



Universidade Luterana do Brasil
Faculdade de Informática

Disciplina de Engenharia de Software
Professor Luís Fernando Garcia - www.garcia.pro.br

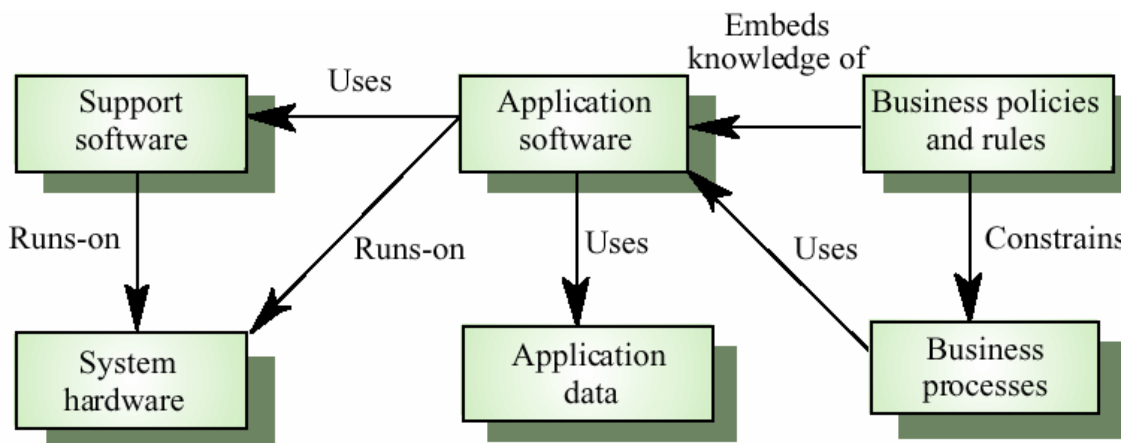
EVOLUÇÃO EM ENGENHARIA DE SOFTWARE

10

Sistemas Legados

- O investimento em desenvolvimento de SW é bastante alto
- ROI – utilização do sistema por vários anos
-
- Muitas empresas DEPENDEM de sistemas que tem mais de 20 anos
- Estes sistemas costumam ser FUNDAMENTAIS para o funcionamento estratégico
-
- Sistemas LEGADOS não são os sistemas originalmente fornecidos
- Sofreram alterações – mudanças contínuas
 - Fatores internos e externos
 - Economia nacional e internacional
 - Alterações de leis
 - Mudanças de gestão
 - Mudança de organização estrutural
-
- Mudanças envolveram várias pessoas diferentes – nenhuma pessoa o conhece integralmente atualmente
-
- Fácil – Substituição de equipamentos e redes
- Complicado – Descartar SISTEMAS e substituir por sw mais moderno
-
- Alto nível de RISCOS
- Estratégia de negócio ARRISCADA
 - Raramente existe especificação completa do sistema legado
 - Documentação original perdida
 - Ou se existente pode não incorporar as mudanças
 - Processos corporativos e sistema altamente ENTRELAÇADO
 - Processos criados para A PARTIR do software existente
 - Alteração no SISTEMA = Alteração nos PROCESSOS
 - REGRAS/RESTRICÇÕES corporativas inseridas no sistema
 - E NÃO DOCUMENTADAS em nenhum outro lugar
 - Exemplo – regras de análise de riscos de clientes
 - RISCOS inerentes ao desenvolvimento de um novo sistema
- QUESTÃO
 - MANTER = evita riscos da substituição → MANUTENÇÃO DISPENDIOSA
- **MANUTENÇÃO DE SISTEMAS LEGADOS**
-
- Não há estilo/padrão de programação – diferentes pessoas/equipes mexendo
- Linguagem de programação obsoleta – FALTA DE MÃO DE OBRA
- Documentação do sistema inadequada e DESATUALIZADA –
 - às vezes somente tem-se o CÓDIGO-FONTE
 - às vezes somente o EXECUTÁVEL

- Corrompido a ESTRUTURA do sistema devido a várias manutenções
- Questão da PERFORMANCE e COMPREENSÃO
- Formato de ARMAZENAMENTO de dados em vários tipos de arquivos
- Estrutura dos Sistemas Legados
- Sistemas SOCIOTÉCNICOS baseados em computadores
 - Software
 - Apoio – Sistemas operacionais/Drivers/Linguagens de programação
 - Aplicação – Serviços de negócios
 - Hardware
 - Mainframes – manutenção dispendiosa
 - Dados
 - Volume imenso, inconsistente, duplicados
 - Processos corporativos/Negócios



- MODIFICAÇÕES ENVOLVEM MAIS COMPONENTES QUE SOMENTE O SOFTWARE
- **AVALIAÇÃO DOS SISTEMAS LEGADOS**
- Avaliação do ROI
- AS 4 ESTRATÉGIAS MAIS APROPRIADAS
- Descartar completamente o sistema
- Continuar a manter o sistema
- Transformar o sistema de alguma maneira para melhorar sua facilidade de manutenção
- Substituir o sistema por um novo
- Perspectivas
- Perspectiva de NEGÓCIOS
 - Avaliação do valor do sistema para a empresa
- Perspectiva de SISTEMA
 - Avaliação da qualidade do software de aplicação e HW/SW de apoio
- Avaliação da QUALIDADE do sistema LEGADO
- Avaliação do Processo de Negócios
 - Existe um modelo definido do processo ? Esse modelo é seguido?
 - Mesmos processos para as diferentes partes da organização?
 - Processos em adaptação?
- Avaliação do Ambiente
 - Estabilidade do fornecedor

- Taxa de falhas
- Idade
- Desempenho
- Custos de manutenção
- Interoperabilidade
- Avaliação do Software de Aplicação
 - Facilidade de compreensão
 - Documentação
 - Dados
 - Desempenho
 - Linguagem de Programação
 - Gerenciamento de configuração
 - Dados de teste
 - Habilidades pessoais

Mudanças em Software

- É impossível produzir sistemas de qualquer tamanho que não precisem ser modificados
 - Novos requisitos emergem
 - Os requisitos são modificados
 - Correção de erros
 - Melhorar desempenho
- **ESTRATÉGIAS PARA MUDANÇAS EM SOFTWARE**
 - Manutenção de software
 - Em resposta a requisitos modificados – estrutura permanece estável
 - Transformação de arquitetura
 - Abordagem mais radical
 - Alterações significativas na arquitetura do software
 - Exemplo: arquitetura centralizada → arquitetura cliente-servidor
 - Reengenharia de software
 - Nenhuma funcionalidade nova é adicionada ao sistema
 - Modificado a fim de tornar mais fácil sua compreensão e alteração
- **MANUTENÇÃO**
- Ainda vista como atividade de segunda classe
- Tipos de manutenção:
 - Manutenção para reparar os defeitos no software – 17%
 - Manutenção para adaptar o software a um ambiente operacional diferente – 18%
 - Manutenção para acréscimos de funcionalidades do sistema – 65%
- Fatores envolvidos - custos
 - Estabilidade da equipe – bons profissionais dispersados após conclusão ...
 - Responsabilidade contratual – equipe terceirizada para manutenção ...
 - Habilidade da equipe – pessoal com pouca experiência e sem contato com o sistema
 - Idade e estrutura do programa
- Processo de manutenção
 - Pedidos de alterações

- Análise de impacto
- Planejamento de release
 - Reparo
 - Adaptação
 - Incremento
- Implementação da mudança
- Release do sistema

- **EVOLUÇÃO DA ARQUITETURA**

- Fatores que levam a evolução da arquitetura

- Custos de HW
- Interface com o usuário
- Acesso distribuído aos sistemas

- Importância para os negócios
- Idade do sistema
- Estrutura do sistema
- Políticas de suprimento de hardware

- **REENGENHARIA**

- Reimplementar sistemas legados para que sua manutenção seja mais fácil

-

- REDOCUMENTAR

- ORGANIZAR E REESTRUTURAR O SISTEMA

- TRADUZIR PARA UMA LINGUAGEM DE PROGRAMAÇÃO MAIS MODERNA

- MODIFICAR E ATUALIZAR A ESTRUTURA E OS DADOS

-

- Não modifica a ESTRUTURA nem a FUNCIONALIDADE do software

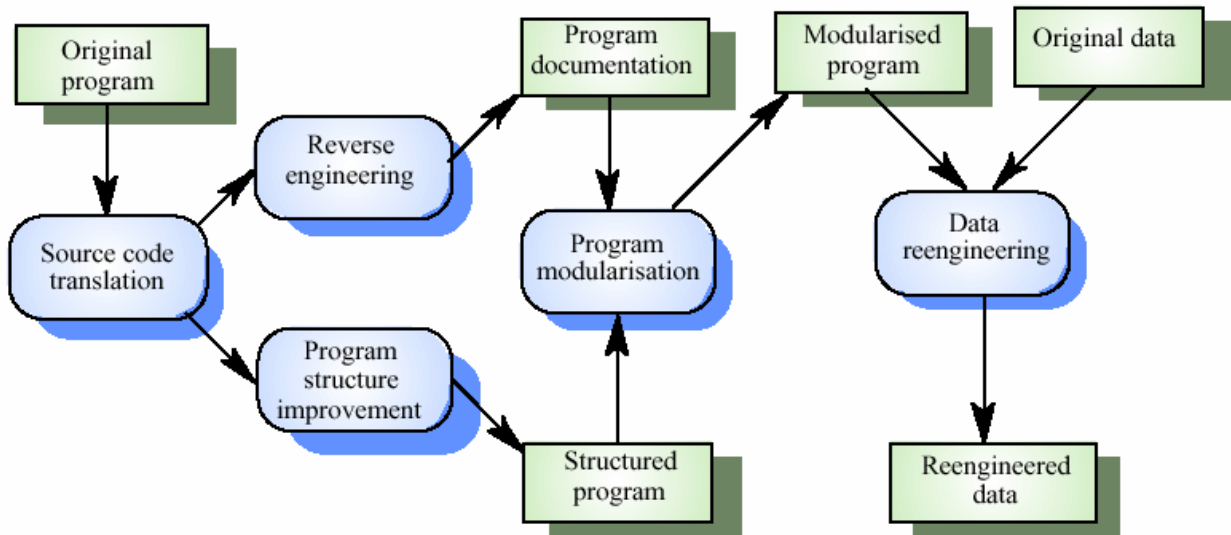
- Vantagens

- Riscos reduzidos
- Custos reduzidos

- Associada a REENGENHARIA DE PROCESSO DE NEGÓCIOS

- Atividades

- Tradução do código-fonte
- Engenharia reversa
- Melhoria da estrutura do programa
- Modularização de programa
- Reengenharia de dados



- Tradução do código-fonte
 - Atualização da plataforma de hardware
 - Escassez de pessoal habilitado
 - Mudanças na política organizacional
 - Falta de suporte de software
- Engenharia reversa
 - Para obter uma especificação de requisitos para a substituição do sistema