



BC-1503
Arquitetura de Computadores



Universidade Federal do ABC

Sistemas Computacionais

Guiou Kobayashi
guiou.kobayashi@ufabc.edu.br

2º Quadrimestre, 2014



CONTEÚDO PROGRAMÁTICO:

- História e Evolução dos Computadores e Sistemas
- Estrutura de Computadores Digitais
- Lógica Digital Binária
- Processamento
- Instruções e linguagem de máquina
- Microprocessadores modernos: pipeline, super escalar, RISC
- Memórias cache e gerenciamento de memórias
- Arquitetura de computadores pessoais
- Arquitetura de Computadores Paralelos
- **Sistemas Computacionais:** desempenho e confiabilidade

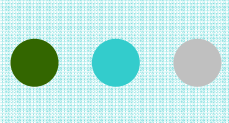


Sistemas Computacionais



Sistemas Computacionais

- O que são **Sistemas Computacionais**?
Essencialmente, são **sistemas** - composição de computadores, softwares e funcionalidades programadas - que são interligadas através de conexões de comunicações para realizarem uma determinada tarefa ou funções.
- As formas de interligação, a flexibilidade e o grau de acoplamento / dependência dos seus componentes, dependem das necessidades e características da **Aplicação**.
- As formas de interligação determinam também a Arquitetura do Sistema: sistemas **internos** (baseados em rede local) e sistemas **externos** (baseados em Internet)



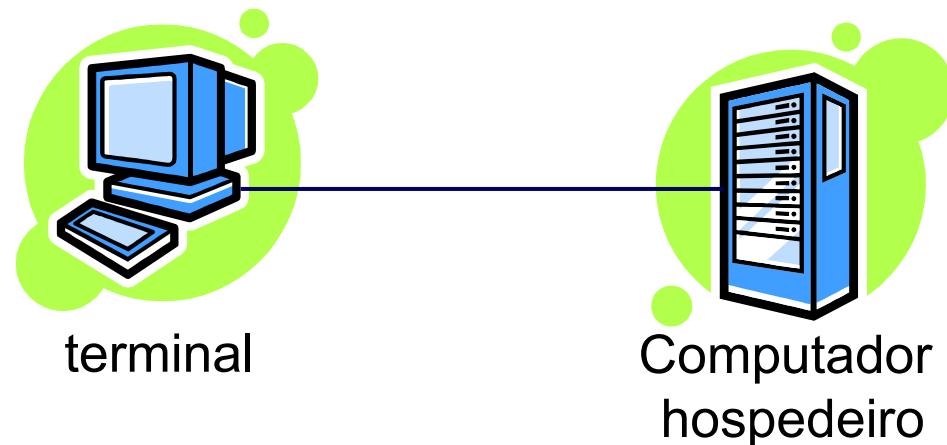
Arquiteturas de Sistemas Distribuídos Internos

- Existem diversas formas de interconexão para compor os sistemas computacionais em rede local. A rede local (Ethernet) é o elemento de comunicação comum, mas a composição e função dos elementos conectados e a hierarquia destas composições, criam diversas arquiteturas.
- Arquiteturas para a conexão em sistemas Locais:
 - Terminal-para-hospedeiro
 - Servidor de Dados
 - Cliente / servidor
 - Intranet

Obs.: Essas arquiteturas também são conhecidas como arquiteturas de rede local.

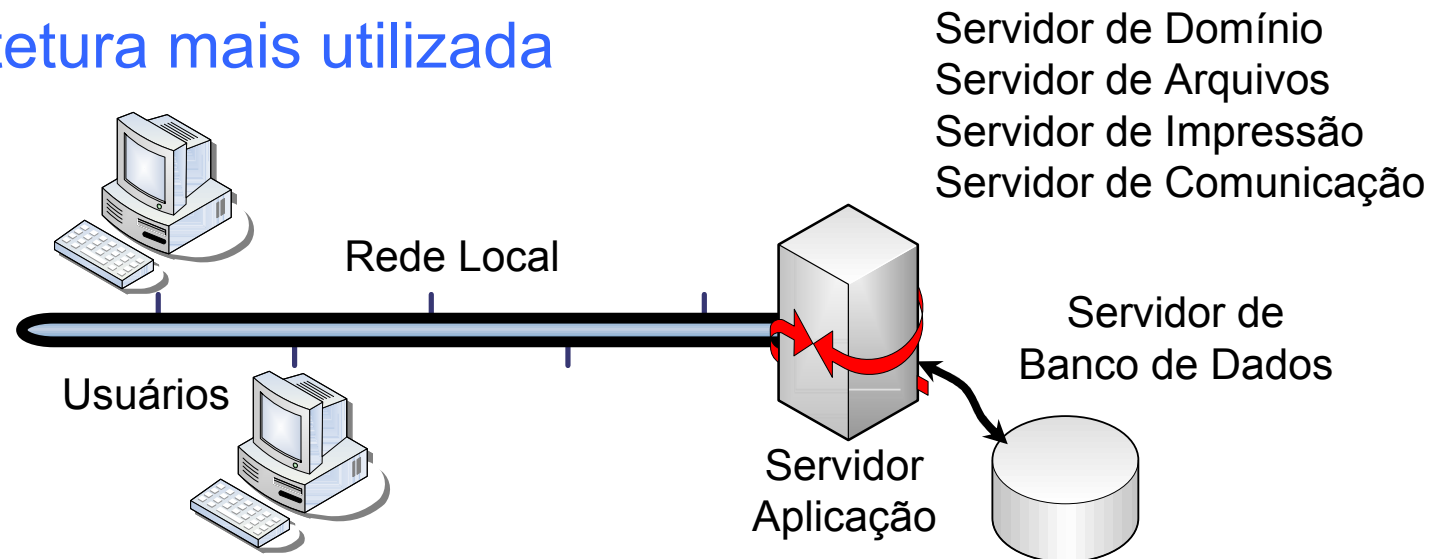
Terminal-para-hospedeiro

- Tanto aplicação quanto BD (Banco de dados) residem em um computador hospedeiro
 - Usuário interage com o hospedeiro por um terminal sem capacidade de processamento da aplicação. O terminal é conhecido também como *thin-client*.
 - \Rightarrow processamento é realizado pelo hospedeiro



Cliente-servidor

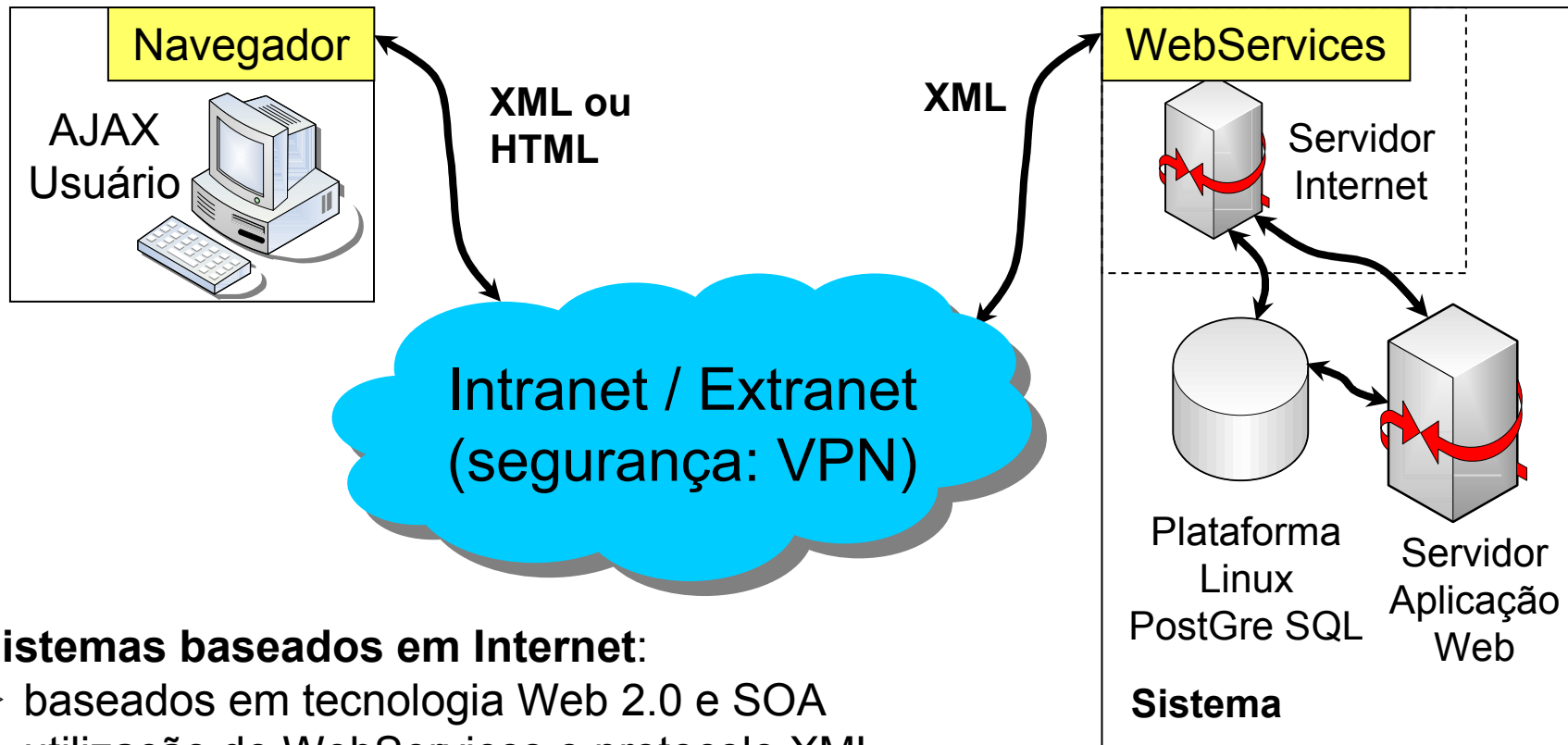
- Composto de diversas plataformas (servidores) com funcionalidades específicas, incluindo aplicações
 - Podem ser acessados por vários clientes na rede
 - Frequentemente PCs de usuários
 - Requisitam serviços aos servidores
 - Arquitetura mais utilizada





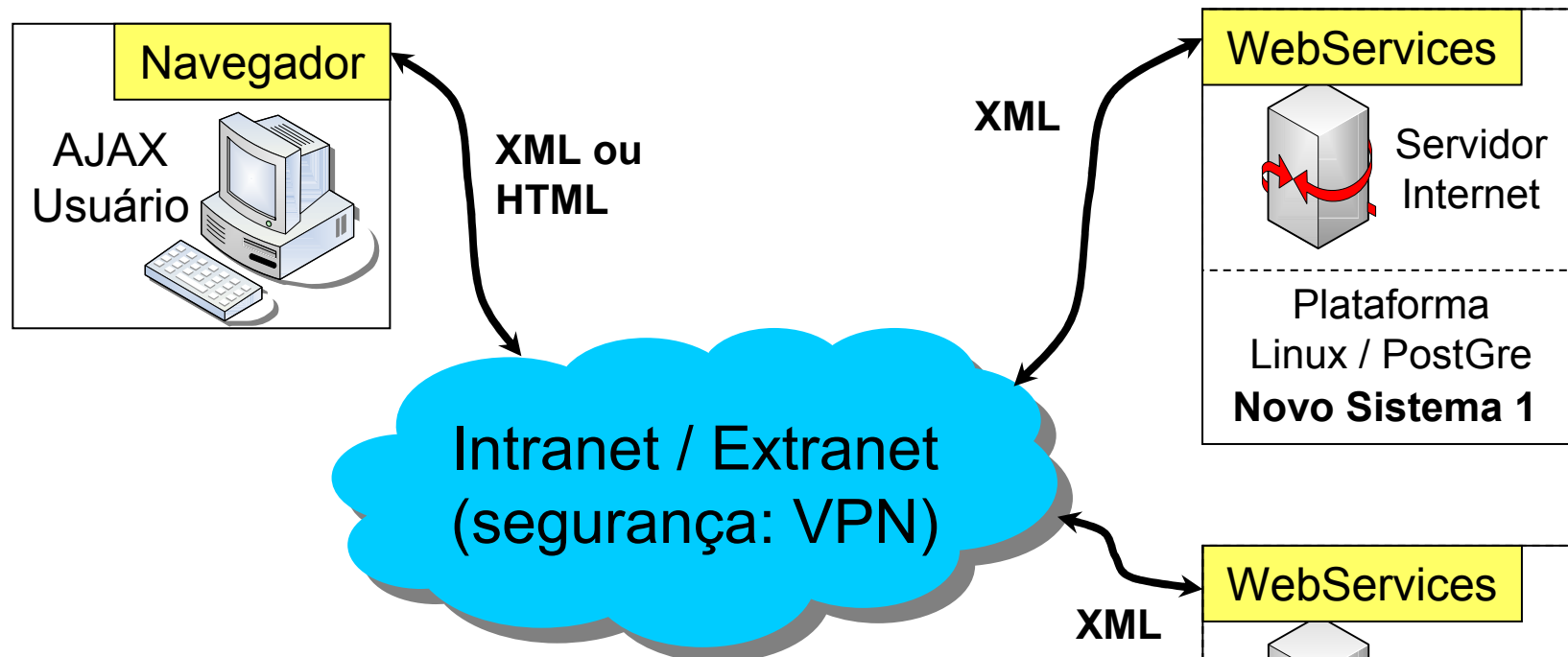
Arquiteturas de Sistemas Distribuídos Externos

- Permitem a conexão de sistemas distribuídos geograficamente. Normalmente baseados na conexão física da Internet pública. Em casos específicos, podem ser utilizadas conexões de comunicação privada para assegurar a segurança (Frame-relay, X25, Linha Privada, etc).
- Arquiteturas para a conexão em sistemas externos:
Entre redes: Internet
 - Intranet / Extranet
 - Cloud computing (computação em nuvem)
- Tecnologias Web 2.0: palavras-chave
 - AJAX: Assynchronous Javascript and XML: execução local (no navegador “browser”) de programas em Java, melhorando a interatividade com o usuário e reduzindo o tráfego de dados
 - XML (eXtended Markup Language: sucessor do HTML)
 - SOA: Service Oriented Architecture
 - Webservices: aplicações com interfaces padronizadas de acesso a serviços pela Internet



Sistemas baseados em Internet:

- baseados em tecnologia Web 2.0 e SOA
- utilização de WebServices e protocolo XML
- comunicação entre sistemas, e entre sistema e usuários: mesma interface XML
- utilização de AJAX e Hibernate no navegador (browser) do usuário: permite execução de aplicativos locais, com interface mais dinâmica e sofisticada.
Exemplo: editor de texto e planilha do Google
- suporte à aplicações móveis, com utilização da tecnologia 3G



Integração de Sistemas:

- todos sistemas utilizam tecnologias Web 2.0 (WebServices e XML), integrados via XML
- introdução gradual de novos sistemas, sem afetar os existentes
- independente de plataforma: basta ter WebServices e XML
- transparente para o usuário, não necessita de atualização



Desempenho de Sistemas



Desempenho: Processamento x Tempo

Desempenho de um Sistema Computacional avalia a capacidade de execução de tarefas de processamento. Existem três parâmetros principais:

- capacidade (estático): determina o volume de dados ou quantidade de tarefas que um dado sistema computacional é capaz de processar em um determinado tempo. Capacidade instalada ou projetada.
- desempenho (dinâmico): determina o volume de processamento sustentado do sistema computacional, conhecido também como desempenho médio.
- tempo de resposta: tempo entre um estímulo (entrada ou disparo em um tempo pré-determinado) e a saída do resultado esperado.



Parâmetros da Capacidade Computacional

A Capacidade é normalmente uma medida absoluta do desempenho do Sistema Computacional, desvinculado da aplicação. Alguns exemplos:

➤ Velocidade de Processamento:

MIPS: Milhões de Instruções Por Segundo

MFLOPS: Milhões de Operações de Ponto Flutuante Por Segundo

- muito utilizado em especificação de capacidade de Supercomputadores
- GFLOPS (Giga); TFLOPS (Tera)

➤ Velocidade de transferência de dados: medida de desempenho de sistemas de comunicação: Gbps (Giga Bits Por Segundo); GBps (Giga Bytes Por Segundo). Outros conceitos importantes: **Overhead** (parte da banda de comunicação utilizado pelo protocolo). Normalmente reduz a capacidade nominal em 50%. **Latência:** tempo até o início da transmissão dos dados

➤ Capacidade da Memória ou do Sistema de Armazenamento (Discos): estabelecidos em GB (Giga Bytes) e TB (Tera Bytes)



BENCHMARK: comparação de Desempenho e Capacidade

Como no caso de processadores e computadores isolados, o Benchmark é utilizado para **comparação** entre **Sistemas Computacionais**, baseado na execução de grupos de aplicativos padronizados. Neste caso, os aplicativos do Benchmark são selecionados do universo das funcionalidades Cliente-Servidor e aplicações Internet.

- Sistemas Corporativos (ERP: Enterprise Resource Planning)
- Sistemas de Controle de Manufatura (MRP: Manufacturing Resource Planning)
- Ferramentas de BI (Business Intelligence)
- Sistemas de Automação Comercial
- Sistemas de Comércio Eletrônico pela Internet (e-Commerce)
- Outros, com uso intensivo de transações em Banco de Dados

Realizados por empresas independentes e auditorias.



Tempo de Resposta

Em **sistemas computacionais**, o tempo de resposta pode ser caracterizado como o tempo que leva para um sistema reagir a determinada entrada.

Mas em **sistemas de processamento de dados** o tempo de resposta percebido pelo usuário é o tempo entre o instante em que um operador entra com um pedido de resposta de um computador e o instante em que o primeiro caractere de resposta é recebido no terminal.

Em **sistemas de tempo real** o tempo de resposta de uma tarefa ou segmento é definido como o tempo decorrido entre o envio (tempo quando a tarefa está pronta para executar) para o tempo quando terminar o seu trabalho (uma expedição).



Confiabilidade de Sistemas



CONCEITOS DA TEORIA DE CONFIABILIDADE

Confiabilidade de um sistema de computador é a qualidade do serviço oferecido, tal que se possa confiar justificadamente neste serviço. O serviço oferecido por um sistema é o comportamento do sistema tal como é percebido pelo seu usuário.

$R(t)$: É a probabilidade de que o sistema desempenhe as suas funções com sucesso, de acordo com as especificações, por um dado período de tempo, $[0, t]$. $R(t)$ é denominada função **confiabilidade**.

$\lambda(t)$: É a função taxa de falhas, que exprime a densidade de probabilidade de falhas no intervalo $[t, t+dt]$, condicionada ao sucesso no intervalo $[0, t]$.

$A(t)$: É a probabilidade de que o sistema desempenhe as suas funções com sucesso, de acordo com as especificações, durante todo o período de operação. $A(t)$ é denominada função **disponibilidade**.

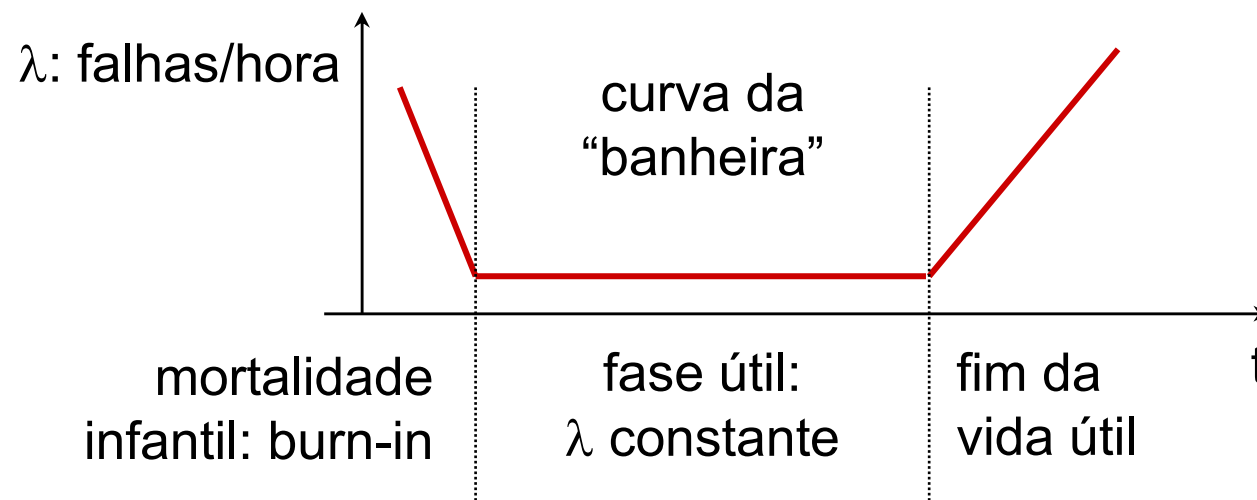
$\mu(t)$: É a função taxa de reparos, que exprime a densidade de probabilidade de reparo do sistema no intervalo $[t, t+dt]$.



COMPORTAMENTO DA FALHA NO TEMPO

Taxa de Falhas de componentes no tempo:

- falha inicial: mortalidade infantil, devidos a falhas no processo de fabricação, garantia da qualidade
- taxa de falhas constante (estatisticamente) durante a vida útil
- fim de vida útil: desgaste



Relação entre a Taxa de Falhas e a Confiabilidade

Confiabilidade:

$$R(t) = e^{-\lambda(t)}$$



Índices de Confiabilidade, Disponibilidade e Manutenibilidade

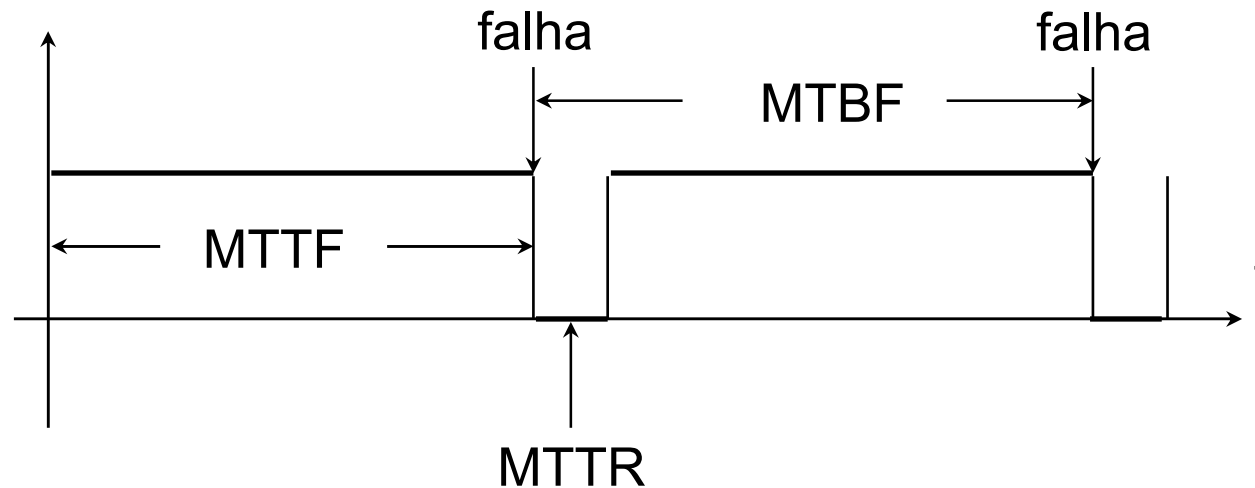
Os índices mais utilizados na especificação e medição da **confiabilidade** são:

MTTF: Mean Time to Fail: tempo até a primeira falha

MTBF: Mean Time Between Failures: intervalo de tempo médio entre as falhas

O índice da Manutenibilidade (capacidade de reparar)

MTTR: Mean Time to Repair: tempo médio para reparar uma falha



$$MTTF = 1/\lambda$$

$$MTTR = 1/\mu$$

Disponibilidade:

$$A = \frac{MTTF}{MTTF + MTTR}$$

Razão entre o serviço oferecido e o tempo fora do ar devido ao reparo



TRATAMENTO DE FALHAS

Etapas do Tratamento de Falhas:

- Detecção da Falha (temporária ou permanente)
- Diagnóstico da Falha (é diferente da Detecção)
- Isolação da Falha (contenção da Falha)
- Sinalização: para acionamento da manutenção
- Operação degradada, reconfiguração, até o reparo
- Reparo (on-line, off-line)
- Recuperação: retorno à operação normal

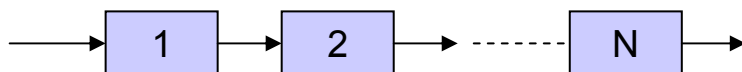


TRATAMENTO DE FALHAS

Algumas Técnicas:

- proteção contra distúrbios externos: alimentação, descargas, intrusão
- sensores detectores de estados anormais (temperatura, sobretensão, etc)
- watch-dog timer, implementado por hardware e controlado por software
- auto-teste periódico de partes do sistema, dados e programas
- auto-diagnóstico: verificação de ajustes e calibrações
- temporização de eventos dinâmicos: comunicações, entradas
- verificação dos limites de validade (dados e parâmetros)
- código detectores e corretores de erro
- tempo de vida de informações e de estados (temporização da validade)
- marcadores de passagem (entrada e saída de rotinas)
- estabelecimento de valores e saídas seguras
- votação / comparação de dados de saída críticos

SISTEMAS SÉRIE E PARALELO PARA CÁLCULO DA CONFIABILIDADE



Sistema Série:

Confiabilidade: $R_T = R_1 \times R_2 \times \dots \times R_N$

se todos sistemas iguais: R

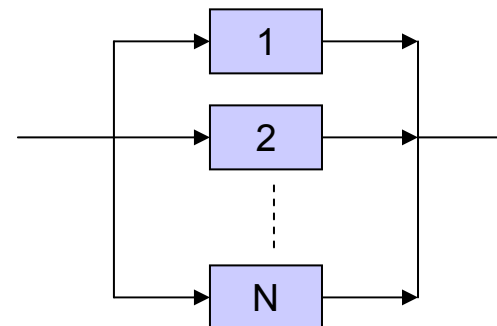
então: $R_T = R^N$

$$MTTF = 1 / N\lambda$$

Exemplo:

Sistema composto por 10 subsistemas em série, com 0,99 de confiabilidade cada:

então: $R_T = 0,99^{10} = 0,90$



Sistema Paralelo:

Confiabilidade: $R_T = 1 - (1 - R_1) \times (1 - R_2) \times \dots \times (1 - R_N)$

se todos sistemas iguais: R

então: $R_T = 1 - (1 - R)^N$

Exemplo:

Sistema composto por 10 subsistemas redundantes, com confiabilidade individual de 0,75:

então: $R_T = 1 - (1 - 0,75)^{10} = 0,9999$

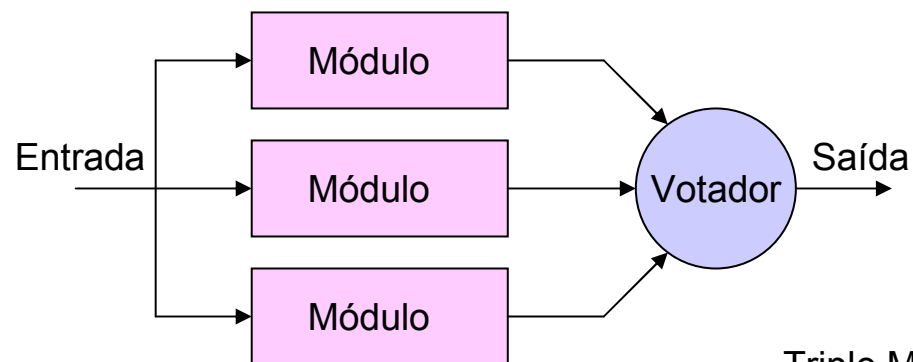


ARQUITETURA DE COMPUTADORES TOLERANTES A FALHA

Redundância Estática

Redundância Estática:

- todos módulos ativos, e executando a mesma função
- cobertura imediata da falha
- não necessita detectar a falha para atuar
- elevada disponibilidade e segurança (Votador)
- necessita sincronismo



Triple Modular Redundancy: TMR

Confiabilidade:

$$R_{\text{TMR}} = \text{Prob. 3 módulos funcionando} \\ + \text{Prob. 2 módulos funcionando}$$

NMR: N-Modular Redundancy



Arquiteturas Paralelas: Balanceamento

