Aalto University
School of Engineering

MEC-E2000 Marine and Ship Systems Engineering

Lecture 2: Foundations of systems engineering and its connection to marine and ship systems

Osiris A. Valdez Banda
Assistant Professor

31.10.2022

Lectures

Propulsion plant management

Auxiliary power management

Auxiliary machinery operation

Navigation and maneuvering

IT and comm. systems

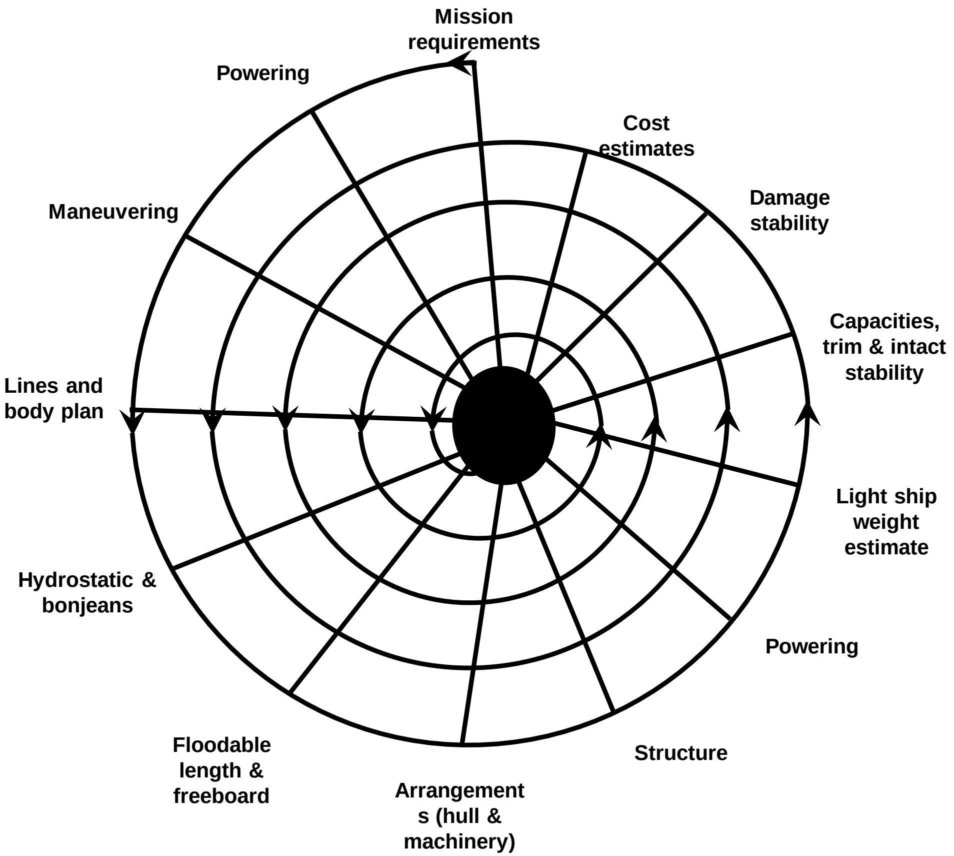
Ballast and trim management

Cargo handling operations

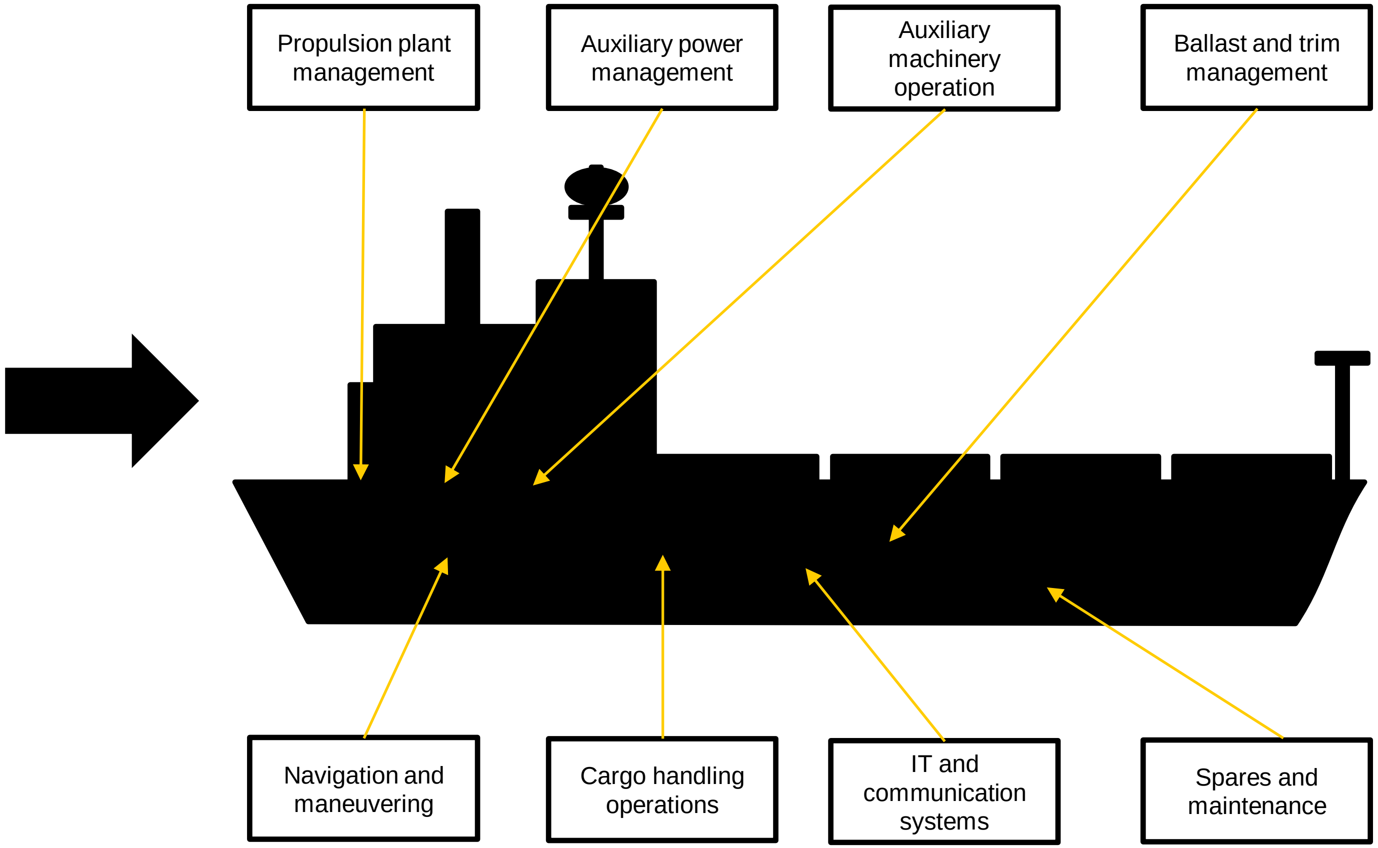
Spares and maintenance

Topic	Time	Lecturer	Affiliation
Introductory lecture	24/10/22	Osiris Valdez Banda	Aalto University
System Engineering	31/10/22	Osiris Valdez Banda	Aalto University
Ship Energy sources	7/11/22	Mia Elg	Deltamarin
Machinery auxiliary systems	14/11/22	Antero Apajalahti	Helsinki Shipyard
HVAC and ballast water treatment systems	21/11/22	Vesa Heikkilä/ Konstantin Tchetchine	Meyer Turku/ Wärtsilä
Safety Systems	28/11/22	Victor Bolbot	Aalto University
Propulsion Systems	09/01/23	Ari Rakkola	Meyer Turku
Deck Machinery and Cargo Handling Systems	16/01/23	Osiris Valdez Banda	Aalto University
Electricity consumption and Power Management System	24/01/23	Mia Elg	Deltamarin
Integrated Bridge and IT systems	31/01/23	Kalevi Tervo	ABB

The perspective and evolution of ship systems in ship design



The traditional ship design spiral by Evans 1959



The design of the next generation of digital ships by Martin Stopford 2018

Intended Learning Outcomes (ILOs)

By this lecture, students will be able to:

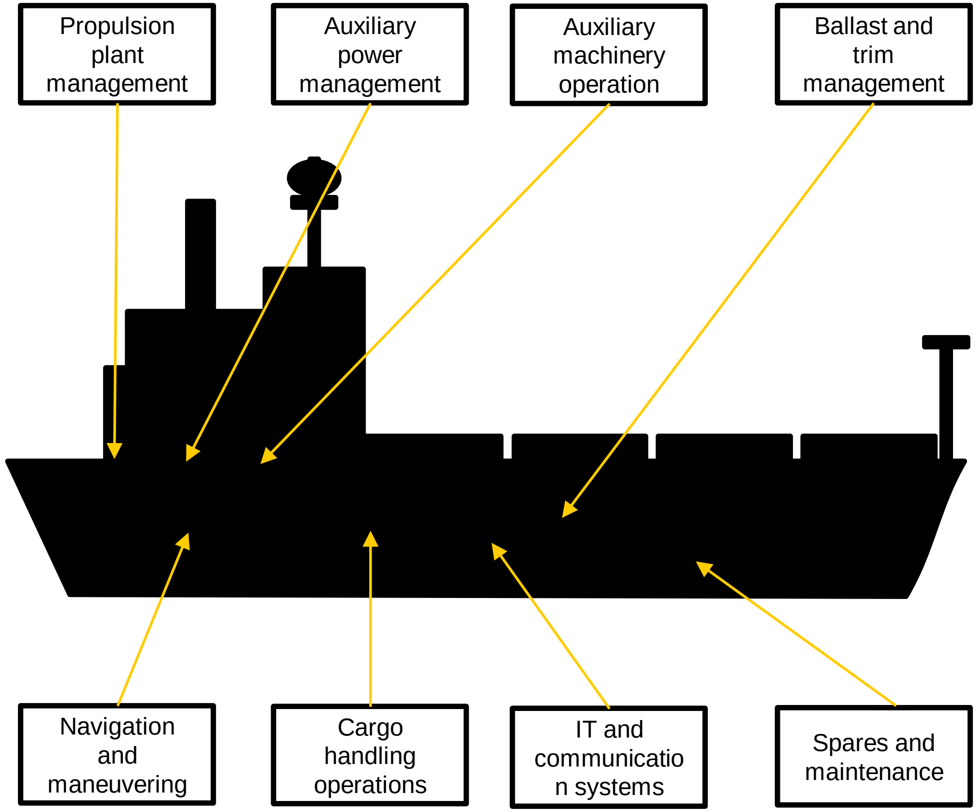
- Understand what systems engineering is, the role of a systems engineer, and how systems engineering relates to the design of ship systems.
- Identify the different components of systems engineering and how those are co-related in the design and operation of ships
- Understand the systematic characteristics of the system engineering process in large projects
- Understand the procurement process of systems engineering and its relation to ship design

What is a system?

- A purposeful collection of interrelated components working together towards some common objective.
- A system may include software, mechanical, electrical and electronic hardware and be operated by “people”.
- System components are dependent on other system components
- The properties and behaviour of system components are inextricably inter-mingled

Systems Engineering (1)

- It is not fundamental mathematics or strict laboratory science
- It is a mix of HR, project management, business, rational decomposition, trade studies, requirements traceability, integration, testing, verification and validation, operations, and end-of-life cycle disposal of systems
- Standardizes the flow-down and traceability of specifications for complex products from customer requirements through production, operation, and disposal.



The design of the next generation of digital ships by Martin Stopford 2018

Systems Engineering (2)

Systems Engineering is an "*interdisciplinary approach*" and means to enable the realization of successful systems.

It focuses on defining *customer needs* and required functionality early in the development cycle, documenting requirements, and then proceeding with design synthesis and system validation while considering the complete problem:

Problem scope:
- Operations
- Performance
- Test
- Manufacturing
- Cost & Schedule
- Training & Support
- Disposal

Systems Engineering (3)

Systems Engineering **integrates** disciplines and specialty groups into a team effort forming a structured development process that proceeds from concept to production to operation.

Systems Engineering considers both the business and the technical needs of all customers with the goal of providing a **quality product** that meets the user needs

Why we need systems engineering

- Rapid technological changes
- Development of advanced technologies
- Constant changes in the operational environment

Effect:

- Embedded software in components (software errors)
- Increased component interactions (errors/risks due to wrong/unsafe interactions)
- Increased complexity (design and operational errors)
- Higher number of components. (large and disrupted models)

Amerigo Vespucci (Italian Navy)

Symphony of the Seas (Royal Caribbean)

Autonomous Ship Concept (Rolls-Royce)

Relevancy of systems engineering in ship design

Many system designers use traditional System Engineering tools and techniques, such as

- Requirements derivation and rationale,
- Functional allocation and decomposition
- Trade studies
- Technical performance measurements
- *Risk analyses*
- **Spiral development**, etc.

The key question is how to achieve simple-yet-elegant, almost-beautiful system designs when faced with extremely complex system design problems (2).

Oasis of the Seas (Royal Caribbean)

The Role of the System Engineer

Any engineer acts as a systems engineer when responsible for the design and implementation of a total system.

The **difference** with “traditional engineering” lies primarily in the greater emphasis on defining goals, the creative generation of alternative designs, the evaluation of alternative designs, and the coordination and control of the diverse tasks that are necessary to create a complex system.

The role of the Systems Engineer is one of the Manager that utilizes a **structured value delivery process.**

Systems Engineering Methodologies

- **The "V" model of system development**

- **The spiral model of system development**

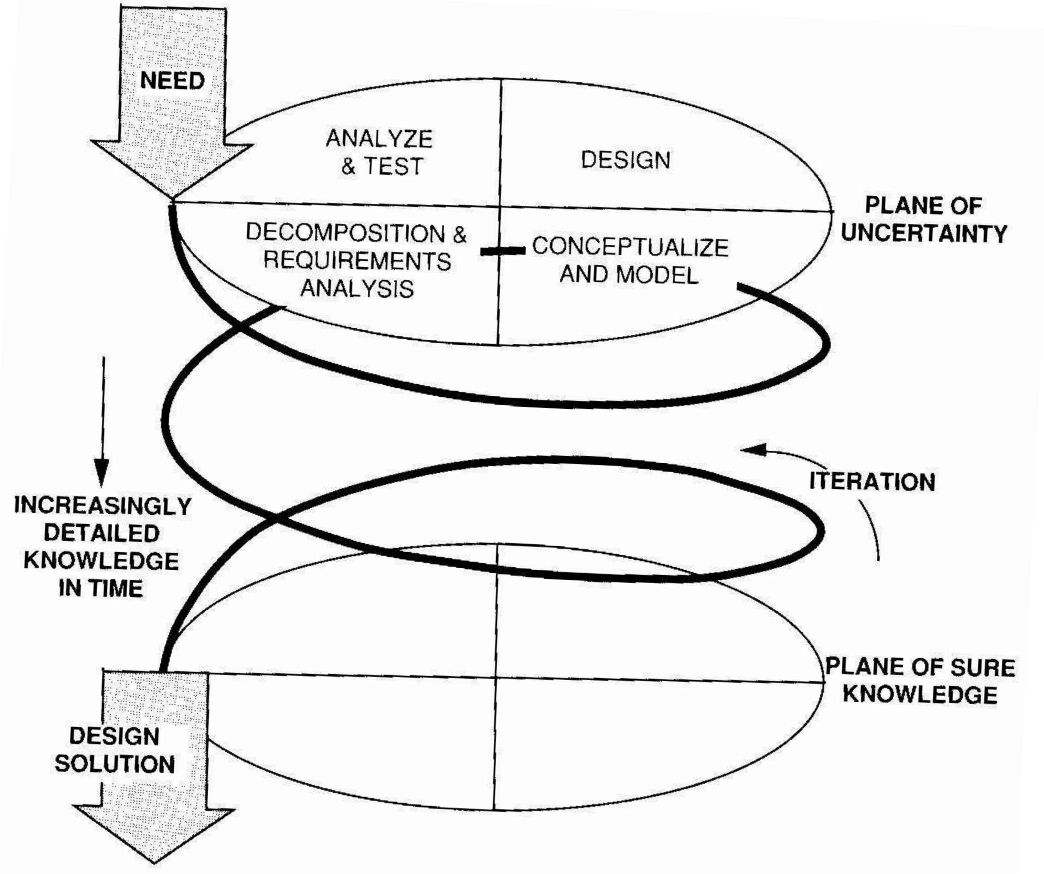


Figure 1-3 The spiral model of system development.

Challenges of systems engineering

Large systems are usually designed to solve 'wicked/complex' problems

Systems engineering requires a great deal of co-ordination across disciplines

- Almost infinite possibilities for design trade-offs across components
- Mutual distrust and lack of understanding across engineering disciplines

Systems must be designed to last many years in a changing environment

Emergent properties

Properties of the system as a whole rather than properties that can be derived from the properties of components of a system

Emergent properties are a **consequence of the relationships** between system components

They can therefore only be **assessed and measured** once the components have been integrated into a system

Types of emergent property

Functional properties

- These appear when all the parts of a system work together to achieve some objective. For example, **a ship has the functional property of being a large transportation mean once it has been properly designed and built. For this, it requires the assembling of all its components.**

Non-functional emergent properties

- Examples are **reliability, performance, safety, and security**. These relate to **the behaviour of the system in its operational environment**.

Examples of emergent properties

The overall weight of the system

- This is an example of an emergent property that can be computed from individual component properties.

The reliability of the system

- This depends on the reliability of system components and the relationships between the components.

The usability of a system

- This is a complex property which is not simply dependent on the system hardware and software but also depends on the *system operators and the environment where it is used*.

System reliability engineering

Because of component inter-dependencies, faults can be propagated through the system

System failures often occur because of unforeseen inter-relationships between components

It is probably impossible to anticipate all possible component relationships

Software reliability measures may give a **false** picture of the system's reliability

Influences on “reliability”

Hardware reliability

- What is the probability of a hardware component failing and how long does it take to repair that component?

Software reliability

- How likely is it that a software component will produce an incorrect output? Software “failure” is usually distinct from hardware failure in that software does not wear out.

Operator reliability

- How likely is it that the operator of a system will make an error?

The ‘shall-not’ properties

Properties such as performance and reliability can be measured

However, some properties are properties that the system should not exhibit

- Safety - the system should not behave in an unsafe way
- Security - the system should not permit unauthorised use

Measuring or assessing and managing these properties is **very hard**

Systems and their environment

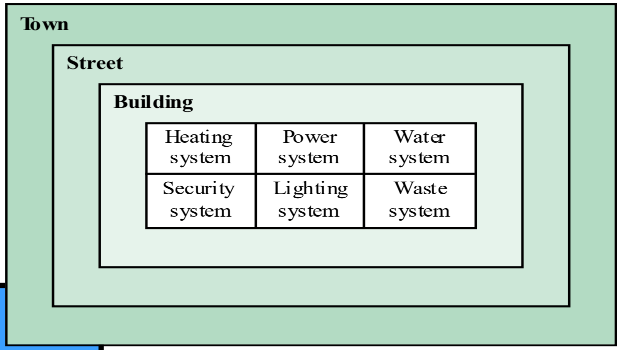
Systems are not independent but **exist in an environment**

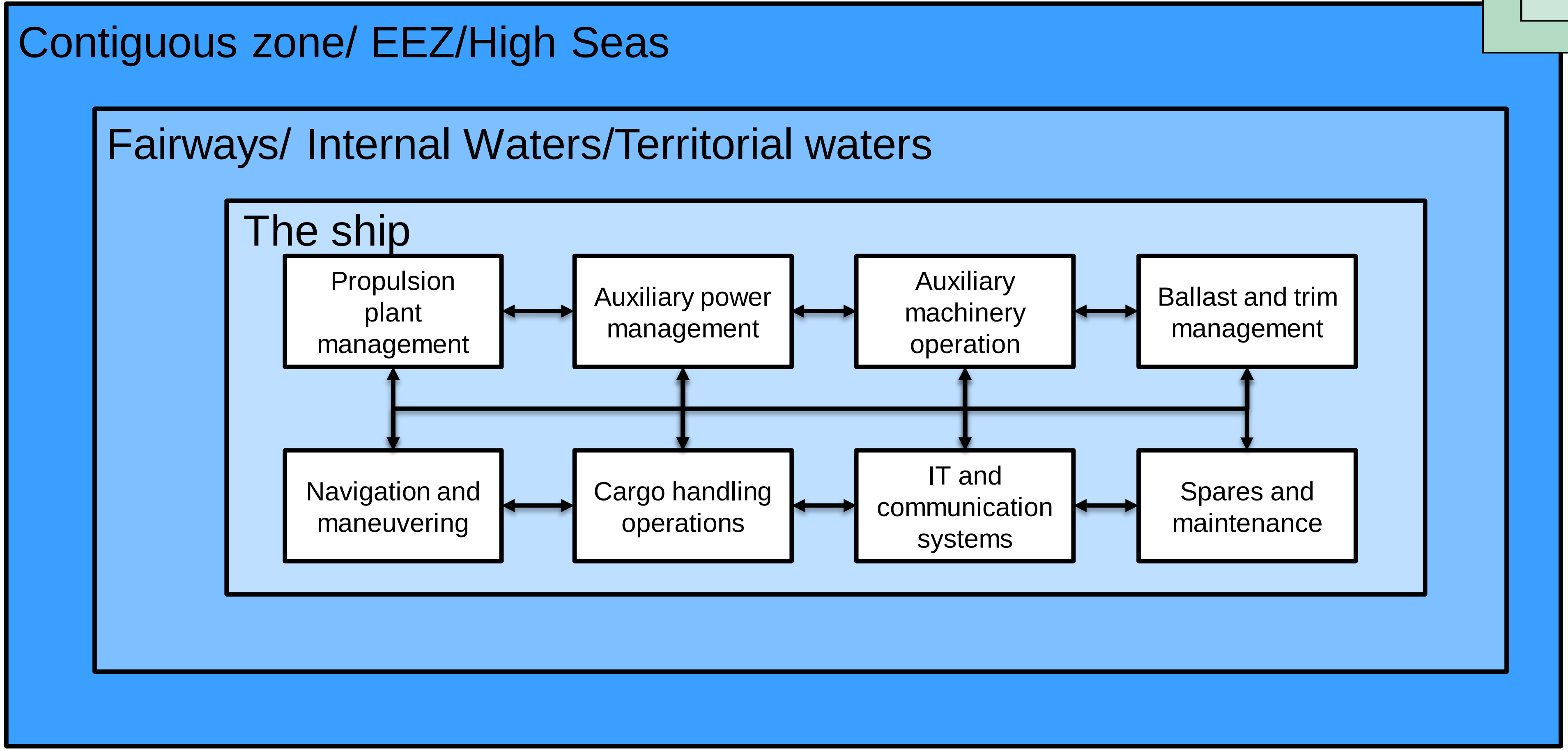
The system’s function may be to change **its environment**

Environment affects the functioning of the system e.g. system may require electrical supply from its environment

The **organizational aspects** as well as the physical environment may be important

System hierarchies (1)





Regulatory requirements, guidelines, acts, etc. at all levels

Human and organisational factors

Process changes

- Does the system require changes to the work processes in the environment?

Job changes

- Does the system de-skill the users in an environment or cause them to change the way they work?

Organisational changes

- Does the system change the political power structure in an organisation?

System architecture modelling

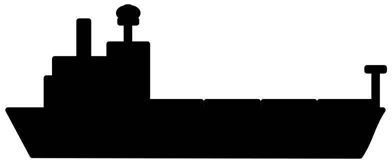
An architectural model presents an **abstract view** of the sub-systems making up a system

May include major **information flows** between sub-systems

Usually presented as a **block diagram**

May identify different types of **functional components** in the model

Ship systems



Propulsion plant management

Auxiliary power management

Auxiliary machinery operation

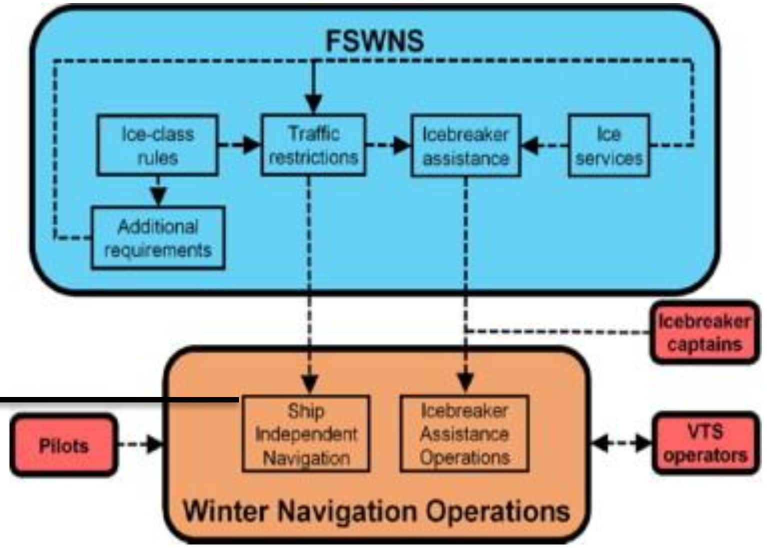
Navigation and maneuvering

IT and comm. systems

Ballast and trim management

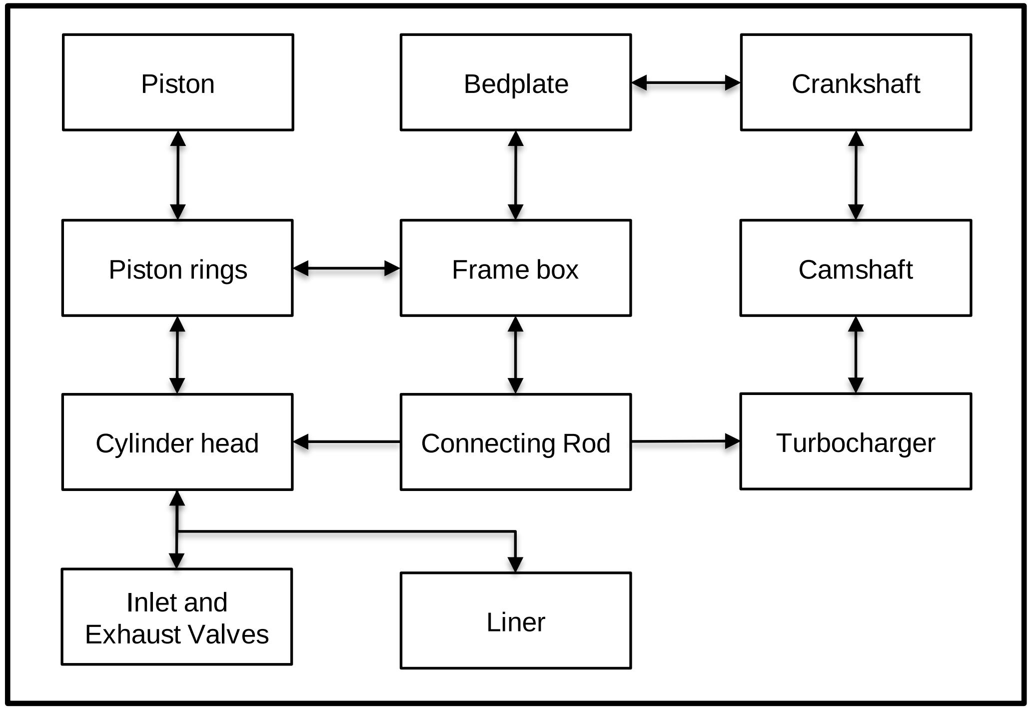
Cargo handling operations

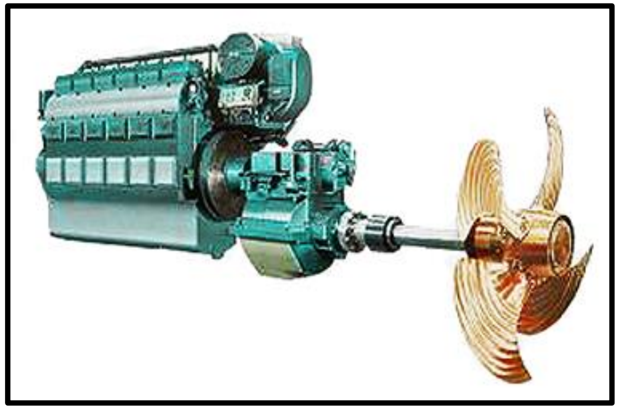
Spares and maintenance



FSWNS by Valdez Banda et al. 2015

Marine Diesel Engine (System Architecture):





4-Stroke Marine Diesel Engine System

Aalto University
School of Engineering

Functional system components

Functional system components

- Sensor components
- Actuator components
- Computation components
- Communication components
- Co-ordination components
- Interface components

System components (1)

Sensor components

- Collect information from the system’s environment e.g., radars linked to the ship navigation system

Actuator components

- Cause some change in the system’s environment e.g., valves in a process control system which increase or decrease material flow in a pipe (e.g., in a ballast water system)

Computation components

- Carry out some computations on an input to produce an output e.g., MSI as part of the Global Maritime Distress and Safety System (GMDSS)

System components (2)

Communication components

- Allow system components to communicate with each other e.g., network linking distributed computers

Co-ordination components

- Co-ordinate the interactions of other system components e.g., manoeuvring control system

Interface components

- Facilitate the interactions of other system components e.g., Electronic Chart Display Information System

Many components are now commonly software controlled

Group task (7 minutes)

List and discuss with other members of your “Ship Concept” group other system components:

Type	Component
Sensor	Collect information from the system’s environment e.g., radars linked to the ship navigation system
Actuator	Cause some change in the system’s environment e.g., valves in a process control system which increase or decrease material flow in a pipe (e.g., in a ballast water system)
Computation	Carry out some computations on an input to produce an output e.g., MSI as part of the Global Maritime Distress and Safety System (GMDSS)
Communication	Allow system components to communicate with each other e.g., network linking distributed computers
Coordination	Co-ordinate the interactions of other system components e.g., maneuvering control system
Interface	Facilitate the interactions of other system components e.g., Electronic Chart Display Information System

The system engineering process (1)

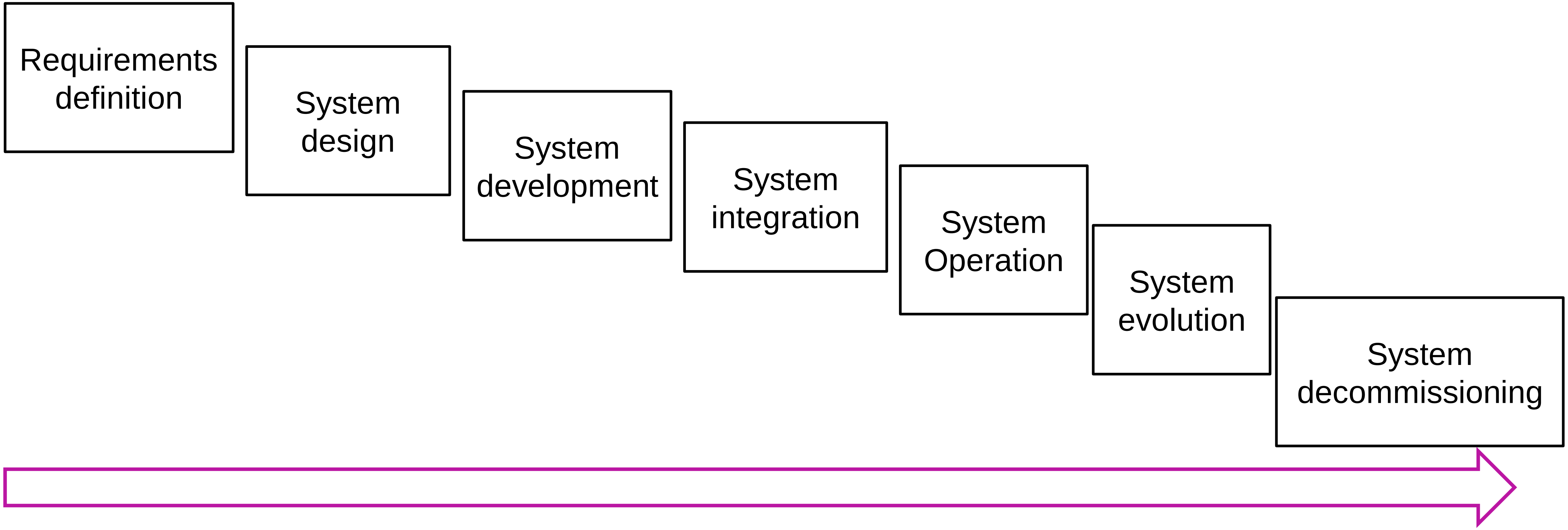
Usually follows a ‘waterfall’ model because of the need for parallel development of different parts of the system

- Little scope for iteration between phases because hardware changes are very expensive. Software may have to compensate for hardware problems

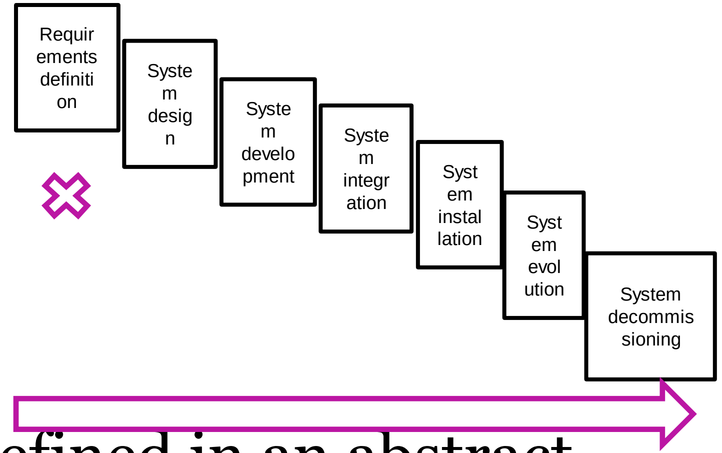
Inevitably involves “engineers” from different disciplines who must work together

- Much scope for misunderstanding here. Different disciplines use different vocabulary and much negotiation is required. Engineers may have personal agendas to fulfil

The system engineering process (2)



System requirements definition

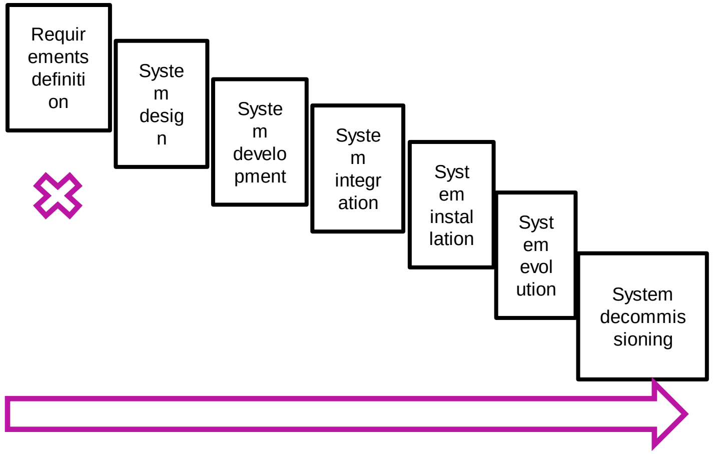


Three types of requirements defined at this stage

- Abstract functional requirements. System functions are defined in an abstract way
- System properties. Non-functional requirements for the system in general are defined
- Undesirable characteristics. Unacceptable system behaviour is specified

Should also define overall organisational objectives for the system

System objectives



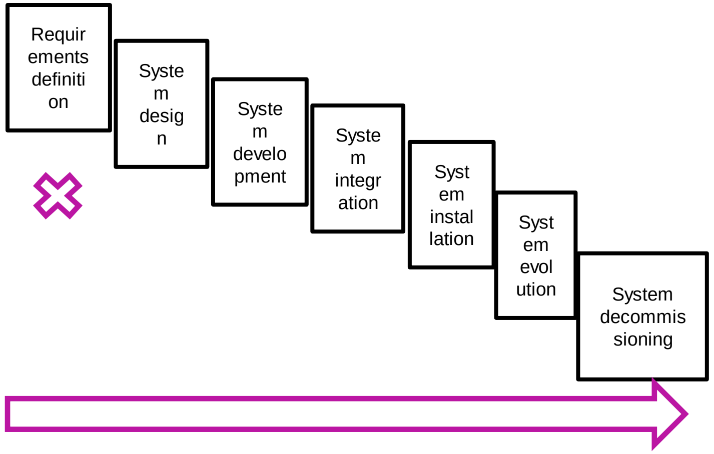
Functional objectives

- Mission of the ship

Organisational objectives

- To ensure that the normal functioning of work carried out on the ship is not seriously disrupted by events such as work processes, management support and wrong regulatory interpretations.

System requirements challenges

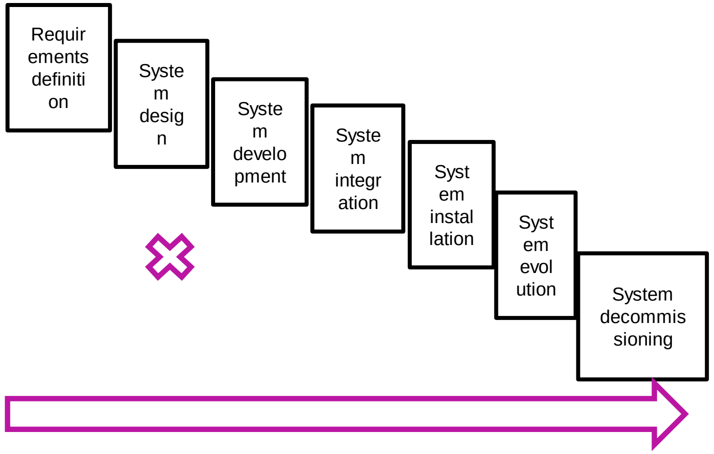


Changing as the system is specified

Must anticipate **hardware/communications developments** over **the lifetime** of the system

Hard to define non-functional requirements (particularly) **without an impression** of the component structure of the system.

The system design process



Partition requirements

- Organise requirements into related groups

Identify sub-systems

- Identify a set of sub-systems which collectively can meet the system requirements

Assign requirements to sub-systems

- Causes difficulties when components need to be integrated

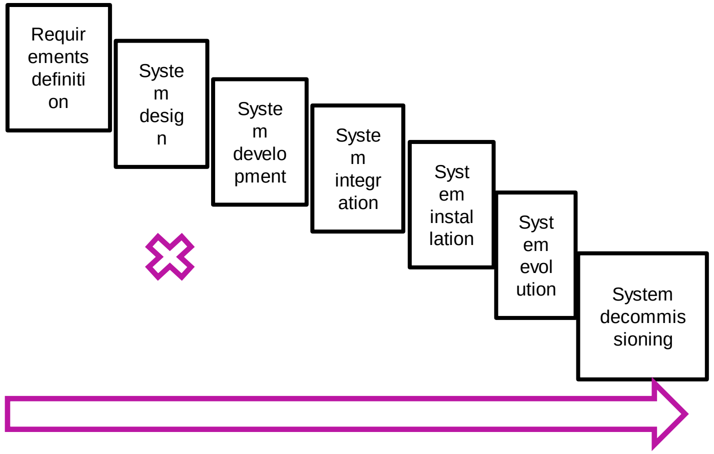
Specify sub-system functionality

- A breakout of functions defined into actions/processes of the systems

Define sub-system interfaces

- Critical activity for parallel sub-system development

System design challenges

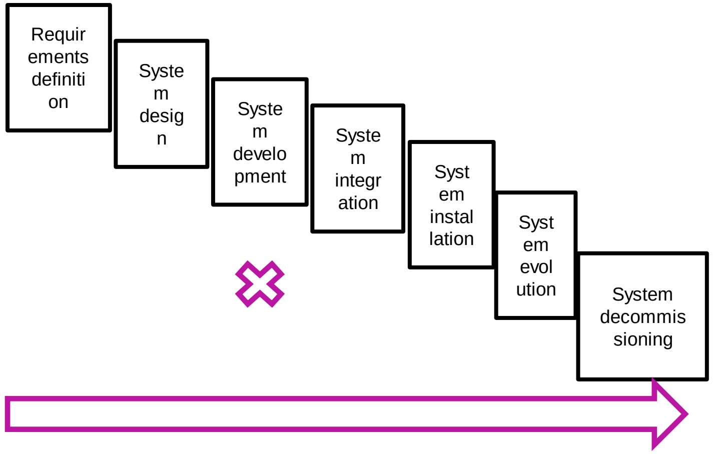


Requirements partitioning to hardware, software and human components may involve **a lot of negotiation**

Difficult design problems **are often assumed** to be readily **solved using software** (*and this will come more and more*)

Hardware platforms may be inappropriate for software requirements so software must compensate for this issue.

Sub-system development



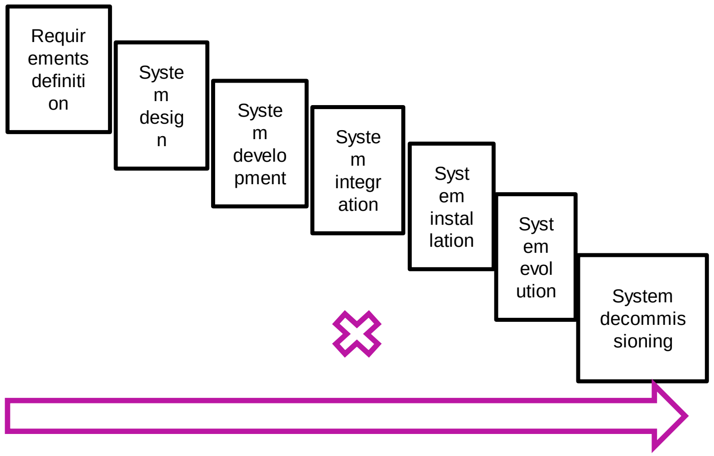
Typically **parallel projects** develop the hardware, software and communications

“May” involve some COTS (Commercial Off-the-Shelf) systems procurement

Lack of communication across implementation teams

Bureaucratic and slow mechanisms for proposing system changes means that the development schedule may be extended because of the **need for rework**

System integration



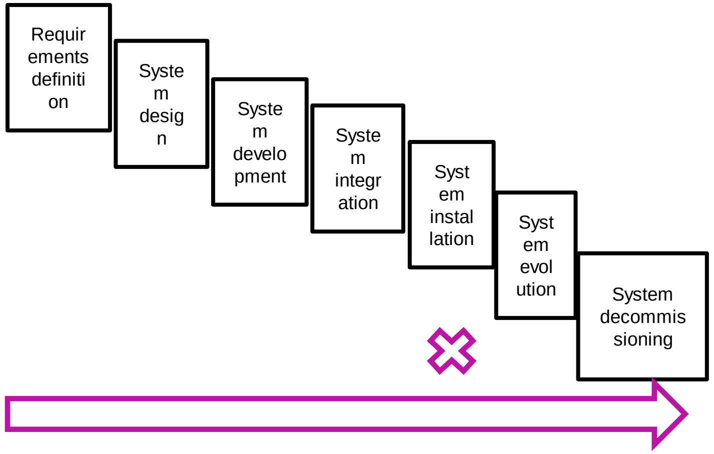
The process of putting hardware, software and people ***together*** *to make a system*

Should be tackled **incrementally** so that sub-systems are integrated one at a time

Interface problems between sub-systems are usually found at this stage

May be problems with **uncoordinated deliveries** of system components

System installation



Reviewing the environmental assumptions for installation

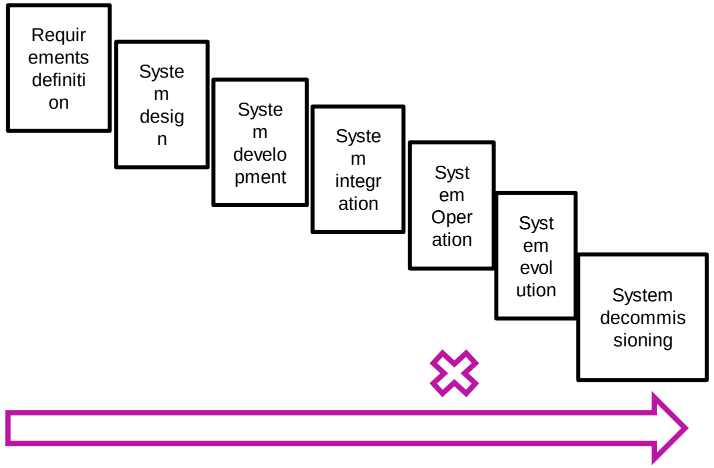
Ensuring the definition of process for the introduction of a new system

Considering the system **may have to coexist** with alternative systems for some time

Considering that it may be **physical installation problems** (e.g. cabling problems)

Operator **training** has to be identified

System operation



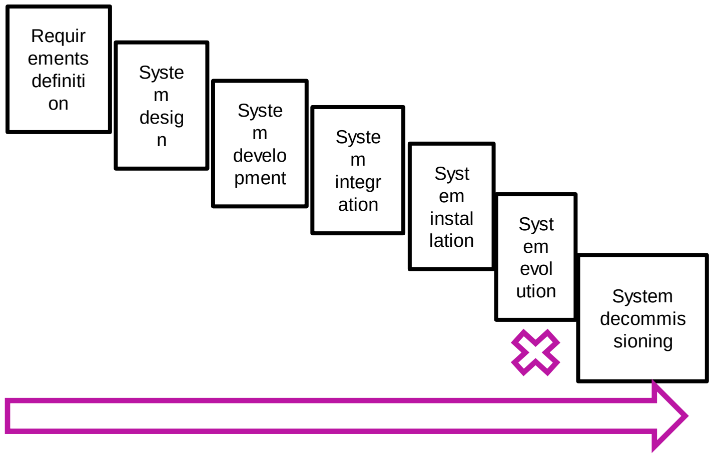
Once the system is in operation, this phase will **bring unforeseen requirements** to light *(part of system's validation)*

This phase reviews the **usability of the system**. Users may use the system in a way which is not anticipated by system designers

May reveal **problems** in the interaction with other systems:

- Physical problems of incompatibility
- Data conversion problems
- Increased operator error rate because of inconsistent interfaces

System evolution



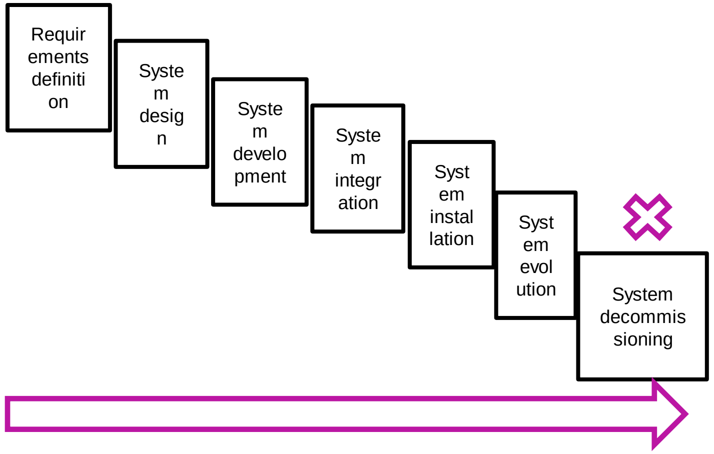
Large systems have a long lifetime. They **must evolve** to meet changing requirements *(part of system's validation)*

Evolution is inherently costly

- Changes must be analysed from a technical and business perspective
- Sub-systems interact so unanticipated problems can arise
- There is rarely a rationale for original design decisions
- System structure is corrupted as changes are made to it

Existing systems which must be maintained are called **legacy systems**

System decommissioning



Taking the system out of service after its useful lifetime

May require the **removal of materials** (e.g. dangerous chemicals) which pollute the environment

- **Should be planned for in the system design** by encapsulation

May require data to be restructured and converted **to be used in some other system**

Discussion (5 min)

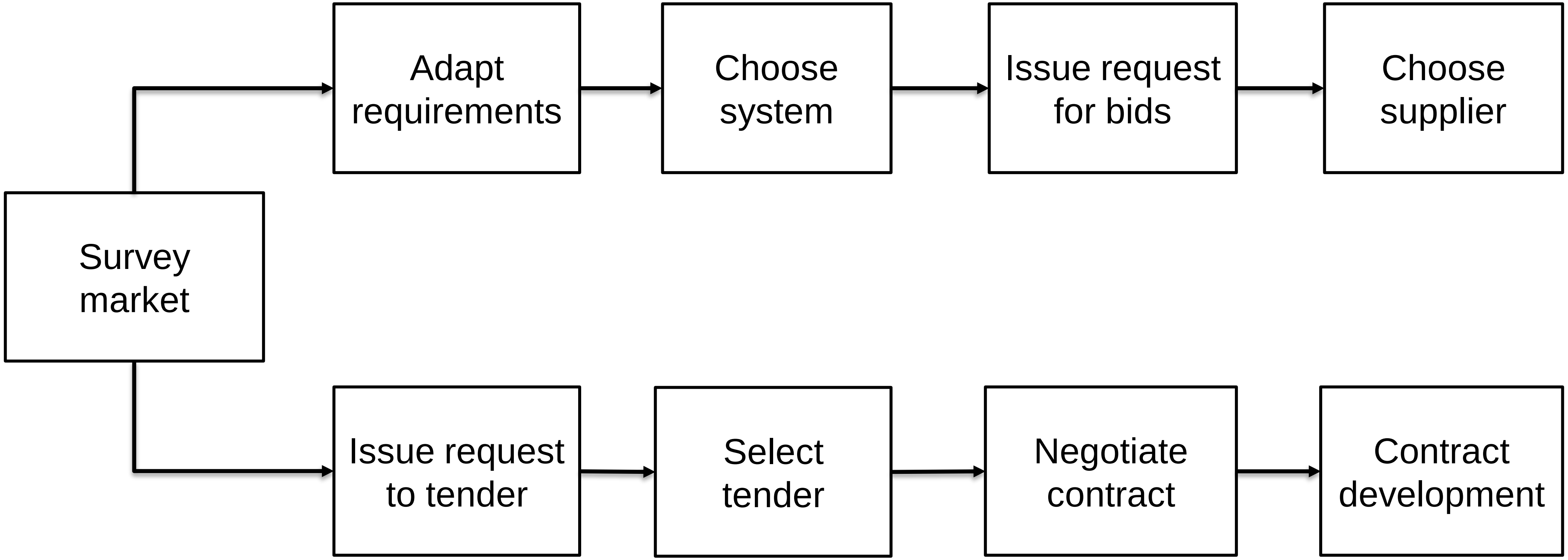
In your concept design,

- **how far you have gone into the ship decommission?**
- **What are the main issues you can anticipate now to this phase?**

Aalto University
School of Engineering

System procurement

The system procurement process (heavily used in ship design)



System procurement

Acquiring a system/product for an organization to meet some need

Some system specifications and architectural designs **are necessary before procurement:**

- You need a specification to **let a contract** for system development
- The specification may allow you **to buy** a commercial off-the-shelf (COTS) system. Almost **always cheaper** than developing a system from scratch**.** ***This relates to the search and selection of systems for a ship concept.***

Contractors and sub-contractors

The procurement of large hardware/software systems **is usually based on some principal contractor**

Sub-contracts are issued to other suppliers to supply parts of the system

The customer liaises with the principal contractor and does not deal directly with sub-contractors.

Aalto University
School of Engineering

Conclusions

Final remarks (1)

System engineering involves input from a **range of disciplines**

Emergent properties are properties that are characteristic of the system as a whole and not its components

System architectural models show major sub-systems and interconnections. They are usually described using block diagrams

Final remarks (2)

System component types are **sensor, actuator, computation, coordination, communication and interface**

The systems engineering process is usually **a waterfall model and includes specification, design, development and integration**.

System procurement is concerned with deciding **which system to buy and who to buy it from**

Conclusions (1)

Systems engineering is demanding! There will never be an easy answer to the problems of complex system development such as the design and selection of ship and marine systems.

With the use of systems engineering, **engineers** won't have all the answers, but **they are better at taking a systems viewpoint.**

Disciplines need to recognize each others' strengths and actively rather than reluctantly cooperate in the systems engineering process. **This is particularly important in the design of advanced/modern ship concepts.**

Conclusions (2)

Is Systems Engineering the Solution to all the World's Systems Problems?

NO

but it does help structure and
manages some of them

Reading materials (links in MyCourses) and Learning logs

Sommerville, I. (2015). *Software engineering 10th Edition,* Chapter 19 and 20.

Kossiakoff, A., Sweet, W. N., Seymour, S. J., & Biemer, S. M. (2011). *Systems engineering principles and practice,* Chapter 1 to 5.

Samad, T., Parisini T. (2011). *Systems of Systems*.

International Council on Systems Engineering (2001). *Systems Engineering,* Web pages.

Learning logs to be returned via MyCourses by 6.11.2022 at 23:59

Aalto University
School of Engineering

Thank you