A Systems Engineering Toolbox

Part 1 – problem identification

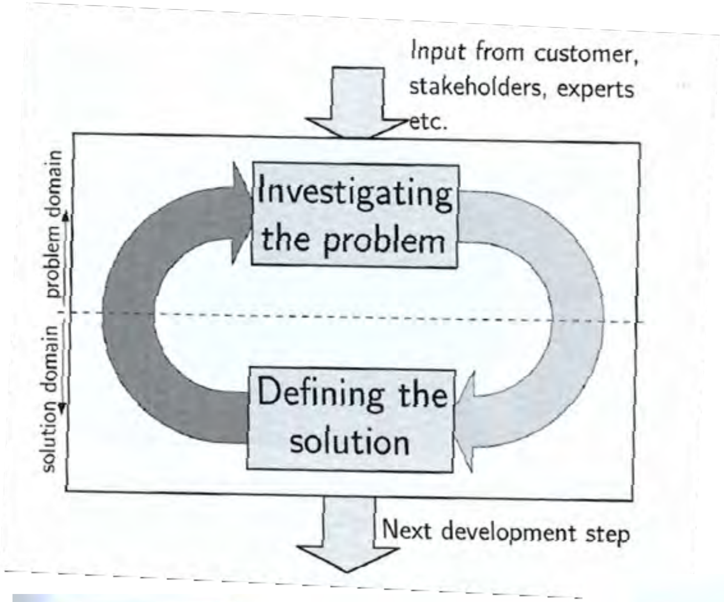
The Systems Engineering Process

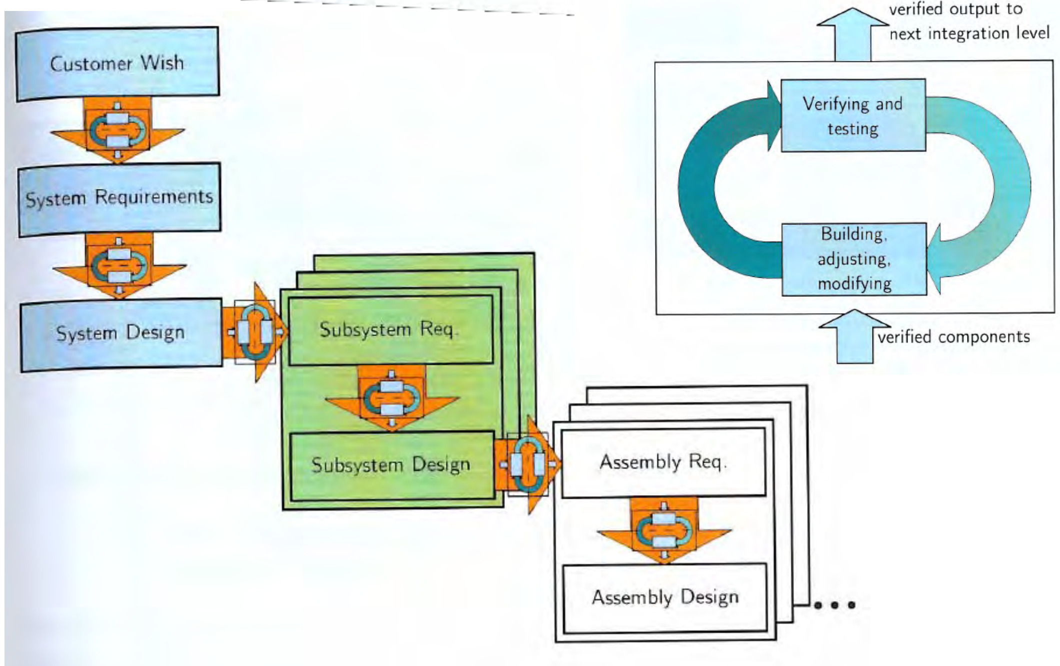
- *“Systems engineering is an interdisciplinary field of engineering and engineering management that focuses on how to design and manage* ***complex systems over their life cycles****. At its core, systems engineering utilizes systems thinking principles to organize this body of knowledge.”* (source Wikipedia)
- A formalised way of solving complex problems using engineering knowledge
- It is not bound to any one discipline
- The solution needs to meet all it’s requirements
- The solution needs to solve the problem
- The solution needs to consider its entire life cycle from concept to disposal

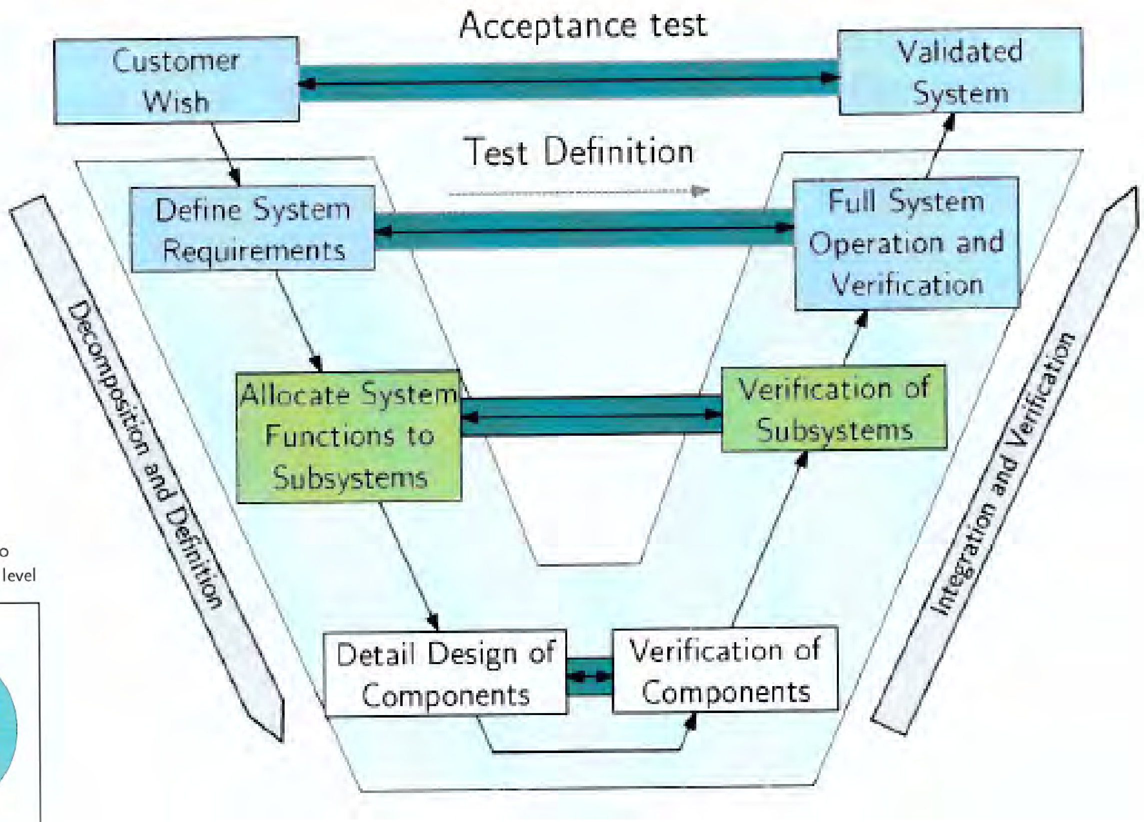
The Systems Engineering Process

1. Identify the problem
2. Explore the nature of the problem
    1. Understand how the solution will work
    2. Understand what functions the solution will perform
3. Identify the requirements of the solution
4. Break the solution down into sub-problems
    - Repeat steps 2 and 3 for each sub-problem
5. Develop solutions
6. Verify that the solution meets the requirements
7. Validate the solution actually solves the original problem.

Will focus on steps 1 – 3 today







Systems Design and Engineering, Bonnema G, Veenvliet KT and Broenink JF, CRC Press

Remember the Mars Climate Orbiter incident from 1999?

This is probably not the best way to validate a system design

CLIMATE SCIENCE RELOCATES

"Scrub that previous message Houston. There is no, I repeat no intelligent life on Mars."

Aspects of Systems thinking

- View the system as dynamic
  - Things will change with time,
  - Need to think of environmental changes over the system lifetime
  - Need to think of the system when it is an infant, consider its mid-life crisis and what happens when it is old and can no longer be repaired.
- Work from specific to generic then back to specific solution (TRIZ)
- Operational thinking – how will it work
- Apply the Scientific Method
- Hierarchical structures (nested systems)
  - Systems have sub-systems
  - Sub-systems have assemblies

Aspects of Systems thinking

- Hierarchical structures (nested systems)
  - Systems have sub-systems
  - Sub-systems have assemblies
- Organisational thinking
  - operating agency organisation
  - Manufacturer or solution provider structure
- Life-cycle thinking
- Safety thinking
- Risk thinking

Safety and Risk thinking

- Critical part of the systems engineering process
- Risk thinking considers
  - Technical aspects such as reliability and efficiency
  - Cost such as development costs, projected maintenance costs, materials, etc
  - Planning
  - External factors that change the nature of the problem
- Safety thinking considers the risks that might cause harm
- Handled by
  - Avoiding – find an alternative where the risk does not occur
  - Mitigating – take measures to reduce the effect of the risk
  - Taking – Take the risk
  - Can also delegate but this only works if there is someone else who can take the blame. (outsourcing)

An example of Risk reduction in a complex system

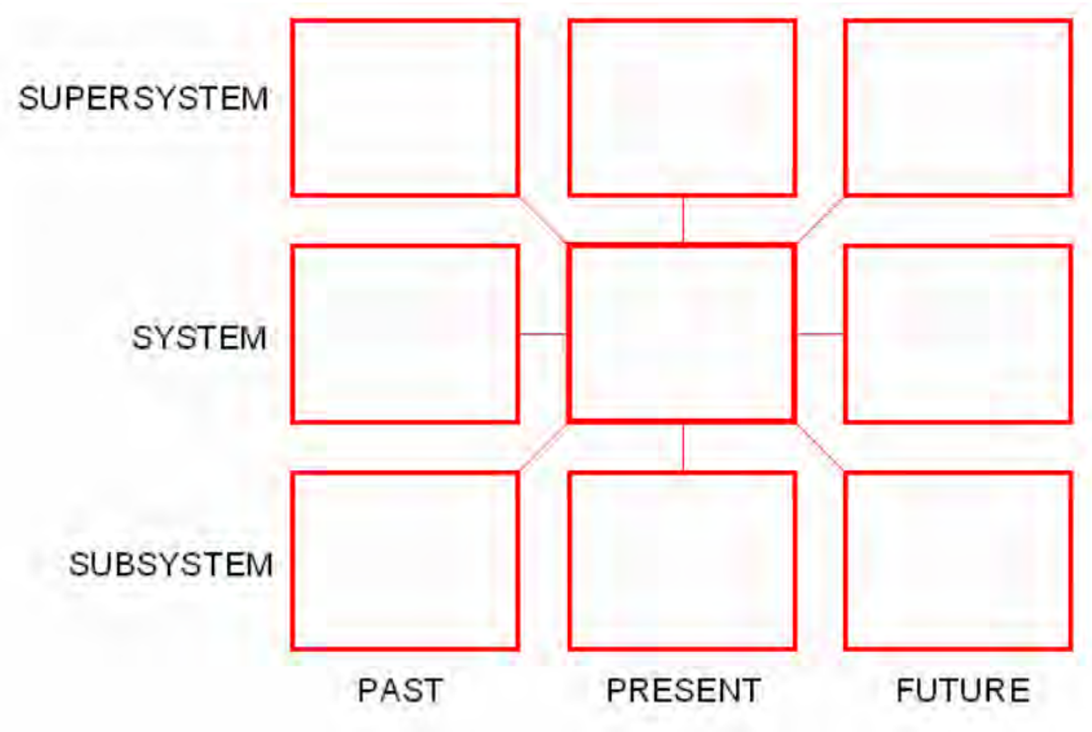
First flight 2 Feb 1974, designed to be unstable in pitch. First use of Fly-by-wire in a system with quadruple redundancy.

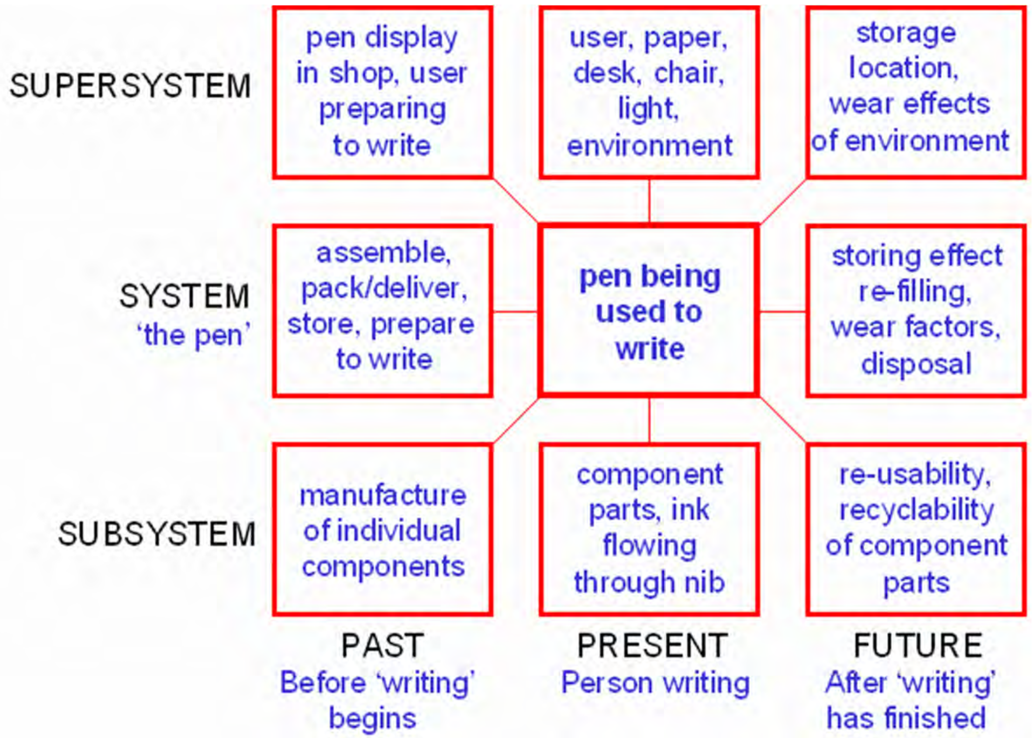
First flight 1 Dec. 1977, designed to the limits of computing power in the '70's. Design uses the FCS from the F16, Engines from the T38, landing gear from the A10 and life support system from the C130.

System design tool A – 9 Windows

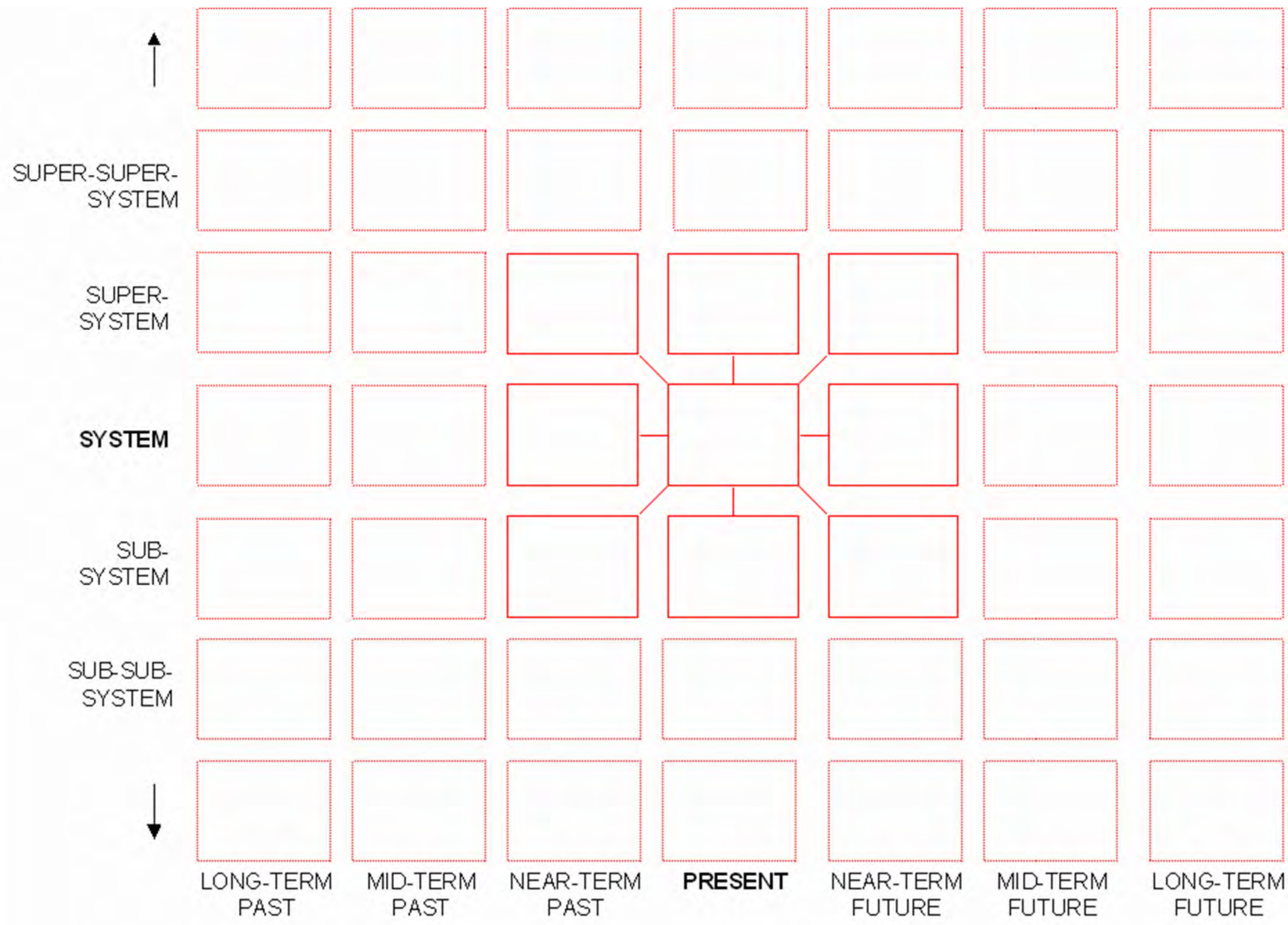
- Also called the System Operator Diagram (TRIZ)
- Puts the system/solution into its hierarchical and temporal context
- Helps tell the story of how the system is seen to work
- Looks at historical context of the system
- Looks at hierarchy of the system
- Identifies the most important goals of the system

The 9 Windows diagram



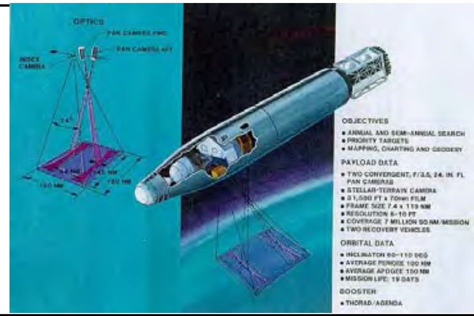
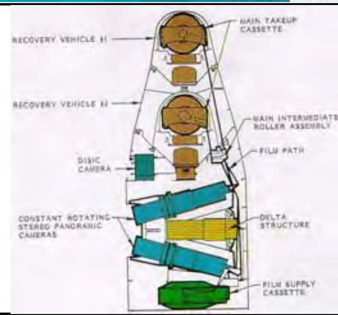


https://triz-journal.com/system-operator-tutorial-1-9-windows-world/



SUPER-SUPER-SYSTEM
SUPER-SYSTEM
SYSTEM
SUB-SYSTEM
SUB-SUB-SYSTEM
LONG-TERM PAST
MID-TERM PAST
NEAR-TERM PAST
PRESENT
NEAR-TERM FUTURE
MID-TERM FUTURE
LONG-TERM FUTURE

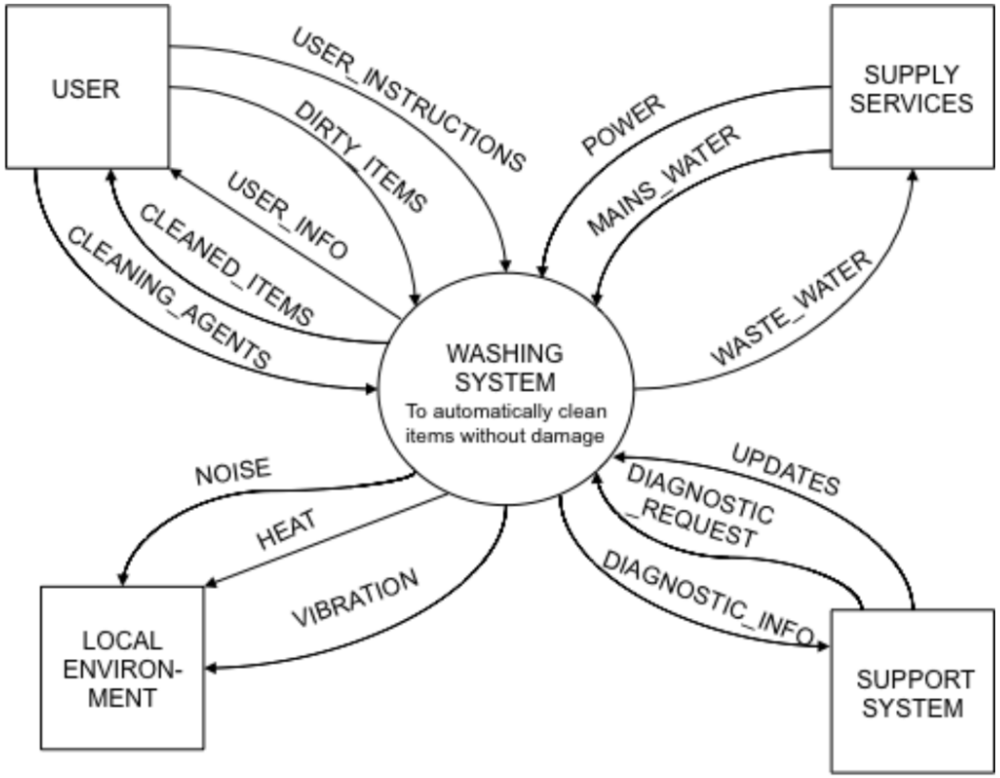
9 Windows Diagram – Surveillance System

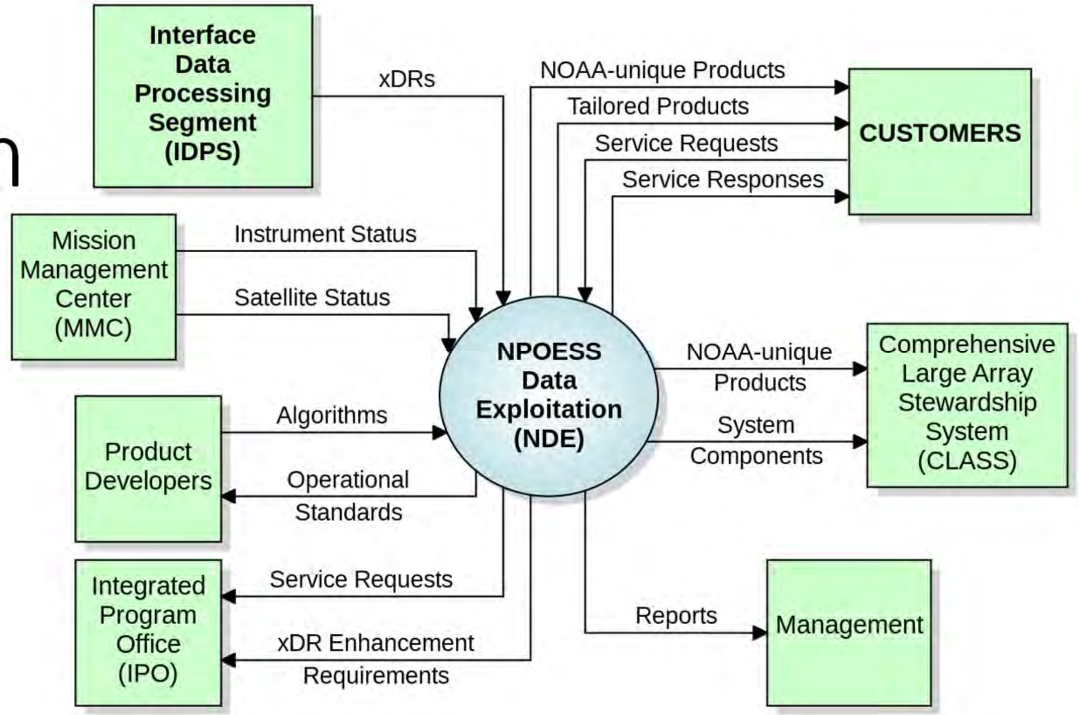
	Past	Now	Future
Operator	**Strategic air command, CIA** **Understand nuclear threat posed by USSR**	**Strategic air command, CIA** **Understand nuclear threat posed by USSR**	**Strategic air command, CIA and NSA** **Understand nuclear threat posed by USSR**
The System			
The Payload			

System Design Tool B – Context Diagrams

- The system context diagram defines the boundary between the system, or part of a system, and its environment, showing the entities that interact with it.
- The context diagram attempts to identify all the parameters or factors that can affect the system but which are not part of the system
- Stakeholder diagrams are context diagrams that focus on people and organisations that interact with the system
- The context diagram includes aspects such as other systems, infrastructure, environmental factors and generally all parameters that cannot be changed
- The stakeholder and context diagrams can be done separately or on the same diagram.

Sample Context Diagram





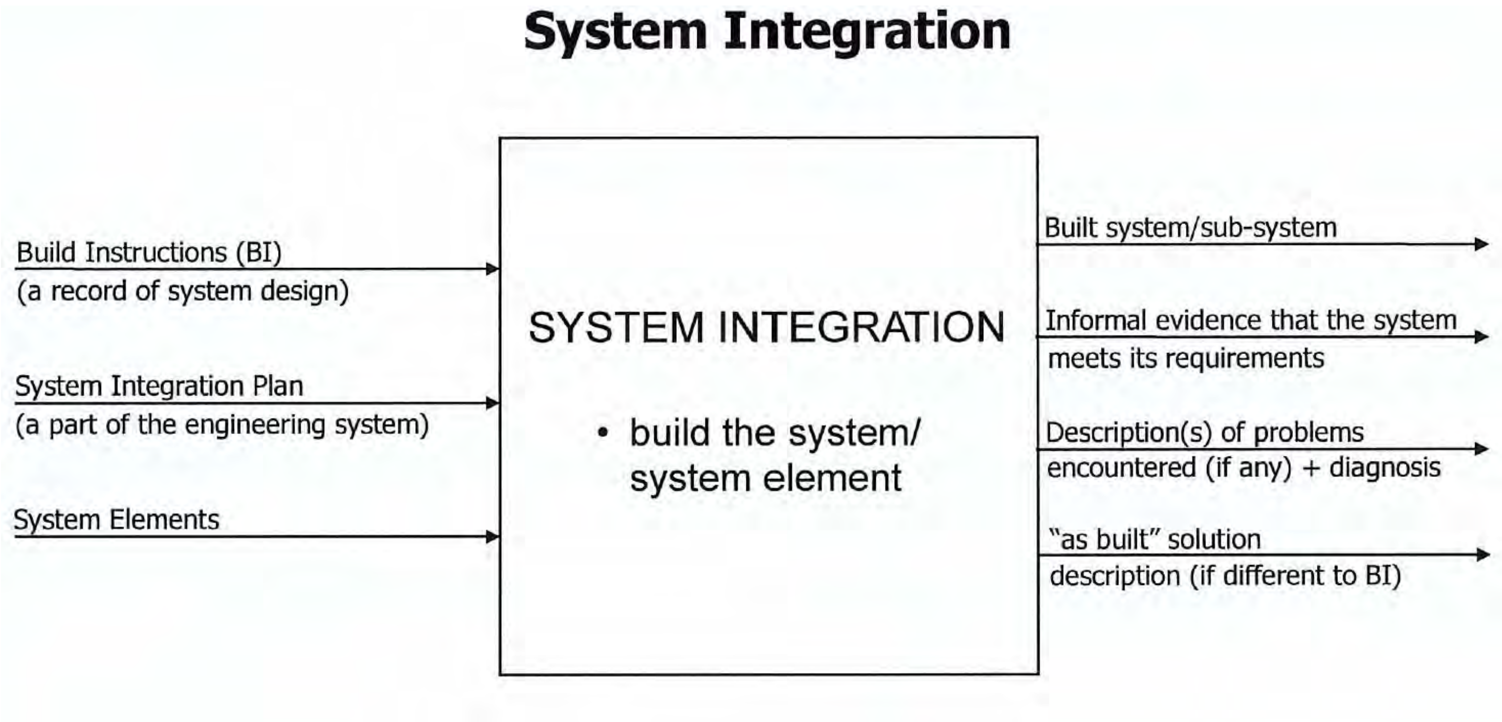
Shows physical entities (objects, users, etc), functions (supply services), concepts (local environment) and the flows of information, energy or matter.

Guidelines on drawing context diagrams can be found at:
https://www.burgehugheswalsh.co.uk/uploaded/1/documents/cd-tool-box-v1.0.pdf

Scenario writing

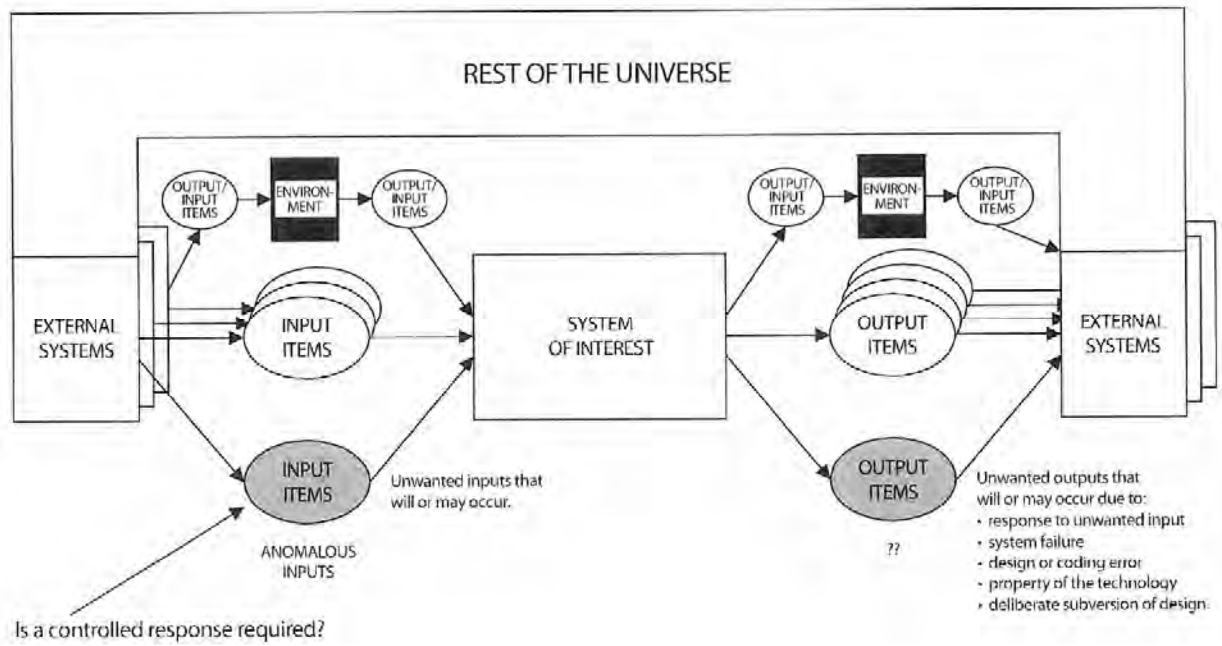
- Also referred to as story boarding
- Tells the story of how the system will be used
- Usually there are a collection of stories that are told from the perspective of the person that operates the system, from the user's perspective, the maintainer, etc
- Story can include pictures but avoids jargon and keeps language simple.
- Scenarios should be discussed with experienced users and other stakeholders to ensure that they consider all use cases
- Try as far as possible to imagine what the operators will see and do
- Confirm that the scenarios ultimately address the original problem.

System Diagrams



System diagrams help to identify interfaces for individual systems or system elements. The diagram includes all information, energy and matter that crosses a system boundary.

Understanding Interfaces



By understanding the nature and context of the problem, it is possible to identify all the interfaces to the system.

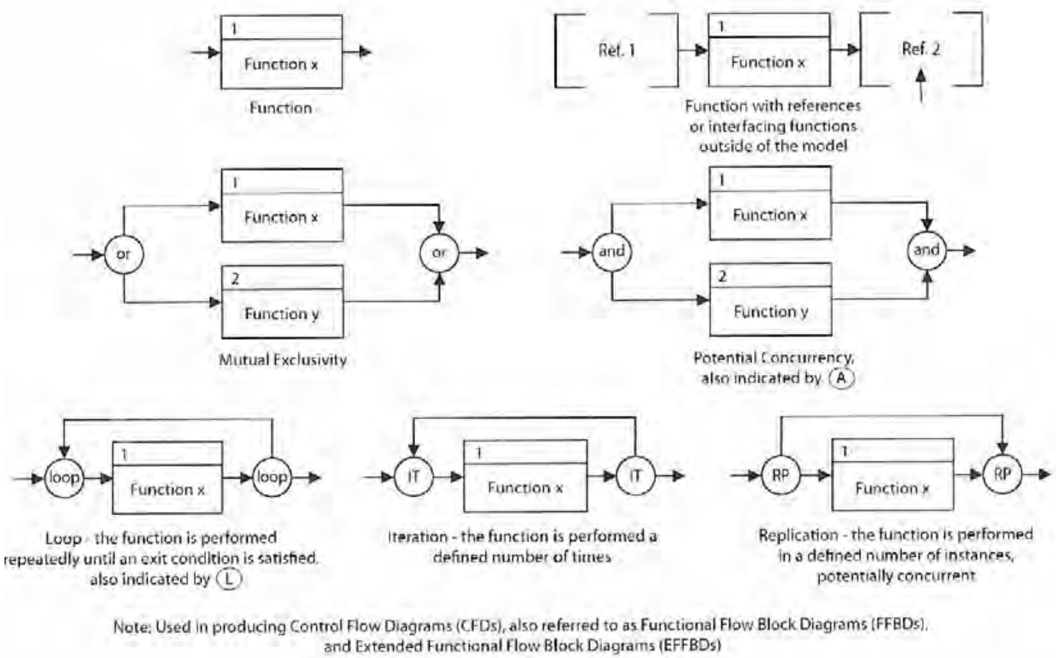
By understanding its functioning, it is possible to understand the interactions of the system with its context.

Only when both are understood, then requirements follow.

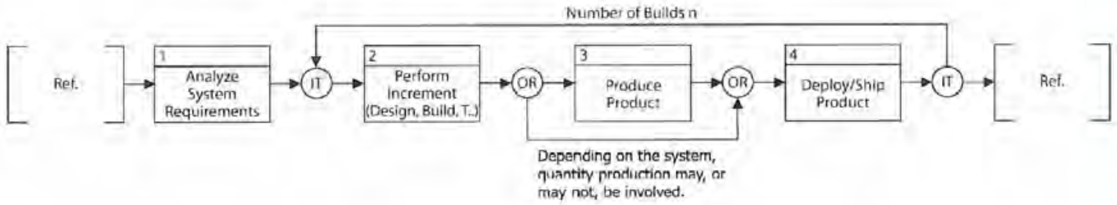
Functional Analysis and Modelling

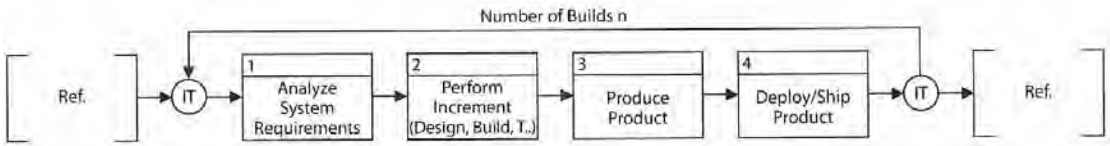
- Used at several stages in the systems engineering process
- Consists of:
  - Functional trees and mind maps
  - Influence diagrams
  - Functional block diagrams
  - State transition diagrams
- Used in the problem definition phase to interpret and analyse the story boards and define requirements

Control and Flow Diagrams

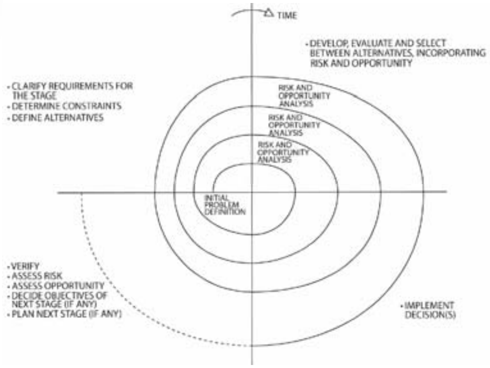


Examples of Development Cycles





Agile software ~~destruction~~ development



Always spiral outwards to the solution

Functional design

- The first part of the design process is the functional design
- Top level, understand how the system will be used
  - Burn a piece of bread
  - Take a photo of a Martian
  - Create AI’s that will rule the world…
- First design decision
  - Buy a new one?
  - Design a new one? (Green fields or clean sheet of paper…)
  - Modify an existing design?
- The user interaction with the first pass of the functional design will result in several important user requirements.