
Extração de Requisitos

Criação: Março 2001

Atualização: Setembro 2008



Referências

I.Sommerville. “Sw Engineering”, 6ª ed, 2001, cap6

P.Jalote. “An Integrated Approach to Sw Engineering”, 2ª ed., 1997, cap3

Ariadne M.B.R.Carvalho, Thelma C.S.Chiossi. “Introdução à Engenharia de Software”. Ed. Unicamp. Cap.2

S.Raghavan, G.Zelesnik, G.Ford. “Lecture Notes on Requirements Elicitation”. Educational Materials. CMU/SEI-94-EM-10, Mar/1994. Obtido na Web em set/2005:

www.sei.cmu.edu/publications/documents/ems/94.em.010.html

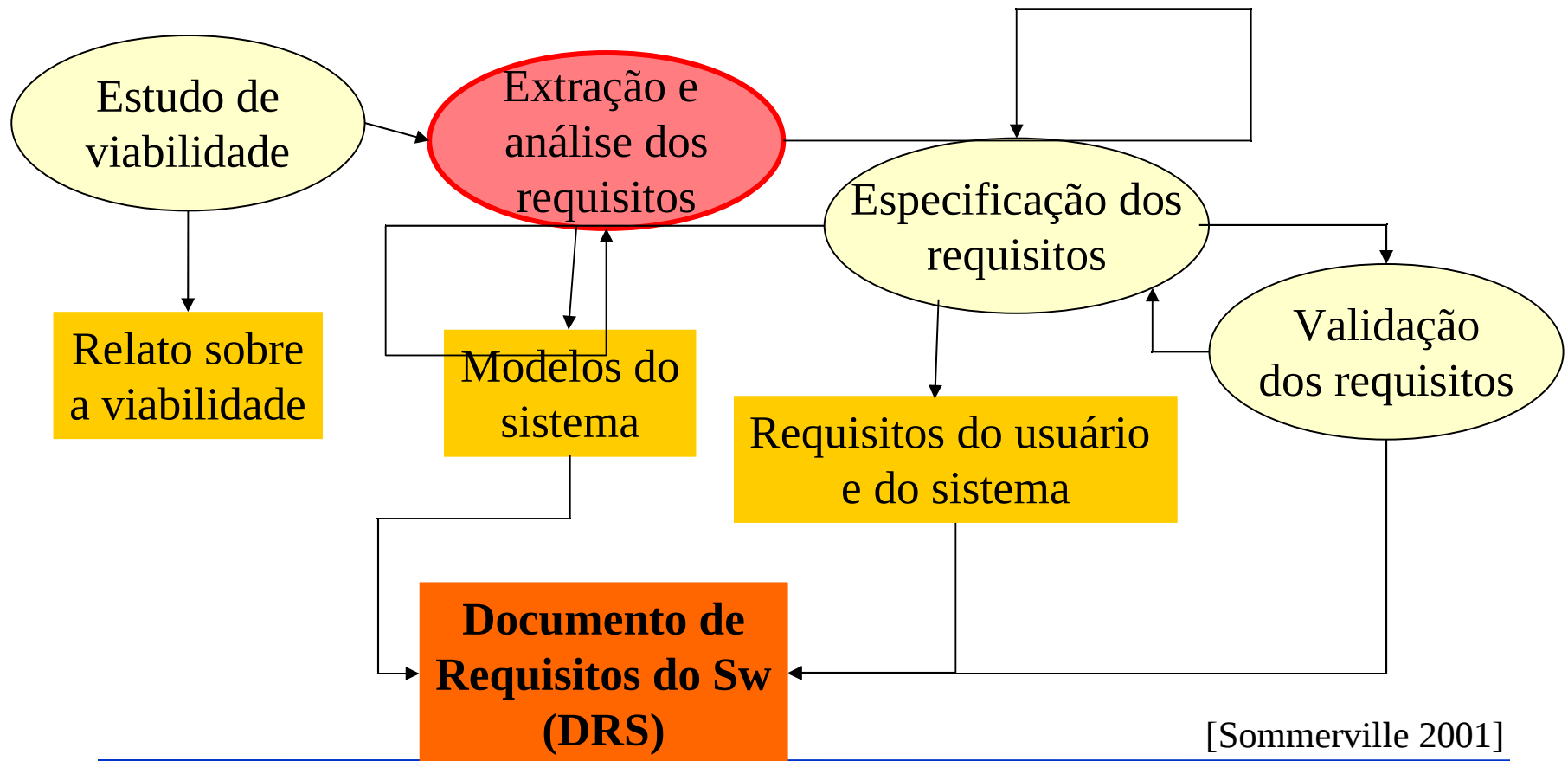


Tópicos

- Dificuldades
- Processo
- Estratégias
- Métodos



O processo de engenharia de requisitos



[Sommerville 2001]



Objetivos

- Descobrir e entender os requisitos através de interação com usuários
- Classificar e organizar os requisitos
- Determinar prioridades



Importância

B.C. by Johnny Hart



Dificuldades - 1

- Usuário desconhece suas reais necessidades e o que o sw pode lhe oferecer
- Desenvolvedor desconhece o domínio do problema
- Desenvolvedor não ouve usuários
- Mal-entendidos entre desenvolvedores e usuários
- Usuário tem dificuldade de tomar decisões pois não pode avaliar as conseqüências



Dificuldades - 2

- Diferentes usuários têm necessidades diferentes
- Fatores políticos dentro da empresa, desconhecidos dos usuários finais, podem afetar requisitos
- Mudanças econômicas ou políticas podem causar mudanças nos requisitos



Falhas típicas nos requisitos

- Fatos incorretos
- Omissões
- Incoerências
- Ambigüidades

➤ Falhas nos requisitos constituem uma das maiores preocupações na produção de software



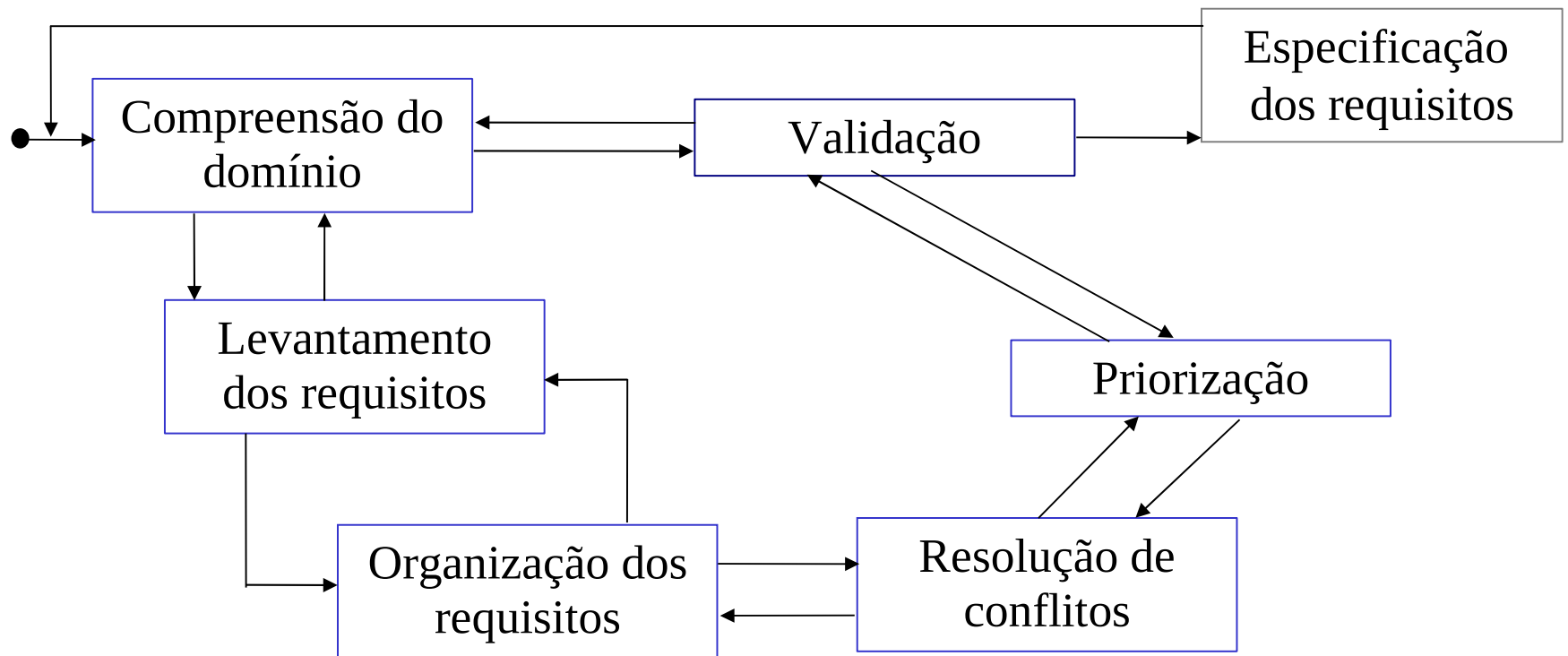
Porquê falhas nos requisitos preocupam

“A parte mais difícil da construção de um sistema de software é decidir, precisamente, o que deve ser construído... Nenhuma outra parte do trabalho afeta mais o sistema resultante se feita errada. Nenhuma outra parte é mais difícil de corrigir depois.”

[Fred Brooks]



Um processo genérico



[Sommerville 2001, c.6.2]



Passos do processo genérico - 1

- Compreensão do domínio
 - o Analista de Requisitos toma conhecimento do domínio do problema.
 - Ex.: descobrir como funciona um caixa eletrônico
- Levantamento de requisitos
 - Analista de Requisitos e clientes/usuários interagem para descobrir os requisitos
 - **Requisitos Funcionais (RF)**
 - Ex.: sistema para caixa eletrônico → realizar saque
 - **Requisitos não funcionais (RNF)**
 - Ex.: sistema de caixa eletrônico → o sistema necessita de controle de acesso



Passos do processo genérico - 2

- Organização dos requisitos
 - os requisitos coletados são organizados de forma coerente
- Resolução de conflitos
 - identificar e resolver conflitos de necessidades dos diferentes clientes / usuários
- Priorização dos requisitos
 - Essenciais
 - Desejáveis
 - Opcionais



Passos do processo genérico - 3

- Validação dos requisitos
 - Ambigüidades, omissões, incorreções → devem ser eliminados
 - Jargões → devem ser esclarecidos
 - Incoerências → reveladas e eliminadas
- Verificação
 - Revisões e outras formas de análise



Outras informações relevantes - 1

- Frequência dos serviços

- Programado

Ex.: todas as manhãs deve ser emitido uma lista dos pedidos recebidos na véspera. O relatório deve conter os pedidos feitos até às 17h e deve ser entregue ao Gerente às 9h da manhã.

- Dirigido por eventos

Ex.: toda vez que o nível de estoque de um item ficar abaixo de um nível de segurança, forneça um relatório de status do item. O relatório deve ser emitido até a manhã seguinte.

- Eventual

Ex.: qual o valor do pedido nº 32764? O fornecedor precisa sabê-lo agora.

[Thelma 2005 – notas de aula]



Outras informações relevantes - 2

- Previsibilidade dos serviços
 - Previsíveis
 - Programado

Ex.: todas as manhãs deve ser emitido uma lista dos pedidos recebidos na véspera. O relatório deve conter os pedidos feitos até às 17h e deve ser entregue ao Gerente às 9h da manhã.
 - Eventual

Ex.: qual o valor do pedido nº 32764? O fornecedor precisa sabê-lo agora.
 - Imprevisíveis
 - Qual o montante de negócios gerado pelo cliente nº 5617 nos últimos 3 anos? Terei uma reunião com ele depois de amanhã.

[Thelma 2005 – notas de aula]



Outras informações relevantes - 3

- Frequência de atualização dos dados
 - Dados atualizados a cada transação

Ex.: qual o saldo do cliente após a realização do saque?
 - Dados atualizados ao final de um período

Ex.: ao final do dia deve ser gerado um relatório com as transações realizadas no caixa eletrônico até às 17h.

[Thelma 2005 – notas de aula]



Levantamento dos requisitos

- Identificar as fontes de informação
 - Interessados: clientes, desenvolvedores, usuários finais e outros que podem ser afetados pelo sistema
 - Livros
 - Documentos
 - Outros sistemas similares
- Coletar fatos



Coleta de fatos

- Leitura de documentos
- Observação
- Entrevistas
- Reuniões
- Questionários
- Engenharia reversa
- Etnografia
- ...



Tarefas [Carvalho e Chiossi 2001]

- Para auxiliar no Levantamento de Requisitos as seguintes tarefas devem ser consideradas:
 - **Perguntar** \Rightarrow encontrar a pessoa apropriada (ex.: usuários)
 - **Observar e inferir** \Rightarrow observar o comportamento dos usuários de um produto existente (manual ou automático) para inferir suas necessidades
 - **Discutir e formular** \Rightarrow discutir com os usuários suas necessidades e, juntamente com eles, formular um entendimento comum dos requisitos.



Tarefas [Carvalho e Chiossi 2001]

- **Negociar** \Rightarrow ajuda na priorização. Negociar com os usuários quais requisitos devem ser incluídos, excluídos ou modificados
 - **Estudar e identificar problemas** \Rightarrow investigar problemas para identificar os requisitos que podem melhorar o produto.
Ex.: o produto atual é muito lento? É pouco seguro (*safe*)?
 - **Supor** \Rightarrow usar a intuição quando não existe acesso ao usuário ou para a criação de um produto inexistente.
-



Técnicas para extração e análise de requisitos [Carvalho e Chiossi 2001]

- Técnicas informais
 - requisitos obtidos através de interação com usuários
 - descrição dos modelos de domínio e do produto é informal, geralmente em linguagem natural
- Técnicas formais
 - construção de um modelo conceitual do problema
 - construção de um protótipo (eventual)



Técnicas informais

- Etnografia
- Entrevistas com clientes / usuários
- “Brainstorming”
- Joint Application Design (JAD)
- PIECES
- ...



Etnografia [Sommerville2001, c.6]

- Conceito
 - Técnica de observação que pode ser usada para a compreensão dos requisitos sociais e organizacionais
- Porquê usá-la em Eng. de Sw
 - A satisfação de requisitos sociais e organizacionais é crucial para o sucesso de um produto



Etnografia

- Como fazer:
 - Imersão no ambiente de trabalho onde o sistema será usado
 - O dia-a-dia de trabalho dos participantes é observado.
- Vantagens:
 - Útil para descobrir **requisitos implícitos**, que refletem a maneira com que as pessoas **realmente** fazem as tarefas e não a maneira como elas **deveriam fazê-las**, conforme descrito em processos formais.
 - Útil para descobrir requisitos derivados da **cooperação com** ou da **expertise** de outros. Um observador externo pode perceber melhor as relações entre as tarefas dos diversos indivíduos dentro da organização.



Entrevista

- Identificar candidatos
 - clientes, usuários, gerentes, pessoas que não usam diretamente o sw mas interagem com usuários, ...
- Preparar a entrevista
 - agendar entrevista e preparar lista de questões
- Conduzir a entrevista
 - informar os objetivos da entrevista, o porquê da mesma, temas abordados, tempo alocado para cada tema, uso da informação obtida
- Finalizar entrevista
 - resumir informações obtidas, indicando pontos ainda em aberto e próximas ações a serem tomadas



“Brainstorming”

Reuniões com um líder. Em cada sessão há:

- geração de idéias, onde cada participante fornece idéias, sem discussão sobre seu mérito
- consolidação das idéias, onde as idéias fornecidas são discutidas, revisadas e organizadas

- 😊 útil no começo do processo de extração de requisitos
- 😊 muito informal, o que pode estimular a criatividade
- 😊 facilidade de ser aprendida
- 😞 os resultados obtidos podem não ter boa qualidade



Joint Application Development (JAD)

- Histórico
 - Proposto em 1977 por Chuck Morris, da IBM
 - Adotado pela IBM Canadá em 1980
 - A IBM formaliza o JAD em 1984
- Princípios básicos
 - reuniões de grupo onde cada participante tem um papel
 - uso de técnicas visuais para melhorar comunicação e entendimento
 - processo bem estruturado
 - uso de documentação padrão que é preenchida e assinada por todos os participantes



JAD

- Apesar de ter “desenvolvimento” no nome, o JAD não é método de desenvolvimento pois só cobre definição e análise de requisitos, bem como projeto.
- Quando usar:
 - Desenvolvimento de sistemas novos
 - Melhorias de sistemas existentes
 - Conversão de sistema
 - Aquisição de produto



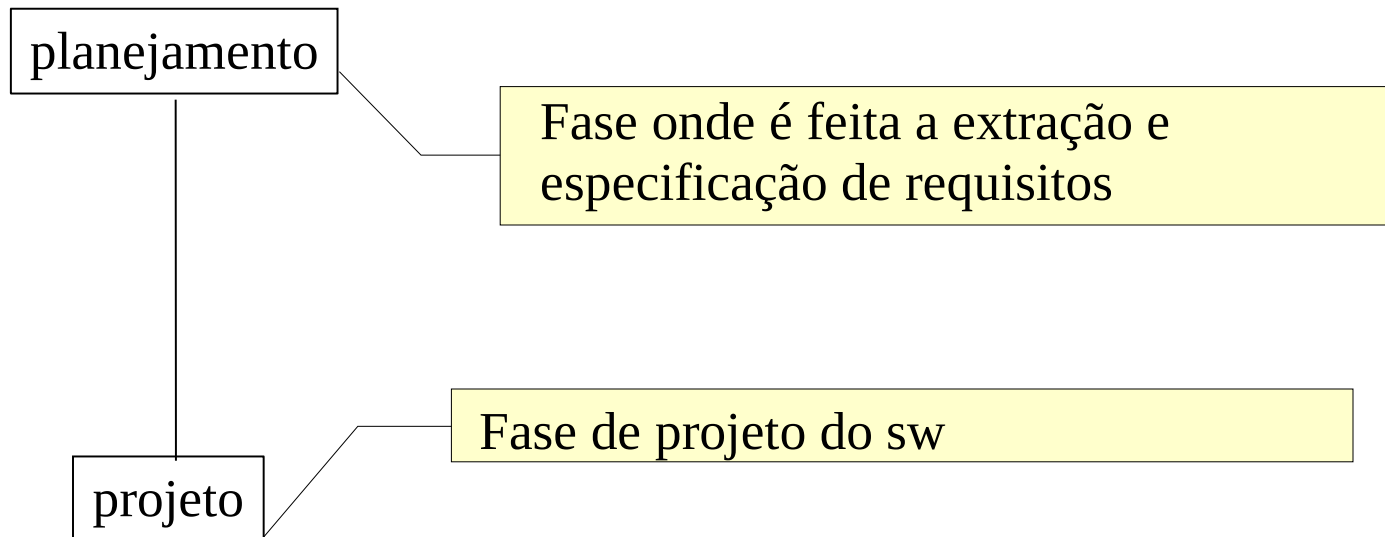
JAD: Participantes

Cada participante tem um papel

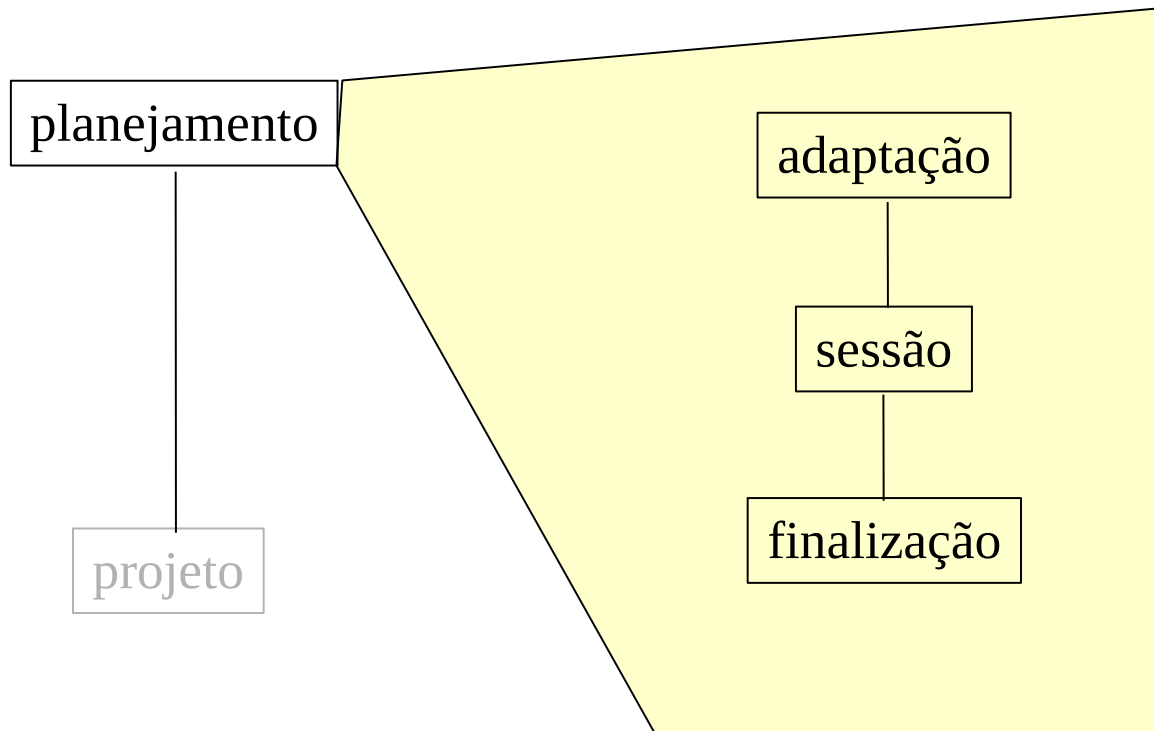
- **Líder:** conhecedor da técnica; coordena as reuniões
- **Engenheiro de requisitos:** responsável pela produção dos documentos de saída das sessões. Pode fazer protótipos se julgar necessário.
- **Executor:** responsável pelo produto. Informa o porquê da construção do produto. Aloca recursos.
- **Representantes dos usuários**
- **Representantes de produtos de sw:** ajudam usuários a compreender as capacitações e limitações de produtos de sw
- **Especialista:** conhecedor de algum tópico específico



JAD: processo



JAD: processo



JAD: atividades de cada fase

- Adaptação
 - adaptação da técnica ao produto de sw desenvolvido
- Sessão
 - reuniões em que cada participante traz idéias e visões diferentes do produto. Ao final de cada sessão tem-se o acordo de todos os participantes a respeito dos requisitos levantados
- Finalização
 - geração do documento de requisitos de sw a partir das informações produzidas durante as sessões



JAD – Mais informações

- www.umsl.edu/~sauter/analysis/488_f01_papers/rottman.htm
- <http://www.cse.fau.edu/~maria/COURSES/CEN4010-SE/jad.pdf>
- Entre inúmeras outras



PIECES [Carvalho e Chiossi 2001]

- **PIECES** – sigla em inglês para desempenho (Performance), Informação e dados, Economia, Controle, Eficiência e Serviços
- Útil para determinar por onde começar:
 - Problemas a serem levantados são organizados de acordo com as seis categorias acima
 - Questões são elaboradas para cada uma das categorias
- Melhores resultados quando existe produto anterior



PIECES - Categorias

- Desempenho (P)
 - Para extrair requisitos de desempenho é preciso fazer perguntas do tipo:
 - Quais tarefas serão executadas?
 - Qual o número de tarefas a serem completadas por unidade de tempo (*throughput*)?
 - Qual a quantidade de tempo necessária para realizar cada tarefa (tempo de resposta)?



PIECES - Categorias

- Informação e dados
 - O produto atual fornece as informações que o usuário necessita?
 - As informações fornecidas são em número e formato que o usuário espera?
 - As informações são fornecidas no tempo certo?



PIECES - Categorias

- Economia
 - O custo de se usar um determinado produto pode estar ligado a dois fatores:
 - Nível do serviço → desempenho
 - Capacidade de lidar com alta demanda nas horas de pico ⇒ necessidade de processadores adicionais, unidades de disco com alta capacidade, conexões de rede, entre outras



PIECES - Categorias

- Controle
 - O acesso ao sistema deve ser restrito a certos usuários?
 - O acesso a algumas informações deve ser restrito?
 - O sistema deve ter a habilidade de monitorar e restaurar a execução após uma situação de erro?
 - O usuário tem a capacidade de abortar a execução do sistema em caso de erro?
 - O usuário pode recuperar dados perdidos?



PIECES - Categorias

- Eficiência
 - Está relacionada com a perda no uso de recursos, ou seja, é a relação entre os recursos efetivamente usados na realização de uma tarefa útil e o total de recursos alocados. A ineficiência pode estar ligada a:
 - Redundâncias desnecessárias
 - ex.: coletar e armazenar o mesmo dado mais de uma vez
 - Uso de algoritmos ou estruturas de dados inadequados para o problema
 - Interface com baixa usabilidade



PIECES - Categorias

- Serviços
 - Quais os serviços que o produto deve realizar?
 - Como estes serviços devem ser fornecidos?
 - Para que outros produtos de software serão prestados serviços?
 - Que produtos prestam serviços para o que está sendo desenvolvido?



Principais pontos aprendidos

