

## Software Design for Reliability Training Workshop Outline

### Introductions

- ✓ Workshop Agenda Review
- ✓ Very Little about the Instructor
- ✓ Introductions and participants expected outcomes

### Software Reliability Basic Concepts

- ✓ What is Software Reliability; The Quality Elephant?
  - Software Quality vs. Reliability vs. Safety
  - Traditional Definition Software Reliability
- ✓ Basic Terminology:
  - Defects, Faults and Failures
  - Failure Rates, Failure Distributions and MTTF
  - Software Availability and Failure Rates
  - Input Robustness
  - Software Fault Tolerance
- ✓ **Round-Robin Discussion**

### Regulatory Requirements & ‘Guidance’ for Software Intensive Products

- ✓ Regulatory Standards for Quality Management Systems
- ✓ Industry specific Regulatory Standards/Guidance
  - Mil Std’s and IEE, IEC guidance
  - Safety Critical Systems (Aviation, Transportation)

### Group Discussion

*How do Standards and Guidance Documents affect your product development methodology?*

- ✓ Effects your development processes
- ✓ Affects your quality & reliability measurement System?
- ✓ Impacts (including constraints) on your software development methodology

### Reliability Measurements, Metrics & Data Analysis

- ✓ Reliability Measurements and Metrics to be Tracked
- ✓ Defect Removal Efficiency
- ✓ Project Critical Defect Tracking and Use
  - Failure Rate Data
  - Data Analysis
  - Failure Density Analysis
  - Failure Causal Analysis

### Emergent Software Tools

- ✓ New generation of Integrated Development Environments
- ✓ Next generation Static Code Analyzers
- ✓ Automated “unit test generator development engines
- ✓ Rapid adaptation of ‘Agile’ development methodologies
- ✓ Model driven design tools
- ✓ **Round-Robin Discussion**

## “Best Practices” Approach to Developing Reliable Software

- ✓ Typical Paths to followed to software reliability
  - Formal Methods
  - H/W-Based Approach
  - Best Practices Approach
- ✓ Best Practice Approach to SW Reliability
- ✓ Best Practices for “Best in Class” Companies
- ✓ Identified Weakness within Development Organizations
- ✓ The Defect Prevention – Detection Strategies
- ✓ Integrating Reliability Practices into your PLC
- ✓ Summary: DfR Based on a “Best Practices” Approach
- ✓ **Round-Robin Discussion**

## Software Root Cause Analysis (RCA) as a process learning tool /w discussion

- ✓ Using ODC as a software defect classification scheme
- ✓ Software defect analysis and root cause analysis
- ✓ Frequent lessons learned
- ✓ **Focused Discussion:** Your current analysis practices of Defect Data

## Software Reliability Modeling

*Purpose: Set overview and context for 2<sup>nd</sup>. half of week*

- ✓ Overview of SW Reliability Predictive Modeling tools
  - Examples of tools, models and use
- ✓ SW Reliability Estimation modeling and tools
  - Examples of tools, and models and use

## Developing a Software Reliability (Quality) Plan

- ✓ SW Reliability Planning: Current State of the Industry
- ✓ A FURPS Framework for organizing Quality-Reliability Requirements
  - **F**unctionality and **U**sability
  - **R**eliability (Safety) and **P**erformance
  - **S**upportability (and **S**erviceability)
- ✓ Implications of Regulated QMS System Requirements
- ✓ **Round-Robin: XX’s current best practices in DfR planning**

## Requirements–Design Phase Best Practices

- ✓ Role of System Architecture and Design
- ✓ Software Design Best Phase Practices
- ✓ Defects of “commission” and “omission” in Design
- ✓ Using the FMEA and design review process as learning tools
  - FMEA as a “bottoms-up” design analysis tool
  - FMEA’s in ‘Safety’ regulated businesses
- ✓ Integrating formal design processes into ‘agile’ methods
- ✓ **Round-Robin: Current State formal SW Design process in your org.**

## **Coding Phase Best Practices**

- ✓ Coding Phase Best Practices
  - Key best practices for reliable code
- ✓ Integrating “new generation static code analysis” into the coding process
  - Unit testing as a verification tool
- ✓ Code Reviews: principles and practices
  - ✓ **Round-Robin: Current State Coding practices in your org.**

## **Importance of a Testing Strategy**

- ✓ “Types of testing” and testing strategy development
- ✓ Deploying the test strategy in ‘agile’ development methodologies
- ✓ Unit Testing Strategies (who does what and when?)
- ✓ System/ Reliability Testing

## **Breakout Group – Test Strategies in ‘agile’ develop @ your company**

*Final topic and structuring to TBD*

## **Unit Test Phase Practices**

- ✓ Software Robustness Failure Modes
- ✓ Input Equivalence Classes (ECs)
  - Guidelines for Defining ECs
  - EC Example
  - Augmenting ECs: Boundary Value Analysis
- ✓ Coverage-Based Unit Testing
- ✓ Code Coverage Standards
- ✓ **Round-Robin Discussion**

## **System Test Phase Best Practices**

*Setting stage for remainder of week*

- ✓ Key types of testing during system testing
- ✓ Operational / Usage Profiles
- ✓ Fitting in verification and validation testing
- ✓ Generating Software Reliability Estimates
- ✓ **Round-Robin Discussion**