

Collaborative complex system design applied to an aircraft system

Eric Thomas, Michel Ravachol

Dassault-Aviation

78 quai Marcel Dassault Cedex 300, 92552 St CLOUD Cedex, FRANCE

eric.thomas@dassault-aviation.com michel.ravachol@dassault-aviation.com

Jean Baptiste Quincy, Martin Malmheden

Dassault Systèmes

10 rue Marcel Dassault, 78946 VELIZY-VILLACOUBLAY Cedex, FRANCE

jean-baptiste.quincy@3ds.com martin.malmheden@3ds.com

Abstract

Aircraft systems have evolved dramatically since the beginning of aviation. Many improvements of performance and safety have been made. Now each sub-system has optimized performance and it is thus difficult to find gains without breakthroughs in architectures or technologies; and this is the objective of the R & D studies towards a more electric aircraft.

Simulations are widely used to explore and justify aircraft architectures [1], but system simulations currently suffer from limitations which make them difficult to use for complex multi-systems analysis. Therefore tools and processes must evolve to accompany these major changes in order to support the designers in their quest of optimized design.

This article deals with new processes and tools which will take part, in a close future, in the determination, the verification and validation of systems architectures. The results presented here were obtained during the CSDL project (Complex Systems Design Lab) partly funded by the French government.

Keywords: Collaborative process, System engineering, MBSE, hybrid DAE, multi-physics, multi-levels, Optimization, Robust Design, Coupling Simulation System - surrogate Models, PLM/SLM integration

1 Introduction

Aircraft vehicle systems are typical examples of complex systems. They are composed of many sub-systems, which overall represent a set of thousands of equipment, and that have more and more interactions between them.

These sub-systems are provided by several companies for integration and must fulfill aircraft requirements.

The efficient study of performance and safety is of prime interest when designing complex systems in a collaborative context. At each stage of the design cycle, system engineers should be able to find optimized architectures of systems according to requirements. Such need is particularly important in the early stages of design when decisions on the aircrafts concepts, systems architecture and partners choice will determine the performance and the future cost of the product.

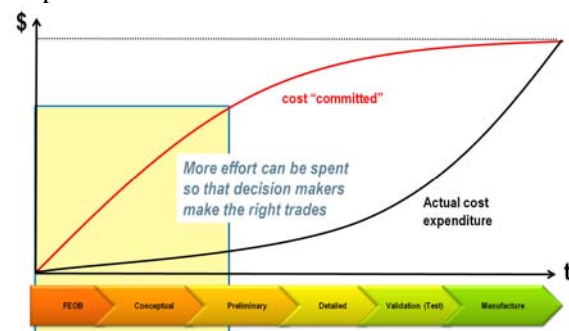


Figure 1 Design phases and effort ramp-up

It is thus necessary to make the right decisions in these early phases. With this intention, the systems architects may find it beneficial to explore spaces of design in a smart automated way in order to identify the points of interest quickly.

The article is structured as follows:

- Section 2 briefly presents aircraft vehicle systems and their design process.
- Section 3 details the requirements for a collaborative tool for complex system design.

- Section 4 explains the solutions developed within the project CSDL, in particular collaborative management of hierarchical multi-physics Modelica models with Dassault Systèmes V6 PLM platform.
- Section 5 presents the challenges ahead to get a full and efficient set of tools and processes for future airplane designs.

2 General information on Aircraft vehicle systems

2.1 Architecture

Aircraft vehicle systems are composed of several sub-systems. The main sub-systems are represented in the following composition (figure 2), here for a conventional aircraft architecture. There are Environment Control System (ECS), Power Plant (engines), Electrical, Braking, Hydraulics or Fuel systems.

They are connected together and to several other parts like control systems, passengers, environmental conditions or system properties [2].

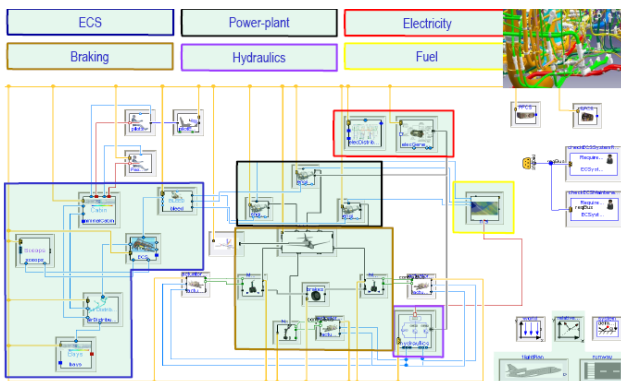


Figure 2: Systems architecture of a conventional aircraft

Each of these sub-systems is itself composed of sub-systems or equipment. For example the ECS is composed of a bleed (which mixes air flows from the engines), a cold air unit (which in particular manages cold and hot air flows to achieve a good comfort for the passengers in the cabin and a sufficient cooling for the equipment in the bays), a distribution sub-system (pipes and parts for distribution of air to cabin and bays), scoops (to get cold air from external environment ...).

The ECS is connected to engines (power-plants), passengers comfort models, and environmental conditions.

Traditionally, each aircraft system is defined within ATA (Air Transport Association) numbering,

which provides a common referencing standard for all commercial aircraft documentation e. g Chapter 24 is for electrical power or 21 for Air Conditioning and Pressurization. This standard has many benefits on common decomposition of aircraft functions, but tends to segregate sub-systems that may be optimized nearly independently from each other.

In this conventional architecture (fig. 2), electrical systems have only limited interactions with other subsystems. In the case of a more electrical aircraft, that is one of the most significant technology changes for the near future with many expected benefits, electrical equipment will be spread across multiple systems. Therefore, there is a consensus that the way to the truly optimized complex system is through an overall system redesign, including a trans-ATA approach.

2.2 Aircraft systems main activities and design process

The main activities of people involved in the aircraft vehicle systems are:

- System design and integration of the vehicle systems on the aircrafts.
- Follow-up, technical expertise and technical facts processing for the aircraft in service.

The activities are then not limited to design systems only, but also participates in the maintenance, improvements of the aircraft systems along the whole life-cycle (more than 30 years) and in decommissioning.

These activities include participation in the certification process of the aircraft which is necessary to allow the plane to fly; and for which product justification and traceability with respect to the requirements are mandatory.

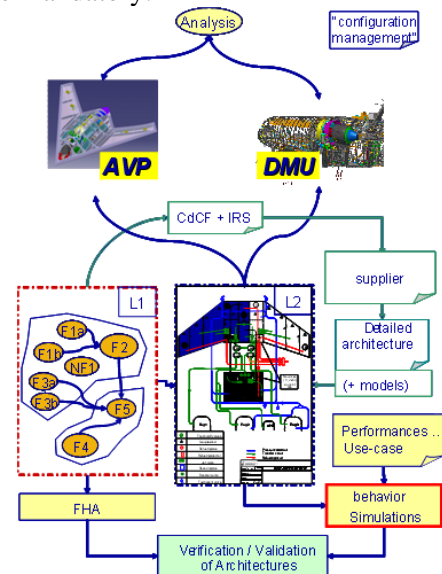


Figure 3: Several of aircraft vehicle systems activities

In the image above a typical workflow between the aircraft manufacturer and its suppliers and partners is presented. Several activities of the verification and validation process made during functional analysis, analysis which participates in justifications (e.g. FHA - Failure Hazard Analysis, behavioural analysis) are also sketched, connected to functional, logical and physical architectures.

3 Requirements for an aircraft systems design platform

Now, it is possible to list requirements for a truly efficient collaborative platform for aircraft systems design and optimization.

3.1 General requirements

The “must have” features of such a collaborative design tool can be listed as follows. They must allow:

- Compatibility with the tool managing the definition: currently 3D Digital Mock-up with Product Lifecycle Management.
- Project management during the entire lifetime of an aircraft (more than 30 years)
- Collaborative work between all stakeholders of the design of the aircraft systems.
- System engineering process: requirements, functional and architecture management e.g. standard architecture descriptions according to ATA decomposition.
- Several architecture analyses, in particular behavioral simulations, based on 3D and system representations.

3.2 Requirements on models for aircraft systems architecture

Model Based System Engineering (MBSE) is a key practice to advance complex systems development and Modelica is a critical enabler of MBSE

But system architecture analysis based on models must also be architecture driven because it is the architecture which must be justified and optimized. Simulations are means for architecture assessment. Therefore, it must be possible to add behaviors to components of sub-systems or directly to the sub-systems of an architecture.

Tools and models must also have several features as described bellow.

3.2.1 Tools for performance analysis

To evaluate the performances of systems architectures during trade-off, analysis based on simulations are widely used, from simulation of 0-D/1-D models to multidimensional models (FEA/CFD...) for more detailed analysis. The architecture components should thus be able to have models with multiple levels of details, chosen according to needs.

In fact, designers should have all models needed to model the behaviors they want according to the types of analysis that are to be done. Current analyses are listed bellow:

- Static analysis for study of energy balance, energy flow distribution or of particular design points
- Dynamic analysis to study analysis along time, or Eigen values.
- DOE (Design Of Experiments) including sensitivity, robustness and optimizations analysis.

And this must be applied on models with nominal and non-nominal behaviours (e.g. when failures occur).

There are also requirements on simulations management, because simulation properties, models, results must also be stored and managed to be usable many years later.

3.2.2 Libraries of models for system engineering

What kind of features would a system engineer like to find in the application libraries?

System engineer wants to have a set of models able to represent the behavior of physical components with a sufficient accuracy for the kind of analysis he/she has to do, and to focus on technical subjects in the way to chose and optimize systems. Then, system engineer would like to find:

- Multi-domain and multi-physics libraries of components for the large range of physics implied in the aircraft systems.
- Versatile components whose physical properties can be parameterized according to product data sheets or with data linked to definition (managed by the PLM.)
- Application libraries with validated components should be valuable, if not essential. Validation in a defined range of application is very important, because it is the base for the re-use and extends of components (which contains knowledge of the company).
- Switches to enable a model validity checker or not. Supposing that a validity model is already defined (see properties [2]).

- Switches to define physical hypotheses: consider static or dynamic behavior, nominal or non-nominal behaviors.
- Published additional data which can help to set simulations. For example stochastic data are often added to models afterwards by system engineers. It is not logical that such information is not usually included in models provided by partners. In fact they are the best specialists for publishing such useful information at the right places in the models. A general mechanism for publishing such data should be studied to enable this process.

3.3 Requirements for model interfaces and model exchanges

To allow connectivity of models (equipment or sub-systems), it is important that standard interfaces are defined, and that more complex interfaces could be derived from them. These standards must be applied by all partners, and managed like other interfaces.

The tool shall manage:

- IP for model exchange (integrate models of partners, provide to partners system models).
- Interface between sub-systems. In particular it must allow change of components (sharing a particular interface) as defined below in the application example when surrogate models may change.

After decomposition in black or grey boxes, simulations of systems should remain efficient (see requirements below.)

Functional Mock-up Interface, FMI [7].) can be used for encapsulation of Modelica models and other model code as soon as it respects previous requirements.

3.4 Requirements for simulations

For early verification of an architecture, quick evaluations based on thousands of simulations are required to explore the design space. Therefore, system simulation is often used because it is far quicker than 2D-3D FEA or CFD computations. They are used to find robust and optimized designs by use of sensitivity, robustness and optimization process. They must also take into account variability of architectures, parameters defined as a range or as a stochastic distribution. It is also necessary to be able to increase granularity of certain equipment models that has proven particularly sensitive or to incorporate new observers only available in detailed models.

Tools often allow co-simulation between 0-1D models and 2D-3D models. However, it is not really adapted to early verification because they can lead to slow simulations, which are often not compatible with efficient optimizations processes (with several parameters to optimize and having multiple criteria) which require a large number of computations.

Computation time is critical because simulations must be feasible within time constraints to get results, analyze them and choose the optimized architecture with a good level of confidence often after several interactions. It is also important to have sufficiently fast simulations in order to make early decisions and explore alternative architecture designs during a decision review. To allow such quick calculations High Performance Computing (HPC) features, parallel computations, and distribution of simulations on adequate hardware are other key factors.

3.5 Requirements for model debugging

The previous sections suppose that models simulate without problems. But it is well known that complex systems written in a natural physical language such as Modelica often gives sets of hybrid Differential Algebraic Equations with non-linear equations that can be difficult to initialize and solve.

Even if Dymola and DBM, the Dymola kernel integrated in Catia V6, is very efficient; performance and convergence of the initial problem also depends a lot on the quality of the code written by the author of the model as well as the how well the iteration variables of the initial problem have been/can be set. Features like the homotopy operator [5] help the user to solve initialization equation systems by providing a simplified model requiring less start values of iteration variables of the initial problem. However, it is important that such features could be used both by model developers and by final advanced users (see published properties and features in next section).

It is also important that the simulation tool help users to localize the cause of problem. Many features have been introduced in Dymola. Following new features can help:

- More (visual) features to quickly locate important information (e.g. component highlights, model comparisons ...).
- Structural analysis to study architecture of models to localized ways of simulation improvements (causality, algebraic loops, invertibility ...).
- Other methods will be studied in the near future [8], in particular Modelica models with structural changes and non-nominal behavior integration.

4 Application to an aircraft system

Investigation of these problems for very early stages of design have been done within the project CSDL, which had the objective to develop a comprehensive collaborative environment for decision making at the earliest stage of a project.

It tries also to take into account that process and associated tools must help designers along all the lifecycle of an aircraft, from early stage to operational service, including justification to requirements traceability.

It is applied to the design of an environmental control system.

4.1 Description of the system

An environmental control system (ECS) was selected because it combines several demonstrative features which can be applied to other systems afterwards.

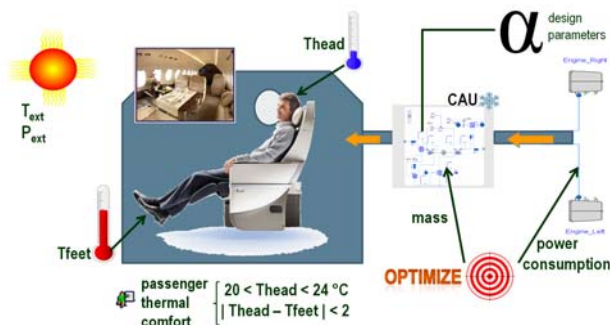


Figure 4: ECS Sizing engineering problem

For this reason, a generic model of ECS was previously used as a base during ITEA2 Eurosyslib [6] for properties modelling (see [2]) and will be used to enhance several modelling features during ITEA2 Modrio [8]. In CSDL it is used to investigate multi-level modelling and collaborative design.

This generic model is a 0-1D model written in Modelica. It is composed of basic sub-systems. Air flow comes from two engines modelled as boundaries with fixed pressure and temperature. A bleed mixes the two flows and provides the resulting flow to the Cold Air Unit (CAU) which regulates mass flow and energy given to the Cabin. Usually the energy flow rate coming from the CAU is provided to the different parts of the Cabin and to the Bays through a complex piping system. In this example, only a Cabin is taken into account.

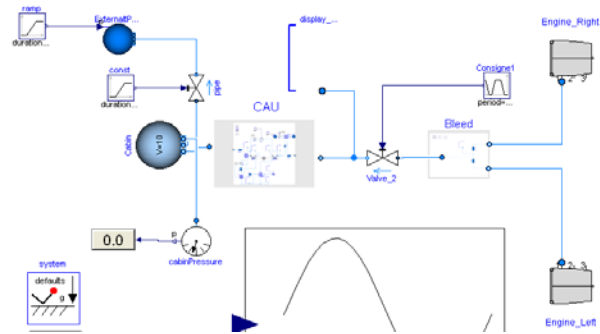


Figure 5: Generic ECS

The CAU is composed of a compressor, a turbine, heat exchangers, pipes and a regulating valve controlled by a PI controller which uses the measured Cabin temperature and a temperature set point for the regulation, as shown in figure 6.

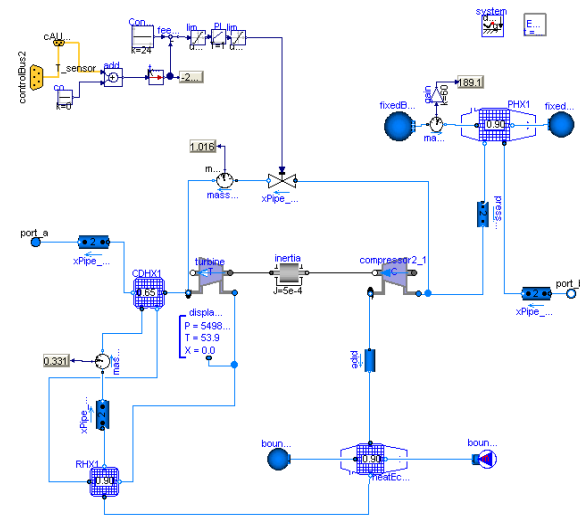


Figure 6: Cold Air Unit

4.2 Surrogate model

For rough assessment, a cabin modelled as a volume or some combination of volumes and heat wall exchanges may be sufficient. But, for more detailed insight, in particular for passenger comfort, it is more suitable to calculate the air flow in cabin using CFD codes. A usual method is to co-simulate the two models. For assessment based on small number of calculations, it is possible to do this; but optimizing the system may require too many simulations to be run.

As for passenger comfort optimization, where insight of only a couple of variables in the cabin are required, it is better to build a reduced surrogate model from CFD and optimise the system using it as described in figure 7.

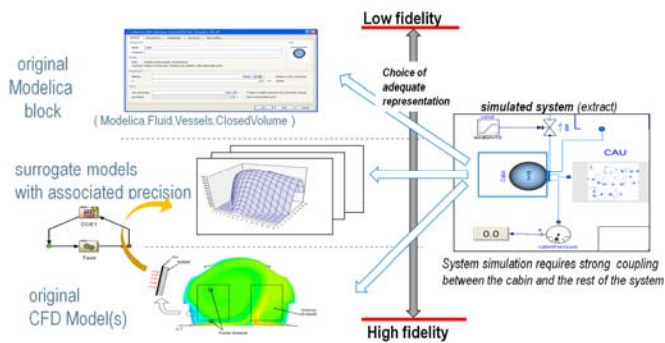


Figure 7: Cabin modeling options

Several types of surrogate models can be used to approximate the CFD response, RBF (Radial Basis Function) being one of them. A surrogate model is a parameterized function. In our use case, inputs are temperature T and velocity u of the air injected into the Cabin, plus external temperature T_{ext} . Output are temperatures at several selected points in the cabin: T_{feet} , T_{head} and T_{sensor} which are temperatures around passenger feet and head, sensor used for temperature control feedback.

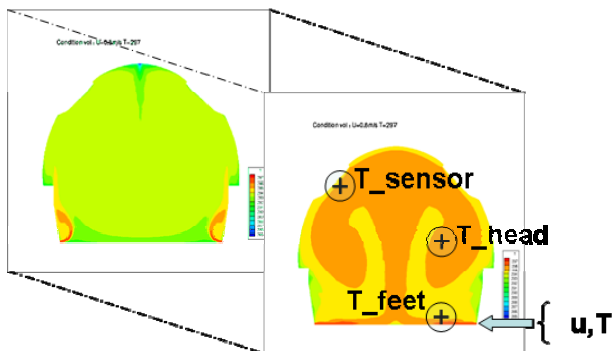


Figure 8: Surrogate model inputs and outputs

The function is expressed by a mathematical formulation that is parameterized by a set of weights. These weights are computed so that the surrogate model matches the CFD response.

4.3 Surrogate model integration in Modelica

To integrate the new model, we need to modify the interface between the CAU and the Cabin to define a common interface that is usable for a number of models both Modelica native and imported ones.

Exchangeable models are declared as replaceable and constrained by the specified base model which manages the interface connections compatibility to other sub-systems. It is done in a similar way as made in the Modelica library called VehicleInterface.

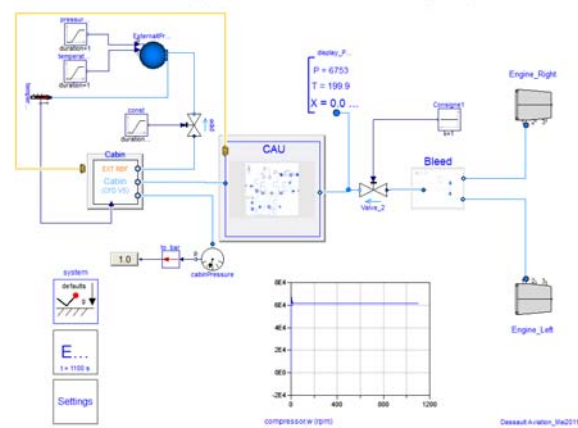


Figure 9: Modelica model with replaceable components

Therefore, when implementing the system template with a new Cabin representation, only models having a compatible interface are proposed to the user, as shown in the following figure.

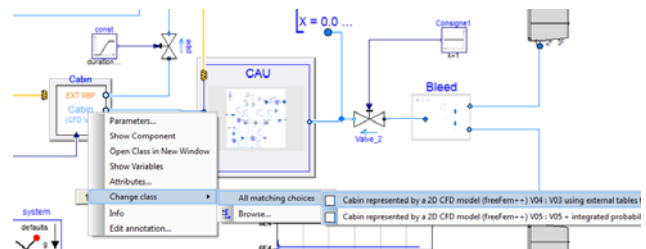


Figure 10: Interchangeable Cabin models

Internal parts of compatible models are then defined as can be seen below:

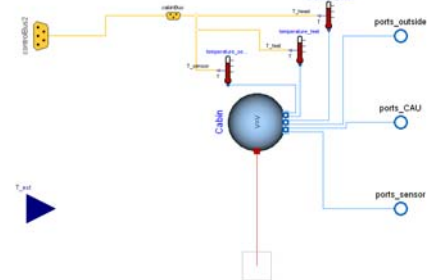


Figure 11: compatible interface with a Volume model

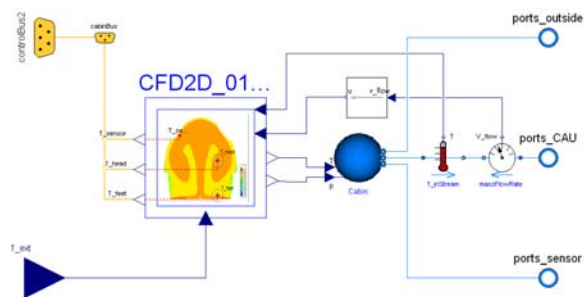


Figure 12: same compatible interface with a CFD model

Models allowed to be used can be both native Modelica models, reduced models such as RBFs implemented in Modelica but reading data exported from Isight at runtime, FMU's or other. This ap-

proach shows how a flexible common system structure can be defined using the redeclare/replaceable constructs to allow simple configuration of a large number of architecture design alternatives incorporating different levels of granularity and origin of the subsystems depending on what is the subject matter.

4.4 Stochastic distribution in Modelica

Stochastic properties of parameters used for analysis like robustness are often added to model afterwards when needed. Such properties should be associated to the model by the company that provides products or sub-systems; Attempts to define standard definition of uncertainties have been done, e.g. as expressed in [4]. But it is not yet standardized, even if it should be.

We have then tried to add these properties in a way that will be easy to use for adding such metadata in existing models. It is done by defining base classes for distributions and extending the models with these base classes (here adding a tab in the Dymola graphical interface with additional parameters for probability distributions.)

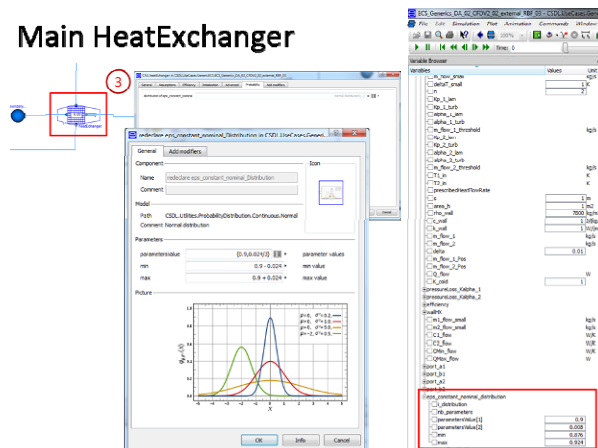


Figure 13 Stochastic data definition within Dymola

Such meta-data should be managed by the tool with publishing mechanisms. The following figure shows an Isight workflow where these stochastic properties defined within the model are mapped in order to be reused in a robustness analysis.

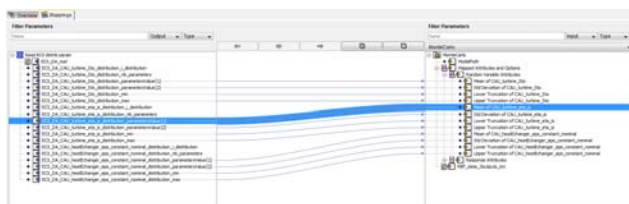


Figure 14 Stochastic data extraction

4.5 Design process

During the design process, several activities must be carried out. Only main ones are presented. The purpose is not to be exhaustive, but to show workflows and illustrate what must be done and how it could be done. These activities are iterative and must create formal links between architectures and products with valid requirements (see [1]). They are also collaborative (see next chapter).

4.5.1 Engineering Requirements

Passenger thermal comfort should be guaranteed for a whole range of operating conditions. Some specific operating conditions corresponding to external temperature extrema have been chosen as dimensioning test cases.

Moreover, several objectives have been set: minimum mass for the system, minimum mass flow rate extract from the engines.

Among all design space parameters of the model, several parameters have been selected: turbine efficiency and nozzle area, main heat exchanger efficiency

More types of requirement for an aircraft ECS may be found in [2].

4.5.2 Functional analysis and Logical architecture

A simple decomposition of the functional and logical views are presented in next figure. The functional view represents what the system should do, and the logical view represents how it is implemented. The logical view shows here that Engines and ECS are parts of two different ATA (ATA 71 for "Power Plant" and ATA 21 for ECS, exactly "Air Conditioning and Pressurization")

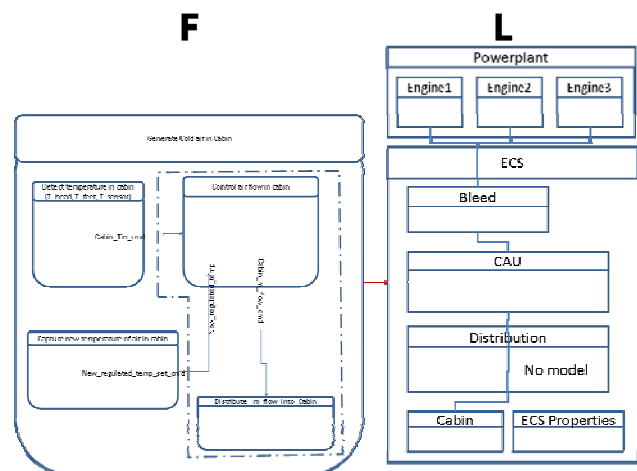


Figure 15 Functional and Logical Views

4.6 Collaborative process

To manage systems and build previous models, specific skills are required. Several actors may interact in aircraft manufacturer units or in partner companies.

4.6.1 Actors

To study the collaborative process, several actors have been identified and defined in the following table:






Actor		Role description
Project Lead		Responsible of the planning the engineering activities within the project
Architect Aircraft		<ul style="list-style-type: none"> - In charge of aircraft architecture - Specify targets for subsystem and arbitrates trade-offs between subsystems.
Thermal Architect		<ul style="list-style-type: none"> - In charge of architecture for the air-conditioning ECS subsystem - Studies alternatives and optimizes the design
CAE (CFD) Analyst		<ul style="list-style-type: none"> - In charge of CAE based studies in his field: thermal studies - CFD multi-D calculations - Surrogate models generation
Method Engineer(s)		Provides a set of prepackaged simulation services that can be used by CAE analysts, system architects and specialists

Figure 16: Set of involved actors

4.6.2 Collaborative Workflow

A workflow describing the engineering process has been defined, as shown below.

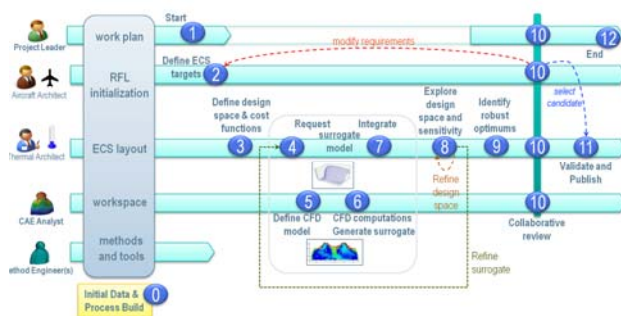


Figure 17: Collaborative process

Some of the steps are supported by a simulation services automated in Isight, as described below.

4.6.3 Design process

4.6.3.1 Sensitivity analysis

Sensitivity is the first analysis performed on a model. It helps identifying important parameters to focus on, and parameters on which tolerances may be relaxed.

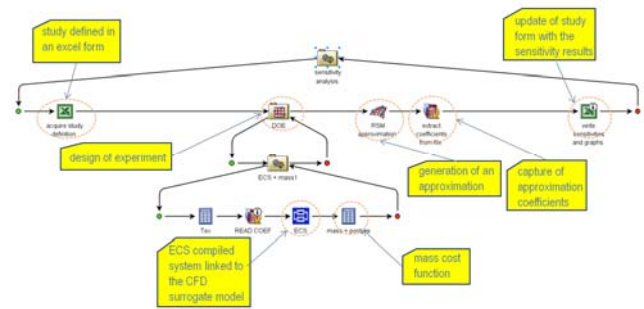


Figure 18 Isight sensitivity analysis configured by a spreadsheet

4.6.3.2 Optimization

The final aim is to produce optimized systems according to multi-objective requirements. It is then an important activity among all design activities.

4.7 Leveraging V6 RFLP

As we mentioned earlier, efficient collaboration between stakeholders is a key ingredient. V6 CATIA Systems enables such collaboration by:

- Providing a unique data referential for requirements (R), functional decomposition (F), logical product definition (L) including 0-1D models (cf. lower part of picture 20 showing the ECS), physical product definition (P) including CAE multi-dimensional models.
- Tracing dependencies of these data through implement relationships (cf. right hand side of picture 20 showing implemented/implementing relationships thru the R-F-L-P cascade),
- Tracing additional dependencies by capturing data flow of simulation processes (detailed in next section).



Figure 19: Mapping of use case data to V6 data referential (RFLP and Simulation)

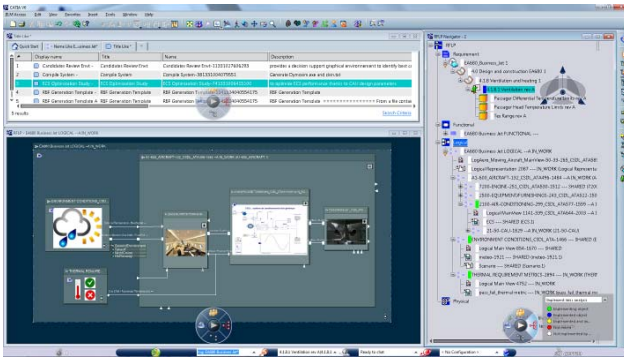


Figure 20 Thermal Architect V6 cockpit: ECS RFLP (right), ECS system (bottom) and design exploration services (top)

In this way, not only each stakeholder can manage the lifecycle of his/her own data properly but also have access to the data published by other stakeholders and author his/her data in this context. Out-of-sync situations can be properly detected in the case some upstream data is revised.

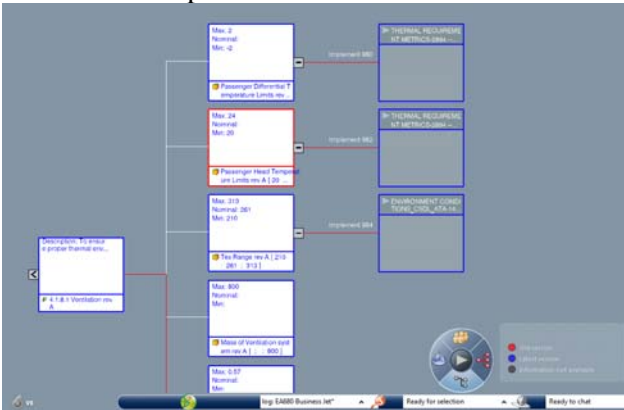


Figure 21 Compass showing that a system reuses an old version of a requirement parameter (outlined in red).

4.8 Providing on-the-shelf services for the Thermal Architect



Figure 22: Services to the Thermal Architect

Through its process integration capability, Isight enables Method Engineers to build automated simulation services intended to the Thermal Architect and CAE analyst. Complementarily, simulation data management capabilities of SIMULIA V6 Scenario Definition module are used to manage the lifecycle of these services and to deploy them within the en-

terprise. Moreover, it will manage the data relative to each usage of these services.

These simulation services are intended to be generic enough so that they are applicable on a class of design problems, such that, once a service is published by the Method Engineer, this service can be used on different designs without requiring rework by the consumer of this service.

After instantiation by the end user, the V6 impact graph functionality will enable to completely trace the data flow of the simulation data produced by these services, so that the end-users will be able to understand which data contributed to the generation of a particular data. The example below shows the dependency of an optimized design candidate on:

- the parameterized system architecture (data created by the Thermal architect)
- the CFD model used to generate the surrogate model that is fed into the 0-1D modelling (data created by the CAE analyst).

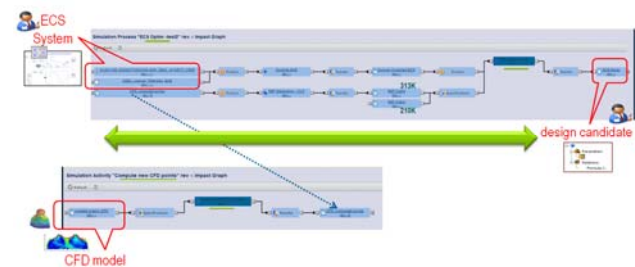


Figure 23: Traceability

4.9 Parameter management

PLM parameters can be defined within the requirements by the Aircraft Architect and reused e.g. within the Logical system by the Thermal Architect.

These PLM parameters, which have a lifecycle of their own and are likely to be revisioned, can be used to publish requirements characteristics such as expected Cabin temperature range (e.g. between 20 and 24°C), range of operating conditions (e.g. external temperature between -60 and 40°C) that the aircraft can be exposed to, as well as performance targets (e.g. maximum mass).

These PLM parameters are then available downstream in the R-F-L-P cascade, and can be used locally to valuate Knowledgeware parameters that parameterize applicative V6 data like CATIA System Modelica models.

Reuse of Knowledgeware parameters in the Modelica models creates links between parameters in the Modelica models and other data in order to ensure consistency between teams of different engineering disciplines that normally do not have much direct

interaction. An example that we show is how a parameter from the requirements like the external temperature range is reused to drive the values of the external temperature within the alternative models of the Environment of the ECS.

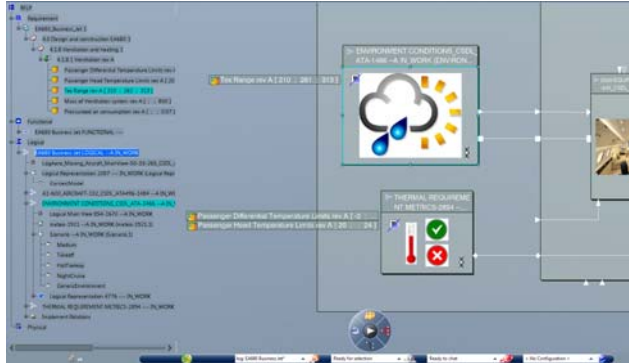


Figure 24: Parameter Flow from Requirements to System

4.10 Decision support interactive environment

In order to identify the design points of interest and to be able to compare these design points, there is a need for a graphical environment that is able to show two complementary views of the engineering problem (cf. figure 25): the analytical view focusing on the performance and constraints (cf. figures 26 and 27), and the behavior centric view that shows, for a specific design point, the associated simulation results (0-1D, CFD, etc...) showing how the virtual product behaves.

This graphical environment is fed with the results generated by design exploration processes mentioned in section 4.8 and is itself packaged as a service to ensure efficiency, consistency and traceability, quite important characteristics for the decisions that will be taken using this environment.

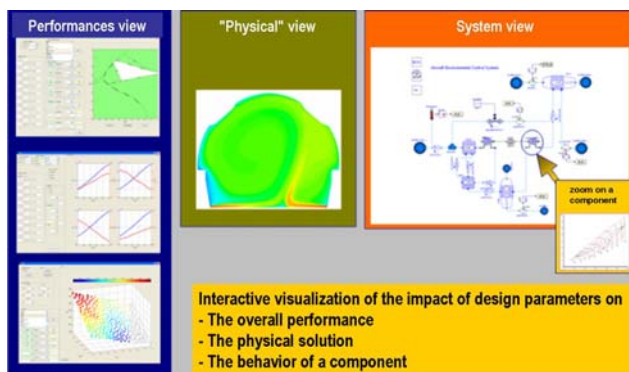


Figure 25: Graphical Environment for Decision Support

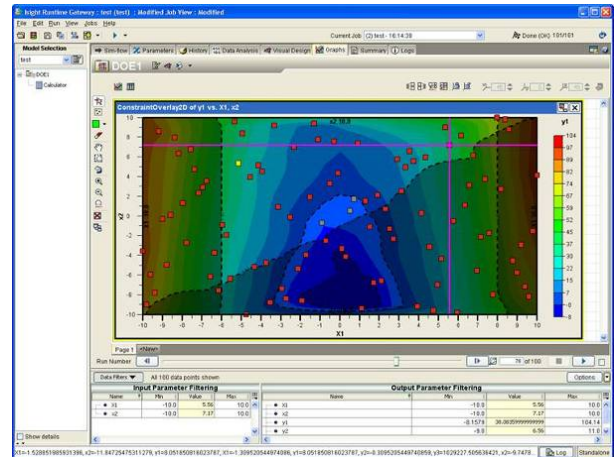


Figure 26: Decision views

In addition, using surrogate model it is possible make interactive request offline. For instance, the feasible domain can be interactively visualized for any combination of design parameters and constraints (cf. figure 27)

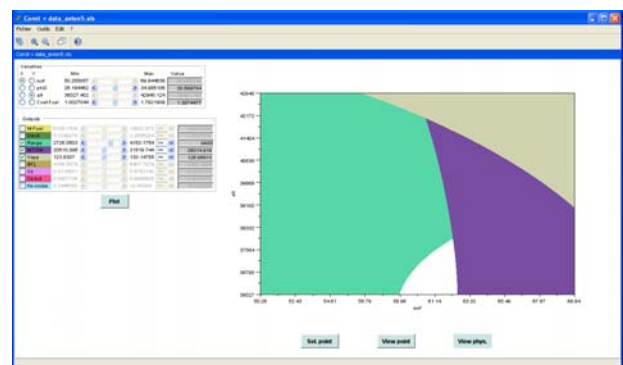


Figure 27: Interactive feasible domain

The ability to performed interactive analysis is a major towards performing an interactive “what if analysis”.

5 Conclusions

In this article, we have tried to sum up what should be a truly efficient tool for aircrafts systems design. A lot of work has been done to obtain a cutting edge tool which includes system management in a PLM framework.

The purpose is to help designers to focus on important problems in a more and more complex context by providing smart tools that allow them to perform their task more efficiently.

For system simulation, Modelica is a key factor. Many enhancements of the language have made it the leading modeling language for physical modeling. Last but not least is the new integration of synchronous semantics in Modelica 3.3 which allows state of art modeling of control systems and digital electrical systems.

Modelica is spreading rapidly in aerospace applications. Even if the language is much more efficient than other languages, there are still some challenges to have efficient simulations involving large hybrid models of complex multi-systems architectures. Most of these challenges will be studied in the next big European project MODRIO (Model Driven Physical Systems Operation).

Acknowledgements

This work was partially supported by the French government through the Systematic CSDL project.

References

- [1] Advances in Product Modelling and Simulation at Dassault Aviation; Lionel de la Sayette; RTO AVT Symposium, April 2002.
- [2] Modelling of System Properties in a Modelica Framework Audrey Jardin, Daniel Bouskela, Thuy, Nguyen, Nancy Ruel (EDF R&D), Eric Thomas, Laurent Chastanet (Dassault-Aviation), Raphaël Schoenig, Sandrine Loembé (Dassault-Systèmes), Modelica Conference 2011.
- [3] Projet CSDL :, Conception en phase amont de systèmes complexes sur la plateforme PLM/SLM de Dassault Systèmes Michel Ravachol, Jean-Baptiste Quincy, NAFEMS Paris, june 2012.
- [4] Modelling of Uncertainties with Modelica Daniel Bouskela, Audrey Jardin (EDF R&D), Zakia Benjelloun-Touimi (IFP Energies nouvelles), Peter Aronsson (MathCore

Engineering), Peter Fritzson (Linköping University) Modelica Conference 2011.

Project Sites

- [5] www.modelica.org
- [6] www.eurosyslib.org
- [7] www.modelisar.com
- [8] www.ITEA2.org / Modrio

