

5-DAY COURSE

Requirements, OCD & CONOPS

in Military Capability Development

Defining military capability problems and solutions

Why Requirements, OCD & CONOPS?

Major military capability programs frequently suffer delays, cost overruns, and reduced operational capability, not because of technology, but because of ineffective problem definition and solution development processes. In many cases, requirements are defined before there is a shared and validated understanding of how a capability will be used, or worse, requirements to be met by the capability solution are not defined at all.

This course addresses that malaise by focusing in workshop format on the disciplined development of Operational Concept Descriptions (OCD) - how the capability will be used and by whom, and required characteristics of any solution for that solution to be fit for the intended use. Participants learn how to define the capability problem properly, maintaining separation of problem from solution, and avoiding systemic issues that arise when technology decisions are made too early. Problem drives solution - the course then focuses on capability solution development, a Concept of Operations (CONOPS) being a widely used representation of the operational solution. Participants will learn how to apply systems thinking to reliably and efficiently develop an optimum capability solution.

Clear problem definition is the single highest-leverage activity in capability development. While the examples and workshop system are from military capability development, the methods taught are directly transferable to any large system or enterprise involving people, processes, and technology.

What You Will Learn

Participants will learn:

- How to define a capability need
- The role of an Operational Concept Description (OCD) in doing so
- How to develop an OCD
- The distinctly different nature and purpose of a Concept of Operations (CONOPS)
- How to develop a CONOPS
- Migration to acquisition-ready capability solution documentation.

The course also addresses common misconceptions, relevant public-domain standards, and how these artifacts support decision-making and operational effectiveness.

Earn CE/CPD Credit



**PMI Talent Triangle®
Suggested PDUs**

- Ways of Working - 30
- Business Acumen - 8
- Power Skills - 2



**INCOSE Certified
Systems Engineering
Professional (CSEP)**

- 40 Continuing
Education PDUs

20,000+ Professionals Trained Across 43 Countries



PPI-008949-3-US

© Copyright and all other rights reserved Project Performance International 1992–2026.
All trademarks, logos and brand names are the property of their respective owners.

ppi-int.com

Who Should Attend and Why

This course is intended for professionals involved in capability development, including military capability developers, engineers, systems engineering practitioners, requirements managers, and project or program managers. It will be particularly valuable for those shaping capability development policy or practice.

If you influence what capability is developed, why it is developed, or how success is measured, this course is for you.

Although oriented toward military capability, the approaches taught apply equally to complex business systems, infrastructure programs, and large enterprise initiatives.

Training Methods and Materials

The course is delivered through a structured mix of instructor-led explanation, examples, and substantial workshop activity. A single evolving capability scenario is used throughout the course to illustrate concepts and work products, reinforcing how OCDs, capability system requirements specifications, and CONOPS relate in practice.

Participants receive a comprehensive training manual, a workbook with workshops simulating development of a substantial military capability, and a reference set of example capability and materiel-level documents, including OCDs, CONOPS, requirements specifications, and statements of work.

The emphasis is on practical understanding that can be applied immediately in real programs.

Participants also receive complimentary access to PPI's Systems Engineering Goldmine and Systems Engineering Tools Database for continued learning and reference.

PPI Training Reviews



"I recommend this course offered by PPI to improve knowledge for professionals working in the Defense field. Instructors are considered references in this area."

**Course participant,
Brazil**



"Understanding OCDs and analyzing customer needs made a huge difference to my work performance."

**Course participant,
South Africa**



"The best thing about the course was the high quality of the course and the caliber of the presenter."

**Course participant,
Brazil**

Why PPI?

Trusted Worldwide

PPI delivers outstanding training and consulting to many hundreds of enterprises worldwide, from Fortune 100 companies (presently 19% of them) to small startups. PPI is a truly international company, with personnel based in eight countries, and clients across six continents benefiting from our work.

PPI Presenters

PPI's presenters are internationally recognized systems engineering practitioners and consultants who bring decades of real-world experience, ensuring every concept taught is value-adding, practical, relevant and immediately implementable.



Requirements, OCD & CONOPS in Military Capability Development 5-Day Course – Course Outline

1. Concepts and Definitions

- Defining the problem, and developing a complete solution
- Life cycle basis of problem definition and solution
- Example requirements and Measure of Effectiveness (MOE) relating to a capability
- Example physical levels of solution definition relating to a capability
- Definition: problem
- 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
- Concepts related to problem description
 - Definition: need
 - Definition: want
 - Definition: desire
 - Definition: intent
 - Definition: expectation
 - Definition: constraint
- Definition: Operational Concept Description (OCD)/ Concept of Use (CONUSE)
- Definition: solution
- Definition: solution description
- Definition: architecture
- Definition: architectural design description
- Definition: architectural framework
- Definition: Concept of Operation(s) (CONOPS)/ Operational Solution Description (OSD)
- Definition: detailed design
- Definition: stakeholder
- Definition: verification
- Definition: verification requirement
- Definition: verification requirements specification
- Definition: validation

2. General Concepts of Problem-Solving

- Problem definition
- Emergence
- Divergence
- Convergence
- Systems thinking

3. Relationships Between Requirements Document, OCD, Architectural Design Description, CONOPS

- A system solution
- Systems of systems
- Types of reviews

4. Types of Requirements

- Eight basic types
- **Workshop 1 – categorizing requirements for a capability by type**

5. The Quality of Requirements

- Requirements quality attributes

6. Requirements Analysis for the Capability: OCD and SyRS Content

- Purpose of requirements analysis and its relationship to OCD
- Relationship to CONOPS
- Requirements analysis (capture and validation) methodology
- Context analysis, and relationship to OCD
- **Workshop 2 – context analysis for a capability system**
- States and modes analysis, and relationship to OCD
- **Workshop 3 – states and modes for a capability system**
- Parsing analysis, and relationship to OCD
- **Workshop 4 – parsing analysis**
- Functional analysis, and relationship to OCD
- **Workshop 5 – functional analysis for a capability system**
- Rest of scenario analysis, and relationship to OCD
- Entity Relationship Attribute (ERA) analysis, and relationship to OCD
- Out-of-range analysis, and relationship to OCD
- Value analysis, and relationship to OCD
- **Workshop 6 – building a capability system value (system effectiveness) model**
- Operational effectiveness
- Operational effectiveness versus overall effectiveness
- Extracting information for an OCD and requirements document from users (and others)
- Example applications of AI to requirements analysis

7. Operational Concept Description (OCD) as a Document

7.1 Content and Purpose of an OCD

- Users and uses of an OCD, in detail
- Types of OCD
- Principles regarding content
- Use cases, mission profiles, scenarios and the OCD
- How does support relate to an OCD?
- Relationship of the OCD to the requirements document
- OCD standards and guides
 - Operational Concept Documents, DID DI-



Requirements, OCD & CONOPS in Military Capability Development 5-Day Course – Course Outline (Continued)

MCCR-80023, Software Design Description (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
- American National Standard Institute (ANSI)/American Institute of Aeronautics and Astronautics (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 Military Standard (MIL-STD) 498, U.S.A. Department of Defense, 1994
- Institution of Electrical and Electronic Engineers (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/International Council on Systems Engineering (INCOSE)/AIAA, G-043:2012
- Project Performance International's (PPI's) OCD/ CONUSE DID
- Who should prepare an OCD
- Timing of preparation of an OCD versus requirements document

7.2 Preparing an OCD

- Characteristics of a good OCD
- **Workshop 7 – review of sample OCDs**
- Pitfalls in OCD preparation
- **Workshop 8 – review of an OCD for a military aircraft system**
- **Workshop 9 – preparing a basic OCD**
- Use of graphics in OCDs
- Level of detail in the OCD
- Design content – when, and when not?
- Makeup of an OCD development team
- The role of users
- Beyond the basic OCD
- Extending OCDs to other stakeholders
- Pitfalls in preparing OCDs

8. CONOPS/Operational Solution Description

8.1 Content and Purpose of a CONOPS

8.2 Relationship of CONOPS to Overall Solution

8.3 Solution Development Strategies

- The solution domain: key concepts, relationships, and work products
- **Workshop 10 – principles of CONOPS development**
- Waterfall, incremental, evolutionary and spiral capability development approaches
- **Workshop 11 – capability system development strategies**

8.4 Concepts of Architecture – Physical and Logical, in CONOPS 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 (CIs)
- Relationship of CI definition to future system integration
- **Workshop 12 – physical conceptualization of capability system solution**
- Example applications of AI to initial physical conceptualization
- 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 CONOPS Development

- Functional modeling in CONOPS 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**
- Failure Modes, Effects and Criticality Analysis (FMECA) in functional solution
- Performance thread analysis
- Example applications of AI to logical design
- Systems modeling language (SysML), and alternative languages incorporating behavior modeling



Requirements, OCD & CONOPS in Military Capability Development 5-Day Course – Course Outline (Continued)

- Other functional modeling languages
- Software tools supporting functional and physical solution
- Pitfalls in functional solution development

8.6 Return to Physical Solution Development in CONOPS Development

- Facilities, procedures, people, and other types of solution element
- Some common pitfalls in developing CONOPS
- 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 structure
- Interface requirements specifications versus interface design descriptions
- Further example applications of AI to physical design
- Some common pitfalls in interface engineering

8.7 Decision-Making in CONOPS 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 a capability problem**
- Cost/capability, return on investment and like concepts
- Iterative optimization of solution – an effective methodology
- Software tools supporting CONOPS decision-making
- Example applications of AI to trade studies
- Common pitfalls in CONOPS development

8.8 CONOPS Document

- CONOPS Template
- Example CONOPS

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

10. Summary and Key Points

- Action plan

11. References and Recommended Reading



www.ppi-int.com

*systems/product engineering training & consulting
for project success ...*

