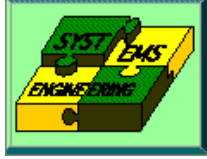
George Mason University



SYST 101: Intro to Systems

Lecture 13

Mar. 4, 2003

C. Wells, SEOR Dept.

Announcements

- Next Class: Midterm exam, in class. Open book, open notes
- Next Week:
  - Midterm break

Agenda

- What is a System, & Key Terminology
- Functions, Functional Decomposition
- Issues Analysis
- Interfaces

Overview/Summary

- All systems exist within other systems.
  - And systems have systems within them.
  - Endless nesting of systems within systems, from the universe down to subatomic particles.
- Systems interact with each other.
  - Both with systems “external” to themselves,
  - And with their “internal” subsystems.

Systems Analysis

- Systems Analysis: The understanding of the structure of a system, and its behavior
  - Structure: what composes a system, what are its subsystems, and what are its interfaces.
  - Behavior: how a system reacts to stimuli from its interfaces (external and internal)

Systems Engineering

- The development or modification of systems to have a desired behavior.
  - Reliably.
  - Without undesired side effects.
  - And in a cost-effective manner.

Systems Analysis

- Analysis of the need
  - Stakeholder requirements
  - House of Quality
- Analysis of the environment & context
  - Issues Analysis
  - External systems and interfaces

Systems Engineering

- Development and design of the new system or system modifications
- Clear, unambiguous representation
  - Functional Decomposition
- Defined, controlled process for bringing the system into being.

What is a System?

- Numerous definitions everywhere
- A System is:
  - A set of interacting components that together accomplish some goal or behavior; it exists within an environment, and can interact with that environment.

Key Terms

- Scope of the System
- Mission or Goals
- Requirements
- Stakeholders
- Lifecycle
- Interactions
- Behavior

Scope of the System

- What is included in your system, and what is not.
- The System's *Boundaries*

Mission or Goals

- What is the system supposed to do?
- How well does it need to do it?
  - Performance
- Criteria for success

Requirements

- Based on the Mission/Goals
- More detail
- Must be clear
- Must be *testable*
  - Someone else should be able to test whether your system satisfies the requirement or not

Stakeholders

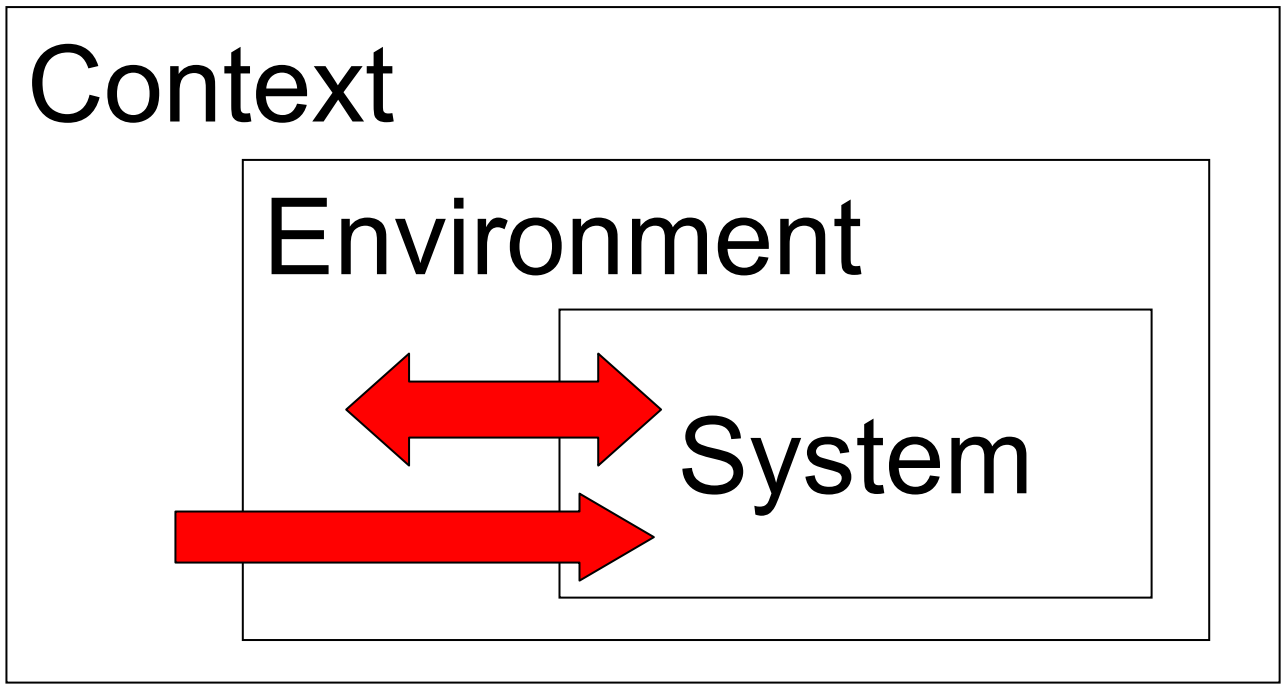
- All of the people or organizations that care about or are impacted by the system.
- Everyone who needs to have input into how the system will function or how it will be used.

Lifecycle

- Systems usually undergo a “life”
  - Starting with initial ideas and concepts
  - Through the design process,
  - Then they’re developed and tested,
  - Deployed in the field or commercial arena,
  - Maintained and operated,
  - Retired and removed from use.
- Examples where Lifecycle problems exist?

Interactions

- Every system interacts with it's environment.
- And the Context affects the System too.



Environment vs Context

- There are things outside of the system that
  - Can affect the system AND
  - Can be affected by the system.
  - This defines the *Environment* of the system.
- There are things outside the system which
  - Can affect the system BUT
  - Cannot be affected by the system.
  - This defines the *Context* of the system

Behavior

- Defines what a system needs to do or does in response to stimuli
- *Stimuli* (plural, stimulus): Various events, conditions or occurences that stimulate a reaction in the system.
- Systems are usually purchased for their behavior, not their appearance.

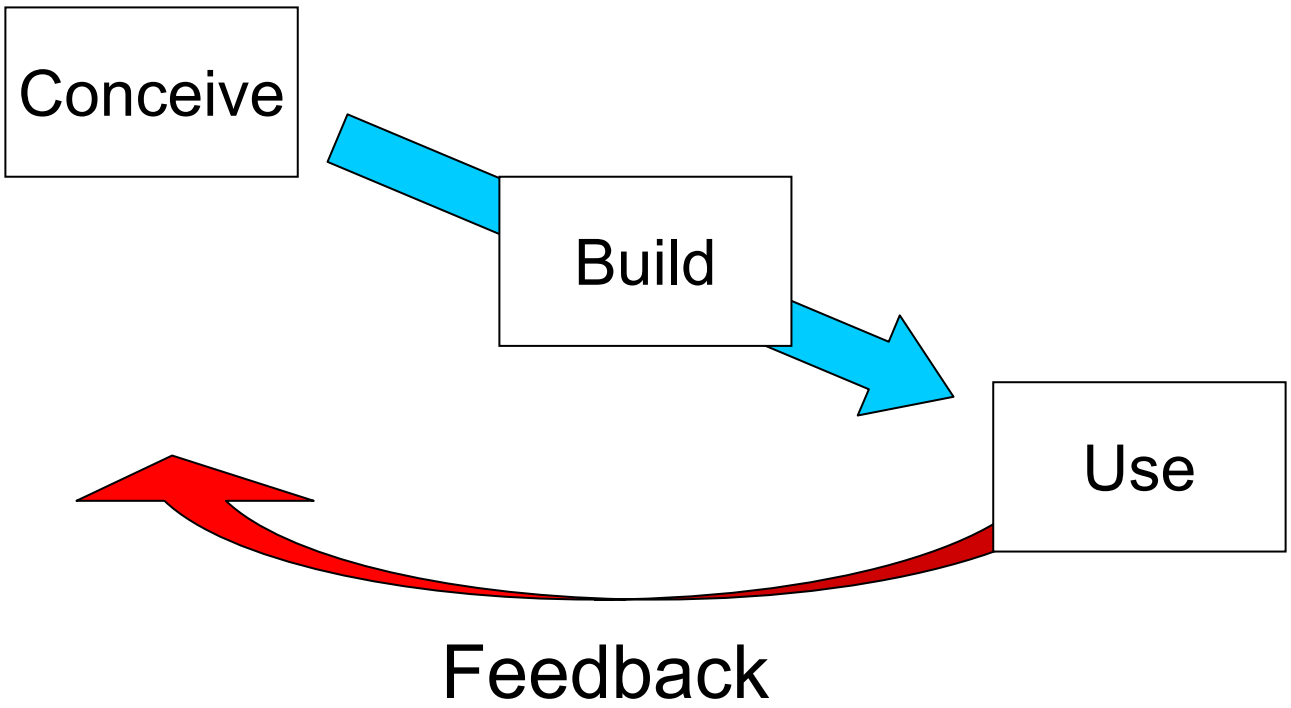
So, Systems Engineering Is:

- Learning the mental processes, tools, and ways of thinking that help you figure out all these aspects.
- Learning to apply these tools in order to develop the best system you can with the resources you have.

Functions, Feedback, House of Quality

- Functions (Process) & Feedback
- Functional Decomposition
- Issues Analysis
- House of Quality

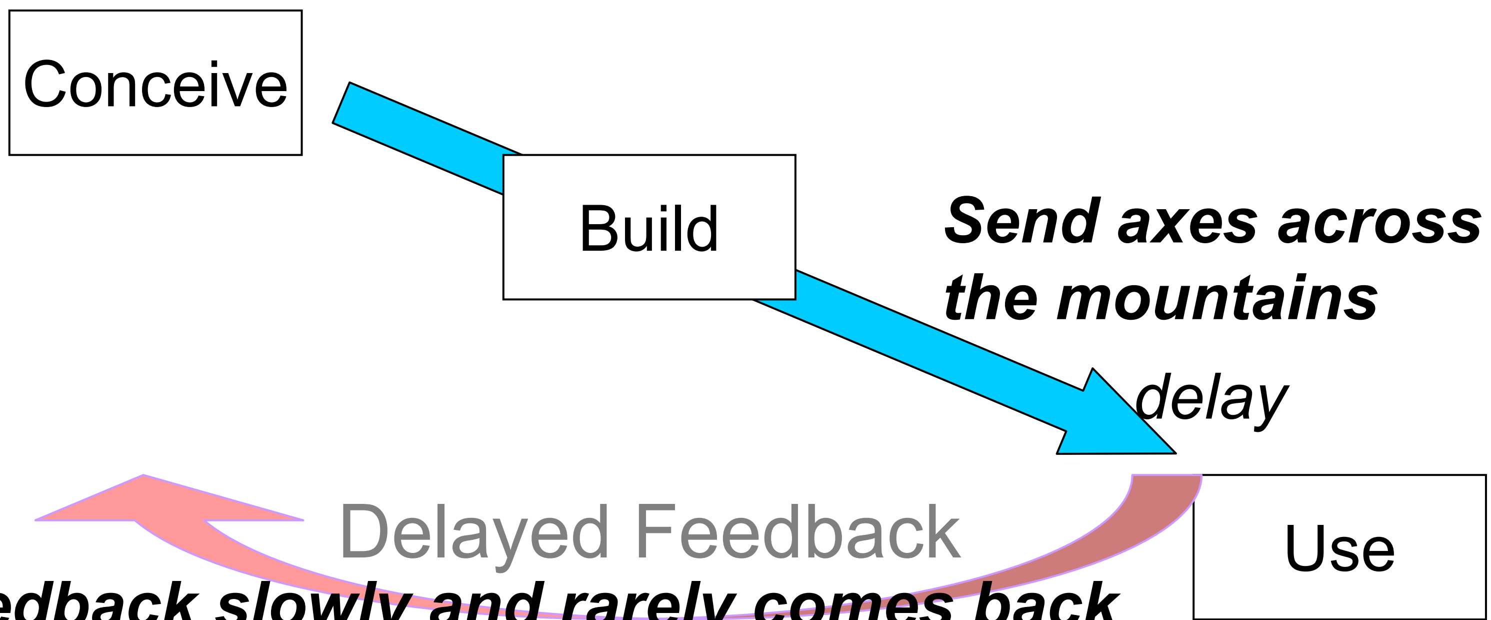
How Do Things Get Improved?



- By Feedback, where the use of the first version provides input to the second version.

Altering the Feedback Loop

- What happens when the feedback is inefficient or significantly delayed?

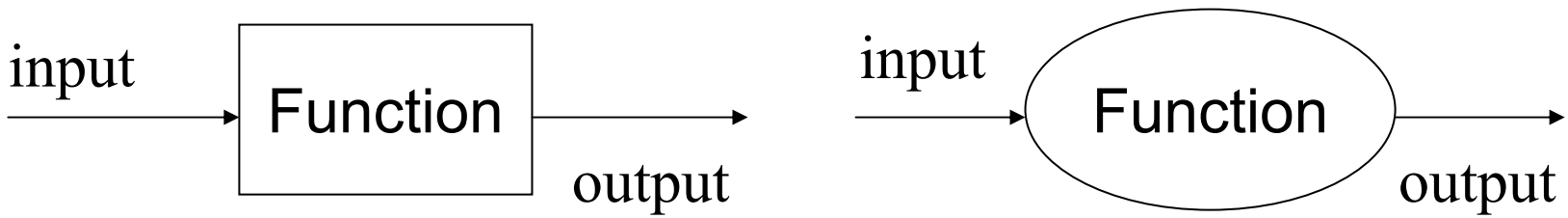


Send axes across the mountains

Feedback slowly and rarely comes back over the mountains from the users

Processes, Activities, Functions

- All are essentially mean the same thing.
- A function *does* something.
- A function has *inputs* and *outputs*.
- Often graphically represented as a box or ellipse.

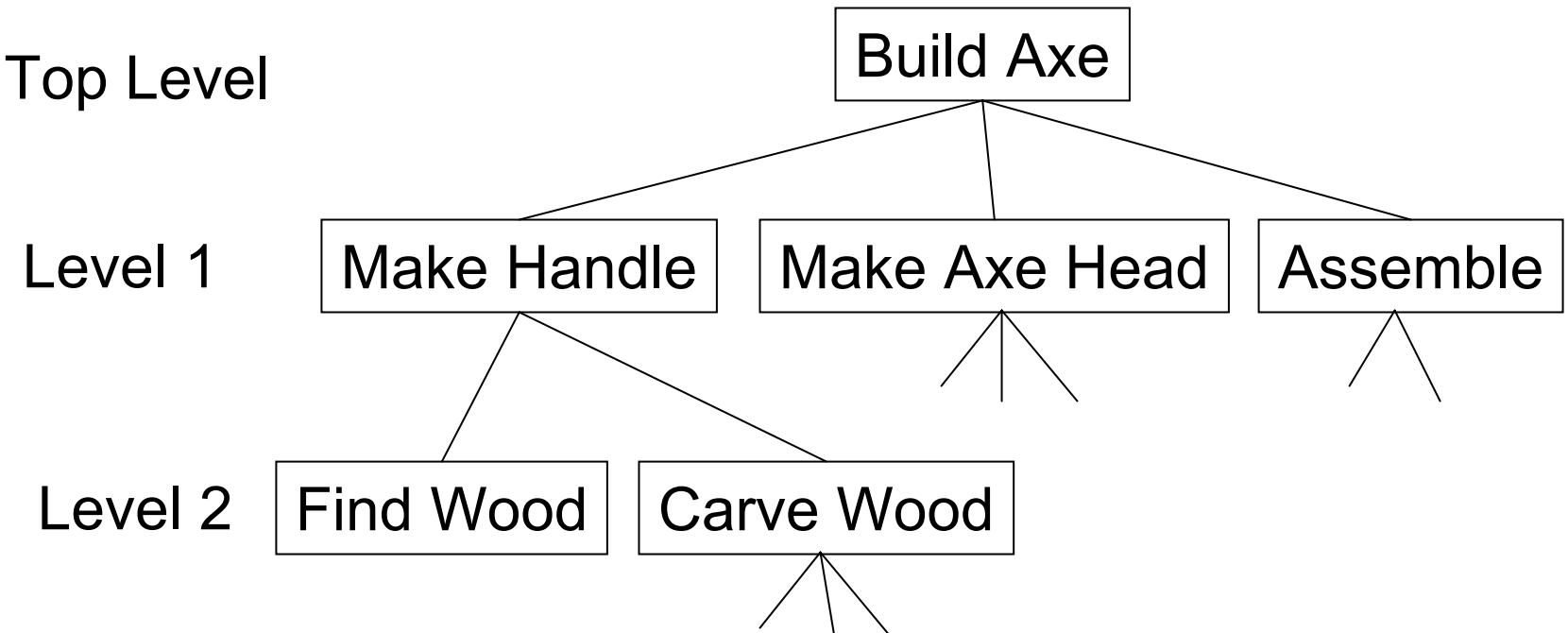


Functional Decomposition

- Any activity can usually be broken down, or decomposed, into smaller activities.
- And those are broken down into more detailed activities, and so on...
- And the result is a hierarchical "decomposition" tree of functions

Details of "Build Axe"

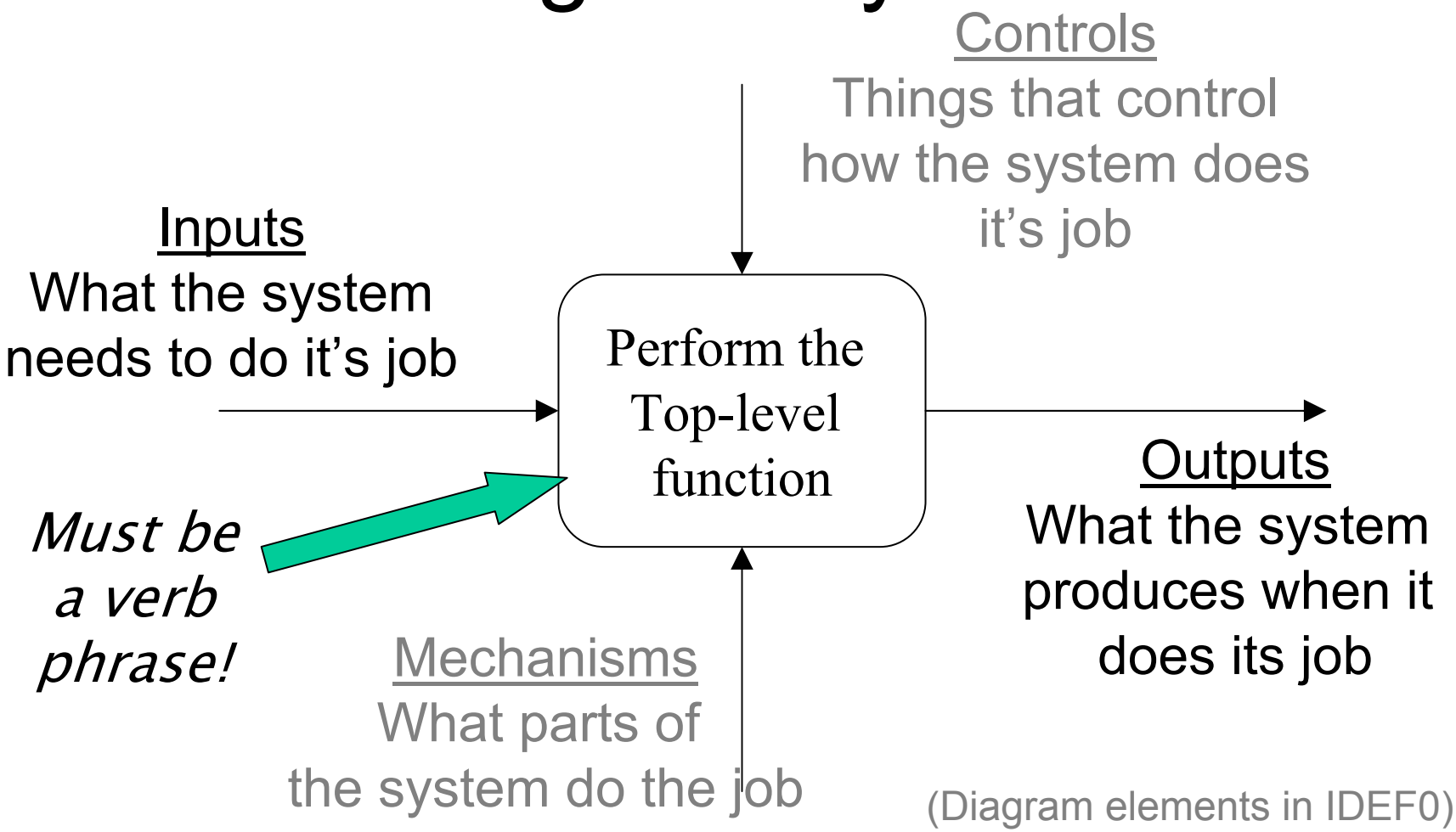
- The hierarchy branches out as it gets more detailed, resembling an upside-down tree.



Process Description

- Just knowing the functions is not sufficient
  - What's the order? What makes me decide to this OR that? When can I start doing a function?
- Dynamic descriptions are also important.

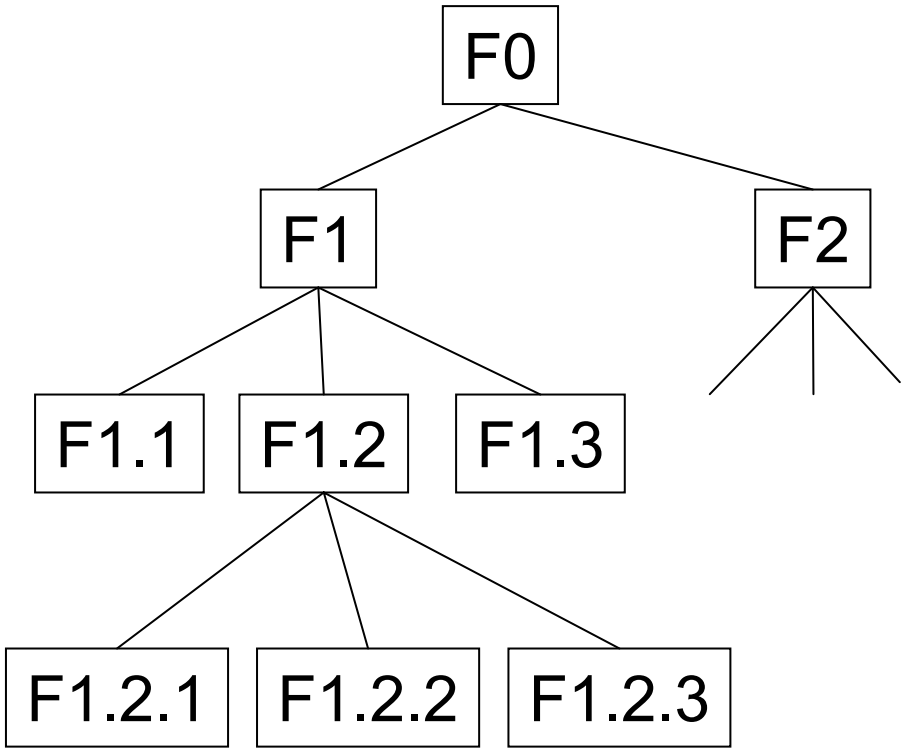
Diagram Syntax



(Diagram elements in IDEF0)

Functional Decomposition – Representation Techniques

Graphical Representation



Outline Form

- F0
  - F1
    - F1.1
    - F1.2
      - F1.2.1
      - F1.2.2
      - F1.2.3
    - F1.3
  - F2

Utility

- "Cost-Benefit"
  - A standard term used often
  - All systems have an associated "cost"
    - Cost to buy, time to learn and use, maintenance costs
  - All (good) systems have a benefit when used.
  - Is what you get out of them worth what you put in?

Issues Analysis

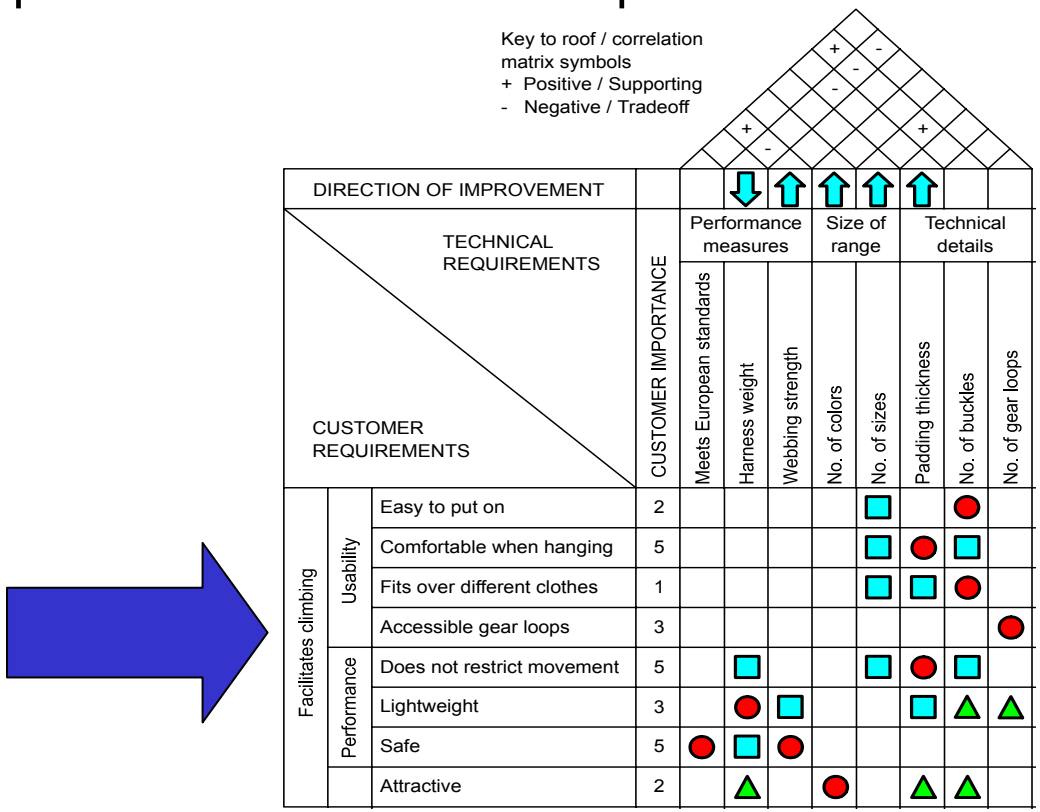
- What's important to achieving these goals?
- How does each issue (wheelbase, wheel size...) relate to the goals and subgoals?
- The goals and subgoals are often called the Customer or Stakeholder Requirements.
- The issues that help you achieve these goals are technical requirements.
- The relations from one category to the next must be kept clear throughout the lifecycle.

Issues Analysis

- Requires logical and careful thinking about the desired end result, and how you plan to get there.
- May require re-thinking your concepts and plans as you proceed.
- May require mathematical analysis or computer simulation.
  - Calculus, Analytical geometry,

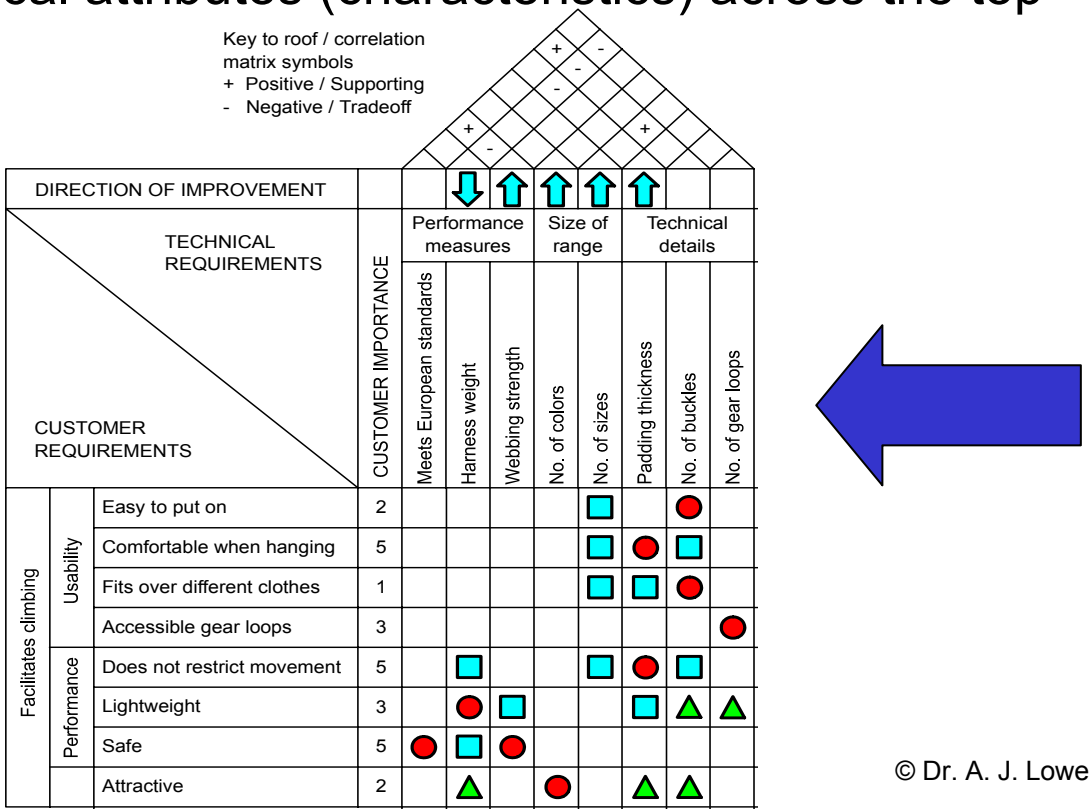
HOQ Step 1

- Step 1: List the customer requirements down the side



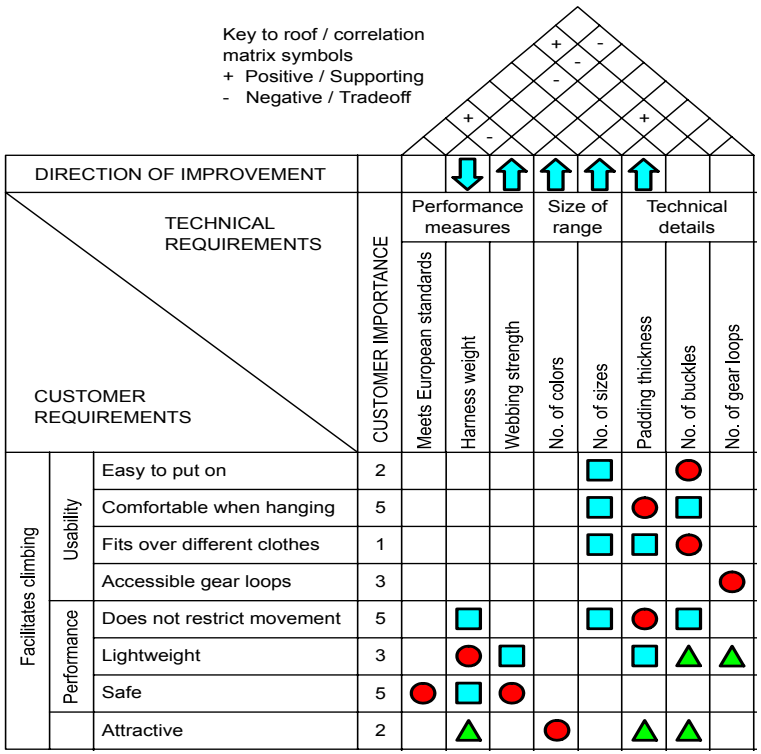
HOQ Step 2

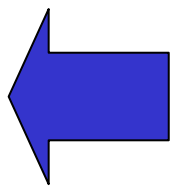
- List the Technical attributes (characteristics) across the top



HOQ Step 3

- How does each technical attribute support each customer requirement? Strongly? Weakly?





Key to interrelationship matrix symbols

- Strong interrelationship
- Medium interreltionship
- Weak interrelationship

Assumptions in Analysis

- Always starts with simplifying assumptions.
  - Solve the easy problem first, then add complicating factors and issues
- Always keep in mind your assumptions
  - You not really solving the real problem, you're solving something similar (you hope)

Issue Formulation

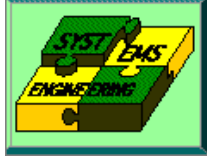
- How to determine what's important and what's not?
  - How do you know when you've captured all the issues?

Defining Desired Behavior

- One starting point: Function flow diagrams
  - Flowcharts
  - Decision processes
- Structured analysis and object-oriented techniques addressed in SYST 301 & 520.

Systems, Interfaces, Functions & Events

- If two systems can affect each other, then there must be some kind of interface between them
- Specifying an interface is a way of specifying how two systems are allowed to interact.



System Interfaces

- Systems connect to each other through interfaces
- In man-made systems, the interfaces are pretty easy to see...

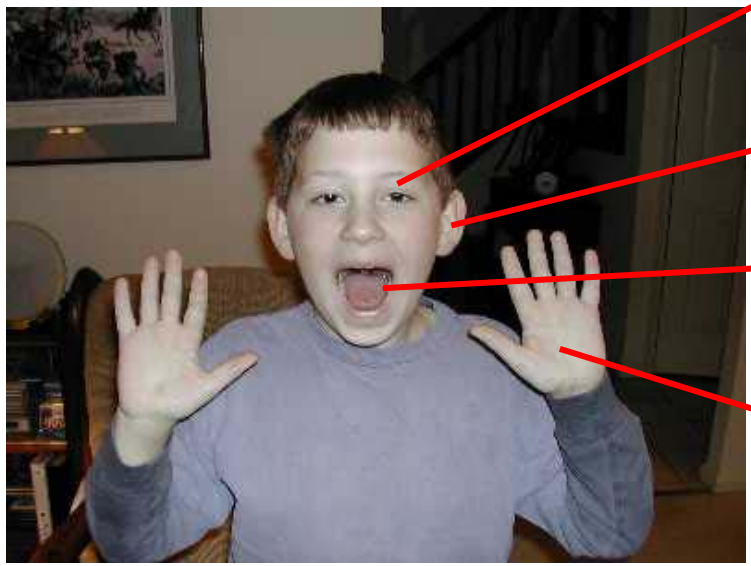
Interfaces on Common Systems

- The components of a PC have interfaces to each other

- and some of them have interfaces to you.

Interfaces to a familiar system

- Some of the interfaces to the system Human Being



visual input

audio input

food input, speech output

tactile input.
manipulation output

Events

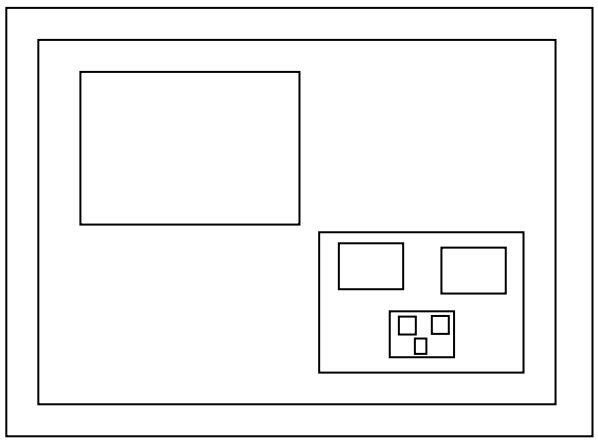
- Events are things that happen in or to your system
- Events usually have relatively short time durations.
  - Functions, on the other hand, can take a long time to perform

Example Events & Systems

- System: Road Traffic System
  - Events: Accident, Repair Activity starts
- System: PC
  - Event: Type on keyboard, move & click mouse
- System: Human Being
  - Event: Burns hand on stove, sees pretty picture, gets hungry

Internal vs External Events

- Systems are made up of sub-systems.
  - And often a system can be viewed as a sub-system to some larger system



Systems inside systems inside systems ...

Internal vs External Events

- A system can experience events that come from external systems, or can experience events that come from one of its internal systems
  - When you get hungry, your brain sub-system is responding to low-sugar signals from your endocrine system.

Events vs Functions

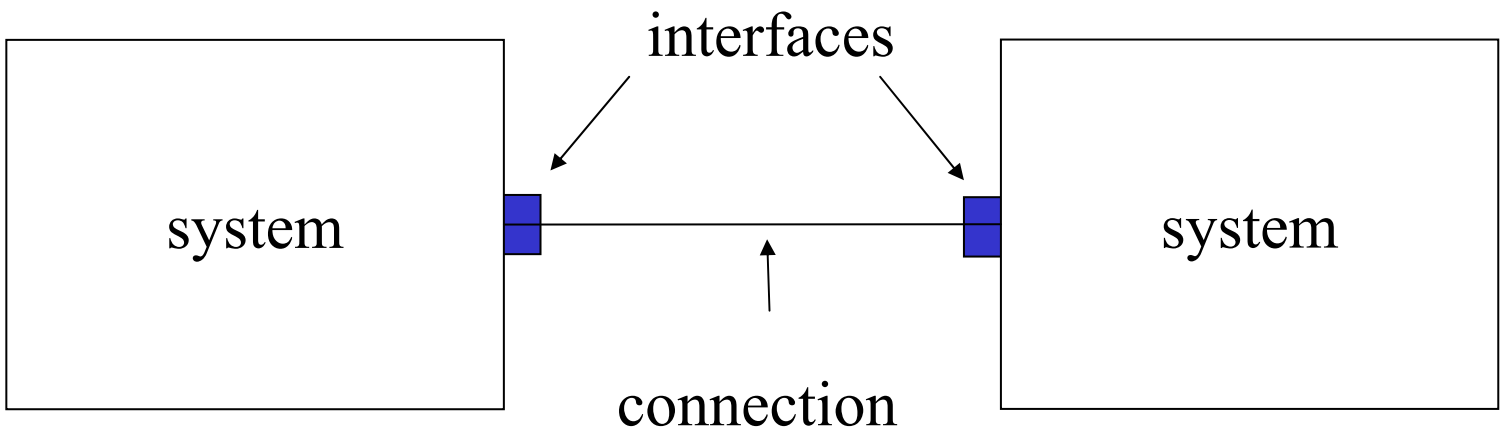
- A common modeling technique:
- Envision systems as responding to events by performing a function.
- Events “trigger” functions
- Biology: Stimulus-response

Events & Interfaces

- Systems relate to each other through their interfaces
  - Events are often “transmitted” through some sort of interface.
  - Interfaces are much easier to see in man-made systems
  - Sometimes not so easy to see in natural systems.
    - That’s what makes medicine so hard...

Events, Interfaces & Functions

- Basic modeling concept:



Stimulus-Response

- Basic general form: “When event {a} comes in over interface {i}, then do function xyz.”
  - Optional: “and send event {b} out over interface {k}.”