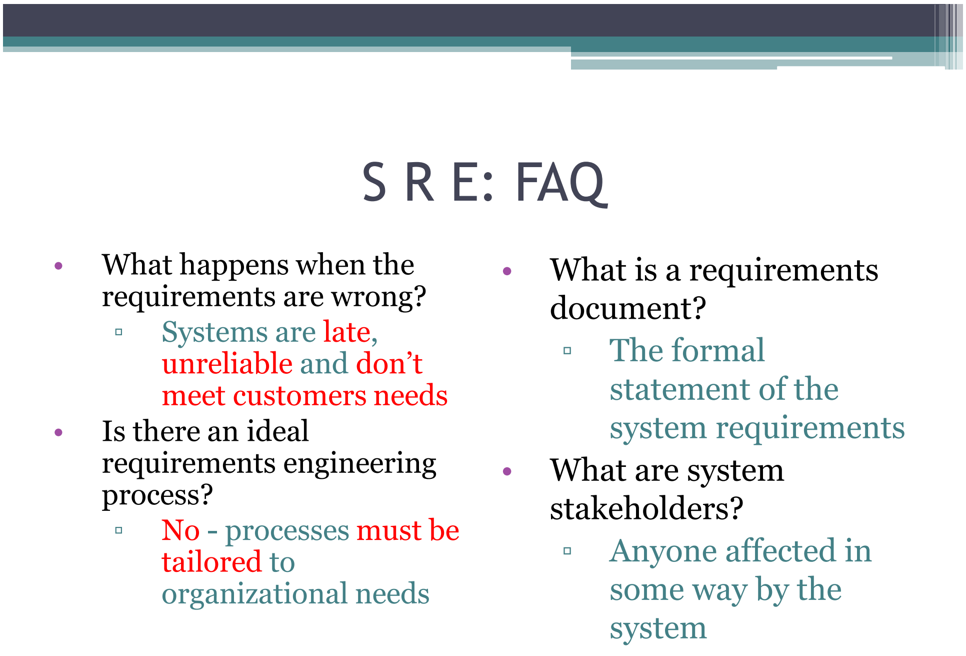
# Software Requirement Engineering

Muhammad Waseem
Lecture 1
Date: 12.02.2014
Department of Computer Science and Software Engineering, FBAS ,International Islamic University Islamabad

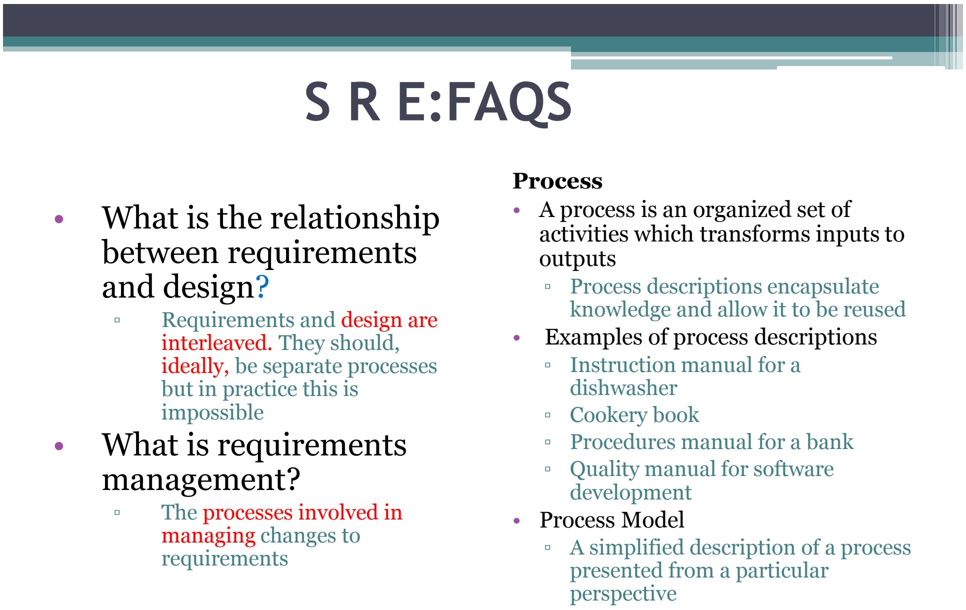
## Requirements  and Requirements Engineering

| **Requirements** | **Requirements Engineering** |
|---|---|
| It is a system service or constraint. | A *process* in which "what is to be done" is elicited, modeled and communicated (Freeman)<br>The *descriptions of* the services and constraints are the requirements for the system" (Somerville)<br>The *process of* finding out, analyzing, documenting and checking these services and constraints is called Requirements Engineering." (Somerville) |



## S R E: FAQ

- What happens when the requirements are wrong?
  - Systems are late, unreliable and don’t meet customers needs
- Is there an ideal requirements engineering process?
  - No - processes must be tailored to organizational needs
- What is a requirements document?
  - The formal statement of the system requirements
- What are system stakeholders?
  - Anyone affected in some way by the system



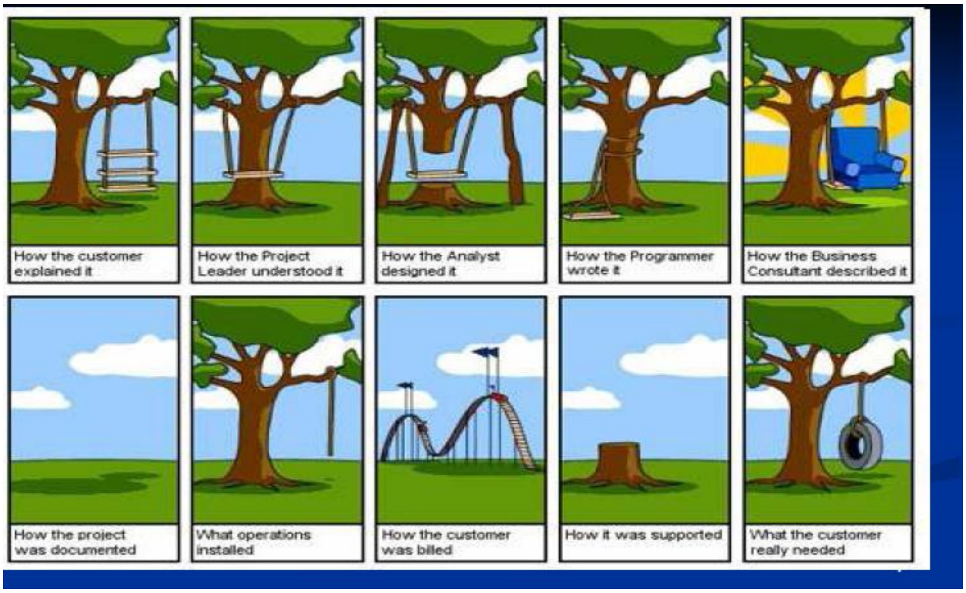
## S R E:FAQS

- What is the relationship between requirements and design?
  - Requirements and design are interleaved. They should, ideally, be separate processes but in practice this is impossible
- What is requirements management?
  - The processes involved in managing changes to requirements

**Process**

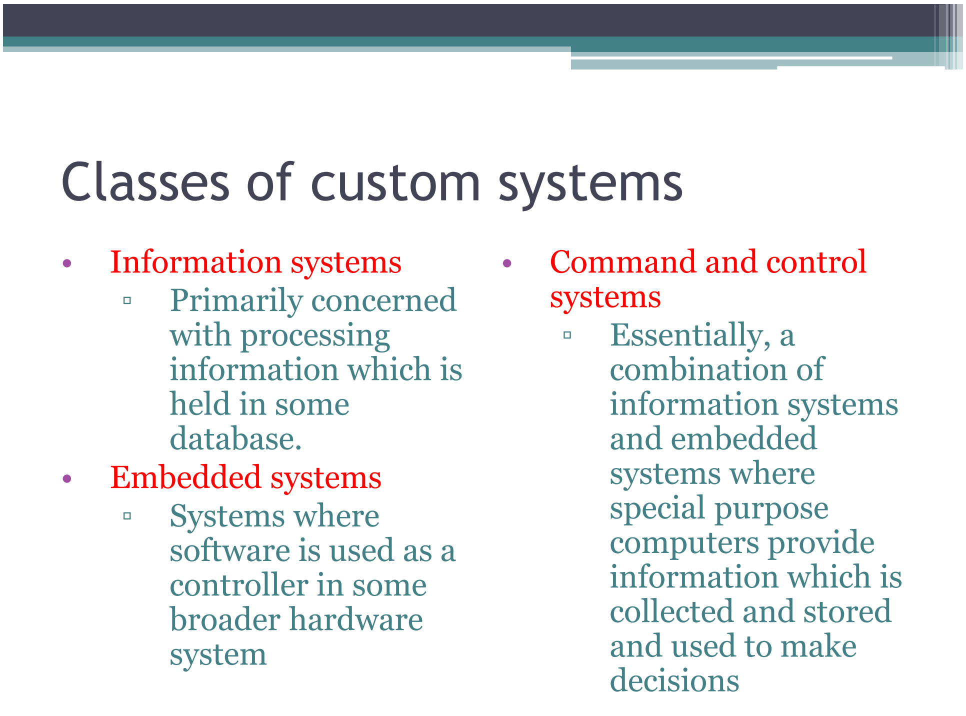
- A process is an organized set of activities which transforms inputs to outputs
  - Process descriptions encapsulate knowledge and allow it to be reused
- Examples of process descriptions
  - Instruction manual for a dishwasher
  - Cookery book
  - Procedures manual for a bank
  - Quality manual for software development
- Process Model
  - A simplified description of a process presented from a particular perspective

## Very Famous Requirements Story


How the customer explained it
How the Project Leader understood it
How the Analyst designed it
How the Programmer wrote it
How the Business Consultant described it
How the project was documented
What operations installed
How the customer was billed
How it was supported
What the customer really needed

## Systems Engineering

- There is a close relationship between software and more general system requirements
- Computer-based systems fall into two broad categories:
  - User-configured systems where a purchaser puts together a system from existing software products
  - Custom systems where a customer produces a set of requirements for hardware/software system and a contractor develops and delivers that system



# Classes of custom systems

- Information systems
  - Primarily concerned with processing information which is held in some database.
- Embedded systems
  - Systems where software is used as a controller in some broader hardware system
- Command and control systems
  - Essentially, a combination of information systems and embedded systems where special purpose computers provide information which is collected and stored and used to make decisions

# Classes of custom systems

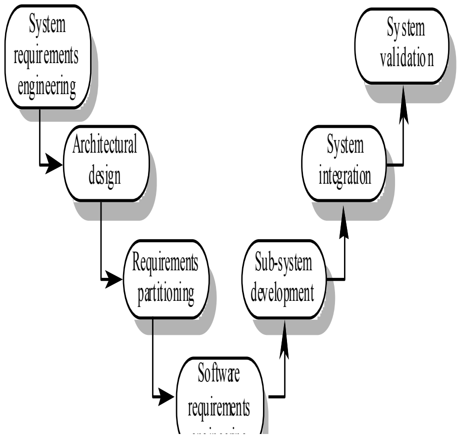
- Information systems
  - **Executive Support System**
  - **Management Information Systems**
  - **Decision-Support Systems**
  - **Knowledge Management Systems**
  - **Transaction Processing Systems**
  - **Office Automation Systems**
- Embedded systems
  - Telecom
  - Smart Cards,
  - Missiles and Satellites,
  - Computer Networking,
- Command and control systems
  - Reading Assignment

# Emergent properties

- Emergent properties are properties of the system as a whole and only emerge once al of its individual sub-systems have been integrated
- Examples of emergent properties
    - Reliability
    - Maintainability
    - Performance
    - Usability
    - Security
    - Safety

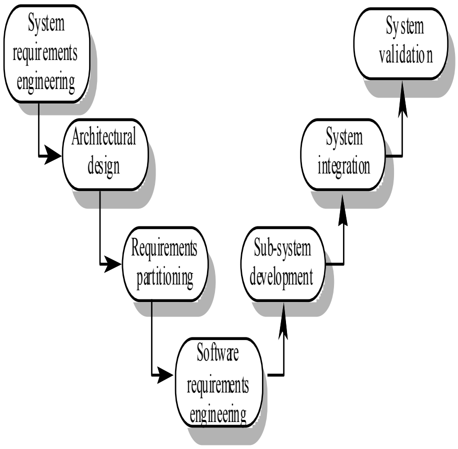
# The Systems Engineering Process

- System requirements engineering
    - The requirements for the system as a whole are established and written to be understandable to all stakeholders
- Architectural design
    - The system is decomposed into sub-systems
- Requirements partitioning
    - Requirements are allocated to these sub-systems
- Software requirements engineering
    - More detailed system requirements are derived for the system software



# The systems engineering process

- Sub-system development
  - The hardware and software sub-systems are designed and implemented in parallel.
- System integration
  - The hardware and software sub-systems are put together to make up the system.
- System validation
  - The system is validated against its requirements.


System requirements engineering
Architectural design
Requirements partitioning
Software requirements engineering
Sub-system development
System integration
System validation

# Some observations about RE

- **RE is not necessarily a sequential process:**
  - **RE is a set of activities that continue throughout the development process**
- **The problem statement will be imperfect**
  - **RE models are approximations of the world**
    - **will contain inaccuracies and inconsistencies**
    - **will omit some information.**
    - **detailed analysis can reduce the risk that these will cause serious problems…**
    - **" …but that risk can never be reduced to zero**
- **Perfecting a specification may not be cost-effective**
  - **Requirements analysis has a cost**
  - **For different projects, the cost-benefit balance will be different**
- **Problem statement should never be treated as fixed**
  - **Change is inevitable, and therefore must be planned for**
  - **There should be a way of incorporating changes periodically**

# Importance of RE

- Problems
  - Increased reliance on software
    - " E.g. cars, dishwashers, cell phones, web services, …
  - Software now the biggest cost element for mission critical systems
    - E.g. Boeing 777
  - Wastage on failed projects
    - E.g. 1997 GAO report: $145 billion over 6 years on software that was never delivered
  - High consequences of failure
    - " E.g. Ariane 5: $500 million payload
    - " E.g. Intel Pentium bug: $475 million
- Key factors:
  - Certification costs
    - E.g. Boeing 777: >40% of software budget spent on testing
  - Re-work from defect removal
    - E.g. Motorola: 60-80% of software budget (was) spent on re-work
  - Changing Requirements
    - E.g. California DMV system

# What do Requirements Engineers do?

A Requirements Engineer is an agent of change

- The requirements engineer must:
  - identify the "problem"/"opportunity"
    - Which problem needs to be solved? (identify problem Boundaries)
    - Where is the problem? (understand the Context/Problem Domain)
    - "Whose problem is it? (identify Stakeholders)
    - " Why does it need solving? (identify the stakeholders' Goals)
    - " How might a software system help? (collect some Scenarios)
    - " When does it need solving? (identify Development Constraints)
    - " What might prevent us solving it? (identify Feasibility and Risk)
- and become an expert in the problem domain

# Key points

- Requirements define what the system should provide and define system constraints
- Requirements problems lead to late delivery and change requests after the system is in use
- Requirements engineering is concerned with eliciting, analyzing, and documenting the system requirements
- Systems engineering is concerned with systems as a whole including hardware, software and operational processes
- The requirements document is the definitive specification of requirements for customers, engineers and managers.
- The requirements document should include a system overview, glossary, statement of the functional requirements and the operational constraints

- Thanks