AREVA

AREVA

Simulation Software SIVAT

SImulation based VAlidation Tool

Dr. Steffen Richter

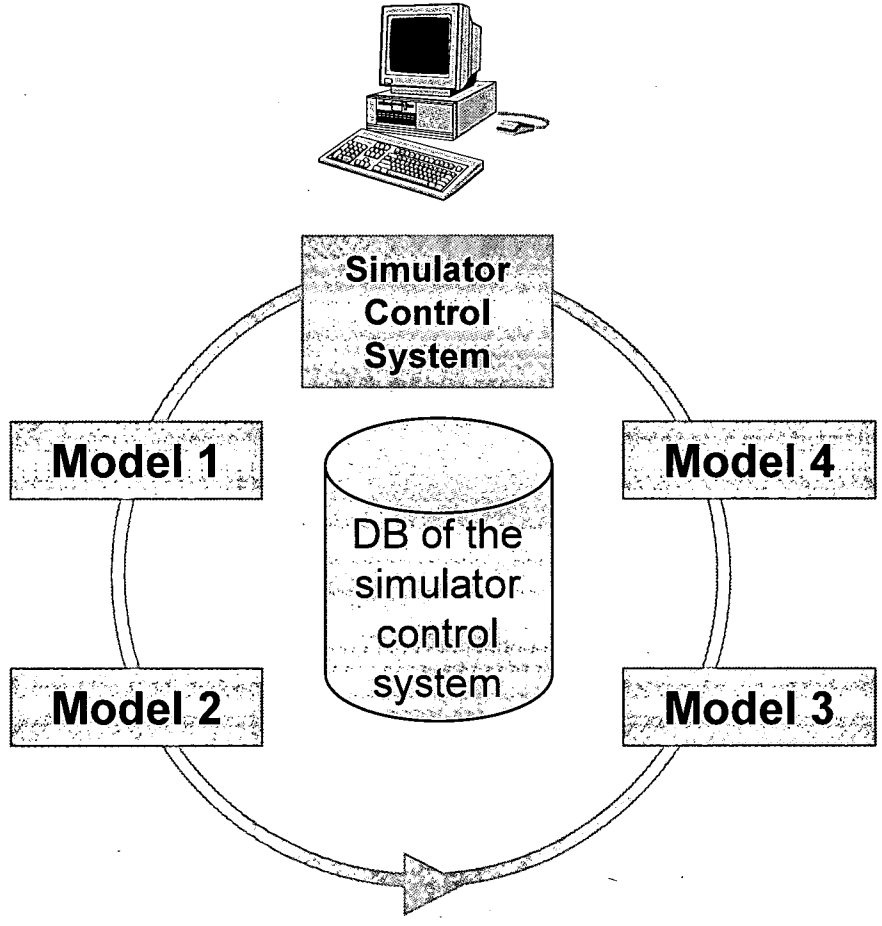
Basics of Simulation

Requirements:

- Models of the system/process
- Simulator control system

Main features:

- visibility of variables
- restart ability



SIVAT in the Engineering Process

Purpose of the Simulation using SIVAT

> *Validation of the application software functionality of a specified TXS I&C system*

> *Verification of the specified I&C system as against the functional requirements*

> *Reduce the effort for error correction in the test field*

> *Features:*

- *Automatic generation of the simulator program*
- *Wide variety of manipulation functions (i.e. built-in malfunctions)*

2 Tasks of SIVAT

Basic requirements for the TXS simulator

Basic requirements for the TXS simulator (cont.)

Software Validation by SIVAT Capabilities

Application of SIVAT
Project references (from 1998 through 2005)

Plant/nuclear power plant	TXS system

Components of SIVAT Tool

Generation of the TXS simulator
Tasks of CATS-SDE

Principle of the TXS simulation

Components of the TXS simulator

Components of the TXS simulator
Special models

Handling of Input/Output Signals Models TXS-INP and TXS-OUT

Creation of the TXS models

Organisation of data in the simulator DB

Principle of communication in SIVAT

Using the Simulator

Using test scripts for simulation

Test script - example

Test script - result

Monitoring simulation process using the dynamic FDE

Scope of application of SIVAT

- *Starting simulation as early as possible in the course of engineering*
- *Increase efficiency of the V&V activities during project implementation of TXS I&C systems*
- *Evaluation of the effects of planned changes in installed systems*
- *Reconstruction of events in installed systems*

TXS I&C System Validation in the Test Field Application of ERBUS TXS

Handling of SIVAT Examples

- > *Select scope of simulation*
- > *Define execution sequence of models*
- > *Monitor and set signals*
- > *Simulate malfunctions*

Manipulation functions - Select scope of simulation

Manipulation functions - sequence of the models

Manipulation functions - Monitoring and setting of signals

Manipulation functions - Malfunctions example: one faulty I/O board

Manipulation functions - Malfunctions example: two faulty I/O boards

Manipulation functions - Malfunctions example: three faulty I/O boards

Life Cycle of SIVAT Tool

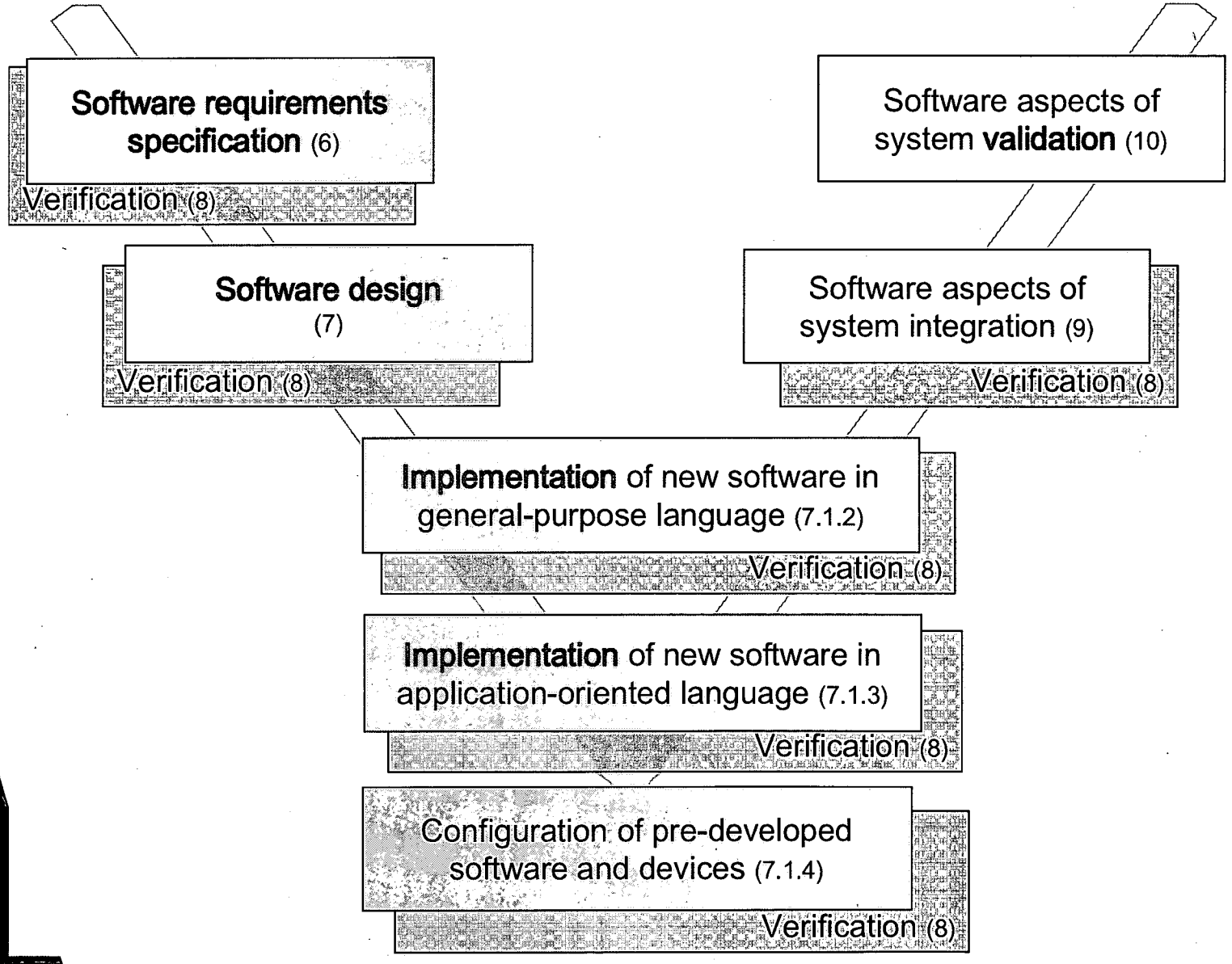
AREVA

TELEPERM XS
System Software Development for Safety Applications

Development process
Configuration management
Change control

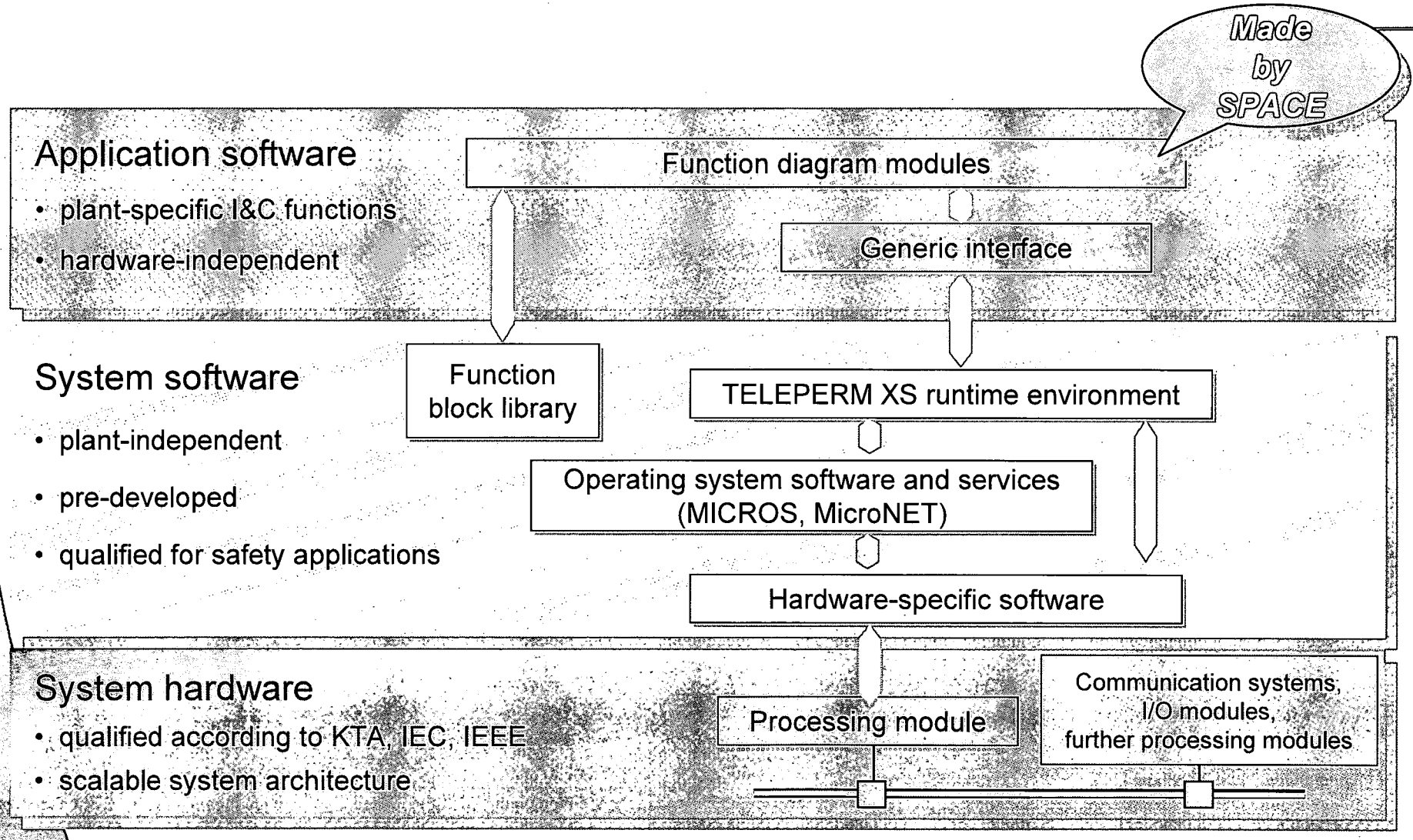
Dr. Steffen Richter

V-Model according to IEC 60880 (2006)



TELEPERM XS Software Engineering Procedures
Development Phases

TELEPERM XS System Platform Architecture
Layered Software Structure on a Processing Module



TELEPERM XS Configuration Management

Configuration identification

Configuration Management
Process requirements according to ISO and IEEE

Configuration identification

- *Well-defined structure of configuration items (CI)*
- *Unique identifier for each configuration item*
- *Version control for each configuration item*
- *Application of a software configuration management tool (SCM tool)*

Configuration control

- *Formalized modification process*
- *Compatibility considerations*
- *Expert user board (EUB) as a decision-making body, CCB (supervisory board)*
- *Sound quality assurance, release procedure*

TELEPERM XS Configuration Management
Version control

TELEPERM XS Software Configuration Management
Change Process

Overview of the SIVAT Development (and Maintenance)

Andreas Künzel

NLTD-G / AREVA NP GmbH

Stages of the SIVAT Development

- *Initial Development of SIVAT*
  - *According to the phase plan described in FAW TXS-1.1*
- *Validation of SIVAT*
  - *Concept: Test against data recorded in the test fields*
- *Maintenance of SIVAT*
  - *According to the process described in FAW TXS-1.5*

Documentation of Initial Development

> ***Frame Requirement Specification***
Rahmenlastenheft TXS-Simulator
KWU NLL4/98/042

> ***Requirement Specification***
Lastenheft SIMM
KWU NLL4/98/068

> ***Functional Specification***
TXS Pflichtenheft, Version 01.21: Generator CATS-SDE für die TXS-Simulationsumgebung
KWU NLLZ ST/99/023b

Validation of Initial SIVAT Release

> *Concept: Tests against data from the test bays*

> *Test Documentation*

- ***Test Results (NPPs Unterweser and Emsland):***
  Vergleich der Ergebnisse des Prüffeldes (n-cpu) und der Offline (1-cpu) Simulationsumgebung
  KWU NLL4/1998/180a

- ***Test Results (NPP Philippsburg)***
  Auswertung der SIVAT Tests der LT-Funktionen
  KWU NLL4/2000/032

Maintenance of SIVAT 1/2

Maintenance of SIVAT 2/2

- *Use in the Project Teams*
  Comparison of results from test fields and SIVAT runs
- *Creation of Change Requests in case of*
  - *adjustments to environment (e.g. operating system, database, Core-SW),*
  - *required new features or*
  - *differences in results from simulation and test bay.*
- *Analysis and Implementation of these Change Requests (including Reviews)*
- *CR Implementation Tests and Integration Tests*
- *Functional Regression Tests*
- *→ Release*

Described in Report:
TELEPERM XS Simulation – Concept for testing the functional behavior of the validation tool SIVAT
NGLTD/2006/en/0015A

Quality Assurance 1/3

> *Documentation of Changes, for example in file headers:*

```
// Date          Author          Change
// ==========    ============    ===============
```

Quality Assurance 2/3

> *All Source codes in Software Control Management System*

➔ *All changes are tracable*

Quality Assurance 3/3

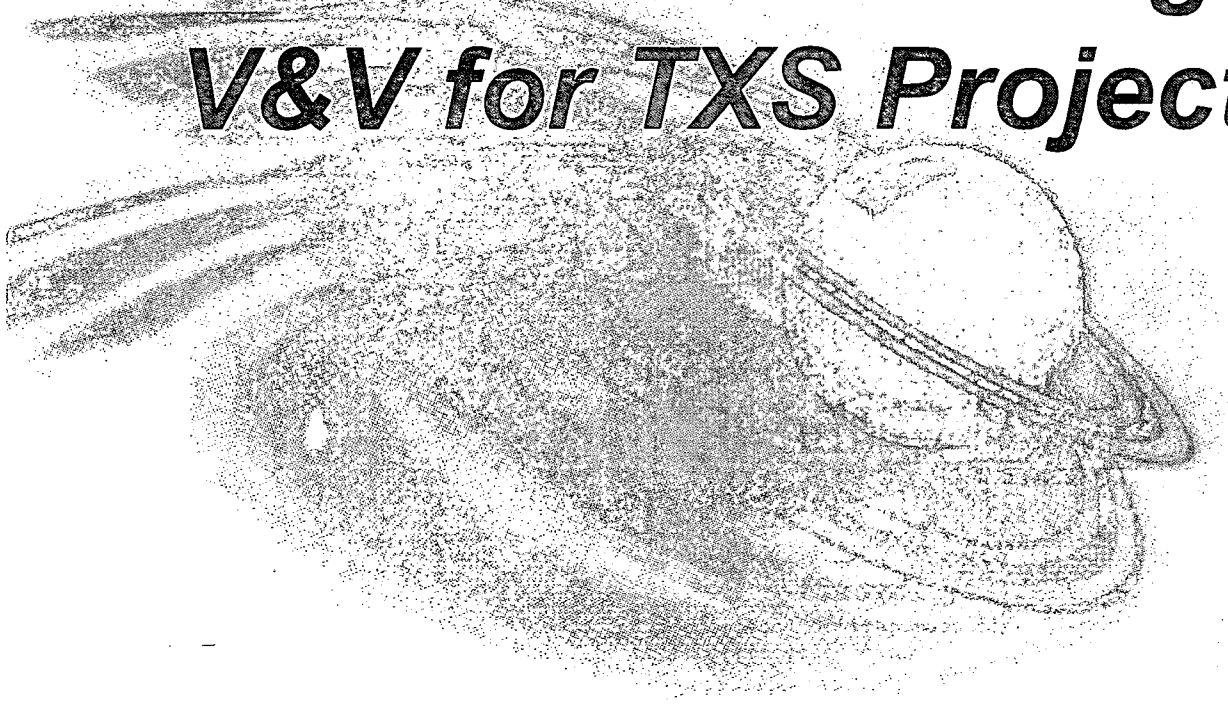
> *Reviews for*
  - *Analysis of Change Requests,*
  - *Implementations,*
  - *Documentation (e.g., test documentation, user manuals, …)*

  are done by an appropriate developer

> *Reviews for*
  - *Tests*

  are done by an appropriate tester.

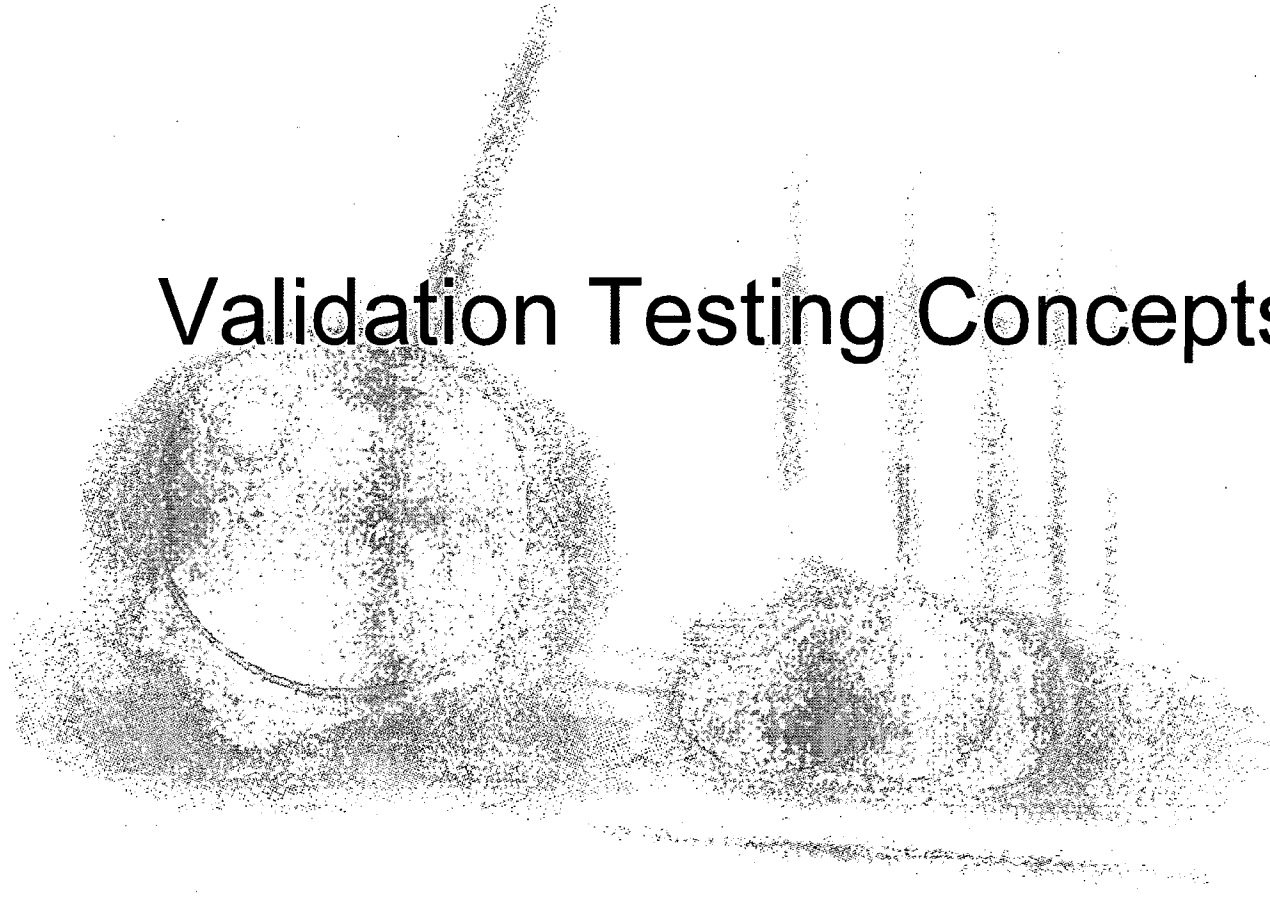
Discussion of Testing and V&V for TXS Projects



Agenda

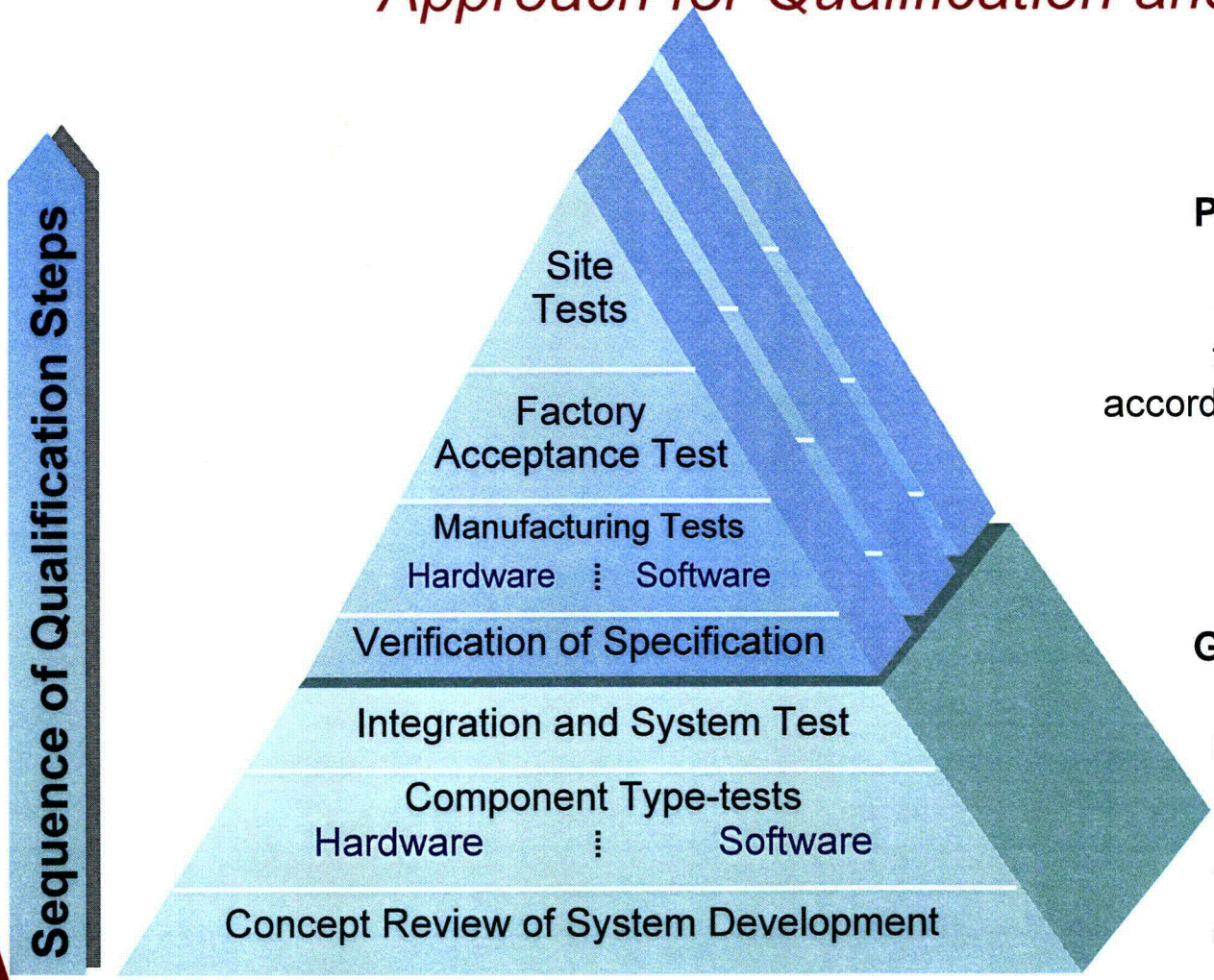
- SIVAT Development — Dr. Richter
  - Development Process
  - Development Documents
  - Validation Process and Results
  - Third Party Evaluations
  - Configuration Management
- SIVAT Tool — Dr. Richter
  - Tool Simulation
  - Tool Capabilities
  - Tool Limitations
  - Tool Demonstration
- Validation Testing Concepts — Mark Burzynski
  - Relationship of Testing Layers
  - Alignment with IEEE Std 1012-1998
  - Validation Testing with SIVAT
  - Validation Testing in Test Field
  - Examples to Explain SIVAT and FAT Coverage
- Project-Specific Verification and Validation Program — Dr. Yang
  - Requirements Phase Overview
  - Design Phase
  - Implementation Phase
  - Testing Phase

Validation Testing Concepts



TELEPERM XS

Approach for Qualification and Licensing



Project-Specific Qualification of Application to be performed according to V&V plan for each project

Generic System Qualification performed once and maintained for the TELEPERM XS system platform

TELEPERM XS

Approach for Qualification and Licensing

- Type Test of Components
  - Software type tests were designed based on the principles detailed in KTA 3503
  - Detailed requirements for the software tests are included in IEC 880
  - Approach is valid for software components whose functions and interface are clearly specified and which were developed according to a phase model
  - Characteristics of these components can be regarded as system invariants; their specification and function were tested independently of the application, there is no need to repeat the tests for each application

TELEPERM XS
Approach for Qualification and Licensing

- Plant-Independent System Test
  - Plant-independent system test performed on a representative system architecture to demonstrate key safety features of TELEPERM XS
  - First group of test objectives included correct interplay of the individual qualified software components and typical features of safety guidance system, such as failure detection and masking or directed failure behavior
  - Relevant system characteristics of TELEPERM XS were shown in a second group of test objectives, including:
    - Correct execution of the application function
    - Repercussion-free application functions and independence of system behavior from the process engineering process
    - Deterministic system behavior to validate that task processing time and communications are completely determined by project planning and are not affected by an event (i.e., system loads and the runtime behavior are predictable)
    - Functional behavior is not unduly affected by test, maintenance and diagnostics
    - Identification and reporting of failures of individual components via selected instances
    - Correct control of cabinet signaling devices
    - Limiting the failure effects on specified areas
  - Test reports and certificates issued by third party to document test passed

TELEPERM XS
Approach for Qualification and Licensing

- Delimitation between Plant-Independent and Project-Specific Tests
  - Plant-specific system test include configuration tests, tests of the interfaces to field signals and other I&C systems, failure behavior and establishment of the performance characteristics data
    - Hardware and software configurations and correct settings of plug-in jumpers, DIP switches etc. must be checked
    - Particular attention must be paid to signal status processing by function diagrams
    - Plant-specific implementation of operation mode release and the intended purpose of the individual operation modes are tested
    - Plant-specific I&C functions implemented in the function diagrams are tested
      - Component type tests address structural characteristics, but not function diagram organization (i.e., correct functionality, fault containment strategies, response time requirements, and communication links)

TELEPERM XS

Alignment with IEEE Std 1012 Testing Activities

- IEEE Std 1012-1998 describes four testing activities:
  - Component Testing [1]
  - Integration Testing
  - System Testing
  - Acceptance Testing
- IEEE Std 1012-1998 Figure 2 shows a progression of test activities occurring during the development process.

[1] Component Testing: Testing conducted to verify the correct implementation of the design and compliance with program requirements for one software element (e.g., unit, module) or a collection of software elements. (Clause 3.1.3)

TELEPERM XS

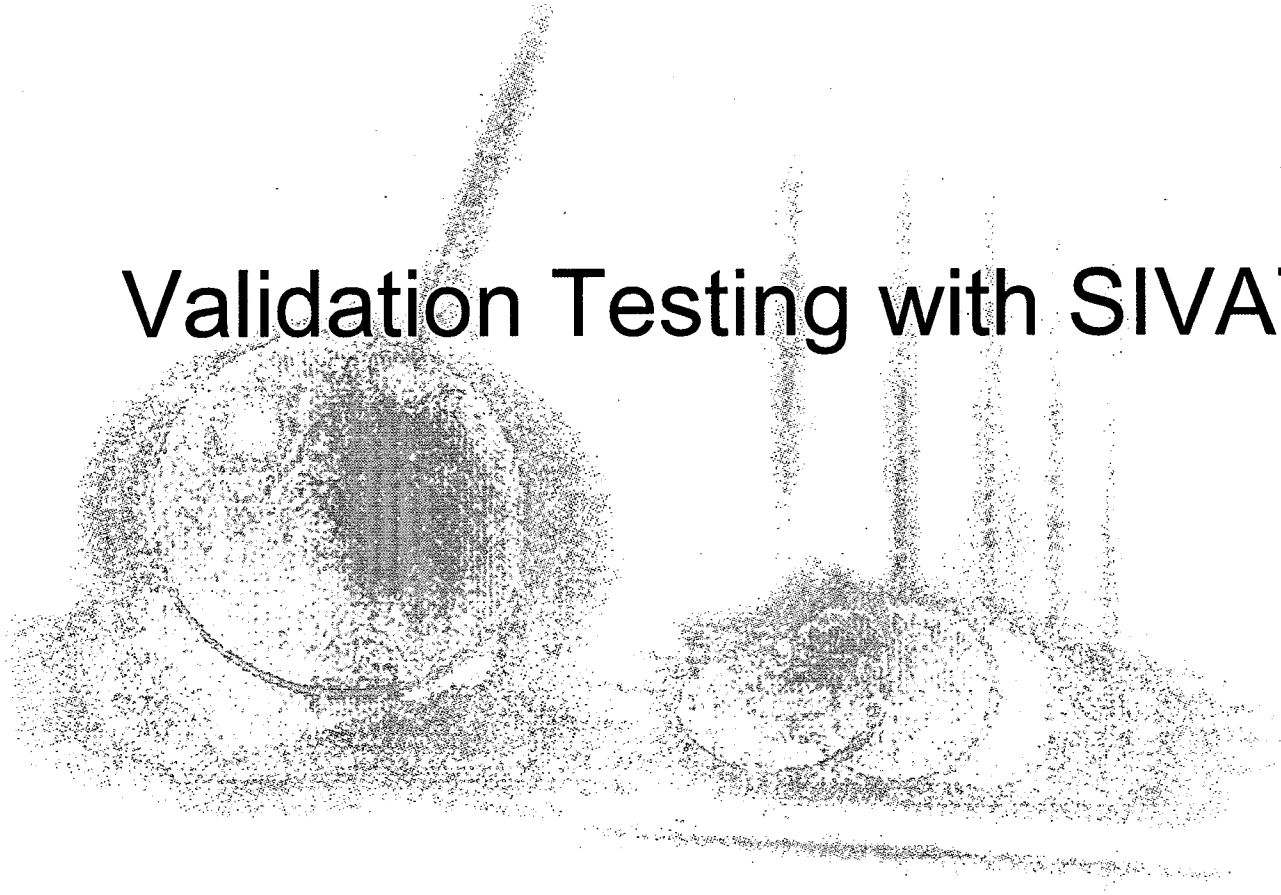
Alignment with IEEE Std 1012 Testing Activities

IEEE Std 1012 Testing Activity	Generic TXS Testing	Project-Specific Testing
Component Testing	**X** (hardware and software type tests)	**Not Applicable** (based on use of qualified hardware and software modules)
Integration Testing	**X**	Application Software: SIVAT for integration of Function Block modules **Optional X** (see Note 1)
		System Components: Pre-FAT prerequisites and procedure dry runs (manufacturing tests)
System Testing	**X**	**X** (integrated in system testing, including FAT, based on use of qualified system components and development tools)
Acceptance Testing	**Not Applicable**	

Legend: **X** indicates alignment with IEEE Std 1012-1998 testing.

Note 1 – Additional application software integration and functional test cases to validate engineering I&C functionality are added to the scope of system validation testing for the case where SIVAT testing is not used for application software integration and functional testing to satisfy IEEE Std 1012-1998 validation requirements. Validation testing with SIVAT is performed as an Implementation Activity task.

Validation Testing with SIVAT



TELEPERM XS
Validation Testing with SIVAT

- SIVAT can be used to perform application software integration and functional testing
  - SIVAT is one layer of validation testing to ensure software quality
- Benefit of application software validation testing with SIVAT is early detection of faults
  - Balance between validation testing during FAT and performing application software validation testing with SIVAT earlier in the process
  - IEEE Std 1008-1987 recognizes that:

    There are significant economic benefits in the early detection of faults. This implies that test set development should start as soon as practical following availability of the unit requirements documentation because of the resulting requirements verification and validation. It also implies that as much as practical should be tested at the unit level. (Paragraph B2.4)
- Early detection of application software faults through validation testing with SIVAT serves to reduce project risks earlier in development process

TELEPERM XS
Validation Testing with SIVAT

TELEPERM XS
Validation Testing with SIVAT

- Goal of project-specific SIVAT testing is to validate correct implementation of functionality specified in Software Design Description
- Minimum criteria are:

TELEPERM XS
Validation Testing with SIVAT

- Test of the Required I&C Functionality

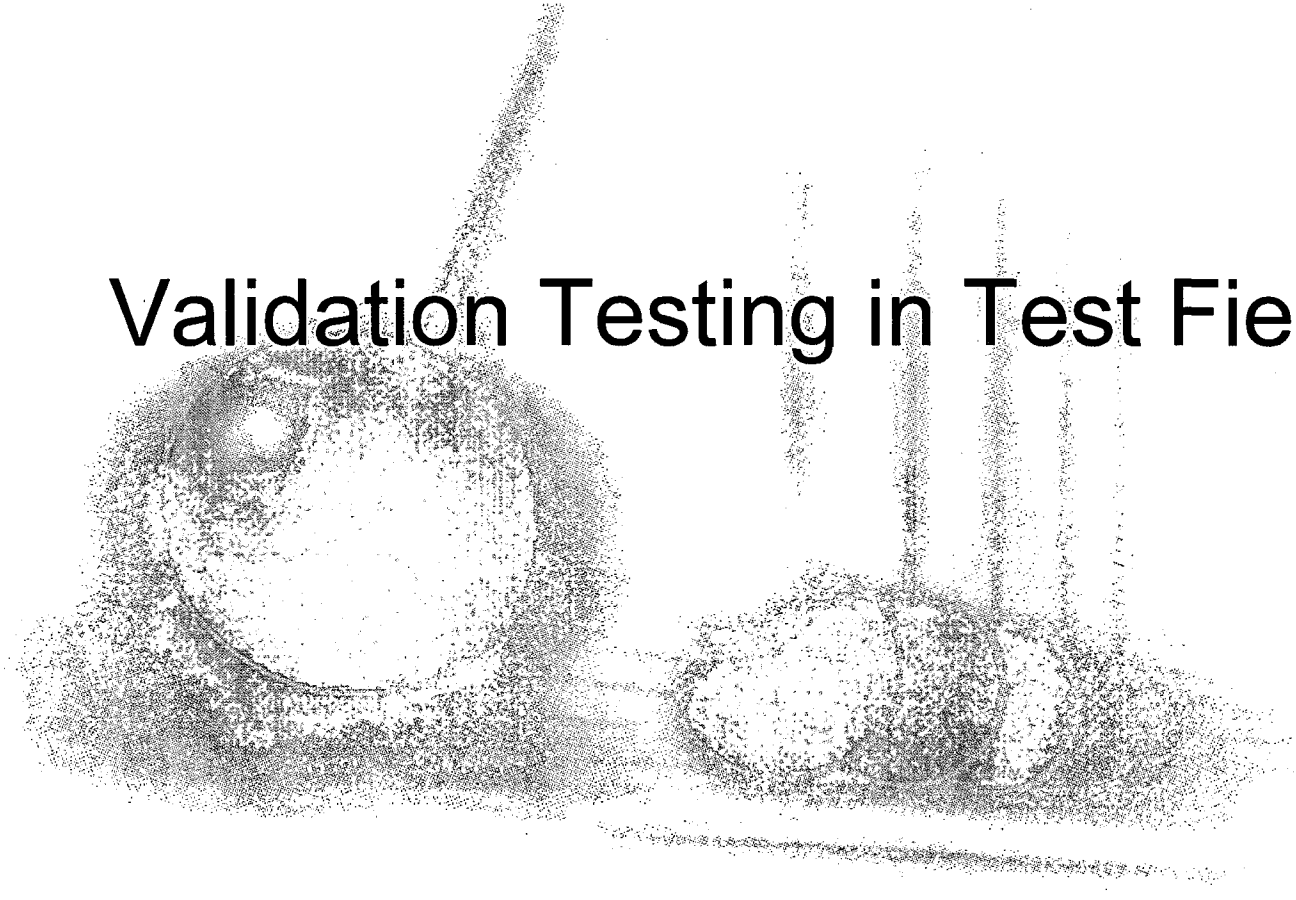
TELEPERM XS
Validation Testing with SIVAT

- The following system characteristics are not tested by SIVAT:

TELEPERM XS
Validation Testing with SIVAT

- Testing with SIVAT is optional but preferred approach to TXS application software integration testing, since it leads to early detection and correction of application software faults and reduce project risks
  - Testing with SIVAT can serve as module or unit testing (i.e., function diagram or function diagram group testing)
  - Can also serve as integration testing of the TXS application software (i.e., testing of all TXS application software modules working together) within limitations of simulation
- Additional testing is performed as part of manufacturing tests to address limitations of simulation testing
- SIVAT test cases are designed to be repeated to support validation of future changes to TXS application software

Validation Testing in Test Field



TELEPERM XS
Validation Testing in Test Field

- The system test in the test field, including FAT, fulfills system integration and acceptance testing requirements
  - Always includes test cases to address limitations of SIVAT testing
  - Additional test cases added to validate I&C functionality where SIVAT testing is not used to satisfy IEEE Std 1012-1998 TXS application software validation requirements
- FAT demonstrates to the customer that the finished system meets the functional and safety requirements

TELEPERM XS
Validation Testing in Test Field

- FAT Prerequisite Tests (Manufacturing Tests)

TELEPERM XS
Validation Testing in Test Field

- Test of Required I&C Functionality

TELEPERM XS
Validation Testing in Test Field

- Test of Process Engineering Requirements

TELEPERM XS
Validation Testing in Test Field

- Hardware Failure Tests

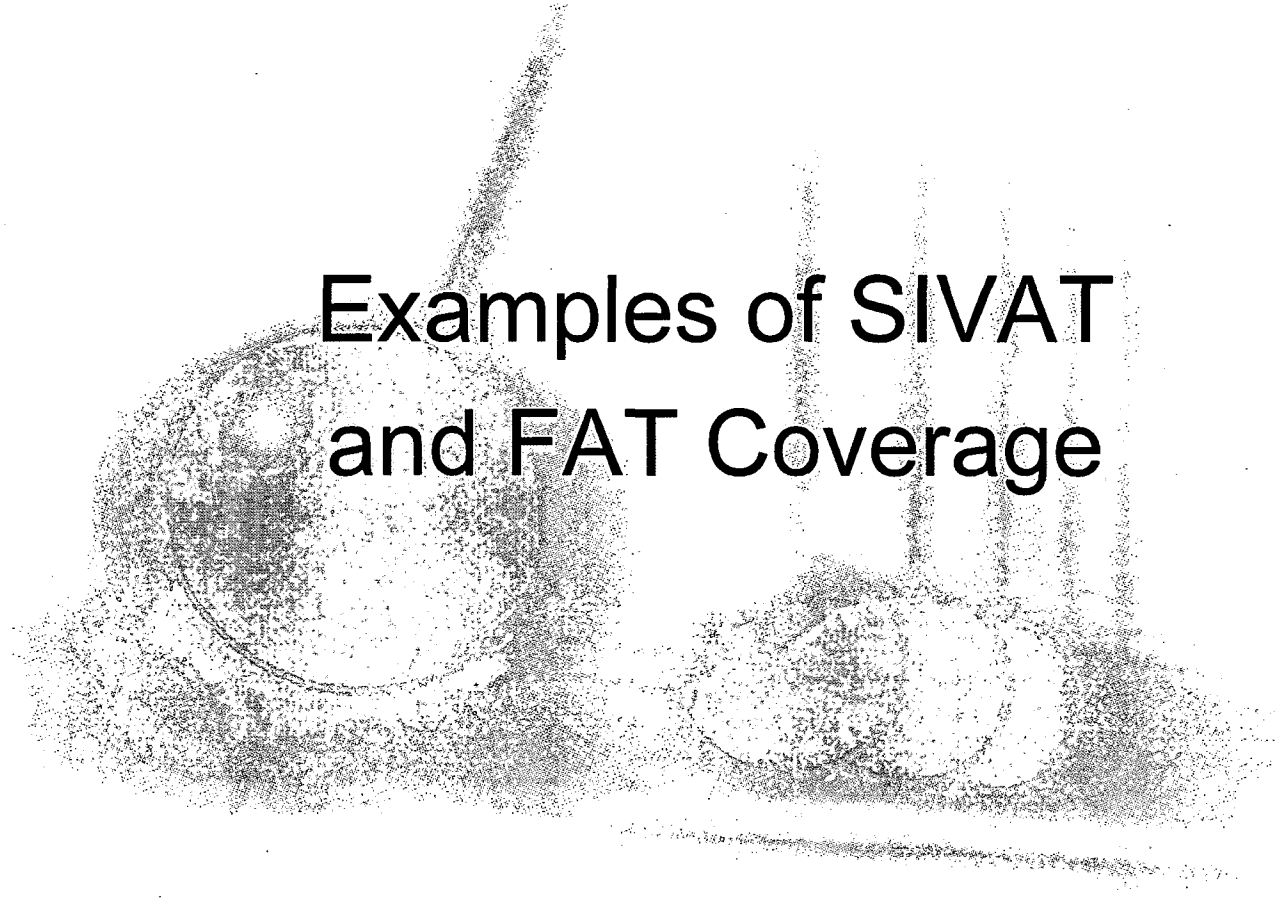
TELEPERM XS
Validation Testing in Test Field

- Other Test Field Validation Tests

TELEPERM XS
Validation Testing with SIVAT

- Project-specific test field validation tests satisfy IEEE Std 1012-1998 system and acceptance testing requirements for project-specific TXS systems
  - This testing always includes test cases to address the limitations of simulation testing with SIVAT.
- FAT prerequisite tests (or manufacturing tests) provide integration testing of the TXS System hardware components, which addresses one limitation of the simulation environment (i.e., hardware interface).
- Various system tests are preformed during FAT to address simulation environment limitation (i.e., dynamic effects associated with hardware interfaces).
- Additional application software integration and functional test cases to validate engineering I&C functionality are added where SIVAT testing is not used to satisfy IEEE Std 1012-1998 validation requirements for TXS application software validation

Examples of SIVAT and FAT Coverage



TELEPERM XS
Examples of SIVAT and FAT Coverage

- Scenario 1 – No Change to Oconee RPS and ESFAS Specifications

TELEPERM XS

Examples of SIVAT and FAT Coverage

- Scenario 2 – Credit for SIVAT Using Existing Oconee SIVAT Test Cases

TELEPERM XS
Examples of SIVAT and FAT Coverage

- Scenario 3 – Use of Optimized Oconee SIVAT Test Cases

Project-Specific Verification and Validation Program

TELEPERM XS
Project-Specific V&V Program

- TXS application software V&V process
  - Comprehensive and objective assessment of software products and processes throughout software life cycle.
- V&V Activities and Tasks drawn from IEEE Std 1012-1998
  - Requirements Phase V&V
  - Design Phase V&V
  - Implementation V&V
  - Test Phase V&V

TELEPERM XS
Requirements Phase V&V

- Software Requirements Traceability Analysis
  - Requirements Tracing between the SRS and system functional requirements
- Software Requirements Evaluation
  - Ensures that SRS adequately defines software requirements necessary to perform intended functions
- Interface Analysis
- Criticality Analysis
- Configuration Management Assessment
- Hazard Analysis
- Risk Analysis
- Security Assessment

TELEPERM XS
Design Phase V&V

- Traceability Analysis
  - Tracing between SDD and SRS
- Software Design Evaluation
  - Evaluates SDD in accordance with established standards, practices, and conventions
- Interface Analysis
- Criticality Analysis
- SIVAT Test Plan Generation
  - Test of the Application Software functionality specified in the SDD
- Acceptance Test Plan Generation
  - Test of hardware and software for total system, from cabinet input terminals to output terminals, and peripheral items such as TXS Service Unit and TXS Gateway
- SIVAT Test Specification and Procedure Generation
  - Test design and test cases to define software features to be tested, approach refinements, and pass/fail criteria
- Hazard Analysis
- Risk Analysis
- Security Assessment

TELEPERM XS
Implementation Phase V&V

- Traceability Analysis

[ ]

- Source Code and Source Code Documentation Evaluation

[ ]

- Interface Analysis
- SIVAT Test Report and Test Incident Report Verification (SIL 4, 3, 2)
  - SIVAT Test Report and SIVAT Test Incident Report validate that application software satisfies test acceptance criteria with no remaining unresolved test incidents
- Hazard Analysis
- Risk Analysis
- Security Assessment

TELEPERM XS
Test Phase V&V

- Acceptance Test Specification and Procedure Generation
  - Test design and test cases to define system features to be tested, approach refinements, and pass/fail criteria
- Traceability Analysis
  - Traces Acceptance Test Procedures to the Acceptance Test Plan and to the software requirements in the Functional Requirements Specification
- Acceptance Test Verification
  - Acceptance Test Report Test Incident Report validate that system satisfies test acceptance criteria with no remaining unresolved test incidents
  - Verifies correct versions of software were used in the Acceptance Test
- Hazard Analysis
- Risk Analysis
- Security Assessment

TELEPERM XS
Verification & Validation Summary

Questions?