

Engenharia de Sistemas

- Projetar, implementar, desenvolver e operar sistemas que incluem hardware, software e pessoas

Objetivos

- Explicar porque um sistema de software é afetado por questões mais abrangentes de engenharia de sistemas
- Apresentar o conceito de propriedades emergentes de sistemas, tais como confiabilidade e segurança
- Explicar porque o ambiente do sistema deve ser considerado no processo de projeto de sistemas
- Explicar engenharia de sistemas e os processos de suprimento de sistemas

Tópicos abordados

- Propriedades do sistema emergentes
- Sistemas e o seu ambiente
- Modelagem de sistemas
- O processo de engenharia de sistemas
- Suprimento de sistemas

O que é um sistema?

- Uma coleção útil de componentes inter-relacionados, que funcionam em conjunto para atingir algum objetivo comum.
- Um sistema pode incluir software, hardware de natureza mecânica, elétrica e eletrônica e ser operado por pessoas.
- Os componentes do sistema são dependentes de outros componentes do sistema
- As propriedades e comportamento dos componentes do sistema estão intrinsecamente interligados

Problemas da engenharia de sistemas

- Sistemas grandes são normalmente projetados para resolver *problemas intrincados (wicked problems)*
- Engenharia de sistemas requer grande coordenação entre áreas diversas
 - Possibilidades quase infinitas de combinações de projeto entre componentes
 - Desconfiança mútua e falta de compreensão entre áreas de engenharia
- Os sistemas devem ser projetados para durar muitos anos em um ambiente de mudanças

Software e engenharia de sistemas

- A proporção de software nos sistemas está aumentando. Equipamentos eletrônicos de uso geral gerenciados por software estão substituindo sistemas de fins específicos
- Problemas de engenharia de sistemas são similares aos problemas de engenharia de software
- Software é (infelizmente) visto como um problema em engenharia de sistemas. Muitos projetos de sistemas grandes sofrem atrasos por causa de problemas de software

Propriedades emergentes

- Propriedades do sistema como um todo, que não podem ser derivadas das propriedades dos componentes de um sistema
- Propriedades emergentes são uma consequência dos relacionamentos entre os componentes do sistema
- Elas podem, portanto, ser verificadas e medidas somente quando os componentes tiverem sido integrados ao sistema

Exemplos de propriedades emergentes

- *O peso total do sistema*
 - Este é um exemplo de uma propriedade emergente que pode ser computada a partir das propriedades individuais dos componentes.
- *A confiabilidade do sistema*
 - Isso depende da confiabilidade dos componentes do sistema e dos relacionamentos entre os componentes.
- *A facilidade de uso do sistema*
 - Essa é uma propriedade complexa, que não depende simplesmente do hardware e do software do sistema, mas também dos operadores do sistema e do ambiente onde ele é utilizado.

Tipos de propriedades emergentes

- Propriedades funcionais
 - Aparecem quando todas as partes de um sistema trabalham juntas para atingir um objetivo. Por exemplo: uma bicicleta possui a propriedade funcional de ser um dispositivo de transporte quando ela for montada a partir dos seus componentes.
- Propriedades emergentes não-funcionais
 - Os exemplos são a confiabilidade, o desempenho, segurança e proteção. Estão relacionadas ao comportamento do sistema no seu ambiente operacional. Elas são frequentemente críticas para os sistemas baseados em computador, uma vez que a não obtenção dos níveis mínimos nestas propriedades podem tornar o sistema inutilizável.

Confiabilidade de Sistemas

- Falhas podem se propagar através do sistema, em função da inter-dependência entre componentes
- Falhas nos sistemas frequentemente ocorrem em função de inter-relacionamentos não previstos entre componentes
- É provavelmente impossível antecipar todos os relacionamentos entre componentes
- As medidas de confiabilidade de software podem dar uma falsa idéia da confiabilidade do sistema

Influências sobre a confiabilidade

- *Confiabilidade de hardware*
 - Qual é a probabilidade de um componente de hardware falhar e quanto tempo será necessário para consertar tal componente?
- *Confiabilidade de software*
 - Qual a probabilidade de um componente de software produzir uma saída errada? Falhas de software são normalmente diferentes de falhas de hardware, uma vez que o software não se desgasta.
- *Confiabilidade do operador*
 - Qual a probabilidade de que o operador de um sistema cometa um erro?

Relacionamentos de confiabilidade

- Falhas de hardware podem gerar sinais espúrios, que estão fora da faixa de entradas esperada pelo software
- Erros de software podem causar a ativação de alarmes, os quais causam cansaço no operador e o levam a cometer erros
- O ambiente no qual o sistema está instalado pode afetar sua confiabilidade

As propriedades 'shall-not'

- Propriedades tais como performance e confiabilidade podem ser medidas
- Entretanto, existem propriedades que o sistema não deveria exibir
 - Segurança – o sistema não deve se comportar de uma maneira insegura
 - Proteção – o sistema não deve permitir uso não autorizado
- É muito difícil medir ou avaliar essas propriedades

Sistemas e o seu ambiente

- Os sistemas não são independentes, mas existem dentro de um ambiente
- A função do sistema pode ser alterar o seu ambiente
- O ambiente afeta o funcionamento do sistema. Por exemplo, o sistema pode exigir energia elétrica do seu ambiente
- O ambiente organizacional pode ser tão importante quanto o físico

Hierarquia de sistemas



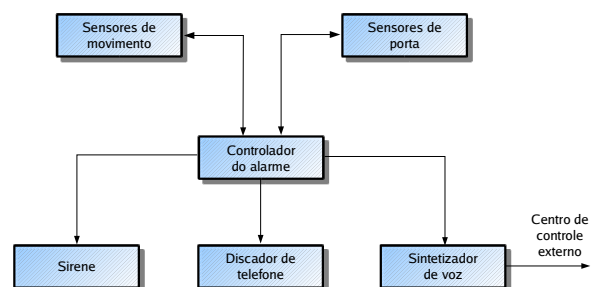
Fatores humanos e organizacionais

- *Mudanças no processo*
 - O sistema requer mudanças no processo de trabalho, no ambiente?
- *Mudanças nas tarefas*
 - Os sistemas diminuem a habilidade dos usuários em um ambiente ou faz com que eles modifiquem o modo como trabalham?
- *Mudanças organizacionais*
 - O sistema modifica a estrutura de poder em uma organização?

Modelagem da arquitetura do sistema

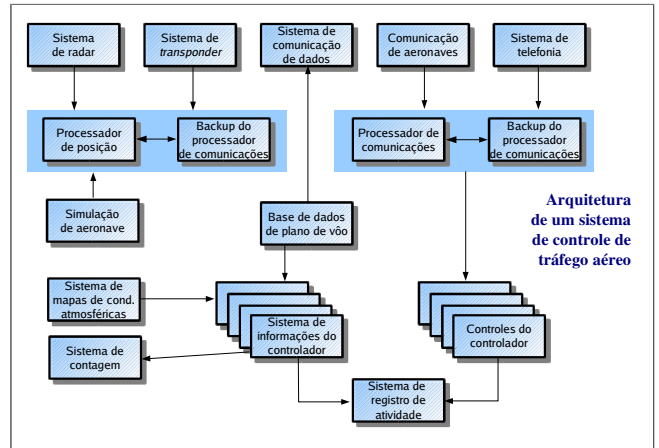
- Um modelo arquitetônico apresenta uma visão abstrata dos sub-sistemas que compõem um sistema
- Pode incluir os principais fluxos de informação entre os sub-sistemas
- Apresentado normalmente como um diagrama de bloco
- Pode identificar diferentes tipos de componentes funcionais no modelo

Sistema de alarme contra intrusos



Tipos de componentes no sistema de alarme

- **Sensor**
 - Sensor de movimento, sensor de porta
- **Atuador**
 - Sirene
- **Comunicação**
 - Discador de telefone
- **Coordenação**
 - Controlador do alarme
- **Interface**
 - Sintetizador de voz



Componentes funcionais de sistema

- Componentes de sensores
- Componentes de atuadores
- Componentes de computação
- Componentes de comunicação
- Componentes de coordenação
- Componentes de interface

Componentes de sistema

- **Componentes de sensor**
 - Coletam informações do ambiente do sistema, p. ex., os radares em um sistema de controle de tráfego aéreo
- **Componentes atuadores**
 - Causam algum tipo de mudança no ambiente do sistema, p. ex., válvulas em um sistema de controle de processo que aumentam ou diminuem o fluxo de material em um tubo
- **Componentes de computação**
 - Realizam algumas computações sobre uma entrada para produzir uma saída, p. ex., um processador de ponto flutuante em um sistema de computador

Componentes de sistema

- **Componentes de comunicação**
 - Permite que os componentes do sistema comuniquem-se entre si. Por exemplo: a rede que conecta computadores distribuídos
- **Componentes de coordenação**
 - Coordinam as interações de outros componentes do sistema, p. ex., um escalonador em um sistema de tempo real
- **Componentes de interface**
 - Facilitam as interações de outros componentes do sistema, p. ex., a interface do operador
- Todos os componentes hoje em dia são controlados por software

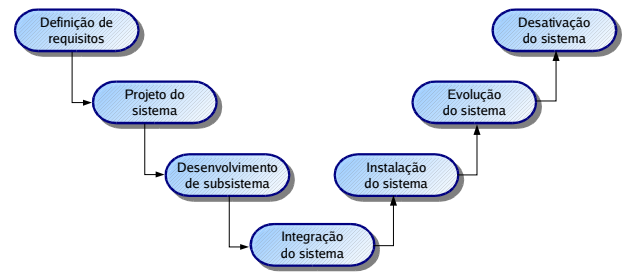
Tipos de componentes no sistema de alarme

- **Sensor**
 - Sensor de movimento, sensor de porta
- **Atuador**
 - Sirene
- **Comunicação**
 - Discador de telefone
- **Coordenação**
 - Controlador do alarme
- **Interface**
 - Sintetizador de voz

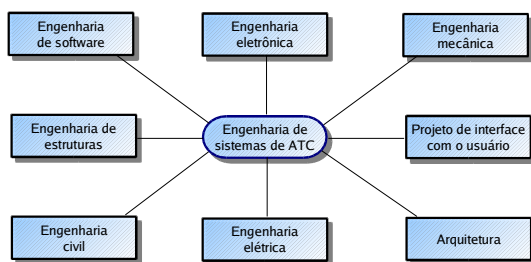
O processo de engenharia de sistemas

- Segue normalmente um modelo “cascata”, devido a necessidade de desenvolvimento paralelo de partes diferentes do sistema
 - Pouco espaço para interação entre fases porque as mudanças de hardware são muito caras. O software pode ter que compensar os problemas de hardware
- Envolve inevitavelmente engenheiros de áreas diferentes, que devem trabalhar juntos
 - Grande possibilidade de mal-entendidos. Áreas diferentes usam vocabulários diferentes, e muita negociação é necessária. Os engenheiros podem ter agendas pessoais para preencher

O processo de engenharia de sistemas



Envolvimento inter-disciplinar



Definição de requisitos de sistema

- Três tipos de requisitos são definidos neste estágio
 - *Requisitos funcionais abstratos*. As funções do sistema são definidas de um modo abstrato
 - *Propriedades do sistema*. São definidos os requisitos não-funcionais para o sistema em geral
 - *Características indesejáveis*. São especificados os comportamentos não aceitáveis do sistema
- Devem ser definidos também os objetivos da empresa como um todo para o sistema

Objetivos do sistema

- Objetivos funcionais
 - Fornecer um sistema de alarme contra incêndios e contra intrusos para o edifício, com o objetivo de divulgar avisos internos e externos referentes a incêndios ou à entrada de pessoa não autorizada
- Objetivos organizacionais
 - Assegurar que o trabalho normal realizado no edifício não seja seriamente perturbado por ocorrências como incêndios e a entrada de pessoas não autorizadas

Problemas nos requisitos de sistema

- Mudanças à medida em que o sistema está sendo especificado
- Deve antecipar desenvolvimentos de hardware e comunicações durante a vida útil do sistema
- Requisitos não-funcionais são (particularmente) difíceis de definir sem uma impressão da estrutura dos componentes do sistema

O processo de projeto de sistemas

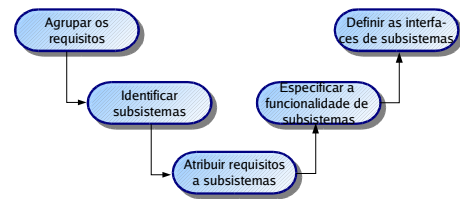
- Agrupar os requisitos
 - Organizar os requisitos em grupos relacionados
- Identificar subsistemas
 - Identificar um conjunto de subsistemas que podem, em conjunto, atender aos requisitos do sistema
- Atribuir requisitos a subsistemas
 - Podem ocorrer problemas específicos quando COTS são integrados
- Especificar funcionalidade dos subsistemas
- Definir interfaces de subsistemas
 - Atividade crítica para desenvolvimento paralelo de subsistemas

©Ian Sommerville 2000

Software Engineering, 6th edition, Chapter 2

Slide 31

O processo de projeto de sistemas



©Ian Sommerville 2000

Software Engineering, 6th edition, Chapter 2

Slide 32

Problemas de projeto de sistemas

- A atribuição de requisitos para componentes de hardware, software e humanos pode envolver uma boa dose de negociação
- Problemas difíceis de projeto são freqüentemente atribuídos a serem prontamente resolvidos usando software
- As plataformas de hardware podem ser inapropriadas para os requisitos de sistema, então o software deve compensar esse problema

©Ian Sommerville 2000

Software Engineering, 6th edition, Chapter 2

Slide 33

Desenvolvimento de subsistemas

- Tipicamente, há projetos paralelos para desenvolver o hardware, o software, o hardware e as comunicações
- Pode envolver a aquisição de alguns COTS
- Falta de comunicação entre equipes de implementação
- Um mecanismo burocrático e lento para propor alterações no sistema significa que o calendário de desenvolvimento tenha que ser estendido em função da necessidade de retrabalho

©Ian Sommerville 2000

Software Engineering, 6th edition, Chapter 2

Slide 34

Integração de sistemas

- O processo de unir hardware, software e pessoas para fazer um sistema
- Deve ser conduzida de forma incremental, para que os subsistemas sejam integrados um por vez
- Problemas de interface entre subsistemas são normalmente encontrados nesta fase
- Os problemas podem ser causados por entregas desencontradas de componentes do sistema

©Ian Sommerville 2000

Software Engineering, 6th edition, Chapter 2

Slide 35

Instalação de sistemas

- Suposições ambientais podem estar incorretas
- Pode haver resistência humana à introdução de um novo sistema
- O sistema pode ter que coexistir com sistemas alternativos por algum tempo
- Podem haver problemas de instalação física (por exemplo, problemas de cabeamento)
- O nível de treinamento dos operadores tem que ser identificado

©Ian Sommerville 2000

Software Engineering, 6th edition, Chapter 2

Slide 36

Operação de sistemas

- Poderá trazer à tona requisitos não previstos
- Os usuários poderão utilizar o sistema de uma forma que não foi antecipada pelos projetistas de sistema
- Pode revelar problemas na interação com outros sistemas
 - Problemas físicos de incompatibilidade
 - Problemas de conversão de dados
 - Alta taxa de erros do operador por causa de interfaces inconsistentes

Evolução de sistemas

- Grandes sistemas têm uma vida longa. Eles devem evoluir para satisfazer mudanças nos requisitos
- A evolução é inerentemente custosa
 - As mudanças devem ser analisadas sob uma perspectiva técnica e de negócios
 - Subsistemas interagem, portanto problemas não antecipados podem surgir
 - Raramente existem explicações para decisões do projeto original
 - A estrutura do sistema é corrompida à medida em que mudanças são feitas
- Sistemas existentes que devem ser mantidos são às vezes chamados de **sistemas legados**

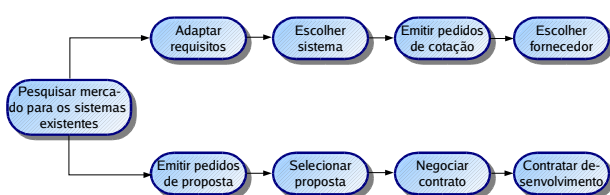
Desativação de sistemas

- Tirar de serviço o sistema após a sua vida útil
- Pode exigir a remoção de materiais (p.ex., produtos químicos perigosos) que poluem o ambiente
 - Deve ser planejado durante o projeto do sistema
- Pode exigir que os dados sejam reestruturados e convertidos para serem utilizados em algum outro sistema

Suprimento de sistemas

- Aquisição de um sistema para uma organização para satisfazer alguma necessidade
- Alguma especificação do sistema e projeto arquitetural são normalmente necessárias antes do suprimento
 - Necessita-se de uma especificação para definir um contrato para o desenvolvimento do sistema
 - A especificação pode permitir a compra de um sistema de prateleira (*commercial off-the-shelf* - COTS). Quase sempre mais barato do que desenvolver um sistema a partir do zero

O processo de suprimento de sistemas



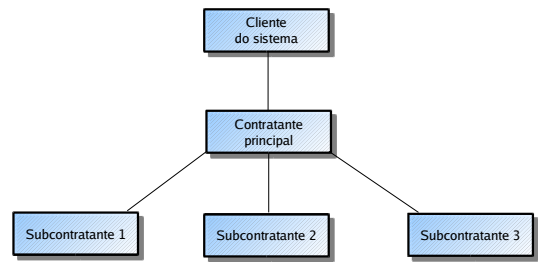
Questões de suprimento

- Os requisitos podem ter que ser modificados para combinar com os recursos de componentes de prateleira
- As especificações de requisitos podem ser parte do contrato para o desenvolvimento do sistema
- Normalmente, existe um período de negociação de contrato para mudanças no acordo após ter sido escolhido o contratado para construir o sistema

Contratantes e subcontratantes

- O suprimento de grande sistemas de hardware/software baseia-se normalmente em torno de algum contratante principal
- Subcontratos são oferecidos a outros fornecedores para fornecer partes do sistema
- O cliente interage com o contratante principal e não negocia diretamente com subcontratantes

Modelo de contratante/subcontratante



Pontos principais

- Engenharia de sistemas requer a entrada de uma série de disciplinas de engenharia
- Propriedades emergentes são propriedades características do sistema como um todo, e não de suas partes componentes
- Os modelos de arquitetura de sistemas mostram os principais subsistemas e inter-conexões. Eles são normalmente descritos usando diagramas de bloco

Pontos principais

- Os tipos de componentes de sistemas são sensores, atuadores, de computação, de coordenação, de comunicação e de interface
- O processo de engenharia de sistemas é normalmente um modelo cascata e inclui especificação, projeto, desenvolvimento e integração
- O suprimento de sistemas preocupa-se em decidir qual sistema comprar e como comprá-lo

Conclusões

- Engenharia de sistemas é difícil! Nunca haverá uma resposta fácil para os problemas do desenvolvimento de sistemas complexos
- Engenheiros de software não têm todas as respostas, mas podem ser melhores em ter um ponto de vista dos sistemas
- Disciplinas de engenharia necessitam reconhecer as potencialidades de cada um e cooperar ativamente em vez de relutantemente no processo de engenharia de sistemas