Towards Certification by Simulation with model-based continuous Engineering Processes showcased on eVTOL Application

Marcel Gottschall
Research Associate
Research and Innovation,
ESI Group
Dresden, Germany

Bastian Binder
Research Associate
Research and Innovation,
ESI Group
Dresden, Germany

Alexis Castel
Technical Manager
Customer Experience,
ESI Group
Farmington Hills, MI, U.S.

ABSTRACT

Aerospace industry OEMs and suppliers are progressing their engineering processes and performance to the next maturity level gearing to digital thread solutions. Current challenges like continuous engineering, virtual certification, distributed development, consolidated virtual proving grounds, homologation, digital twin and operational applications, require well informed decision making in a comprehensive, reliable, traceable and customizable environment. In particular, in aerospace domain, with widespread tight collaborative ecosystems between integrators and suppliers, the capability of tracing each decision and its underlying artifacts becomes a key value of an engineering platform.
This paper will outline a middleware approach to reuse generated artifacts and their relationships in a federated engineering environment supporting a "best tool for the job" approach by introducing a layer providing unification and consistency throughout all managed artifacts. Based on an exemplary eVTOL setup, the benefits of integrated data and workflows from specification to virtual design verification are highlighted to motivate their value towards realisation of MBSE methodologies.

INTRODUCTION

With the ongoing digitalization of todays product development and operation, new opportunities arise, but also new challenges need to be handled. Applying incremental automation on digital artefacts (models, processes) and adding AI-technologies like machine learning, shows the direction of future product development, or engineering workflows in general, towards the long-term goal of autonomous processes as shown in figure 1.

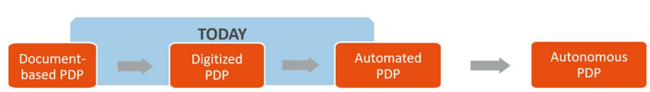


Figure 1. Evolution of product development process (PDP) in general.

On the one hand, these capabilities and possibilities of digitalization rapidly increase the efficiently controllable complexity of a product and its underlying processes. Lifecycles encompasse a varying range of engineering disciplines which typically span many levels of abstraction, in particular within the design phase: From high-level abstractions, such as stakeholder identification or requirements management, over system design endeavors to low-level abstractions such as 3D component design, multi-domain simulations or different analysis methods applied up to manufacturing and operation. On the other hand, traditionally these processes, methods and disciplines are fundamentally isolated based on each discipline's view of the specific system of interest. This separation is further reinforced by the usage of different, mostly incompatible, authoring tools, workflows and even languages between these *silos* and usually diminishes the potential of popular digital twins = models of the product.
Hence, recent polls along all industries identified digital continuity and potential of a holistic handling of systems (usually refered to as model-based systems engineering MBSE) as one of the most pressing points affecting their daily business outcomes and gains. Achieving more agility and flexibility, thus resulting in reduced time to market and development costs is a key request from industrial partners, in particular in aerospace with its high certification demands. Moving processes and decisions of the early stages into the virtual domain will enable significant improvements if consistency and strong traceability are maintained, figure 2.

As a CorePartner to CleanSky2 project MISSION[1], (Ref. 1), ESI contributes to developments of a software toolchain that supports seamless integrated workflows within aerospace design processes along the different levels of system decomposition (aircraft, system, component) and system integration from requirements to certification. It demonstrated a multi-tool platform implementation using model-based integration techniques to achieve fully traceable, early stage requirements

Presented at the Vertical Flight Society's 78th Annual Forum & Technology Display, Ft. Worth, Texas, USA, May 10–12, 2022.

[1] https://www.clean-aviation.eu/clean-sky-2

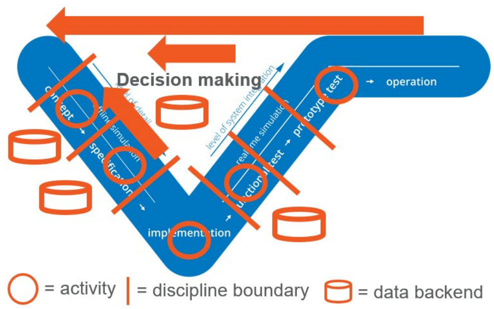


Figure 2. Simulation based artifacts, decision making and frontloading throughout the development process.

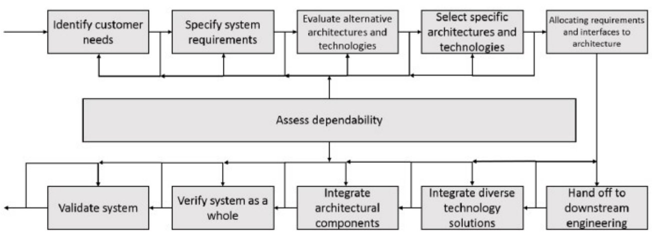


Figure 3. Multilevel integration of complex system design activities handled by SE approach, (Ref. 2).

verification by virtual testing, (Refs. 3, 4).

Based on an exemplary eVTOL setup, the benefits of such tracable integration from specification to virtual design verification are demonstrated in this paper to adress the major challenges of optimized endurance and safety considerations. It will outline a middleware approach to reuse produced artifacts and their relationships in a federated engineering environment. By introducing a layer providing unification and consistency throughout all managed artifacts, the presented approach evolves simple unmanaged processes like ”import-export-integration” towards an efficient managed digital thread application. Thus, enabling authorized certification credits by simulation is a major milestone on the way to future *zero physical prototype* visions.

ENGINEERING AND INTEGRATION TECHNOLOGY

The increasing complexity of systems, not only in aerospace, led to definition of methodologies like systems engineering that approach the system of interest in an holistic view, and their model based derivates using the systems digital representation. Similar to all these processes are the key activities shown in figure 3. Such methods require a certain level of integration or chaining of artifacts and processes (”breaking the silos”) along the lifecycle shown in figure 2. Hence, modern product development itself is evolving towards a comprehensive, continuously engaged approach to support engineers at all levels facing the challenges of an increasingly complex landscape of specification, definition, simulation and validation artifacts. Keeping track requires well informed decision

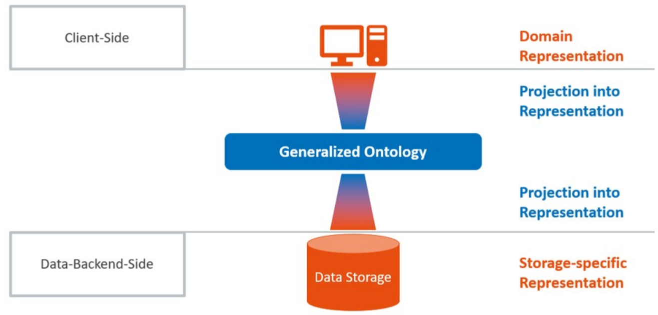


Figure 4. Mapping of data ontology in the middleware layer.

making and subsequent tracing, particularly in aerospace domain, with widespread tight collaborative setups between integrators and suppliers.

To overcome the divide described above and to enable collaboration within multitool and multipartner environments, ESI implemented a technology supporting the main goals/paradigms

- Tool agnostic approach (generalized data) for a customers “best tool for the job” choice and to avoid a critical single vendor dependency as it is characteristic for monolythic solutions recently available on the market
- Open standard based, adaptable, scalable technology
- Full traceability by linked and managed data approach
- Reuse of data and single point of access
- Agile techniques, Continuous Engineering, Continuous Integration

Technically simplified, a unifying layer is placed in between the user or client side as data creators or consumers, and the backend side as the data storage, figure 4. Within this microservice layer based on OSLC[2], the data is managed, linked and exposed to the participants of the process according to the required information. In case of multiple data backends, the service layer acts as a single point of access to each client just exposing the relevant parts of the information. Developed in previous projects and proven in various customer activities, the approach has shown its qualification to handle the multilevel integration of artefacts (data) - tools - maturity levels - domains for a system design phase with high architectural and performance definition workload - like it is the scope of the following sections. Aside the pure data exchange aspect, it provides a high level of flexibility regarding necessary capabilities like user and change management or workflow definition. Depending on the data backend tooling in this regard, such functionality can be also provided by the service layer.
Even though it is not part of the present paper, it should be remarked, that the validity and capabilities of the approach

[2] https://open-services.net/

extend beyond the product development into prototype production and testing, and even into manufacturing and field operations by providing a virtual twin platform. It enables the required integration of the ideal design model (virtual twin), and the model of the real system after production and in operation augmented and modified by real measurements from the field (digital twin). With this integration technology, handling and exposing the relevant data and information throughout the whole lifecycle, and adding AI methods (e.g. machine learning) onto these data, novel real time twin applications like monitoring and predictive maintainance become possible and have been demonstrated.

For simplicity, the illustration of the process integration in this paper will be focussed to a key segment of the left part of the development cycle, figure 2, in a generic top down approach:
After specification of the system in a requirements management phase, a systems architecture need to be designed and verified on a functional level. This step is typically executed in a SysML[3] tool where more or less formal processes will result in a logical and technical decomposition of the system based on the given requirements. The aspects of interest are seamlessly propagated via the backend as requirements for the next design step, which is the sizing and performance validation of the various components and their integration. Usually described as behavioral modelling, that *system simulation* step becomes a key virtual decision gate for subsequent efforts of "classic" simulation activities like FEM or CFD. Outputs are design parameters on subsystem and component level like a mechanical power demand or a diameter for a detailed 3D design and analysis. Again, the available artefacts are provided to the backend and only relevant information will be *consumed* and exposed by the corresponding 3D tooling. This overarching process will be enabled by the middleware layer and followed in a similar way throughout the integration phase of the lifecycle. Incremental virtual testing methods from Model-in-the-Loop up to Hardware-in-the-Loop and corresponding processes significantly benefit from the already available information about the system of interest.

A specific visual example about such *Agile Design* application for sizing and performance verification considerations and more detailed explanations are subject of the next section.

SAMPLE USE CASE

As introduced above, an implementation of MBSE methodology enabling seamless integration and virtual verification of requirements is applied to reduce development time and costs of an eVTOL system. The holistic model-based engineering approach demonstrated in the paper depicts a certain part of the full lifecycle as shown in figure 5 where the system architecture, sizing and performance are defined - as critical steps most benefitting from an early stage, virtual verification. The analysis of these aspects under various environmental conditions results in a wide range of test case scenarios where physical prototype testing methods cannot serve sufficiently, in particular for safety considerations like emergency handling.

[3]Systems Modelling Language

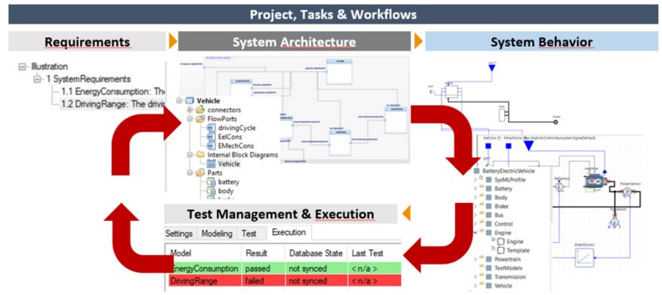


Figure 5. Sample design workflow with focus on architecture sizing and performance verification activities.

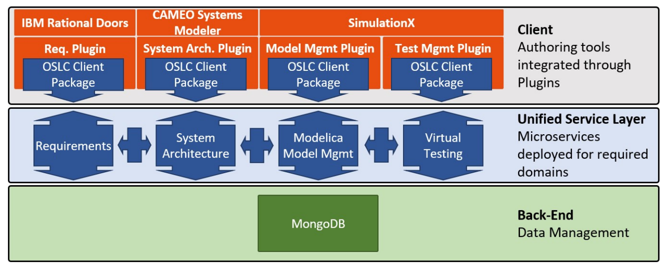


Figure 6. Applied multitool environment integrated by middleware layer supporting systems engineering steps.

Task and Tooling

Driven by a task defined as "**Design a semi-autonomous eVTOL for UAM application with meaningful payload and endurance fulfilling corresponding certification regulations like EASA MOC SC-VTOL**", a sample multitool environment as shown in figure 6 is set up. It emphasizes the chaining capabilites by integrating multivendor tooling as it is common in highly collaborative workflows along the lifecycle. To fulfill the performance and safety verification task in a traceable, closed loop fashion, here it is comprised of

- *IBM DOORS* as a very popular example for defining and managing requirements in a product development context, in particular in aerospace domain
- *No Magic CAMEO Systems Modeler* as SysML tool representing systems engineering tasks leading to an appropriate system architecture. It should be pointed out, that ESI as solution partner for virtualisation but neither an eVTOL OEM nor a supplier of physical components has no specific, publishable data for *real* eVTOL systems. Hence, the demonstration is based on a given system defined in an internal industry solution showcase and the activities on architectural level do not describe formal processes for architecture exploration in SysML like

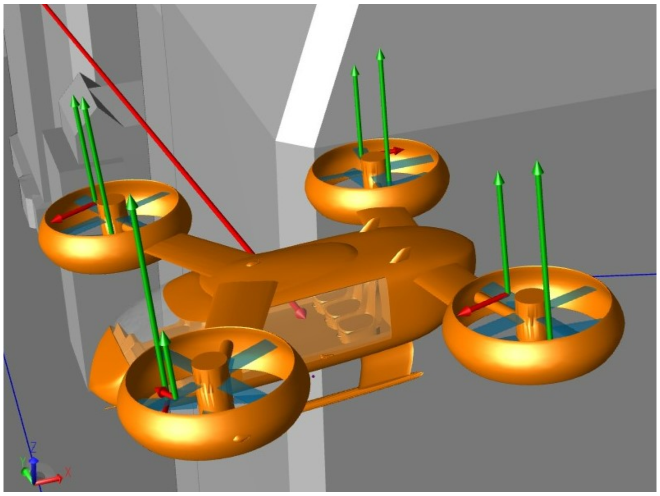


Figure 7. 3D visualisation of the given vehicle architecture.

SYSMOD, (Ref. 5). Major architectural properties are already fixed in this example.

- *ESI SimulationX* as a system simulation tool using acausal physics based modeling implementing the open standard Modelica[4] language for dynamic analysis of multidomain physical and cyber-physical systems. Aside of the solver, it provides the user with domain specific libraries of models and various model integration and exchange capabilities support like FMI[5].

System under Design

With the focus on the engineering workflow and qualitative decision making along the different variants of the system, any description of equations is avoided for simplicity.
As it is shown in figure 6, an open source database is applied in the backend without any workflow convenience features. Thus, the virtual testmanagement and -execution activities are provided by the system modeling tool, supported by the corresponding service capabilities. However, it should be remarked that various setups with more complex backend systems like PLM, (Ref. 6) or dedicated test authoring and testmanagement systems, (Ref. 7), have been realized.

Dynamic multidomain system models in systems engineering context enable virtual requirements verification. Automatically derived from previous architectural options analysis, these models allow sizing (e.g. endurance), performance validation (e.g. cooling system, flight dynamics, vibration) and safety analysis techniques (e.g. emergency landing) depending on various settings of parameters, transient scenarios and control strategies in the frame of model-in-the-loop virtual tests.

In this paper, the architecture shown in figure 7 shall represent the "vision" of an OEM regarding a battery electric-driven VTOL for air taxi application. To limit the dimensions, it shall rely on 4 counterrotating rotors using fixed pitch blades and shall provide seven seats including crew. Major challenges and key aspects to be fulfilled regarding the vehicle design are defined by regulations released by FAA, EASA a.o., as mentioned above. An exemplary selection of requirements that shall be verified here by virtual testing is shown below:

R1 "The thrust and drivetrain design should allow for lift off at MTOW of 1.6 tons (including payload) while maintaining maximum eLoads (current) on battery and electrical system."

R2 "The applied cooling system shall maintain safe temperatures in the battery under maximum load and most adverse conditions for at least 5 min."

R3 "The automatic flight control system shall be able to maintain attitude and follow a predefined trajectory within allowable deviations of 15 deg and 10 m, respectively, under normal operating and given environmental conditions."

R4 "The automatic flight control system shall be able to detect an abnormal situation of a single rotor power loss of 20 % and execute a safe emergency landing under given environmental conditions."

R5 "The battery voltage drop for peak current (= 5 times maximum current) must stay below 200 V."

In the listing above, R1 - R4 represent requirements that demand the full vehicle system running a certain scenario. These simulation results also provide the load cases at different subsystems and components like R5, to enable *real* stimulation of the virtual system under test. In addition, they serve as inputs to subsequent 3D design and analysis tasks like noise qualification - particularly important in urban area applications - instead of a "lab-like" testing with artifical load profiles. In the sections below, the requirements are referenced by R1..R5.

In order to enable proper system level verification, the performance model implementing the logical architecture includes

- 3D MBS mechanics (fuselage, individual rotorblades representing aerodynamics)
- 1D rotational mechanics (drivetrain including gearboxes)
- Electrical components (battery circuit and brushless DC machines)
- Thermal components (cooling of battery and systems)
- Signal components and state machines (attitude and position control system)

Flight dynamic test scenarios are modeled by user defined trajectory (attitude and position over time) and a 3D wind distribution representing a certain requirement or testcase. The complex system simulation model provides realtime capability, which is important for high maturity level verification

[4]https://modelica.org/modelicalanguage.html
[5]https://fmi-standard.org/

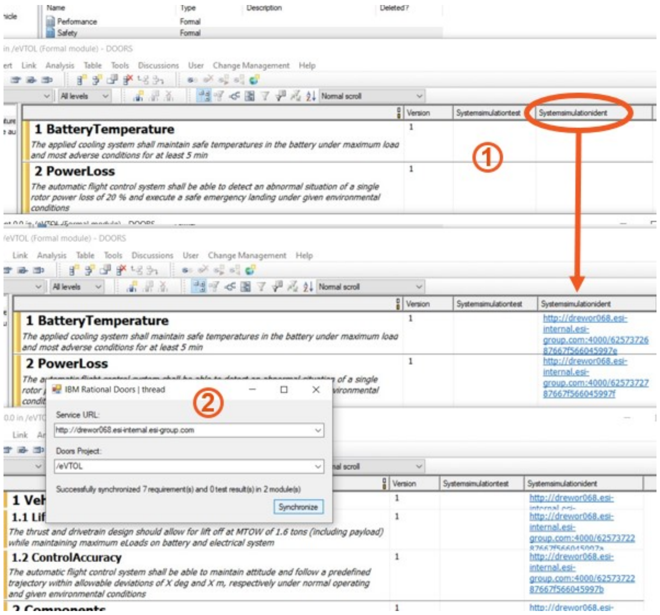


Figure 8. Requirements collection and backend synchronization.

methods like Hardware-in-the-Loop, and a benefit for simulation based early stage design steps like design space exploration or system optimization requiring hundreds of simulation runs. Moreover, formal methods like FTA (Fault Tree Analysis) and SRA (System Reliability Analysis) based on the system architecture and fault augmentation are intrinsically supported.

INTEGRATED DECISION MAKING

A demonstrated walkthrough the process described above is used to visualize the benefits of user convenience, time and cost reduction as well as artefact traceability for the challenging eVTOL development phase.

Design Cycle

A typical product lifecycle starts with the customer needs and their translation into requirements for the system to be developed, figure 3. In this example, the requirements are solicted in a DOORS project as shown in figure 8. Each requirement contains an empty attribute for identification (*Systemsimulationident*) and the verification by simulation result (*Systemsimulationtest*). Once the requirements are collected, they are synchronized to the data backend utilizing a plugin providing access to the service layer (see figure 6) to make the information available for other clients participating to design process. During the data creation, the unique identifier is generated in the backbone and written to the corresponding attribute (column) in the requirements management tool.

Going down the design phase of the v-cycle model, the next step is the architectural design, usually refered to as the systems engineering and executed in a SysML tool. Depending on the complexity of the system under design, the process

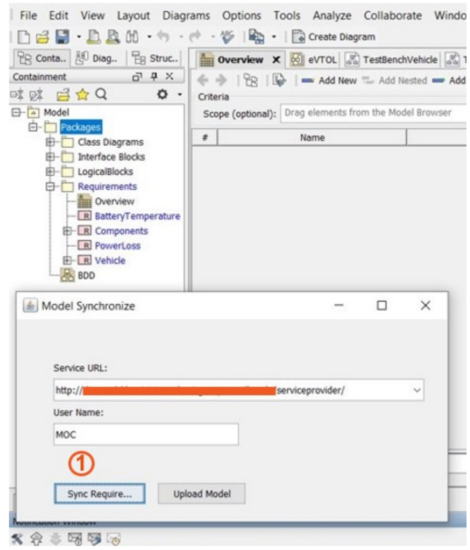


Figure 9. Requirements synchronization into empty SysML model.

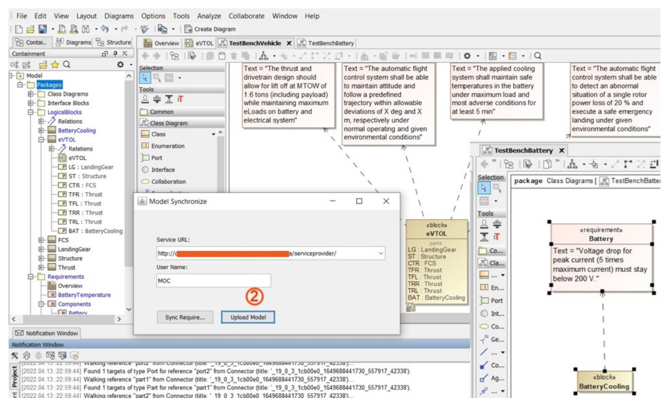


Figure 10. Linkage between requirements and system components and backend synchronization.

starts with functional architecture that satisfies the requirements. Based on these results, a logical and technical architecture is derived that implements this functional design and is the required output for subsequent steps. As mentioned in the section above, in this example this kind of iterative step is skipped since the top level architecture is already fixed due to availability of data.

Starting with an empty model, the requirements are loaded to CAMEO, again utilizing a tool specific plugin, see figure 9. After a more or less complex formal process of designing the logical architecture using stereotyped connections, the relevant components of the system are linked to the corresponding requirements they need to fulfill, as shown in figure 10. In this example, it is the overall vehicle for R1 - R4 and the battery component fulfilling R5, compare section above. This is the key information for test generation later on, as well as for establishing trace links between the artefacts. Finally, all the information contained in the SysML model is synchronized to the backend, however it should be emphasized that only rel-

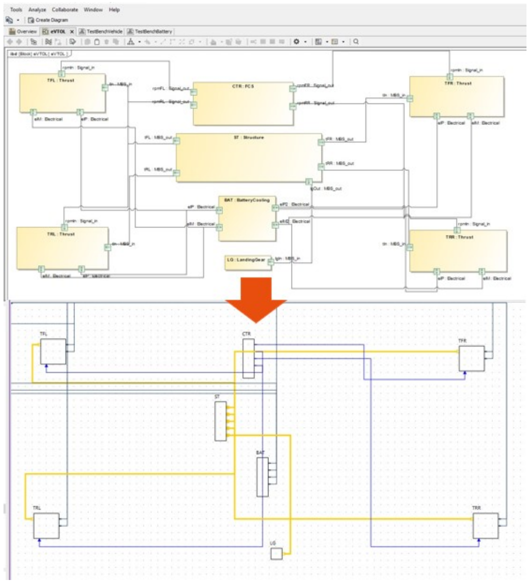


Figure 11. Logical architecture defined in SysML (top) and transferred to Modelica (down).

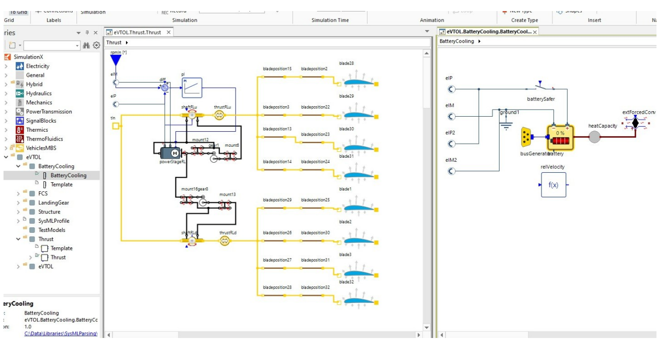


Figure 12. Components modeled with physical behavior.

evant parts of the data required by specific design steps are exposed by the service layer.

Following the top down approach, the empty logical system architecture need to be enriched with physical behaviour to enable performance simulations. For this reason, the information of the system decomposition is transferred to SimulationX, see figure 11 by connecting to the database, extracting the relevant data and automatically generating a corresponding library within the system simulation tool using a plugin. With this automation, the simulation engineer is provided with data that is both, limiting his degree of freedom regarding modeling since he cannot change the components and their connection type, but also increasing the useability and user convenience since all required information is available in his tool. In a next step, the logical components behavior is modeled using available libraries as the common activity of the simulation engineer. See figure 12 for the example of the thrust unit comprised of electrical machine, gearboxes and drivetrain and rotors (left) as well as the battery pack and the passive convective cooling system.

After synchronizing the models with the backend, the system shall be verified against the requirements at the current maturity level by virtual model-in-the-loop testing. This early stage verification enables informed design decision and avoids expensive rework in later stages. As mentioned above, the functionality of testing is also executed by the system simulation tool. For this reason, the plugin is listing all testcases that can be found in the backend based on the information provided in the architectural step before and that can be verified by performance simulation, see figure 13. Each testcase is testing a specific component under a certain scenario defined by the requirement. The plugin automatically generates a testmodel for each testcase by adding the corresponding component of the system as a unit under test (uut). Next, the simulation engineer augments the testmodel with appropriate stimulation and evaluation according to the specific requirement linked in the testmodel. Figure 13 visualises the method for the "lift off verification test" (R1) with corresponding scenario models for trajectory and model parameters (top left), as well as electrical circuit current evaluation under specific condition of zero ground contact force (down right). SimulationX also provides dedicated modeling libraries for such purpose. Again, the finished testmodels are synchronized with the backend and ready for automatic test execution.

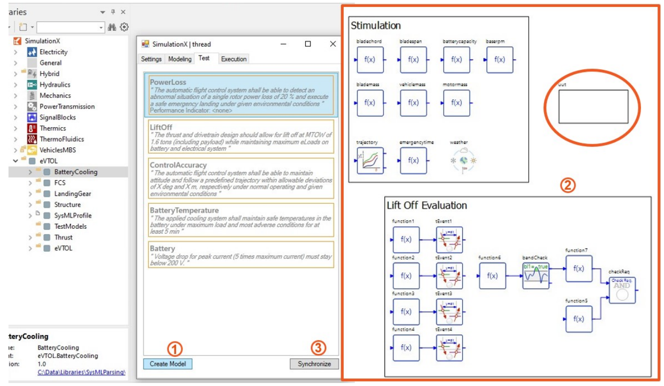


Figure 13. Automatic test model creation, scenario and evaluation augmentation and backend synchronization.

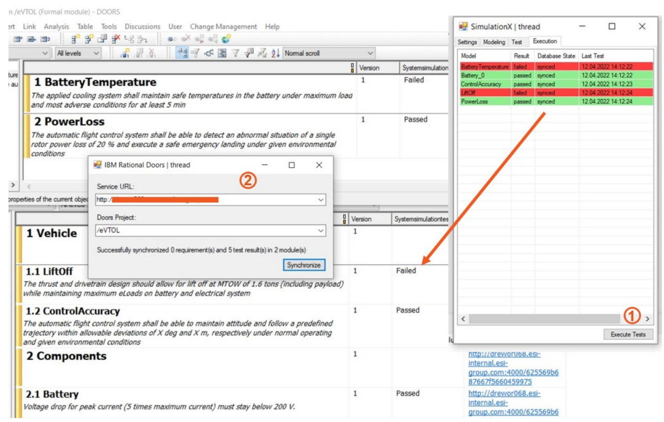


Figure 14. Automatic test execution, backend synchronization and requirements verification.

In a final step, from a process perspective, the testmodels - representing a specific version or variant of the system under design - need to be simulated to derive the testverdict whether the requirement is fulfilled or not and add this information to the backend. As shown in figure 14, in this example the test execution is handled by the plugin. However, the model execution can also be automatically triggered from external test management systems like it was demonstrated in (Ref. 7). After the model execution, the test verdicts are added to the testcases in the backend and synchronized to the corresponding attribute in the requirements management tool, figure 14. It shall be remarked, that all tracelinks between the various artefacts automatically stay intact in the backend as a necessary prerequisite for gaining certification credits by simulation.
Below, the described test execution will be utilized to verify different design variants on subsystem and component level regarding their performance. Only in case of *failed* results, a deeper "visual" analysis of the reason using the model is taking place in order to improve the system. As already mentioned, the design iterations in this paper will be limited to the system performance level for demonstration purpose since the system architecture in terms of number of rotors and electrical machines, rotor size and battery-driven setup is frozen. However, in real integrator-supplier processes, the iterating loops will also overleap the discipline boundaries, e.g. a system level simulation result demands an architectural change like an additional connection in order to meet a certain requirement. These workflows and their level of automation are usually customer defined and may require impact analysis on system structure.

Base Design

Before starting the performance analysis using the system model, it should be pointed out that CFD calculations by *ESI VisualCFD/OpenFOAM* support the physics modelling. On the one hand, preliminary studies of the aerodynamic lift, drag and moment characteristics of the given symmetric rotorblades are used to model the aerodynamic behaviour as a function of angle of attack. The full 3D calculations of the staged rotors confirm the counterrotating setup with respect to efficiency for a given distance between the rotors. With the CFD results, different performance of top and down rotor can be considered in the system model representation. Moreover, they showed that the blade angle of attack is almost independent from rotational speed due to the suction effect because of the casing around the rotors, see figure 7. This fact however eases the modeling in the system model. On the other hand, CFD provides a (steady state) 3D wind distribution acting on the vehicle defined for the specific scenario of an urban area flight as shown in figure 15.

Applying these environmental conditions, the baseline design variant of the eVTOL model is used to execute the different testcases as described above. First simulations shall prove the capability of "lift off" and allow subsequent tests of flight control system for attitude and position. Figure 16 shows the result of a corresponding test run with the baseline model. Since

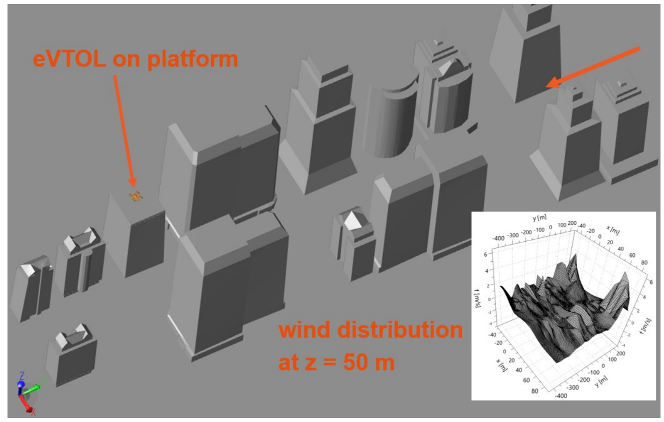
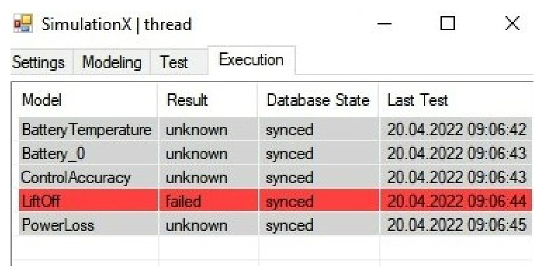


Figure 15. Urban test scenery and examplary wind distribution from CFD.


SimulationX | thread

Settings | Modeling | Test | Execution

Model	Result	Database State	Last Test
BatteryTemperature	unknown	synced	20.04.2022 09:06:42
Battery_0	unknown	synced	20.04.2022 09:06:43
ControlAccuracy	unknown	synced	20.04.2022 09:06:43
LiftOff	failed	synced	20.04.2022 09:06:44
PowerLoss	unknown	synced	20.04.2022 09:06:45

Figure 16. Test results for the baseline design.

the initial requirement of "lift off" was not passed, the other requirements have not been tested during the simulation as flying is an initial condition for them. Looking at the transient results, figure 17 visualizes the reason for the failed requirement. The vehicle is obviously not capable of following the given vertical trajectory, it is not lifting off at all. Hence, the landing gear contact forces to the ground do not disappear which is in turn the condition for the other requirement tests. Minimal movement in X, Y and yaw direction is monitored as a result of the wind acting on the vehicle at the platform and the predefined sliding friction force on the landing gear contact points.
Looking similar to the shown altitude preset curve, figure 17, the thrust control saturation of the flight control system visualizes the correct function by trying to increase the rotational speed of the electrical machines to gain more lift. From the CFD calculations it is known, that a rotational rotor speed of around 1.200 rpm is required to generate lift to hover the eVTOL at overall mass of 1.6 t. However, figure 18 shows the rotational speed of the rotors does never exceed 600 rpm, which is caused by strong voltage drop in the electrical circuit.

Further analysis with the dynamic model shows - among others - that the major reason are the inefficient rotors with a very high power demand for a given rotational speed (poor lift/drag ratio known from CFD). High torque, to be provided by the electrical machines, results in the shown voltage breakdown from battery circuit and thus, very limited rotational speeds. Various tests with different settings of the electrical

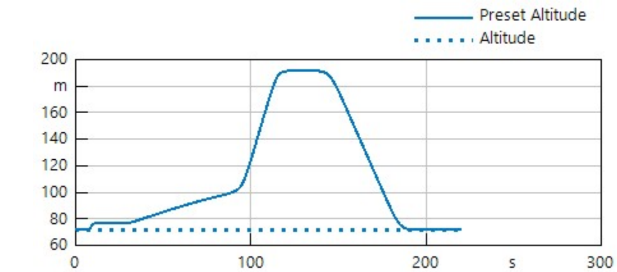


Figure 17. Analysis of the eVTOL altitude compared to preset.

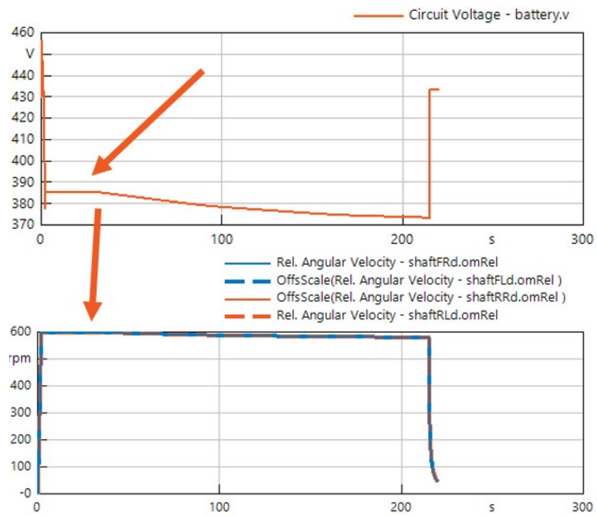


Figure 18. Analysis of the battery voltage and resulting rotor speed.

drivetrain and batterypack show, this effect cannot be circumvented. Neither a higher battery voltage by a different layout nor higher gearbox ratios in order to reduce the required torque from the machine (but further lowering the rotor speed) will solve the issue. Hence, requirements verification with the baseline geometry can be stopped and another design loop is required.

Improved Design

As stated above, a different aerodynamic rotor design is the only solution to solve the problem of the insufficient lift generation within the given boundary conditions regarding architecture, rotor sizing and MTOW. In fact, it is not surprising since the initial rotor blades used simple, symmetric NACA foils for manufacturing reasons with very low aerodynamic efficiency. In particular for battery electric mobility, highest possible efficiencies at all levels are necessary to reach useable endurances.

In this example, the airfoils of the blades are changed to asymmetric ones with twice the lift/drag ratio, which is much more meaningful for a rotor just working into a single direction (upwards). With a first view on figure 20, it can be stated that the major goal is fulfilled - the vehicle is able to fly. From

Model	Result	Database State	Last Test
BatteryTemperature	passed	synced	20.04.2022 09:18:00
Battery_0	passed	synced	20.04.2022 09:18:01
ControlAccuracy	failed	synced	20.04.2022 09:18:02
LiftOff	passed	synced	20.04.2022 09:18:02
PowerLoss	failed	synced	20.04.2022 09:18:03

Figure 19. Test results for the improved rotor design.

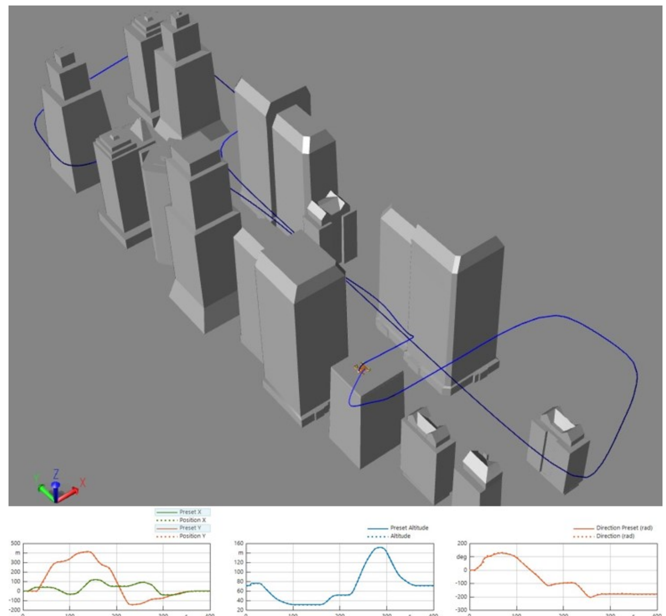


Figure 20. Flight path (top) as a result of the given trajectory (down) and environmental conditions.

a qualitative point of view, the eVTOL keeps track under the predefined 3D wind conditions. Several analysis in the subsystems (not shown) verify

- less power consumption due to the lower torque requirement, lower eloads as well as feasible battery and machine parameters
- less thermal power and less cooling demand

compared to the "beyond the edge" parametrization of the baseline model in order to get it flying. Thus, integration between controller and system is much smoother, providing a higher capability to compensate external loads like wind gusts acting on the vehicle along its flight path.

Coming back to the quantitative requirements regarding position accuracy, figure 19 shows a failed test run for the corresponding testcase, even though the qualitative deviations between preset and actual position are remarkably small along the 400 s scenario, see figure 20 low. A detailed look into the evaluation in figure 21 (top) shows, that pitch angle, yaw deviation and Y position left the allowed envelope defined by

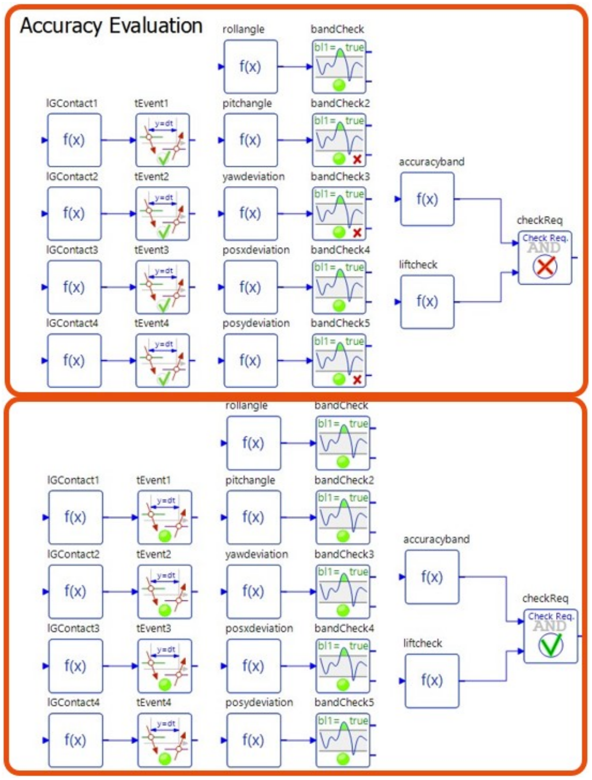
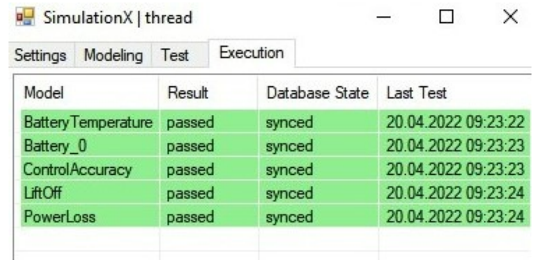


Figure 21. Evaluation with narrow boundaries (top) and wider range (down).



Model	Result	Database State	Last Test
BatteryTemperature	passed	synced	20.04.2022 09:23:22
Battery_0	passed	synced	20.04.2022 09:23:23
ControlAccuracy	passed	synced	20.04.2022 09:23:23
LiftOff	passed	synced	20.04.2022 09:23:24
PowerLoss	passed	synced	20.04.2022 09:23:24

Figure 22. Test results for the final controller design.

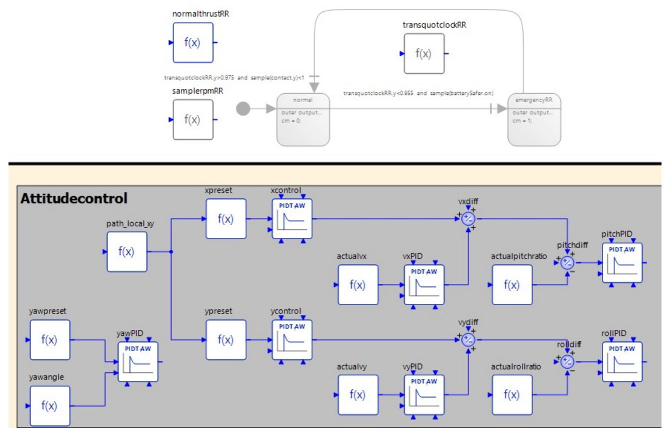


Figure 23. Improved controller design.

requirement R3, at least once during the flight scenario (red crosses in the corresponding blocks). This behaviour is a result of the flight control parametrization and the external conditions and lead to the overall "failed" verdict at the right. Depending on the use case and potential certification demands, it needs to be decided whether the control system is "sharpened" with detrimental effects on stability, or a wider envelope is acceptable which in turn would change the requirement. After applying the second option in this example, the verification changes to "passed" as shown in figure 21 (down).

Using the maximum current charged from the battery during the scenario simulation, the testmodel for requirement R5 as battery component test is stimulated with a corresponding peak current to verify the allowable voltage drop of 200 V. Also, with the iterated subsystem and component parameters, the maximum heat test of requirement R2 can be conducted. In this testmodel, a different trajectory (lift off - steady hovering at constant altitude - landing) is used to represent most adverse conditions for a passive convective cooling system because of high eloads and almost no airflow at the fuselage bottom (no vehicle motion, calm wind behind the taller building). However, it is shown that the passive cooling is able to keep the temperature rise in the battery below 60 degC within the 5 min scenario.

Apart from the normal operation behavior discussed above, the improved design is still not passing the emergency handling requirement R4, as shown in figure 19. To simulate the emergency situation of a powerloss on a single rotor unit, the rotational speed command of the electrical machine (in this example rear right unit) is modified at a user defined simulation time and limited to 80 % of the actual mean speed. It should be remarked, that such phenomena can be easily modeled using the fault augmentation capabilities in *SimulationX* in a way that a faulty gearbox results in higher friction finally leading the reduced speeds. However, for simplicity, the effect is directly added to the model in this demonstrator. With a closer look to the corresponding test model, it turned out that the applied two stage cascaded attitude control system is obviously not able to handle the significant impact to the flight dynamics in case of single rotor unit power loss. The model gets unstable regarding attitude because of following the presets and simply crashes into ground or building. Hence, another loop is required.

Final Design

It turned out in this virtual design process, that the controller stategy of a cascaded position and attitude control is not appropriate for specific abnormal operation like a partial single engine loss. In such situation, position control should be avoided in terms of following a predefined path which might be impossible because of the asymmetric thrust generation, finally a controller is overcharged and vehicle gets unstable.

Therefore, a statemachine shown in figure 23 is added to the model to enable the determination of different operating modes depending on specific criteria. In this case, the

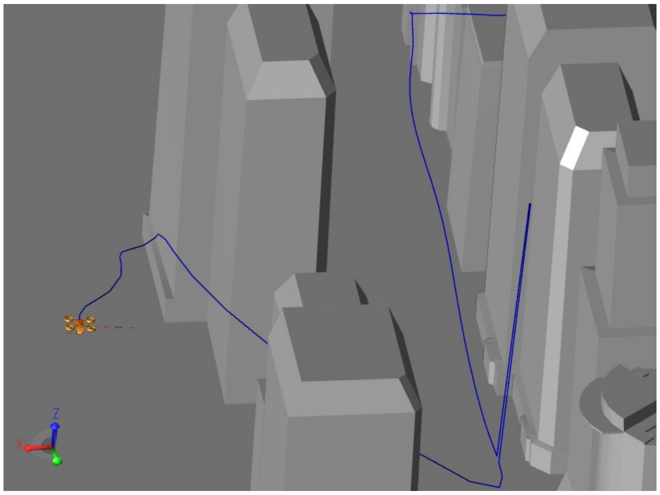


Figure 24. Flight path and safe landing after emergency situation.

statemachine monitors the rotor speeds and switches to emergency operation when a rotor falls below a threshold of 95 % of its actual speed command. Within the present scenarios, the normal deviation between command and actual speed is less than 2 % because of the rotor inertia. The statemachine switches back to normal operation when all rotors step over 97 % desired speed value. Utilizing this operating mode state, an additional cascade for vehicle velocity control is added to the controller in figure 23. For the particular abnormal situation in this example, the controller strategy changes to

1. stabilizing attitude regarding pitch and roll angle
2. maintaining current vehicle position (zero velocity)
3. starting controlled descent with a given rate until safe touchdown

Because of the physics, the vehicle starts to rotate around its vertical axis when a rotor unit generates less torque than the others, hence the overall momentum is not balanced anymore leading to a "rotating touch down". However, the rotational speed of this phenomena depends on the vehicle mass versus rotor momentum as well as starting altitude.
Here, the emergency is introduced at 55 s to the "standard" trajectory discussed above. The resulting behaviour of the vehicle is shown in figure 24. After the power loss happens, it takes around 20 s where the vehicle stabilizes its attitude while it is moved through the buildings because of the wind situation, compare figure 25 (top). Once the stable position is established at around 80 s, it starts the descent with a constant rate leading to increasing deviations to the predefined trajectory. The continously increasing yaw angle in this phase, figure 25 (down), visualises the rotation of the vehicle also visible by the corkscrew line in the last phase of figure 24.

With this extension to the model, all requirements are fulfilled in a final test run as shown in figure 22. Since the normal operation parameters do not change, the modification only affects the result of the emergency testcase R4. Taking this into

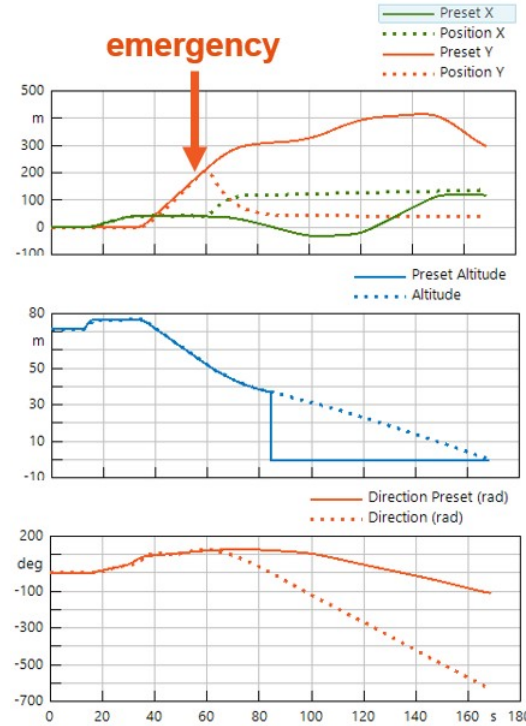


Figure 25. Trajectory after emergency situation.

account, a "final" setup of the system is determined providing sizing parameters and derived component requirements to subsequent design steps.

Lessons Learned

Apart from the flight dynamics, the battery sizing and cooling is a crucial aspect for eVTOL represented by corresponding requirements. With realistic machine parametrization, the battery drained down to 42 % in the demonstrated 400 s scenario, which is not surprising taking into account the very large size and mass of the vehicle. At the moment, the battery capacity is set to 100 kWh resulting in a battery mass of around 500 kg. Increasing the battery capacity will increase the mass as well, which requires higher electrical power, which is in turn draining the battery faster.

With the demonstrated design process, the well known bottle neck of battery technology for electrical mobility, and in particular for aviation, is confirmed by the simulations. All in all, a design based on rotor lift only is questionable for electric vehicles with respect to sufficient endurance.

CONCLUSIONS

The present tool-agnostic integration approach successfully created a "single-point-of-access" with the service layer enabling collaboration with previously incompatible authoring tools without having to rely on unmanaged export/import schemes. Through continuous synchronization of the behavioral model and automated test execution, continuous engineering can be leveraged within the product development

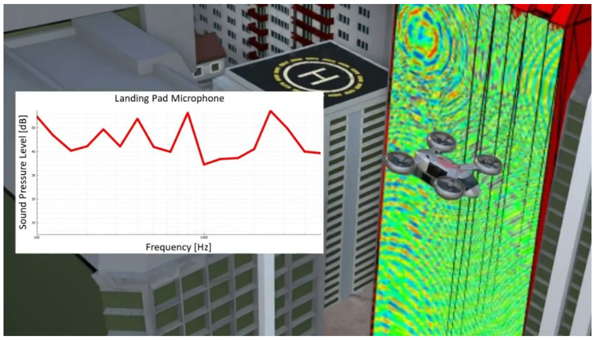


Figure 26. External accoustic analysis driven by the load-cases from performance model (screenshot).

while a strong traceability of decision and underlying data is maintained.
A federateable integration solution allows close mirroring of the users IT infrastructure and high degree of modularity enables the solution to target exactly the defined use cases. In addition, a unified, consistent service layer to handle all communication with authoring tools furthermore enables rapid extension of the environment and builds a viable foundation for automatisms and analysis across discipline boundaries.
Taking this into account, the solution supports not only the demonstrated particular part of the design phase where behavioral models are derived from architectural decisions. These physics based models significantly improve the development process through frontloading by virtual testing to verify performance aspects like lift off verification and maximum eloads on battery or controller design. Aside of such sizing analysis tasks, they serve also for early stage optimization and trade off studies on vehicle trajectory and energy consumption. Also safe emergency situation verification, potentially extended by detailed sensor models for autonomous vehicle systems, is a key benefit of the integrated model-in-the-loop virtual testing for eVTOL applications.

As part of work in progress, the validity and capabilities of the shown ESI chaining implementation extend into detailed 3D analysis and design like CAD for manufacturing or accoustics for noise qualification. Driven by realistic load profiles from full scenario analysis, not only accoustics design and certification as shown in figure 26 is enabled, but also trajectory optimization with a given technical system particularely for urban applications, as one of the key topics in the current european CleanAviation program, becomes feasible.

On the path towards more automatic steps in develpment processes, current activities of ESI in a project called *CYBERTECH*[6] look into features for semi-automatic test generation from information already available in SysML level, thus further increasing the usability and efficiency of workflows and processes around complex, safety critical systems like modern eVTOL.

Author contact: Dr. Marcel Gottschall
marcel.gottschall@esi-group.com

[6]https://www.ase-cybertech.de/

APPENDIX

Table 1. Glossary of selected abbreviations used in the text.

MBSE	model based system engineering
OSLC	open services for lifecycle integration
EASA	European Union Aviation Safety Agency
MBS	Multi Body Systems
MTOW	Maximum Take Off Weight
UAM	Urban Air Mobility

ACKNOWLEDGMENTS

Parts of the described work and technical solutions have been developed under and funded by CleanSky2 MISSION project. The support is gratefully acknowledged.

REFERENCES

1. Valdivia-Guerrero, V., Foley, R., Riverso, S., Govindaraju, P., Burgio, G., Mangeruca, L., Ferrari, A., Gottschall, M., Blochwitz, T., Himmler, A., Bloch, S., Taylor, D., Hayes-McCoy, D., “Modelling and Simulation Tools for Systems Integration on Aircraft” SAE 2016 Aerospace Systems and Technology Conference, DOI 10.4271/2016-01-2052

2. Douglass, P., *Agile Systems Engineering*, Morgan Kaufmann, Burlington, Massachusetts, 2016

3. Binder, B., Gottschall, M., Blochwitz, T. “A novel framework approach enabling Model-based Requirements Validation and Verification” AIAA Aviation 2018 Modeling and Simulation Technologies Conference, DOI 10.2514/6.2018-3892.

4. Burgio, G., Mangeruca, L., Ferrari, A., Carloni, M., Valdivia- Guerrero, V., Albiol-Tendillo, L., Govindaraju, P., Gottschall, M., Oelsner, O., Reglitz, S., Stavesand, J., Himmler, A., Yapi, L., “Framework for Modelling and Simulation of Multi-Physics Aircraft Systems with Distributed Electronic Controllers” SAE 2017 AeroTech Congress and Exhibition, DOI 10.4271/2017-01-2115.

5. Weilkins, T., *Systems Engineering mit SysML/UML*, dpunkt Verlag, Heidelberg, 2014

6. Binder, B., Gottschall, M., Blochwitz, T., Kirsch, L., “PLM- and microservice-based integration approach of MBSE and System Simulation for federated continuous engineering platforms” NAFEMS World Congress 2019, Paper NWC-19-114

7. Gottschall, M., Binder, B., Reglitz, S., Saada, H., Cremona, F., Burgio, G., “Novel Framework Approach for Model-Based Process Integration from Requirements to Verification Demonstrated on a Complex, Cyber-Physical Aircraft System” SAE 2018 Aerospace Systems and Technology Conference, DOI 10.4271/2018-01-1947.