Ada Usage in HMI for Onboard Safety Critical Applications

Clara María Arcones Gabriel
Enrique Chicharro López
Ismael Lafoz Pastor

Air Refueling Domain
Mission Systems CoC
AIRBUS DEFENCE & SPACE
MFCD System

- Multifunction and Control Display (MFCD): interface with refuelling operator to control and display the Air to Air Refuelling (AAR) Systems information, warning and indications.
MFCD System

• Multifunction and Control Display (MFCD): interface with refuelling operator to control and display the Air to Air Refuelling (AAR) Systems information, warning and indications.

• MFCD is not just an HMI application but an interface to perform the Refuelling Operation.

• It is a certified application, DO178B, level DAL B, due to it can set values whose failure conditions are considered Harazardous (example BOOM envelope)
MFCD Development

- MFCD HMI developed with VAPS 6.3.1 (Virtual Avionics Prototyping System)
- MFCD SW application developed with Ada 95
- Partitioning-based SW Architecture
- Esterline PU-2000
- Esterline CHDD-268
- VxWorks 653 (2.2.1.1)
- QCG 2.0 / VAPS 6.3.1

Generating QCG code is basically a mapping of metafile commands to runtime commands. This runtime commands are code generated as static, constant data structures. This data is interpreted by the runtime library, which also reads the dynamic channel data, to produce the application’s runtime behavior.

The QCG runtime library was developed using the MISRA C standard. MISRA C is an initiative to define a safe C subset and coding standard for embedded systems.
Basic VAPS File Types

Frames
• .FRM Contain Graphics Objects and Data Connections.

Metafile
• .VMF ASCII Meta Tagged version of Frame file.
  ➢ QCG code is generated from VAPS metafiles
  ➢ Metafiles are ASCII Human Readable versions of binary Frames.
  ➢ Frames must be converted to Metafiles before they can be built in QCG.

Channels
• .CHA Data Structure Objects, are used to connect data to Graphical objects.
Process

- C code is imported into Ada 95 with “pragma import”
- Type is created to handler frames and channels:

```
type Channel_Name_T is 
array (Channel_T) of String (1 .. 21);
```

Where Channel_T is a type with all the VAPS Channels.

Two constants were created, one to handle channels and one to handle frames.

- `Frame_Name_1_C : constant Channel_Name_T := ....`
- `Channel_Name_C : constant Channel_Name_T := .....`

```ada
Channel_Name_C : constant Channel_Name_T :=
(Aircraft_Common_Page
Aircraft_List
Aircraft_Main_Page
Bevs_Camera_Page
Bevs_Image_Page)

Frame_Name_1_C : constant Channel_Name_T :=
(Aircraft_Common_Page
Aircraft_List
Aircraft_Main_Page
Bevs_Camera_Page
Bevs_Image_Page
Bevs_Lights_Laser_Page
Bevs_Main_Page)
```

```
VAPS Channels

VAPS Channel is a data structure that allows information to be sent to, from and between VAPS objects and frames.

Channels has:

- **SCOPE:**
  - Local
  - Session

- **TYPE:**
  - Fast
  - Queued

- One or more “members”
  - Scalar (1 element)
  - 1,2,3 dimensional array of elements

There is one to one mapping relationship between VAPS channels and MFCD Ada code, so in both files the channels members has to be written in the same order.

```ada
type Channel_T is record
  Aircraft_Speed, Altitude
  Left_Pod_Amber_Signal_Light, Left_Pod_Box
  Left_Pod_Green_Signal_Light, Left_Pod_Hose_Length
  Aircraft_Speed.State, Aircraft_Speed.Variant
  Aircraft_Speed.Value
  Altitude.State, Altitude.Variant
  Altitude.Value
  LP_Amber_Signal_Light.State, LP_Amber_Signal_Light.Variant
  LP_Amber_Signal_Light.Value
end record;
```
Additional Uses

This application SW has been reused to develop other useful tools:

- **Simulator**
  - To evaluate HMI aspects and behaviour quick and easy in a desktop computer.
  - Debugger.

- **Offline Refuelling Operation Simulator (OROS)**
  - Replay air refuelling mission on ground including operator actions.
Simulator

VAPS Code Generators

- CCG Lite 3.1 → C Code Generator (Host – Windows)
  - Allows users to create executables or libraries (dll) based on VAPS. Output is ANSI-standard C code.
- QCG 2.0 → Qualified C Generator (Target)
  - Conformance to DO-178B level A.

Application

- Client, Server, running on a PC:
  - Simulated PUs and Displays running on desktop or
  - Real PUs and Displays
- All buses simulated through Ethernet sockets
- Deployment possible on multiple computers
Offline Refuelling Operation Simulator (OROS)
Offline Refuelling Operation Simulator (OROS)
Offline Refuelling Operation Simulator (OROS)
D2CU System

- Two D2CU (Data Display Control Unit) are installed in the cockpit to provide Information to the pilots about the C295 Aircraft Data for this Fly By Wire Prototype under ClearSky2 project.
D2CU Development

• **HW**
  • Esterline PU-2000 & CHDD-268

• **SW**
  • Partitioning-based SW Architecture (4 partitions)
  • VxWorks 653 (2.2.1.1)
  • D2CU developed with SCADE, HMI (Display) + Behaviour (Suite)
    • KCG6.6 - Qualifiable as DO-330 TQL-1 tool under DO-178C or DO-178B
  • IO & Moding partitions developed with Ada 95
Process (Lessons Learnt)

• **First Prototype**
  - Each new model integration took several weeks. Mapping Ada Types and C-structs generated by KCG are complex.
  - SCADE Display KCG has been qualified as development tool for DO-178B up to level A

• **Actual Status**
  - Separate C/Ada Partitions communicating data through Apex Ports.
  - Data Interface between partitions defined in Doors.
  - Several model integration done in one day.
  - Types range protection included in partition communication to avoid abnormal situations.

• **Future**
  - AESA Flight Permit for first flight expected by end 2016.
D2CU Architecture

• Current D2CU Partition Architecture

![Diagram showing the current D2CU Partition Architecture with nodes for SCADE_Display, Moding, IO, SCADE_Suite, C Code Generated with KCG, C Manual Code, SDY_Data, Suite_Outputs, IO_Data, Bus_Status, Master_Slave_Status, Ada95, and IO.]
D2CU & SCADE Tools

• SCADE Display
  An easy environment to create HMI specification. It includes some simulation capabilities allowing the user to test in a computer before loading the sw into the target. Reducing time/costs.

• SCADE Suite
  - Simple logic for activating some indications.
  - Complex logics (Master/slave selection and other) remain in Ada95

• Requirements Gateway
  This tools allows an easy way to trace DOORS requirements to any object in the specification/model.

• SCADE Test
  • Automatic test tool including HMI (currently being evaluated).
Questions?