A Product Line Architecture for Army Aviation Diagnostics and Maintenance: Views and Evolution

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Sholom Cohen
Ken Capolongo, CE-LCMC
May 2007

Contents / Agenda

Advanced Multiplex Test System
Product Line Concepts
  • Definition
  • Application to AMTS
Architecture Views
  • System Context
  • Software
Evolution
  • Product line growth
  • Tele-maintenance
  • Condition-Based Maintenance
AMTS Product Description

A line of off-board (non-embedded) diagnostic products to:

- support Army and Joint Aviation weapon platform maintenance
- assist maintainers
- support all maintenance levels (flight-line, intermediate, and depot)
- diagnose and repair avionics data bus networks faults

1553B System for AH-64A

- Doppler,Mission equipment, Target Acquisition and Designation System
- HARS, Fire Control System
- FDL, Flight Data Link
- MRTU, Multi-Radio Terminal Unit
- BCC, Bus Controller MDLS, Multiple Display Line Replaceable Unit
- IHADSS, Integrated Helmet and Display Sight System
- TADS, Target Acquisition and Designation System
- DASS, Digital Automatic Stabilization System
- FCS, Flight Control System
- ORT, Optical Relay Tube
- HIPS, Heading and Inertial Platform System
- NAV, Navigation System
- GCS, Ground Control Station
- IRU, Inertial Reference Unit
- SPS, Secondary Power System
- ECS, Environmental Control System
AMTS Business Goals

- Improve productivity of maintenance activities and eliminate false diagnoses
  - Increases aircraft readiness
  - Reduces aircraft operation and sustainment (O&S) costs
- Meet anticipated product demand with current resources
- Reduce product development time and cost
- Minimize customer’s cost of entry
- Provide products for various applications /maintenance levels
  - Operational: Aviation or Vehicle Platforms
  - Intermediate: Line Replaceable Unit (LRU)
  - Depot: from LRU to card level
- Support collaborative (i.e., tele-) maintenance

What’s a Product Line?

A software product line is defined as

- “A set of software-intensive systems that share a common, managed set of features satisfying the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way.”
# Product Line Features and Variations

<table>
<thead>
<tr>
<th>Platforms supported</th>
<th>Bus Type</th>
<th>Message handling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Helo</td>
<td>1553 (1553A, 1553B)</td>
<td>From text</td>
</tr>
<tr>
<td>Apache A</td>
<td>Ethernet</td>
<td>From XML</td>
</tr>
<tr>
<td>Apache D</td>
<td>1773</td>
<td>Message analysis</td>
</tr>
<tr>
<td>UH-60</td>
<td>ARINC (commercial aircraft)</td>
<td>Read</td>
</tr>
<tr>
<td>Aircraft</td>
<td>CAN? (automotive)</td>
<td>Write (for sim/stim tool)</td>
</tr>
<tr>
<td>F-18</td>
<td>Single bus</td>
<td>Test level</td>
</tr>
<tr>
<td>C-17 (under discussion)</td>
<td>Multiple bus</td>
<td>Operational</td>
</tr>
<tr>
<td>Ground vehicles</td>
<td>Single type</td>
<td>Intermediate</td>
</tr>
<tr>
<td>M1A1 Abrams</td>
<td>Mixed type</td>
<td>Depot</td>
</tr>
<tr>
<td>Variations within each by tail # or other vehicle feature</td>
<td>Nested (i.e., bus within bus; e.g., JTRS)</td>
<td>Development</td>
</tr>
</tbody>
</table>

## Variation Example – Maintainer Interface

AMTS has detected a 1553 bus problem with the RHE or its associated GTU and/or cabling. To further fault isolate, connect AMTS Bus Cable Adapter AA1292964 (BMHE labeled) between the RHE and the 1553 mid bus as follows:

1. Locate the BMHE (Point B in picture)
2. Use the BMHE and connect the BMHE to the 1553 bus
3. The BMHE should now be connected to the 1553 bus
4. Replace the BMHE with the correct 1553 bus cable adapter
5. Ensure proper cabling
6. Verify proper function
How AMTS Was Created

- Leveraged years of experience supporting avionics maintenance and developing software/hardware tools
- Exploited MIL-STD-1553 commonality
- Established management support to develop prototype
- Mined existing assets
- Incorporated iterative/reactive Product Line engineering approach

AMTS Decomposition View

- Core Environment - common modules across the product line
- Product Configuration - modules that tailor core environment for specific platforms
Core Environment Decomposition View

Common assets for building testing systems for specific aviation platforms, systems, and subsystems.

Core Environment (CE)

- **Console**: data driven software utilizes
  - Non-specific data and graphics to provide a universal data bus analyzer
  - Specific data, graphics, and software modules to provide platform data bus specific diagnostics
- **CID**: software adaptor utilizes
  - Vendor libraries to provide device communication independent of manufacturer

Common assets include a 1553 data bus generic monitoring system without platform specific data (Bus Tester Tool Kit).

- Every possible LRU on data bus, up to 32
- Specific LRU's unknown. Where located on bus unknown
- Configuration capability to be added in future with tools to build plug-ins
Product Configuration (PC)

Product core assets and product specifics

• employ reverse engineering of platform system to understand behavior for database system information and message repository
• Use Multiplex ICD to identify product-specific messages
• Use Bus controller SRS to understand behavior

Product modules

• Core environment asset instances (console and CID)
• Product core (PCore) & PCore instances
• Product specifics (e.g., Read Codes for AH-64A)

AMTS Layered Context

Core Environment

Product

Cabling

AMTS Core Environment

Cabling

AMTS Instance (contains CE, product core, and product specifics)

Any Aviation platform (generic)

Aviation platform (specific)
Layer View – Core and Product Configuration

Core Environment

Behavior view on next Slide

Sequence Diagram for Get Message Stack
Evolution – Product Line Growth

AMTS Product Line

- Console team builds assets:
  - Console assets
  - CID assets

Product Configuration Team builds products:

- Product Core
- Product Specific
- Product Requirements

CID Team builds assets:

- CID Assets

SLV Product Line

- Builds loader/verifiers for:
  - IDM
  - ARC220
  - EGI

AMAT Product Line

- Builds LRU tester for:
  - IDM
  - ARC220
  - EGI

Evolution – Tele-maintenance

Remote Client

AMTS Tablet PC

Core Environment (CE)

- Console
- CDU
- Controller
- Server
- Multi-File-IO
- Multi-Media
- Configuration
- Multimedia

Common Interface Driver (CID)

- Device Adapter
- Vendor Library

Key:

- Subsystem
- Module
- Sub-Module
- Repository
Tele-maintenance Support

Evolution – Condition Based Maintenance

Need expressed by maintainers in field
Aviation Responsive Maintenance System (ARMS) to consolidate and provide situational awareness picture
- of aircraft readiness for use in CBM
- of current data state across the fleet
- for playback on AMTS (collect data (one-hour’s worth) and the SME who is providing assistance has it for analysis.

Use existing built-in analysis capabilities as model to determine how to retrofit this in off-board for all aircraft

Product line approach is an enabler
Evolution for CBM

Evolution – Changes to Behavior for CBM
Summary

AMTS architecture supports multiple and growing list of diagnostic/maintenance products

Changes managed through architecture evolution
  • Identification of new goals (e.g., adding CBM capability)
  • Technology changes (e.g., secure collaboration support)

Variation mechanisms not covered in this presentation – possible topic for working group discussions