BENEFITS OF INTEGRATION PLATFORMS IN PRELIMINARY AIRCRAFT DESIGN

By Ronald M. Finke, Steffen Schäufele, Boris Wonneberger, Christophe Blondel, Chris Lynas, Henry Leach, Marylene Duffau
Airbus Operations

Abstract: An important step in the development of a new aircraft is the preliminary design, which determines the configuration itself and key parameters about cabin**,** fuselage, wing, landing gear, control surfaces and engines. The design has to fulfil a certain number of constraints, but also to optimize a given cost function. Therefore the engineering team has to investigate a wide range of fundamentally different design alternatives and provide numerical results by using a tool-chain of multi-disciplinary *and* multi-level methods. Thus it is necessary to have an efficient and robust way of design space exploration, without losing the accuracy of the investigations. In 2008, Airbus Future Projects identified Isight as the most promising COTS solution as successor to its existing platforms. Since then additional modules and capabilities have been developed by and under the lead of Airbus. Main improvements deal with design space visualisation, handling of large sets of parameters, computational geometry and collaboration. The main focus of this paper is on design space visualisation and collaboration.

Keywords: Aircraft Design, Multidisciplinary optimisation, Integration Platform

1. Preliminary aircraft design task

Out of continuous market survey and long-term company strategy a set of top-level aircraft requirements (TLARs) is established for a new air-transport product. These usually include the number of passengers, the design range, the cruise speed and requirements for the operational performance.

Based on these TLARs, air-traffic and airport rules, the preliminary design group will produce families of different aircraft configurations, which all fulfil the given requirements. This phase is mostly based on existing aircraft, semi-empirical methods and 2D drawings. Parametric low complexity 3D models are used as well to study component integration, operational aspects and emergency conditions.

Within the next phase, several trades are performed to optimise all configurations individually with regard to the given cost-function. In addition, design robustness have to be looked at with care, too [1].

At the end of the preliminary design phase, the best configuration will be selected and the key parameters about cabin**,** fuselage, wing and engines will be used for more detailed investigations about structures, flight physics, systems and manufacturing process.

Labour cost for the preliminary design itself accounts only for a small fraction of the total costs of the aircraft project, but it very much defines the total manufacturing and operating costs of the final product. It is thus very beneficial (cost-effective) to improve the preliminary design stage (eight more numerous or more detailed for a given amount of study time, or lower study lead time for quicker decision making)

2. Integration platform

In 2008, Airbus Future Projects began a market survey to find a successor for several integration platforms, which had been used so far. Therefore it was chosen to explicitly analyse all so far established manual, semi-manual and computer-driven design processes. Key properties were found to be: iterative, inverse, knowledge-based, multi-granular, multi-disciplinary, trans-national with a large & growing set of data and drawings.

To keep Airbus' ability to create world-class A/C designs, these findings were translated into a criteria catalogue for the to-be-COTS solution. A team of engineers thoroughly investigated a large number of available solutions. In parallel, communication was established to end-users, management and internal IT-specialists. After 6 months, Isight was identified as the most promising solution.

However, some additions and improvements were identified to be necessary: 1.) Design space visualisation with contour and carpet plots, 2.) Improvement and customisation of the report function, 3.) a way to reverse the all or part of simulation flow, 4.) more ways to undo user inputs.
All these inputs were assembled into a business requirements dossier, which is still used in the technical discussions between Dassault/Simulia and Airbus.

3. Design space: Constraints, sampling & visualisation

Due to the iterative nature and the uncertainties involved in the preliminary design process, straightforward application of optimisers is usually sub-optimal in finding the best aircraft for a given set of TLARs. The overall aircraft design engineer is rather interested in understanding the heavily constrained design space around the initial design [2].

Since 2008, various teams built a number of different aircraft models with different complexities with Isight. Full factorial design-of-experience (DOE) runs were launched for a systematic design space visualisation. Using state-of-the-art CATIA PCs, run-times of up to three hours were observed.

As a result, more detailed software requirements concerning the post processing of the result data were established. For example: The post processing of the result must be independent of the boundaries defined in the design model. It must allow interpolating within the result data, extrapolating beyond them and allowing to change the constraint line values, all without the need for additional executions of the model.

To visualise the design space, ***contour plots*** with constraint overlay are appropriate when using two design parameters (e.g. Wing area and span). That is essentially a 2D chart with the design variables on X and Y axis. The target function is displayed as labelled ISO-lines. The area of valid designs is created by a Boolean overlay of multiple dependent variables, which are limited by min/max values individually.

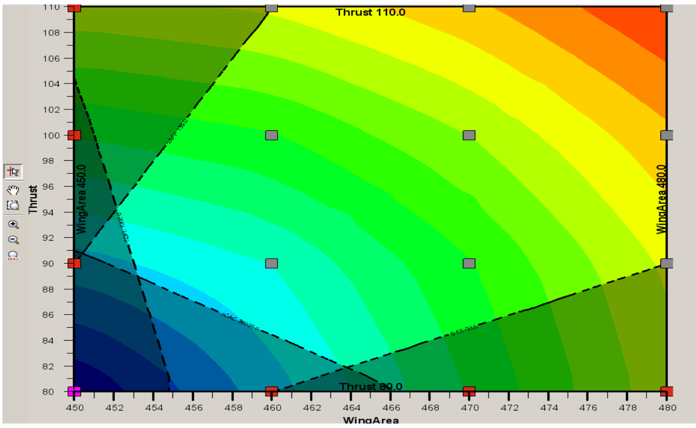

Figure 1: Contour plot for two inputs versus one response, with constraints, infeasible area shaded.

Another convenient way of plotting one response versus two inputs is a ***carpet plot***. That is a 2D chart with the response W plotted on the ordinate and the inputs X and Y substituted by a display function v = K_0 + K_x * X + K_y * Y which is then plotted as the (dimensionless) abscissa. Thus, a carpet plot is a 3D grid line surface plot projected onto a W-V-plane with W being the response value and V being a combination of the inputs X and Y. Accordingly, the factors K_0, K_x and K_y define the direction of projection.
The carpet plot formulation can be extended to include a third input variable Z by adding a term K_z * Z which makes it extraordinarily interesting for aircraft design. The resulting graph is a series of *staggered carpet plots* also called ***lattice plot.***

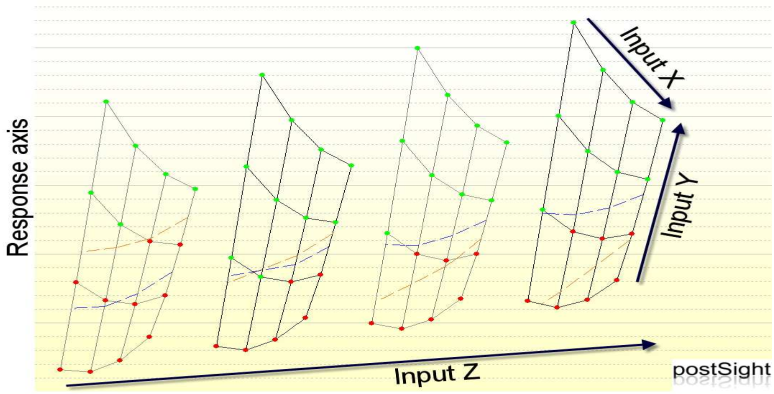

Figure 2: Lattice plot for three inputs versus one response, infeasible design points in red, two constraints (blue and orange dashed lines)

4. Collaboration

A main goal of introducing ISight in Airbus' preliminary aircraft design department was to simplify collaborative work in the project teams during the model build-up phase and to secure reproducibility of results of models for the future. Due to the inherent complexity of the design process each user of the test-team received access to the Isight Run-Time and Design-Gateway. Fiper was so far not considered to be part of this pilot project. Using this set-up, some issues and restrictions in aspects of collaborative teamwork were revealed.

- Lack of transparency, stability and performance issues with growing model complexity
- Difficult installation process with growing wrapped tool-set
- Limited exchangeability and collaborative work during model build-up and distribution

To cope with these issues the project team analysed the Ways-of-Working (WoW) with Isight and evaluated beneficial features for collaboration in the existing environment. Finally, a new process alleviating the model build-up phase was developed, tested and deployed.

Analysis of Way-of-Working:

Based on user-feedback, four different WoW were identified for model building, see Figure 4-1.

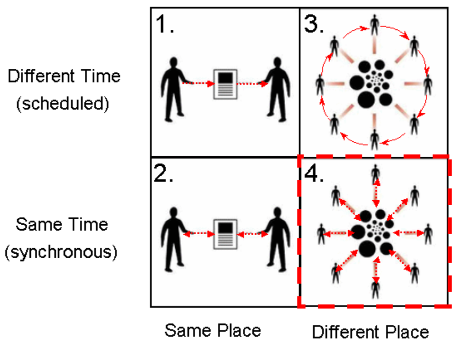

Figure 4-1: Platform supported Ways-of-Working [3]

Further analysis was performed on the availability and distribution of data in the process of Isight model building. Here it became obvious that Isight alone spreads model relevant information in various places.

- Project Model Directory
- Local library Directory (locallib)
- Job Database Directory
- Runtime Environment Directory

The first two locations are of main importance for the collaborative working. The "Project Directory" not only contains the Isight project file, but ideally all the model relevant data structured in a transparent and easy to update manner. The "Local Library" contains all the Isight related published components necessary to run an existing model. Since these components are updated regularly with every new Isight version the reproducibility of Isight model results is a challenge. For this it is necessary to preserve and control the “Local Library” for individual Isight models.

The Isight user has different data storage systems to cope with the different communication and data volume requirements. Common-Off-The-Shelf (COTS) and In-House developed applications are made available through official network drives for quick and redundant access. Manuals, documentations and other Isight related information is shared within the user group via an in-house Wiki platform.

New Collaborative Process:

To enhance collaborative work in a transnational IT environment during the phase of building an Isight model four essential requirements have been identified:

1 Easy access and update of Isight model data
2 Multi-User Isight model build-up
3 Reproducibility of results
4 Distribution of workload, knowledge and responsibility

To benefit from all these demands a more efficient WoW is necessary. Main idea of the first point is to enhance the transparency between Isight model in the Design-Gateway and the location of relevant input and output data by applying the same data structure, see Figure 4-2. The Isight SimFlow and the file system need to be organized and structured accordingly to allow a separation of work tasks [4]. Also, when distributing an Isight project model the main “modeldir” path can be utilized to define paths to the underlying data structure. Consequently, the Isight model can be easily distributed transnational without any further configuration work.

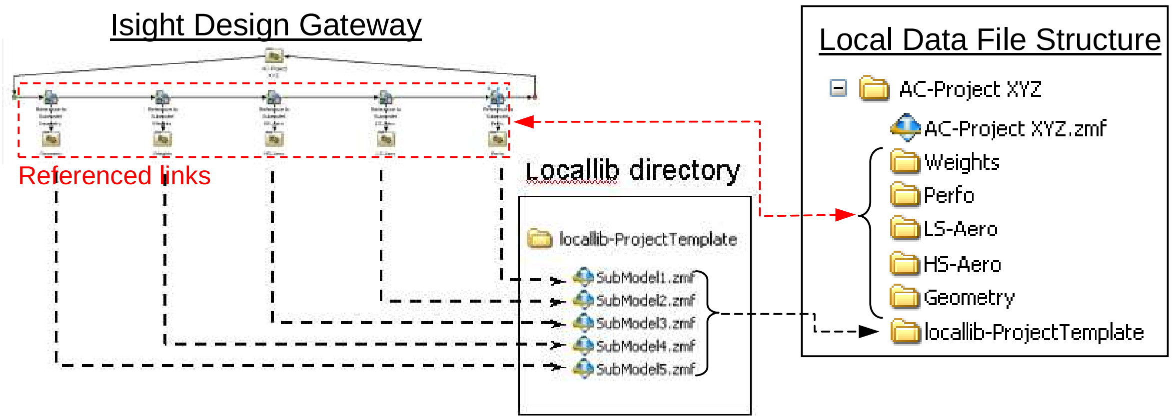

Figure 4-2: Design-Gateway & Local Data Structure Guideline

To allow multiple users to work on an Isight model simultaneously the different model disciplines need to be outsourced as separate sub-models. This can be achieved by transforming the individual task components into “**referenced links**” components. By doing this, the user gains access to individual Isight sub-models, which are published as unique components in the local library. This enables an Isight user to independently work on a certain Submodel and its corresponding data file structure without facing danger of interfering with other update work. As can be seen in Figure 4-2 the “locallib“ is included in the project directory. By configuring a designated project profile in Isight this locallib will be preserved for this model only and hence ensure reproducibility and exchangeability. This data-structure setup further enhances the collaboration capabilities tremendously when implementing a synchronisation process via a PDM system.

5. Conclusion

The introduction of Isight as a new integrated sizing platform has not only opened new opportunities for the OAD engineer, but also set new challenges on the ways-of-workings and method & tools developments. Isight provides various interfaces to integrate commonly used software tools to be linked in a tool chain. This forces the user of such model to not only understand all the implemented tools but also know the feasibilities and limitations of the model setup.

Although Isight is a very flexible sizing platform leaving great freedom to the user in integrating their methods, it has become clear that the method wrapping and to some extent the model building work has much in common to source code programming in a very graphical way. Therefore we see great importance to provide a software package, which secures the exchangeability of models and reproducibility of calculation results in a transnational business environment.

For the near future we see some more improvement potential in the post-processing capabilities, the Excel component performance, better mappings via aggregate structures and finally the Platform’s GUI which still has some missing state-of-the art functionalities such as “Undo”, “Window Maximizing” as well as better “Model Documentation” and “Report Generation” capabilities.

References:

1. Badufle, C., Blondel, C., Druot, T., and Duffau, M. (2006b). Robustness Criterion in Aircraft Sizing Studies. In Booklet of abstracts of the 7th International Conference devoted to Multi-Objective Programming and Goal Programming, Tours, Loire Valley.

2. Zhou, J. L., Tits, A. L., and Lawrence, C. T. (1997). User's guide for ffsqp version 3.7: A fortran code for solving constrained nonlinear (minimax) optimization problems, generating iterates satisfying all inequality and linear constraints. Technical Report TR-92-107r2, Systems Research center, University of Maryland, College Pard, MD 20742.

3. Collaboration Technologies URL: http://www.masternewmedia.org/news/2008/01/06/online_collaboration_technologies_new.htm [cited Januar 2008]

4. Alexandrov, N. M. and Lewis, R. M. (1999). Comparative properties of collaborative optimization and other approaches to MDO. Technical Report TR-99-24, NASA/ICASE.