiva



Tratamento de falhas

- diagnóstico
  - identifica a causa do erro e armazena informação de localização e tipo



Isolamento

- confinamento
  - latência de falha pode provocar espalhamento de dados inválidos
  - o confinamento estabelece limites para a propagação do dano

depende de decisões de projeto do sistema

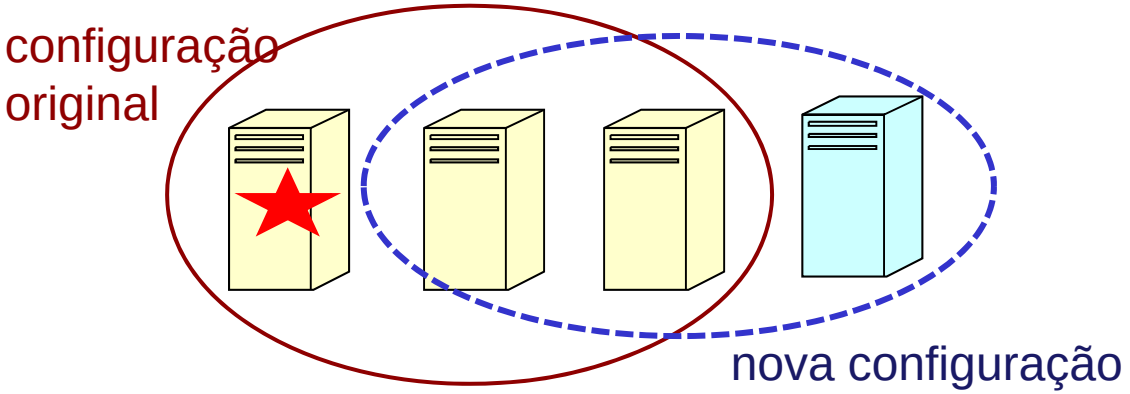
facilita isolamento

restringir fluxo de informações: evitar fluxos acidentais, estabelecer interfaces de verificação para detecção

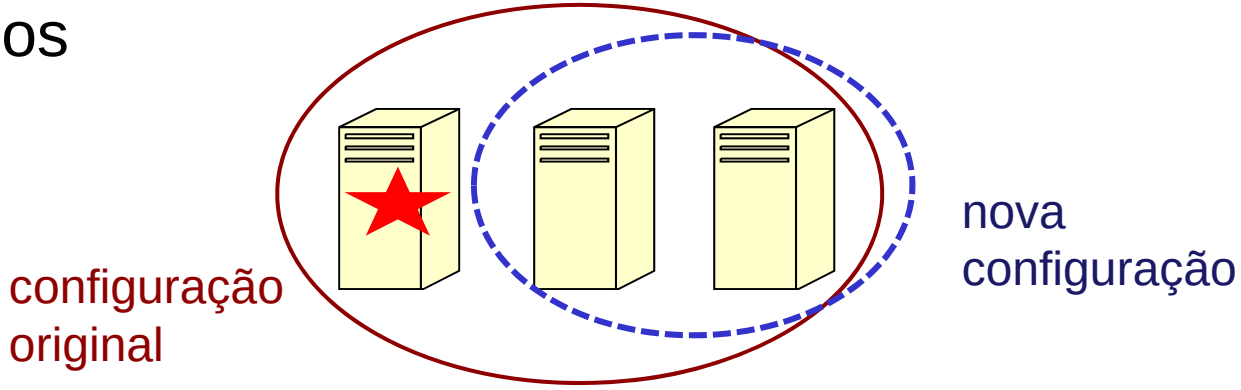
- isolamento
  - exclui componente com falha do sistema
  - o isolamento pode ser físico ou lógico
  - falhas isoladas devem ser posteriormente removidas

Reconfiguração

- chaveia para componentes redundantes em espera

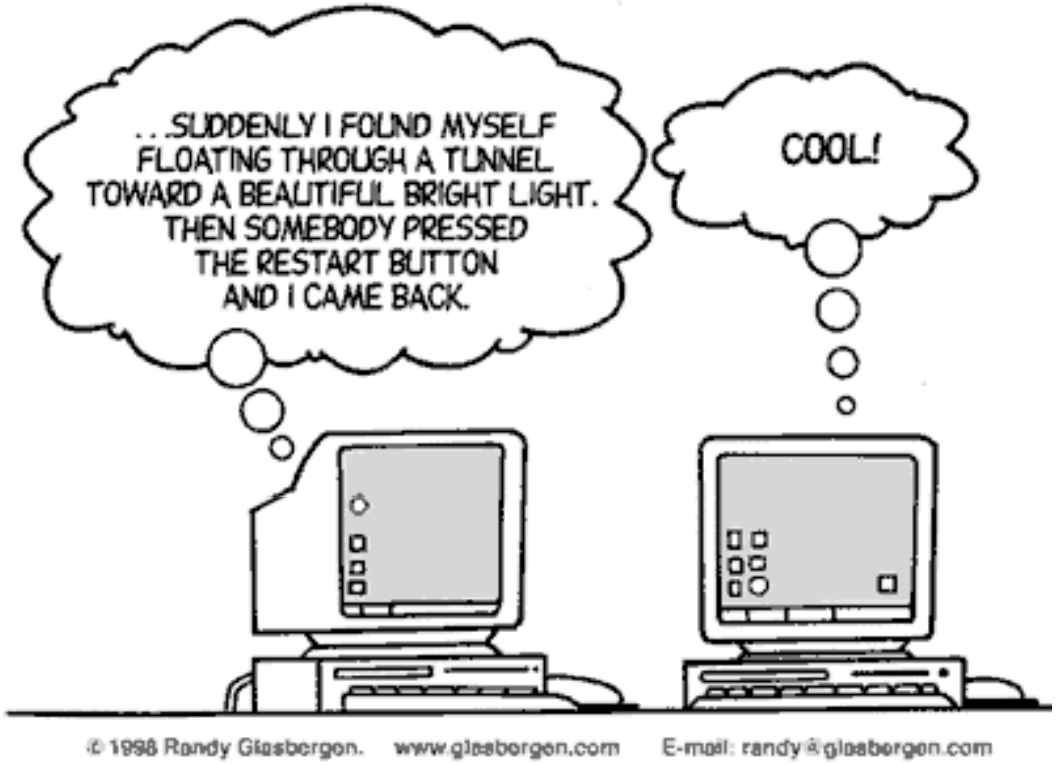


- redistribui tarefas entre componentes não defeituosos



Reinicialização

- verifica, atualiza e guarda a nova configuração
- atualiza informações de configuração do sistema

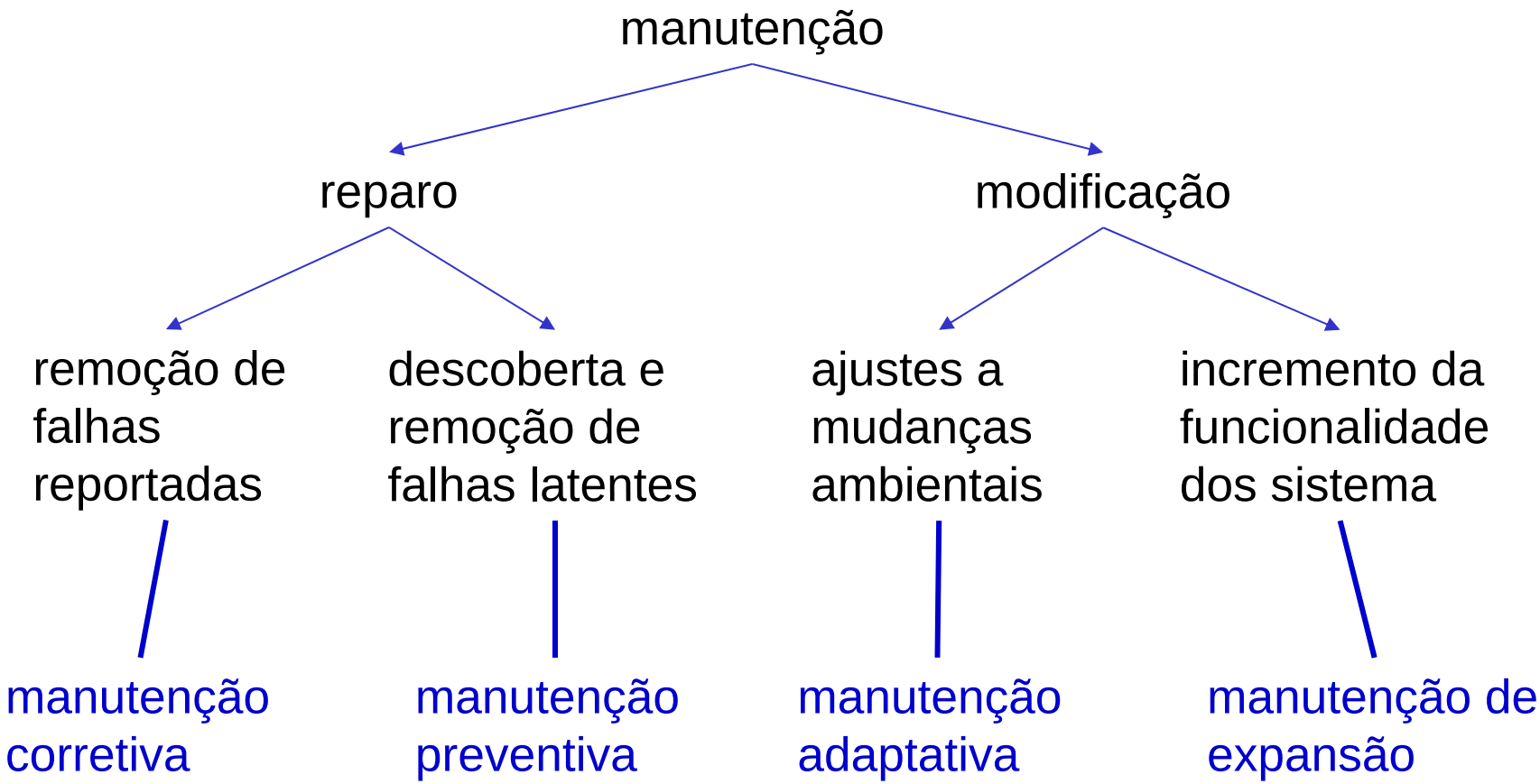


TF vs manutenção

- reparo e TF são temas relacionados
  - manutenção envolve a ação de um agente externo
  - manutenção
    - reparo
    - modificação
  - reparo é uma atividade de remoção de falhas
    - remoção de falhas e tolerância a falhas são meios para alcançar dependabilidade

Avizienis, Laprie, Randell, Landwehr. **Basic Concepts and Taxonomy of Dependable and Secure Computing.** IEEE Trans. on dep. and secure comp. 2004

Manutenção


manutenção
reparo
modificação
remoção de falhas reportadas
descoberta e remoção de falhas latentes
ajustes a mudanças ambientais
incremento da funcionalidade dos sistema
manutenção corretiva
manutenção preventiva
manutenção adaptativa
manutenção de expansão

Bibliografia

- capítulo de livro
  - Johnson, Barry. An introduction to the design na analysis of the fault-tolerante systems, cap 1. **Fault-Tolerant System Design**. Prentice Hall, New Jersey, 1996
- artigos
  - Avizienis, Laprie, Randell, Landwehr. **Basic Concepts and Taxonomy of Dependable and Secure Computing.** IEEE Trans. on dep. and secure comp. 2004
  - Nelson, V. – Fault Tolerant Computing: Fundamental Concepts. IEEE Computer. 1990