

Case Report

Engineering Development Simulator-Based Aircraft Development: A Pilot-in-the-Loop Engineering Approach

Aykut Oksuz* 

Advanced Training Systems, Turkish Aerospace (TA), Ankara, Turkey

Abstract

Engineering Development Simulator (EDS) and Pilot Training Simulator (PTS) Serve Fundamentally Different Purposes in an Aircraft Development Program. PTS Are Normally Developed After the Aircraft Design Has Reached a Mature Configuration and the Functional and Behavioral Characteristics of the Aircraft Systems Have Been Exactly Defined for Pilot Trainings Aligned with Aircraft Delivery in the Program. In Contrast, EDS Is Established While Aircraft Design Is Still Evolving Through Program Plan. It Represents a Selected Design Baseline in a Real-time, Pilot-In-The-Loop (PIL) Environment So That Emerging Aircraft, Aircraft Systems, Cockpit Environment and Human-machine Interface Characteristics Can Be Evaluated Before the Corresponding Configuration Is Fully Available for Flight. This Paper Presents a Development and Utilization Methodology for EDS Created for Combat Aircraft Development Program. The Paper Describes the Rationale for Establishing the Simulator, Its Principal Functional Components, the Relationship Between Aircraft Design Baselines and Simulation Baselines, the Integration of Mathematical and Behavioral Models, and the Iterative Process Through Which Pilot Feedback Is Transferred to Aircraft and Aircraft Systems Design Activities. A Quantitative Framework Is Proposed to Measure the Engineering Contribution of the EDS Through Requirement-oriented, PIL Testing. The Simulator's Engineering Workload Is Demonstrated Using Project Data Compromising 1,200 Evaluated Requirements, 6,920 Test Points, 232 Simulator Test Runs and 360 Closed Engineering Findings. These Measured Data Are Used to Characterize Simulator Test Productivity and Establish a Flight-Test-Equivalent (FTE) Metric Based on the Assumed Number of Comparable Test Points Per Representative Flight Opportunity. A Scenario-based Flight-Test Reduction (FTR) Analysis Is Then Performed by Varying Both the Test Point Per Flight Assumption and the Corresponding Aircraft Flight Test Baseline. The Results Demonstrated How the Estimated Flight Test Contribution of the EDS Changes Under Different Baseline and Productivity Assumptions. The Simulator's Engineering Approach Provides a Framework for Quantifying EDS Engineering Activity While Distinguishing Measured Simulator Results from Assumption-based Flight Test Equivalence. The EDS Therefore Provides a Repeatable PIL Environment in Which the Aircraft Development Issues Can Be Identified, Investigated and Resolved Before or in Parallel with Aircraft Flight Testing, Supporting Earlier Design Iteration and Reducing Dependence on the Aircraft for Initial Engineering Evaluation.

Keywords

Engineering Development Simulator, Pilot In The Loop Simulation, Aircraft Development, Flight Testing, Simulator Based Development, Cockpit Human Machine Interface, Test Point Efficiency, Flight Test Equivalent

*Correspondence: Aykut Oksuz (aykut.oksuz@tai.com.tr)

Received: 21 September 2026; Accepted: 4 October 2026; Published: 26 October 2026



Copyright: © The Author(s), 2026. Published by Science Publishing Group. This is an **Open Access** article, distributed under the terms of the Creative Commons Attribution 4.0 License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution and reproduction in any medium, provided the original work is properly cited.

1. Introduction

An important distinction exists between EDS and PTS where PTS is generally based on a mature aircraft configuration. Its primary purpose is to reproduce sufficiently stable aircraft and system behavior so that pilots can learn, practice, and be evaluated against defined procedures. The development process therefore depends strongly on established aircraft design data, finalized system behavior, and approved interfaces. On the other hand, aircraft development programs require continuous interaction between analytical design, system integration, human factors evaluation, ground testing, and flight testing. As the aircraft configuration evolves, decisions made in one discipline can influence several other disciplines. A change in a flight-control law or mission system operational flow, for example, may alter pilot workload, cockpit indications, system functional logic, or the behavior of connected systems. Consequently, EDS is an environment in which these interactions can be evaluated before the complete aircraft is available can provide substantial engineering value.

An EDS has a different lifecycle. It must become useful before the aircraft and its systems are fully mature. The simulator consequently cannot simply wait for complete design documentation. Instead, it must represent an agreed aircraft baseline, identify the maturity and limitations of each model, and evolve together with the aircraft design. This makes configuration management, model maturity tracking, rapid integration, and pilot feedback management central elements of the simulator methodology.

The present study addresses this resolution by treating the EDS as a structured engineering environment within the aircraft development lifecycle. Rather than evaluating the simulator only through individual handling-quality experiments or isolated design studies, the study quantifies its use through simulator test runs, test points, requirements evaluated, simulator operating time, and engineering findings closed. The demonstrated architecture uses a rehosted flight model to support aircraft-level development, flight-control-law development, system evaluation, and envelope-expansion activities. In addition, a planned rehosted operational flight program (OFP) architecture is proposed to extend the EDS toward mission-system integration, where radar, electro-optical/infrared (EO/IR), electronic support measures (ESM), identification friend or foe (IFF), and electronic counter-countermeasures (ECCM), etc. and other mission system functions can be evaluated against synthetic entities, including their employment and contribution to mission performance.

2. Literature Review

Building on recorded first flight, flight simulation is widely recognized as a valuable tool for supporting aircraft design decisions and the evaluation of new flight and control concepts before full-scale flight testing. Modern simulation environments extend this concept by integrating real-time aircraft

models, cockpit interfaces, hardware interfaces, and other elements of the target system while keeping the pilots in the loop. The literature therefore establishes a clear progression from early use of simulation for aircraft design and cockpit evaluation toward PIL flight control development, model-based integration, flight test preparation, and broader aircraft-system research. However, previous studies generally focus on a particular engineering application or simulator capability. Limited quantitative evidence is available that links sustained simulator activity to requirement coverage, test point workload, simulator utilization, and engineering finding closure within a single aircraft development program.

Simulators has long been used as an engineering tool in aircraft development, particularly for evaluating aircraft handling qualities, pilot-vehicle interaction, flight control concepts, cockpit configurations, and operational scenarios. An early example directly addressing this role was reported by Baldwin and Drewett [1], who described an Air Combat Simulator developed to support the aircraft development process. Their work used experienced pilots and aircraft designers in a PIL environment to evaluate proposed aircraft configurations, cockpit concepts, controls, displays, and realistic mission scenarios before major production design commitments. The simulator was also used to collect information on aircraft performance, handling qualities, controls, and weapon delivery for feedback into the aircraft design process. This work established an important early precedent for treating simulators as an active engineering environment rather than solely as a pilot-training device.

The role of PIL simulation in flight control development was subsequently examined more explicitly by Parrag, Knotts, and Weingarten [2]. They discussed the complementary use of ground-based and in-flight simulation for flight control system development and emphasized the need for PIL evaluation for highly augmented aircraft configurations. Their work positioned simulation as a means of identifying and resolving critical design issues and of evaluating the closed-loop pilot-aircraft system and flying-qualities before relying exclusively on flight testing.

The growth of model-based design has further increased the importance of simulation integration within the aircraft development lifecycle. Durak et al. [3] demonstrated the integration of air vehicle models into a real-time flight-simulation environment to support PIL testing within a model-based rotorcraft design process. Their work highlighted the need for rapid and repeatable model integration so that the simulator can keep pace with iterative aircraft development. This is particularly relevant to engineering development simulators, in which the aircraft model and the simulator configuration must evolve together.

Requirements-based simulation provides another important foundation for the present methodology. Van't Hoff et al. [4]

proposed a requirements-based approach for the use of simulation in rotorcraft certification and emphasized the need to relate simulation applications and fidelity to the intended verification objectives. This perspective is complementary to the present study rather than defining only whether a simulation can support a requirement, the present case study records the engineering workload associated with requirements actually evaluated in the simulator. The resulting requirement, test point, run, and time metrics provide an empirical description of simulator use within an aircraft development campaign.

More recent engineering applications have extended simulation beyond isolated control-law or handling-quality studies toward broader aircraft design and flight test preparation. Bradfield and West [5] described an engineering simulator developed for the Airlander 10 program to support aircraft design evaluation, aircraft behavior and performance modelling, pilot and flight test personnel preparation, operational strategy development, and flight test risk assessment. Their work illustrates how a dedicated engineering simulator can become an active part of the development and flight test planning process and can be used to investigate failure scenarios before aircraft testing.

A further evolution is represented by the Future Systems Simulator reported by Korek et al. [6]. The simulator was developed as a modular engineering environment for research and development across aircraft systems, human factors, future cockpit design, aircraft electrification, and alternative control approaches. Such architectures demonstrate the value of modular simulation environments in which different aircraft and system models can be integrated and evaluated as the design evolves.

Recent engineering applications have extended simulation toward digital-twin. Aydemir et al. [7] proposed an engineering simulation approach that integrates a digital twin of an iron-bird test rig with an engineering flight simulator, using PIL simulation to support virtual flight testing and certification-oriented evaluation. Their study demonstrated how data and models derived from a physical test environment can be incorporated into an engineering simulator and compared with corresponding flight test results. Wang et al. [8] presented a systems engineering-based flight simulation device framework in which the engineering simulator evolves alongside aircraft development. The study describes aircraft-level human-in-the-loop simulation for flight-quality assessment, cockpit design, control law development, systems requirement validation, integration and verification. This work highlights the role of engineering simulation as a continuously evolving development platform rather than solely as a training or standalone research tool. Zhang et al. [9] proposed a generalized digital-twin framework for fixed-wing aircraft that integrates physical and virtual aircraft representations with data and service layers to support flight accompaniment and real-time simulation. This work provides relevant architectural context for EDS environments in which aircraft models, simulation data, and operational interfaces must remain consistent

with the evolving aircraft configuration. Wang et al. [10] review digital virtual-flight simulation methods and their use in aircraft design and flying-qualities assessment. These studies provide relevant architectural context for evolving simulation environments, although they do not report the same combination of project-level workload measures used in the present study.

Other recent studies address simulator implementation, model validation, and repeatability. Portapas [11] reports the development of an eVTOL flight simulator and an assessment of its longitudinal handling characteristics. Moenck et al. [12] discuss implementation challenges and opportunities for digital twins in aircraft production and maintenance, while R ädel et al. [13] demonstrated the use of digital-twin-based data acquisition and validation for manufacturing simulation in an aircraft production context, illustrating the broader role of validated digital representations in aerospace engineering. Waitman et al. [14] demonstrated aircraft-in-the-loop validation of a fault tolerant flight control system by coupling a research aircraft operated by a pilot on the ground with an emulation computer that reproduced the aircraft flight dynamics. This study illustrates how integrated simulation and real aircraft hardware can be used to evaluate flight control behavior under controlled test conditions. Schiller et al. [15] developed and validated a toolbox for generating reproducible flight profiles for simulator studies, demonstrating the importance of repeatable and controlled simulation inputs for systematic evaluation. Taken together, this literature supports the technical context for EDS development but also highlights the value of reporting measured engineering workload separately from inferred flight test equivalence.

3. Materials and Methods

3.1. EDS Concept

Simulation provides a controlled and repeatable environment in which aircraft and aircraft systems development activities can be performed without to the constraints associated with real flight operations. This is particularly important for single seat aircraft, where PIL evaluations cannot always be performed with additional personnel and repeated as frequently as required. A simulator enables engineers and pilots to investigate selected aircraft behaviors, procedures, techniques and system functions repeatedly under controlled conditions while limiting exposure to aircraft, personnel and operational resources. The ability to reproduce the same scenario multiple times also supports systematic comparison of different configurations and facilitates the identification and resolution of development issues. Consequently, simulation can provide a practical and efficient means of building engineering knowledge and pilot proficiency while reducing the need to rely exclusively on aircraft availability during early development activities.

EDS can be implemented in different configurations depending on the intended engineering application. As depicted in Figure 1, the simulator configurations considered in this study include dome, desktop and virtual reality (VR) simulators. A dome simulator provides a large field of view immersive environment, typically using a physical cockpit and a surrounding visual display, and is suitable for evaluations requiring a high level of pilot immersion and spatial awareness. A VR simulator uses a head-mounted display to provide an immersive virtual cockpit and visual environment, offering a

flexible alternative when physical cockpit hardware or a large visual system is not required. These configurations allow the EDS to be tailored to the engineering question, available resources, and required level of fidelity. A desktop simulator provides a more compact and flexible setup in which the pilot interacts with the simulated aircraft through computer-based displays and controls, making it suitable for rapid engineering assessments and repeated testing.

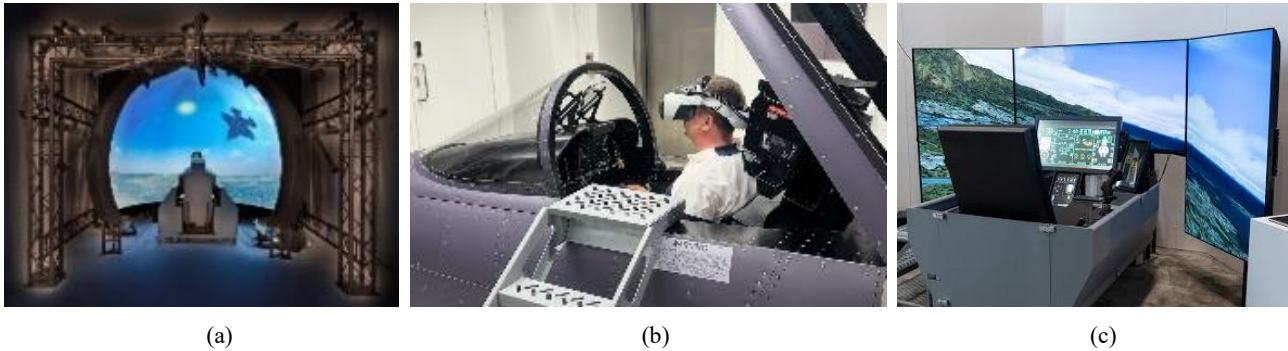


Figure 1. Representative EDS Configurations: (a) Dome Simulator, (b) VR Simulator, (c) Desktop Simulator.

The hardware components may vary depending on the selected physical simulation environment, such as the dome, VR, or desktop simulator. The relative approximate unit costs of these physical simulator environments are presented in Figure 2.

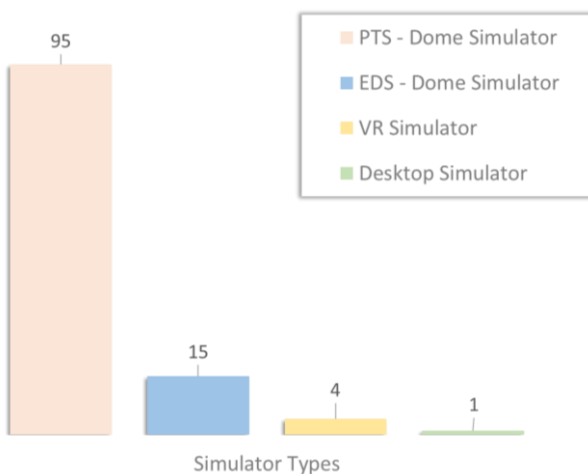


Figure 2. Approximate Unit Cost Comparison of Simulator Types.

3.2. Baseline Synchronization and Configuration Management

The EDS evolves together with the aircraft program and is

therefore managed according to the corresponding development baseline. Each simulator configuration identifies the maturity and known limitations of the models, software and interfaces included in the release. This allows the simulator to support activities such as flight control law development system performance evaluation, requirement-based testing and engineering assessment before or alongside aircraft flight testing. The simulation environment may incorporate aircraft and systems models, cockpit interfaces, environmental conditions and representative operational or tactical entities. Aircraft development is characterized by frequent changes in configuration. A simulator that is not synchronized with the current aircraft baseline may generate misleading feedback. The proposed methodology therefore uses a baseline-driven process. Each simulator release is associated with a defined aircraft configuration, system requirements set, model version, interface definition, and known limitation list. Configuration management is therefore not only a software-management activity it is a mechanism for maintaining traceability between aircraft design decisions, simulation assumptions, pilot observations, engineering changes, and subsequent verification activities.

3.3. Simulator Architecture

In this study EDS architecture based on dome simulator is organized into interconnected hardware and software elements that reproduce the aircraft, pilot interface, and simulated operational environment.

The main components of the hardware architecture from product tree view are illustrated in Figure 3. The hardware

configuration includes the cockpit mock-up and associated pilot interfaces, visual-system components, enabling hardware and software, and the computing infrastructure required to execute the aircraft and system models in real time. The dome simulator is a physical environment equipped with specially designed optical and projection surfaces, within which the synthetic environment is presented to the pilot. The system

provides a wide field of view (FoV), covering approximately $\pm 150^\circ$ in the horizontal direction and $+50^\circ/-105^\circ$ in the vertical direction. This extensive FoV, combined with the cockpit mock-up, enables the pilot to perform flight tasks within a visual and physical interaction environment that closely approximates the conditions of an actual flight.

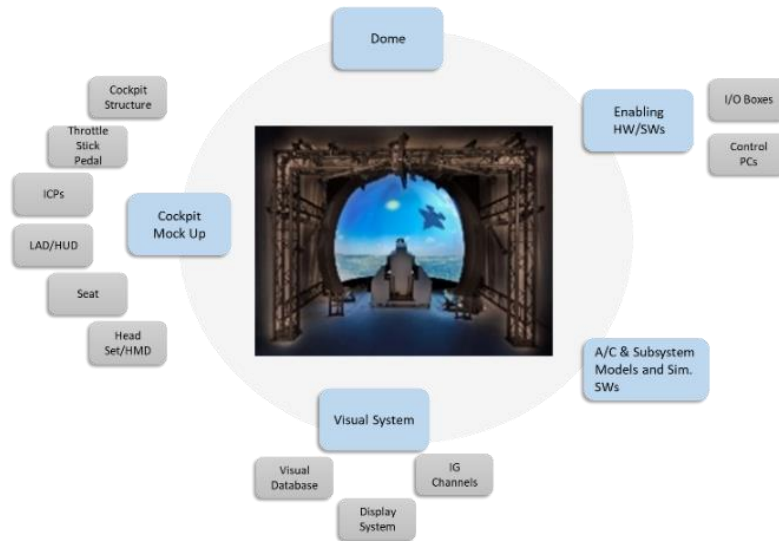


Figure 3. EDS Product Tree.

The visual system consists of reflective projectors that provide image projection onto the dome surface, equipment responsible for generating high-quality visual outputs such as Out-the-Window (OTW), Infrared (IR), and DayTV imagery, and the visual database, which contains the features required to represent the synthetic environment. These features include a terrain database representing the surrounding terrain, three-dimensional objects, thermal maps, and the simulation of environmental conditions such as day and night.

The cockpit mock-up represents the physical cockpit environment in which the pilot controls the aircraft. It consists of cockpit panels, flight control interfaces such as the stick, throttle, and pedals, a helmet and wide field-of-view display systems, and the pilot seat. These components provide the pilot with a physical control and interaction environment that closely resembles the cockpit of an actual aircraft.

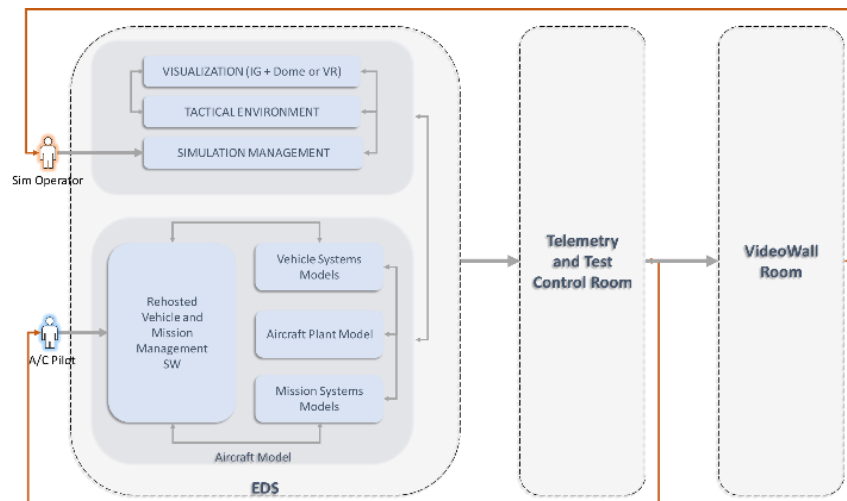


Figure 4. EDS Software Top Level Architecture and Test Environment.

At the software level, the architecture integrates the aircraft plant model, vehicle and mission systems models, visual and tactical environments, and simulation-management functions, as shown in Figure 4. In addition, the Telemetry and Control Room is presented as the environment in which the flight or mission is monitored and controlled based on pilot inputs. The Video Wall Room provides an operational environment in which all entities and interactions within the simulation environment can be monitored and controlled by the simulator operator and test/mission manager.

The Aircraft (A/C) models consist of three main categories: the plant model, vehicle systems, and avionics and mission systems. The aircraft plant model includes the aerodynamic and flight dynamics representation together with relevant flight control, propulsion, weight-and-balance, landing-gear, hydraulic, actuator, navigation, and other aircraft-system models. Vehicle systems include electrical power system models, life support and crew escape models, engine start system models, environmental control system models, and other aircraft systems. Avionics and mission systems include radar, electro-optical/infrared (EO/IR), electronic support measures (ESM), identification friend or foe (IFF), and electronic counter-countermeasures (ECCM), datalink, etc. and weapon models. Vehicle system and mission system models can be incorporated according to the objective of the simulation.

The tactical environment provides an operational framework for executing red and blue platforms, including their Radar Cross Section (RCS) and Infrared (IR) signatures, and for managing platform maneuvering behaviors using rule-based coding. The Simulation Management component enables the execution of the aircraft plant model within the tactical environment, manages all system models at the required dynamic simulation time resolution, synchronization and controls data flow across the simulation network.

A key element of the EDS software architecture is the use of rehosted aircraft software as vehicle systems and mission systems management. The rehosted vehicle systems software enables the simulator to execute the aircraft flight control and vehicle behavior within the same development framework used to evaluate the corresponding aircraft configuration. This approach allows changes in flight control laws, aircraft models, and system behavior to be assessed in a controlled PIL environment before their full verification on the aircraft. The same architecture also provides rehosted mission systems software to achieve a high degree of behavioral consistency between the simulator and the actual aircraft. This approach enables the simulation environment to reproduce aircraft system behavior and software level interactions with a high degree of fidelity.

These system models represent the functional and operational characteristics of the aircraft systems, including their performance, operating modes and state transitions, functional behavior, warnings and fault outputs within the operational envelope, and software data interfaces with higher and lower-

level software components.

3.4. PIL Test and Quantitative Evaluation Methodology

EDS engineering process combines PIL evaluation with requirement-oriented test execution. Each test activity begins with the definition of the applicable aircraft baseline and engineering objective, followed by scenario preparation and identification of the relevant test points. During execution, the pilot interacts with the simulated aircraft using representative controls and cockpit interfaces, while objective data and pilot observations are collected. The resulting observations are reviewed by the engineering team, and identified issues are converted into findings, design actions, or model updates. The updated configuration can then be reintroduced into the simulator for subsequent evaluation, establishing a closed engineering feedback loop.

To quantify simulator activity, the study uses directly measurable parameters including the number of simulator test runs, test points, simulator operating hours, requirements evaluated, and engineering findings closed. A major challenge in quantifying simulator benefit is the absence of a complete counterfactual dataset: in many development programs, the exact number of flight tests that would have been conducted without the simulator is not known. Reporting an assumed FTR without a documented baseline can therefore overstate the result.

The methodology proposed here separates directly measured simulator quantities from inferred flight test-equivalent quantities. Directly measured quantities are simulator test count, total test point count, total simulator execution time, test point execution time, scenario coverage, and the number of design issues or changes identified through simulator use.

Let define N_{sim} as the number of simulator test runs and TP_{total} as the total number of test points. The average number of test points generated per simulator test is calculated as:

$$TP_{avg} [-] = TP_{total} [-] / N_{sim} [-] \quad (1)$$

The test point generation rate is calculated as:

$$TP_{rate} [1/hrs] = TP_{total} [-] / t_{total} [hrs] \quad (2)$$

where t_{total} is the total simulator execution time in hours. These metrics describe measured EDS workload without assuming that a simulator test point is directly equivalent to an aircraft flight test. For the flight test-equivalent analysis, the simulator test points are mapped to an assumed number of comparable test points that can be addressed during one representative aircraft flight opportunity.

For a test campaign i , let $N_{sim,i}$ be the number of simulator test points and $TP_{flight,i}$ be the independently established average number of equivalent test points that can be achieved during one representative flight opportunity.

The number of FTE quantity is calculated as:

$$N_{FTE} [-] = \Sigma (N_{sim,i} [-] / TP_{flight,i} [-]) \quad (3)$$

Because a complete historical flight test baseline is not available for the present case, the study uses a scenario-based sensitivity analysis. The current dataset contains 6,920 measured simulator test points. A common TP_{flight} assumption is first applied to the total test point set, using illustrative values of 10, 15, 20, 25, and 30 comparable test points per flight opportunity. These values are analysis assumptions and are not measured flight test productivity values. Category-specific $TP_{flight,i}$ values can be introduced when planned or conducted flight test data become available.

The corresponding FTR is calculated against an explicitly assumed aircraft flight test baseline, N_{FTE} :

$$FTR [\%] = [1 - (N_{FTE} [-] / N_{flight} [-])] \times 100 \quad (4)$$

For illustration, three baseline scenarios of 400, 600, and 800 flight test opportunities to represent low, medium and

high volumes are evaluated. The resulting FTR values are reported as a sensitivity analysis rather than as an observed program result. A negative FTR value indicates that the assumed simulator-equivalent workload exceeds the assumed baseline and therefore signals that the selected assumptions are not mutually representative of the same test scope. The estimated FTR should not be interpreted as an observed reduction in the actual flight test program. It represents a scenario-based estimate obtained by comparing the FTE workload generated by the EDS with an assumed corresponding flight test baseline. If verified historical flight test records become available, the assumed baseline and TP_{flight} values should be replaced with documented values and the analysis repeated.

The contribution of the EDS is quantitatively assessed using a set of complementary metrics shown in Table 1 that capture test generation efficiency, test point generation rate, and the representation of flight test opportunities. These metrics provide a consistent basis for evaluating the extent to which EDS-based simulation can support and reduce the need for physical flight testing.

Table 1. Proposed quantitative metrics for evaluating EDS contribution.

Metric	Definition	Interpretation
Test generation efficiency, TP_{avg}	Equation 1	Average test points per simulator run
Test point generation rate, TP_{rate}	Equation 2	Test points per simulator hour
Flight-Test-Equivalent, FTE	Equation 3	Estimated flight opportunities represented
Flight-Test Reduction, FTR	Equation 4	Applicable with an actual/estimated flight baseline

4. Results and Discussions

The measured EDS activity provides the quantitative basis for evaluating its engineering contribution. During the evaluated development period, the simulator supported 360 hours of PIL operation, 6,920 test points, 232 simulator runs, 1,200 evaluated requirements, and 360 closed engineering findings.

These results are reported at the test point level because a single simulator run can address multiple engineering questions. The measured workload is therefore characterized using test points, simulator hours, requirements, runs, and findings rather than simulator-session count alone.

The principal measured and calculated indicators are summarized in Table 2. The 12,000-requirement value is used only for the extended planning projection and is not used to report measured requirement coverage.

Table 2. Project-specific simulator and flight test efficiency results to be populated from verified records.

Metric	Value	Interpretation
Simulator test runs	232	Observed Data
Simulator flight hours	360 h	Observed Data
Requirements tested in simulator	1200	Observed Data
Test points executed	6920	Observed Data

Metric	Value	Interpretation
Findings closed	360	Observed Data
Provisional total aircraft validation requirements	12000	Planning projection assumption
Average test points per simulator run	29.83	Calculated Data
Test points per simulator hour	19.22	Calculated Data
Test points per tested requirement	5.77	Calculated Data
Simulator hours per tested requirement	0.30	Calculated Data
Closed findings per tested requirement	0.30	Calculated Data (That descriptive indicator does not mean a defect rate.)
FTE	230.7-692.0*	Analytical Estimation (Scenario range for TP_{flight} / flight opportunity = 10 - 30)
Estimated FTR	Scenario dependent	Analytical Estimation (That depends on assumed aircraft flight opportunity baseline and TP_{flight})

4.1. EDS Utilization and Engineering Contribution

To date, the EDS has accumulated a total of 360 hours of PIL operation, during which 6,920 test points have been evaluated. This corresponds to an average test point generation rate of 19.22 test points per simulator hour. The recorded ac-

tivity demonstrates that the simulator is being used as a multidisciplinary aircraft development and integration environment rather than solely for flight performance assessment.

The distribution of the evaluated test points is shown in Figure 5. Performance and Flying Qualities account for 15%, Flight Performance for 11%, Sub-system Performance for 21%, Flight Control Law Development for 36%, and other activities for 17%. Flight Control Law Development is therefore the largest individual category, accounting for more than one-third of the total test point workload.

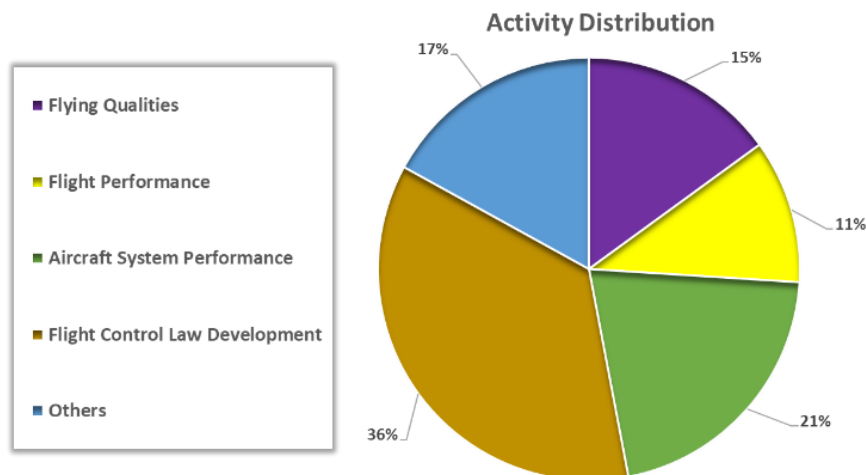


Figure 5. Distribution of Simulator Test Points by Engineering Activity.

The requirement and finding data further show that the EDS was used as an engineering feedback environment rather than only as a scenario-execution platform. The 1,200 evaluated requirements and 360 closed findings provide outcome-oriented indicators of simulator-supported development activity in

Phase 1.

For planning purposes in Phase 2, 12,000 aircraft validation requirements are assumed as a program-level reference case through evolution of the EDS from aircraft envelope develop-

ment toward mission-system integration and operational evaluation. This is an explicit planning assumption as depicted in Figure 6, not a verified program total. Using the measured 5.77 test points per evaluated requirement gives a proportional projection of approximately 69,200 test points for a 12,000-

requirement scope. At the measured productivity levels, this corresponds to approximately 2,320 simulator runs and 3,600 simulator hours. These projected values are planning estimates and are not treated as measured results.

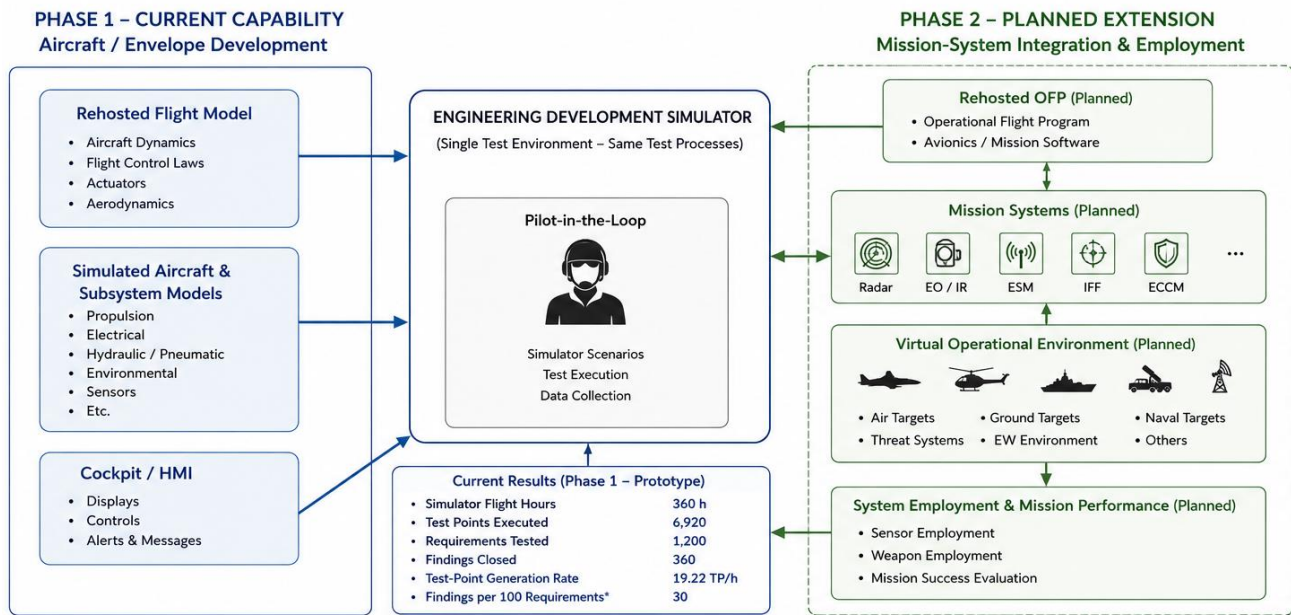


Figure 6. Evolution of the EDS from aircraft envelope development toward mission-system integration and operational evaluation.

4.2. Scenario-Based FTE and FTR Analysis

A scenario-based sensitivity analysis was performed using the 6,920 measured test points. Because the corresponding aircraft flight test baseline is not yet documented for the same scope, both TP_{flight} and the baseline flight test count are treated

as explicit assumptions.

For TP_{flight} values of 10, 15, 20, 25, and 30 test points per flight opportunity, the resulting FTE values are 692.0, 461.3, 346.0, 276.8, and 230.7 flight opportunities, respectively listed in Table 3. FTE decreases as more comparable test points are assumed to be combined within one flight opportunity.

Table 3. Scenario-based FTE and FTR sensitivity for the measured 6,920 simulator test points.

TP_{flight}	FTE	FTR @400 baseline	FTR @600 baseline	FTR @800 baseline	Interpretation
10	692.0	-73.0%	-15.3%	13.5%	Low assumed flight productivity; FTE exceeds 400 and 600 baselines
15	461.3	-15.3%	23.1%	42.3%	Intermediate sensitivity case
20	346.0	13.5%	42.3%	56.8%	Intermediate sensitivity case
25	276.8	30.8%	53.9%	65.4%	Higher assumed flight productivity
30	230.7	42.3%	61.6%	71.2%	High assumed flight productivity; lower FTE

The FTR sensitivity is strongly dependent on these assumptions. For a 600-flight baseline, the calculated FTR ranges from -15.3% at 10 TP_{flight} to 61.6% at 30 TP_{flight} . The negative value does not represent an increase in the actual flight test program; it occurs because the calculated FTE exceeds the assumed baseline. The sensitivity analysis therefore demonstrates the importance of documenting both the flight test

scope and the test-point productivity used in the mapping.

Figure 7 illustrates this sensitivity for the 400-, 600-, and 800-flight baselines. The curves are scenario results, not predictions of the actual flight test program. When documented flight test planning or historical data become available, the same framework can be recalculated using category-specific $TP_{flight,i}$ values and the corresponding baseline.

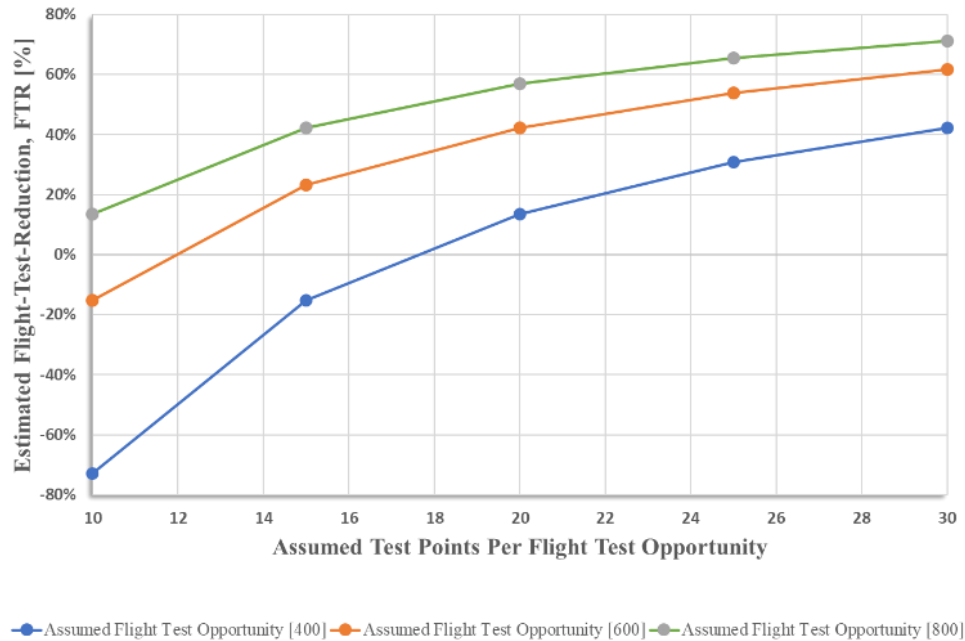


Figure 7. Sensitivity of estimated FTR to assumed test point productivity and flight test baseline.

The sensitivity analysis shows that flight-test-equivalent and flight-test reduction values depend strongly on the assumed mapping between simulator test points and aircraft flight opportunities. These results should therefore be interpreted as a transparent sensitivity analysis rather than as an observed program-level reduction.

5. Conclusions

This study presented an EDS methodology that integrates evolving aircraft baselines, real-time models, cockpit interfaces, configuration management, and PIL evaluation within the aircraft development lifecycle.

The application demonstrates a measurable engineering workload: 232 simulator runs and 360 simulator hours supported 6,920 test points, 1,200 evaluated requirements, and 360 closed findings. These results show that the EDS can provide a repeatable engineering environment for early identification, investigation, and closure of aircraft development issues. The proposed FTE/FTR framework translates the measured simulator workload into representative flight test opportunities under explicit TP_{flight} and flight test baseline assumptions.

The planned rehosted OFP extension provides a path toward mission-system integration and evaluation against synthetic entities, extending the same PIL and configuration-management approach beyond aircraft-level development.

Overall, the EDS should be viewed as complementary to flight testing: it moves suitable discovery and integration activities earlier in the development cycle, while aircraft flight testing remains necessary for aircraft-specific phenomena and final verification. When documented flight test records become available, the assumed TP_{flight} values and flight test baseline can be replaced with program data and the FTE/FTR analysis repeated. This provides a direct path from the present scenario-based methodology to a program-specific assessment.

Abbreviations

A/C	Aircraft
EDS	Engineering Development Simulator
eVTOL	Electric Vertical Take-Off and Landing
FoV	Field of View
FTE	Flight-Test-Equivalent
FTR	Flight-Test Reduction

HMD	Head Mounted Display
HMI	Human-Machine Interface
HUD	Head-Up Display
ICP	Integrated Control Panel
IG	Image Generator
LAD	Large Area Display
N _{sim}	Number of Simulator Test Runs
N _{total}	Number of Test Points
OFP	Operational Flight Program
PIL	Pilot-In-The-Loop
PTS	Pilot Training Simulator
TP _{flight}	Test Points Per Flight Opportunity
TP _{avg}	Average Test Points Per Simulator Test
TP _{rate}	Test Point Generation Rate
TP	Test Point
VR	Virtual Reality

Author Contributions

Aykut Oksuz: Conceptualization, Formal Analysis, Methodology, Writing – original draft

Data Availability Statement

The data is available from the corresponding author upon reasonable request.

Conflicts of Interest

The author declares no conflicts of interest.

References

- [1] Baldwin, D. M., and Drewett, J. D. "The Air Combat Simulator and Its Role in the Aircraft Development Process", AIAA Paper 86-2682, Aircraft Systems, Design & Technology Meeting, Dayton, Ohio, USA, October 20-22, 1986. <https://doi.org/10.2514/6.1986-2682>
- [2] Parrag, M., Knotts, L., and Weingarten, N. "Use of Ground-Based and In-Flight Simulation for Flight Control System Development", AIAA Paper 90-1286, 5th Biannual Flight Test Conference, Ontario, California, USA, 1990, pp. 215-224. <https://doi.org/10.2514/6.1990-1286>
- [3] Durak, U., Gerlach, T., Ozturk, A., Kargin, V., Aydemir, H., and Zengin, U. "Flight Simulator Model Integration for Supporting Pilot-in-the-Loop Testing in Model-Based Rotorcraft Design", AIAA Modeling and Simulation Technologies Conference, 2016. <https://doi.org/10.2514/6.2016-4004>
- [4] van't Hoff, J., et al. "Preliminary Guidelines for a Requirements-Based Approach to Certification by Simulation for Rotorcraft", 48th European Rotorcraft Forum Conference Proceedings / Technical publication, 2022.
- [5] Bradfield, W., and West, R. N. "Use of an Engineering Simulator to Inform Design and Flight Test Planning of a Novel Hybrid Aircraft", Society of Flight Test Engineers European Chapter, 2022.
- [6] Korek, W. T., Beecroft, P., Lone, M., et al. "Simulation Framework and Development of the Future Systems Simulator", The Aeronautical Journal, 128(1330), pp. 2754-2780, 2024. <https://doi.org/10.1017/aer.2024.91>
- [7] Aydemir, H., Zengin, U., Pinon Fischer, O. J., and Durak, U. "From Iron Birds to Digital Twins with Engineering Simulators: Toward Virtual Certification." Journal of Aerospace Computing, Information, and Communication, 22(11), 2025. <https://doi.org/10.2514/1.1011569>
- [8] Wang, H., Zhang, C., and Chen, Y. "System Engineering-Based Flight Simulation Device Design Technology and Application Research." In: Pham, D. T., Lei, Y., and Lou, Y. (eds.), Mechanical Design and Simulation: Exploring Innovations for the Future, MDS 2024, Lecture Notes in Mechanical engineering, pp. 251-267. Springer, Singapore, 2025. https://doi.org/10.1007/978-981-97-7887-4_21
- [9] Zhang, W., Deng, J., Liu, X., and Lu, J. "FlightTwin: A Generalized Digital Twin Accompanying Flight Framework for Fixed-Wing Aircraft." IEEE Access, 12, 125194-125210, 2024. <https://doi.org/10.1109/ACCESS.2024.3454197>
- [10] Wang, L., Niu, Y., Liu, H., Wang, J., Wang, X., and Yue, T. "Aircraft Digital Virtual Flight Simulation Method and Its Application." Acta Aeronautica et Astronautica Sinica, 46(5), 531543, 2025. <https://doi.org/10.7527/S1000-6893.2024.31543>
- [11] Portapas, V. "Developing Alsim AL250 Based eVTOL Flight Simulator." Aviation, 29(1), 39-47, 2025. <https://doi.org/10.3846/aviation.2025.23167>
- [12] Moenck, K., Rath, J.-E., Koch, J., Wendt, A., Kalscheuer, F., Schüppstuhl, T., et al. "Digital Twins in Aircraft Production and MRO: Challenges and Opportunities." CEAS Aeronautical Journal, 15, 1051-1067, 2024. <https://doi.org/10.1007/s13272-024-00740-y>
- [13] Rädcl, M., Denker, B., van de Kamp, B., Liebers, N., et al. "Towards the Validation of Manufacturing Simulations by Means of Digital Twins: Conception, Implementation and Data Acquisition for a Composite Aircraft Moveable Manufacturing Process." CEAS Aeronautical Journal, 16, 399-416, 2025. <https://doi.org/10.1007/s13272-024-00800-3>
- [14] Waitman, S., Marcos, A., and Sato, M. "Design and Aircraft-in-the-Loop Validation of Structured-H_∞ Self-Scheduled Fault-Tolerant Controllers." International Journal of Robust and Nonlinear Control, 2024. <https://doi.org/10.1002/rnc.7088>
- [15] Schiller, A. "Implementation of a Toolbox for the Generation of Flight Profiles for Use in Simulator Studies." CEAS Aeronautical Journal, 15, 1175-1188, 2024. <https://doi.org/10.1007/s13272-024-00755-5>

Biography



Aykut Oksuz is a Mechanical Engineer with 18 years of experience in the defense industry. He currently works at Turkish Aerospace (TA), where he contributes to aircraft development, requirements engineering, modeling and simulation (M&S), and simulation-based development activities. Prior to joining TA, he worked at ROKETSAN, where he was involved in flight mechanics and missile development activities. He holds an MSc degree from Middle East Technical University (METU).