Presented by

Dr. Oliver Meyer / Verified Systems International

Julien Pfefferkorn / Airbus

Automated IMA Module Configuration Testing

Overview

1. Integrated Modular Avionics – Introduction and Methods
2. Test Objectives
3. Test Setup and Architecture
4. Test Validation
5. Test Execution and Results
6. Lessons Learned

1 – Integrated Modular Avionics

- Situation in current Aircraft
  - Separate controllers for each application
  - Specific hardware for each Aircraft type
  - Different interfaces, operating systems etc.
  - High expenses for design, validation, verification, qualification, maintenance etc.

- Integrated Modular Avionics (IMA)
  - Unified hardware platform for avionics controllers
  - Multiple applications on a single controller
  - Technology transparency – independence of HW / SW
  - Highly configurable
  - Currently used for A380 and A400M

1 – IMA Module Characteristics

- Hard Real-Time capability
- Spatial and temporal partitioning

- Large variety of standardized I/O types
  - Discretes, Analogues, NTC, PT100, TSI, …
  - Data busses: AFDX, CAN, ARINC-429
- Unified and configurable I/O access

- Standard API for C and Ada (ARINC 653)

1 – Configurability (Interface Control Document: ICD)

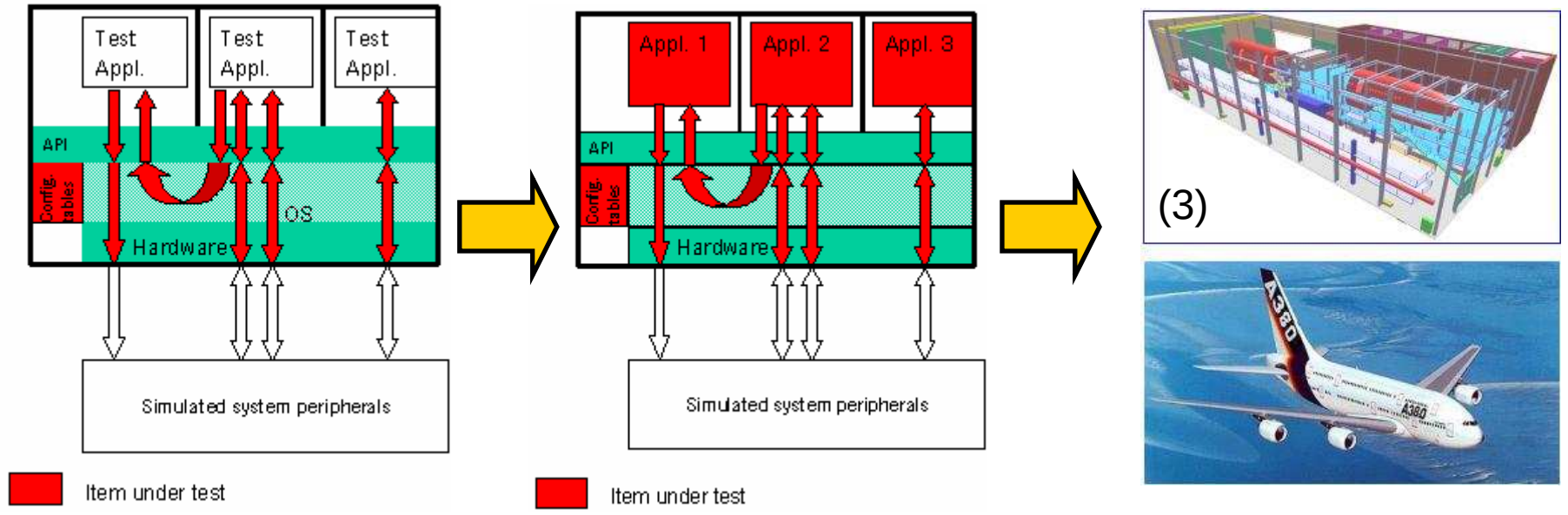
- Large number of potential applications require flexibility
- ICD contains definitions for
  - I/O resources allocation, routing, data formats
  - Scheduling of partitions
  - Memory and Non Volatile Memory (NVM) allocation
  - Health monitoring
  - General module characteristics (Caching, RAM sizes, …)

- ICD layers are configured by different parties with different responsibilities
  - Each application developer
  - Module Integrator

Summarized: High Complexity of ICD entries

1 – IMA Testing Approach for Cabin Systems

- Incremental strategy
  - Configured IMA modules tests on respective test bench (1)
  - Integrated IMA modules tests on respective test bench (2)
  - System tests on respective system test benches and Cabin 0 (3)
  - Ground tests and flight tests in real Aircraft (4)



2 – Test Objectives

- Configured IMA modules tests enable:
  - Decoupling of application specific issues and IMA modules configuration issues
  - Early failure detection
    - Detection of issues before there is an impact on the respective system
    - Improved failure detection and isolation between
      - OS, I/O, configuration
      - Application

2 – Test Objectives

Objective: Dynamic testing of IMA configuration

- Basic ICD content checking
  - Verify ICD consistency
  - Verify ICD interpretation by the IMA module
- Development process and debugging support
  - Detect and correct ICD inconsistencies as early as possible
  - Ease the localisation of anomalies
- Robustness checking of spatial and temporal partitioning

2 – Automated Tests with Verified's RT-Tester

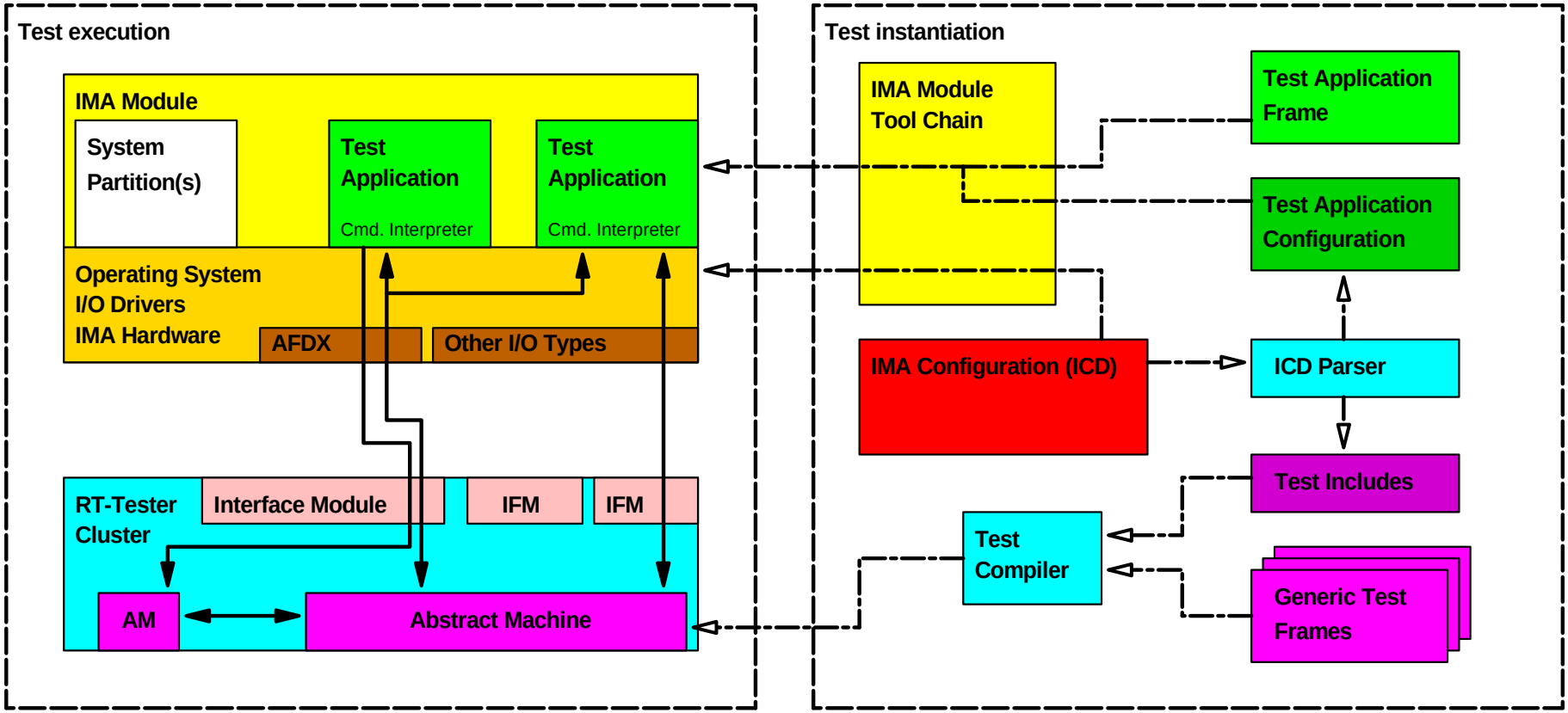
Large number of ICDs to be tested

→ Automated Testing

Verified Systems RT-Tester:

- Specification-based Hard Real-Time testing
- Scalable test cluster architecture
- Testing of functional and temporal behaviour
- On-the-fly SUT response evaluation

3 – Test Setup Architecture



AFDX: Avionics Full Duplex Switched Ethernet

AM: Abstract Machine

ICD: Interface Control Document

IFM: Interface Module

IMA: Integrated Modular Avionics

3 – Test Classes: Normal Behaviour

- I/O Read / Write access
- Bandwidth limitations and debouncing times
- I/O status and value monitoring (readback)

- Memory allocation and access / restrictions
- NVM access and timing

- Scheduling in maximum load conditions

- Power consumption in various operating cases

3 – Test Classes: General Robustness

- Illegal / wrong port access
- Illegal data transmission / reception
- Data range excess

- Memory allocation overflows
- Memory access to illegal areas
- Illegal / wrong NVM access

3 – Test Classes: Partitioning

- Memory access of other partitions
- NVM access of other partitions

- Access ports of other partitions
- I/O port independencies

- Scheduling violations and health monitoring

4 – Test and Test Environment Validation

- Manual validation:

  Test Procedures against Test Design document

- Runtime validation by Mutation Testing
  - Systematic modification of ICD on IMA module only
  - Derivation of violation(s) to be detected
  - Test instantiation and execution based on original ICD
  - Test result comparison against expected violation(s)

- Runtime validation revealed two errors in the test environment

5 – Test Preparation for Test Execution

Test of new ICD batch:

- About 1 hour preparation per SUT
- Import of configuration to be tested
  - Identification of commanding channels for test application
  - No change required in test environment
- Preparation of SUT
  - Loading of configuration to be tested on IMA module
  - Load of test application on IMA module

5 – Test Execution

- Fully automated

  - Enables test execution also over night and at week-ends
    - 4 to 36 hours test execution duration depending on:
      - Requested coverage
      - IMA module configuration complexity

  - Enables high level of regression testing

    “Also as a consequence of the introduction of new bugs, program maintenance requires far more system testing per statement written than any other programming. Theoretically, after each fix one must run the entire batch of test cases previously run against the system, to ensure that it has not been damaged in an obscure way. In practice, such regression testing must indeed approximate this theoretical idea […] .”

    Fred Brooks, The Mythical Man Month (page 122)

5 – Test results analysis and documentation

- Automated analysis and documentation
  - 1 day per SUT
  - Automatic pass / fail analysis for each test case
  - Post-processing of log files
    - Accelerates error cause analysis
    - Performed in parallel to execution of following test case
  - Full traceability of the test process
    - System requirements
    - Test objectives
    - Test cases
    - Test execution
    - Problem reports

5 – Test results analysis and documentation

- About 80 different discrepancies detected in IMA modules prototypes using configured IMA tests
  - Type of issues
    - I/O (75%)
    - Memory (RAM) (15%)
    - Non Volatile Memory (5%)
    - Miscellaneous (5%)
  - Issue cause
    - Configuration issues (50%)
    - OS issues (15%)
    - Documentation (e.g. user guide) (10%)
    - Test environment (20%)

6 – A380 configured IMA tests lessons learned

- Development cycle duration reduced
  - Drastically reduced test campaign duration (reduced by approx. 80%)
  - Break even = 3 test campaign iterations
    - 10 test campaign iterations until 03/2006 for each SUT type
    - Several iterations will be required for different A380 versions
    - Test environment can be partly reused for A400M Tests
- Automated configured IMA tests enabled regression tests
- Accelerated failure detection and analysis through decoupling of OS / configuration and application issues
  - Maturity of IMA configuration and IMA controller reached at early stage of development