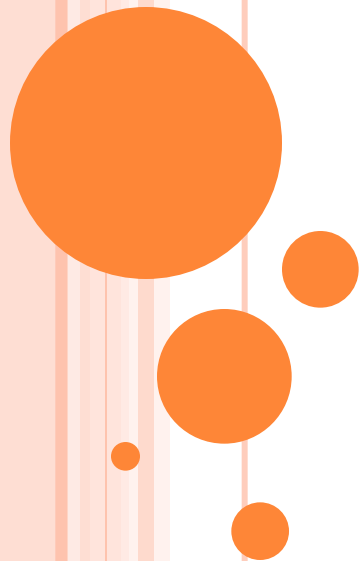


**Universidade Federal do Acre**  
**Centro de Ciências Exatas e Tecnológicas**  
**Curso de Bacharelado em Sistemas de Informação**

**Engenharia de Software II**

# **Gerenciamento de Pessoal**

**Prof. Daricélio Moreira Soares**



# ORGANIZAÇÃO DE EQUIPES E GERENCIAMENTO DE PESSOAL

- Problemas
  - Escolha de pessoal
  - Alocação de atividades (tarefas)
  - Gerenciamento de Pessoal
  - Estilos de trabalho
  - Canais de Comunicação
  - O mito do Pessoa-Mês
  - Dependência entre tarefas em projetos de software
- Organização da Equipe
  - Clássica – programador-chefe
  - Sincronizar-e-Estabilizar
  - Extreme programming



# ESCOLHENDO O PESSOAL

- Capacidade para desempenhar a atividade
- Interesse e motivação no trabalho
- Experiência com
  - aplicações semelhantes
  - ferramentas ou linguagens semelhantes
  - técnicas semelhantes
  - ambiente de desenvolvimento semelhante
- Treinamento
- Capacidade para se comunicar com outras pessoas
- Capacidade para compartilhar responsabilidades
- Habilidades de gerenciamento



# ALOCAÇÃO DE ATIVIDADES PARA MEMBROS DA EQUIPE

- A alocação pessoa-atividade depende de:
  - Habilidade e experiência na atividade
  - Esforço necessário para cada atividade
- Exemplos de atividades de um projeto:
  - Levantamento de requisitos
  - Descrição de cenários
  - Modelagem de Casos de Uso
  - Arquitetura do sistema e do software
  - Projeto do Banco de Dados
  - Codificação e teste de unidades de software
  - Integração e teste de software
  - Manutenção



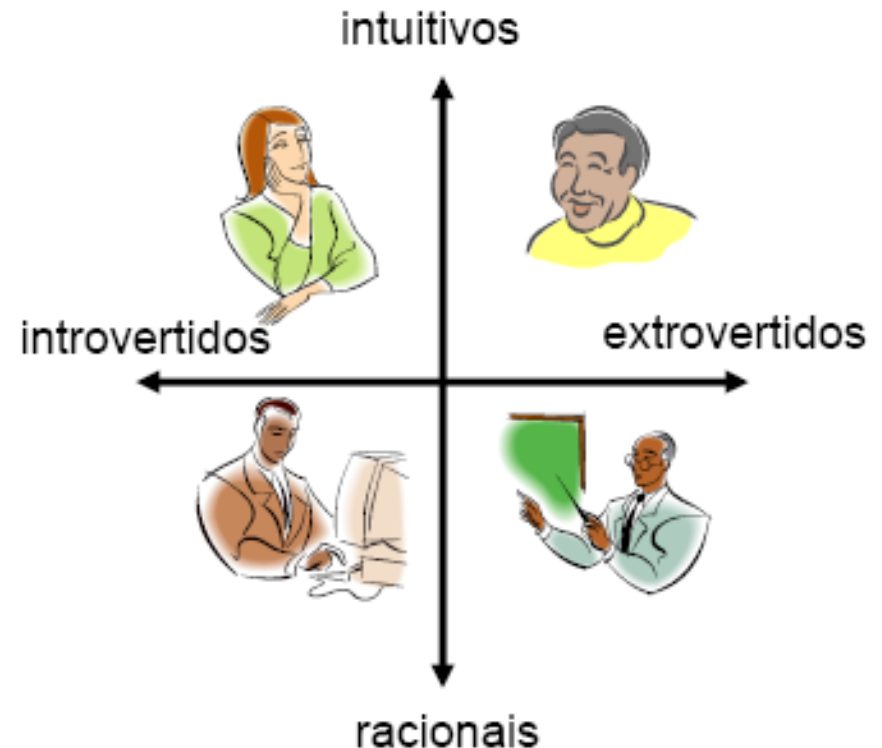
# GERENCIAMENTO DE PESSOAL

- Engenharia de Software é uma atividade cognitiva.
- Limitações cognitivas afetam o processo de de software
- Tarefas da gestão de pessoal:
  - Resolver de problemas - Usar o pessoal disponível
  - Motivar –os membros do time, de acordo com capacidade, experiência e estilo de trabalho
  - Planejar – o que as pessoas têm que fazer
  - Estimar – com que rapidez elas podem produzir
  - Controlar – controlar as atividades das pessoas
  - Organizar – a maneira com as pessoas trabalham



# ESTILOS DE TRABALHO

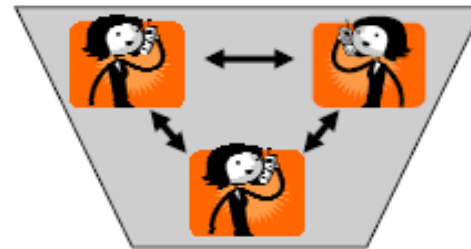
- Extrovertidos
  - tendem a afirmar suas idéias
- Introversos:
  - pedem sugestões
- Intuitivos
  - baseiam suas decisões nos sentimentos
- Racionais
  - baseiam suas decisões em fatos



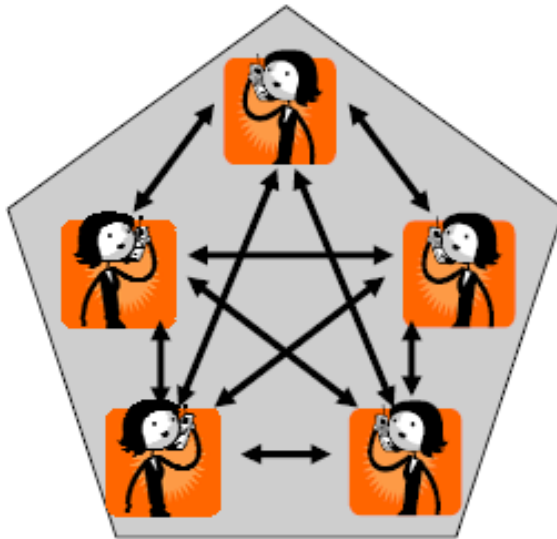
# CANAIS DE COMUNICAÇÃO EM EQUIPES



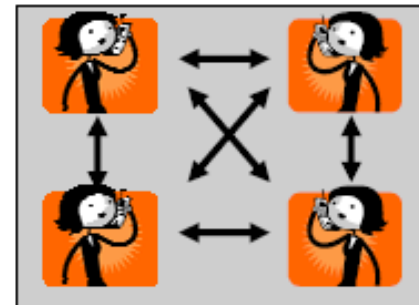
2 pessoas – 1 canal



3 pessoas – 3 canais



5 pessoas – 10 canais

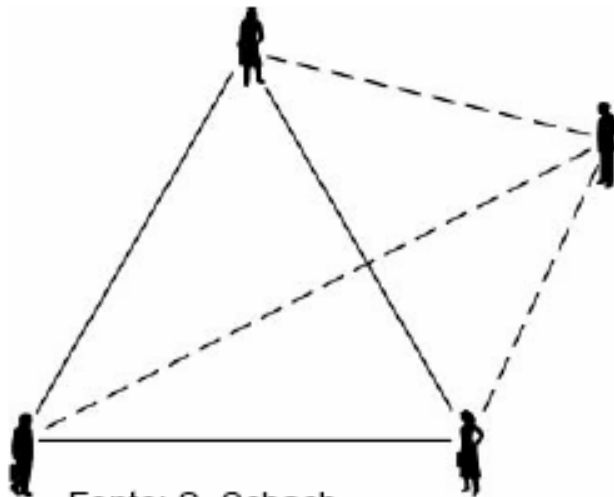


4 pessoas – 6 canais



# CANAIS DE COMUNICAÇÃO - PROBLEMA

- O trabalho em equipe estabelece canais de comunicação entre os membros.
- Os canais dependem:
  - Da dependência entre as tarefas
  - Da organização da equipe.



Fonte: S. Schach

- Exemplo:  
**Existem** programadores trabalhando em um projeto. O prazo final está se aproximando rapidamente. O que fazer?
- Solução:
  - Contratar um novo programador para a equipe
- Mas...
  - Os outros três têm que explicar o que já foi feito e o que ainda precisa ser feito.
  - Isto pode atrasar ainda mais o trabalho.





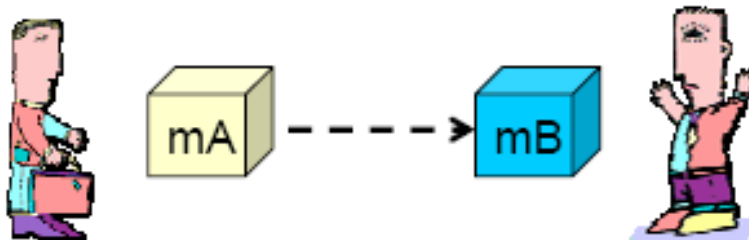
# COMPARTILHAMENTO DE TAREFAS

- Quando é possível compartilhar tarefas?
  - Se um trabalhador pode colher um laranjal em 10 dias, quantos trabalhadores são necessários para colher laranjal em 1 dia?
  - Se uma mulher produz um bebê em 9 meses, quantas mulheres são necessárias para produzir um bebê em um mês?
- Algumas tarefas da engenharia de software podem ser compartilhadas
  - Entrevistas com diversos clientes para levantamento de requisitos
  - Elaboração dos modelos da arquitetura de software
  - Codificação das diferentes unidades (módulos e componentes) de um software
  - Testes de unidades



# DEPENDÊNCIA ENTRE TAREFAS EM PROJETOS DE SOFTWARE

- Exemplo:
  - Fred e João são responsáveis por programar 2 módulos, mA e mB. O módulo mA depende de mB.
- Que problemas podem ocorrer?
  - Fred e João programam mA e ninguém codifica mB.
  - Fred programa mA e João programa mB e...
    - Fred termina a codificação de mA antes de João terminar mB. O módulo mA só poderá ser testado quando mB tiver sido concluído e testado.
    - Ou, quando mA chama mB ele passa 4 parâmetros, mas o código de João para mB requer 5 parâmetros...
    - Ou, a ordem dos parâmetros em mA e mB são diferentes...
    - Ou, os tipos de dados podem ser diferentes



# ORGANIZAÇÃO DA EQUIPE (TIME)

- A organização do equipe depende:
  - da experiência trabalho dos membros da equipe
  - estilo de trabalho dos membros da equipe
  - do número de pessoas na equipe
  - dos estilos de gerenciamento adotado no projeto
  - das atividades a serem realizadas



# TIMES DEMOCRÁTICOS

- Evita a programação egoísta:
  - Programador trata o código como seu
  - Rejeição à achar falhas no seu código
- Times democráticos
  - Reestrutura o ambiente social e valores dos programadores
  - Grupos de 10 programadores “sem-ego”
  - O time todo é dono do código
  - Cometer erros é considerado normal
  - Encoraja o time a encontrar erros
  - Maior produtividade
  - Atuação colaborativa na resolução de problemas difíceis
- Problemas
  - Precisa desabrochar espontaneamente
  - Dificuldade de gerenciamento



# CLÁSSICA – PROGRAMADOR-CHEFE

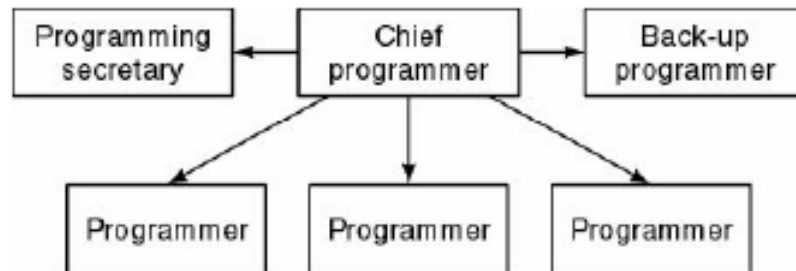
- Idéia básica: equipe de cirurgia médica
  - Cirurgião-chefe, Cirurgião-auxiliar, anestesista, enfermeiros, outros especialistas.
- Aspectos chaves: Hierarquia e Especialidades
- Equipe básica
  - Programador-chefe
    - Capacidade gerencial associada a alta capacidade de programação.
    - Responsável pela arquitetura do software.
    - Aloca as tarefas de programação e controla as interfaces, interações e sincronismos.
  - Programador reserva
    - Substituto do PC. Deve conhecer tanto do projeto quanto ele e ter a mesma capacidade técnica e gerencial.
  - Secretário (bibliotecário)
    - Pessoa ligada ao PC e responsável por toda a organização dos códigos, modelos e documentos produzidos
  - Programadores



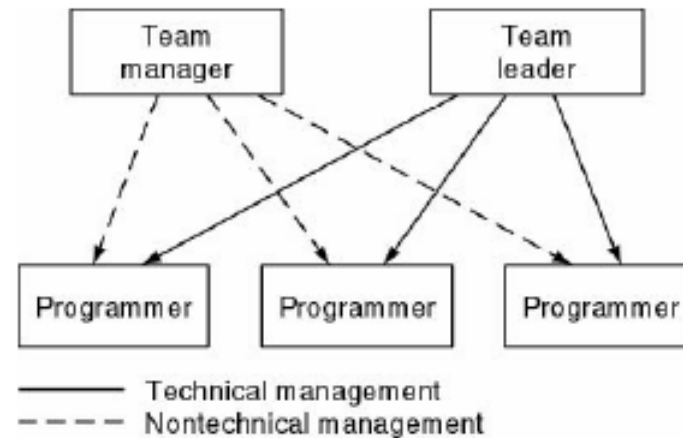
# CLÁSSICA – PROGRAMADOR-CHEFE

- Dificuldades em encontrar numa mesma pessoa as habilidades de gerência e técnicas: arquitetura e programação.
- O programador-reserva deve ter as mesmas capacidades, mas sempre tem um salário menor e não fica muito tempo no cargo.
- O secretário tem que ter conhecimento de biblioteconomia aliado ao de programação e modelagem

Organização clássica



Solução alternativa



# SINCRONIZAR-E-ESTABILIZAR

- Pequenos times em paralelo
  - 3 a 8 programadores e testadores (um-a-um)
  - Para cada time é dada uma especificação que podem ser desenhada como quiser.
- Regras
  - Sincronização diária
  - Os códigos gerados devem ser compatibilizados e integrados.
  - Armazenados numa base de dados em cada sincronização.
- Funciona na Microsoft. Funcionará em todas as empresas?
  - A Microsoft tem um time de jovens e competentes programadores
  - A competitividade entre eles é muito forte. Perfil de nerds.



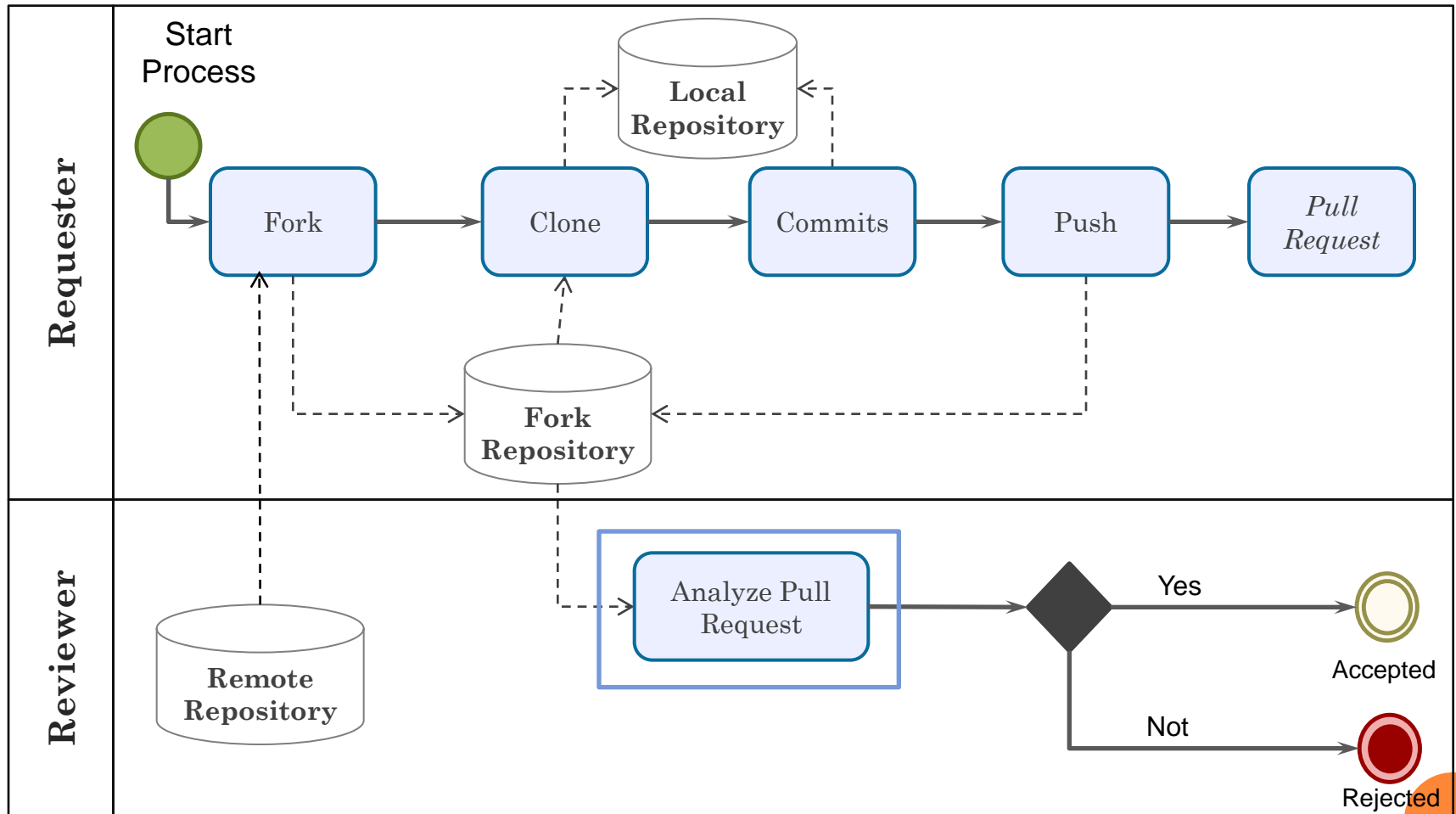
# EXTREME PROGRAMMING

- Características do XP
  - Todo o código é escrito por duplas de programadores (programação em pares).
- Vantagens da programação em pares
  - Casos de testes são elaborados por um dos programadores.
  - O conhecimento sobre o sistema não é perdido se um dos programadores sai do time.
  - Programadores inexperientes podem aprender com o seu colega.
  - O compartilhamento do computador promove uma programação mais impessoal e menos egoísta.



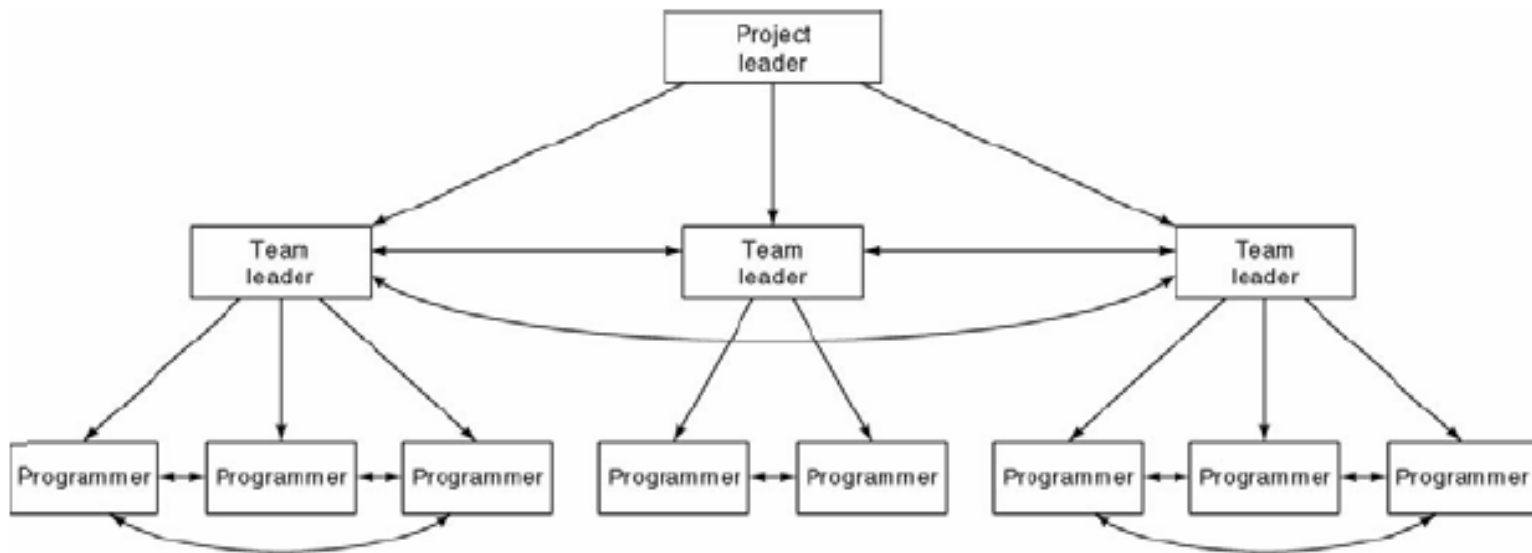


# DESENVOLVIMENTO DISTRIBUÍDO (COLABORATIVO)



# TENDÊNCIAS

- Existe um líder de projeto responsável pelo gerenciamento do projeto e arquitetura do software
- Várias equipes com poucos programadores, lideradas pelos líderes de times.



# DÚVIDAS

