



**HAL**  
open science

## Flight Test Engineer Station for A350 aircraft

Yves Marcet, Jean-Francois Dausse, Guillaume Monsarrat, Olivier Gallot

► **To cite this version:**

Yves Marcet, Jean-Francois Dausse, Guillaume Monsarrat, Olivier Gallot. Flight Test Engineer Station for A350 aircraft. Embedded Real Time Software and Systems (ERTS2012), Feb 2012, Toulouse, France. hal-02189905

**HAL Id: hal-02189905**

**<https://hal.science/hal-02189905>**

Submitted on 20 Jul 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Flight Test Engineer Station for A350 aircraft

Yves Marcet – Architect & Project Leader  
Jean-Francois Dausse – Architect  
Guillaume Monsarrat – Architect  
Olivier Gallot – Architect  
AIRBUS  
EVIDA M6324  
316, route de Bayonne  
31060 TOULOUSE Cedex 09

## 1. SUMMARY

*On development aircraft, the Flight Test Engineer Station (FTES) allows onboard engineers to lead flight tests.*

*This system gives access in real time to the aircraft parameters (~350.000 monitored parameters on A380), allows the control and the monitoring of flight test specific systems (water ballast, data recorders, video system...) and monitors the whole flight test installation.*

*Regardless of the amount of information to display, the graphical user interface should be “user friendly” and “easy to use” enough to allow engineers to focus on testing. The system is based on an enhanced multi-screen display system and on a tablet PC interface for some specific controls.*

*To fulfil the high level of computing needs, the FTES is based on a distributed architecture: several multi-core computers are interconnected via Ethernet. This modular architecture allows fault tolerance concepts and offers an attractive level of scalability. Virtualization products are used to rationalize and optimise the number of servers.*

*Due to operational constraints, the system should be autonomous and based on robust technical solutions. The hardware, operating system, middleware and software are designed or customised accordingly.*

## 2. INTRODUCTION

In order to deliver a new aircraft family to airline companies, Airbus performs a series of tests to obtain the airworthiness certificate. These tests last about a year and are performed in extreme conditions (hot/cold weather, severe vibrations, hard braking, stall...) to explore the entire aircraft operational domain. Flight tests are still performed after the certification and during the entire aircraft lifetime (several years) as they provide engineers a concrete feedback on the aircraft's ongoing improvements.

During the test campaign, one or two onboard Flight Test Engineers (FTE) are in charge of the flight test programs. The FTE have to follow the predefined test procedures, to guide the pilots in their manoeuvre and to monitor aircraft parameters in real-time for measurements validation. In order to fulfil these tasks, FTE rely on a dedicated system called the Flight Test Engineer Station (FTES).

This station (figure1) is a key system of the Flight Test Installation (FTI). Its target is to collect, process and display in real-time all measurements (temperatures, pressures, structural efforts ...) and acquired data from the aircraft, to drive flight test specific equipments and to give in real-time the FTI functional status.

Since very first Airbus flight tests, several generations of FTES have been designed. The evolutions have followed engineers' new ways of working, aircraft specificities and technologies improvements.



Figure 1: Flight Test Engineer Station and associated servers for A380 (2005)

This paper presents the Flight Test Engineer Station which will be set up in A350 aircraft.

The development of this system started end of 2008. The main features of the station were based on A380 and A400M operational feedback and with the main objectives of simplifying the system (to ease its set up and maintenance) and harmonizing the interfaces.

This system is currently at the integration bench step (figure 2); the components of the station are connected together for final ground test.

The installation in the aircraft for the first operational tests is scheduled early 2012. A350 first flight is due in March 2013. As for all previous installation, this system will have an operational lifetime of 10 years.



Figure 2: A350 FTES - Integration bench

### 3. A350 FTES SPECIFICITIES

#### 3.1 GLOBAL OVERVIEW OF THE STATION

The A350 FTES integrates two identical half-station (figure 3) that can accommodate one engineer and gives him access to four main screens and one keyboard/mouse. On each main screen, several graphical applications sorted by flight discipline are available to display aircraft parameters or control/monitor systems.

In the upper central area, the same system than the one available to the pilots in the cockpit displays primary flight parameters.

Between the two engineers, a central console gathers FTI global controls. Security controls remain on mechanical buttons. A touch screen panel hosts a graphical application for all other controls:

- Flight test recorders management
- Activation / deactivation of specific systems (de-icing, cameras, engine systems...)
- Printers management

A pool of dedicated computers manages the different components of the station.

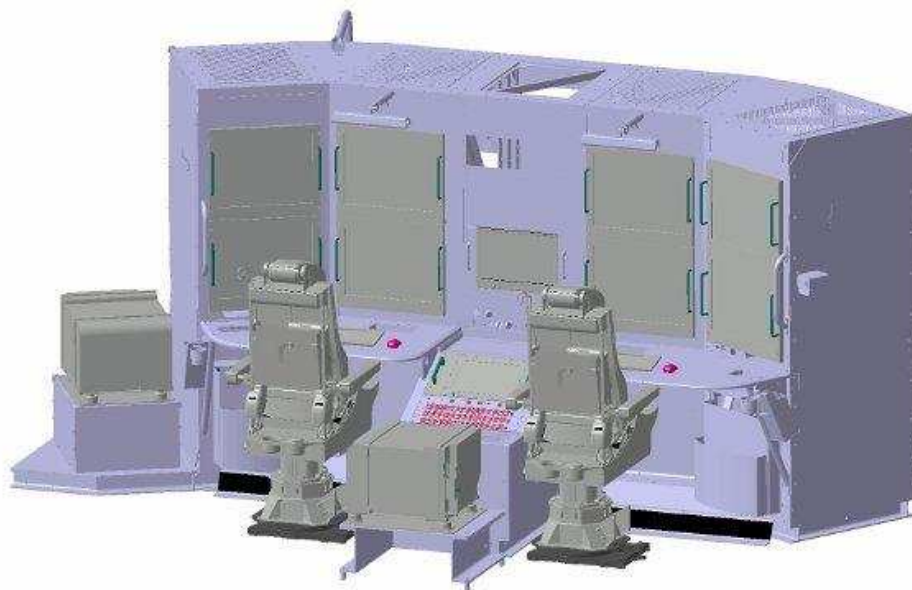


Figure 3: A350 FTES - Digital mock-up

### 3.2 MAIN FUNCTIONS

The FTES main goals are:

- Real time aircraft parameters monitoring  
On the A350 aircraft, ~400.000 parameters are processed in real-time, they come from flight test specific measurements chains (5000 sensors) and avionic buses.
- Real time parameters computation. Some data (e.g. weight) cannot be acquired directly from sensors. They are calculated using initial values and measurable parameters during the flight (e.g. for weight calculation, the application uses the initial weight and the fuel consumption measurement). Once computed, these results are available like any other measurement parameter.
- Video streams management. Digital video cameras are focused on specific aircraft parts (landing gear, rudder, wings...). The standard and high definition streams are recorded and can be displayed in real-time onto the station.
- Driving specific equipments. Some flight test dedicated systems can perform specific measurements or interact with the aircraft: trailing cone (figure 4), electric load benches, Water-Ballast... Engineers should be able to control and monitor these equipments.

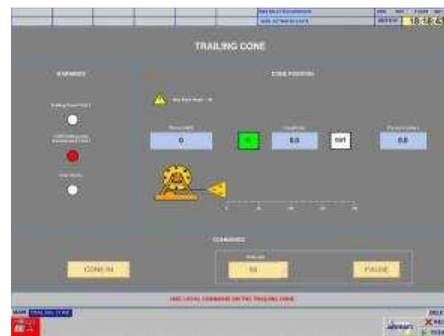
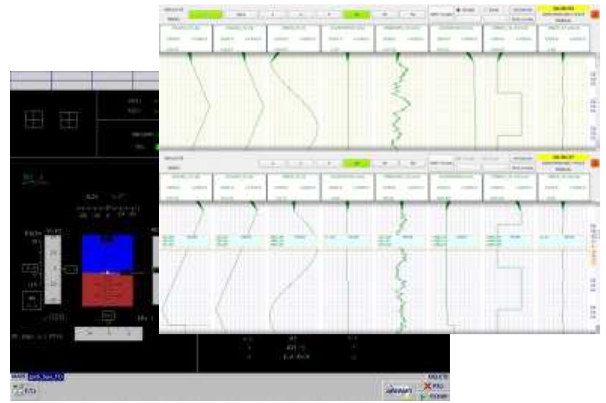
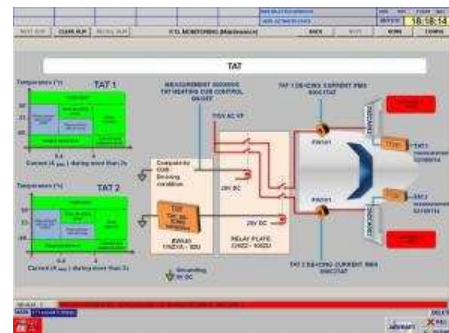


Figure 4: Trailing cone equipment and control page

- Flight test installation status monitoring.  
The FTI operating status is monitored in real-time by the FTES computers. When a failure occurs on a system, functional impacts and corrective actions are displayed.
- Calculation programs hosting. In order to obtain flight test results as soon as possible, automatic calculation programs are hosted on the FTES servers. Results are stored on removable disks and analysed after each flight in the ground data processing centre.
- Building and managing the Telemetry message. Development aircraft are monitored by the ground test team via the telemetry link. Due to a restricted bandwidth (5 Mbits/s), only few parameters (selected by the FTE using a specific application) are sent to the ground data centre. To increase the number of sent parameters, an optimization process compresses in real-time the data.
- Remote display on cockpit screens. In order to help pilots in their manoeuvres, the FTES can send some flight test visualizations on the cockpit display screens.





### 3.3 MAIN OPERATING REQUIREMENTS

The FTES has to fulfil many requirements and constraints. Some of them are linked to the aircraft testing environment, others are related to Airbus' ways of managing flight tests.

#### Operational requirements:

- The system shall be highly reliable despite the harsh environment. Depending on the flight phase a system failure could affect flight safety. The system unavailability could have strong costing impact due to the high cost of a flight hour.
- The overall computing power of the system shall be sufficient enough to host all required functions. A350 data processing should require at least 50% extra processing resources than A380 (~30x1.5GHz processors distributed over several servers).
- Despite its complexity the FTES should be seen as a simple and efficient tool. Station ergonomics shall be very intuitive and "user friendly". During critical flight tests, engineers should stay focused on the aircraft behaviour.
- The system is fully autonomous. There is no link with ground facilities and users (FTE or level one support team) are not computer administrators. The system can be used 16 hrs a day and the sole manual action is pressing the general on/off power switch.

#### Environmental constraints:

- During flight test campaign, the aircraft is exposed to extreme environmental constraints. Onboard systems should be qualified for high vibration levels (up to 6G for shocks), to wide temperature ranges (-15°C / +55°C for operational, -55°C / +85°C for storage) and electromagnetic fields (not to disrupt or being disrupted by the aircraft system radiations).
- Aeronautic life cycles are long: the system operational lifetime shall be at least 10 years. The entire system shall be kept fully operational during this period.
- The maintenance process in the aircraft should often be performed within a limited time. The computers area can be hardly accessible, system size and weight shall be limited (19", 6 Rack Units, 50kg) to ease replacement or handling operations.

#### Architectural requirements:

- The system will host new applications which will be developed throughout the all A350 flight test campaign. More than 250 embedded applications were developed for A380, the A350 certification should require at least as many applications.
- The FTES should integrate interoperability capabilities. Embedded applications are usually developed by the flight test centre ground team, but the system shall also be able to launch software from other Airbus entities designed and developed in other frameworks.
- The architecture should be flexible enough to be adapted from one aircraft to another. It should also be modular enough in order to easily increase or decrease the system computing power.

## 4. IMPLEMENTATION

Many solutions exist in the field of embedded computers but none of them fully fulfil all the previously described requirements. On one hand some embedded computers are rugged enough to meet environmental constraints but the computing power is considerably below the expected one. On the other hand, computing centre solutions are powerful enough but they are not compliant with aircraft testing operational constraints.

A complete rugged system with such a computation power would have such a prohibitive cost that Airbus has decided to design a dedicated solution based on standard hardware and software. This solution, between the computing centre and the embedded calculators, addresses all the specific requirements of the FTES.

### 4.1 FTES SYSTEMS IN THE FLIGHT TEST INSTALLATION

Within the FTI, the computers of the station have many links with the other aircraft systems (figure 5).

The main link of the system is the FTI data bus to acquire the aircraft parameters.

As for A380, the flight test architecture is divided into four levels: sensors level, acquisition level, concentration / distribution level and recording / analysis level.

Measurements and buses acquisition are gathered and sent via Ethernet to the FTES computers. For A350, these flight test data will represent a total flow of 130 Mbits/s.

Up to 12 cameras (standard or high definition) are to be installed in the aircraft. Some video streams are sent in real-time (without any delay caused by compression) for immediate display onto the station screens. Others are encoded to be stored on specific recorders.

The FTES system should be connected to many flight test specific equipments in order to allow their control and monitoring. Depending on the equipment, different protocols and buses are used (Ethernet, RS, Arinc 818, electrical relays ...).

For some specific functions, the FTES computers have direct interaction with the aircraft. These links are restricted and are closely followed in a dedicated risk assessment study.

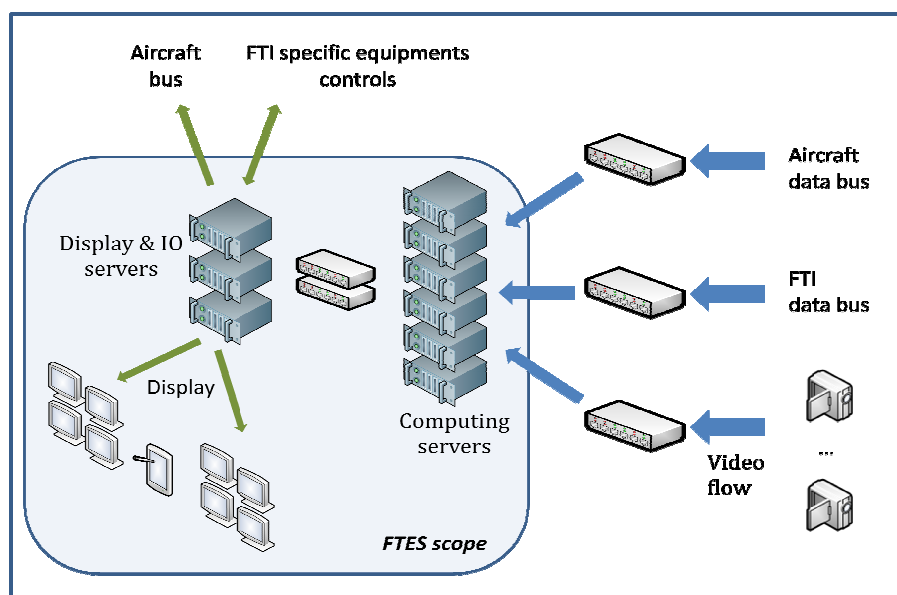


Figure 5: FTES systems in the flight test installation

## 4.2 MAIN PRINCIPLES AND GLOBAL ARCHITECTURE

The overall architecture is widely inspired of computers clusters.

Four main functions are identified for FTES computers:

- Computation and graphical animation
- Displaying
- Input / Output management
- Storage

Each function is realized by a specific cluster. The nodes of the clusters are the servers described in §4.2.

Interconnection is a key point in clusters design. The FTES interconnect is based on Gigabit Ethernet. It is not the best performing network for interconnection but its solid and ubiquitous software support makes it a good choice in this application. The bandwidth is sufficient for internal exchanges and the latency is low enough for communication real-time requirements.

To avoid bottlenecks, ensure the quality of service and for redundancy needs, several links (Figure 6) are defined to separate data.

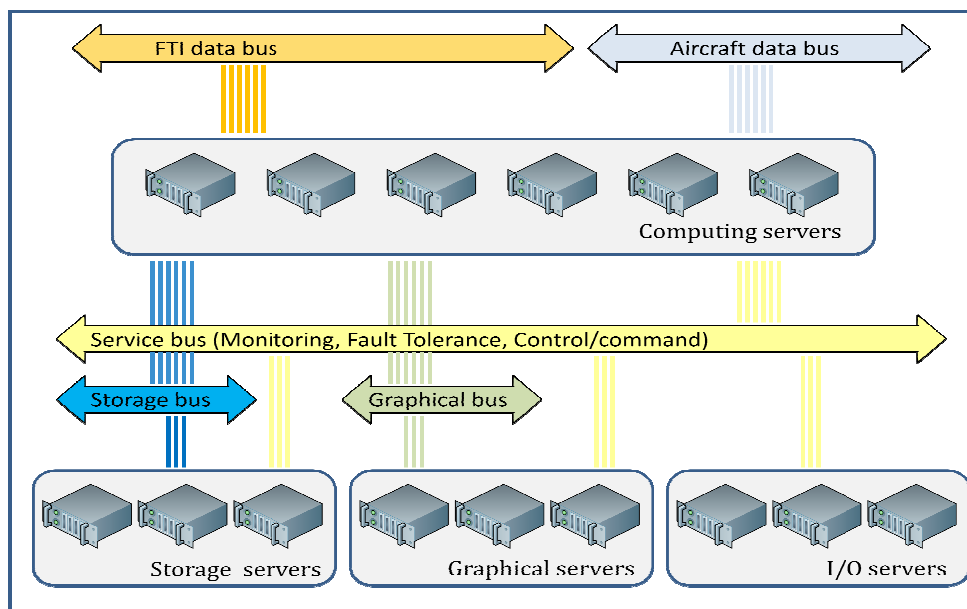


Figure 6: Overall description of FTES system architecture for A350

The main points of this distributed architecture are:

- A high level of modularity: according to the computational power needed, the cluster composition can be tuned from one installation (one aircraft) to another one.
- Scalability: if needed, some extra nodes can be easily added in a cluster.
- Easy maintenance: a node is easy to handle or remove from the aircraft.



### 4.3 HARDWARE

Except for storage, FTES computers are based on standard servers from computing centre. They integrate several multi-core x86 / 64bits processors. Mechanical customizations as well as appropriate choices of internal devices (such as solid-state drives) allow these systems to fulfill the environmental constraints.

A specific node has been designed for each function:

- The computing and graphical animation node is designed upon an off-the-shelf telecom server. This product integrates several Gigabits interfaces, several Nehalem generation CPU's and has a long lifetime support.
- The display and Input / Output nodes are built upon a rugged standard-based server which allows the integration of several specific cards.
- The storage node is a dedicated system. Its specific design allows the hot-swap disks removal. The system is ruggedized and is qualified for a large number of insertion/removal of the media.

In order to reach the targeted durability of 10 years, the system lifetime was a major criterion for the computers selection.

This long continuity will also be ensured by the definition of standard and common interfaces between nodes and the possibility to replace a system by its next generation keeping the same interfaces.

### 4.4 OPERATING SYSTEM

A lot of the top level requirements have a direct impact on the Operating System (OS). This major component of the system should support a large range of computer hardware components. It should be autonomous and have a high level of robustness. It should also integrate real-time features.

A dedicated GNU/Linux OS has been developed based on a commercial distribution. This system is installed on each node of the clusters. The availability of the OS source code is a crucial factor to modify any component that does not satisfy the FTES needs.

On this Operating System, the main customisations allow:

- The removal of human administration on the computers. The system is fully autonomous, it can start/shutdown or be updated with no human intervention.
- The improvement of the global robustness of the OS. The system is "ageing resistant" and can handle erroneous actions from users.
- The improvement of the global reactivity of the system. The start-up phase is quick, the addition of standard extensions insure soft real time performance (low-latency, 1ms accuracy).
- The integration of low level software (to update or monitor the system for instance).
- The maintenance simplification with the removal of unnecessary services and components.

In order to be able to host Windows-based applications, two different interoperability solutions were implemented:

- A Windows emulator tool can launch compatible Windows programs in the Linux OS.
- For more complex applications which may require a whole environment, a virtualization product starts a full Windows virtual computer in which any native Windows application can run without any modification.

## 4.5 MIDDLEWARE

The highly distributed architecture and the cluster organisation easily enable the implementation of continuity of service solutions in order to handle hardware or software potential failure.

Airbus has developed and patented (see §6) a fault tolerance middleware for the FTES. At start up, this middleware launches all the applications on the cluster nodes. When a failure occurs, this layer can dynamically reallocate the software over the available resources. While running, the software execution contexts are exchanged between nodes. In case of reallocation the software restarts with its previous context.

The peculiarity of this middleware (compared to existing solutions) is the lack of master election; each node is identical and plays the same role than the others.

A predefined policy sets out the software allocation rules over the nodes. While running, each node sends a heartbeat over the network in order to inform the others about its attendance. By listening and analysing all these signals, each node can determine the complete configuration of the cluster and then apply the associated policy to distribute the applications.

An internal communication layer, closely linked to the fault tolerant middleware, allows applications to exchange messages regardless of their execution node.

From a user point of view, the software awarding policy defines the mandatory functions of the FTES. In case of failure, the reallocation process lasts a few seconds and the fallen services restart with the same context as before the failure.

## 4.6 GRAPHICAL CONCEPTS AND DISPLAY MANAGEMENT

The Graphical User Interface (GUI), displayed on the main screens of the station (§3.1), are mainly developed within Qt framework. As for the overall station design, the GUI should be ergonomic, easy to use and very intuitive.

Because of the system complexity and the large number of various programs to display (measurement parameters, specific aircraft views, video streaming...), simplifying the GUI is not so easy.

Indeed, one computer cannot run all these programs, so they are distributed over different nodes of the cluster, but displayed on a single server. This display server must hide the complexity of the architecture, in order to have a GUI as simple as one obtained on a single machine.

To achieve these goals, we have defined the following principles:

### One click, one fonction

On each GUI, a main desktop (Figure 7) presents to FTE the list of the tests programs sorted by aircraft disciplines (Engine, Load, Electricity...) and functions (Flight Display, Generic tools, Peripherals command, Video). The engineer clicks on one thumbnail to display a program on the related screen: it's simple « One Click One function ».

Many pages can be displayed on the same screen. For simplicity and general effectiveness reasons, all these pages are grouped into a container which allows the navigation between pages in a single screen.

The pages are submitted to some specific rules: most of them can be displayed on several screens simultaneously, except the ones which send commands to calculators or peripherals (to avoid control conflicts).



Figure 7: FTES Main screen

### Application display management

The pages are built on the same model. A page is divided into several frames which can be generated by different nodes of the cluster.

The display server acts as a “software multiplexer” to mix on the same screen these different layouts. The X11 protocol and client/server architecture allow this behaviour. General testing programs, system-monitoring, aircraft warnings, HD uncompressed video streaming, cockpit display screens are displayed with the same rules and technology.

For the overall harmonisation, all graphical contents must follow some global recommendations which define the common actions and the ergonomic rules for all the pages: behaviour on external actions, keyboard shortcuts...

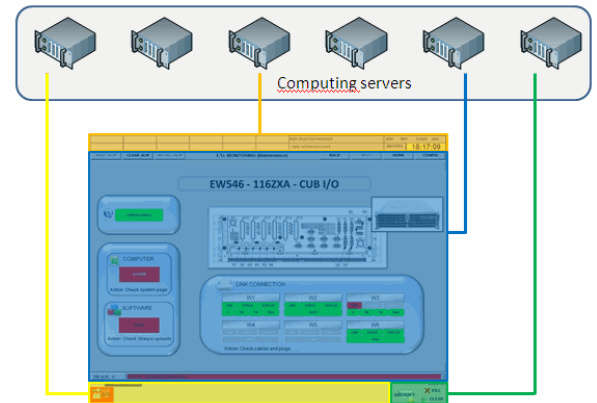
### User profile

A comprehensive set of pages allow configuration settings. Throughout the flight test campaign, FTE can work on several aircrafts as a dedicated page allows them to keep and share their personal profile on USB key.

### “User-friendly” touch screen panel

With the aim to gather several functions in a single system, a touch screen in the centre of the FTES is shared by both engineers. This device integrates some functions previously performed by hardware components (mechanical buttons, aircraft chronograph...). In order to ease the usage of this new system, the graphic rendering and the behaviour of the virtual components is exactly the same than the physical ones.

This equipment also communicates with the eight screens to perform snapshots and to propose in some configurations a virtual keyboard.



**Figure 8:** Application display management

## 5. CONCLUSION

The A350 FTES was designed from the A380/A400M FTES basis with a constant effort of optimisation and simplification: the overall architecture has been rethought in order to lower the number of computers while making significant power gains by taking benefit from technology improvements.

The major improvements are:

- A simplification of the general architecture with removal of unnecessary redundancies, fewer machines while increasing the overall reliability and computing power.
- A simple ergonomics with a larger share given to computer systems at the expense of electronic or mechanical controls.

This system has been designed and developed in 3 years and it has been fully integrated end of 2011. It will then be setup in the aircraft to host various flight test applications before being used to lead the A350 flight test campaign in 2013.

The architecture chosen has produced a reliable and robust system from standard equipments. More generally, this architecture can address the requirements of high computing power in embedded systems subjected to environmental stresses which are far from data centres ones.

## REFERENCES

- <http://ftp.rta.nato.int/public//PubFullText/RTO/MP/RTO-MP-SCI-162///MP-SCI-162-10.pdf> - Conduct of Flight Tests and On-Board Computing for the A380
- <http://www.patentgenius.com/patent/7693986.html> - Patent “Test flight on-board processing system and method”
- <http://www.scl.ameslab.gov/Publications/Brett/Interconnect.pdf> - Cluster Interconnect Overview
- <http://hal.inria.fr/docs/00/07/33/39/PDF/RR-3350.pdf> - Fault Tolerance Software Architecture