* **Aula 1 - 31/10/2016**
* Disciplina Teórica
* Network File System
  + Pasta do Windows mapeadas por outras maquinas
  + Não sabemos o que há por baixo
  + Se o servidor não cair, você não percebe
  + NFS de origem?
* Email antigamente x Email na Web
  + Grande disponibilidade
  + Rápido
  + Fácil acesso
  + Não é relevante a localização
* Computação distribuída: Em um ambiente de rede, os serviços sãos usados para fornecer serviços abstratos e possantes de forma imperceptível.
  + Internet dá suporte a isso; Fornece padronização.
  + Pilha OSI é muito completa, a Internet só usa parte dela para ser mais prática.
  + A computação distribuída foi facilitada pela internet.
* **Aula 2 - 07/11/2016**
* Área que parte de vários conceitos.
* Antes dos computadores as pessoas conseguiam sobrevier.
  + Registros sob a forma de papel.
  + Transferência física de informações.
  + A informação é naturalmente distribuída, isso a torna difícil de manter consistente,
* Evolução da computação:
  + Década de 60:
    - Mainframes comerciais.
    - Difícil trabalhar com muitos dados em cima de uma empresa pois estava mais sujeito a ocorrer erros.
    - MF, informação manipulada de forma mais rápida e mais precisa.
    - Processamento em modo batch.
    - Cartões perfurados executados sequencialmente.
    - Interface com o usuário era difícil, na verdade inexistente.
    - Podia gerar grande tempo de espera.
    - Com a popularização as coisas começaram a demorar ainda mais.
    - Duas insatisfações: interface e demora.
  + Década de 70:
    - Surgimento dos terminais burros. Dava maior interação ao processo.
    - Mudou o tipo de processamento: criou os conceitos de time share e chaveamento entre processos.
  + Década de 80:
    - Surgimento dos PCS. Distribui a execução dos Mainframes para os PCS.
    - Ambiente hostil?
    - Computadores mudam muito rápido (hardware e software)
    - Não havia HD.
    - Disquetes para transferir programas para a memória.
  + Década de 90:
    - Novos vírus e HDS.
    - Processamento distribuído pode gerar inconsistências.
    - Surgimento da conexão por rede, compartilhamento de recursos e informações.
    - Mobilidade, surgimento dos PDAS.
    - Caos para consistência.
    - Sistemas distribuídos organizam as coisas.
* **Aula 3 - 09/11/2016**
* Avanços tecnológicos ampliam a problemática da cópia.
* Cria problema de inconsistência.
* Computação veio após as redes, elas as explora.
* Software teve seu desenvolvimento separado da computação distribuída.
* Modelo cliente X servidor
  + Arquitetura de software é um conjunto de correntes com papeis definidos e algumas restrições.
  + Cliente servidor evoluiu com o tempo.
* Ondas dos Sistemas Distribuídos.
* 1ª Onda – Servidor de Arquivos
  + Separação entre dados e processos.
  + Aplicações compartilham dados de modo consistente.
  + Servidor bloqueia arquivos em uso.
  + Programador tinha total controle sobre arquivos e aplicação.
  + Passar arquivos grandes virou proibitivo.
  + Percebeu-se a existência do “CRUD”
* 2ª Onda – Banco de dados
  + Dá capacidade ao servidor de arquivos de operar sobre os dados.
  + Melhor granularidade de acesso.
  + Modelo relacional. Problema-> Relacionamentos distanciavam as tabelas, não existia joins.
  + SQL – Evitar consultas constantes.
  + A popularização provoca a renovação.
* 3ª Onda – Banco de dados distribuídos.
  + Banco de dados onde está abstraída a localização
  + ?
  + ?
  + ?
* **Aula 4 - 16/11/2016**
* 4ª Onda – Mudança de paradigma.
  + Com o passar do tempo o software precisou de mais coisas.
  + Engenharia de software surgiu. Trazendo aspectos da engenharia comum para a produção de software com a intenção de conseguir os mesmos benefícios: preço e tempo. Problema ->software é intangível.
  + Paradigma: forma de pensar o problema.
  + Paradigma estruturado era muito predominanante.
  + Paradigma de entidade e relacionamentos.
  + Engenharia de software pensava primeiro nos dados.
  + Na visão estruturada: Tendência de separar dados e processos.
  + Existem dados usados por mais de 1 processo.
* 5ª Onda – Surgimento do POO
  + Conceito de classe.
  + Objetos são entidades de memória.
  + Encapsulamento garante o desacoplamento e a coesão.
  + Atributps = dados
  + Métodos = processos.
  + Persistência é necessária.
  + Herança garante a construção de u software mais simples.
  + Variáveis estáticas na rede.
  + Herança não combina com distribuição.
  + Distribuir um ramo de herança, diferente granularidade, distribuição de componentes. Paradigma Orientado a componentes.
  + Hoje em dia tudo convive junto.
* **Aula 5 - 21/11/2016**
* Empresas precisa te um parque de TI bom para dar suporte ao seu negócio; começa a ficar inviável com o tamanho.
* Empresas alugam equipamentos e sérvios de uma empresa.
* É mais justo, pois você só paga o que usa. Ex: Amazon.
* Nuvem tem problema com segurança.
* Nuvens hibridas: parte dentro e parte fora. Ex: bancos
* Virtualização
  + Não trabalhar com hardware real, mas sim virtuais. 1 máquina virtual pode ser várias máquinas reais.
* Colocation:
  + Usuário gerencia fisicamente.
* Elasticidade:
  + Aumentar/diminuir oferta do serviço sem prejudicar o usuário, ajudada pela virtualização.
* IOT
  + Conexão de redes -> Conexão de pessoas -> Conexão de coisas
  + Conecta tudo na nuvem, cria um tráfego imenso e passa a ser inviável.
  + Ideal seria ter um serviço de filtro local.
  + Conceito de névoa: “nuvem” mais próxima do usuário
  + Orvalho.
  + Níveis de processamento.
* **Aula 6 - 28/11/2016**
* Computação distribuidasurgiu para simplificar o uso de sistemas distribuídos.
* Nem tudo centralizado não é distribuído e nem tudo distribuído é sistema distribuído.
* Sistema distribuído é um conjunto independente e autônomo de computadores que aparecem para o usuário como um único computador. Conjunto formado por hardware e software que aparecem omo um único sistema computacional.
* Para ter distribuição é preciso ter software.
* Hardware.
  + Vários pontos de processamento.
  + Taxonomia de Flynn
    - Várias CPUS e várias memórias.
      * Depende da memória
      * Se as memórias para de funcionar o sistema para.
      * Existe a possibilidade de processos, existe um recurso restritivo. Acoplamento grande com a memória. Sistema fortemente acoplado.
    - Várias CPUS junto com memórias.
      * Sistema fracamente acoplado.
      * Unidades independentes.
      * Sistemas distribuídos usam autonomia dos fracamente acoplados para atingir algumas propriedades.
* Rede:
  + O sistema distribuído precisa de uma rede.
  + Independência e autonomia:
  + Características importantes
    - Confiabilidade /disponibilidade
      * Garantia de que o sistema estará disponível quando você precisar dele.
      * Replicação: Consegue via software, mas precisa de um hardware.
    - Escalabilidade
      * Consegue de modo flexível absorver/remover recursos.
      * Escalabilidade de um sistema centralizado é bem menor que de um sistema distribuído.
    - Segurança
      * Sistema espalhado é difícil de controlar.
      * Uso de rede piora isso.
    - Transparências:
      * Fazer com que a rede pareça apenas 1 computador. Caro!!!
      * Acesso
        + Manipulação transparente.
      * Localização.
        + Localização transparente.
      * Migração.
        + Migrar sem perceber. Ex: Base de dados redirecionada.
      * Concorrência
        + Sistema resolve problemas de concorrência automaticamente
      * Falha.
        + Usuário não percebe falhas. Complexo.
      * Desempenho.
      * Replicação
        + Para garantir disponibilidade,
      * Paralelismo
        + Quebra de código.
      * Escala
        + Incluir recurso de forma automática.
      * Não da pra fazer tudo.
      * Paralelo != distribuído
* **Aula 7 – 30/11/2016**
* Transparências são complicadas, mas precisa-se faze a rede ser transparente.
* Protocolos viabilizam a comunicação na rede.
* Necessidade de padronização ->OSI->Sistema aberto/público.
* OSI
  + Quebra um problema complexo e problemas menores.
  + Pilha de protocolos de tipos diferentes de problemas.
    - Camada física – Meio físico
    - Camada de enlace.
    - Camada de rede – endereçamento.
    - Camada de transporte – Conexão fim a fim, controle de fluxo e congestionamento.
    - Camada de seção.
    - Camada de Apresentação – XDR
    - Camada de Aplicação.
* TCP/IP
  + Camadas
    - Camada física + Enlace (Acesso ao meio)
    - Rede – IP
    - Transporte – TCP/UDP
    - Aplicação
  + É necessário entender a rede
  + Socket = IP+Porta
  + Aplicação de rede: Cliente/Servidor.
* **Aula 8 – 05/12/2016**
* Visão do usuário
  + Vê todos os computadores como 1 só.
* As transparências escondem características da rede, pode ser difícil ter um sistema com todas as transparências.
* Objetivo da computação distribuída não é odesempenho, mas sim permitir a colaboração.
* Sistemas Distribuidos/ Cliente servidor?
* Formas de implementar:
  + Implementar as características de distribuição em um SO.
  + Implementação 1: Todos tinham que tero mesmo SO, pois não havia padronização.
  + Implementação 2:Qualquer SO, mas usava-se o mesmo software plataforma.
  + Implementação 2: Com a POO criou-se o conceito de Middleware, camada de software acima do SO com funcionalidades homogêneas. Um middleware padronizado para todos.
* Implementando transparência de localização e acesso.
  + Localização:
    - Endereçamento:
      * IP/Porta explicitamente escritos no código, não tem transparência.
      * Broadcast é transparente, mas tem problemas. Na internet é inviável,mas á viável em rede local.
      * Servidor de nomes, é uma solução para a Internet. Ponto de falha minimizado com a réplica. Diferença DNS e Servidor de nomes?
      * Cliente/servidor permite a transparência de localização através do broadcast e servidor de nomes.
* **Aula 9 – 07/11/2016**
* Naturalmente não é possível o cliente/servidor implementar transparência de acesso.
* Acesso:
  + Ver coisas locais e remotas como locais.
  + Chamada de procedimento remoto RPC
  + Sintaxe local, o código não muda só o que está por baixo.
  + Pré-Compilador: Gera stubs, código intermediário. RPC
    - Código de desvio é criado dinamicamente.
    - Stubs empacotam para padronização.