THE DESIGN, DEVELOPMENT AND INTEGRATION OF THE COMPLEX AVIONICS SYSTEMS

P. Schirlé
Avions Marcel Dassault - Bréguet Aviation
Saint-Cloud - France

Abstract

The improvement of the technologies during the past years have helped the digitalization and one can be sure that this trend will increase more and more ; this digitalization is characterized by the use of data buses and the growing part of software in equipment. With the advent of VHSIC, distributed processing, artificial intelligence, sensor data fusion, etc..., the technologies are blurring the clear functional allocation defined for the "black-box". These factors have important consequences on the cost of development and finalization. Moreover, the diversification (number of system's versions) and the growing complexity (systems more and more efficient, allowing an increasing number of missions) raises the problem of design and development concepts, and software products re-usability. To take these new trends into account, the AVIONS MARCEL DASSAULT - BREGUET AVIATION company has conducted important works, some of them jointly with French avionics manufacturers, to define a system design and development methodology and its associated workshop.

1 - CONTEXT AND HISTORIC ACCOUNT

Nowadays, military avionics systems reach a very high level of complexity and represent half of the cost of a modern military aircraft.

The evolution of these systems during the last two decades can be analysed on the operational, technological and methodological planes.

- The operational evolution is linked to the increase of the polyvalence of the systems which results in larger operational resources, and in a tighter and tighter integration of the functions within the same system and between airborne or ground based systems. (The System is then the sum of several airborne systems or ground based systems). Originally limited to the traditional operational functions such as navigation, air-to-ground bombing and/or air-to-air interception, the systems have gradually grown richer with a panoply of specific functions of sophisticated weapons, of electronic warfare devices or of reconnaissance equipment.

  The integration of these functions aims at obtaining a maximum operational efficiency by :

  . the optimization of the physical resources (sensors, actuators, data processing units) due to data fusion and to data exchange networks between aircraft and/or ground and maritime environments.

  . the optimization of the human resources, due to particularly well designed ergonomics of the man/machine interface, ensuring a high level dialogue with the pilots, the system selecting itself the data useful to each phase of the mission and presenting them under the most appropriate synthetic shape.

  The highly evolutive character of the systems is more and more confirmed. It must be possible to have the operational enveloppe evolve easily : introduction of new functions without modifying the operation of the former or improvement of the pre-existing functions through the technological evolutions.

- The technological evolution of the avionics systems mainly results in :

  . more and more complex functional and physical architecture integrating as well a new generation of sensors as historically independent systems such as engine, flight control systems, fuel or brake systems and generalizing the use of digital multiplexed connections between items of equipment.

  . massive introduction of software, bringing considerable flexibility and opening but leading to specific problems the mastery of which still proves nowadays difficult.

Schematically, this evolution presents three phases :

. Decentralized systems, not or slightly integrated :

In the sixties, the military avionics systems were made of specifically aeronautic equipment such as radars, air data computer or inertial navigation units, or instrument panel equipment, slightly interfaced through dedicated analog connections. Each equipment then represents a well identified operational function (flying, navigation, interception etc...), so that the functional analysis of the system is limited to the natural allocation of these functions to the corresponding black boxes and to the definition of the servicing interfaces between these boxes.

At the beginning of the seventies, software appears in these aeronautical equipment, improving their own characteristics. The design of the software is then approached within a frame restricted to each equipment, independently of the functional analysis of the system.

. Centralized and integrated systems :

A few years later, a new race of equipment, the "pure" computers appear in the architecture of the systems. These items of equipment, not attached to a particular operational function or to a resource linked to physics accept digital processing of the data, therefore software. They ensure, in a centralized manner, the integration and allocation of all the resources of the system with a view to a larger operational efficiency. Their software, called system software, represents an upstream functional layer with respect to the specific software of the equipment. It is also the result of a different design's procedure : its definition issues from the analysis of the entire system and no longer from the particular analysis of an equipment or of the function to which it is related.

. Decentralized and integrated systems :

The present tendency for the development of the systems is mixed : we see a decentralization of processing, the system software being notably distributed in a hierarchic way among several items of equipment, specific or not, but also an integration since each operational function is satisfied through a serie of functional modules installed in numerous equipment. Their development involves a particular functional analysis, leading to a functional architecture distinct from the hardware architecture, thus blurring the clear operational allocation of each box.

The airborne avionics systems developped at present by the "AVIONS MARCEL DASSAULT - BREGUET AVIATION" include more than a hundred equipment, half of which are strongly digitalized and the major part of which are functionally dependent on software. The volume of system software reaches several mega bytes, the number of data exchanged between equipment and/or functional modules exceeds 30.000 and the delivery of data on the data bus is of several mega-bit per second.

- The methodological evolution is the consequence required for the operational and technological evolutions aiming at keeping the mastery of the development of these large systems. The main axes of this evolution concern :

  . the system approach of the design
  . the quality assurance in design
  . the industrial organization for development
  . software specificities
  . software tools for development assistance.
  . system approach : from the first stages of the design, the former systems could be cut, a priori, into autonomous entities such as radio-communication, radio-navigation, flight control systems, engine control or such as radar, air-data computer, etc... each of these entities being the subject of a separate design. The integration, the decentralization and the complexity have modified the rules of the game. In fact, the design of the global system, which is up-stream of the equipment designing tasks, proves to be the most difficult and delicate. A new science, systemics, emerges, bearing on the recent functional analysis techniques and having its own methodology and tools. The practice of systemics (or system approach) involves a general knowledge of available technological resources with a view to their association within a system. Furthermore, it leads to a standardization of the downstream development methods.

  . Quality assurance in design : the importance and volume of the design work linked to the development of the systems, the increased difficulty of evaluating the quality (particularly the reliability) of the delivered product and the cost of the late corrections make the quality assurance activities rise from the manufacturing level to the design level. The quality of the product is more and more demonstrated by the qualification of the methods used for its development than through the product itself. Documents of quality assurance of systems or software are witnesses to this evolution.

. Industrial organization

The size of the present systems involves putting into common resources and competences distributed in numerous industrial companies (the number of persons involved in the development of an avionics system for MIRAGE 2000 aircraft exceeds 25 000 ...). Only a strict methodology can be used as a support for the new industrial organizations thus enabling :

* the definition of the task and responsibilities of all the persons involved.
* the ensurance of a harmonious development by reinforcing the visibility.

. The specificities of the software :

The specification of a software consists, from a requirement expressed in term of "usefulness", in refining during successive stages and according to a repetitive process, the written expression of this requirement until it is given a shape which can be directly interpreted by a software machine : the code. The production of a software only concerns its compilation and its reproduction. The design work of the software is of the same nature as the design work of the system, the whole being therefore the fruit of a continuous methodological procedure.

Consequently, the methodologies of development of the software will be coherent with the methodology of development of the system. Finally we must note that the specification work inherent to a software component of the system represents the sum of the specification work at every step of development (system then software), related to this component. This finding illustrates the non-obvious problems of ownership of the software !

. Software tools for development assistance :

The methodologies of development, of the software first then of the system today are supported by more and more numerous and more and more sophisticated software tools. These tools are assistance to the design and to the validation and are grouped in workshops.

The appearance of the software has led to defining and setting into place software workshops which are nowadays numerous and varied but bearing very neighbouring methodologies in their principles. The appearance of the system approach, more recent, creates the need of system workshops, covering the software workshops and ensuring the assistance for the upstream design of these systems.

2 - METHODOLOGY OF DEVELOPMENT, SYSTEM WORKSHOP AND SOFTWARE WORKSHOPS

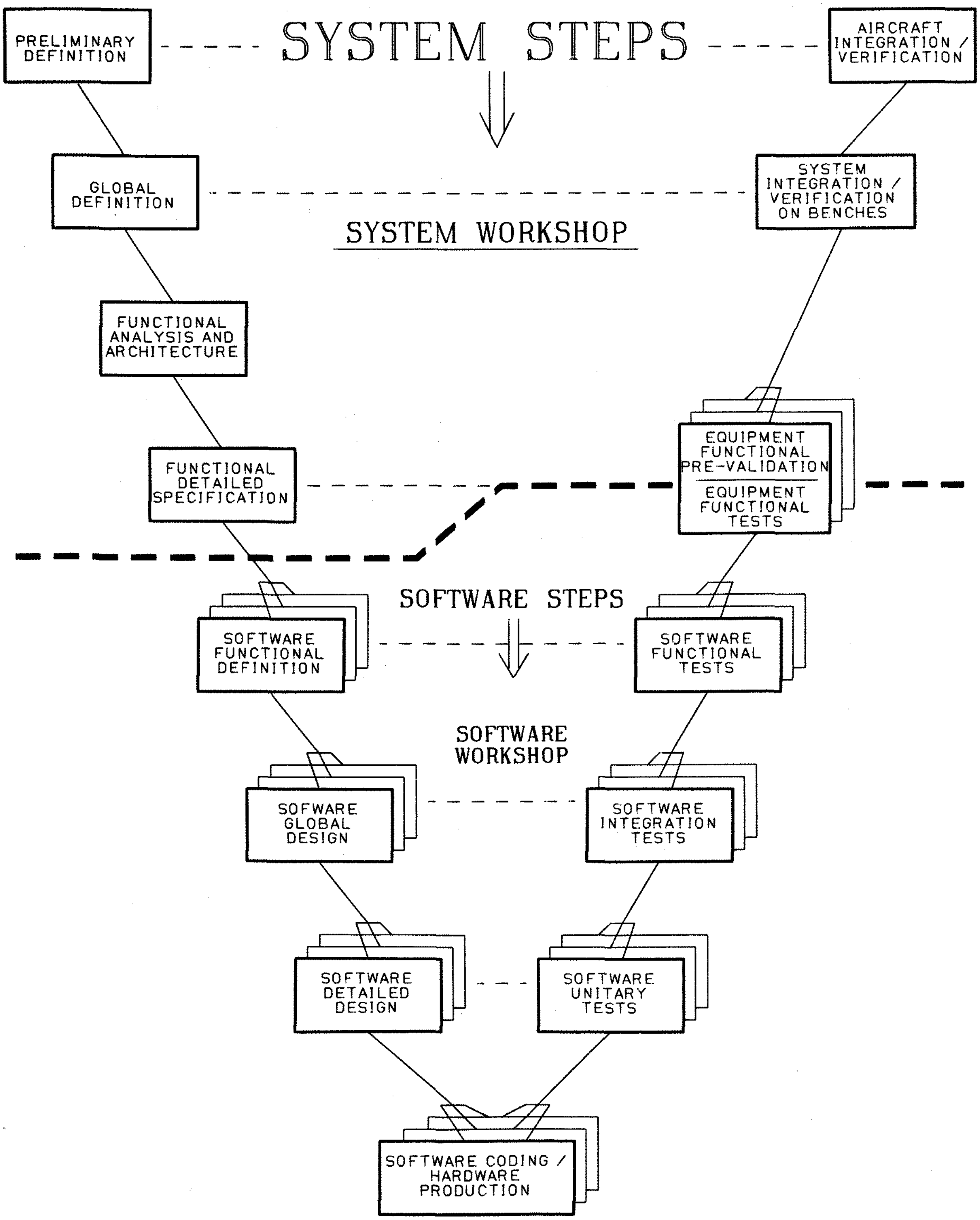
In this particularly evolutive context, the AVIONS MARCEL DASSAULT - BREGUET AVIATION company has been obliged, from its experience of these systems and by accepting heavy investments, to adapt itself quickly so as to keep its mastery of complex avionics systems :

- by defining an original methodology of development of systems.
- by defining and manufacturing the corresponding system workshop.
- by taking part in the definition and evaluation of the software workshops.

This methodology precisely defines the steps of the development with the tasks, products and means which are related. The design steps (descending branch of the "V" representing the life cycle) and symmetric steps of verification (rising branch of the "V"), thus including the methodologies proper to the development of the software are then described.

The following drawing sums up the different steps of the development cycle, from a system and software point of view. The system steps (above the dotted line) are unique and put under the system manufacturer responsability. The software steps (under the dotted line) are multiples and put under the different equipment or software manufacturers responsability.

AVIONICS DEVELOPMENT METHODOLOGY ABSTRACT



The system workshop is made of a coherent assembly of interconnectable, multi-user and multi-version software tools. It includes general tools (such as a powerful documentation word processing tool : SCRIBE or a tool for graphic specification of display entities : MITIA) as well as specific tools which will be dealt in the following chapter.

The system workshop includes a capacity of centralized configuration management through the SEFERIS tool which manages the versions of all the products made, through the tool with which they are produced.

Due to the natural increase of the relations between all the partners taking part in the development of the systems, it has seemed necessary to organize and reinforce these relations, more particularly for all that concerns the design and maintenance of the system and software.

The study made by I.T.I., which has assembled since 1983, the aircraft manufacturers (AEROSPATIALE and AVIONS MARCEL DASSAULT - BREGUET AVIATION) and equipment suppliers (CROUZET, ELECTRONIQUE SERGE DASSAULT, SAGEM, SFIM, SFENA and THOMSON-CSF) will enable at the beginning of 1989, the formalization of the relations between the partner manufacturers through the generalized use of the system workshop and of compatible software workshops, thus enabling obtaining a better quality for a lesser cost and profiting for the acquired competences and experiences.

3 - STEPS OF THE METHODOLOGY OF DEVELOPMENT

3.1 - Preliminary definition

The role of this step is to define the frame within which the system will be specified due to an analysis of the requested missions and a first study of the organization of the system.

The analysis of the missions bears on various feasibility studies and pre-studies of performance which will enable ensurance of the global feasibility of the system as concerns delays and operational performance. These studies are supported by a special tool : SAMOS enabling concepts validation by the operation of different models. It is concretized by a list of operational functions necessary to the production of the mission envelope, describes their possibilities of superposition or exclusion and highlights a set of needs concerning particularly the operating modes and the performance of the main sensors.

The study of organization of the system leads to a first definition of the hardware architecture of the system (list of equipment, and stores, geographical fitting-out of the cockpit, fitting-out of the bays, etc...).

The specific means used during this stage are the C.A.O. (CATIA) as well as various fitting-out mock-ups.

3.2 - Global definition of system

The step of global definition consists in defining precisely the usefulness that the system must have (and not how the system will be built !)

The result of the step is the description, in terms of operational scenario (therefore seen from the user) of the starting and of the nominal operation of the system for all the functions that it ensures. This description results in two types of documents :

- General rule documents :

  These documents describe once and for all, the utilization rules and philosophy applicable to every operational function, with which the system is and will be fitted. They guarantee thus the coherent operation of the system, relative to each operational function. Reception basis for the operational functions with which the system will be fitted, all these documents will be a reference for the writing of the global specifications and a starting point for the functional analysis of the system.

  Examples :

  . General rules for use of controls and displays : multiplexing criteria of the controls and displays, menus of the multi-purpose control keyboards, etc...
  . General rules of superposition of the operational functions : selection, superposition, exclusion, memorization, etc...
  . General rules for failure and malfunction signalization : levels and principles of signalization, coherence of messages.
  . General rules for maintenance
  . etc...

- Global specification documents

  Each global specification document describes the utilization scenario of the system with respect to a given operational function. Each operational function is therefore subjected to a specification which is written in compliance with the general rules. Since these specifications are independent, they can be elaborated autonomously and asynchronously. The coherence and the independence of the global specifications are ensured by the general rules.

The general rules and the global specifications are validated due to a tool for specification assistance setting into action the dynamic aspect : OASIS. This tool, built around a representative mock-up of the cockpit, enables the simulation of the sequences, controls and displays thus enabling a more effective validation of the scenarios by the pilots.

3.3 - Functional analysis and architecture :

The role of this step is to proceed to the functional analysis of the system and to deduce its functional architecture. The functional architecture file (product of this step) describes the solution fulfilling the needs expressed at the global specification stage.

The step consists of two phases :

- the construction of the functional architecture graph of the system.
- the integration of the functional architecture in the hardware architecture.

3.3.1 - Construction of the functional architecture graph

The method used consists in a progressive hierarchical breakdown of the system in functional elements, according to accurate criteria. Each successive level of breakdown involves :

- a justification of the breakdown made enabling the comprehension and approval of the selections and describing the constraints taken into account.
- a systematic collection of the induced interfaces between components of the architecture
- a progressive refining of the definition of these interfaces with respect to the breakdown level immediately above.

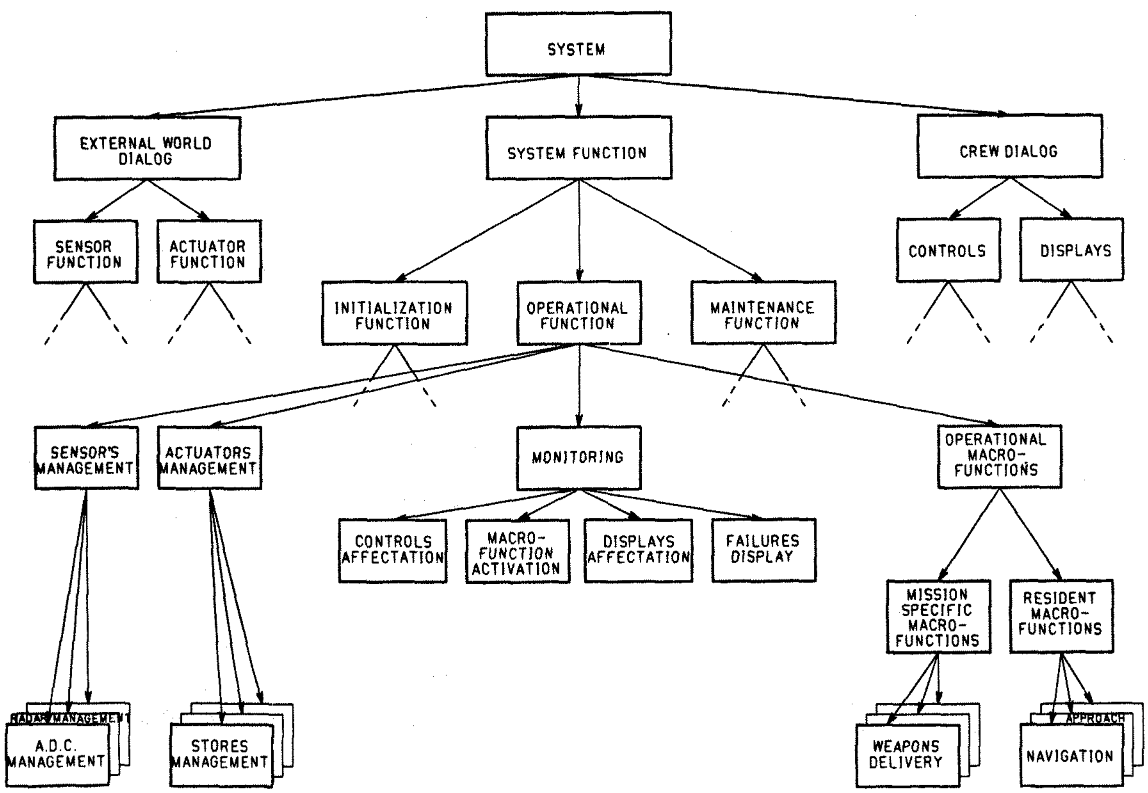
The functional graph describes the system according to a coherent arborescence. The group of the final components of the breakdown, called functional modules and their interfaces represent the functional architecture of the system. These modules will be specified then produced in hardware or software.

The constraints taken into account for the creation of the graph can be :

- quality constraints, particularly flexibility ; these constraints impose independence rules between functional modules and rules grouping tasks in the modules.
- operàtional constraints expressed in the general rule documents. Their analysis enables the disengagement of the "functional heart" structure of the system, group of particular modules the role of which will be to ensure the management of the other modules of the architecture.

The following drawing gives a uncomplete and non-realistic example of a system functional graph.

FUNCTIONAL ANALYSIS EXAMPLE



The definition task of the functional graph is assisted by a system design tool (OCS). This tool enables the assisted graphical construction, the coherent grouping of the interfaces at every level of breakdown and the identification of the functional chains (dependence of data).

3.3.2 - Integration of the functional architecture in hardware architecture

This phase consists in integrating the functional architecture previously defined and the hardware architecture suggested during the preliminary definition. The functional modules are then distributed in the equipment and identified (hardware or software).

This distribution defines the production compromise taking into account several factors :

- technological factors : according to the type of processing to be carried out, selection of the most adapted equipment.
- optimization of the connectics : we look for a distribution of modules minimizing the exchange flux between equipment and privileging "the shortest way" for the data critical from a real time point of view.
- particular quality factors : e.g. the modules bearing the critical functions from a safety point of view will be grouped in the same equipment and/or redundant in several items of equipment.
- determinant factors of savoir-faire or of industrial organization.

This task is supported by an architecture assistance tool : OEA, which enables, from functional data identified under the OCS tool, obtaining automatically the load of the data bus connecting the equipment, compared with a given distribution, and therefore optimizing this distribution of the functional modules in the equipment.

Due to the allocation of the modules in the equipment, the functional data centralized by OCS are broken down into three categories :

- the inter-equipment digital data which, due to a G.I.N. tool, will be processed and formatted so as to form the data bus messages and frames.
- the inter-equipment analog data which will be processed by a SINOPTICS tool so as to elaborate the synoptic wiring diagrams.
- the data exchanged between modules of an identical equipment.

3.4 - Detailed functional specification

The step of the detailed functional specification consists in producing the specification of the transfer functions of each functional module identified in the architecture.

Ideally these documents must contain "all that is necessary and only that which is necessary for the production of software and hardware" and therefore represent a contractual definition with regards to the producers. The modularity itself of this document (an autonomous document per functional module) induces a sharp production constraint, since the software architecture of each equipment must comply with the breakdown imposed by the functional analysis of the system.

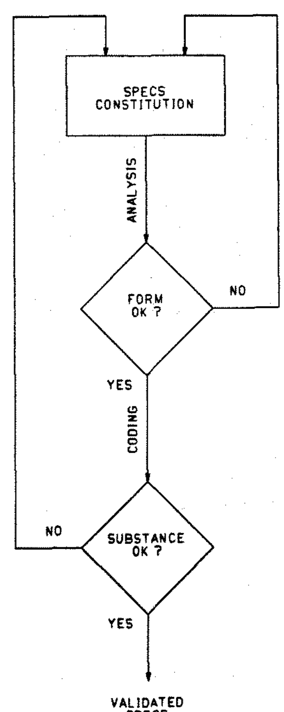
The detailed functional specifications are written in formal or semi-formal language by using a computer augmented specification tool : D.L.A.O. This tool enables the specification of the transfer functions from the collection of interfaces issuing from OCS, by using a proper language close to the natural language. Furthermore, it guarantees the coherence and homogeneīty of all the specifications.

The validation of the documents is made by the joint utilisation of the two techniques :

- a traditional technique of cross readings between various concerned persons
- a rapid prototyping technique of the specifications.

Systematically used for the critical modules, the rapid prototyping consists in a simulation of the functional contents of the specifications in a software environment.

It enables the verification of both the substance and the form of the specifications according to the following principle :



The specification writers can, due to the rapid prototyping tool, proceed themselves to functional tests of all or part of the specified system by using a battery of combinatory or random automatic tests, thus creating true operational scenario, played and re-played on the prototype.

3.5 - Development of software and hardware

The specific developments of the hardware and software are tasks giving rise to the stage of development of the system mentioned above. They do not enter into the frame of this document and concern the following stages of development :

- functional definition of software
- global design of software
- detailed design of software
- coding/production of the equipment
- unitary test of software
- integration tests of software
- functional tests of software
- functional tests of the items of equipment

Remark : the methodologies of software development from the system specifications as well as the corresponding software workshops can be slightly different according to producers.

Nevertheless, the industrial architect of the system must make sure of their adequation to the development methodology of the system by audits or through standards incorporated by each producer.

3.6 - Functional prevalidation of the equipment

The purpose of this step is to proceed to a separate functional evaluation of each developped equipment in order to be sure of a sufficient level of quality before the integration of the system.

The principle consists in unfolding functional scenarios at the equipment level to check the functions incorporated in the equipment and measuring various ranges of operation (writing rate for a display equipment, load of computation and memory load for a computer).

The test scenarios used can be issued from varied sources as :

- results of rapid prototyping of the specifications
- scenarios produced by means of the system integration bench (see para 2.7)
- scenarios recorded in flight (see para.2.8)
- Environment simulations.

The means used for the functional prevalidation are multiple : means of system developments, data processing means and specific or unspecific tools for environment generation.

3.7 - Integration and verification on system integration bench

The purpose of this step is to make sure, before integration in the aircraft, that the system is in compliance with its global definition, of the coherence of the successive configurations, and to evaluate the behaviour of the system through all its utilization domain (aircraft domain, case of failure, tolerance to failures etc...).

The integration benches enable the easy operation of the equipment forming the system and of the panoply of integration tools. They have :

- the group of aircraft equipment of the integrated system
- offset supports for the installation of sensors
- a wiring of the same type as that of the aircraft
- an electrical distribution identical to that of the aircraft
- analog and digital monitoring means
- digital and analog interface simulators
- access panels to the system data
- interfaces with a real time software complex
- realistic installation of the man/machine dialogue equipment

The environment of the airborne avionics system is re-created on the bench due to a software complex enabling the stimulation and simulation functions. These two functions aim at producing a realistic environment evolving with the parameters having the same dynamics as on aircraft.

- The stimulation consists in replaying on the bench a scenario recorded on aircraft so as to put the system in a state identical to that encountered in flight.

- The simulation consists in producing an interactive scenario, thus enabling the piloting of the system and studying its answers in all its utilization domain.

The integration benches nowadays enable proceeding with most of the verification and qualification tasks of the systems, preserving the flight tests for a few critical tests for which the bench is not sufficiently representative (particulary for physical environment).

Their advantages are their cost and utilization flexibility (compared to those of an aircraft), their availability, their ease of evolution, their present representativity and at last their ever increasing analysis power.

3.8 - Integration and verification on aircraft

The purpose of this step is to check and guarantee the operation of the airborne system. The tasks include ground tests and flight tests.

On the ground are carried out integration tests requiring the complete aircraft, maintenance tests, or electromagnetic resistance tests in an anechoic chamber.

In flight are carried out domain opening tests, store separation tests, complementary functional tests of the tests made on the integration bench and finally, particular operational evaluation on request of the aircraft manufacturer himself or his customers.

The means used are numerous and varied, namely but not exhaustively : prototype aircraft, test installations, data link devices, listening booths, real time software complex, mission preparation and read-out means and various ground support equipment.

4 - CONCLUSION

The complex avionics systems are nowadays subjected to constraining requirements as concerns quality and performance.

The registering and production of the targets appointed to the systems goes through the mastery of the development process.

This mastery is obtained in compliance with a strict methodology, continuously covering the whole of the development cycle and which is at the basis of every Quality actions.

This methodology is assisted by a panoply of tools forming the system workshop and the software worhshops.

The methodology summed up in this document has been progressively installed in the AVIONS MARCEL DASSAULT - BREGUET AVIATION design offices since 1982. Its entire incorporation through the use of the system workshop dates back to 1987 for the development of the "MIRAGE 2000 AIRCRAFT new cockpit" systems. The system workshop used in a multi-industrial context is at present installed for the development of the RAFALE D aircraft and the HERMES spatial shuttle, the systems of which will be made in co-operation with several aircraft industries.

5 - GLOSSARY

- I.T.I. : "Integration du traitement de l'information"
- SCRIBE : documentation facility tool
- S.E.F.E.R.I.S : "Suivi des Evolutions des Fiches Equipements Réalisant l'Intégration du Système"
- O.A.S.I.S. : "Outil d'Aide à la Spécification des Informations Systèmes"
- M.I.T.I.A. : "Moyen Informatique pour le Traitement de l'Imagerie Avionique"
- O.C.S. : "Outil de Conception Système"
- O.E.A. : "Outil d'Etude d'Architecture"
- D.L.A.O. : "Définition de Logiciel Assistée par Ordinateur"
- G.I.N. : "Gestion des Interfaces Numériques"
- SYNOPTICS : Wiring diagrams design tool
- C.A.T.I.A : "Conception Assistée Tridimensionnelle Interactive"
- S.A.M.O.S. : "Structure Adaptée à la Modélisation Opérationnelle du Système".