

Training Program

**SOFTWARE REUSE &
INTEGRATION PROFESSIONAL**



Introduction

Overview

This 4 days Software Reuse courses is specifically is designed to give participants the hands on experience in developing and reusing software.

Duration

4 – days workshop.

Program Objectives

Learning Objectives

By the end of the course participants should:

- ❖ Understand how developing software for reuse impacts application developers
- ❖ Understand how developing software for reuse is different than developing software in general
- ❖ Understand how programming language features facilitate or hinders software reuse.
- ❖ Understand the role that a software architecture plays in developing software for reuse.
- ❖ Gain hands-on experience in developing software for reuse, or reusing software

Target Audience

This course is intended for those as below:

- ❖ Software Engineer
- ❖ Designer
- ❖ Developer
- ❖ Software Architect
- ❖ Technical Lead
- ❖ Project Manager
- ❖ Product Line Manager
- ❖ Software Engineering Manager

Training Contents

Module 1: Introduction to Software Reuse

- ❖ What reuse is and what it is not?
- ❖ The benefits and limitations for asset reuse.
- ❖ What artifacts can be reused?
- ❖ The difference between application and domain engineering reuse
- ❖ The tradeoffs between white-box and black-box reuse
- ❖ The critical aspects of a software reuse program.
- ❖ The difference in State of the Art and State of the Practice in Reuse Management, Reuse Techniques, Initiatives
- ❖ The Organizational, Technical, and Economic aspects of Software Reuse

Module 2: Software Reuse Program

- ❖ The infrastructure for a systematic software reuse program.
- ❖ The support services differ for software reuse
- ❖ The barriers to establishing a reuse program
- ❖ Implementing a reuse program with small steps
- ❖ What makes a software asset reusable?
- ❖ Acquiring a reusable asset
- ❖ The challenge of finding the right separation of components
- ❖ The issues that affect the domain engineering life cycle

Module 3: Domain Analysis

- ❖ What domain engineering is?
- ❖ The activities involved in finding reusable components
- ❖ The issues that a domain analysis method must address
- ❖ What a domain is, and what abstraction is with respect to domain analysis?
- ❖ Some of the heuristics used to determine the fixed and variable parts of components

Module 4: Reuse Code

- ❖ Three programming paradigms: Procedural or Functional, Logic, Object Oriented
- ❖ Mechanisms of Object Oriented programming that enhance reusability
- ❖ Overloading and generosity contributes to reusable code
- ❖ Subtyping and Sub-classing facilitate software reuse

Module 5: Reusable Architectural Frameworks.

- ❖ Architectures are objects of reuse
- ❖ Architectures lend to reuse
- ❖ Develop reusable architectural frameworks
- ❖ Java RMI provides the infrastructure to develop reuse frameworks

Module 6: Component Storage and Retrieval

- ❖ Introduction to software libraries
- ❖ Software libraries play for successful deployment of software reuse
- ❖ Characteristics of software libraries
- ❖ Issues that arise in the design and deployment of software libraries
- ❖ Challenges of integrating components

Module 8: Component-Based Software Engineering (CBSE)

- ❖ What is Component-Based Software Development?
- ❖ What does component mean in the context of CBSD?
- ❖ The “component model” in CBSD
- ❖ The technical and business issues associated with CBSD

Module 7: Software Reuse Metrics

- ❖ Basic software engineering metrics
- ❖ The three parts of component engineering metrics: Concept, Content, Context
- ❖ Application reuse metrics
- ❖ Domain engineering metrics
- ❖ Organizational metrics
- ❖ Various major reuse metric and economic models
- ❖ Limitations of these models
- ❖ Critical ideas behind a metrics program

Module 9: Product Line Engineering

- ❖ What is the Product Line Engineering (PLE)?
- ❖ Product Line Engineering fits within software reuse
- ❖ PLE vs CBSD.
- ❖ SYNTHESIS, a Product Line Engineering methodology

Module 10: COTS Based Development

- ❖ What is COTS Based Development?
- ❖ COTS Based Development vs CBSD
- ❖ Limitations and challenges to COTS Based Development
- ❖ The economic incentives to COTS Based Development
- ❖ Verification and Validation in COTS Based Development

Key Features

- 32 hours of high quality learning
- Quizzes in the end of each chapter
- 4 Real life exercises and case study
- Learn from experience and passionate trainers
- Quick reference card, toolkits
- Course completion exams
- Apply the knowledge and technique to handle the role
- Invited to our professional community
- Certificate of Completion

