Introduction to Systems Engineering | Lectures | 2019 | Undergraduate

Presentation · January 2019

DOI: 10.13140/RG.2.2.22737.35683

Ali Zahid
Pasqal

Lecture 1 & 2

ENGR 301 – Systems Engineering
Spring 2019

Introduction

Ali Zahid

Tuesday & Tuesday @ 15th & 22nd January 2019 @ 15 30 – 16 45

Agenda

Lecture 1 Tuesday 15th January 2019

- Introductions
- Aims and Objectives of this course
- Course Readings
- Course Plan
- Grading
- A brief Introduction to Systems Engineering

Lecture 2 – Thursday 17th January 2019

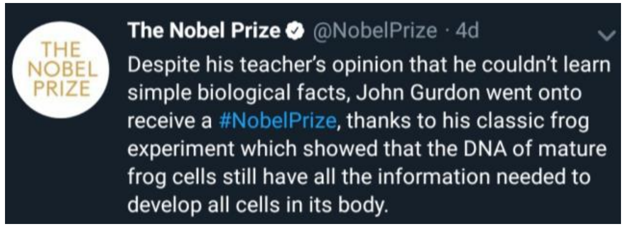
- What is a System and why do we need to study them?
- Systems – A Formal Introduction
- System and it’s environment
- Systems Engineering Project

Nature News & Comment @NatureN... · 18h

CERN has unveiled its bold dream to build a 100-kilometre super-collider.

Next-generation LHC: CERN lays out plans for €21-billion supercollider

nature.com



The Nobel Prize @NobelPrize · 4d

Despite his teacher’s opinion that he couldn’t learn simple biological facts, John Gurdon went onto receive a #NobelPrize, thanks to his classic frog experiment which showed that the DNA of mature frog cells still have all the information needed to develop all cells in its body.

Twitter @humansaboutaliens

Things I will ask with a 100% probability ! 😈

Aims & Objectives

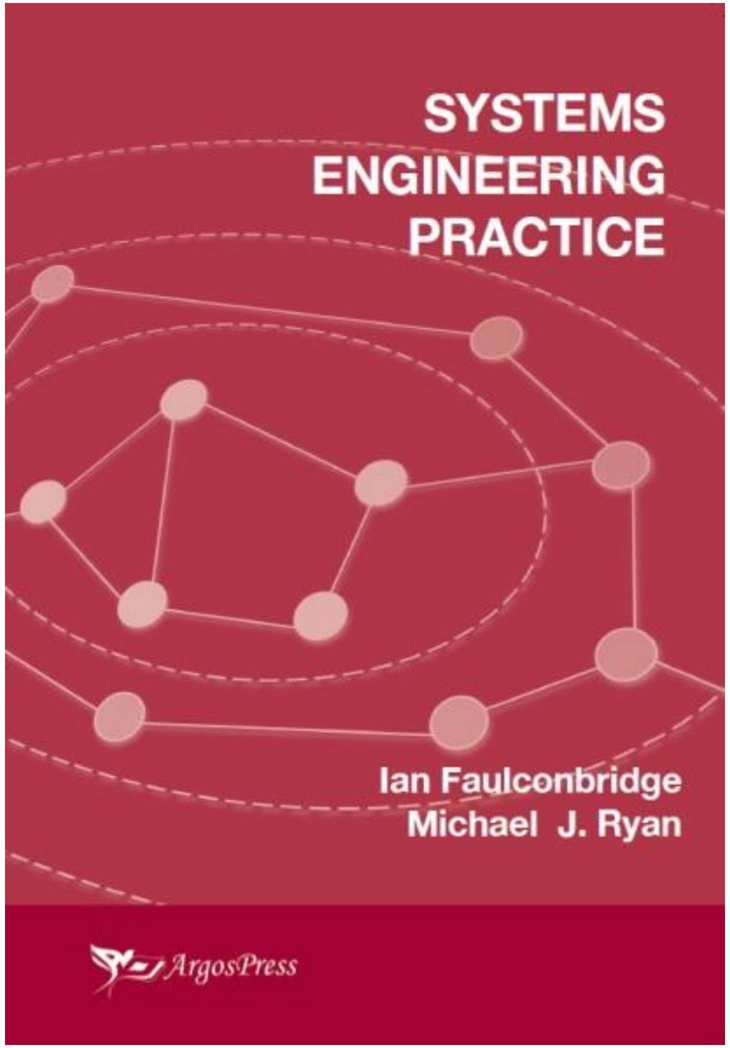
Rationale	Systems Engineering is an interdisciplinary holistic approach to enable the realization of successful systems with a focus on defining customer needs and required functionality early in the development cycle through documenting requirements, then proceeding with design synthesis and system validation while considering the complete problem. Keeping in view the need for an integration of multi-dimensional features in engineering, project management and engineering design, this course provides the students with a hands-on approach to Systems Engineering through practical application of the theoretical concepts learnt in the lectures.
Aims	This course identifies a system and narrows down its broad definitions to focus on human-made or modified systems. Students will be made aware to the broad phases and activities that a system moves through during its life cycle, from early identification of the need for the system, exploration of options, functional design, physical design, detailed design and development, construction and production, utilization and support and then, finally, retirement. Students would be required to form explicit documentation of the requirements of the client, the key elements of the system architecture, and the way those key elements can meet the requirements.
Objectives	• Understand the proper classification of a system, its components, boundaries and complexities. • Recognize the principles and applications of concept development phase, engineering development phase and post development phase. • Comprehend the necessity of System Engineering Management for integration, evaluation, operations and support of a system. • Development of a Systems Engineering Life Cycle Model (SELCM) – Student Project.
Format	Students are expected to attend every lecture so that they can correlate and link the concepts in this course. This is a progressive course with consistent concept building and application. The students are expected to pay attention to the concepts and topics covered in the class and diligently peruse the required readings from their main book (Faulconbridge). For giving the students a hands-on experience in design, documenting, and implementing a system, there would be a practical Project Class every alternate week on Thursday (@ 3:30 PM – 4:45 PM) in the Playground. Students are expected to bring their own laptops with Microsoft Project and Microsoft Visio installed. NO sharing would be allowed. The students are expected to follow the given instructions and do the project, on their laptops, alongside the instructor during the Project Classes. The End Term Examination will include an individual Project + a written exam, to implement the concepts learnt during the course.

System Engineering is a Perspective – a way of thinking
It is not derivative from any of the sciences – yet it has implications and applicability in every science

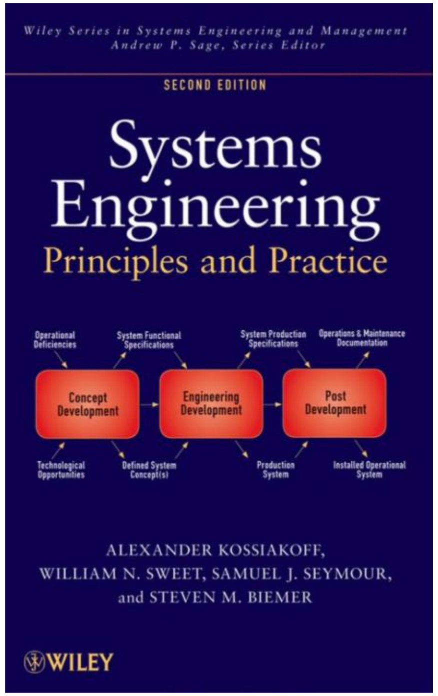
The 1 major requirement of Systems Engineering is Documentation
Document everything!!

Readings

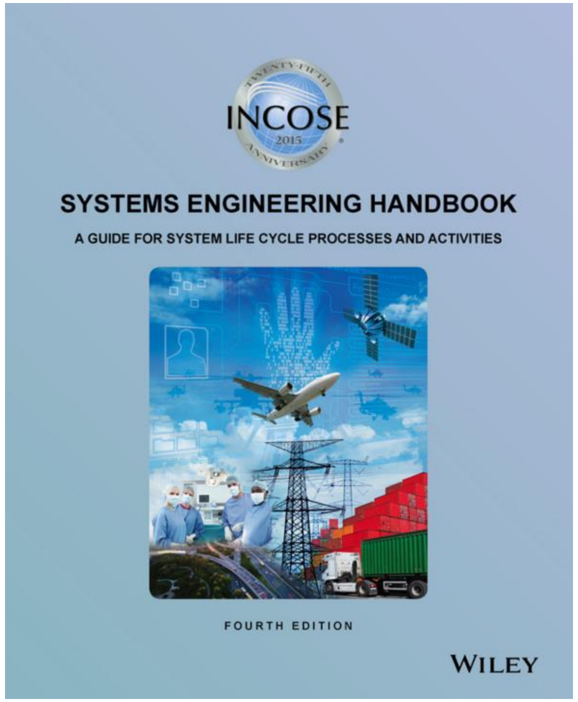
We will be exclusively covering topics of Faulconbridge
Several topics from Kossiakoff will be covered
The slides would have a reference in the bottom left regarding which book to use
PMBOK Guide will be used for our Project classes



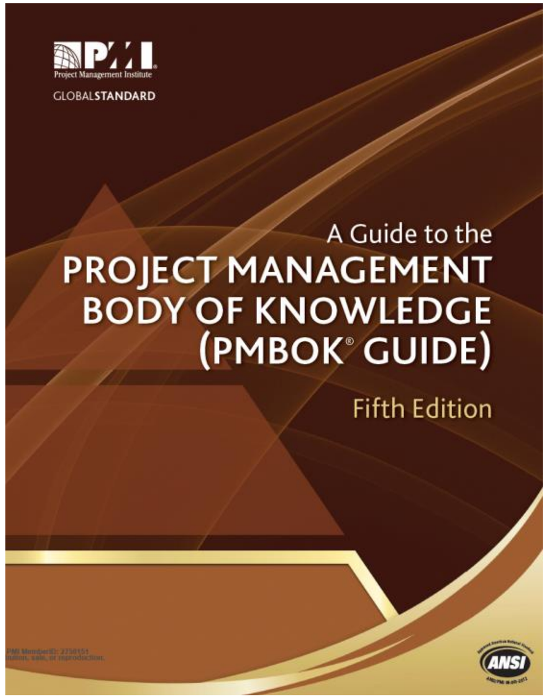
Ian Faulconbridge & Michael J.Ryan
Systems Engineering Practice
Argo Press
Main book



Kossiakoff, Sweet & Seymour
Systems Engineering
Principles & Practice
Wiley International Inc.



INCOSE
Systems Engineering Handbook
4th Edition



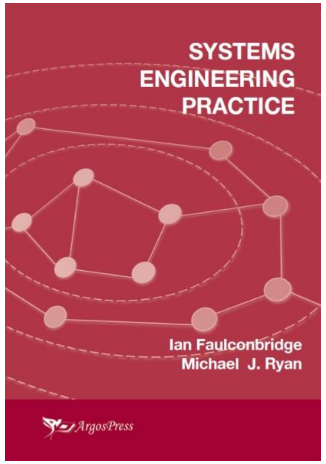
Project Management Institute
Project Management Handbook
PMBOK Guide
5th Edition

Course Weekly Plan

Topics	Week	Chapters	Assessments
Introduction	1	1	-
Requirement Engineering Framework	2	2	Assignment 1
Play-Ground	2	-	Project Work
Conceptual Design	3	3	Quiz 1
Play – Ground	3		
Conceptual Design	4	3	-
Preminilary Design	5	4	Assignment 2
Preminilary Design	6	4	Quiz 2
Detailed Design and Development	7	5	-
Detailed Design and Development	8	5	Assignment 3 Mid Term
Play – Ground	8	-	
Construction & Production	9	6	-
Operational Use & System Support	10	7	Quiz 3
Play – Ground	10	-	Assignment 4
Systems Engineering Management	11	8	Quiz 4
Play – Ground	11	-	-
Systems Engineering Management	12	8	Assignment 5
Play – Ground	12	-	
Systems Engineering Standards	13	9	
Play – Ground	13	-	
Systems Engineering Standards	14	9	Quiz 5
Play – Ground	14	-	
Play – Ground	15	-	

These topics can be amended, and additional topics can be added depending on what I deem is necessary for you to understand the course.

Students are expected to bring their own laptops + the installed programs.
A manual will be provided to the students before the start of the class to follow.
The instructor will provide a demonstration to the workings of the programs and what needs to be done



All the Chapter Readings are from this book.
It will be uploaded on LMS or you can ask me to email it to you.

Grading

	Grade	GPA Point	Percentage Range
Grading Scale	A+	4.00	[95 – 100]
	A	4.00	[90 – 95)
	A-	3.67	[85 – 90)
	B+	3.33	[80 – 85)
	B	3.00	[75 – 80)
	B-	2.67	[70 – 75)
	C+	2.33	[67 – 70)
	C	2.00	[63 – 67)
	C-	1.67	[60 – 63)
	F	0.00	[0 – 60)

	CLO	Description	Domain Level
Specific Course Learning Outcomes [CLOs]	CLO 1	Identify a system, its components, boundaries, complexities, hierarchies and get acquainted with the System Life Cycle.	COG 1
	CLO 2	Comprehend the need for design development, engineering development and post development in an integrated system engineering approach.	COG 2
	CLO 3	Demonstrate the systems engineering and systems management approach through investigating various relevant aspects of system conceptual & engineering design, requirement engineering, system engineering standards and validation.	COG 3
	CLO 4	Determine the various components of systems engineering in different disciplines and validate their manifestation in relevant case studies.	COG 4
	CLO 5	Design a System Engineering and Management Life Cycle.	COG 5

	Items	Weights (%)	How many?
Marks Breakdown	Assignments	21	7
	Quiz / Tests	20	5
	Mid Term	24	1
	Project	15	1
	End Term	20	1

Plagiarism
Instructor's Policy

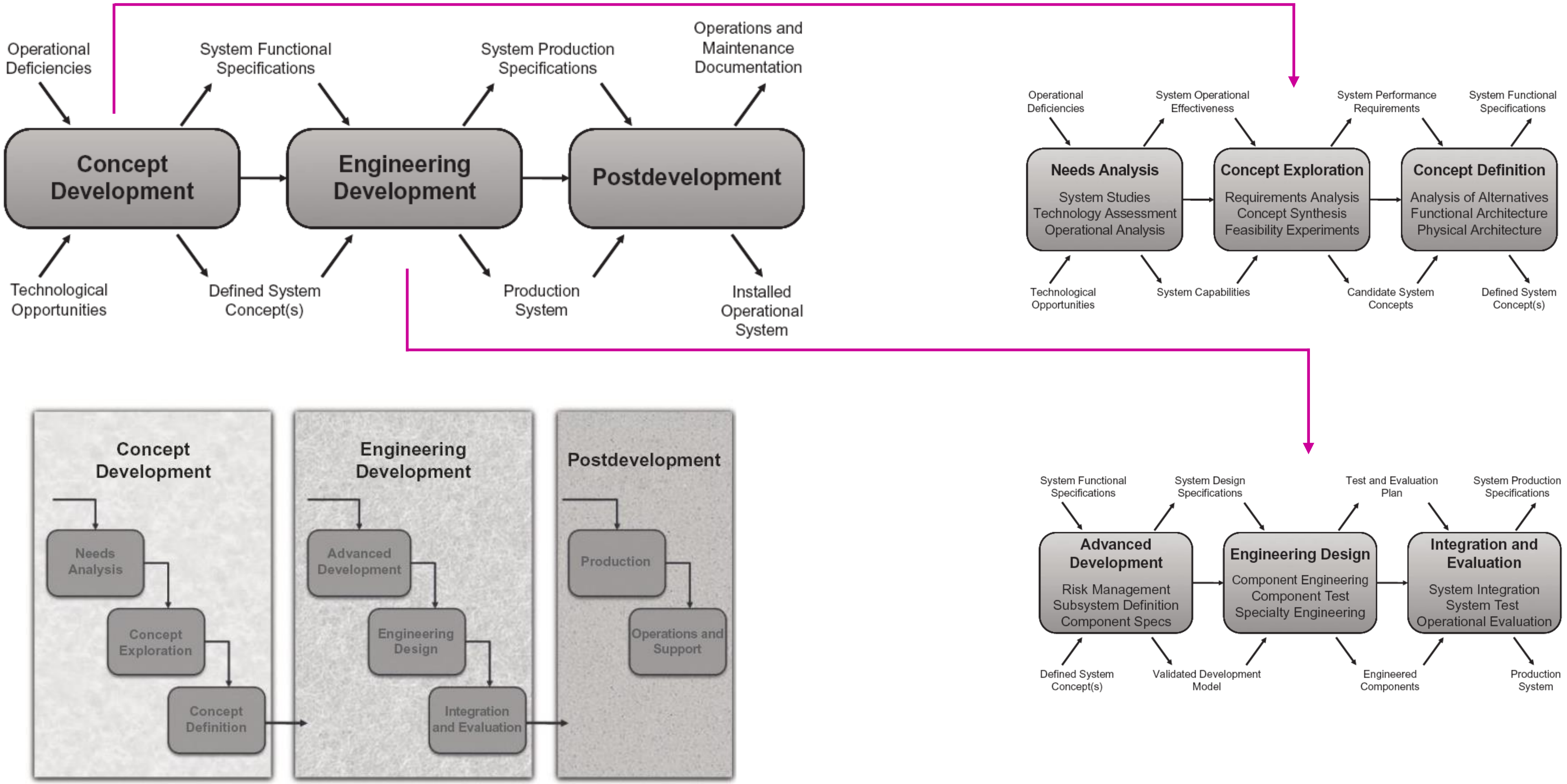
I am strictly against plagiarism. If you answer questions in your assignments and DO NOT give the reference from where you have acquired the written content, I will give a zero in the whole assignment section.

All 7 assignments would be marked zero. If you take something from Wikipedia, for example, there should be a reference that it is take from a particular article of Wikipedia. It is similar for all content – text and figures. This is the only warning that will be given, and I have given it here in writing.

- Assignment 6 and Assignment 7 will be assessed on class participation (how vocal you are during the lectures and do you ask questions or just sit with a blank face + do you let me know if you aren't understanding something) + how many times you get the answers right when asked to enumerate a concept, on the white board, in the lectures.

** All Quizzes and Assignments carry equal marks

What exactly will we study? – Basic Blocks



Operational Deficiencies
System Functional Specifications
System Production Specifications
Operations and Maintenance Documentation

Concept Development
Engineering Development
Postdevelopment

Technological Opportunities
Defined System Concept(s)
Production System
Installed Operational System

Operational Deficiencies
System Operational Effectiveness
System Performance Requirements
System Functional Specifications

Needs Analysis
System Studies
Technology Assessment
Operational Analysis

Concept Exploration
Requirements Analysis
Concept Synthesis
Feasibility Experiments

Concept Definition
Analysis of Alternatives
Functional Architecture
Physical Architecture

Technological Opportunities
System Capabilities
Candidate System Concepts
Defined System Concept(s)

Concept Development
Needs Analysis
Concept Exploration
Concept Definition

Engineering Development
Advanced Development
Engineering Design
Integration and Evaluation

Postdevelopment
Production
Operations and Support

System Functional Specifications
System Design Specifications
Test and Evaluation Plan
System Production Specifications

Advanced Development
Risk Management
Subsystem Definition
Component Specs

Engineering Design
Component Engineering
Component Test
Specialty Engineering

Integration and Evaluation
System Integration
System Test
Operational Evaluation

Defined System Concept(s)
Validated Development Model
Engineered Components
Production System

Reference: Kossiakoff, Sweet, Seymour

System – A Formal Introduction

First, we need to define a System – to get acquainted with our point of interest

A very simplistic & layman-ish definition of a System [courtesy of the Greeks]:

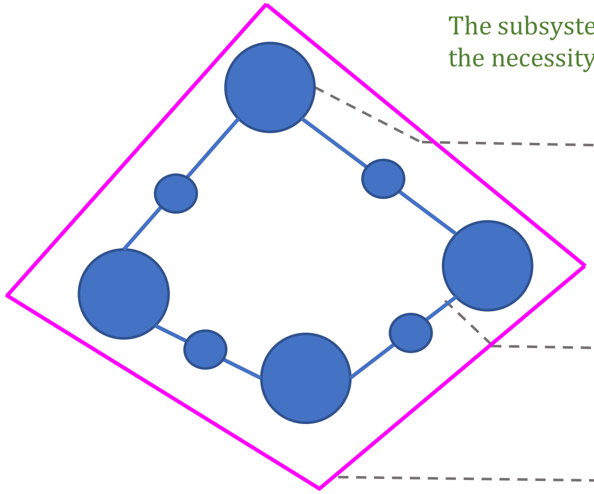
'A whole resulting from grouping together a number of things in a particular manner for a particular reason'

Now let's consider formal definitions that are more 'professional'

'A combination of interacting elements organized to achieve one or more stated purposes'

'A system is an integrated set of elements, subsystems, or assemblies that accomplish a defined objective'

The subsystem can & ARE rearranged in a hierarchy implying the necessity of one subsystem before another



System Elements

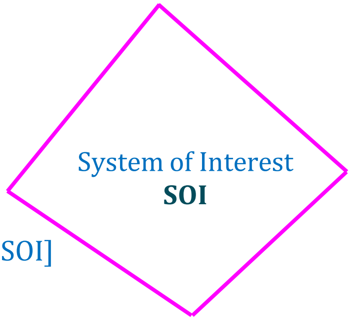
Interactions
Interconnections

Boundary

A system will have 3 major components:

- Multiple System Elements or Sub-Systems
- Interactions between System Elements
- A boundary marking the ends of the System

Anything inside the boundary is known as the System of Interest [SOI]



System of Interest
SOI

Defined – we will consider it as an axiom – a statement that we will consider to be true for future arguments

'…Learn the rules … then play better …'

This is the essence of a System – understand this simple sentence.
We will build every concept around it.

- Collection of some tasks
- Internal elements - System Elements – Sub Systems
- Internal elements have interconnections between them
- There is a defined objective
- Some purpose must be achieved
- Once identified – a system will have a system boundary

The purpose of the system is known as it's <u>mission</u>

- Must be clearly stated by business management + stake holders
- Once the purpose is defined it marks the start of the design process
- It is the test for systems ' fitness-for-purpose'

The mission of the System must ALWAYS provide a solution to a business problem.

There must be a business NEED that drives the necessity | purpose of a system.

Profit should always be made.

Segregating the System into its 3 elements:

i. Subsystems
ii. Interactions
iii. Boundary

is not accidental but is a consequence of the designing phase of the System Engineering Process.

Deliberately Designed

IMPORTANT

The concept + design phase is the most important phase of all!!!

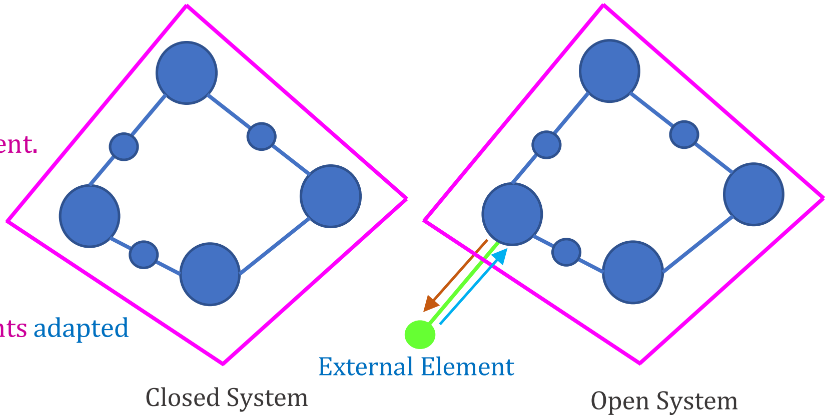
Reference: Kossiakoff, Sweet, Seymour & Falconbridge

System – Classifications

There are 4 generic classifications of a System:

1. Closed | Open System

A Closed System does NOT interact with the outside environment – it is isolated – it does NOT accept inputs from the outside environment.
An Open System is interactive with its environment – it accepts inputs across the boundary and sends outputs across its boundary ✓



2. Natural | Human Made – Human Modified System

Natural Systems contain natural elements and are a result of natural processes [plants, humans, animals, thunderstorms etc..]
Human Made | Human Modified Systems are generated through the efforts of humans + contain human-made elements | natural elements adapted by humans. ✓

3. Physical | Conceptual System

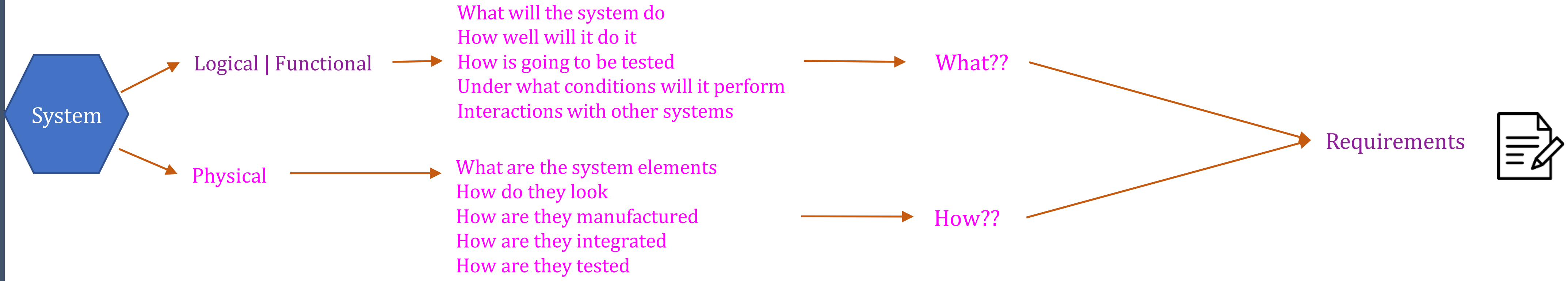
Physical systems have a physical form [an airplane, a computer hardware, cellphones, submarine] ✓
Conceptual systems do NOT have a physical form [belief, string theory, morality]

4. Precedented | Un - Precedented System

A precedented system has been produced before ✓
An un-precedented system has NOT been produced before + requires substantial research and development

Our focus is on Systems that are:
- Open
- Human Made | Human Modified ✓
- Physical
- Precedented

Closed + Natural + Conceptual + Un-Precedented Systems:
- Can be modelled
- Requires time + effort
- Is NOT a Business NEED



System

Logical | Functional

What will the system do
How well will it do it
How is going to be tested
Under what conditions will it perform
Interactions with other systems

What??

Physical

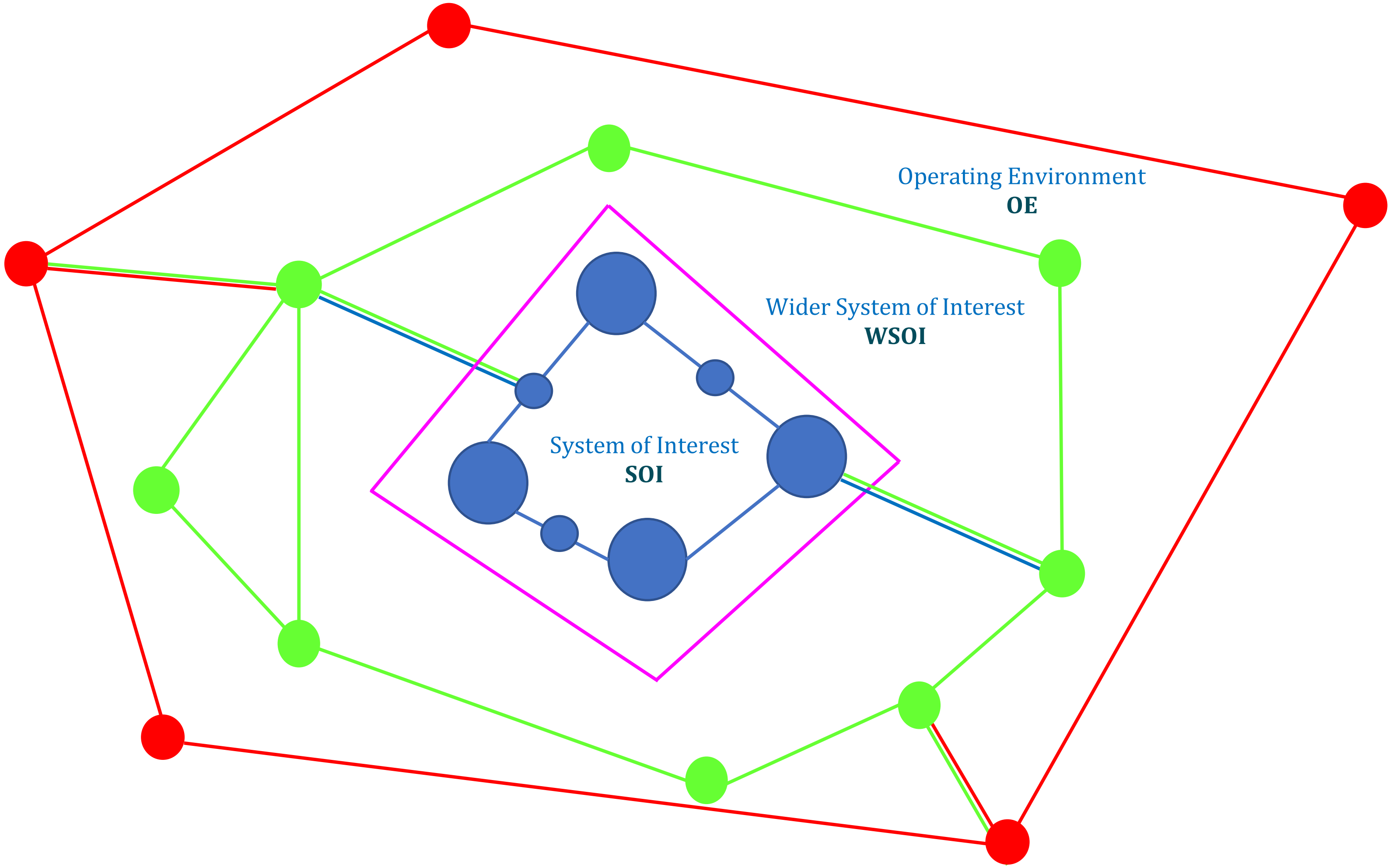
What are the system elements
How do they look
How are they manufactured
How are they integrated
How are they tested

How??

Requirements

Reference: Kossiakoff, Sweet & Seymour - Systems Engineering Principles & Practice - Wiley International Inc

System and it's Environment



Reference: Kossiakoff, Sweet & Seymour - Systems Engineering Principles & Practice - Wiley International Inc

Lecture 3 + Lecture 4

ENGR 301 – Introduction to Systems Engineering

Spring 2019

System Life Cycle

Ali Zahid

Tuesday 29th and 31st January 2019 @ 15 30 – 16 45

Agenda

Lecture 3 Tuesday 29th January 2019

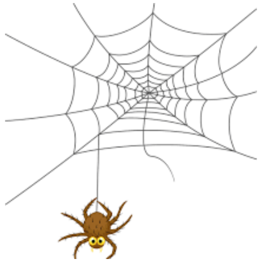
- System as a Product
- Hierarchical Description of a System

Lecture 4 Tuesday 31st January 2019

- System – OF – Systems [SOS]
- System Life Cycle
- Quiz

Important Notions from the previous Lectures

Formal + Informal Description of a System
System Boundary
System and It's Environment
Application to a Particle Accelerator [Project]



Twitter @humansaboutaliens

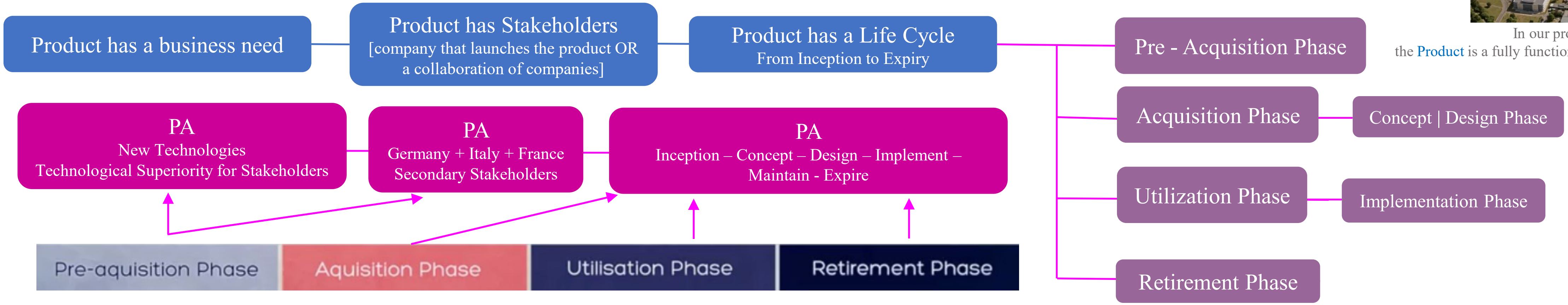
Things I will ask with a 100% probability !

System – As a Product

A System can also be considered as a Product – since the deliverable of a System is similar to what is expected of a Product from a business perspective

In very simplistic terms – the Product perspective applied to a System is very easy to comprehend

In our project
the Product is a fully functioning Particle Accelerator



Product has a business need

Product has Stakeholders
[company that launches the product OR a collaboration of companies]

Product has a Life Cycle
From Inception to Expiry

PA
New Technologies
Technological Superiority for Stakeholders

PA
Germany + Italy + France
Secondary Stakeholders

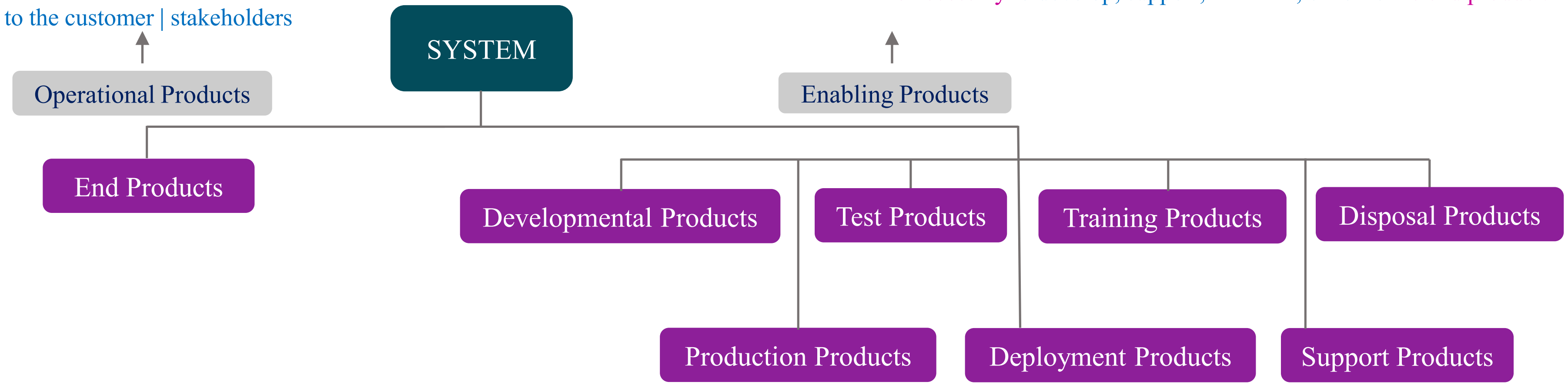
PA
Inception – Concept – Design – Implement – Maintain - Expire

Pre - Acquisition Phase

Acquisition Phase — Concept | Design Phase

Utilization Phase — Implementation Phase

Retirement Phase



SYSTEM

A product which is DELIVERED to the customer | stakeholders

Operational Products

- End Products

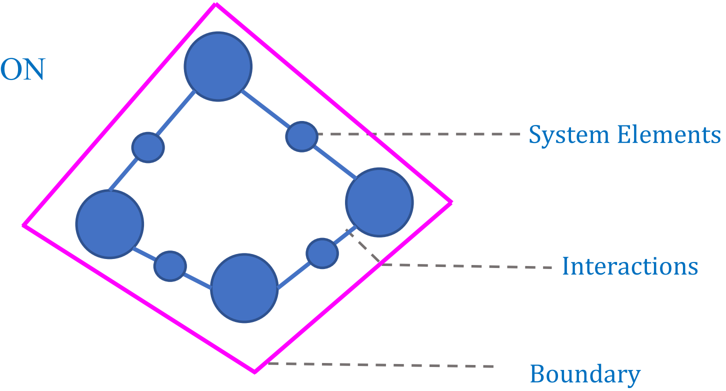
A product which is NOT DELIVERED to the customer | stakeholders, but which is necessary to develop, support, maintain, or retire the end product.

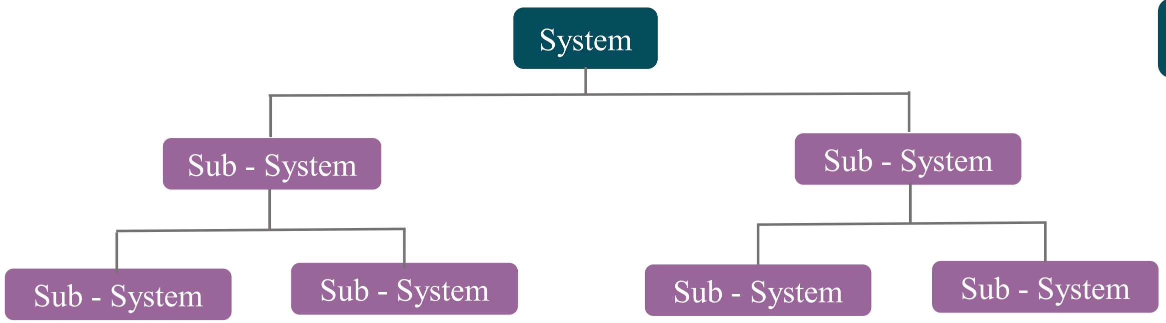
Enabling Products

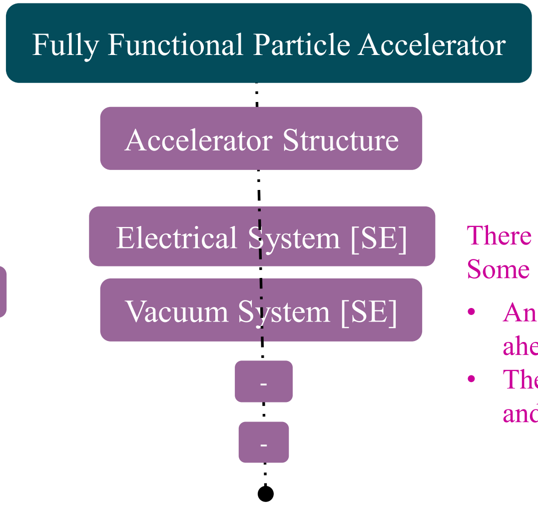
- Developmental Products
- Test Products
- Training Products
- Disposal Products
- Production Products
- Deployment Products
- Support Products

System Hierarchy

Sub-systems are fundamental elements of a System
Every individual sub-system, INSIDE the System, contributes to the System's Mission by performing a particular dedicated FUNCTION
Some sub-systems are much more important than the others OR they have precedence over the other sub-system
Some functions must be carried out first before another function is able to contribute to the System







A System - on the basis of its Sub-Systems - has a hierarchical structure
'Consider a System to be a hierarchical composition of System Elements'
Some tasks | sub-systems always precedes others

There is a preferential hierarchy based on functionality
Some sub-systems must be implemented first

- An accelerator structure should be present first before you can go ahead and implement the electrical system
- The electrical system should be present first before you can go ahead and implement the vacuum system

Whatever the System Hierarchy is, it will follow as generic Systems' Life Cycle →

Pre-aquisition Phase	Aquisition Phase	Utilisation Phase	Retirement Phase

These phases in the Systems' Life Cycle can be modified to include other specific tasks [like in a Particle Accelerator] BUT they would have the same generic purpose

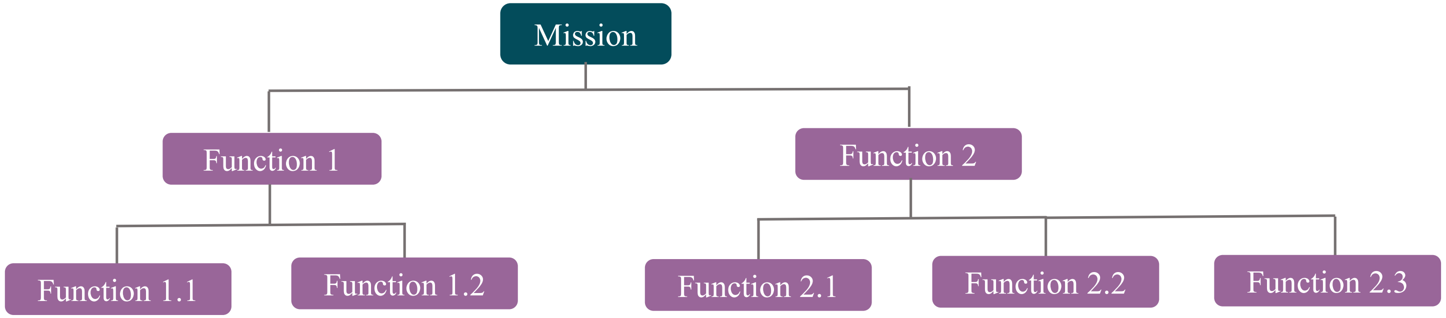
Every System has its own Systems Engineering Framework – Our Particle Accelerator uses a specific SE Framework called openSE developed @ CERN [details in coming lectures]

System Hierarchy

- Logical | Functional → System's Mission broken down into a hierarchical structure of its major functions
  Functional Architecture → Cooling function – Beam Focusing function – Acceleration function
- Physical → System's Mission broken down into a hierarchical structure in terms of physical components
  Physical Architecture → Refrigeration – Lens – Electromagnetic Track

System Hierarchy – Schematic Representation

Logical | Functional Hierarchy



Mission

Function 1

Function 2

Function 1.1

Function 1.2

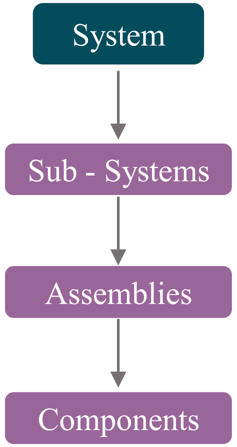
Function 2.1

Function 2.2

Function 2.3

System breakdown in terms of its different functions [sub-systems]

Physical Hierarchy



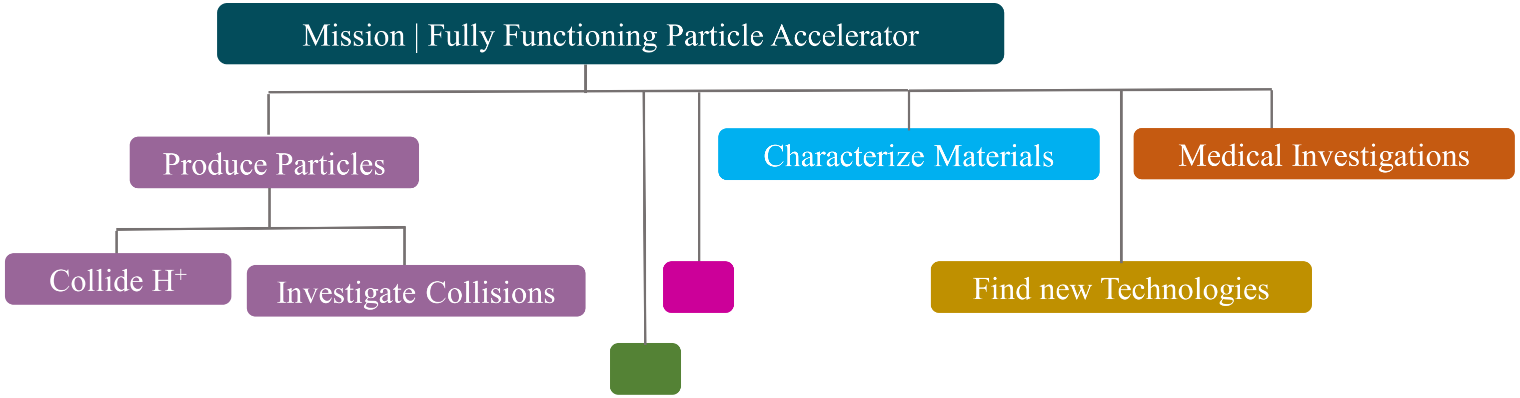
System

Sub - Systems

Assemblies

Components

System breakdown in terms of its physical constituents



Mission | Fully Functioning Particle Accelerator

Produce Particles

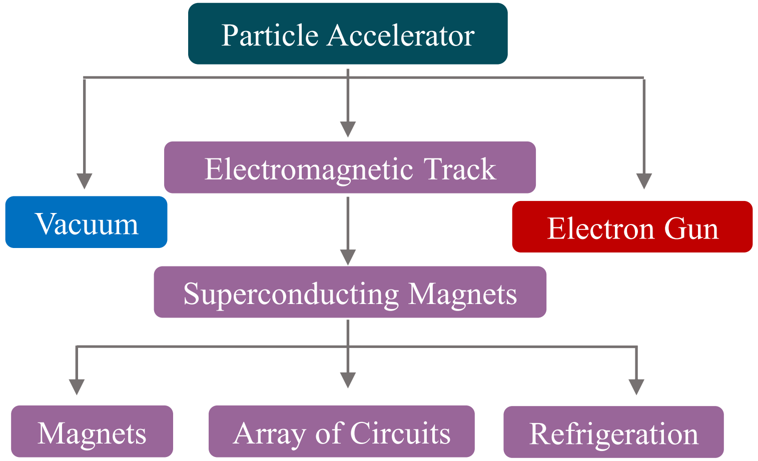
Characterize Materials

Medical Investigations

Collide H^+

Investigate Collisions

Find new Technologies



Particle Accelerator

Electromagnetic Track

Vacuum

Electron Gun

Superconducting Magnets

Magnets

Array of Circuits

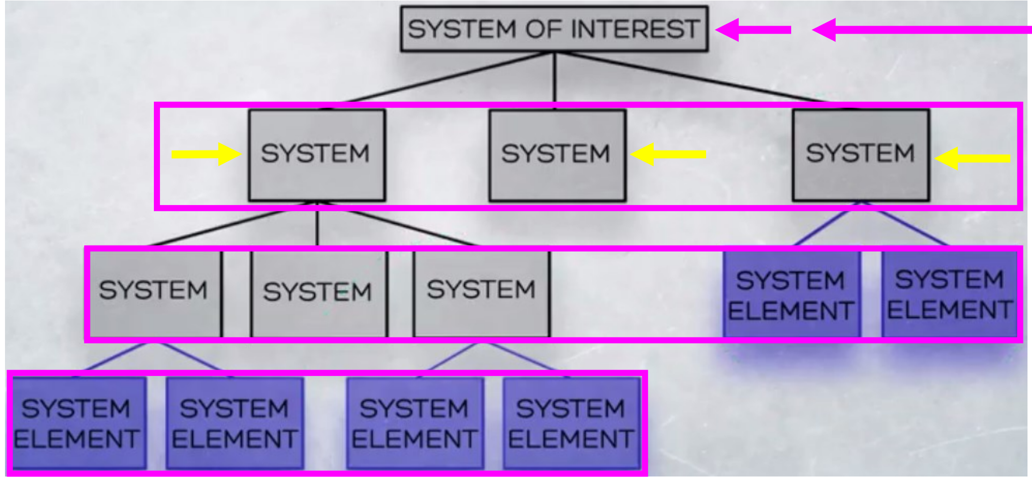
Refrigeration

System of Systems - SOS

Sub-elements | sub-systems which are an integral part of a Systems [SOI] might be individual Systems in their own rights – they have their own distinctive characteristic of a System

Which means a Sub-system inside a System might be a System with its own rights:

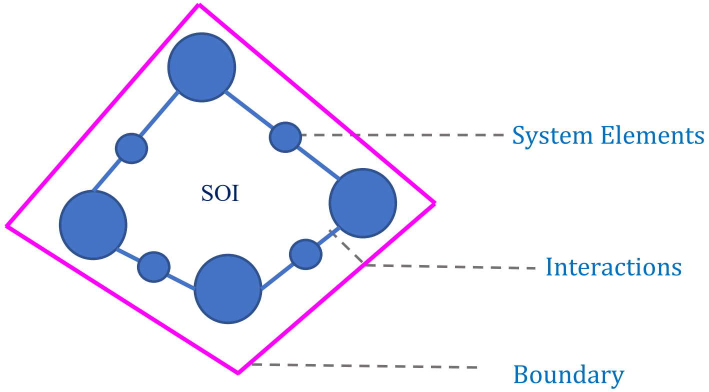
- System Elements | Sub-Systems
- Interconnection between the Sub-Systems inside that System
- A boundary which defines the System



System Of Systems [SoS]

You will find Systems inside Systems inside System [SOI]

Systems that have their own 'System-like characteristics'

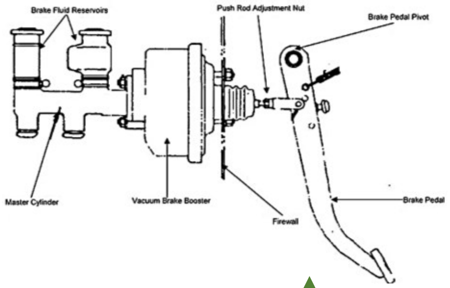


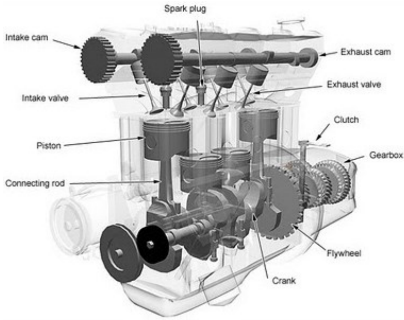
SoS elements are Systems in their own rights:

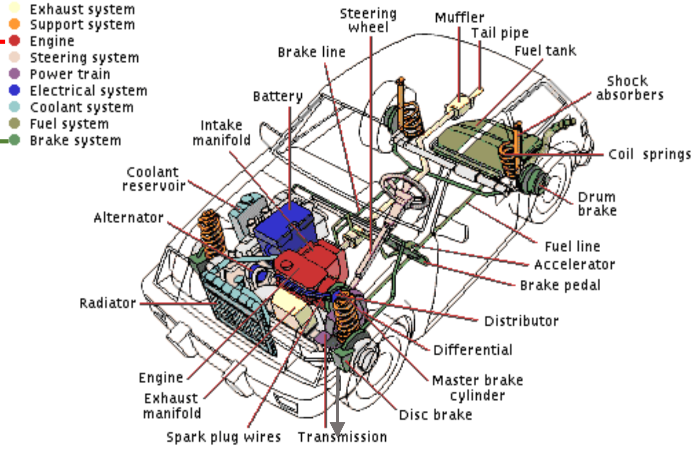
- Managerially Independent
- Operationally Independent

SoS elements, since they are Systems, are optimized for their OWN PURPOSE before being part of the SoS

System Elements on the other hand – are optimized for the purpose of the System of Interest [SOI]







The Car System with its Sub-Systems
Most of the Sub-Systems are Systems on their own

System – As a Product

- Managerial Independence
- Operational Independence

Optimized for the System's Purpose
System IS optimized

Optimized for their OWN Purpose
SoS is most likely NOT optimized

😈

Lecture 5 + Lecture 6

ENGR 301 – Introduction to Systems Engineering
Spring 2019

System Life Cycle

Ali Zahid

Tuesday 19th and 21st February 2019 @ 15 30 – 16 45

Agenda

Lecture 5 Tuesday 19th February 2019

- System Life Cycle + Responsibilities of the Parties Involved in the System
- Activities in the System Life Cycle

Lecture 6 Thursday 21st February 2019

- Utilization and Retirement Phase
- What is Systems Engineering?
- Top – Down Approach

Important Notions from the previous Lectures

Formal + Informal Description of a System
System Boundary
System and It’s Environment
Systems of System [SoS]
System Level and Sub-System Level Architecture

Things I will ask with a 100% probability ! 😈

System Life Cycle – 4 phases

A System has a Life
It is brought into existence (1) for a specific purpose, used (2) and then disposed (3) when it no longer serves its purpose.

During its LIFE, the system goes through various phases + activities which are indicative of the MATURITY of the System.

System Life Cycle

Pre – acquisition Phase | Acquisition Phase | Utilization Phase | Retirement Phase

Knowledge about every phase is imperative for a Systems Engineer
The phases are standardized and established so that Systems can be dissected + compared + diagnosed
EVERY System will go through these phases
Additional Phases can be added depending on the complexity of the System

1 Pre – acquisition Phase

The System starts in this Phase
Idea generated due to a Business Need + supported by a Business Case
Ensures that only feasible + cost-effective + predictably successful ideas go to the acquisition phase
Organization spends funds on research and development for feasibility

2 Acquisition Phase

Focused on bringing the System into existence
The System is DEFINED in terms of:
- Business Needs + Requirements
- Stakeholder Needs + Requirements
- System Requirements

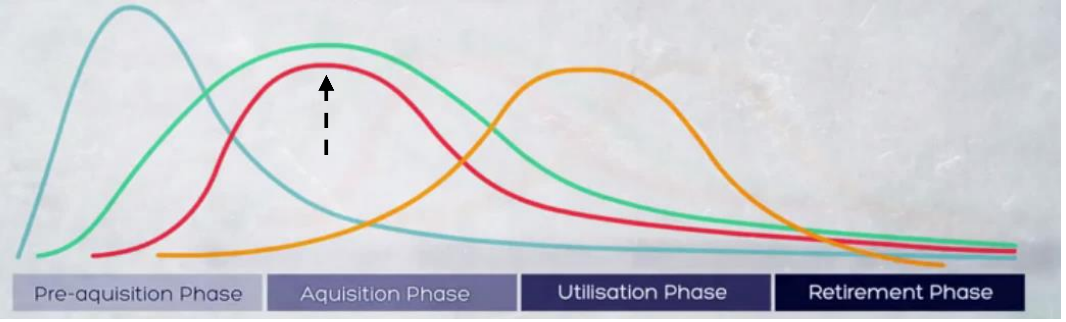
Engaging a CONTRACTOR to develop | deliver the System

3 Utilization Phase

System is Operated and Supported during this Phase
Upgrades + modifications are possible (NO major redesign) to:
- Rectify performance shortfalls
- Meet varying operational requirements
- Enhance performance reliability

4 Retirement Phase

The System has no longer any use + a better System is conceived
System can no longer meet the required functions
No longer cost effective to keep the System operational
Since a framework for the retired System exists – another System can be designed to take its place

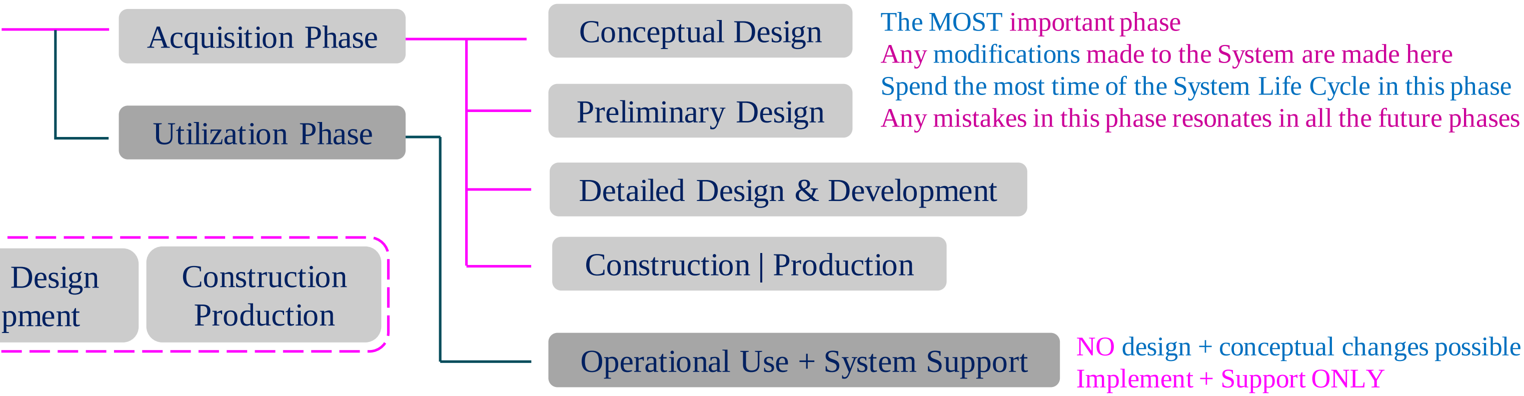


Systems Engineering —— Project Management ——
Operations Management —— Business Management ——

Responsibilities of the Parties Involved

Activities - Phases of Systems Life Cycle

Systems Engineering is predominately effective + relevant + related to



The MOST important phase
Any modifications made to the System are made here
Spend the most time of the System Life Cycle in this phase
Any mistakes in this phase resonates in all the future phases

NO design + conceptual changes possible
Implement + Support ONLY

A Systems Engineer job is largely focused on the:

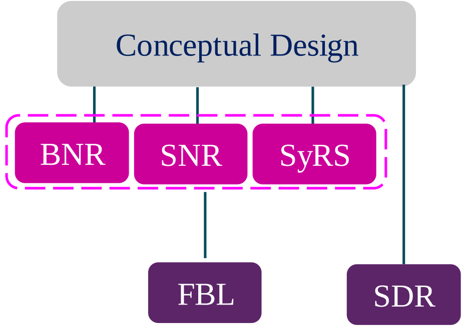
- Acquisition Phase
- Utilization Phase

Conceptual Design

Clearly well-defined Requirements – System Requirements are defined
Provides a Formal Process through which the Business Needs and Requirements [BNR] are articulated + confirmed by the Stakeholders
Mission Statement translated | refined to complete logical description of the System of Interest [SOI]

BNR further elaborated at the business operation level into Stakeholder Needs & Requirements [SNR] [finances + timelines + expectations]
SNR further elaborated by requirement engineers – System Requirement Specifications [SyRS]

BNR + SNR + SyRS = Functional Baseline [FBL] – describes the WHAT and WHY of the System – System Level Logical Architecture
Conceptual Design Phase ends with the System Design Review [SDR]
SDR - provides a formal record of design decisions + design acceptance + communicates design to Stakeholders + confirms FBL



A structural design that gives as much detail as possible WITHOUT constraining the architecture to a particular technology or environment | conceptual | abstract |

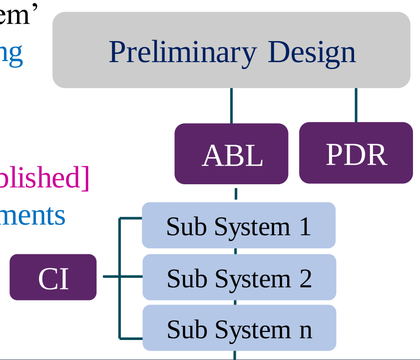
↑

Preminilary Design

Converts the logical architecture [grouping] in the initial FBL to description of physical sub-systems [physical architecture] – 'HOWS of the System'
Logical design shifts to physical design – draw the engineering drawing on a particular software with all technical specifications – serious designing
Transition from the problem domain into the solution domain

At the end of the Preminilary Design process a subsystem – level design is obtained – Allocated Baseline [ABL]
ABL – logical architecture defined in the FBL is further refined, investigated and detailed [Physical + Functional Sub-system Hierarchies are established]
ABL – the functionality of the System is allocated to physical building blocks called Configuration Items [CI] – contains sub system level requirements

Ends with a Preliminary Design Review [PDR] – assess the technical risk and satisfaction of the FBL

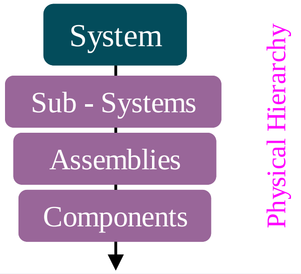


Activities - Phases of Systems Life Cycle

Detailed Design Development

ABL developed in the Preliminary Design is used to complete development of Sub Systems + Assemblies + Components in the System
Prototyping – Proof of Idea
System Design Validation through Testing + Evaluation
The System is now well defined by numerous products making up the System – sub systems + assemblies + components
The System definition [including functional + physical hierarchies] should be necessarily detailed – commencement of Production activities

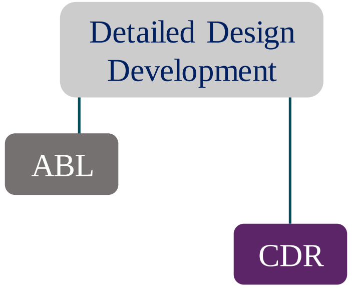
Establishment of the initial Product Base Line [PBL] @ Critical Design Review [CDR]
CDR – final design review
Official acceptance of the design + ensures design compatibility + System assessment for commencement of Production

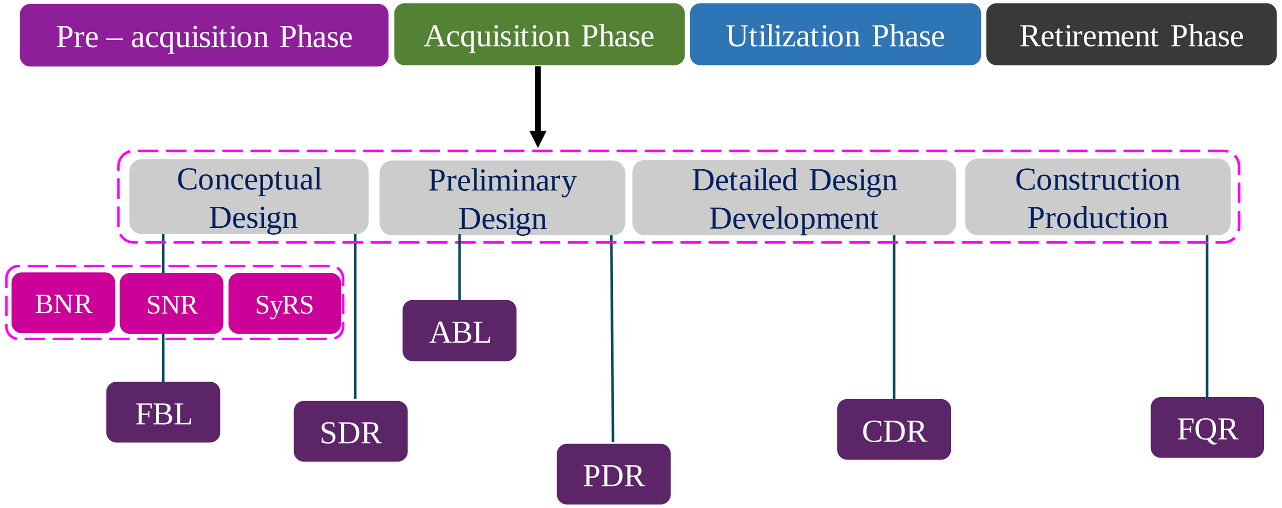


System's Mission broken down into a hierarchical structure in terms of physical components
Physical Architecture

Construction Production

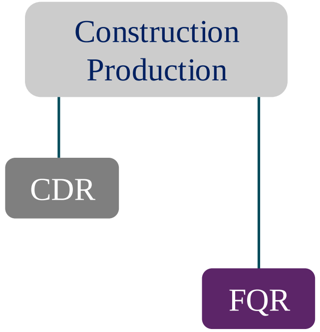
Consulting the Product Baseline [PBL | CDR], the components of the System produced in accordance with design specifications in CDR
From System to Sub-Systems to Assemblies to Components – Top Down Approach
This phase ends with the Formal Qualification Review [FQR] – customer accepts the System on the basis of the FQR
Testing + Evaluation carried out to ascertain final system configuration compliance with the requirements in SyRS





Documentations required in the Acquisition Phase:

- SDR [System Design Report]
- PDR [Preliminary Design Report]
- CDR [Critical Design Report]
- FQR [Formal Qualification Report]



Activities - Phases of Systems Life Cycle

Utilization Phase — Operational Support | Systems Support

In the Utilization Phase they main activities for a System's Engineer are restricted to:
- Operational Support
- Systems Support

In the utilization phase – the System is in service i.e. it has been implemented.
The System's engineer needs to make sure that the implementation is done right!

The System Engineer | Architect has a holistic perspective of the implemented System.
He or she knows the sub-systems inside out + is the best person for troubleshooting | diagnosing the System

System Engineering activities that are particularly important in this phase include:
- Modification of any sub-system
- Rectify performance shortfalls
- Meet changing operational requirements
- Training the personnel to whom the System is delivered

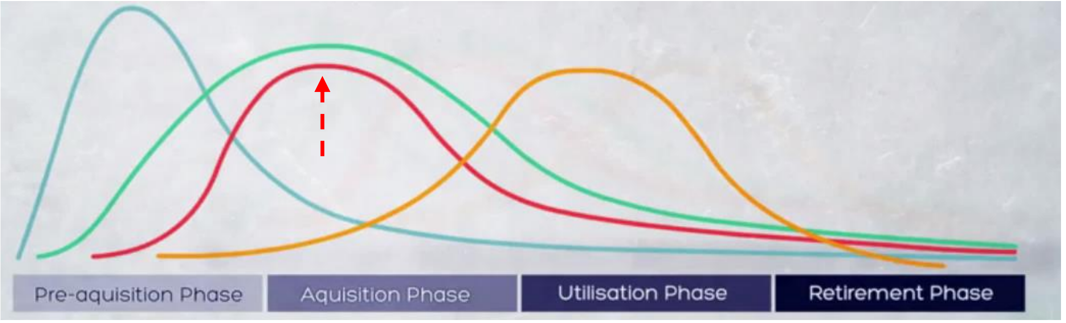
Systems of System
SoS

Retirement Phase — Complete retirement | Integration into another System

The role of a Systems Engineer is limited in this phase but NOT redundant.

Either the implemented System would be completely shutdown OR it will be integrated into another System to serve THAT system's mission [SoS].

The System Engineer who designed the 'soon to be retired' System is aware of the System's capabilities and provide important feedback and support during the System integration in the Retirement Phase.



Systems Engineering ——
Project Management ——
Operations Management ——
Business Management ——

Documentations required in the Acquisition Phase:
- SDR [System Design Report]
- PDR [Preliminary Design Report]
- CDR [Critical Design Report]
- FQR [Formal Qualification Report]


NASA
Systems Engineering
Handbook

Pre – acquisition Phase | Acquisition Phase | Utilization Phase | Retirement Phase

The main phase – where ONLY a System Engineer can contribute – is the Acquisition Phase!!

That's why NASA keeps a separate department for Systems Engineering.

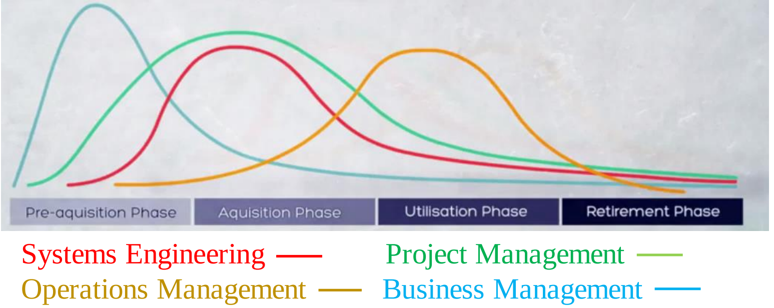
So what exactly is Systems Engineering? – Formal Definitions

In the previous lecture we got to know that a System's Engineer role is predominately relevant in the Acquisition Phase of the System Life Cycle

A System's Engineer is predominately concerned with:

- Conceptual Design
- Preliminary Design
- Detailed Design and Development
- Construction and Production


Acquisition Phase
Conceptual Design
Preliminary Design
Detailed Design Development
Construction Production



AND we already saw that in all these phases of the System's Life Cycle – there are many documentations to be produced.
Documentations help to standardize the process so that it is comparable and can be diagnosed | troubleshooted

So what exactly is Systems Engineering?
Let's look at formal definitions [you are expected to at least remember one]

"Systems engineering is the management function which controls the total system development effort for the purpose of achieving an optimum balance of all system elements. It is a process which transforms an operational need into a description of system parameters and integrates those parameters to optimize the overall system effectiveness"

Management Function | Controls System Development Effort | Balancing System Elements | Operational Needs to System Parameters

"An interdisciplinary approach encompassing the entire technical effort to evolve and verify an integrated and life cycle balanced set of system, people, product, and process solutions that satisfy customer needs. Systems engineering encompasses: the technical efforts related to the development, manufacturing, verification, deployment, operations, support, disposal of, and user training for, system products and processes; the definition and management of the system configuration; the translation of the system definition into work breakdown structures; and development of information for management decision making"

Interdisciplinary | Technical | Stakeholder Needs | System Life Cycle | Defining + Managing the System Configuration | Work Break Down Structures | Decision Making

"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: operations, cost and schedule, performance, training and support, test, manufacturing, and disposal. SE considers both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs"

Stakeholder needs | Defining Needs + Functionality in the Development Cycle [Acquisition Phase] | Documentation | Validation | Complete System Perspective

Top – Down Approach – Very Effective

Traditionally engineering design methods dependent on a BOTTOM - UP approach:

Components ⟶ Assemblies ⟶ Sub-Systems ⟶ System

This approach is very effective if we have a simplistic well-defined problem + straightforward
BUT if we have a complicated System + interconnected dependencies it cannot be solved with the bottom-up approach

Bottom-up approach is dependent on:

- existing components - which are selected + integrated into the System
- less complexity
- Sub-elements are often locally optimized as compared to global optimization for a System's Mission

The Top-Down approach works very well for complex Systems since it:

- Looks at the System as a whole | holistic in nature
- Understands each sub-system through the System's Mission
- Understands sub-systems and their interaction with System's environment + interfaces

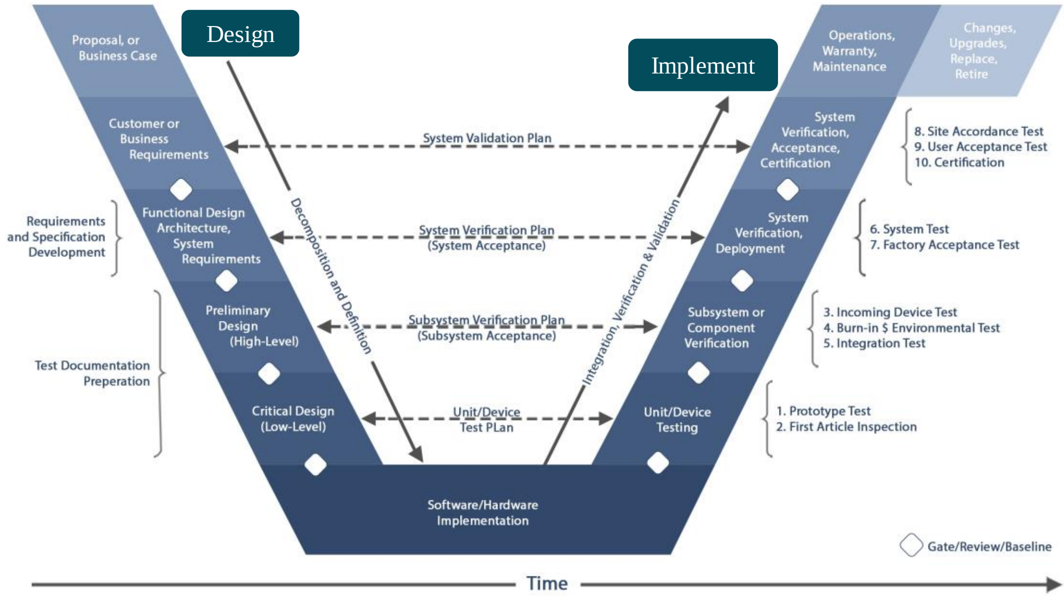
Top – Down approach requires the formation of System – Level Requirements [Bottom Up doesn't]
Each subsequent level of the System from Top to Bottom is understood with respect to the System's Mission

System ⟶ Sub-Systems ⟶ Assemblies ⟶ Components

Designing a System

Recurrent themes in Systems Engineering definitions:

- Top-down approach
- Requirements engineering
- Life-cycle focus
- System optimization and balance
- Integration of specializations and disciplines
- Management



The V-Model for Systems Engineering
Design Down – Integrate Up

Implementing a System

For System Integration we go Bottom – Up
At each step of the integration – Sub-System Level Architecture – integration testing is carried out to ascertain successful integration
Components ⟶ Assemblies ⟶ Sub-Systems ⟶ System

Design DOWN – Integrate UP This is the most important feature of Systems Engineering | System Thinking

The V-Model for Systems Engineering
The V-model summarizes the main steps to be taken in conjunction with the corresponding deliverables within the system framework.
It describes the activities to be performed and the results that have to be produced during System development.

Lecture 7 + Lecture 8

ENGR 301 – Introduction to Systems Engineering
Spring 2019

Requirements Engineering 1

Ali Zahid

Tuesday 26th and Thursday 28th February 2019 @ 15 30 – 16 45

HABIB UNIVERSITY

Agenda

Lecture 7 & 8 - Tuesday 26th February 2019 & Thursday 28th February 2019

- Benefits of Systems Engineering
- Requirements Engineering
- Systems Engineering Framework & Process
- The Systems Engineering V-Model
- Quiz

Things I will ask with a 100% probability ! 😈

Benefits of Systems Engineering – Yes there are!

Yes there are various benefits of Systems Engineering – that's why many large-scale operations | companies | organizations predominately use Systems Engineering

The benefits of Systems Engineering stems from the distinct phases of the Systems Engineering Life Cycle where a Systems Engineer is effectively required

Acquisition Phase – Conceptual Design | Preliminary Design | Detailed Design Development | Construction Production

Utilization Phase – Operational Support | Systems Support

The Acquisition Phase + Utilization Phase
The predominate domain for Systems Engineering

So how does the Systems Engineering principles benefits an organization?

- Life Cycle Cost [LCC] Savings → Systems Engineering focuses on the ENTIRE life cycle
  Some phases are more predominant BUT the life cycle focus is the main attribute of SE
- Reduction in overall Acquisition Schedule
- Reduction in Risk [extensive risk management in Systems Engineering at the Design + Integration Phases]
- Quality System

There is also an argument that SE adds too much Complexity + Cost + Time to the Acquisition Phase

TRUE!! The Acquisition Phase requires the most effort. BUT it pays off when you don't have any problems in the Implementation Phase.

'Benefits of Systems Engineering far outweighs any costs' – Boeing

SE allows us to get the requirements (Requirements Engineering) correct + precise early in the acquisition phase and monitors its (requirements) integration into the subsequent design + has a built in Risk Management Procedure

Reduces potential for costly + time consuming changes in the Implementation Phase

The SE framework has to balance the Ease of Making Changes and the Cost of Making Changes

It is very important to know when the changes in SE life cycle can be made !!!

BUT above all this the main purpose of SE is to bring forward a 'quality system' into service

Quality → measure of the fitness-for-purpose

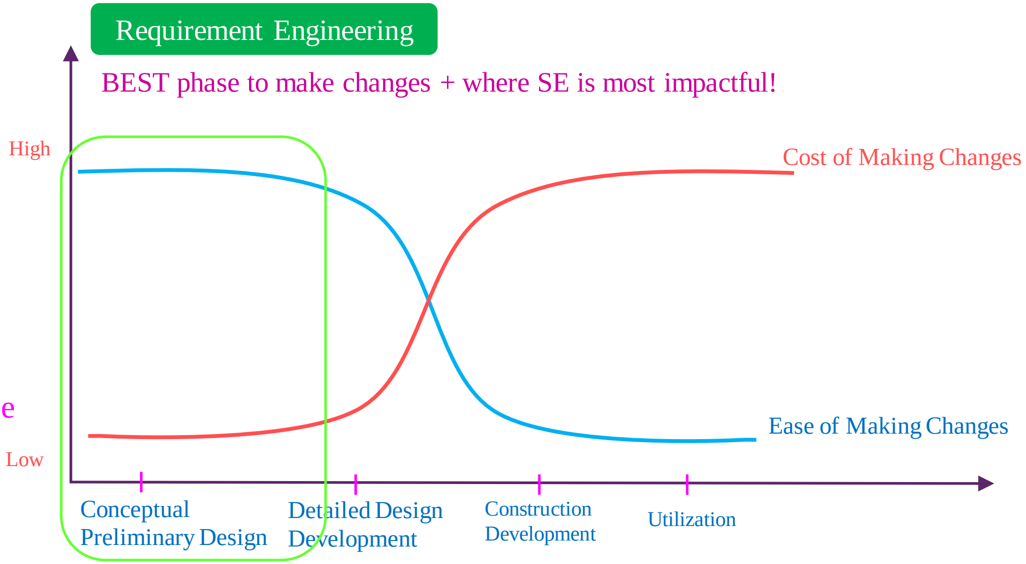
By maintaining a focus on Requirements + their traceability into design – SE helps to ensure:

- Specified Requirements accurately reflect the business + stakeholder NEEDS
- The resulting System meets the specified requirements

Requirement Engineering is apparently a very important part of the Conceptual + Preliminary Design Phase

Requirements Engineering is the rigorous process which manages the transitions from needs into statements of requirements which form the basis of the functional + physical design.

NEEDS → **Logical Design** + **Physical Design**


Requirement Engineering

BEST phase to make changes + where SE is most impactful!

Benefits of Systems Engineering – What does Boeing says about it?

INCOSE International Symposium

Session 8: Track 5 Session 8: System Architecture

5.8.4 The Boeing Standard Systems Engineering Approach To Systems Architecting and Its Application To The V-22 Osprey

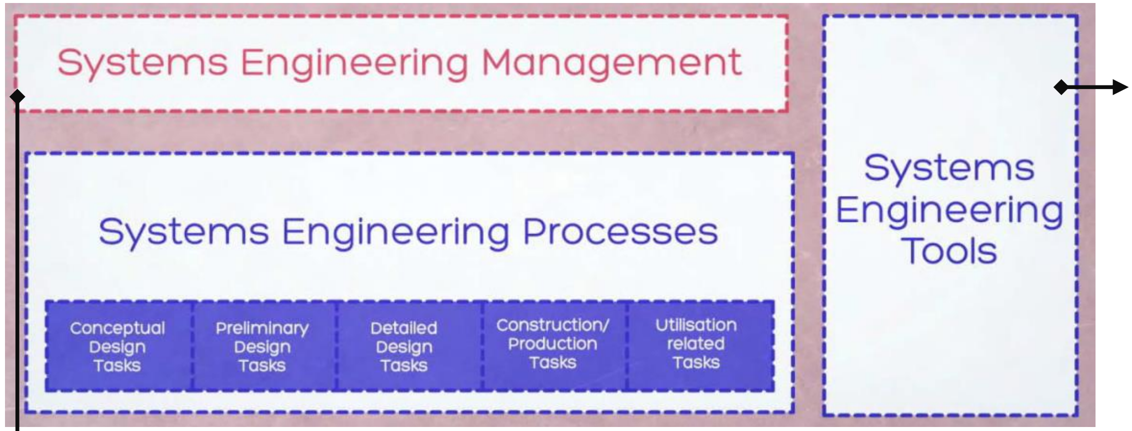
Robert S Bruff, Dr Cihan H Dagli

Architecture and Systems Engineering: Models and Methods to Manage Complex Systems

A four-course online program leading to a Professional Certificate from the Massachusetts Institute of Technology.

System Engineering Processes + Framework

The one problem with Systems Engineering is with its broad mandate – it gets confusing !
What is included what is not? Does a Systems Engineer have to manage people, finances or does he or she only has to design things on paper + software?
SE has a very broad application spectrum – we need to know its framework!

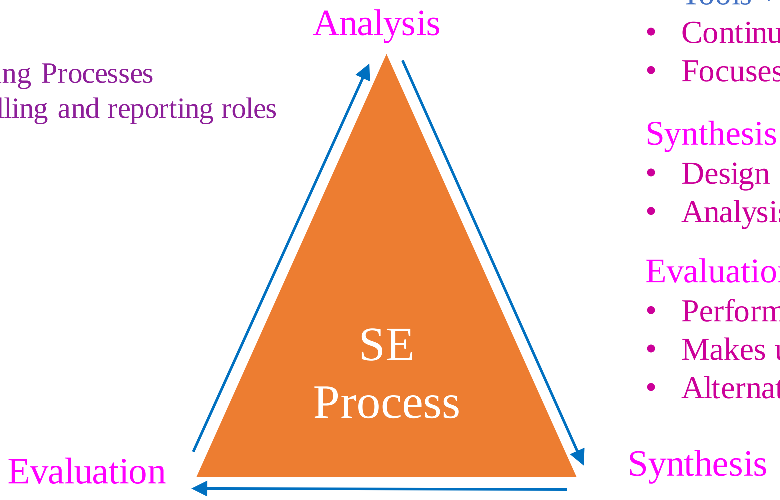


Tools are available to help System Engineers for undertake their tasks effectively
Tools include Techniques + Processes + Information Systems + Standards
Tools are distinguished into → Tools for Management + Tools for Processes
Management Tools → includes popular systems engineering standards + capability maturity models
Process Tools → includes requirement management systems + analysis * synthesis * evaluation tools

Oversees the ENTIRE Systems Engineering Processes
Plays a role monitoring, directing, controlling and reporting roles

Is responsible for:

- Technical Reviews + Audits
- Testing + Evaluation
- Technical Risk Management
- Configuration Management
- Implementing Standards
- Integration Management



Systems Engineering relies on the continual application of simple problem-solving process of analysis + synthesis + evaluation

Analysis

- Commences with the perceived need of the System
- Development of formal requirements – what the system going to do + constraints + interfaces
- Tools + techniques that are going to applied
- Continues through out the Acquisition Phase
- Focuses initially at the LOGICAL design and then progresses to the PHYSICAL design

Synthesis

- Design + Creation function of SE
- Analysis tells you what you NEED to do – Synthesis tells you HOW to do it

Evaluation

- Performed throughout the Acquisition Phase
- Makes use of DOCUMENTATIONS
- Alternatives + decision evaluation + tradeoffs between requirements & design

This Analysis – Synthesis - Evaluation loops is applied iteratively throughout the SE Life Cycle

Benefits of Systems Engineering – How does Tokamaks benefit from it?

> "Fusion Energy is going to be the most important source of electricity in the 22nd Century.
>
> Dr David Kingham, Co Founder – Tokamak Energy"

tokamak energy

Lecture 9 + Lecture 10

ENGR 301 – Introduction to Systems Engineering
Spring 2019

Requirements Engineering 2

Ali Zahid

Thursday 21st and Tuesday 26th March 2019 @ 15 30 – 16 45

Agenda

Lecture 9 - Thursday 21st March 2019

- Needs vs Requirements

Lecture 10 - Tuesday 26th March 2019

- What is a Requirement?
- What exactly is Requirements Engineering?

Things I will ask with a 100% probability !

Needs vs Requirements [Documentations]

There is a technical difference between Needs & Requirements. The difference arise from a distinction of what is being achieved at different Sub-System levels

Needs → capabilities defined in business language at the business management | business operations level [Stakeholder Needs | Business Needs]
Requirements → formalized structured statements that can be validated, catering to a need
There can be more than 1 REQUIREMENT defined for any NEED.

NEEDS ↔ Requirement Analysis ↔ REQUIREMENTS

Enterprise	Enterprise Leadership sets the enterprise strategies and concepts of Operations
Business Management	Management derives Business Needs | Constraints | Requirements formalized
Business Operations	Stakeholder define their Needs & Requirements
System Sub - System	Final system designed in a Logical + Physical view

Problem Domain
Defining + Analyzing the Problem

Responsibility of the owners
Logical | Functional Descriptions
WHAT needs to be solved?

Solution Domain
Finding Solutions for the Problem

Responsibility of the people implementing
System description – engineering + physical
HOW to solve it?

	Needs	Requirements	Translating Needs to Requirements
Enterprise	Enterprise Strategies		
Business Management	Business Needs	Business Requirements	Business Needs and Requirements Definition Process [BNR]
Business Operations	Stakeholder Needs	Stakeholder Requirements	Stakeholder Needs & Requirements Definition Process [SNR]
System	System Needs	System Requirements	System Requirements Definition Process
Sub - System	Sub - System Needs	Sub - System Requirements	

Problem Domain
Defining + Analyzing the Problem

Solution Domain
Finding Solutions for your Problem

NEEDS must be translated into REQURIEMENTS – therein comes Requirement Analysis

At each responsibility level, the needs at THAT level must be translated to Requirements. We need a formalized + standardized format for requirements so that the later stages can track.
ALWAYS know in which domain the System is with respect to the System Life Cycle – Solution Domain OR Problem Domain

Problem Domain – think logic | think functions | think System's purpose | think needs (stakeholder and business)
Solution Domain – think components | think what to do for implementation | think engineering + physical designs

Needs vs Requirements [Documentations]

Enterprise			Concept of Operations [Con-Ops]
Business Management	Business Needs & Requirements Definition Process [BNR]	Business Requirement Specification BRS	Preliminary Life-Cycle Concepts [PLCD] + Ops-Con
Business Operations	Stakeholder Needs & Requirements Definition Process [SNR]	Stakeholder Requirement Specification StRS	Life – Cycle Concepts [LCD]
System	System Requirements Definition Process	System Requirements Specification SyRS	Life-Cycle + Acquisition + Deployment +Support + Retirement
Sub - System			
	Translating Needs to Requirements →	Documents Required for transitioning	Needs being Translated

- Business Needs & Requirements Definition Process [BNR]

Business Needs are defined keeping into view the Organization's objectives + goals + vision

Organization's goals + intended purpose of leading the organization : Con-Ops [Concept of Operations] – long range strategic plans + annual operational plan + how to tackle the stakeholders

Considering the Con-Ops, the Business NEED is translated into a REQUIREMENT through the Preliminary Life-Cycle Concept Documents [PLCD]

PLCD tackles 2 very important documents = Preliminary Acquisition Concept + Preliminary Operational Concept Ops-Con

- Stakeholder Needs & Requirements Definition Process [SNR]

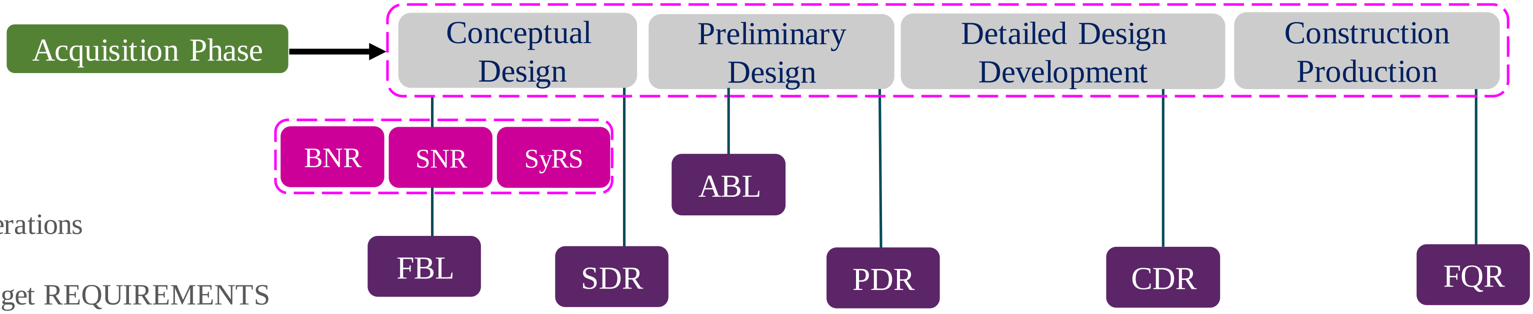
Using the Con-Ops + PLCD + Life Cycle Documents [LCD], the Stakeholder Needs are translated into formal structured set of Stakeholder Requirements through the Stakeholder Requirement Specification [StRS]

- System Requirements Definition Process

Requirements in StRS are transformed into System Requirements and documented in the System Requirement Specification [SyRS]

One of the most important translation of Needs at different levels is the transition between Con-Ops [@ the Enterprise Level] → Ops-Con [@ the Business Management Level]

Con-Ops provides the context for Ops-Con
Ops-Con contains the NEEDS of the system within its business context
When translated into requirements the Ops-Con forms the BRS



Let's make it simple:
There are NEEDS and there are REQUIREMENTS
NEEDS must be translated into REQUIREMENTS
NEEDS start from the Enterprise to the Business Management to Business Operations
NEEDS at each level contains the NEEDS of the previous level
NEEDS at each level decides what PROCESS must be undertaking in order to get REQUIREMENTS

So what exactly is a Requirement ?!?

Requirements are important since they are formalized structured statements that can be validated, catering to a need
Requirements result from a formal transformation of one or more needs into an AGREED-TO obligation for an entity to perform some function | posses some quality

Requirements translates the NEEDS [which are defined in business language by the Enterprise + Business Management + Business Operations] into:

- Something the System MUST do
- Quality | attribute that a System MUST posses →
- Constraints in which the System MUST be operational

These are all decided by NEEDS

- Stakeholders Needs → NEEDS precedes Requirements → NEEDS always must be translated to REQUIREMENTS
- Business Needs + Enterprise Needs

There are 2 kinds of requirements:

i. Functional Requirements → something the System SHOULD do or PROVIDE (hugely dependent on the purpose | mission of the System)
ii. Non-functional Requirements → a condition the System should meet; a property, quality or attribute the System must have; constraints under which the System operates

Requirements Statements [Business, Stakeholders, System] are supported by + are inclusive of:

- Performance + verification + rationale statements in support of each requirement
- Definitions of other Systems which the System integrates with + to which it interfaces
- Information about the application domain in which the System must operate

Adjectives are often added to describe the nature of the requirement

Operational Requirements Safety Requirements System Requirements
Stakeholder Requirements Design Requirements Regulatory Requirements

A Requirement gives information about the WHAT not the HOW. It is an integral part of the Problem Domain.
We need requirement analysis AND requirements because:

- It defines the SCOPE of the System
- Everyone involved in the System should have a valid INPUT [what are they doing??]
- Justify expenditure
- Report on progress
- Able to decided termination