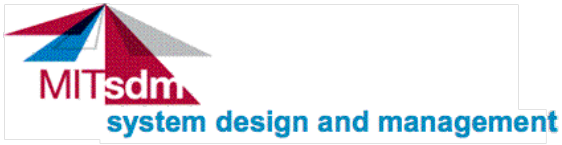
ESD.33 -- Systems Engineering

Session #4
Requirements Engineering

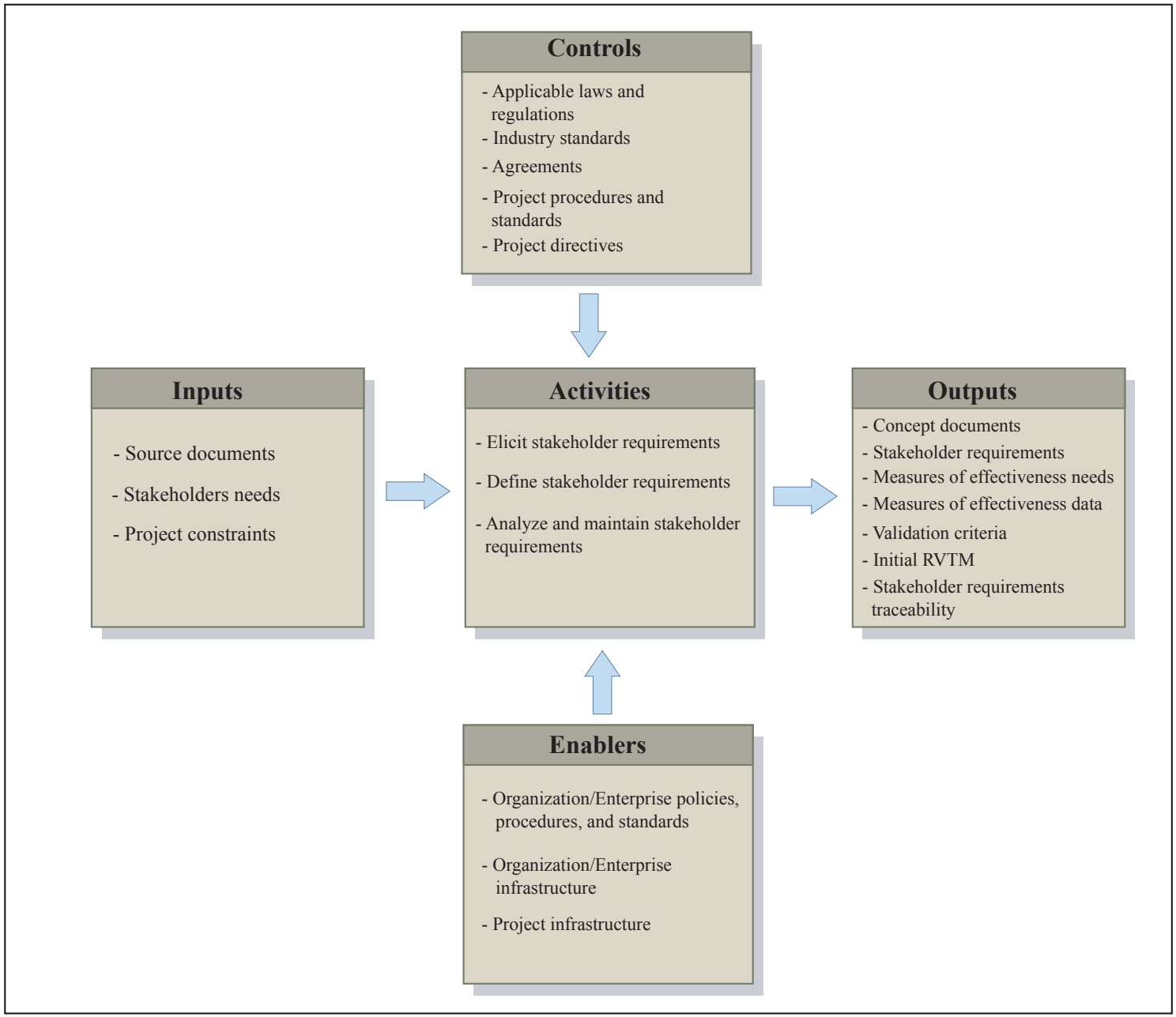
Pat Hale



Purpose

As stated in ISO/IEC 15288:2008: *The purpose of the Stakeholder Requirements Definition Process is to define the requirements for a system that can provide the services needed by users and other stakeholders in a defined environment. It identifies stakeholders, or stakeholder classes, involved with the system throughout its life cycle, and their needs, expectations, and desires. It analyzes and transforms these into a common set of stakeholder requirements that express the intended interaction the system will have with its operational environment and that are the reference against which each resulting operational service is validated*

What is the RE Process?



INCOSE Systems Engineering Handbook version 3.2 January 2010

Image by MIT OpenCourseWare.

Why ‘Engineer’ Requirements?

- Provide early assurance that all top-level requirements are fully satisfied in the product, with traceability to where they are satisfied.
- Prevent unintentional addition of cost (avoid ‘gold plating’).
- Establish clear responsibility for requirements implementation.
- Provide clear visibility across teams into requirements allocation and cross-functional interactions.
- Easily and quickly assess the impact of changes to requirements.
- Provide data for early and thorough validation and verification of requirements and design artifacts.
- *Avoid unpleasant downstream surprises!*

Requirements Management

Requirements Management is the process by which:

- System/Product needs are captured from the market, business strategy, customers and other sources.
- Requirements are synthesized from captured needs and the meanings and interpretations are validated with the original sources.
- Commitment to requirements is obtained from management and project personnel.
- A history of changes and change rationale is maintained as requirements evolve.
- Bi-directional traceability is maintained among requirements, project plans and work products.
- Inconsistencies are identified and resolved among requirements, project plans and work products.

What is a Requirement?

- Well-stated requirements exhibit the following attributes:
  - The requirement is **Necessary**
    - What would happen if you didn't include this requirement?
  - The requirement is **Verifiable**
    - How will you know you have met the requirement?
  - The requirement is **Attainable**
- Requirements are concise and unambiguous
- Good requirements are solution-neutral
- Requirements are consistent (non-contradictory)

Requirements Semantics

- Requirements use '*shall*'
  - *The system shall provide…*
  - *The system shall be capable of…*
  - *The system shall weigh…*
- Statements of fact use *'will'*
  - Often part of a lead-in scenario prefacing one or more requirements
- Goals use *'should'*
  - Used for design goals where quantifiable requirements cannot be applied to a desired attribute

Summary: Attributes of Good Requirements

- What…not how
- Unambiguous, clear and complete
- Verifiable by test or independent analysis
- Necessary and appropriate for system hierarchy level
- Feasible
- Consistent (traceable) with requirements above and below in the system hierarchy

Requirements Baselines

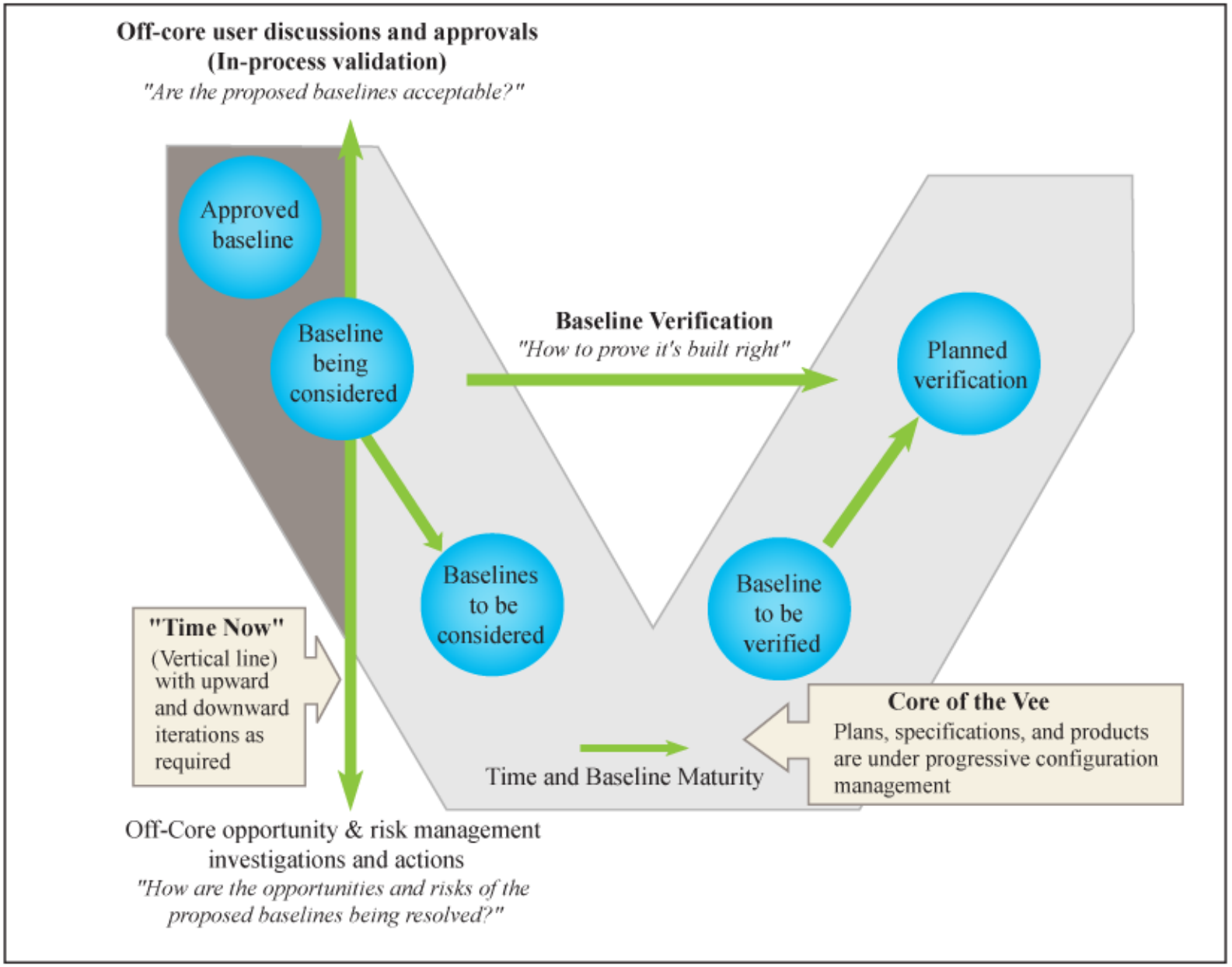
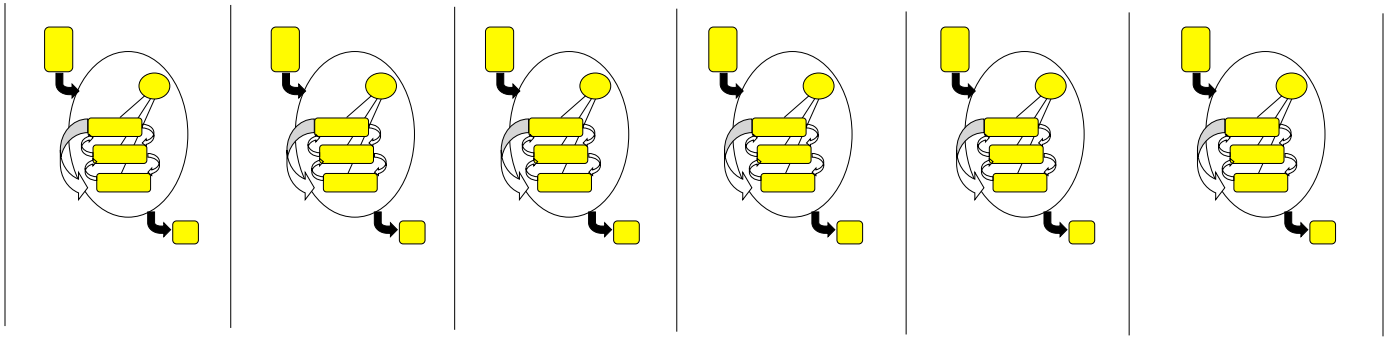


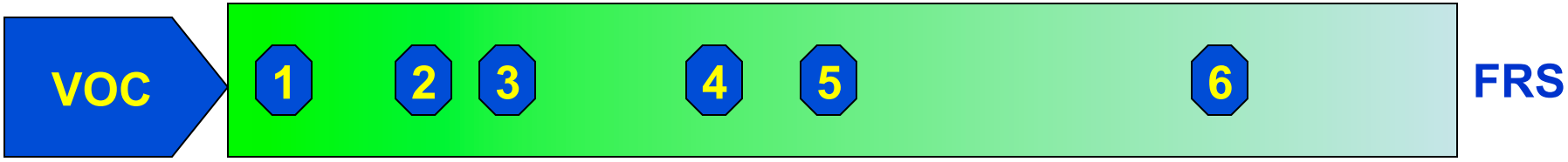
Image by MIT OpenCourseWare.

Product Commercialization
Phases 1-6:

Requirements Baselines in CDOV

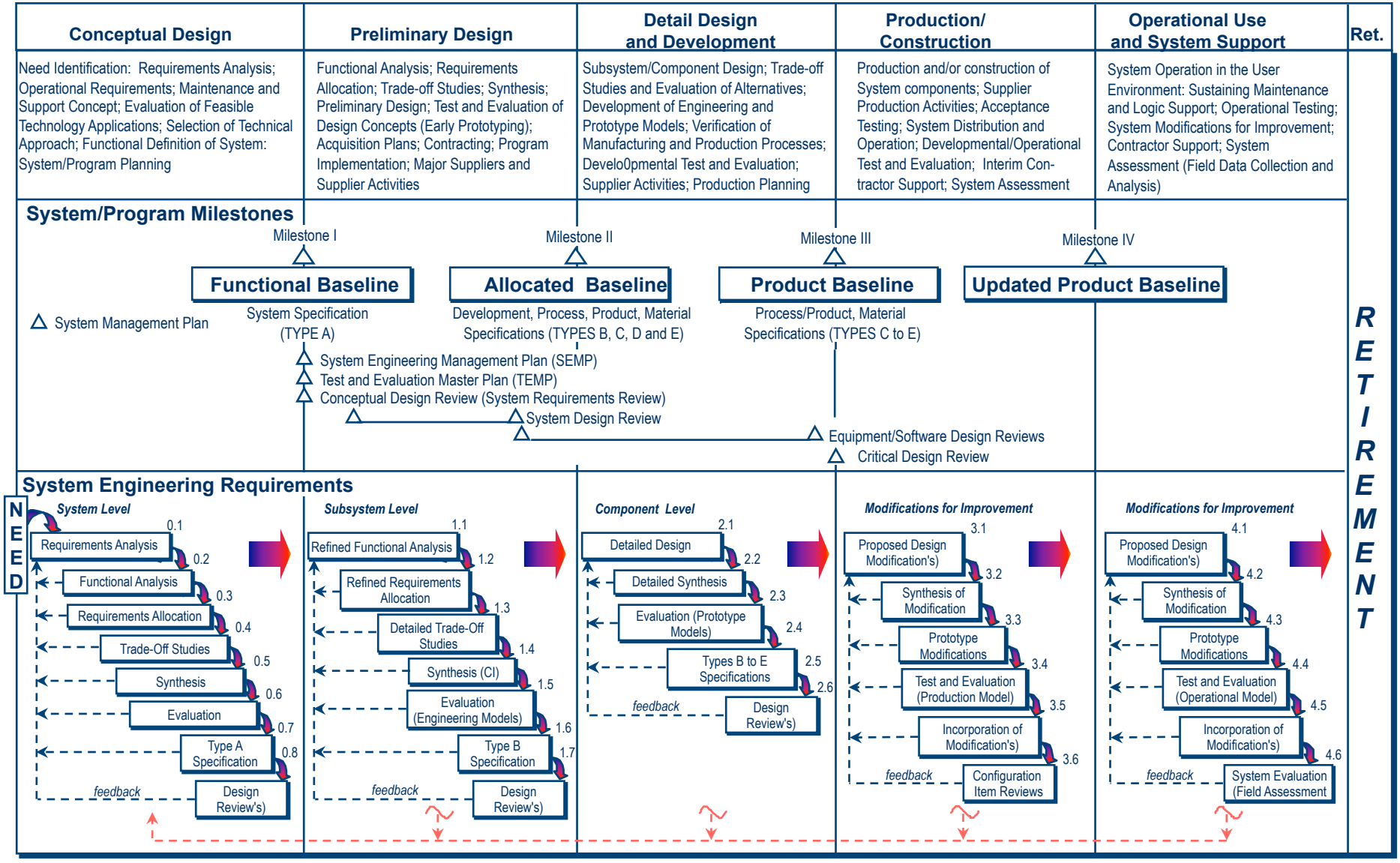
Concept	Design	Optimize Sub-Systems	Optimize System	Verify Product Design	Verify Production Process





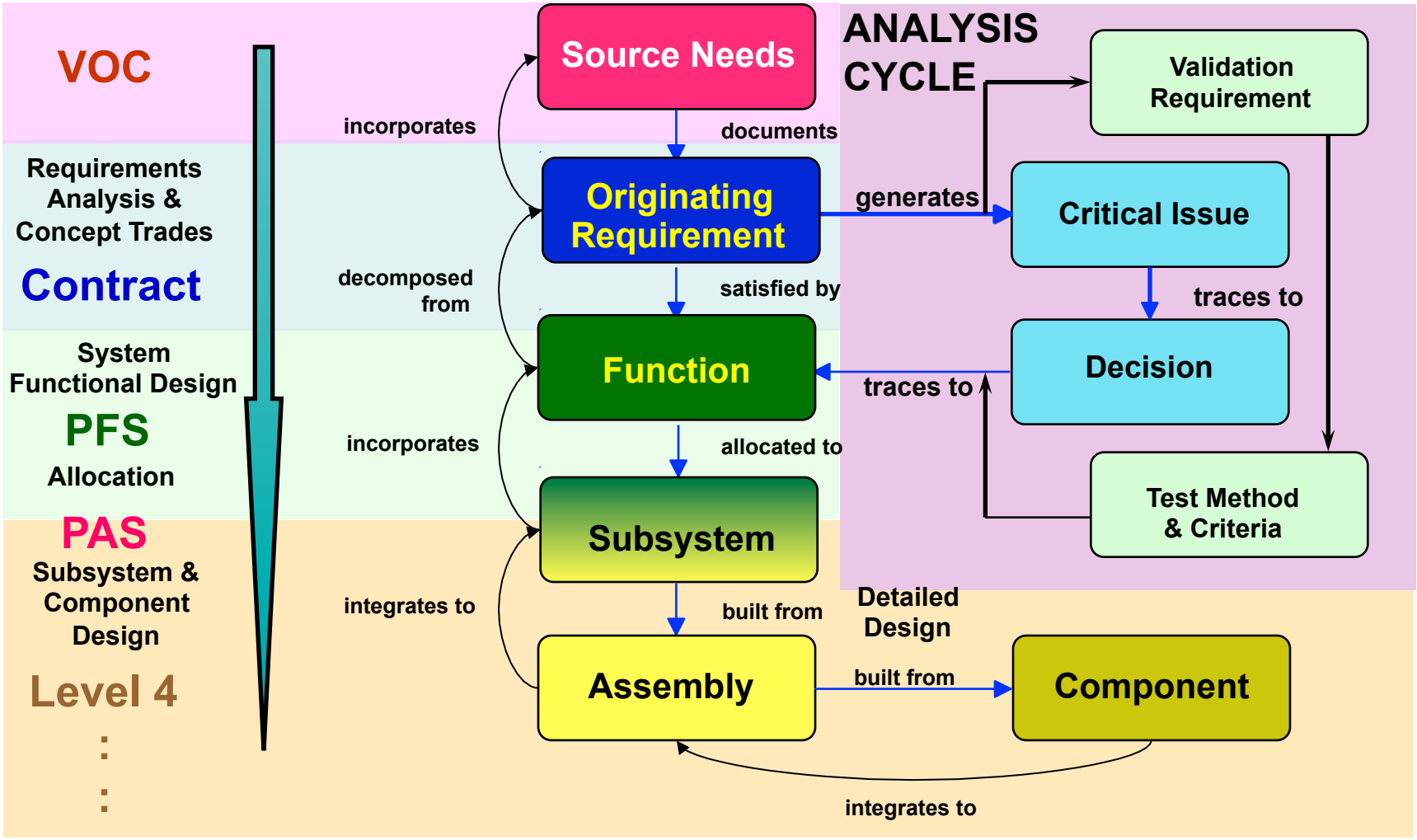
1- Initial baseline
2- Validated baseline (Contract)
3- System functional baseline (PFS)
4- Allocated baseline (PAS)
5- Component baseline
6- Verified product baseline

System Life Cycle

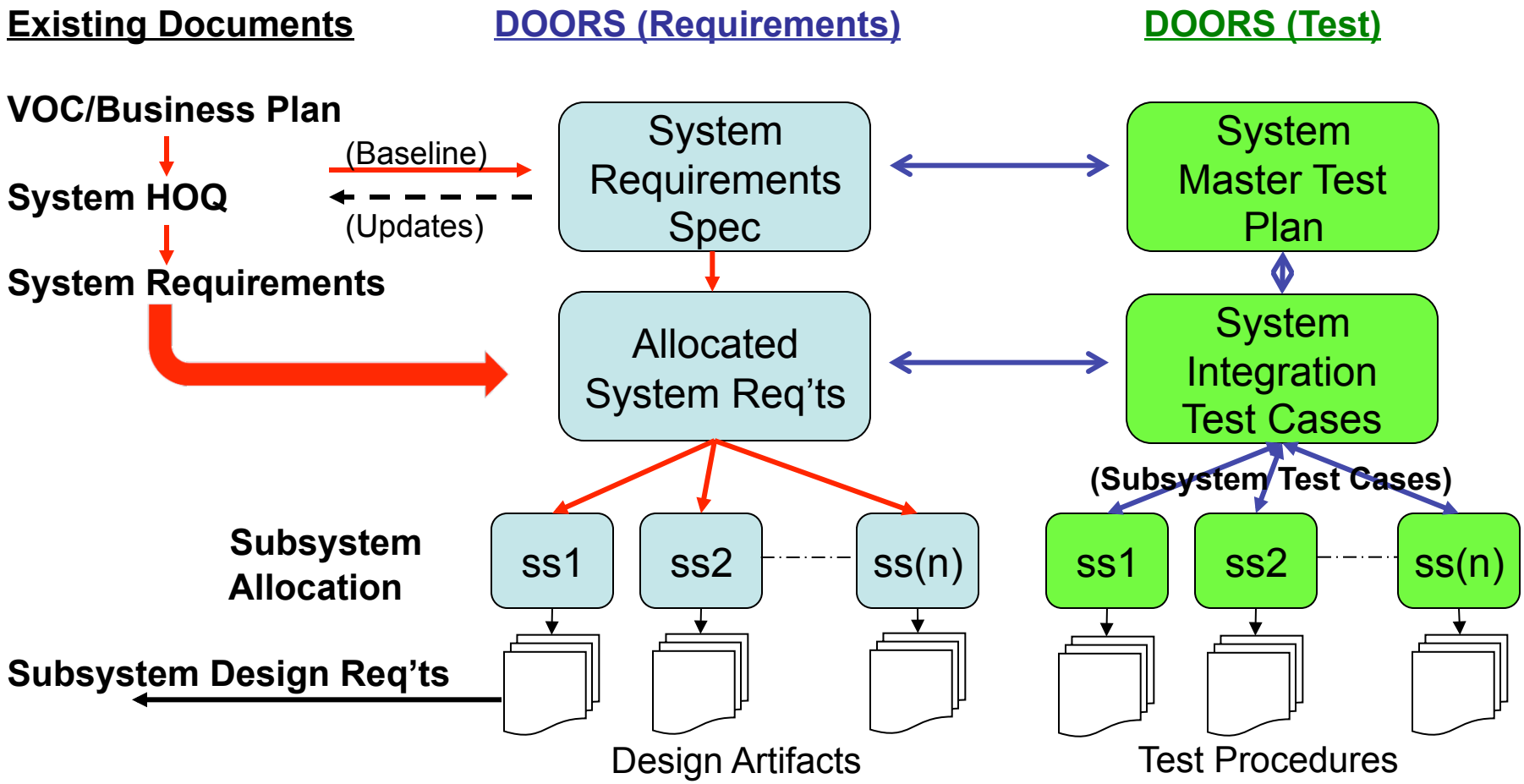


Adapted from Systems Engineering lecture slides at University of Witwatersrand, Johannesburg, South Africa (Dr. R. Siriram)

Requirements Hierarchy & Traceability



The requirement is **Verifiable**



Requirements Change Authority

VOC

Baseline
Requirements Analysis & Concept Trades

Contract

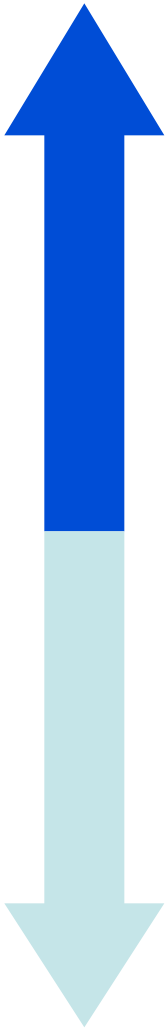
System Functional Design

PFS

Allocation

PAS
Subsystem & Component Design

Level 4
:
Detailed Component/Part Specifications



Pre-PAS:
- Changes/new requirements proposed by any development team member
- Review by Systems team and other Stakeholders (e.g. PMT members) as required
- Approval by Systems team/Chief Systems Engineer

Post-PAS:
- Changes/new requirements proposed by any development team member
- Review by Subsystem team with appropriate representation from other affected teams
- Decision by Subsystem team leader with primary design responsibility
- Weekly review of changes and change proposals by Systems team

Implementation Plan

Task Description	Who?[1]	Target Date
• Propose Draft RM process	JW, Cons.	• 10/31/2003
• Select DOORS versions, obtain training	LD	• 11/3/2003
• Develop requirements schema for RM & CPM (link, no common schema)	JW, SS, Cons.	• 11/12/2003
• Review and Approve Change Review Process	JC	• 11/17/2003
• Review/Import existing requirements, refine to meet RM criteria	JW, SS, KH, Cons.	• 12/28/2003
• Certify PFS baseline in DOORS	JW, DD, Mkt, Test	• 1/10/2004
• Full process implementation	NGD Team	• 1Q04

System-level Implementation

(Example from a backup tape drive project)

Documents to DOORS Modules

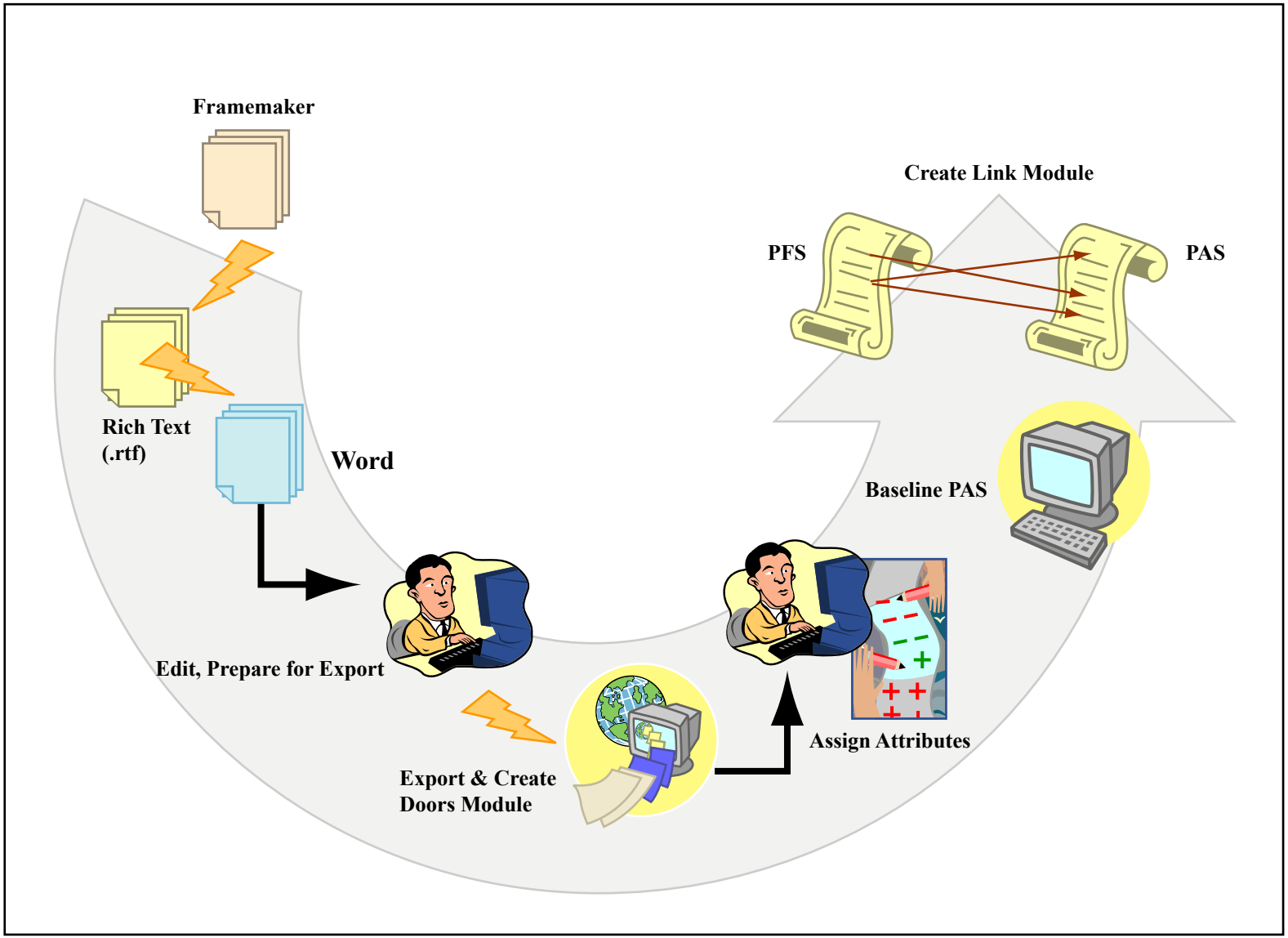


Image by MIT OpenCourseWare.

NGD Doors Directory Structure

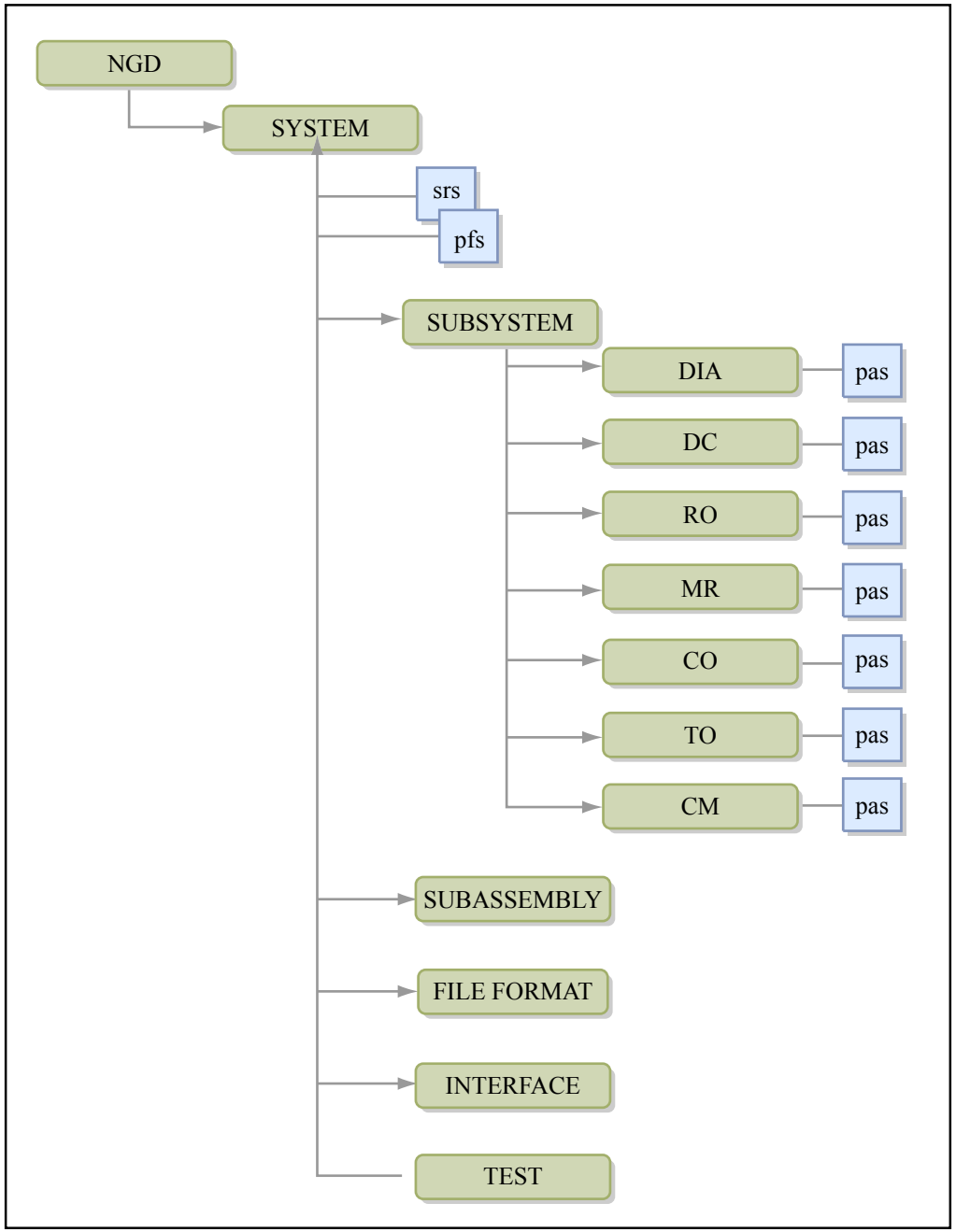


Image by MIT OpenCourseWare.

Formatting PAS in Word for Export into Doors

Keep heading styles in the following form:
Heading 1, Heading 2, Heading 3, …,etc.

Notes:

- Framemaker documents use Sec-#, Sec-#.#, Sec-#.#.#. Export Framemaker documents to Word and convert them to Heading 1, Heading 2, etc. before exporting into Doors.
- Keep all other styles consistent throughout (e.g. figure heading, table heading, etc.)

The System Engineering Process

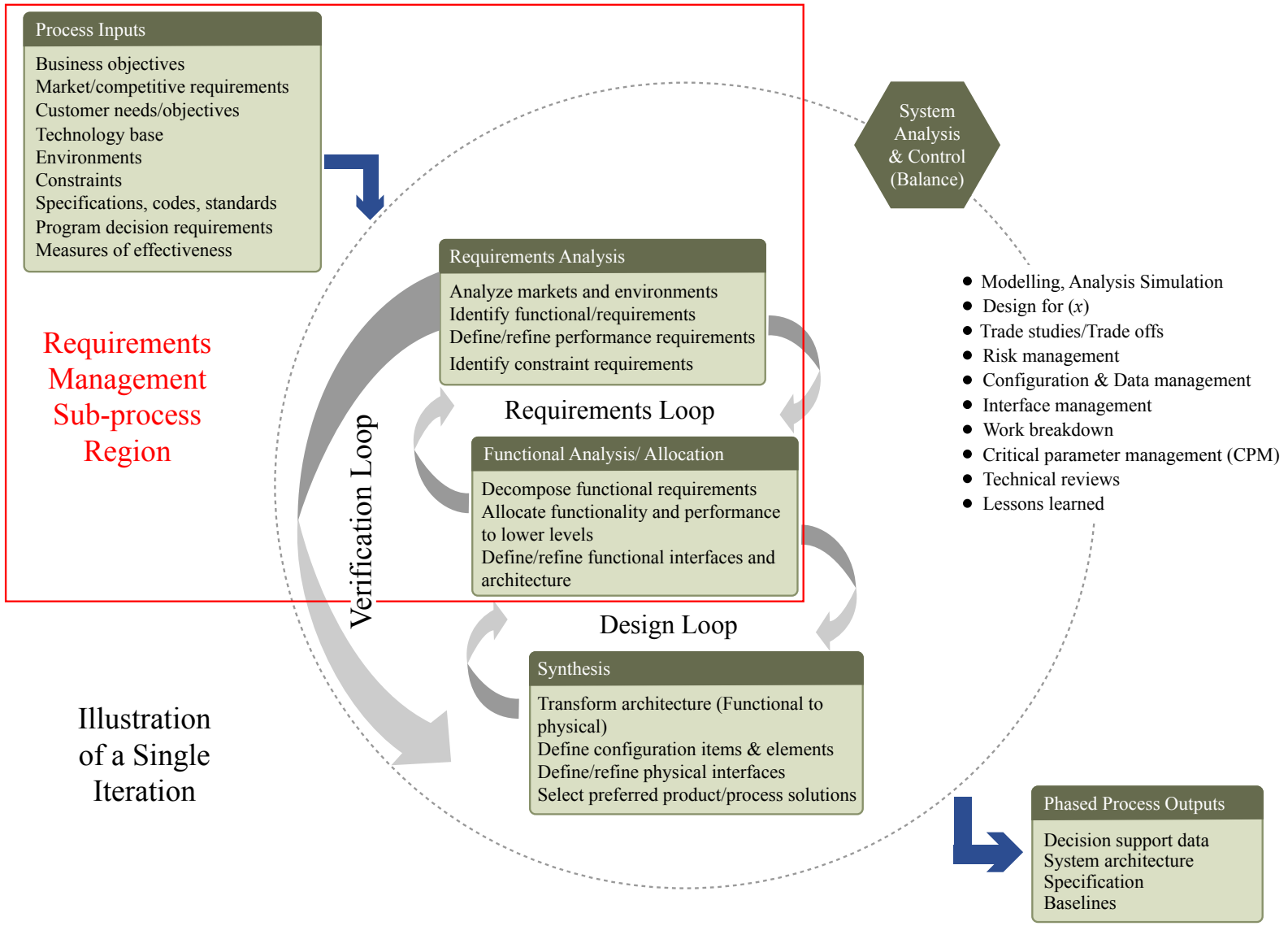
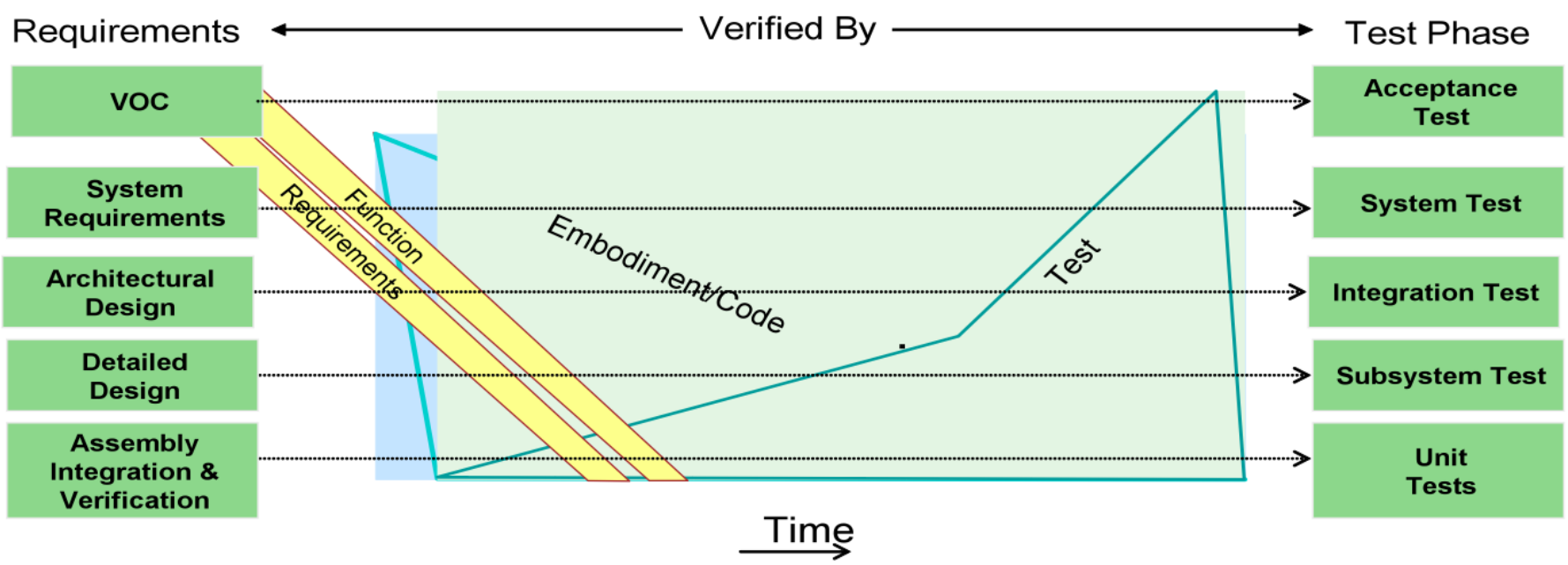


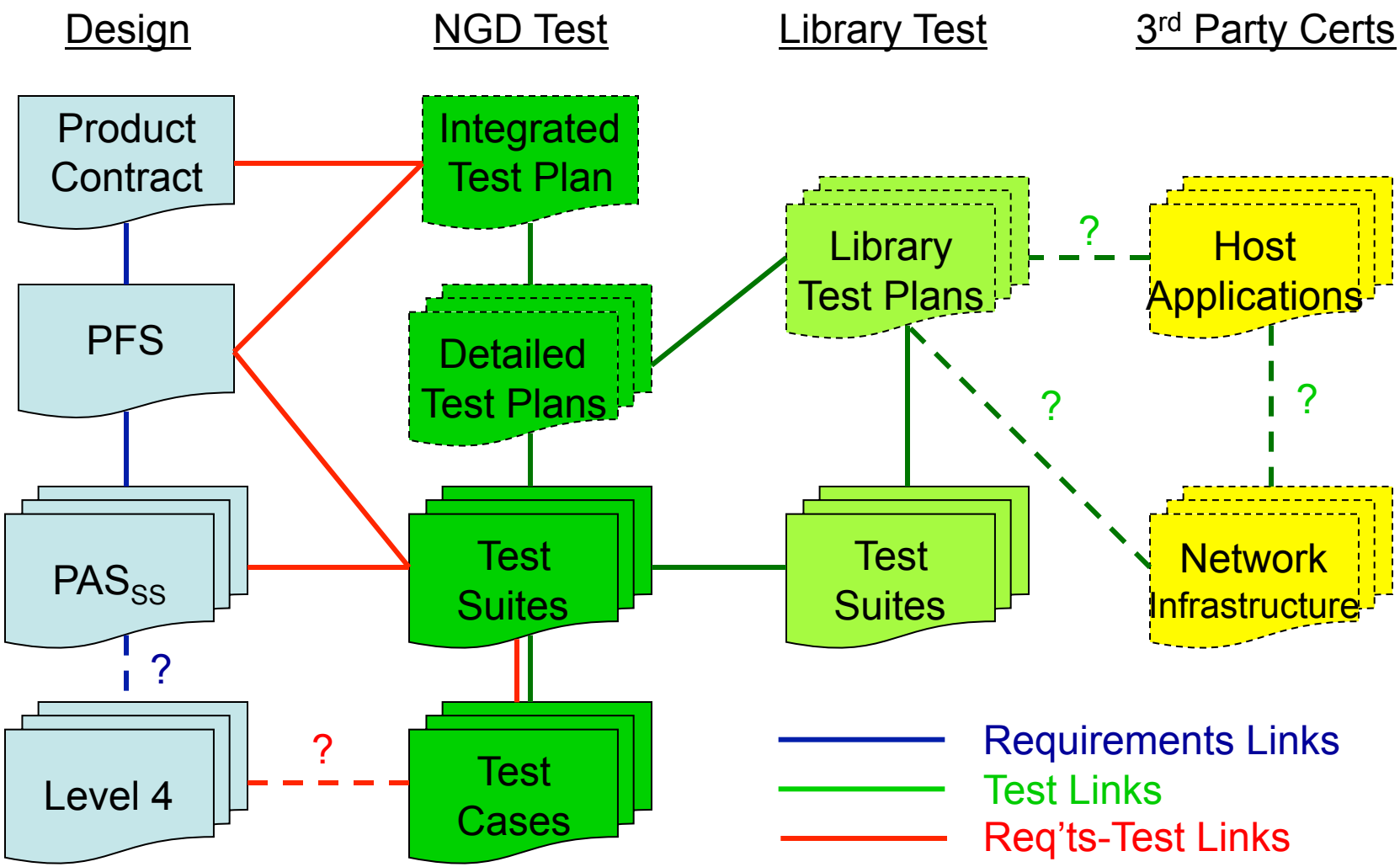
Illustration of a Single Iteration

Image by MIT OpenCourseWare.

Another View of the ‘V’ Model



NGD Requirements Structure & Links



Module Schema

Test Suite Module

Field	Type	Req'd?	Inherit	Description
Object_ID	Index	Auto	N/A	Prefix w/number assigned by DOORS
Test_Name	Text	Yes	No	Shorthand nomenclature for test suite
Req_Source	Enum/1	Opt.	Yes?	Values= PC, PFS, PAS(RO...)
Document	Text	Yes	No	External test reference document
Test_Purpose	Enum/M	Yes	Yes	Values= Reliability, Performance, Standards, Regulatory, Interface, Functional
Test_Environ	Text	Yes	No	Description of conditions and equipment, software environment necessary for test suite conduct

MIT OpenCourseWare
http://ocw.mit.edu

ESD.33 Systems Engineering

Summer 2010

For information about citing these materials or our Terms of Use, visit: http://ocw.mit.edu/terms.