

Engenharia de Software I




Prof. Salatiel Luz Marinho




e-mail: salatiel.marinho@docente.unip.br

Papel Evolutivo do Software

- Podemos dizer que o software atualmente pode ser visto como tendo dois papéis: Software Produto e software como veículo de entrega do produto, um pouco confuso no começo, porém tudo ficará mais claro quando virmos o que é o software produto.
 - No papel evolutivo do software, podemos dizer que a importância do mesmo vem passando por mudanças significativas há pelo menos 50 anos.
- 


Papel Evolutivo do Software

- **1970 – 1980:** Período visto como o surgimento do software, foi visto pela sociedade e pelos pensadores da época como uma nova revolução industrial, outros pensadores disseram que essa era a terceira onda de mudanças da história humana.
 - Transformação de uma sociedade industrial em uma sociedade da informação;
 - A informação e o conhecimento como centro do poder.
- 


Papel Evolutivo do Software

- **1990:** Houve o que os pensadores descreveram como a mudança do poder, agora com os computadores houve uma democratização do conhecimento(antigamente o conhecimento estava restrito a um núcleo de pessoas que se mantinham no poder, principalmente a economia militar).
- Surgiram uma série de livros anti-computadores, que enfatizavam diversas preocupações, mas não colocavam os benefícios.
- Veio na segunda metade dos anos 90 a ascensão e a ressurreição dos programadores.

Papel Evolutivo do Software

- **2000:** Comentários a respeito da bomba relógio - O Bug do Milênio, aonde todos os autores escreveram sobre a parada de todos os sistemas, o fim do mundo, baseado em sistemas.
 - **2001 até dias atuais:** A indústria do software tornou-se e vem se afirmando como o fator dominante na economia do mundo industrializado.
- 

Software / Software Produto

- Para que fique mais fácil compreender o que realmente é software é importante conhecer suas características, pois são essas que o torna, diferente de todas as outras coisas que podem ser construídas. E essas características são também relativamente diferentes das encontradas nos hardwares.
- 

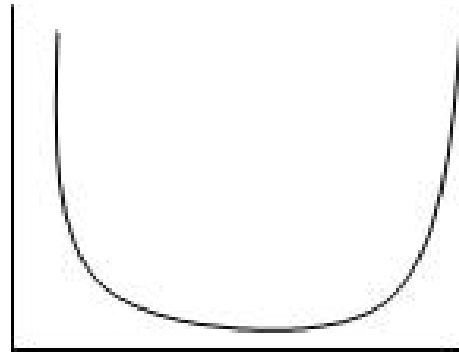
O software é desenvolvido por um processo de Engenharia e não é fabricado no sentido clássico

- O que significa o sentido clássico da produção, seria imaginar uma linha de produção de um produto qualquer, por exemplo de um carro, aonde em cada etapa, são encaixados peças e o produto vai caminhando pela fábrica, até que tudo fique pronto e passe por uma aprovação de qualidade, o software diferente disso não tem peças que vão sendo agregadas e não caminha por uma linha de produção, sua fabricação independe de máquinas e depende muito mais das pessoas e sua especialização no assunto, sua qualidade envolve outros indicadores e seus problemas apesar de serem mais fáceis de solucionar são de outras proporções.

O software não se desgasta

- Para verificar que o software não se desgasta é interessante fazer uma análise da curva de falhas entre o hardware e o software.

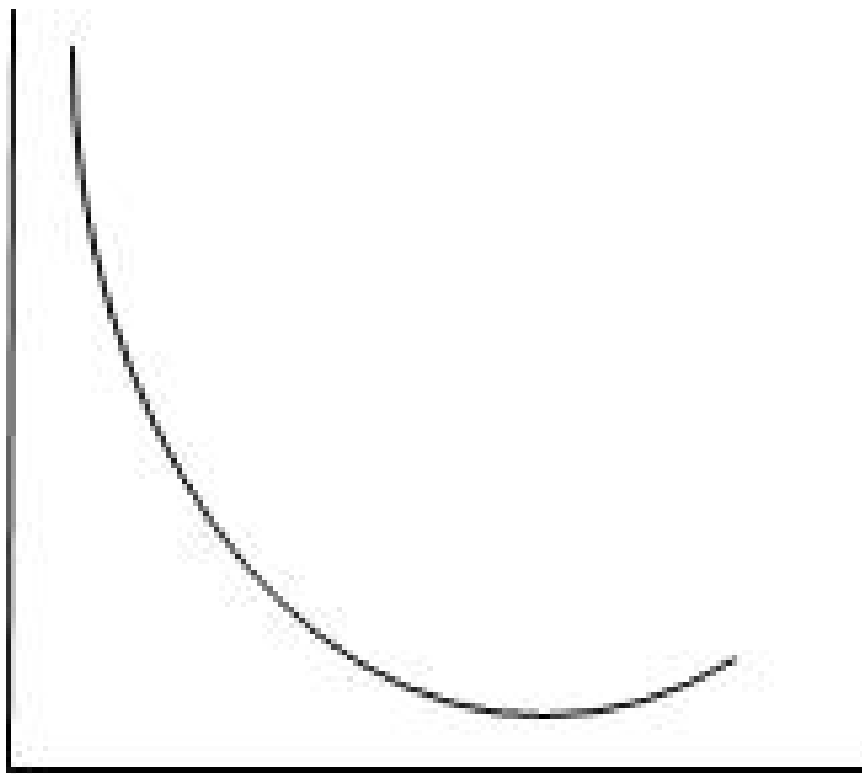
Curva de falhas do Hardware




O software não se desgasta

- O gráfico conhecido como o gráfico da banheira, representa a curva de desgastes do hardware, notem que a quantidade de falha de um hardware é muito grande no momento de sua criação podendo acontecer o que chamamos de mortalidade infantil do hardware, depois que essa taxa de erros se estabiliza ela permanece um tempo inalterada, porém a medida que o tempo passa e os males ambientais (pó, umidade, calor) vão ocorrendo as falhas do hardware voltam a ocorrer e o mesmo tem então um índice de desgaste grande e precisa ser retirado de uso.

Curvas de falha do Software




Curvas de falha do Software

- A curva ideal do software, seria uma curva que começaria com grandes quantidades de falhas, pois o software estaria no seu surgimento, portanto, conteria muitos acertos e ajustes, e correções de bug e depois essa curva iria se achatando e não mais subiria ficando assim o software em pleno uso. Porém essa curva não é ideal ela contém vários picos aonde os erros aumentam e diminuem, é conhecida como curva dentada, isso ocorre porque a cada alteração que é solicitada no software o mesmo sofre alterações e é passível de novos defeitos, nem sempre essa curva de erros chega a se estabilizar e novas demandas são necessárias, fazendo com o que o software não se desgaste mas sim se deteriore com o tempo.
- 

A produção é sobre encomenda

- Apesar de a algum tempo muitos software serem produzidos empacotados para diversos públicos alvos e a produção de componentes ser uma vertente em crescimento, a grande maioria dos softwares ainda são feitos sobre encomenda, ou senão se transformam para atender as necessidades do cliente, diferente de um automóvel ou de um hardware, de um micro-ondas, etc.


Requisitos Funcionais

- Os requisitos funcionais descrevem o que o sistema deve fazer. Eles especificam as funcionalidades e comportamentos que o software deve ter para atender às necessidades dos usuários. Exemplos incluem:
 - Adicionar novas tarefas em um aplicativo de gerenciamento de tarefas.
 - Processar pagamentos em um sistema de e-commerce.
 - Gerenciar perfis de usuários em uma rede social.
- 

Requisitos Não Funcionais

- Os requisitos não funcionais descrevem como o sistema deve operar. Eles se referem à qualidade e às restrições do sistema, como desempenho, segurança e usabilidade. Exemplos incluem:
 - Desempenho: O sistema deve ser capaz de processar 1000 transações por segundo.
 - Segurança: Os dados dos usuários devem ser criptografados.
 - Usabilidade: A interface deve ser intuitiva e fácil de usar.

Exemplos Práticos

- **Sistema de E-commerce:**
 - **Funcional:** O sistema deve permitir que os usuários adicionem produtos ao carrinho e façam o checkout.
 - **Não Funcional:** O tempo de resposta do sistema deve ser inferior a 2 segundos durante o checkout.
 - **Aplicativo de Rede Social:**
 - **Funcional:** O aplicativo deve permitir que os usuários publiquem fotos e comentem nas postagens.
 - **Não Funcional:** O aplicativo deve estar disponível 99.9% do tempo (alta disponibilidade).
- 

Exemplo - Software




Software Produto

- Atualmente os engenheiros de software estão envolvidos no desenvolvimento de produto de software, ou seja, software que pode ser vendido para seus clientes.
- Existem dois tipos de softwares produtos fundamentais:
- **1- Produtos Genéricos:** São os softwares do tipo stand-alone, produzidos por uma organização de desenvolvimento e vendido no mercado para qualquer cliente que esteja disposto a comprar.
- **2- Produtos sob encomenda:** São os sistemas encomendados por um cliente. O software é desenvolvido especialmente para aquele cliente com as características voltadas para apenas um cliente.


A Natureza Mutável do Software

- Hoje em dia sete amplas categorias de softwares de computadores apresentam desafios contínuos para os engenheiros de software, a partir dessas verificamos o porquê de dizer a natureza mutável do software.
- **Software de Sistemas:** Software de sistema é uma coleção de programas escritos para servir a outros programas. A área de software de sistemas é caracterizada por interação intensa com o hardware, uso intenso de usuários, possui estrutura de dados complexas (**Exemplos: SO, Sistemas de rede**)
- **Software de Aplicação:** O software de aplicação consiste em programas isolados que resolvem uma necessidade específica do negócio da empresa (**Exemplo: Software de sistema de tempo real, de pontos de vendas**).


A Natureza Mutável do Software

- **Softwares Científicos ou de Engenharia:** Antigamente esses eram caracterizados por algoritmos que processavam grandes quantidades de números. Hoje em dia essas aplicações estão mais próximas de software de sistemas. Exemplo (software de controle de riscos – Astronômicos, da NASA).
 - **Softwares Embutidos:** O software embutido reside dentro de um produto ou sistema e é usado para implementar e controlar características e funções para o usuário final e para o próprio sistema. Os softwares embutidos podem realizar tarefas muito limitadas e também tarefas da mais alta capacidade de um software. (controles remotos).
 - **Software para linha de produto:** Projetado para fornecer uma capacidade específica a ser usada por muitos clientes diferentes, esse software pode focar para um mercado limitado e especial ou dirigir-se ao mercado de consumo em massa. (pacote Office, Windows)
- 

A Natureza Mutável do Software

- **Aplicações da WEB:** Dos mais variados tipos.
 - **Software para inteligência artificial:** O softwares de IA faz uso de algoritmos não numéricos para resolver problemas complexos que não são passíveis de computação e analise direta.
- 

Trabalho

- **Realizar o levantamento da construção de um software para uma lanchonete da UNIP;**
 - **Categorizar o software conforme apresentado em sala;**
 - **Listar principais vantagens e desvantagens do software;**
 - **Se a lanchonete já tiver um software realizar um estudo do mesmo, categorizar o software e listar as principais vantagens e desvantagens.**
- 

Dúvidas

