# ENGINEERING SUCCESSFUL INFRASTRUCTURE SYSTEMS

## APPLYING A SYSTEMS APPROACH FOR ADDED VALUE

**5-DAY COURSE**

This course is a five-day immersion in the successful engineering of infrastructure that delivers maximum value to stakeholders. The course is based on systems thinking, here applied to projects and engineering. And so, the course has a strong systems engineering foundation. Systems engineering as a discipline has been proven to substantially reduce costs, reduce project durations and increase client satisfaction.


*"Infrastructure development is changing. There is more emphasis on integrated work on a life cycle basis, using a more diverse range of technologies, with more open communication between clients, contractors and other stakeholders. Developers are making improvements in efficiency and schedule by preventing rework and making better use of available products and knowledge. And benefits are flowing to customers through greater satisfaction of their needs. This course teaches how."*

*- Robert Halligan FIE Aust CPEng IntPE(Aus)*

PROJECT PERFORMANCE INTERNATIONAL

**www.ppi-int.com**

# COURSE OUTLINE

## 1. Concepts and Definitions

- defining the problem, and developing a complete solution
- life cycle basis of problem definition & solution
- example requirements and MOEs relating to an infrastructure-related capability
- example physical levels of solution definition relating to an infrastructure-related capability solution
- problem description definitions
  - definition: requirement/threshold
  - definition: measure of effectiveness
  - definition: measure of performance
  - definition: target/goal/objective
  - definition: value (effectiveness) model
  - definition: operational effectiveness
  - definition: requirement importance
- definition: requirements specification
- definition: design specification
- concepts related to problem description
  - definition: need
  - definition: want
  - definition: desire
  - definition: intent
  - definition: expectation
  - definition: constraint
- definition: CONUSE – Concept of Use (also called OCD, OpsCon)
- definition: solution
- definition: solution description
- definition: architecture
- definition: architectural design description
- definition: architecture framework
- definition: OSD – Operational Solution Description
- definition: detailed design
- definition: stakeholder
- definition: verification
- definition: verification requirement
- definition: verification requirements specification
- definition: validation

## 2. Systems Concepts

- systems thinking
- problem definition
- emergence
- divergence
- convergence

## 3. Applying Systems Concepts

- a system solution
- "systems of systems"
- systems of autonomously managed systems
- modeling with System Dynamics
- functional modeling
- process models
- styles of development
- critical infrastructure development
- life cycle models

## 4. Types of Requirements

- eight basic types
- ***workshop 1 - categorizing requirements for an infrastructure system by type***

## 5. The Quality of Requirements

- requirements quality attributes

## 6. Requirements Analysis for Infrastructure Systems

- purpose of requirements analysis and its relationship to CONUSE
- relationship to OSD
- requirements analysis (capture & validation) methodology
- context analysis, and relationship to CONUSE
- ***workshop 2 - context analysis for an infrastructure system***
- states & modes analysis
- ***workshop 3 - states and modes for an infrastructure system***
- parsing analysis
- ***workshop 4 - parsing analysis***
- functional analysis, and relationship to CONUSE
- ***workshop 5 - functional analysis for an infrastructure system***
- rest of scenario analysis
- ERA analysis
- out-of-range analysis
- value analysis
- ***workshop 6 - building an infrastructure system value (system effectiveness) model***
- operational effectiveness
- operational effectiveness versus overall effectiveness
- extracting information for a CONUSE and requirements document from users (and others)

## 7. CONUSE as a Document

### 7.1 Content and Purpose of a CONUSE

- users and uses of a CONUSE, in detail
- principles regarding content
- user stories, use cases, scenarios and the CONUSE
- how does support relate to a CONUSE?
- relationship of the CONUSE to the requirements document
- CONUSE standards and guides
  - Operational Concept Documents, DID DI-MCCR-80023, SDD Documentation Set – Data Item Descriptions for DoD-STD-2167, U.S.A. Department of Defense, 1985
  - Concept Data Item Description, SMA-DID-P100, NASA Product Specification Document Standard, Release 4.3, 1989
  - ANSI/AIAA G-043-1992, Guide for the Preparation of Operational Concept Documents, 1992
  - Operational Concept Description (OCD), DID DI-IPSC-81430, Data Item Descriptions for MIL-STD- 498, U.S.A. Department of Defense, 1994
  - IEEE Standard 1362, IEEE Guide for Information Technology – System Definition – Concept of Operations Document, 1998
  - ACC Instruction 10-650, Development and Use of Concepts of Operations, U.S. Department of the Air Force, 1998
  - Guide for the Preparation of Operational Concept Documents, ANSI/INCOSE/AIAA, G-043:2012
  - PPI's OCD/CONUSE DID
- who should prepare a CONUSE
- timing of preparation of a CONUSE versus requirements document

### 7.2 Preparing a CONUSE

- characteristics of a good CONUSE
- ***workshop 7 - review of sample CONUSEs***
- pitfalls in CONUSE preparation
- ***workshop 8 - review of a CONUSE for maritime infrastructure***
- ***workshop 9 - preparing a basic CONUSE***
- use of graphics in CONUSEs
- level of detail in the CONUSE
- design content – when, and when not?
- makeup of a CONUSE development team
- the role of users
- beyond the basic CONUSE
- extending the CONUSE to other stakeholders
- pitfalls in preparing CONUSEs

## 8. OPERATIONAL SOLUTION DESCRIPTION (OSD)

### 8.1 Content and Purpose of an OSD

### 8.2 Relationship of the OSD to Overall Solution

### 8.3 Styles of Solution Development

- the solution domain: key concepts, relationships, and work products
- ***workshop 10 - principles of OSD development***
- waterfall, incremental, evolutionary and spiral capability development approaches
- ***workshop 11 - infrastructure solution development strategies***

### 8.4 Concepts of Architecture – Physical and Logical, in OSD Development

- physical architecture (structural view) – basic concepts
- the role of technology and innovation
- techniques for stimulating innovation in solution development
- use of design driver requirements
- perspiration engineering: configuration items
- criteria for selecting configuration items
- relationship of CI definition to future system integration
- ***workshop 12 - physical conceptualization of an infrastructure solution***
- logical architecture – basic concepts of model-based architecting
- logical architecture related to physical architecture
- useful forms of logical representation – functional, state-based, mathematical, ...
- model-based design in practice

### 8.5 Functional Modeling in OSD Development

- functional modeling in OSD development – how to do it
  - functional analysis/architecture process
  - item flow and control flow
  - coupling, cohesion, connectivity
  - unallocatable and allocatable functions
  - pitfalls in defining functions
- ***workshop 13 - a simple functional solution***
- ***workshop 14 - physical and functional solution***
- FMEA in functional solution
- performance thread analysis
- languages incorporating behavior modeling
- software tools supporting functional and physical solution
- pitfalls in functional solution development

### 8.6 Return to Physical Solution Development in OSD Development

- facilities, procedures, people, and other types of solution element
- some common pitfalls in developing OSD
- adding the detail to the solution
- solution creates requirements – the duality of requirements and solution
- interface engineering
- evolution of interfaces in solutions having levels of physical structure
- interface requirements specifications versus interface design descriptions
- some common pitfalls in interface engineering

### 8.7 Decision-Making in OSD Development

- solution development for feasibility
- solution development for effectiveness: approach to solution optimization
  - the role of MOEs and goals
  - using a value (system effectiveness) model
  - taking account of risk relating to goals
  - taking account of risk relating to satisfaction of requirements
  - event-based uncertainty
  - risk-aversion
  - ***workshop 15 - using a value (system effectiveness) model in developing solution for an infrastructure system problem***
  - cost/capability, return on investment and like concepts
  - iterative optimization of solution – an effective methodology
  - software tools supporting OSD decision-making
- common pitfalls in OSD development

### 8.8 OSD Document

- OSD Template
- Example OSD

## 9. Development of Requirements and Requirements Specifications for Elements of Solution

## 10. Summary and Key Points

- action plan

## 11. References and Recommended Reading


**To register visit our website or call our friendly registration team:**

PO Box 2385
Ringwood North
Victoria, 3134 Australia

+61 3 9876 7345

contact@ppi-int.com

www.ppi-int.com


PROJECT PERFORMANCE INTERNATIONAL