Collaborative Design of a Business Jet Family Using the AGILE 4.0 MBSE Environment

Bussemaker, J.H.; Ciampa, P.D.; Singh, J. ; Fioriti, M. ; Cabaleiro, C. ; Wang, Z.; Peeters, D.M.J.; Della Vecchia, Pierluigi; Hansmann, P. ; More Authors

DOI
10.2514/6.2022-3934

Publication date
2022

Document Version
Final published version

Published in
AIAA AVIATION 2022 Forum

Citation (APA)
Bussemaker, J. H., Ciampa, P. D., Singh, J., Fioriti, M., Cabaleiro, C., Wang, Z., Peeters, D. M. J., Della Vecchia, P., Hansmann, P., & More Authors (2022). Collaborative Design of a Business Jet Family Using the AGILE 4.0 MBSE Environment. In *AIAA AVIATION 2022 Forum: June 27-July 1, 2022, Chicago, IL & Virtual* Article AIAA 2022-3934 (AIAA AVIATION 2022 Forum). American Institute of Aeronautics and Astronautics Inc. (AIAA). https://doi.org/10.2514/6.2022-3934

AIAA AVIATION Forum
June 27-July 1, 2022, Chicago, IL & Virtual
AIAA AVIATION 2022 Forum

10.2514/6.2022-3934

Collaborative Design of a Business Jet Family Using the AGILE 4.0 MBSE Environment

J.H. Bussemaker[1] and P.D. Ciampa[2]
German Aerospace Center (DLR), Hamburg, Germany

J. Singh[3]
Bombardier Aviation, Montreal, Canada

M. Fioriti[4] and C. Cabaleiro[5]
Politecnico di Torino, Turin, Italy

Z. Wang[6] and D.M.J. Peeters[7]
Delft University of Technology, Delft, The Netherlands

P. Hansmann[8]
Rheinisch-Westfälische Technische Hochschule (RWTH), Aachen, Germany

P. Della Vecchia[9] and M. Mandorino[10]
Università di Napoli Federico II (UNINA), Naples, Italy

This paper presents the collaborative model-based design of a business jet family. In family design, a trade-off is made between aircraft performance, reducing fuel burn, and commonality, reducing manufacturing costs. The family is designed using Model-Based Systems Engineering (MBSE) methods developed in the AGILE 4.0 project. The EC-funded AGILE 4.0 project extends the scope of the preliminary aircraft design process to also include systems engineering phases and new design domains like manufacturing, maintenance, and certification. Stakeholders, needs, requirements, and architecture models of the business jet family are presented. Then, the collaborative Multidisciplinary Design Analysis and Optimization (MDAO) capabilities are used to integrate various aircraft design disciplines, including overall aircraft design, onboard systems design, wing structural sizing, tailplane

---

[1] Researcher, Aircraft Design & System Integration, Institute of System Architectures in Aeronautics, jasper.bussemaker@dlr.de
[2] Head of MDO Group, Aircraft Design & System Integration, Institute of System Architectures in Aeronautics, AIAA MDO TC Member, pier.ciampa@dlr.de
[3] Engineer, Advanced Design, Jasveer.Singh@aero.bombardier.com
[4] Assistant Professor, Department of Mechanical and Aerospace Engineering, AIAA Member, marco.fioriti@polito.it
[5] PhD student, Department of Mechanical and Aerospace Engineering, carlos.cabaleiro@polito.it
[6] Researcher, Aerospace Structures and Computational Mechanics, Faculty of Aerospace Engineering, Z.Wang-16@tudelft.nl
[7] Assistant Professor, Aerospace Structures and Computational Mechanics, Faculty of Aerospace Engineering, D.M.J.Peeters@tudelft.nl
[8] Researcher, Institute of Aerospace Systems, philipp.hansmann@ilr.rwth-aachen.de
[9] Assistant Professor, Department of Aerospace Engineering, pierluigi.dellavecchia@unina.it
[10] Researcher, Department of Aerospace Engineering, massimo.mandorino@unina.it

sizing, mission analysis, and cost estimation. Decisions regarding the degree of commonality are implemented by optionally fixing the design of a shared component when sizing an aircraft.

I. Nomenclature

ADSG = Architecture Design Space Graph
AEA = All Electric Aircraft
CFD = Computational Fluid Dynamics
CPACS = Common Parametric Aircraft Configuration Schema
DOE = Design of Experiments
DSO = Data Schema Operation
MBSE = Model Based Systems Engineering
MDAO = Multidisciplinary Design Analysis and Optimization
MEA = More Electric Aircraft
MTOW = Maximum Take-Off Weight
NRC = Non-Recurring Costs
OAD = Overall Aircraft Design
OEI = One Engine Inoperative
OEM = Original Equipment Manufacturer
PIDO = Process Integration and Design Optimization
QOI = Quantity of Interest
RCE = Remote Component Environment
TLARs = Top-Level Aircraft Requirements
XDSM = eXtended Design Structure Matrix

II. Introduction

The development of a business jet has always been a capital-intensive business; it became even more so due to an increase in system complexities, increase in demand for high performance and fuel efficiency, and the push for sustainability. This trend is expected to continue in the future. Because of this, single aircraft programs are becoming less viable options for aircraft manufacturers. One way to increase the viability of a program is to develop an aircraft family: two or more aircraft with common components that cater to different market segments [1]. Any additional aircraft will only cost a fraction of non-recurring (i.e. design, certification, testing) costs of the baseline aircraft and yet, it can increase the market share significantly.

The underlying trade-off driving the design of an aircraft family is one of manufacturing costs (including design and certification) and operating costs. These two aspects can also be represented by level of commonality (also known as commonality index [2]) and aircraft performance in terms of fuel burn. The disadvantage of having common components is that the smaller aircraft will be less efficient than the corresponding single-point design. The more the commonality, the less the efficiency. Product family design in general comes in two flavors [3]: using scalable design parameters (e.g. cabin length) or developing modular product architectures. Usually a combination of the two is applied, and using these two methods a design space can be constructed where different levels of commonality can be selected for.

Aircraft design, and by extension aircraft family design, is a highly multidisciplinary process and involves the definition and design of complex systems. The continuous increase in complexity of aerospace systems requires the development of new design methodologies leveraging models and physics-based simulation and integrating the system design in a Model-Based Systems Engineering (MBSE) process [4]. The aircraft family design presented in this paper is one of seven application cases in the EC-funded AGILE 4.0 project [5]. This project builds on collaborative aircraft design methodologies previously developed in the AGILE project [6], and extends it with Model Based Systems Engineering (MBSE) capabilities, including requirements modeling and system architecting. Applying such methodologies allows the consideration of the entire aircraft lifecycle, including certification, manufacturing, and maintenance. Technologies developed in the AGILE 4.0 project are integrated in the AGILE 4.0 Development System that enables users to apply these technologies from a single platform accessible online, to enable collaboration between partners residing in different departments, organizations, and/or countries.

This paper presents the design of a family of business jets, led by Bombardier Aviation, integrated by the German Aerospace Center (DLR), and supported by Politecnico di Torino, Delft University of Technology, Rheinisch-

Westfälische Technische Hochschule (RWTH), and Università di Napoli Federico II (UNINA) as disciplinary experts. Each aircraft has a T-tail configuration (see Fig. 1) and is designed to carry 8 passengers with varying design ranges and cabin lengths. On the family level, design decisions include whether to share the wing, engine, onboard systems, landing gear, and/or tailplane designs. On the aircraft level, design parameters include wing sweep, wing thickness ratio, and flap planform parameters. For each aircraft its design mission is evaluated for fuel burn, which is combined into a family-level weighted fuel burn. Aircraft are sized by coupling an overall aircraft design tool with higher-fidelity analyses, including onboard systems design, tailplane sizing, and wing mass estimation. Additionally, recurring and non-recurring costs are calculated as the main basis for the trade-off between manufacturing and operating costs.

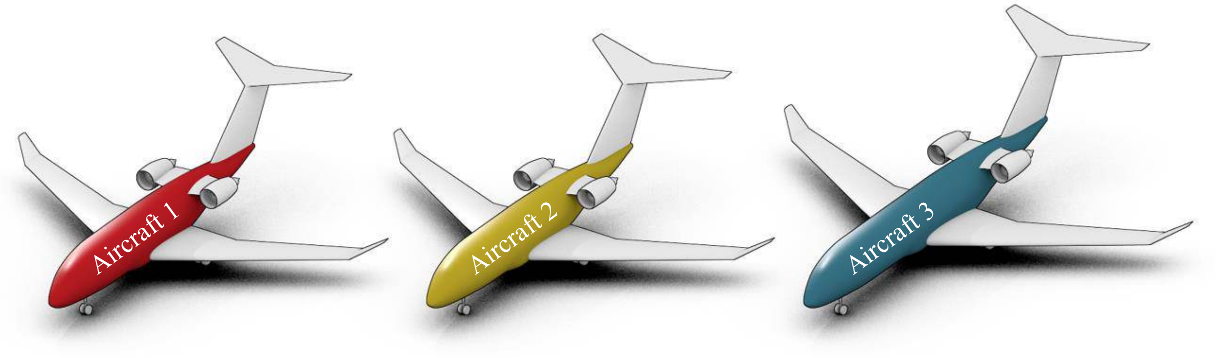


Fig. 1 Notional render of the business jet family consisting of three aircraft designed for different ranges and cabin lengths, showing aircraft with longer design ranges and larger cabins to the right, numbered 1, 2 and 3.

The paper is structured around the two main themes of the AGILE 4.0 project: MBSE and Multidisciplinary Design Analysis and Optimization (MDAO). In the section on MBSE the system definition in terms of stakeholders, needs, requirements, and architecture is presented. In the section on MDAO the system design process in terms of disciplines and executed workflow is presented. Then, results are presented and the paper is concluded.

III. Model-Based System Definition

This section presents the Model-Based system definition for the business jet aircraft family, most of it done by the DLR supported by Bombardier. This part forms the basis for the system design, as all artifacts and decisions related to the system should be traceable all the way back to stakeholder needs, and is shown in Fig. 2. The stakeholders and needs identification and requirements modeling process applied here is presented in [7]. The system of interest considered in this section is the business jet family: the individual business jets are considered sub-systems.

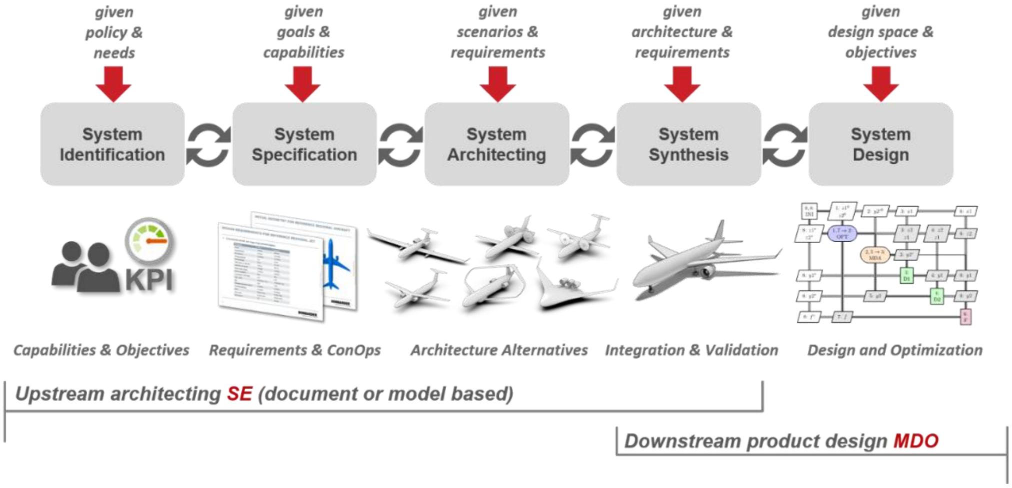


Fig. 2 Complex system development framework developed in AGILE 4.0, bridging upstream architecting (e.g. a systems engineering approach) and downstream product design (e.g. a multidisciplinary design and optimization approach). Figure from [4].

The rest of this section introduces the systems engineering steps. First, stakeholders and needs are identified in the System Identification phase. Then, requirements are defined in the System Specification phase. The section finishes with a description of the functional and logical architecture models, as part of the System Architecting phase.

A. Stakeholders and Needs Identification

The first step in the system definition is the identification of the system stakeholders and their needs. Stakeholders have an interest in the system under development. Stakeholders express needs: something about the system that must, should, or may be realized (e.g. the Passengers have the need to “reach many destinations worldwide”). Needs exist in the mind of the stakeholder and are often expressed in fuzzy and/or general (i.e. ambiguous) terms. Each need can only specify one aspect of the system.

The following stakeholders are identified for the business jet family application case: OEM, Operator, Engine OEM, Passengers, Pilots, and Regulatory Authorities. It is noted that this list will include additional stakeholders for a real system design project. Needs are identified for each stakeholder, and a SysML model is automatically generated using the AGILE 4.0 MBSE Environment. Fig. 3 shows the needs model for a stakeholder.

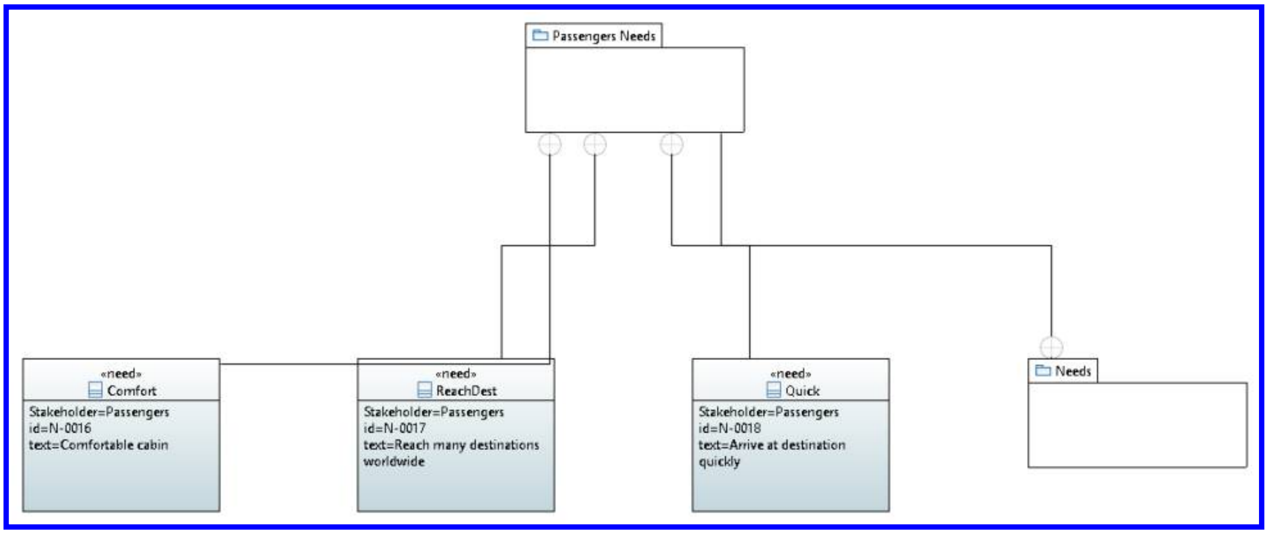
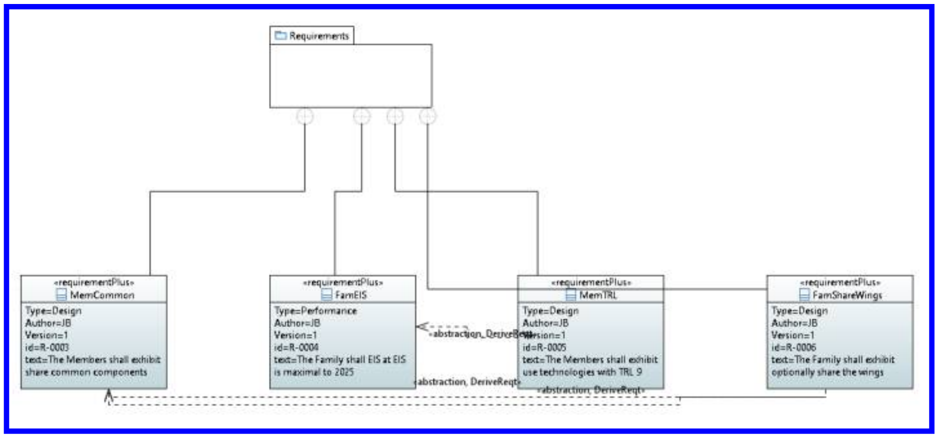


Fig. 3 Needs model for the needs of the Passengers stakeholder.

B. Requirements Modeling

Requirements are a transformation of stakeholder needs with the goal to make them unambiguous and consistent (e.g. the aircraft shall have a maximum landing length of 1 km). Each requirement is directly derived from a need or from a parent requirement, enabling traceability. Requirements consist of attributes and a requirement statement. The statement has a type (e.g. functional, performance, etc.), a pattern, and is subject to certain rules. The pattern and rules ensure completeness, consistency, unambiguousness and automatic verification. More information regarding this can be found in [7].

Requirements have been defined and grouped into the following categories: general, airport performance, mission performance, cabin, handling qualities, environmental performance, and regulatory. A subset of the requirements model is shown in Fig. 4.



MemCommon:
Type=Design
Author=JB
Version=1
id=R-0003
text=The Members shall exhibit share common components

FamEIS:
Type=Performance
Author=JB
Version=1
id=R-0004
text=The Family shall EIS at EIS is maximal to 2025

MemTRL:
Type=Design
Author=JB
Version=1
id=R-0005
text=The Members shall exhibit use technologies with TRL 9

FamShareWings:
Type=Design
Author=JB
Version=1
id=R-0006
text=The Family shall exhibit optionally share the wings

Fig. 4 The "General" requirements set showing several requirements related to family size and introduction into service.

The Top-Level Aircraft Requirements (TLARs) are also part of the requirements model. Due to their significance to the design process, they are also listed in Table 1.

Table 1 Business jet family TLARs. Value applies to all aircraft if one value is given.

Parameter	Aircraft 1	Aircraft 2	Aircraft 3
Long-range cruise speed	Mach 0.85	Same as Aircraft 1	
High-speed cruise speed	Mach 0.90	Same as Aircraft 1	
Design payload	8 passengers	Same as Aircraft 1	
Design range	2500 NM	3000 NM	3500 NM
Cabin length	25 ft (7.62 m)	25 ft (7.62 m)	35 ft (10.67 m)
Balanced field length	5000 ft (1524 m)	Same as Aircraft 1	
Landing field length	2500 ft (762 m)	Same as Aircraft 1	
Initial cruise altitude	41000 ft	Same as Aircraft 1	

C. Functional, Logical and Physical Architecture Design Space

Once system requirements have been derived, the next step of the AGILE 4.0 MBSE process aims at creating multiple alternatives of system architectures. This part of the process is called system architecting, and it produces three types of architectures: functional, logical, and physical. The system architecting process applied here is presented in [8].

The functional architecture consists of use cases defined from functional requirements. Each use case specifies how a system may be used by its users. Use cases contain boundary functions: these are the functions that the system should perform in order to fulfill its requirements, independent of *how* the system will perform these functions.

For the business jet application case, several use cases were defined containing boundary functions related to different levels of elaboration (of the system): for example, the system should "Contain Passengers", "Regular Temperature", and "Transport Passengers". This section will focus on the "Transport Passengers" boundary function, as it forms the basis for the logical architecture of the business jet family.

After the definition of the functional architecture, the logical architecture design space is modeled: it describes all the possible logical architectures that can be considered in the design of the system. A logical architecture is a mapping between functions and components, and thereby defines how the system performs its functions; it defines the "solution" to the design problem. A physical architecture extends this by elaborating more on the components to prepare them for actual implementation: for example, components are further characterized, connections are established if needed, and Quantities of Interest (QOIs) are defined. The architecture instances generated from the design space form the bridge between the MBSE phase and the MDAO part, in that these architectures describe the final product to be designed and analyzed for performance. In this context, QOIs represent inputs to or outputs from the MDAO process: static inputs, design variables, objectives, constraints, or output metrics. The design space model is implemented using the Architecture Design Space Graph (ADSG), a graph-based description of system architectures that enables the systematic generation of architecture instances by optimization [9]. The ADSG is modeled using ADORE, a web-based tool developed by the DLR that offers a convenient user interface [10].

Fig. 5 shows the architecture design space model as created in ADORE. It shows how the "*Transport Passengers*" function is fulfilled by the three *Business Jet* components, defined using a multi-fulfillment element. The three business jets then follow a similar decomposition scheme, where several functions are induced and further fulfilled. For some of the induced functions, a decision exists (as shown by blue-dashed lines) on how to fulfill them. For example, for the "*Generate Lift 1*" function, there is a choice between "*Wing 1*" or "*Shared Wing 1*", which represents a choice between a wing designed for the first business jet, or using a shared wing (shared with the second business jet). For more information on how components can be shared between the aircraft, refer to section IV.A.

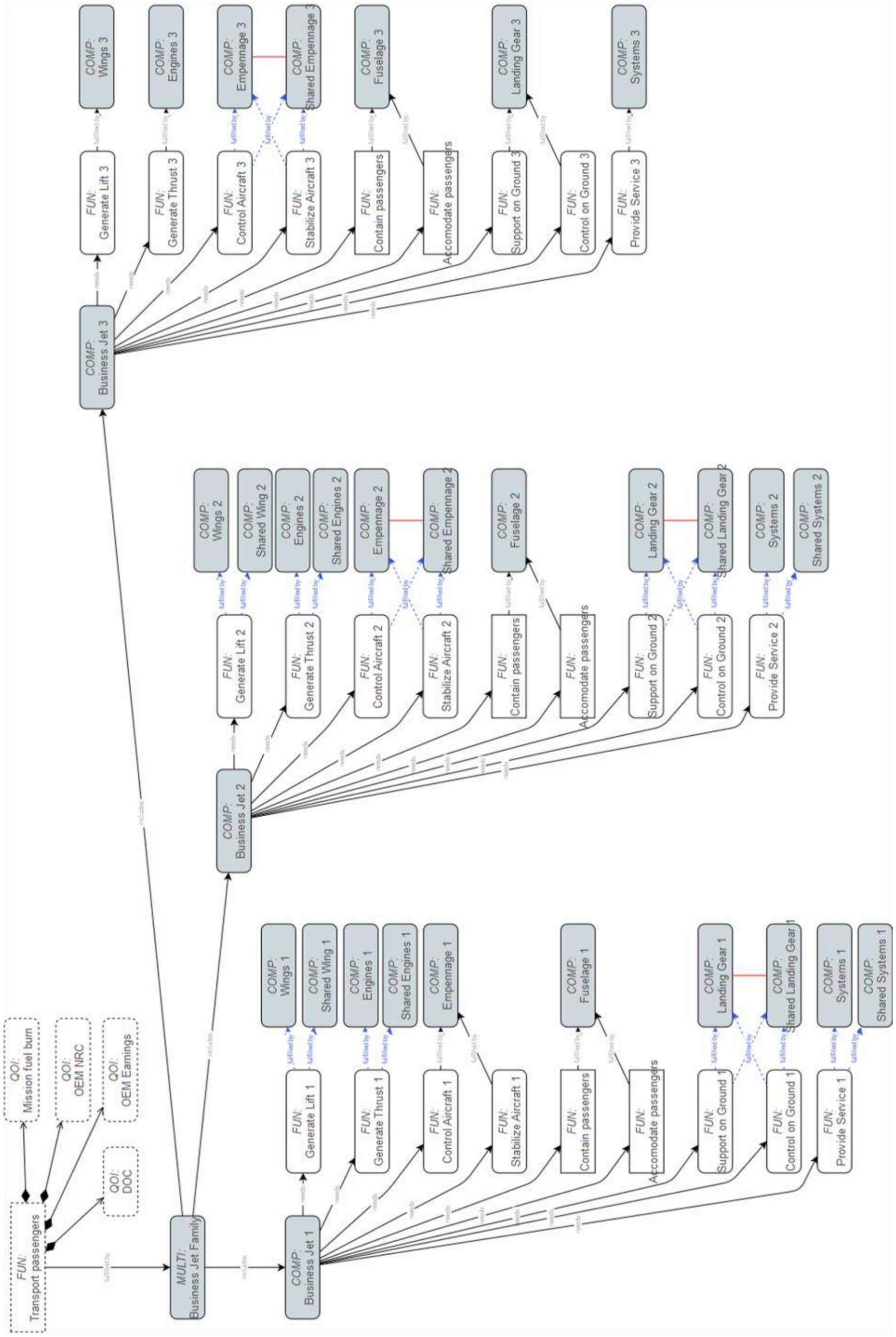


Fig. 5 Business jet family architecture design space model, showing three aircraft-level architectures with component sharing decisions (decisions are shown using blue-dashed lines).

The MDAO chain will need and provide many metrics, defined using QOI elements in the architecture design space model. ADORE as integrated in the collaborative MBSE environment enables QOIs to be linked to upstream non-functional requirements, thereby enabling traceability from stakeholder needs all the way to MDAO inputs and outputs. Fig. 6 shows what the QOI definition for one of the business jets looks like: here it cannot be seen whether they are defined as inputs or outputs, however that data is defined in the model.

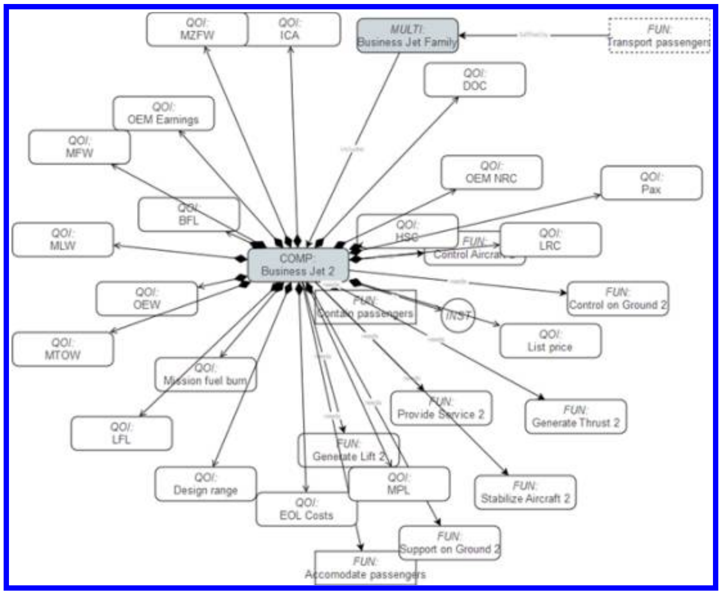


Fig. 6 Detailed view of one of the business jet components, showing related functions and QOIs.

Architectural decisions are automatically identified from the architecture design space model. The model for the business jet includes 19 architectural decisions, including 10 component sharing decisions and 9 wing design variables. For more information on the component sharing logic, refer to section IV.A.1. The 9 wing design parameters include the wing sweep angle, thickness-to-chord (t/c) ratio, and rear spar chordwise position for each of the three aircraft. One important phenomenon to note here is that of design variable hierarchy: if for an aircraft the wing is shared, its corresponding wing design variables are not active as the wing design variables of the originating aircraft are used for the shared wing design. In terms of the optimization problem, this means that for some values of the categorical component sharing variables, some of the continuous wing design variables might be inactive. This concept is a common occurrence in architecture optimization problems and dealt with appropriately by ADORE [9].

Finally, it is important to note that since the architecture design space is tightly related to the MDAO capabilities, the development of both the MDAO capabilities and the architecture design space model is an iterative exercise [11]. The architecture design space model drives the decisions that should be accounted for and the metrics that should be optimized for, whereas the MDAO capabilities determine which decisions can actually be accounted for and which metrics can be analyzed.

IV. Multidisciplinary Analysis and Design Toolchain

Once system architectures have been defined, these architectures should be integrated into a complete product model that should be correctly sized and of which the performance should be estimated. Since the system of interest here is the aircraft family, the aircraft family is considered to be the product to be sized.

Since aircraft design is a highly multidisciplinary process, collaborative MDAO techniques developed in the AGILE and AGILE 4.0 projects are leveraged. This enables the formulation of collaborative MDAO workflows that automatically communicate product data in a common data schema across organizational boundaries [12]. This enables the use of all relevant design disciplines and associated expertise needed to design the product up to a desired level of fidelity.

CPACS is used as the common data language for representing the aircraft being designed [13]. The use of such a common language reduces the amount of data interfaces needed to be developed and moves the responsibility for implementation from the system integrator to the tool owners, both leading to time savings in design projects and increased reusability across projects.

A. Design Competencies

This section introduces the design competencies from different organizations used to design the aircraft family. In general, the following design tools cooperate on the aircraft-level (i.e. all related to one specific aircraft). The cost estimation discipline and commonality switches operate between aircraft designs and thus on family-level.

1. *Overall Aircraft Design: openAD*

Overall Aircraft Design (OAD) is performed using openAD, a knowledge-based aircraft design tool developed by the DLR [14]. openAD parameterizes an aircraft on a high level using over 1000 internal parameters, covering many aircraft disciplines on a handbook or low-fidelity level, including performance sizing, mass estimation, simple mission simulation, stability & control, geometry layout. Its input is specified by fixing certain internal parameters, such as those related to the TLARs (e.g. design range, design payload, cruise speed and altitude, etc.). Its output is a CPACS model representing a consistent aircraft design. Integration with higher-fidelity analyses is achieved by additionally overriding input parameters related to higher-fidelity results with these high-fidelity results, for example wing mass, mission fuel burn, and subsystem energy consumptions. OAD from TLARs is referred to as *OAD Initialization*, whereas OAD from higher-fidelity analyses is referred to as *OAD Synthesis*.

Component commonality logic is implemented using a commonality-switching tool. This tool prepares openAD input by optionally overriding component-related parameters from another family member. For example, if a wing is shared between two family members, the inputs for the smaller aircraft are fixed to match the design of the larger aircraft. This principle is visualized in Fig. 7.

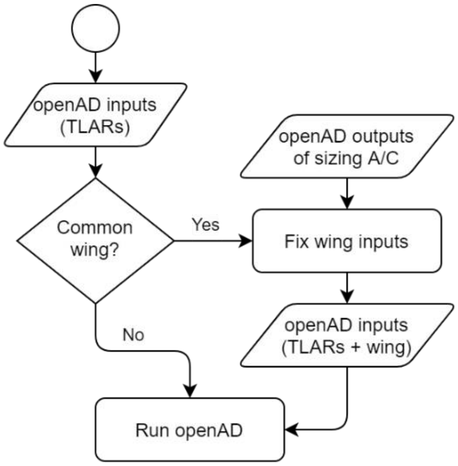


Fig. 7 Commonality switching logic, shown for a common wing. The openAD input for the non-sizing aircraft is optionally overwritten by the wing output of the sizing aircraft.

For wings, engines, landing gear, and onboard systems the largest aircraft in the family provides the sizing case, whereas for the tailplane the smallest aircraft provides the sizing case. This scheme is visualized in Fig. 8. It can be seen that this introduces a feedback loop because the tailplane design depends on aircraft 1 (the smallest aircraft).

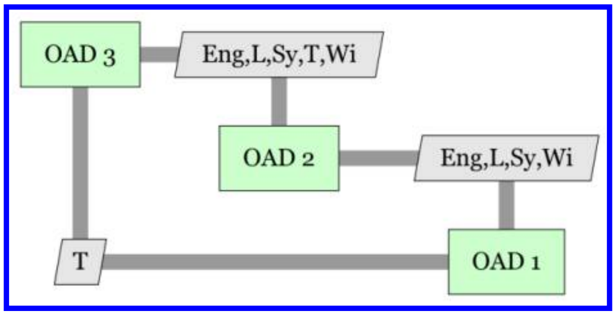


Fig. 8 Component sharing scheme if all components are shared. OAD represents the aircraft. Shared components include the engine (Eng), landing gear (L), onboard systems (Sy), tailplane (T), and wing (Wi).

2. *Onboard System Design: ASTRID*

Onboard systems are sized using ASTRID, a tool developed by Politecnico di Torino [15, 16]. The tool needs data regarding aircraft geometry, main aircraft masses and some specific inputs necessary to define the specific systems architecture to be designed. With these specific inputs, ASTRID can design systems architecture with different electrification levels from the state-of-the-art technology to More Electric Aircraft (MEA) and All Electric Aircraft (AEA) concepts. ASTRID uses both semi-empirical and physics-based algorithm to calculate system masses, power required, and bleed air required. Firstly, all main utility systems (i.e. avionics, flight control, landing gear, etc.) are designed. Then, considering the utility systems loads and their power requirement, a power budget for each phase of the mission profile is defined. Starting from this information, ASTRID designs the power distribution and generation systems (i.e. electric, pneumatic, hydraulic systems). Finally, the whole engine power offtakes are provided. The results are provided at system and main equipment level. The design process is shown in Fig. 9.

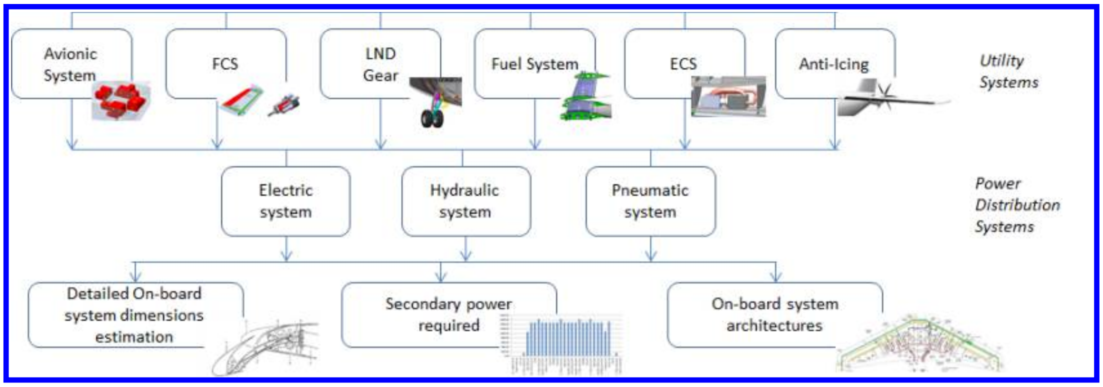


Fig. 9 ASTRID onboard systems design process [16].

3. *Flap CL_{max} Estimation*

Flap CL_{max} and drag increase is estimated using a tool developed by UNINA. This tool uses low-fidelity physics-based methods to do this estimation, which is needed to correctly calculate landing and take-off performance in openAD. To perform its calculations, the tool needs the sized aircraft and flap geometry parameters.

The tool initially computes the wing aerodynamic characteristics necessary to define the wing lift trend and distribution. This step is executed considering the airfoils aerodynamic characteristics until stall and post stall conditions and their distribution along the wingspan. The methodology adopted is a modification of Nasa Blackwell standard vortex-lattice procedure, capable to predict wing aerodynamic characteristics [17, 18]. The method allows to obtain reliable analyses (error is lower than 7%), fast (less than 5 seconds for a complete analysis) and easy to use. The following step includes aerodynamic characteristics computation of the wing with high lift devices, obtained through Sforza semi-empirical methodologies [19]. The high lift devices typology, distribution and geometry are considered. The modification of lift and drag curves due to each single high lift device and the whole wing are computed considering the deflection of each device. In this way, it is possible to compute the new lift and drag curves for take-off and landing configurations. Fig. 10 and Fig. 11 show two examples of output automatically generated by the tool: the representation of high lift device on the wing and the wing CL curve for clean and landing configurations, respectively.

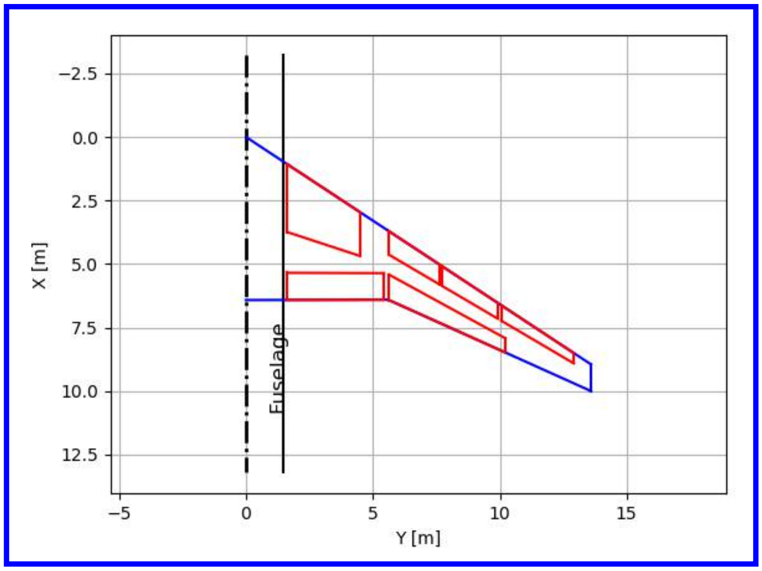


Fig. 10 Representation of high-lift device distribution obtained through the Flap CL_{max} Estimation tool.

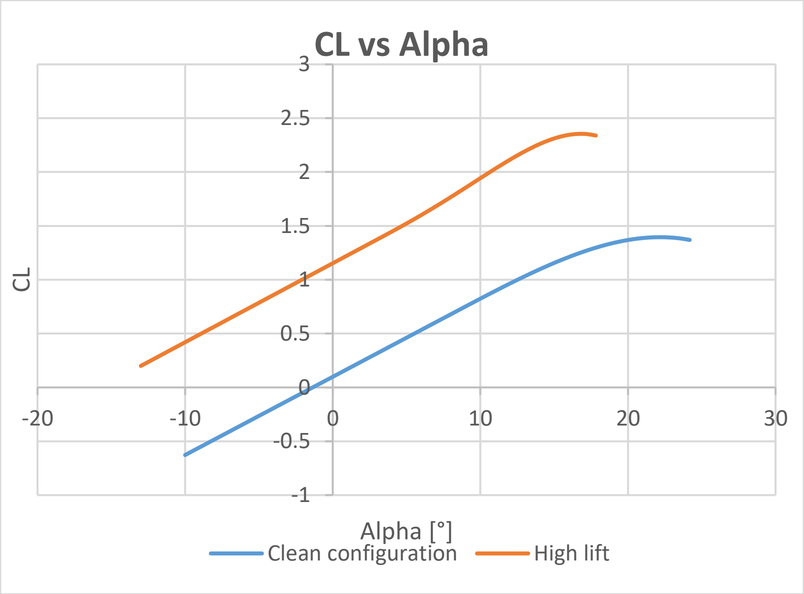


Fig. 11 CL_{max} Estimation tool output. Wing CL curve for clean and landing configuration.

The CL_{max} tool is hosted remotely by UNINA. To reduce runtime due to data transfer and tool execution, the tool is integrated in the workflow using a multi-fidelity Kriging surrogate model.

4. Tailplane Sizing

Horizontal- and vertical tailplane areas are sized by a low-fidelity physics-based method considering certification constraints, One Engine Inoperative (OEI) conditions, and stability and controllability.

Since the fuselage has a significant impact on both horizontal and vertical tailplane design, the first step of this tool consists in computing the aerodynamics of the fuselage and its effect on aircraft stability. A methodology, developed by UNINA through CFD analyses, is exploited to estimate fuselage aerodynamic drag, pitching, and yawing moment coefficients [20]. Several geometry factors such as fuselage components fitness ratio, windshield and tail cone angle are accounted to evaluate, through charts and diagrams, the drag, pitching and yawing moment contribution coming from nose, tail, and main body of the fuselage. Then, the horizontal tail plane sizing is computed ensuring the stability during cruise and the controllability during landing. For what concern the vertical tail plane sizing, a semi-empirical method for the evaluation of aircraft directional stability is employed [21]. The methodology includes the definition of factors useful to account for the interference between fuselage, vertical tailplane, wing, and horizontal tailplane. In Fig. 12 an example of the tailplane sizing tool application is shown. The complete aircraft before and after the sizing of horizontal and vertical tailplane is represented.

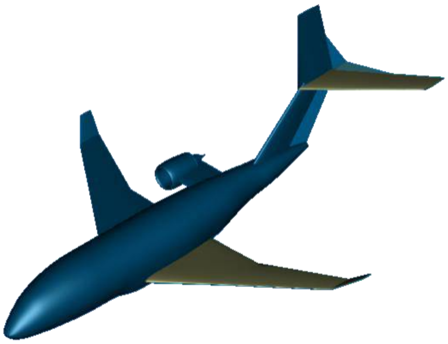


Fig. 12 Aircraft visualization before and after the sizing. The original tail is displayed semi-transparent.

Similarly to the CL_{max} tool of UNINA, the tailplane sizing tool is also integrated in the workflow using a multi-fidelity Kriging surrogate model.

5. *Wing Mass Estimation: PROTEUS*

The wing mass is estimated by performing composite wing structural design using PROTEUS, a tool developed by Delft University of Technology [22]. Wing mass is minimized for cruise conditions by tuning lamination parameters and thicknesses while making sure aeroelastic stability, angle-of-attack, strength, and buckling constraints are satisfied. The wing structural model is shown in Fig. 13. Total wing mass is calculated from the primary structure mass using empirical relationships from [23].

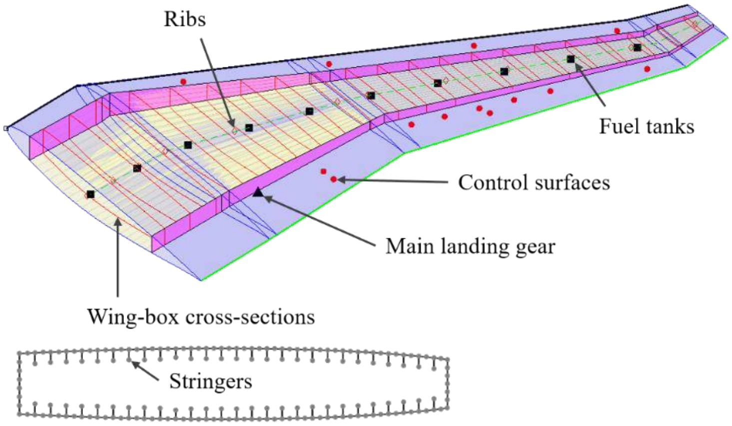


Fig. 13 Visualization of the wing box structural model used for composite wing design, including point-mass models of wing components.

One wing design takes about an hour, therefore to reduce the time needed to design the aircraft family, the wing mass estimation results are integrated in the MDAO workflow using a surrogate model. A multi-fidelity Kriging model is constructed that predicts the wing mass based on five input parameters: Maximum Take-Off Weight (MTOW), wing area, wing sweep, thickness ratio, and the rear spar location (% of chord).

6. *Mission Simulation: AMC*

openAD contains rudimentary mission simulation calculations in order to determine how much fuel is needed to complete the mission and thereby size the fuel tanks and weights [14]. However, these calculations are based on the Breguet range equation, and are not accurate enough to generate the required output for calculating operating costs. To generate these outputs, the mission calculation tool AMC is used. AMC performs 2D pointwise mission analysis to calculate the fuel burn and generate CPACS trajectories output [14]. As input it needs a mission definition, where there is a choice between various segments. For example, the climb segment can be contained at a constant airspeed or Mach number with full throttle; the vertical profile is calculated automatically. Also, the cruise segment can include a step climb for reducing fuel burn.

7. *Cost Estimation*

Costs are estimated using empirical methods developed by RWTH Aachen [24, 25]. Aircraft-level costs are estimated using established methods, and assumptions on production numbers, daily utilization, operational life, and average number of flights per year.

Commonality effects are considered amongst the non-recurring costs (NRC) of the aircraft program. The individual cost fractions that make up the NRC and that are taken into account within the cost model are shown in Fig. 14 (left). From a cost breakdown of non-recurring costs of a reference aircraft on individual aircraft parts, as shown in Fig. 14 (right) the effect of the commonality of respective parts has been derived. The engines are excluded from this breakdown retrospectively as they are considered as an additional purchase part.

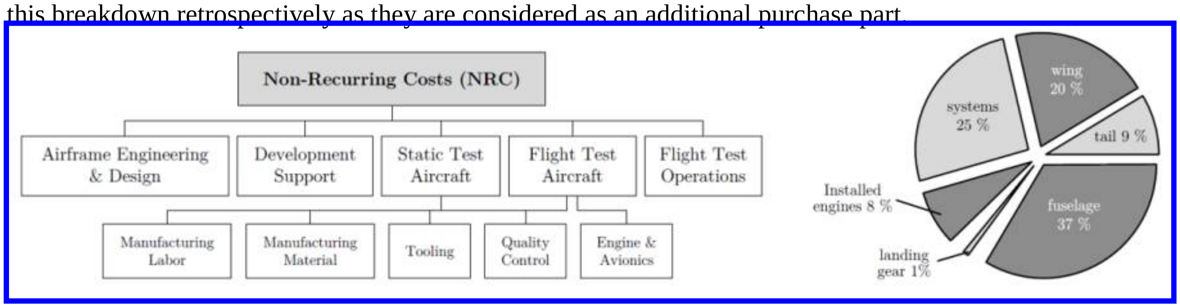


Fig. 14 Cost components of non-recurring cost & breakdown of non-recurring costs on aircraft parts [25].

From the cost fractions and the considered commonality scenario for respective parts, an overall commonality factor can be calculated. This overall commonality factor is applied as a cost reduction factor to relevant NRC. This takes into account that e.g. NRC spent on tools do not have to be spent again for other aircraft derivatives. As the NRC are split up amongst all aircraft derivatives, the NRC of the individual aircraft program can be reduced.

Aircraft direct operating costs are calculated from mission trajectories analyzed by AMC, and assumptions on fuel price and number of flights per year. Costs are aggregated at family-level by averaging recurring costs across family members, and summing non-recurring costs (as it already takes commonality effects of individual aircraft into account).

B. MDAO Workflow

All the tools described in the previous section communicate using CPACS, and for all of them formalized input and output definitions are available. These definitions specify which variables modeled in the CPACS file they need as input and output, and thereby enable the automated formulation of MDAO workflows using technologies developed in the AGILE project [6]. For this application case, the MDAO workflow is modeled using MDAx, a tool developed by the DLR for interactively creating and editing workflows [26].

The business jet family design workflow consists of two levels: the family-level (see Fig. 16) and aircraft-level (see Fig. 15) workflows. The aircraft-level workflow takes care of the design of one aircraft at a time by implementing commonality decisions and coupling higher-fidelity tools to OAD. The family-level workflow couples the three aircraft-level workflows and iterates until all aircraft designs are consistent with each other, after which aircraft-level costs are calculated. Finally, metrics (costs and fuel burn) are aggregated at the family-level.

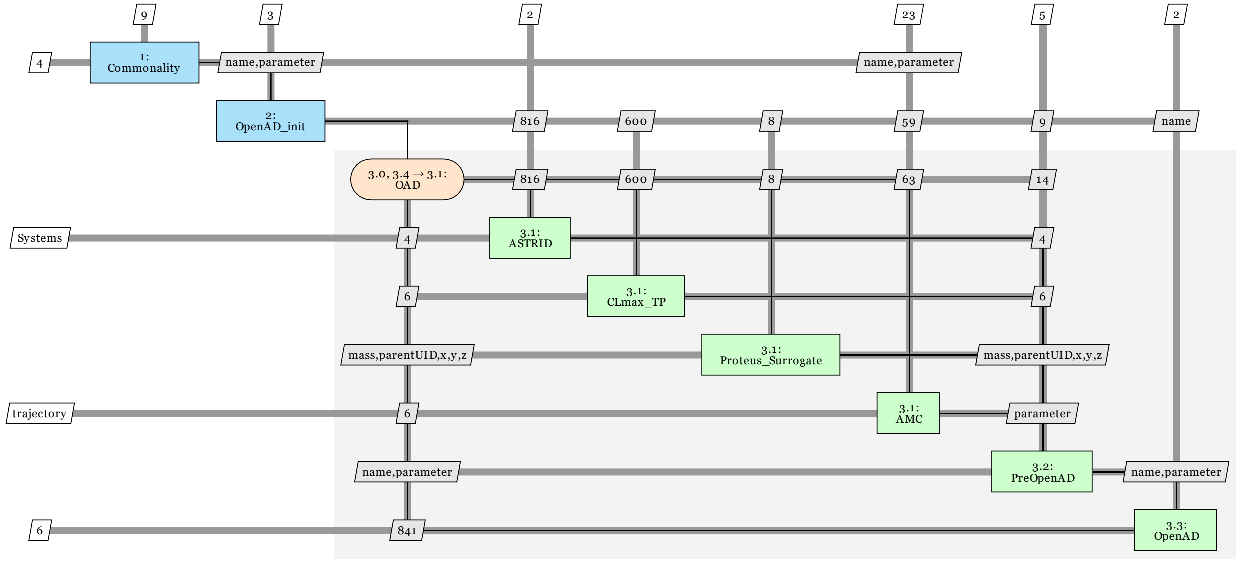


Fig. 15 Extended Design Structure Matrix (XDSM) representation of the aircraft-level workflow implementing commonality logic and coupling higher-fidelity tools to openAD.

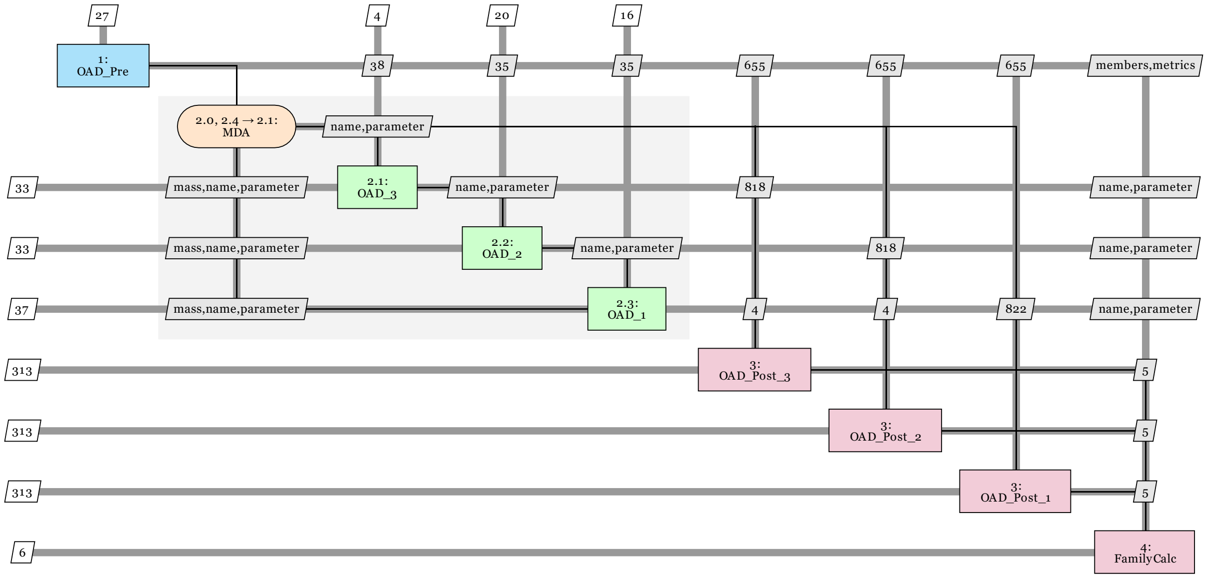


Fig. 16 XDSM representation of the family-level workflow coupling the three aircraft-level workflows (OAD_*). After the aircraft have been designed, costs are calculated in the OAD_Post_* blocks, and metrics are aggregated at the family-level in the FamilyCalc block.

Once the workflow has been modeled, it is exported to RCE (Remote Component Environment)[11], a Process Integration and Design Optimization (PIDO) environment developed by the DLR. Some of the design competences discussed in the previous section are developed by the different partners of the project consortium. Since we are operating in a cross-organizational context, this means that these are executed remotely at the tool owner. To communicate required input and output data across organizational boundaries, a tool previously deployed in AGILE is used: Brics, developed by the project partner NLR [27] [28]. Brics uses a central data server that is located on a neutral server outside of any network to send files between computers. The family-level RCE workflow including Brics calls is automatically created from MDAx and shown in Fig. 17.

---

[11] https://rcenvironment.de/

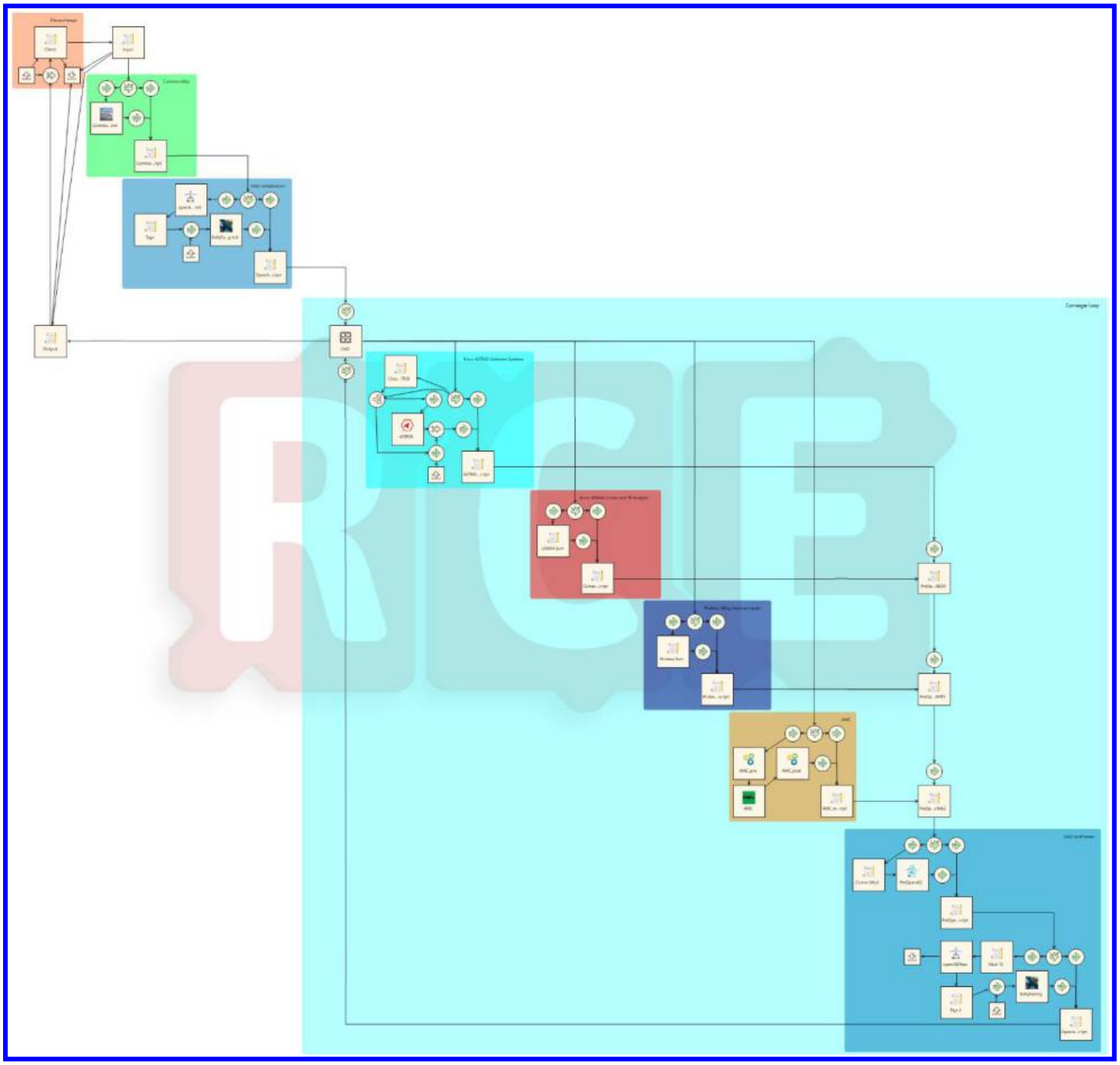
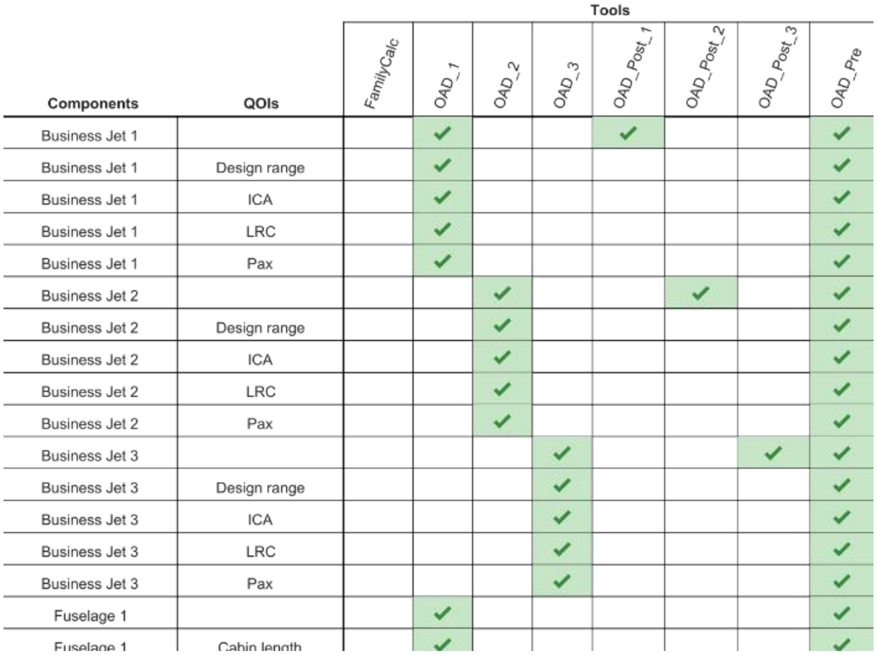


Fig. 17 Aircraft-level RCE workflow. Each colored block along the diagonal represents a different engineering tool. The large turquoise block in the back represents a converger loop.

C. Architecture-to-MDAO Connection

To run a Design of Experiments (DOE) or optimization and enable traceability all across the design process, the architecture design space model must be connected with the MDAO workflow. This is done using a tool called MultiLinQ [10], developed by the DLR and available in the collaborative MBSE environment [29]. MultiLinQ takes analysis tool input/output definitions, as available from the MDAx workflow model, and information about components and associated QOIs in the architecture design space. Then, it allows the user to define Data Schema Operations (DSOs) that define how a component or QOI is mapped to (or from in the case of output) the central data schema, in this case CPACS. For example, a QOI representing a design variable (e.g. wing sweep) might define a DSO that writes its value to a CPACS node defining this parameter as input to openAD. This enables the coupling between architecture model and QOI, so that each generated architecture can be analyzed using the MDAO workflow and output metrics are fully traceable. For more details on how MultiLinQ is integrated in the collaborative MBSE environment and how it can be used to connect ADORE and RCE during an optimization run, the reader is referred to [29].

After defining DSOs, a Component-Tool (CT) matrix can be created to visualize how components and QOIs (as defined in the ADORE architecture design space model) are mapped to disciplinary analysis tools (as defined in the MDAx workflow model). A part of the matrix for the business jet family application case is shown in Fig. 18. Next to verifying that all elements have been correctly mapped, it can also be used to support the selection of included disciplinary tools, and to identify gaps in disciplinary capabilities. It can be seen that indeed all tools needed to provide relevant QOIs are available.



		Tools							
Components	QOIs	FamilyCalc	OAD_1	OAD_2	OAD_3	OAD_Post_1	OAD_Post_2	OAD_Post_3	OAD_Pre
Business Jet 1			✓			✓			✓
Business Jet 1	Design range		✓						✓
Business Jet 1	ICA		✓						✓
Business Jet 1	LRC		✓						✓
Business Jet 1	Pax		✓						✓
Business Jet 2				✓			✓		✓
Business Jet 2	Design range			✓					✓
Business Jet 2	ICA			✓					✓
Business Jet 2	LRC			✓					✓
Business Jet 2	Pax			✓					✓
Business Jet 3					✓			✓	✓
Business Jet 3	Design range				✓				✓
Business Jet 3	ICA				✓				✓
Business Jet 3	LRC				✓				✓
Business Jet 3	Pax				✓				✓
Fuselage 1			✓						✓
Fuselage 1	Cabin length		✓						✓

Fig. 18 MultiLinQ Component-Tool (CT) matrix, showing how components and QOIs (rows) are mapped to disciplinary blocks of the family-level workflow (columns).

V. Results

To test the MDO workflow implementation, a 13-point DOE is executed, each point representing an aircraft family with various levels of component sharing and different values for the wing design variables. Sampling points are generated using a Latin hypercube sampling scheme that additionally considers design variable hierarchy. At the time of running the DOE, the cost estimation tool had not yet been integrated, so no cost-related results are presented. Depending on the amount of iterations (both at family-level and at aircraft-level) needed for convergence, each of the families took between 45 minutes and 1.5 hours to be designed.

The general trend of the impact of component sharing decisions on the aircraft designs is presented in Fig. 19: aircraft commonality represents the total number of components shared within the family, and increasing it leads to heavier aircraft and a higher mission fuel burn; i.e. a reduction in aircraft performance. This is also expected, considering that component sharing means that some of the aircraft are using components that are not designed for their design point. This can also be seen in Fig. 20, where for four components the impact on sharing is displayed: sharing a component leads to oversized components on the aircraft receiving the shared component.

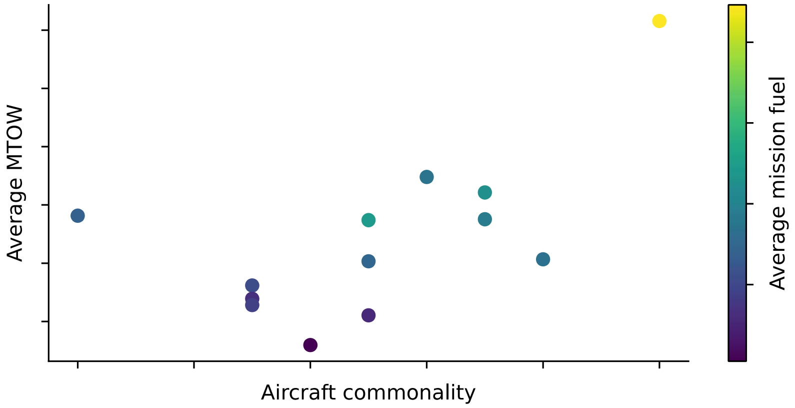
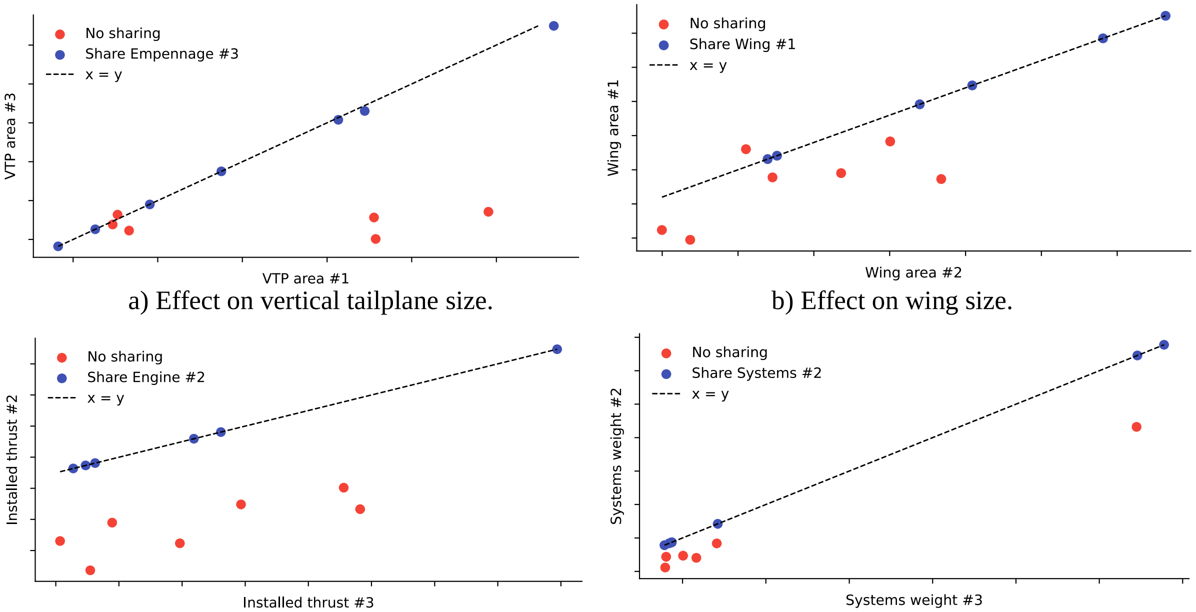


Fig. 19 10-point DOE results plotting commonality (number of shared components) versus family-level average MTOW and average mission fuel burn. The general trend shows an increase in MTOW and fuel burn (i.e. decrease in aircraft performance) for an increasing commonality.



a) Effect on vertical tailplane size.

b) Effect on wing size.

c) Effect on engine size.

d) Effect on onboard systems weight.

Fig. 20 10-point DOE results plotting the influence of component sharing on component size of the smaller aircraft. The general trend shows that components are oversized when shared.

Impact of the three wing design variables, rear spar chordwise location, wing thickness ratio, and sweep angle, are shown in Fig. 21 and Fig. 22. It is shown that, as expected, the rear spar chordwise location influences the maximum lift coefficient that can be reached in landing configuration by enlarging the flaps: moving the rear spar forwards increases the maximum lift coefficient in landing configuration. Increasing the wing thickness ratio also increases the maximum lift coefficient in landing configuration. Furthermore, increasing wing thickness or reducing wing sweep leads to a lighter wing, however also a slightly reduced lift-to-drag ratio (here not shown). Taken together, the three wing design variables enable the optimizer to search for a beneficial trade-off in terms of aircraft performance, even when subject to component sharing effects.

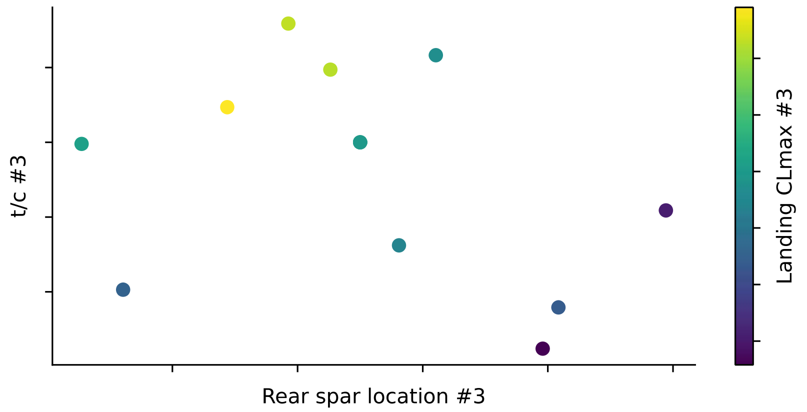


Fig. 21 10-point DOE results plotting the influence of the rear spar chordwise location versus the wing thickness ratio. The general trend shows that for a rear spar located further towards the leading edge or a thicker wing, maximum lift coefficient in landing configuration is increased.

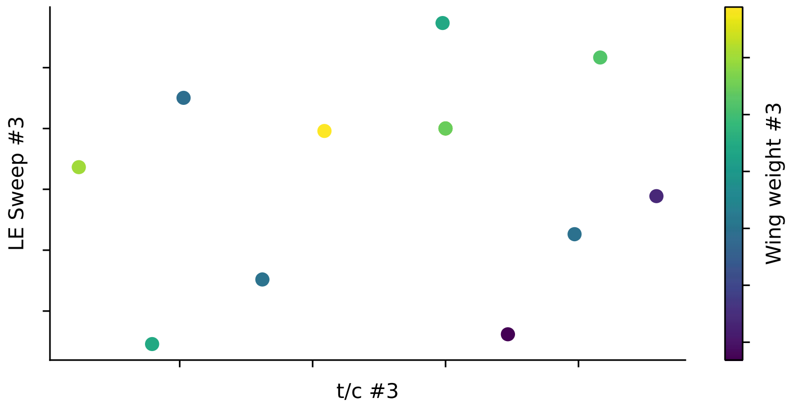


Fig. 22 10-point DOE results plotting the influence of wing thickness ratio versus leading-edge sweep angle. The general trend shows that for a thicker wing or less wing sweep, wing weight is reduced.

From these results it can be seen that the aircraft-design part of the workflow works correctly and produces expected results and trends. In the final phase of the AGILE 4.0 project, the workflow including cost estimation will be further tested and verified using a DOE. Finally, using surrogate-based optimization currently in development as part of the project, the aircraft family will be optimized to find a Pareto front trading non-recurring costs and operating costs. Including cost estimation, it is expected that the level of commonality will correlate with a reduction in non-recurring OEM costs, however also a reduction in aircraft performance and therefore increase in operating costs.

VI. Conclusion

It has been shown that using the methodology and framework developed in AGILE 4.0 it is possible to design a family of business jet aircraft with multiple organizations collaborating to work towards the results. This is done while ensuring traceability from stakeholder needs all the way to the implemented cross-organizational MDAO workflow. All work has been performed in the collaborative online MBSE environment, ensuring that all partners have access to the latest available data.

First, stakeholders, needs and requirements are defined and modeled. From this, the functional architecture and associated logical/physical architecture design space is modeled using ADORE. The architecture design space enables the automated generation of new architectures, and consists of functions mapped to components with Quantities of Interest (QOIs). QOIs represent numerical inputs to or outputs from the architecture analysis framework, and are connected to the MDAO workflow using MultiLinQ. The collaborative MDAO workflow is created from tool input and output definitions using MDAx, and implemented in RCE. Brics is used for cross-organizational data communication.

The MDAO workflow consists of two levels: the aircraft-level workflow that designs one aircraft at a time, and the family-level workflow that integrates the three aircraft designs and calculates family-level metrics. Each aircraft is designed by coupling several higher-fidelity tools to the handbook-based OAD tool openAD. Higher-fidelity disciplines include onboard systems design (ASTRID), maximum lift estimation, tailplane sizing, wing mass estimation (PROTEUS), and mission analysis (AMC). A cost estimation tool is included in the workflow after the aircraft design loop, to calculate non-recurring and operating costs. Results are presented showing the impact of component sharing decisions and wing design variables. In the final phase of the AGILE 4.0 project, optimization will be performed to find a Pareto front trading-off non-recurring and operating costs.

Acknowledgments

The research presented in this paper has been performed in the framework of the AGILE 4.0 project (Towards Cyber-physical Collaborative Aircraft Development) and has received funding from the European Union Horizon 2020 Programme under grant agreement n◦ 815122.

References

[1] A. Jose and M. Tollenaere, "Modular and platform methods for product family design: literature analysis," vol. 16, p. 371–390, June 2005.

[2] T. W. Simpson, "Product platform design and customization: Status and promise," *Artificial Intelligence for Engineering Design, Analysis and Manufacturing,* vol. 18, p. 3–20, February 2004.

[3] J. H. Saleh, G. Mark and N. C. Jordan, "Flexibility: a multi-disciplinary literature review and a research agenda for designing flexible engineering systems," *Journal of Engineering Design,* vol. 20, p. 307–323, June 2009.

[4] P. D. Ciampa and B. Nagel, "Accelerating the Development of Complex Systems in Aeronautics via MBSE and MDAO: a Roadmap to Agility," in *AIAA AVIATION 2021 FORUM*, Virtual Event, 2021.

[5] AGILE4.0, *AGILE4.0 Portal,* 2019.

[6] P. D. Ciampa and B. Nagel, "AGILE Paradigm: The next generation of collaborative MDO for the development of aeronautical systems," *Progress in Aerospace Sciences,* vol. 119, November 2020.

[7] L. Boggero, P. D. Ciampa and B. Nagel, "An MBSE Architectural Framework for the Agile Definition of System Stakeholders, Needs and Requirements," in *AIAA AVIATION 2021 FORUM,* 2021.

[8] L. Boggero, P. D. Ciampa and B. Nagel, "An MBSE Architectural Framework for the Agile Definition of Complex System Architectures," in *AIAA AVIATION 2022 FORUM*, 2022.

[9] J. H. Bussemaker, P. D. Ciampa and B. Nagel, "System Architecture Design Space Exploration: An Approach to Modeling and Optimization," in *AIAA AVIATION 2020 FORUM*, 2020.

[10] J. H. Bussemaker, L. Boggero and P. D. Ciampa, "From System Architecting to System Design and Optimization: A Link Between MBSE and MDAO," *INCOSE International Symposium,* June 2022.

[11] J. H. Bussemaker, P. D. Ciampa and B. Nagel, "System Architecture Design Space Modeling and Optimization Elements," in *32nd Congress of the International Council of the Aeronautical Sciences, ICAS 2020*, Shanghai, 2021.

[12] P. D. Ciampa, P. S. Prakasha, F. Torrigiani, J.-N. Walther, T. Lefebvre, N. Bartoli, H. Timmermans, P. Della Vecchia, L. Stingo, D. Rajpal, I. van Gent, G. La Rocca, M. Fioriti, G. Cerino, R. Maierl, D. Charbonnier, A. Jungo, B. Aigner, K. Anisimov, A. Mirzoyan and M. Voskuijl, "Streamlining Cross-Organizational Aircraft Development: Results from the AGILE Project," in *AIAA Aviation 2019 Forum*, Reston, 2019.

[13] M. Alder, E. Moerland, J. Jepsen and B. Nagel, "Recent Advances in Establishing a Common Language for Aircraft Design with CPACS," in *Aerospace Europe Conference*, 2020.

[14] S. Wöhler, G. Atanasov, D. Silberhorn, B. Fröhler and T. Zill, "Preliminary Aircraft Design with a Multidisciplinary and Multifidelity Design Environment," in *AeroSpace Europe Conference*, Bordeaux, 2020.

[15] S. Chiesa, G. A. Di Meo, M. Fioriti, G. Medici and N. Viola, "ASTRID - Aircraft on Board Systems Sizing and Trade-off Analysis in Initial Design," in *Proceedings of the 10th Research and Education in Aircraft Design Conference, READ*, Brno, 2012.

[16] M. Fioriti, L. Boggero, S. Corpino, P. S. Prakasha, P. D. Ciampa and B. Nagel, "The Effect of Sub-systems Design Parameters on Preliminary Aircraft Design in a Multidisciplinary Design Environment," vol. 29, p. 135–145, 2018.

[17] J. A. Blackwell, "A finite-step method for calculation of theoretical load distributions for arbitrary lifting surface arrangements at subsonic speeds," 1969.

[18] P. D. Vecchia, F. Nicolosi, M. Ruocco, L. Stingo and A. D. Marco, "An improved high-lift aerodynamic prediction method for transport aircraft," vol. 10, p. 795–804, November 2018.

[19] P. M. Sforza, Commercial Airplane Design Principles, Oxford: Elsevier Science, 2014.

[20] F. Nicolosi, P. D. Vecchia, D. Ciliberti and V. Cusati, "Fuselage aerodynamic prediction methods," vol. 55, p. 332–343, August 2016.

[21] D. Ciliberti, P. D. Vecchia, F. Nicolosi and A. D. Marco, "Aircraft directional stability and vertical tail design: A review of semi-empirical methods," vol. 95, p. 140–172, November 2017.

[22] N. P. M. Werter and R. D. Breuker, "A novel dynamic aeroelastic framework for aeroelastic tailoring and structural optimisation," vol. 158, p. 369–386, December 2016.

[23] M. D. Ardema, M. C. Chambers, A. P. Patron, A. S. Hahn, H. Miura and M. D. Moore, "Analytical Fuselage and Wing Weight Estimation of Transport Aircraft," 1996.

[24] T. Lammering, K. Franz, K. Risse, R. Hoernschemeyer and E. Stumpf, "Aircraft Cost Model for Preliminary Design Synthesis," 2012.

[25] K. Schäfer, Conceptual Aircraft Design for Sustainability: Nachhaltigkeitsorientierter Flugzeugvorentwurf, Shaker Verlag, 2017.

[26] A. Page-Risueño, J. H. Bussemaker, P. D. Ciampa and B. Nagel, "MDAx: Agile Generation of Collaborative MDAO Workflows for Complex Systems," in *AIAA AVIATION 2020 FORUM*, 2020.

[27] E. Baalbergen, W. Lammen, N. Noskov, P.-D. Ciampa and E. Moerland, "Integrated collaboration capabilities for competitive aircraft design," vol. 233, p. 00015, 2018.

[28] E. Moerland, P. D. Ciampa, S. Zur, E. Baalbergen, N. Noskov, R. D'Ippolito and R. Lombardi, "Collaborative Architecture supporting the next generation of MDAO within the AGILE paradigm," vol. 119, p. 100637, November 2020.

[29] E. H. Baalbergen, J. Vankan, L. Boggero, J. H. Bussemaker, T. Lefebvre and B. Beijer, "Advancing Cross-Organizational Collaboration in Aircraft Development," in *AIAA AVIATION 2022 FORUM*, Chicago, 2022.

This article has been cited by:

1. Luca Boggero, Pier Davide Ciampa, Björn Nagel. An MBSE architectural framework for the agile definition of complex system architectures . [Abstract] [PDF] [PDF Plus]
2. Erik Baalbergen, Jos Vankan, Luca Boggero, Jasper H. Bussemaker, Thierry Lefebvre, Bastiaan Beijer, Anne-Liza Bruggeman, Massimo Mandorino. Advancing Cross-Organizational Collaboration in Aircraft Development . [Abstract] [PDF] [PDF Plus]