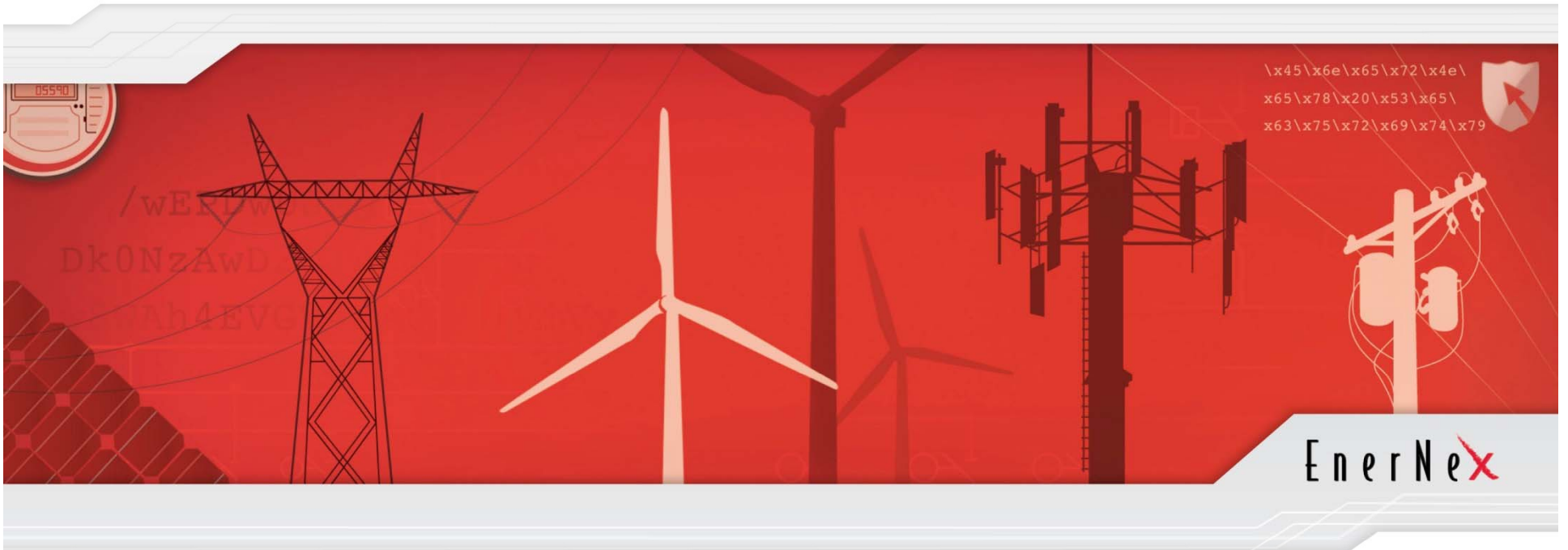


# UCAIug Testing Subcommittee Boot Camp





# Agenda

- ▶ Introduction to 3 UCAlug groups
- ▶ Testing Overview
- ▶ Testing Details
- ▶ Development of a Testing Program



# Introduction to UCAlug groups

- ▶ UCAlug – how did we get here?
  - MMS Forum - 1993
  - AEP Substation Initiative - 1996
  - UCA2 Users Group - 1999
  - UCA International Users Group – 2002
  
- ▶ Why did we get here
  - Share experiences
  - Promote adoption of underlying standards



# IEC 61850 Users Group

- ▶ This was the start of UCAIug
- ▶ Started as protocol with a substation
- ▶ Is now far more than a protocol, a way of life
- ▶ Based upon MMS, ISO, TCP/IP
- ▶ Defines conformance suites for various product
- ▶ Defines performance test for “GOOSE”
- ▶ Created QAP
- ▶ Has Certificate program without logo



# CIM Users Group

- ▶ Second group to join UCAlug
- ▶ Large user group
- ▶ Started as object database standard
- ▶ Now deals with object transport
- ▶ Based upon XML, TCP/IP
- ▶ Defines interoperability suites for various use cases
- ▶ Performs pre-standards tests (verifies standards BEFORE they are issued)
- ▶ Has no formal Certificate program



# Open Smart Grid Users Group

- ▶ This is most recent entry to UCAIug
- ▶ Has diverse user base
- ▶ Device and enterprise components
  - Device : bits and bytes, RF signal strength, etc.
  - Enterprise : XML like CIM
- ▶ Based upon ZigBee (SEP), XML, TCP/IP
- ▶ Will define conformance and IOP suites
- ▶ Has no programs complete at this time



# Agenda

- ▶ Introduction to 3 UCAlug groups
- ▶ **Testing Overview**
- ▶ Testing Details
- ▶ Development of a Testing Program



# Testing Overview - Goals

- ▶ Why do we test?
  - Expose latent defects in system
  - Users expect error-free systems in their application
  - Vendors want independent validation
  - Sometimes: “Check the box”



# Testing Overview - Outcomes

- ▶ What do we expect from testing
  - Error-free report: no problems found (bad?)
  - Report showing areas needing improvement
  - Report showing failure of test (rarely issued)



# Testing Overview – Product types

- ▶ Can test products or services
  - Product: Identified unit with specific input/output relationship, typically at bits and bytes level
  - Service: Software with less well-defined inputs and outputs, generally holding state information
- ▶ Why testing differs
  - Product: stimulate UUT, verify response
  - Service: send context, verify response semantics



# Testing Overview – Test Cases

- ▶ What are test case sources?
  - Implicit standards: “... shall ...”
  - Explicit standards: tests specified by SDO
  - Industry Use Cases: “... Monitors system and issues ...”
  - User use cases: Specific ways user operates system
  - Vendor use cases: Regression testing



# Testing Overview – Test Steps

- ▶ Test procedures document: What will tester do?
  - Requirement for specific test step
  - Preconditions
  - Stimulus
  - Expected response (or range of responses)
  - Verdict definition



# Agenda

- ▶ Introduction to 3 UCAlug groups
- ▶ Testing Overview
- ▶ **Testing Details**
- ▶ Development of a Testing Program



# Testing Details – Types of Tests

- ▶ Conformance – does device/system match spec
- ▶ Pair-wise IOP – does it work with a specific partner?
- ▶ Standards-vetting IOP – does the proposed spec mandate unique solutions?
- ▶ End-to-End IOP – does entire system work?
- ▶ Ad-hoc testing – for specific situations, does the system correctly perform its function?



# Testing Details – Test Reports

- ▶ What was tested
- ▶ How tested (environment, test equipment, etc.)
- ▶ Which tests executed
- ▶ Which tests not executed (and why)
- ▶ Tester comments
- ▶ Overall verdict: pass or fail or inconclusive



# Testing Details – Test Execution

- ▶ Level of test specification
  - Specify test procedure
    - Allows innovation in test scripting
    - Possible differences between testers
  - Define standardized test scripts
    - Requires specification of one test tool
    - Public tool requires much volunteer time to vet
- ▶ Control of tests – ideally voted by User Group
  - Reality – most users don't understand tests



# Testing Details – User Group/SDO

- ▶ SDOs may have national/political agendas
- ▶ Inherent struggle between groups
  - SDO wants stability – stable standards
  - UG wants flexibility – faster evolution
  - Both want appearance of stability for adoption
- ▶ Implication upon testing
  - Test groups must be flexible when interpreting a standard
  - Sometimes deliberately violate written spec



# Agenda

- ▶ Introduction to 3 UCAIug groups
- ▶ Testing Overview
- ▶ Testing Details
- ▶ **Development of a Testing Program**

# Test Program Development

- ▶ Question before starting
  - What programs already exist/known to users?
  - What is funding model?
    - Test development
    - Test program maintenance
    - Test program administration
    - Logo / formal certificates?
  - What can vendors afford? Testers?
  - What problems are users seeing/testing needs

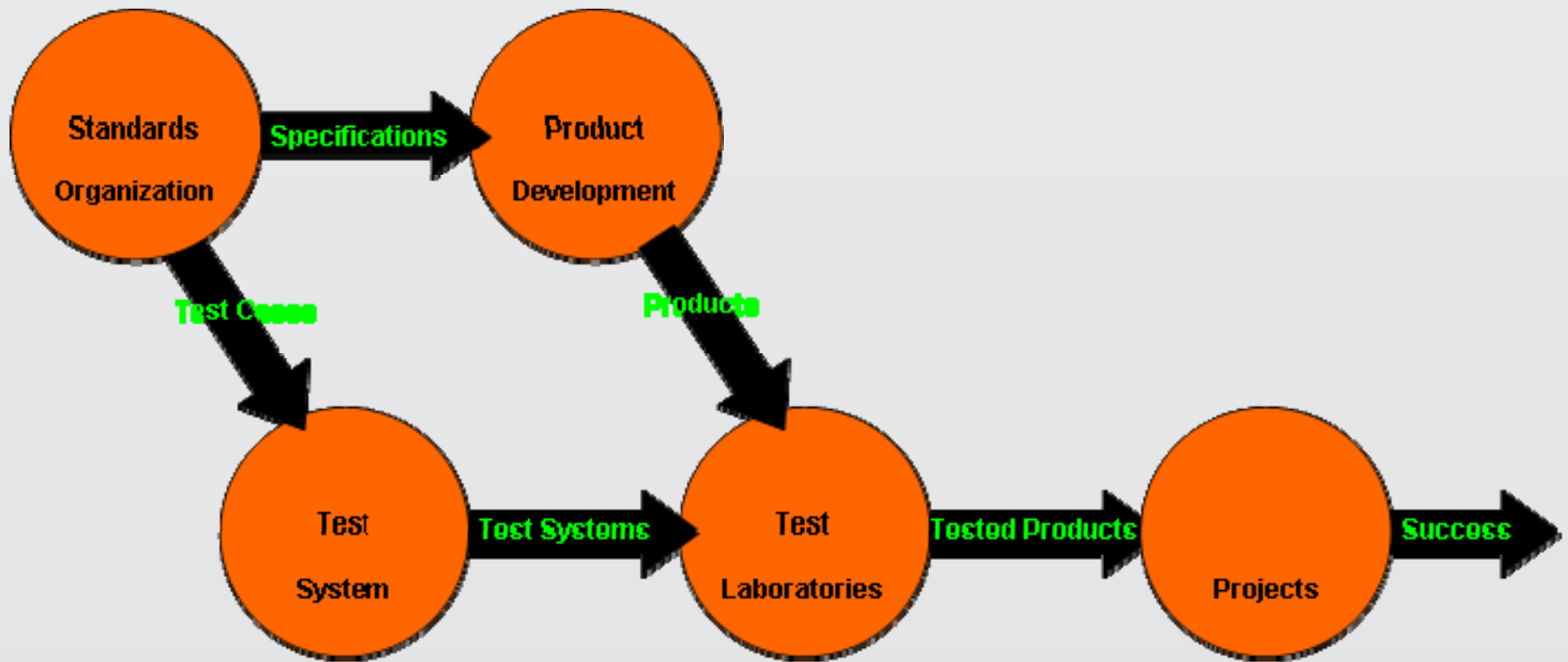


# Quality Assurance Plan

- ▶ Need this as first step
  - Define how testing process functions
  - Defines defect feedback mechanisms
  - Defines quality systems needed by testers
  - Defines funding model for the testing process
  - Defines conformance vs. IOP testing or both
  - Defines whether UG will contract work
- ▶ This is the “business plan” for the testing group
- ▶ Get support from user group of this QAP
- ▶ Iterate until users agree to plan

# UCAIug QAP Process

\x45\x6e\x65\x72\x4e\  
\x65\x78\x20\x53\x65\  
\x63\x75\x72\x69\x74\x79





# Forming Testing Group

- ▶ Seek volunteers
- ▶ Set realistic timetable, don't overwork volunteers
- ▶ Create real project plan, too easy to fall behind
- ▶ Avoid temptation of scope creep
  - Finish planned work unless completion makes no sense
- ▶ Continue to seek volunteers until 1/3 complete
- ▶ Add “unplanned” work at end



# Test Plan reviews

- ▶ Have user group periodically review test plan
  - Don't want a surprise after 6 months work
  - Be willing to change to meet user needs
  - Avoid temptation to satisfy minority of users
  - Ask question: "Is this what our industry needs?"
- ▶ Upon completion of test plan, real work begins

# Test Procedure Development

- ▶ Begin with highest-level of suite
  - Example: Connects, authenticates, basic data
- ▶ Break each of these into logical functions
  - Example: Joins network, creates peer matrix
- ▶ Break each function into individual tests
  - Example: Join network using only default credentials
  - Create both positive and negative tests
- ▶ Break tests into test steps with stimulus/response



# Practical Example – 61850

- ▶ Decided on lightweight process at start
- ▶ QAP:
  - Write test procedures
  - Accredit/monitor testers
  - Arbitrate disputes
  - Post tester-created certificates
- ▶ Results: industry acceptance, low-cost to users



# Practical Example – CIMug

- ▶ Decided on lightweight process at start
- ▶ QAP (informal):
  - Only perform IOP testing
  - IOP tests specific to each IOP
- ▶ Results:
  - IOPs have found errors in pre-IS IEC documents
  - CIM products are interoperable



# Testing Outcome

- ▶ All stakeholders agree on testing process
- ▶ Testing is meaningful to the industry
- ▶ Testing is cost-effective
  - Users view tested products as good “first step”
  - Vendors view testing as more than “check the box”



# Questions

\x45\x6e\x65\x72\x4e\  
x65\x78\x20\x53\x65\  
x63\x75\x72\x69\x74\x79

