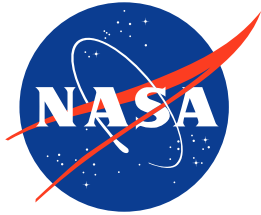


NASA/TM-20250007132



Dynamic Path Planning Automation Architecture for Advanced Air Mobility

Application of a New MBSE Approach to System Architecture Development

Vivek Sharma

Langley Research Center, Hampton, Virginia

August 2025

NASA STI Program...in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA scientific and technical information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI Program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI Program provides access to the NASA Aeronautics and Space Database and its public interface, the NASA Technical Report Server, thus providing one of the largest collection of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- **TECHNICAL PUBLICATION.** Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers, but having less stringent limitations on manuscript length and extent of graphic presentations.
- **TECHNICAL MEMORANDUM.** Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- **CONTRACTOR REPORT.** Scientific and technical findings by NASA-sponsored contractors and grantees.

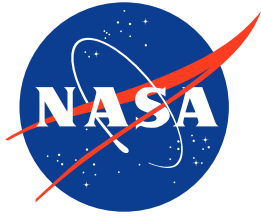
- **CONFERENCE PUBLICATION.** Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.
- **SPECIAL PUBLICATION.** Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.
- **TECHNICAL TRANSLATION.** English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include organizing and publishing research results, distributing specialized research announcements and feeds, providing information desk and personal search support, and enabling data exchange services.

For more information about the NASA STI Program, see the following:

- Access the NASA STI program home page at <http://www.sti.nasa.gov>
- E-mail your question to help@sti.nasa.gov
- Phone the NASA STI Information Desk at 757-864-9658
- Write to:
NASA STI Information Desk
Mail Stop 148
NASA Langley Research Center
Hampton, VA 23681-2199

NASA/TM-20250007132



Dynamic Path Planning Automation Architecture for Advanced Air Mobility

Application of a New MBSE Approach to System Architecture Development

Vivek Sharma

Langley Research Center, Hampton, Virginia

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23681-2199

August 2025

Acknowledgments

The author is grateful to Mr. David J. Wing, Distinguished Research Associate, NASA Langley Research Center, for providing his technical expertise and insightful feedback on the multiple reviews of the architecture.

The author also thanks Ms. Amanda N. Banks, Technical Editor for NASA Langley Research Directorate, for her meticulous review and valuable suggestions that significantly improved the clarity and accuracy of this document.

The use of trademarks or names of manufacturers in this report is for accurate reporting and does not constitute an official endorsement, either expressed or implied, of such products or manufacturers by the National Aeronautics and Space Administration.

Available from:

NASA STI Program / Mail Stop 148
NASA Langley Research Center
Hampton, VA 23681-2199
Fax: 757-864-6500

Abstract

Advanced Air Mobility (AAM) aims to develop an air transportation system for novel air vehicles between local, regional, and intraregional places. Safety and efficiency of increasingly complex AAM operations are expected to require extensive use of automation, ranging from controlling the revolutionary new aircraft to managing the flight paths dynamically in high tempo airspace and aerodrome operations. This document describes the architecture of a dynamic path planning (DPP) automation system in the context of increasingly autonomous operations. The purpose of the DPP automation system is to build, coordinate, and maintain a safe, efficient, and operationally acceptable flight path throughout the flight. To meet the challenges of the evolving AAM operations, the DPP automation system must not only adapt autonomously to the dynamic operating environment, but also respond to potential mission changes, aircraft commands, and user instructions. A comprehensive, rigorous, and flexible architecture is paramount to meeting these complex requirements in the design and development of future DPP automation systems. The DPP automation system architecture will offer guidance to the industry, certification authorities, and service providers in establishing best design practices, community standards, and means of compliance recommendations for industry-wide adoption of dynamic path planning automation systems.

A new model-based systems engineering approach was developed to architect a DPP automation system. The fundamental steps in this approach are the development of *conceptual*, *functional*, *logical*, and *physical* architectures of the system to a level of depth that is suitable for designing the system. The *conceptual* architecture comprises: identifying the primary stakeholders and their concerns; analyzing the stakeholder concerns to derive system goals; defining the system boundary and interfaces in the operational domain; refining the system goals as a set of use cases and measures of performance; and extracting the top-level system requirements. These behavioral and structural elements are then decomposed in the *functional* architecture in order to derive detailed requirements of the DPP automation system. The system requirements serve as the basis for developing an implementation-agnostic *logical* architecture of the DPP automation system and its parts, namely, the flight path initialization, flight path creation, flight path monitoring, flight path evaluation, flight path revision, flight path coordination, input data processing, output data processing, flight contingency management, and supporting subsystems. The key features of the new methodical approach are its simplicity, rigor, and extensibility to disparate automation systems in aviation, space, and science domains.

Contents

1	Introduction	1
1.1	Purpose	1
1.2	Audience	2
1.3	Scope	2
1.4	Organization	3
2	From Concept to Architecture	3
2.1	Concept Overview	3
2.2	Advanced Air Mobility Domain	4
2.3	MBSE Methodology	5
2.4	System Architecture Elements	7
3	Conceptual System Architecture	10
3.1	Synopsis	10
3.2	Stakeholders	12
3.3	System Goals	17
3.4	System Context	21
3.5	System Usage	25
3.6	Measures of Effectiveness	55
3.7	Top-Level Requirements	55
4	Functional System Architecture	61
4.1	Synopsis	61
4.2	Behavioral Decomposition	63
4.3	Structural Decomposition	72
4.4	Functional Interfaces	75
4.5	Functional Requirements Elaboration	79
5	System Requirements Specification	82
5.1	Synopsis	82
5.2	Functional Requirements	83
5.3	Usability Requirements	124
5.4	Design Constraints	129
5.5	Physical Requirements	131
5.6	Interface Requirements	131

6	Logical System Architecture	145
6.1	Synopsis	145
6.2	Domain Elements	147
6.3	System Behavior	150
6.4	System Structure	150
6.5	Measures of Performance	155
6.6	Subsystem Requirements	155
7	Logical Subsystems Architecture	159
7.1	Synopsis	159
7.2	Flight Path Initializer	160
7.3	Flight Path Creator	162
7.4	Flight Path Monitor	164
7.5	Flight Path Evaluator	166
7.6	Flight Path Revisor	169
7.7	Flight Path Coordinator	172
7.8	Input Data Processor	175
7.9	Output Data Processor	178
7.10	Data Repository Manager	180
7.11	Flight Contingency Manager	182
7.12	System Alerts Manager	183
7.13	System Health Monitor	184
7.14	System Operation Manager	184
8	Future Work	185
9	Summary	187
	References	188
A	Abbreviations	189
B	Glossary	190

List of Figures

1	Methodology Overview	8
2	Model Structure	9
3	Stakeholder Concerns	16
4	Functional Goals	19
5	Nonfunctional Goals	20
6	Operational Domain	21
7	System Context: Preflight	24
8	System Context: During flight	25
9	System Usage: Preflight	26
10	System Usage: During Flight	27
11	Use Case: Build Baseline Flight Path	28
12	Use Case: Coordinate Baseline Flight Path	32
13	Use Case: Synchronize Baseline Flight Path	34
14	Use Case: Adapt Flight Path to Mission Change	36
15	Use Case: Adapt Flight Path to Shared Operating Environment	38
16	Use Case: Adapt Flight Path to Sensed Information	40
17	Use Case: Adapt Flight Path to Pilot Observed Event	42
18	Use Case: Assess Proposed Flight Path Change	44
19	Use Case: Mitigate Flight Path Contingency	46
20	Use Case: Update Flight Path	49
21	Use Case: Coordinate Candidate Flight Path	54
22	Build Baseline Flight Path	63
23	Coordinate Baseline Flight Path	64
24	Synchronize Baseline Flight Path	64
25	Adapt Flight Path to Mission Change	65
26	Adapt Flight Path to Shared Operating Environment	66
27	Adapt Flight Path to Sensed Information	67
28	Adapt Flight Path to Pilot Observed Event	68
29	Assess Proposed Flight Path Change	69
30	Update Flight Path	70
31	Coordinate Candidate Flight Path	71
32	Mitigate Flight Path Contingency	72
33	Structural Decomposition	75
34	External Interfaces	76

35	Functional Interfaces	77
36	Interfunctional Communication	78
37	Domain Structure	147
38	Domain Behavior	148
39	Onboard Pilot Actions	149
40	Aircraft Actions	149
41	Logical System Behavior	150
42	Logical System Parts	151
43	External Logical Interfaces	151
44	Internal Logical Interfaces	152
45	Logical System Structure	154
46	Subsystem Requirements Derivation-1	156
47	Subsystem Requirements Derivation-2	157
48	Subsystem Requirements Derivation-3	158
49	Flight Path Initializer Requirements	160
50	Flight Path Initializer Behavior	161
51	Flight Path Initializer Structure	161
52	Flight Path Creator Requirements	162
53	Flight Path Creator Behavior	163
54	Flight Path Creator Structure	163
55	Flight Path Monitor Requirements	164
56	Flight Path Monitor Behavior	165
57	Flight Path Monitor Structure	166
58	Flight Path Evaluator Requirements	167
59	Flight Path Evaluator Behavior	168
60	Flight Path Evaluator Structure	169
61	Flight Path Revisor Requirements	170
62	Flight Path Revisor Behavior	171
63	Flight Path Revisor Structure	172
64	Flight Path Coordinator Requirements	173
65	Flight Path Coordinator Behavior	174
66	Flight Path Coordinator Structure	175
67	Input Data Processor Requirements	176
68	Input Data Processor Behavior	177
69	Input Data Processor Structure	178
70	Output Data Processor Requirements	179

71	Output Data Processor Structure	180
72	Data Repository Manager Requirements	181
73	Data Repository Manager Structure	182
74	Flight Contingency Manager Requirements	183
75	System Alerts Manager Requirements	183
76	System Operation Manager Requirements	184

List of Tables

1	Top-Level Functional Requirements	56
2	Top-Level Usability Requirements	57
3	Top-Level Physical Requirements	57
4	Top-Level Design Constraints	58
5	Top-Level Interface Requirements	58

1 Introduction

Advanced Air Mobility (AAM) is an initiative of the National Aeronautics and Space Administration (NASA), the Federal Aviation Administration (FAA), and the aviation industry to develop an air transportation system that moves people and cargo between local, regional, and intraregional places previously not served or underserved by aviation, using revolutionary new aircraft [1–3]. Realizing the NASA vision for AAM significantly relies on automation and information services to achieve safe, efficient, and scalable operations. NASA is investigating a wide array of capabilities related to aircraft, airspace, infrastructure, and fleet operations, that are needed to make this vision a reality. One such capability is in-flight adaptation of aircraft flight paths to ensure safety and operational acceptability in a dynamically evolving operating environment.

NASA has developed a concept of dynamic path planning (DPP) that enables effective planning and management of the intended flight path in a dynamic AAM operating environment [10]. The concept identifies the need for an automation system that responds autonomously to the dynamic operating environment to ensure that a safe, efficient, and operationally acceptable flight path is available to the users and the flight path execution system throughout the flight. To meet the challenges of the evolving AAM operations, the DPP automation system must not only adapt autonomously to the dynamic operating environment, but also respond to potential mission changes, aircraft commands, and user instructions. This document aims to leverage the DPP concept to architect and design a reference DPP automation system.

A comprehensive, rigorous, and flexible architecture is paramount to meeting the complex requirements in the design and development of future DPP automation systems. A new model-based systems engineering (MBSE) approach was developed to architect the system. The key steps in this approach are the development of (a) conceptual, (b) functional, (c) logical, and (d) physical architectures of the DPP automation system to a level of depth that is suitable for designing the system. The system architecture will offer guidance to the industry, certification authorities, and service providers in establishing best design practices, community standards, and means of compliance recommendations for industry-wide adoption of dynamic path planning automation systems. A rigorously developed DPP system architecture is a key to supporting the verification and validation of AAM concepts related to automated flight path management. One such concept is that of defining the performance requirements for a new operating mode that will rely on automated conflict management [12].

1.1 Purpose

The primary purpose of this document is to describe the system architecture of a future DPP automation system in an AAM operational domain. The set of concept, architecture, and design documents will further the AAM operators’ understanding of the automation capabilities that will be required to operate safely, cooperatively, and efficiently in the increasingly complex AAM operating environment.

A secondary purpose of this document is to demonstrate the application of a new, methodical, rigorous, versatile, and intuitive MBSE approach to architecting an automation system. The complexity and the research nature of dynamic flight path planning are well

suited to provide a rigorous validation of the MBSE approach. The approach is applicable to a broad range of automation systems in the aeronautics and space domains.

1.2 Intended Audience

This document is intended to benefit (a) the DPP automation system designers, system implementers, standards development organizations, regulators, fleet operators, flight planners, data service providers, and researchers; and (b) the systems engineers and researchers responsible for designing complex systems which must account for the fluid stakeholder needs of evolving concepts. The audience may derive value from this document in the following manner:

1. The DPP automation system architecture lays the foundation of system design and implementation.
2. The DPP automation system architecture represents a crucial step towards establishing community standards for the verification and validation of various DPP automation system designs developed by industry.
3. The DPP automation system architecture document facilitates engagement among DPP stakeholders, including NASA, the FAA, AAM fleet operators, service providers, and automation system designers.
4. The DPP automation system architecture document provides NASA researchers with a frame of reference in which to develop design guidelines, means of compliance recommendations, and operational requirements for a reference system.
5. The proposed methodology, by virtue of its simplicity, rigor, and extensibility, offers a robust way to design complex systems in the aviation, space, and science domains.

1.3 Scope

This document presents the system architecture of a DPP automation system, based on the NASA DPP automation concept for AAM operations [10], developed using a new MBSE methodology. A system architecture, according to this methodology, comprises *conceptual*, *functional*, *logical*, and *physical* architecture elements. This document describes the conceptual, functional, and logical architecture elements of a dynamic flight path planning automation system in the context of AAM operational domain. While the system architecture in this document is presented with respect to an AAM operational domain, the approach is extensible to other domains—traditional airline operations and high-altitude platform systems, for instance—that rely on automation of planning and replanning of a flight’s path in response to the changing operating environmental conditions, mission objectives, and user needs. This document provides a glimpse into the application of a new, versatile, model-based systems engineering methodology employed to architect the DPP automation system from first principles.

This version of the document limits the scope of the logical system architecture to the primary DPP automation subsystems; architecting the remaining subsystems is deferred to a future enhancement. Additionally, a physical architecture of the DPP automation

system is out of scope of this document, because its details belong more appropriately in a system design document than in a system architecture description.

1.4 Document Organization

This document is organized as follows:

- §1 provides the purpose, intended audience, and scope of the document.
 - §2 provides a brief overview of the methodology used to develop the architecture.
 - §3 presents an overview of the DPP automation system *conceptual* architecture.
 - §4 presents an overview of the DPP automation system *functional* architecture.
 - §5 specifies a complete set of the DPP automation system requirements.
 - §6 presents an overview of the DPP automation system *logical* architecture.
 - §7 presents an overview of the logical architecture of the system parts.
 - §8 suggests next steps to refine and complete the system architecture.
 - §9 summarizes the DPP automation system architecture presented in this document.
- The document ends with lists of references, abbreviations, and glossary.



Each architecture description section and requirements specification section includes a synopsis. The synopses summarize the sections and contain links to the illustrations in the respective sections, providing the reader a compendious understanding of the DPP automation system architecture.

2 From Operational Concept to System Architecture

The origin of the system architecture is a clearly defined concept of operations of the system in a specified domain. This section provides overviews of: the DPP concept (§2.1); the AAM operational domain (§2.2); the process used to derive the DPP automation system architecture from the concept (§2.3); and system architecture elements (§2.4).

2.1 Dynamic Path Planning Automation Concept

NASA has developed a concept of DPP automation that enables effective planning and management of the intended flight path in a dynamic operating environment [10]. The concept identifies the need for a system that responds autonomously to the dynamic operating environment to ensure that a safe and operationally acceptable flight path is available to the users and the flight path execution system throughout the flight. The

roles of the DPP automation system are fivefold: it creates a flight path with desirable qualities of being feasible, deconflicted, harmonized, flexible, and optimal; it monitors the progress of the flight in a dynamic operating environment; it supports the user in evaluating continued acceptability of the flight path in changing conditions; it revises the flight path as needed to maintain the desired flight path qualities; and it coordinates the flight path with airspace and aerodrome users and service providers. Key users of the DPP automation system include the flight planners, pilots, autonomous aircraft, and flight path coordinators. The concept allows for the system to be installed onboard the aircraft as well as on the ground. The concept is applicable to flight operations both within and beyond the AAM operational environment.

2.2 Advanced Air Mobility Domain

Although the concept of dynamic flight path planning is applicable and essential to safe and efficient flight operations in various aviation environments, an AAM operational environment is selected as the domain in which to develop the DPP automation system architecture due to its complexity and evolving nature. The AAM operational environment defined in this report is characterized by multiple fleet operators, operating aircraft of various types, each commanded by an onboard or a remote pilot, and supported by flight path coordinators. Also, flight planners plan the preflight flight paths and coordinate them with other airspace and aerodrome users. Additionally, data providers share the relevant AAM operating environment data among the domain participants. The key AAM participants and their primary roles with respect to dynamic flight path planning are:

Fleet Operator is responsible for the planning and management of one or more flights in its fleet. The fleet operator also sets operator specific procedural constraints and preferences for flight path planning.

Mission Manager is responsible for specifying a flight's mission. The flight mission may be either a scheduled or an on-demand operation.

Flight Planner is responsible for planning the baseline flight path to execute a specified flight mission, and coordinating it with aerodrome and airspace users.

Pilot-in-Command is responsible for the safe and efficient execution of the flight mission. The pilot may be either onboard the aircraft or at a remote ground location. The pilot may be required to explicitly confirm operational acceptability of a revised flight path prior to its activation.

Aerodrome Coordinator is responsible for coordinating flight paths among the aircraft operating, and planned to be operating, at an aerodrome.

Airspace Coordinator is responsible for coordinating flight paths among the aircraft operating, and planned to be operating, in the airspace.

Aircraft provides (a) sensed operating environment data, (b) sensed avionics data, (c) propulsion system data, and (d) safety-critical maneuver commands to the DPP automation system. The aircraft is also responsible for executing the planned flight path.

Aerodrome Data Provider is responsible for sharing the relevant aerodrome data among the AAM operational domain participants.

Airspace Data Provider is responsible for sharing the relevant airspace data among the AAM operational domain participants.

Atmosphere Data Provider is responsible for sharing the relevant atmosphere data among the AAM operational domain participants.

Obstacle Data Provider is responsible for sharing the relevant obstacle data among the AAM operational domain participants.

Traffic Data Provider is responsible for sharing the relevant traffic data among the AAM operational domain participants.



The collection of the aerodrome, airspace, atmosphere, obstacle, and traffic data constitutes the dynamic operating environment. The DPP automation system responds automatically to changes in the dynamic operating environment. The DPP automation system also takes into consideration the static operating environment, which is comprised of the terrain data, stationary obstacle data, navigation charts, and aerodrome and airspace operating procedures.

2.3 Model-Based Systems Engineering Approach

There are several approaches to architecting an automation system. These range from heuristic analyses, to classical document-based systems engineering, to modern model-based systems engineering. The selected approach depends on the scope and purpose of the architecture. An example of a heuristic approach is to reuse a previous architecture as a starting point for a new system; a benefit may be the speed of creating an architecture, but the primary shortcoming of this approach is that it does not account for the differences between the two systems' concepts of operations. Any system architecture based on a mismatched concept of operation is prone to failure. An example of a classical systems engineering approach is the application of the NASA systems engineering process¹. The existing NASA systems engineering process was analyzed and found to be unsuitable for developing an efficient, reusable, and easily extensible system architecture for the subject DPP automation system. A key disadvantage of this approach, as of any document-based process, is that it tends to produce static documents which once subjected to any revision become disjointed and unmanageable. The process is ill-equipped to handle the frequently changing stakeholder needs and underlying concepts typical of a complex system, especially in an unfolding domain like the AAM. With these considerations, a model-based systems engineering approach was deemed most suitable to architect a DPP automation system.

¹NASA Procedural Requirements NPR 7123.1D: NASA Systems Engineering Processes and Requirements

Every practical MBSE design approach is built upon three pillars: a modeling language, a modeling methodology, and a modeling tool [7]. SysML[®] and MagicDraw[®] were selected as the modeling language and the tool of choice in this work [6, 9]. The search for an existing MBSE methodology for architecting the DPP automation system was unfruitful because none of the methodologies were found to be sufficiently rigorous while also being intuitive to use. A survey of the state-of-the-art MBSE processes and best practices ranked the International Council on Systems Engineering (INCOSE) object-oriented systems engineering method (OOSEM) and Dassault MagicGrid[®] as top contenders for architecting a DPP automation system [4, 8]. While the OOSEM process provides a good overview of what system artifacts to produce and in which order, it does not provide instruction on how to develop them. It was found to be too abstract to be easily applicable to the construction of a DPP automation system. MagicGrid is a methodical process that walks systems engineers through a sequence of steps required to architect a relatively simple system, offers a rationale behind each step, and provides MagicDraw-specific implementation instructions. At the time of this writing, a standard model-based architecture development process does not exist at NASA. A NASA Systems Modeling Handbook For Systems Engineering² attempts to adapt the MagicGrid methodology to comply with the NASA document-based systems engineering process. As a result, it loses the intuitive nature of the MagicGrid approach and falls short of providing a coherent methodology to architect a complex system.

A new, applied MBSE approach was developed from the first principles of systems engineering in order to architect the DPP automation system. The key features of the new methodical approach are its simplicity, rigor, and extensibility to disparate automation systems in aviation, space, and science domains. As a model-based approach, it facilitates improved communication among stakeholders, inherent consistency between the behavioral and structural elements of the system architecture, and an automated traceability between architecture elements. The methodology, summarized in Figure 1, begins with a clear description of the underlying concept. The concept is then formalized into a conceptual system architecture model which provides the contextual and operational views of the system of interest in the selected domain. The behavioral and structural elements of the conceptual architecture are then decomposed in the functional system architecture model in order to specify the system requirements and measures of effectiveness. One or more logical system architectures that meet the system requirements may then be developed, and a trade study based on the measures of effectiveness performed to select an optimal logical system architecture configuration. Any nontrivial system will require building logical architectures of the system parts, i.e., subsystems, identified in the logical system architecture model. The decomposition of the logical system continues recursively until the subsystem elements can be traced unambiguously to satisfy the needs of the stakeholders identified in the conceptual system architecture. At this point, one or more physical system architectures may be developed for a given logical system architecture. The traceability between the system architecture elements within and among the conceptual, functional, logical, and physical architectures is established throughout the process and maintained dynamically by the model.

The architecture at every level (conceptual, functional, logical, and physical) describes

²NASA-HDBK-1009A: NASA Systems Modeling Handbook for Systems Engineering

the system behavior, system structure, system requirements, and system parameters. The structure and behavior of a system represent its static and dynamic views, while the system requirements and parameters define what a built system does and how well it does that, respectively. In the SysML version 1.6 parlance:

- System behavior may be modeled as use case diagrams, activity diagrams, interactions/sequence diagrams, and state machine diagrams.
- System structure may be modeled as block definition and internal block diagrams.
- System requirements may be modeled as requirement tables and diagrams.
- System parameters may be modeled as parametric diagrams.

These model elements of the system architecture are organized as package diagrams. An intuitive, domain-agnostic package structure with basis in systems engineering lifecycle steps is vital to the reusability and extensibility of the methodology.

2.4 System Architecture Elements

The DPP automation system architecture is developed based on a clear DPP concept defined in §2.1, in the AAM domain described in §2.2, using the MBSE methodology presented in §2.3. Figure 2 depicts a domain-agnostic structure of a system architecture model. The system architecture model elements are named and organized to mimic a typical systems engineering lifecycle process. Specifically,

Conceptual architecture of the system describes a black box view of the system in the context of the selected operating environment.

Functional architecture of the system provides further clarity into the system concept, via decomposition of the conceptual behavioral and structural elements, needed to derive system requirements at a sufficient level of detail.

Logical architecture of the system describes an implementation-agnostic collection of functional and structural elements of the system.

Logical architecture of the system parts, i.e., subsystems, describes implementation-agnostic collection of functional and structural elements of the parts of the system, identified previously.

Physical architecture describes an implementation-specific collection of functional and structural elements of the system.

The remainder of this document describes the conceptual, functional, and logical architectural elements that constitute the DPP automation system. A physical system architecture straddles the boundary between the architecture and design phases of systems engineering; therefore, it belongs more appropriately in a system design document than in a system architecture description. System design, derived from the output of this system architecture description, is one of the next steps in the systems engineering process to be addressed in a future document.



Figure 2: DPP automation system architecture model structure.

3 Conceptual System Architecture

This section describes the conceptual architecture of the DPP automation system in an AAM operating environment. The purpose of the conceptual architecture is to formalize the dynamic path planning concept presented in [10]; the formalization enables automated end-to-end traceability between the system concept and system architecture.

3.1 Synopsis

As shown in Figure 1, the steps in developing a conceptual architecture are³:

1. Identify the key stakeholders of the system.
2. Elicit concerns of the identified stakeholders.
3. Analyze and express the relevant stakeholder concerns as system goals.
4. Define the scope of the domain in which the system operates.
5. Define the context of the system in the selected domain.
6. Refine the functional goals by a set of use cases.
7. Refine the nonfunctional goals by system measures of effectiveness.
8. Derive system requirements from the system context, use cases, and measures of effectiveness.

Section 3.2 describes the key stakeholders of the DPP automation system and their concerns. Stakeholders may be humans, organizations, and other systems whose interests need to be considered in designing the system. The DPP automation system stakeholders include pilots, mission managers, flight planners, flight path coordinators, and aircraft, as well as passengers, cargo shippers, impacted communities, and regulatory authorities. Some stakeholders may have unique concerns such as specific regulations, while others' concerns, like safety of flight, may be overlapping. Figure 3 maps the 20 highest priority stakeholders of the DPP automation system to their 48 concerns, grouped in terms of desired capabilities, expected performance, operational and design constraints, and benefits of the research.

Section 3.3 presents an analysis of the stakeholder concerns that offers an insight into the expected capabilities, attributes, and external interactions of the DPP automation system. The list of 48 stakeholder concerns is expressed in the form of 23 system goals, organized as functional goals, performance goals, interface constraints, and design constraints. Figures 4 and 5 trace the system goals to the underlying stakeholder concerns.

Section 3.4 presents a further analysis of the stakeholder concerns that aids in defining the AAM domain participants which interact with the DPP automation system. These participants include a subset (actors) of the stakeholders and the operating environment—the aerodrome, airspace, atmosphere, obstacle, and traffic—data providers. Figure 6

³Although presented as a sequence, these steps are performed iteratively until convergence is achieved with respect to the desired consistency and completeness of the architecture.

defines the data communication interfaces among the AAM operational domain participants, including the ground-based and onboard instances of the DPP automation system. Figure 7 and 8 depict the data which may flow across these interfaces between the DPP automation system and the AAM domain actors before and during the flight, respectively.

Section 3.5 refines the functional goals of the DPP automation system into a set of use cases. Figures 9–10 depict the manner in which the DPP automation system users may use the system before and during the flight. Figures 11–19 describe the activities that constitute the most significant use cases. The flight planner uses the ground-based DPP automation system to build a flight path and coordinate it with aerodrome and airspace flight path coordinators before the flight (§3.5.1, §3.5.2). Once a ground-to-air connection is established with the aircraft, the mission manager works with the pilot to synchronize this flight path with the onboard DPP automation system (§3.5.3). During the flight, authorized users may request the onboard DPP automation system to adapt the flight path to changing conditions, to ensure that the system complies with any safety-critical command, and to monitor flight status and system performance. Requests may encompass, but are not limited to the following:

- The pilot requests the onboard DPP automation system to adapt the flight path in response to a change in flight mission (§3.5.4).
- The mission manager expects the onboard DPP automation system to adapt the flight path autonomously to changes in the operating environment (§3.5.5).
- The mission manager expects the onboard DPP automation system to adapt the flight path autonomously to aircraft-sensed events (§3.5.6).
- The pilot requests the onboard DPP automation system to adapt the flight path in response to an observed event (§3.5.7).
- An authorized user may request the onboard DPP automation system to assess a proposed flight path change (§3.5.8).
- The aircraft or the pilot may command the onboard DPP automation system to pause, stop, or resume flight path planning⁴.
- The mission manager expects the onboard DPP automation system to refresh the flight path periodically (§3.5.11).
- An authorized user may request the onboard DPP automation system to monitor the flight path⁴.
- The mission manager may request the onboard DPP automation system to report its health status periodically⁴.
- The mission manager expects the onboard DPP automation system to adapt the flight path autonomously to mitigate an identified contingency event (§3.5.9).

⁴This use case is not described in the same level of depth as others because the underlying concept requires further investigation at the time of this writing.

Every use case that requires the DPP automation system to update the flight path performs a set of common steps listed in (§3.5.12), including coordination of the candidate flight path with the relevant AAM domain participants (§3.5.13).

Section 3.6 refines the performance goals of the DPP automation system into measures of effectiveness—a set of measurable attributes of the system that determine how well the built system meets its functional goals.

Section 3.7 derives the top-level system requirements from the use cases, system context, and system measures of effectiveness. Specifically, the use cases provide functional requirements (Table 1); the system context translates into interface requirements (Table 5); and the system measures of effectiveness translate into usability requirements, physical requirements, and design constraints (Table 2–4).

3.2 Stakeholders

A stakeholder is any entity—an individual, a system, or an organization—who may have a vested interest in the system, may be affected by the system, or may have an influence on the design of the system. Typical stakeholders are the system users, operators, architects, designers, developers, owners, regulators, and maintainers. Examples of atypical stakeholders include a competitor whose capabilities may influence the design of the system of interest, and a cybercriminal whose potential acts may impose security constraints on the system of interest. Stakeholders can be classified as actors and non-actors. An actor is an external entity—an individual, a system, or an organization—that interacts with the system of interest. Recognizing primary stakeholders and prioritizing their concerns is vital to the architecting, designing, and building of a valid system. Every element of the built system should, either directly or transitively, be traceable to one or more of the stakeholder concerns.

A DPP automation system will be used primarily for planning, monitoring, revising, and coordinating an aircraft’s flight path in order to achieve the specified mission. Additionally, the DPP system architecture should offer guidance to the industry, certification authorities, and service providers in establishing best design practices, community standards, and means of compliance recommendations. A list of primary stakeholders of a DPP automation system in the AAM domain includes:

- | | |
|--------------------------------------|--|
| 1. Fleet operator | 9. System administrator |
| 2. Mission manager | 10. Passenger |
| 3. Flight planner | 11. Cargo shipper |
| 4. Onboard pilot-in-command | 12. Impacted community |
| 5. Remote pilot-in-command | 13. Standards development organization |
| 6. Aerodrome flight path coordinator | 14. Regulation authority |
| 7. Airspace flight path coordinator | 15. Program manager |
| 8. Aircraft systems | 16. Automation system designer |

The first nine stakeholders in this list may play the role of an actor with respect to the DPP automation system, while the other stakeholders are non-actors, i.e., they do not directly interact with the DPP automation system.



The DPP automation system also interacts with non-stakeholder (secondary) actors who share the operating environment data among domain participants, but do not have a stake in either design or operation of the DPP automation system. These include the aerodrome, airspace, atmosphere, obstacle, and traffic data service providers.

Having identified key stakeholders, the next step is to elicit their concerns about the capabilities and design of the DPP automation system. A stakeholder concern may be expressed as a need; as a goal or a desire; as operational and regulatory constraints; or, as physical constraints on the system. A stakeholder may have multiple concerns, and multiple stakeholders may have common concerns. Figure 3 lists the DPP automation system stakeholders and their concerns, grouped in terms of the desired capabilities, expected performance, operational and design constraints, and benefits of the research to the aviation community. A SysML allocation matrix relates concerns to the stakeholders in the direction of the arrows. The list of concerns includes:

Desired Capabilities

1. A system generated flight path must be executable by the selected aircraft.
2. A system generated flight path must be safe with respect to all known operating environment hazards.
3. A system generated flight path must comply with all known operating environment constraints.
4. A system generated flight path must comply with the specified flight mission.
5. A system generated flight path should be predictable.
6. A system generated flight path should comply with user preferences and selected rules.
7. A system generated flight path should contribute to a comfortable flight.
8. A system generated flight path should strive to reduce the noise and other environmental impacts of the flight.
9. The onboard automation system should be initialized with the flight path and associated context information generated by the flight planning system.
10. The system must be able to revise the active flight path automatically in the event of a flight path contingency.
11. The system must construct a baseline flight path to execute a specified flight mission.
12. The system must prioritize all commands issued by the aircraft tactical maneuver management system to pause and resume dynamic flight path planning during the flight.

13. The system must prioritize all tactical commands issued by the pilot to pause and resume dynamic flight path planning during the flight.
14. The system must share the computed flight path with other aircraft in the airspace.
15. The system operating mode should be configurable to nominal ground and onboard operation as well as to shadow mode and simulations for verification purpose.
16. The system should adapt the flight path autonomously in response to a change in aircraft conformance to the planned flight path.
17. The system should adapt the flight path autonomously in response to sensed, observed, or notified changes in operating environment.
18. The system should alert the user if a valid flight path could not be computed to meet the specified flight mission.
19. The system should archive all flight path results and anomalies for post-flight analysis.
20. The system should ensure continued safety, efficiency, and operational acceptability of the planned flight path through the life of the flight.
21. The system should evaluate a proposed flight path change during the flight, and accept it if deemed viable, in order to support safety, capacity, and efficiency goals of the operator, the aerodromes, or the airspace.
22. The system should facilitate coordination of the planned flight path with the selected airspace coordinator for strategic deconfliction against applicable airspace users.
23. The system should facilitate coordination of the planned flight path with the selected aerodrome coordinator for compliance with aerodrome availability schedule.
24. The system should keep track of the flight path qualities periodically and update the flight path in the event of a degradation in flight path qualities.
25. The system should provide an alternative flight path in response to an observed onboard emergency.
26. The system should provide flight progress status periodically and inform about flight path anomalies in a timely manner.
27. The system should provide system health status updates periodically and inform about health anomalies in a timely manner.
28. The system should update the active flight path during flight in response to a new flight mission that enhances fleet-wide efficiency.
29. The system should update the active flight path during flight in response to a new flight mission that addresses a pilot observed change in flight deck or cabin.

Expected Performance

1. The system should be available, i.e., perform as designed, continuously through the flight.
2. The system should continue to operate in the event of external interface failures, even if that means altering the original flight mission.
3. The system should exhibit a high level of integrity, defined in terms of minimizing missed alerts and false alerts.

4. The system should minimize additional pilot workload related to managing the planned flight path in a dynamic operating environment.
5. The system should respond to all user inputs in a timely manner.
6. The system should support autonomous flight operation with and without an onboard pilot.

Operational and Design Constraints

1. Only authorized users and systems should have access to the system.
2. The system must be able to support diverse fleet operator missions.
3. The system must comply with relevant air navigation service provider (ANSP) regulations.
4. The system must have access to the selected aircraft systems, flight path coordinators, and operating environment service providers needed to execute missions.
5. The system must meet all applicable regulatory standards applicable to onboard flight path automation system.
6. The system should be configurable to support an aerodrome coordinator mission to support flight path harmonization with aerodrome users expected to be at the airport during a specified period.
7. The system should be configurable to support an airspace coordinator mission to support flight path harmonization and conflict resolution with airspace users predicted to be in the airspace during a specified period.

Research Benefits

1. The dynamic path planning system architecture should provide guidance on designing a production version.
2. A reference system design should facilitate determining target levels of safety and performance for future operational systems.
3. The architecture must facilitate automated verification and validation of a reference system implementation.
4. The architecture must facilitate timely and cost-effective system design updates as stakeholder needs evolve.
5. The architecture should be standardized and automated for extensibility, reusability, and ease of modifications to support future research objectives.
6. The system architecture should be easily modifiable to comply with disparate requirements imposed by different certification authorities.

The stakeholder concerns lay the groundwork for developing the conceptual architecture of the DPP automation system.

Legend		Actor Stakeholders															Non-Actor Stakeholders									
concern		Aerodrome Coordinator Aircraft Avionics Sensors Aircraft Communication System Aircraft Flight Path Execution Sys Aircraft Propulsion Sensors Aircraft Systems Aircraft Tactical Maneuver Manag Airspace Coordinator Fleet Operator Flight Planner Mission Manager Pilot System Administrator															Cargo Shipper Community Passenger Production System Designer Program Manager Regulation Authority Standards Developer									
Stakeholders Concerns		3	2	1	1	2	1	1	3	12	5	14	12	5			2	1	2	2	3	2	1			
Desired Capabilities		1	2	1	1	2	1	1	1	4	5	14	10	2			1	1	2							
A system generated flight path must be executable ...		4																								
A system generated flight path must be safe with r...		3																								
A system generated flight path must comply with al...		3																								
A system generated flight path must comply with th...		2																								
A system generated flight path should be predictab...																	2									
A system generated flight path should comply with ...		2																								
A system generated flight path should contribute t...																	1									
A system generated flight path should strive to re...																	1									
The onboard automation system should be initialize...		2																								
The system must be able to revise the active fligh...		1																								
The system must construct a baseline flight path t...		2																								
The system must prioritize all commands issued by ...		2																								
The system must prioritize all tactical commands i...		1																								
The system must share the computed flight path wit...		1																								
The system operating mode should be configurable t...		1																								
The system should adapt the flight path autonomou...		3																								
The system should adapt the flight path autonomou...		3																								
The system should alert the user if a valid flight...		1																								
The system should archive all flight path results ...		1																								
The system should ensure continued safety, efficie...		1																								
The system should evaluate a proposed flight path ...		4																								
The system should facilitate coordination of the p...		1																								
The system should facilitate coordination of the p...		1																								
The system should keep track of the flight path qu...		1																								
The system should provide an alternative flight pa...		1																								
The system should provide flight progress status p...		1																								
The system should provide system health status per...		1																								
The system should update the active flight path du...		1																								
The system should update the active flight path du...		1																								
Expected Performance			1							1	5			2	1		1									
The system should be available, i.e., perform as d...		1																								
The system should continue to operate in the event...		1																								
The system should exhibit a high level of integrit...		1																								
The system should minimize additional pilot worklo...		1																								
The system should respond to all user inputs in a ...		5																								
The system should support autonomous flight operat...		1															1									
Operational and Design Constraints			1							1	3			2									2			
Only authorized users and systems must have access...		1																								
The system must be able to support diverse fleet o...		1																								
The system must comply with relevant air navigatio...		1															1									
The system must have access to the selected aircra...		1																								
The system must meet all applicable regulatory sta...		1															1									
The system should be configurable to support an ae...		1																								
The system should be configurable to support an ai...		1																								
Research Benefits																										
The architecture must facilitate automated verific...																	1									
The architecture must facilitate timely and cost-e...																	1									
The architecture should be standardized and automa...																	1									
The dynamic path planning system architecture shou...																	1									
The reference system design should facilitate dete...																	1									
The system architecture should be easily modifiabl...																	1									

Figure 3: DPP automation system stakeholders and their concerns.

3.3 System Goals

Once the primary DPP stakeholders and their concerns have been identified, the next step in conceptual architecture development is to consolidate these concerns into a set of goals which the DPP automation system must meet. The system goals, usually expressed as business requirements in MBSE models, constitute:

- the capabilities that the system must possess in order to meet the stakeholder needs,
- the attributes the system must possess in order to meet the desired performance,
- the interfaces the system must possess in order to operate in the domain, and,
- the attributes the system must possess in order to comply with the design constraints.

The goals are organized as functional goals, performance goals, interface goals, and design goals in the conceptual architecture model.

Functional Goals

1. Flight Path Planning: Plan an initial flight path that complies with the mission specification.
2. Flight Path Synchronization: Ensure that the onboard DPP automation has an acceptable baseline flight path.
3. Operating Environment: Adapt the flight path to changes in the operating environment.
4. Observed Events: Adapt the flight path in response to anomalies observed by the onboard pilot.
5. Flight Mission Change: Update the flight path in response to a change in flight mission.
6. Flight Conformance Monitoring: Monitor flight conformance and, if necessary, update the flight path to account for changes.
7. Tactical Command Override: Comply with all tactical maneuver commands from the aircraft and the pilot.
8. Flight Path Proposal Evaluation: Evaluate a proposed flight path change from the pilot, the mission manager, or an authorized flight path coordinator.
9. Flight Path Contingency Mitigation: Mitigate identified and anticipated flight path contingencies.
10. Flight Path Status: Keep track of the flight path status and anomalies.
11. System Health Status: Monitor system health and anomalies.
12. System Configuration Changes: Determine the impact of changes to system configuration parameters.
13. Flight Path Refresh: Refresh the flight path periodically.
14. Flight Path Update: Build a new candidate flight path, as needed.

15. Flight Path Coordination: Coordinate a newly built flight path with the pilot, the aircraft systems, and the selected flight path coordinators.
16. Community Impact: Minimize noise and environmental impact of the flight.

Performance Goals

1. System Integrity: Ensure that the system provides acceptable level of alerting accuracy and integrity.
2. Pilot Workload: Minimize pilot workload related to flight path management during the flight.
3. System Availability: Ensure that the system performs as designed through the life of the flight.

Interface Goals

1. External Communication: Ensure that the communication links, essential to execute a specified flight mission, exist between the system and the aircraft, operating environment data providers, and authorized users.

Design Goals

1. Regulatory Compliance: Ensure that the system complies with all regulations and certification standards applicable to its domain of operation.
2. System Security: Ensure that the system is accessible only to authorized users.
3. System Extensibility: Ensure that the system supports dynamic flight path planning in multiple aviation domains.

The model maintains traceability between system goals and the parent stakeholder concerns, as shown in Figures 4 and 5. The goals form the basis of the remainder of the conceptual architecture. Specifically, the interface goals define the system context (§3.4), functional goals are refined by use cases (§3.5), performance goals are refined by measures of effectiveness (§3.6), and design goals are included in specifying the top-level system requirements (§3.7).

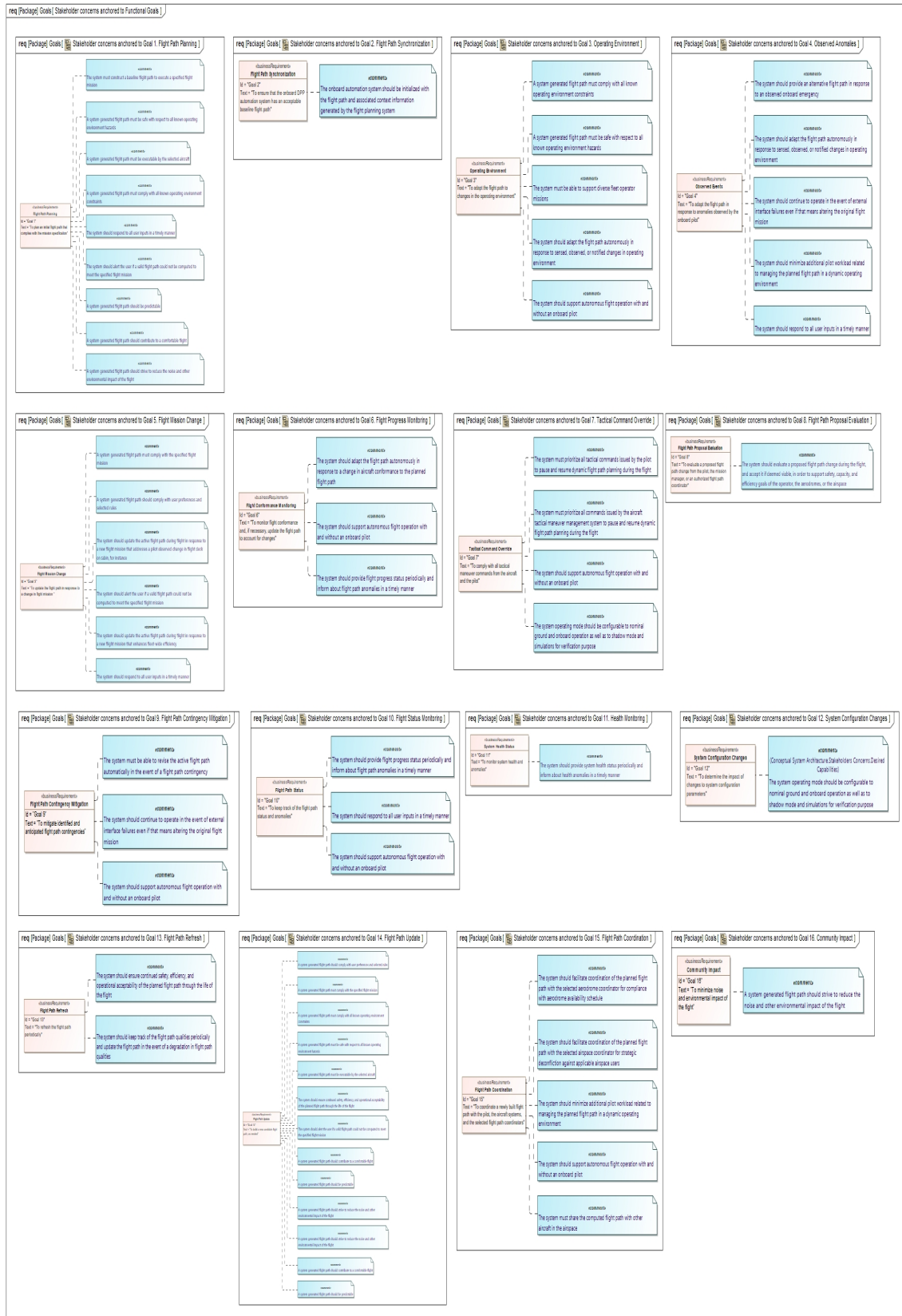


Figure 4: Stakeholder concerns consolidated into functional goals.

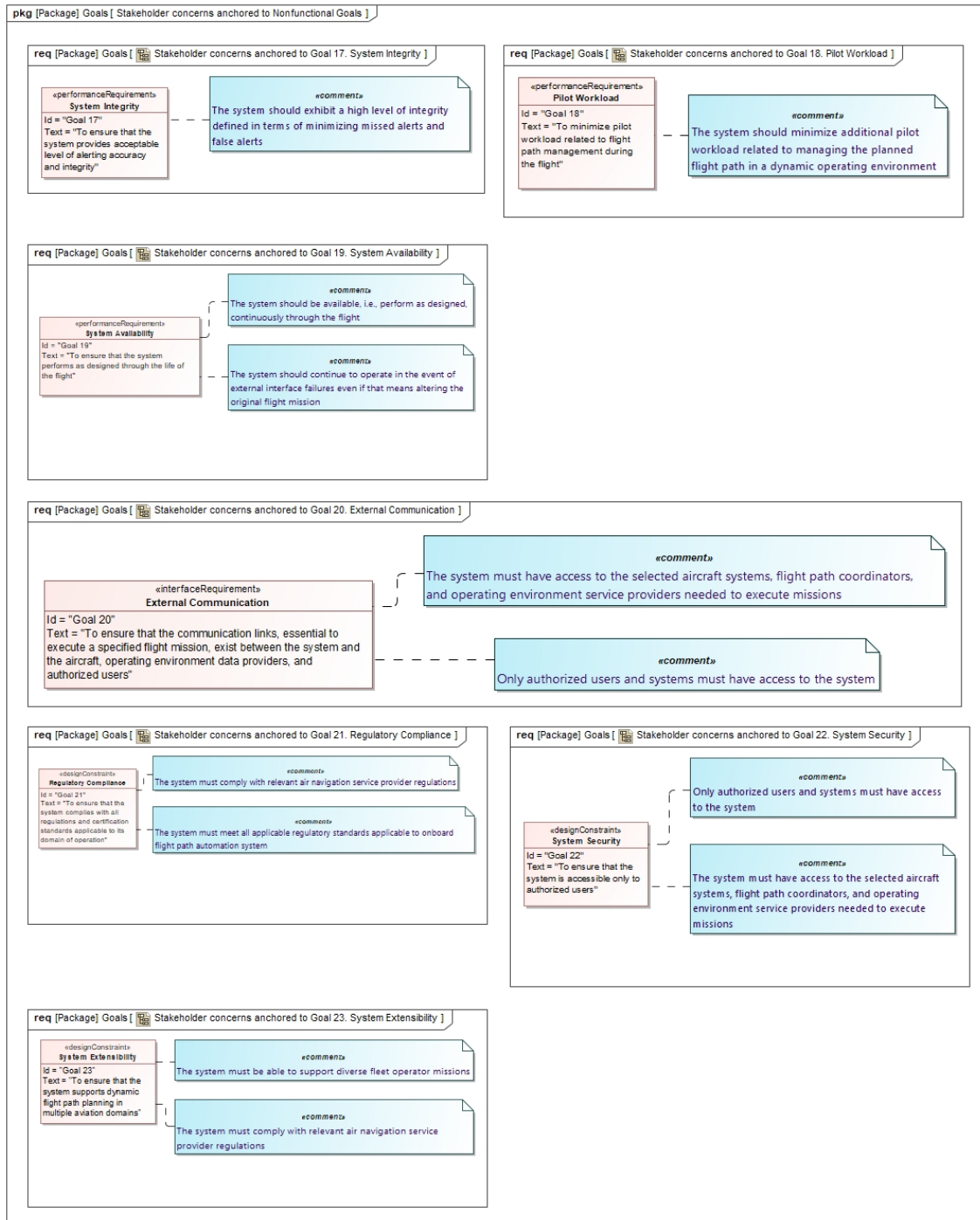


Figure 5: Stakeholder concerns consolidated into nonfunctional goals.

3.4 System Context

The system goal related to the external communication between the DPP automation system and the relevant AAM domain participants sets the system boundary. The system interacts with the primary and secondary actors listed in §3.2. An analysis of the system goals defines the types of data exchanged during these interactions. Figure 6 presents the DPP automation system in the context of the AAM operational domain defined in this report. The domain contains two instances of the DPP automation system (cyan) and the AAM domain participants (yellow) with whom these instances interact. The connectors between the blocks display the type of connections and the multiplicity of the connected entities. The AAM domain is characterized by multiple fleet operators operating aircraft of various types being commanded by onboard and/or remote pilots, and supported by airspace and aerodrome flight path coordinators. Additionally, flight planners are responsible for building the initial flight path and coordinating it with other airspace and aerodrome users. The operating environment data providers share a common operating picture among the domain participants by providing the latest aerodrome, airspace, atmosphere, obstacles, and traffic information.

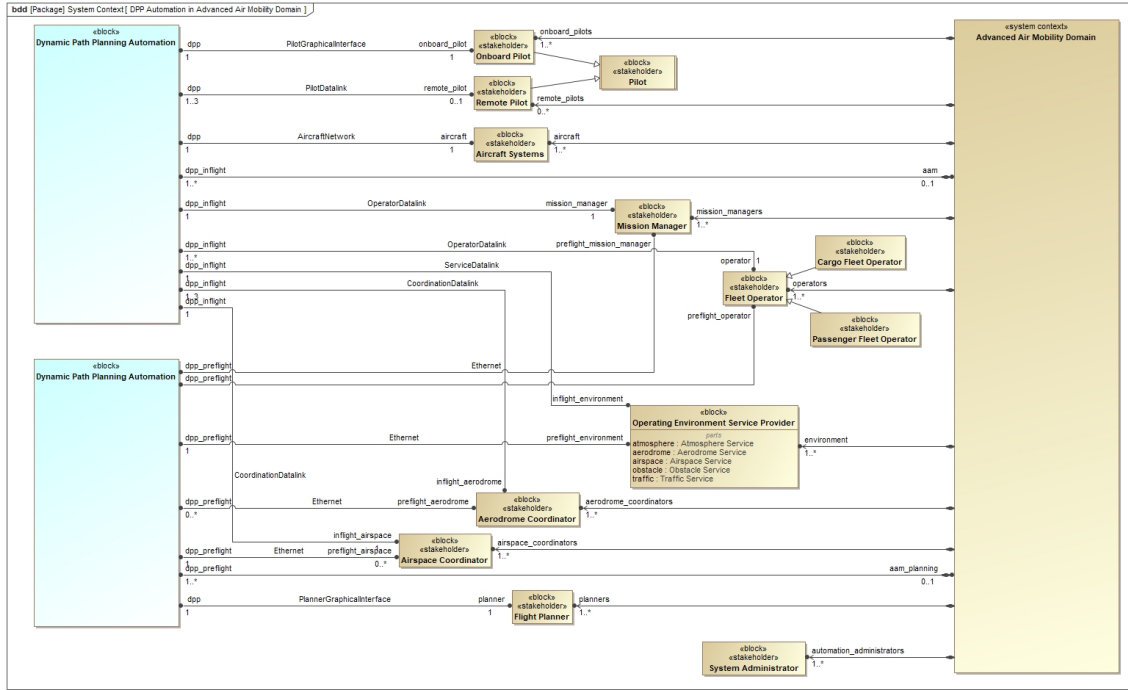


Figure 6: Preflight and inflight roles of DPP automation system in an AAM operational domain.

The concept of dynamic flight path planning is applicable and essential to safe and efficient flight operations in several aviation domains, including airline operations, AAM, UAS, and high-altitude platform systems; the architecture in this document focuses on the DPP automation system in the context of an AAM operational domain. The design of a DPP automation system needs to account for its lifecycle, including design, administration, operation, maintenance, enhancements, and decommissioning; the architecture in this

document focuses on the DPP automation system in operation. A DPP automation system may be used during the preflight phase for planning, the inflight phase for managing the flight path, and the postflight phase for performance analyses; the architecture in this document focuses on the preflight and inflight phases of the flight. To summarize, the scope of the DPP automation system architecture is the system operation during the preflight and inflight phases in an AAM operational domain.

The DPP automation system interacts with the following participants in the context of the AAM operational domain, as shown in Figures 7–8.

Flight planner interacts with the ground-based DPP automation system during the preflight phase via the flight planner graphical user interface. The data that the flight planner sends to the system includes flight planning request, and response to flight path acceptability request. The data that the flight planner receives from the system includes flight path information, system health, relevant alerts, and requests to review the planned flight path.

Onboard pilot interacts with the onboard DPP automation system during the flight via the pilot graphical user interface. The data that the onboard pilot sends to the system includes flight mission change, observed events, tactical override command, flight path proposal, flight path acceptability response, system configuration parameters, and system operating mode. The data that the onboard pilot receives from the system includes flight path information, system health, flight path proposal response, relevant alerts, requests to review a candidate flight path, and system configuration.

Remote pilot interacts with the onboard DPP automation system during the flight via a remote pilot datalink. The data that the remote pilot sends to the system includes flight mission change, flight path status request, flight path proposal, system configuration parameters, and system operating mode. The data that the remote pilot receives from the system includes flight path information, flight path status, system health, flight path proposal response, relevant alerts, and system configuration.

Mission manager interacts with the ground-based DPP automation system during the preflight phase via Ethernet, and with the onboard DPP automation system during the flight via the operator data link. The data that the mission manager sends to the system includes flight path status request and flight path proposal to the onboard system. The data that the mission manager receives from the onboard system includes flight path status, system health, flight path proposal response, and relevant alerts. The mission manager also receives the planned flight path from the ground-based system and sends it to the onboard system, as part of the flight path synchronization before the flight.

Aircraft avionics system interacts with the onboard DPP automation system during the flight via the aircraft network. The data that the aircraft avionics system sends to the system includes sensed aircraft data.

Aircraft propulsion system interacts with the onboard DPP automation system during the flight via the aircraft network. The data that the aircraft propulsion system sends to the system includes sensed propulsion data.

Aircraft tactical maneuver management system interacts with the onboard DPP automation system during the flight via the aircraft network. The data that the aircraft tactical maneuver management system sends to the system includes tactical override command.

Aircraft guidance interacts with the onboard DPP automation system during the flight via the aircraft network. The data that the aircraft guidance system sends to the system includes flight path execution status. The data that the aircraft guidance system receives from the system includes flight path execution request.

Aircraft communication system interacts with the onboard DPP automation system during the flight via the aircraft network. The data that the aircraft communication system sends to the system includes flight path transmission status. The data that the aircraft communication system receives from the system includes flight path transmission request.

Aerodrome coordinator interacts with the ground-based DPP automation system during the preflight phase via Ethernet, and with the onboard DPP automation system during the flight via coordination data link. The data that the aerodrome coordinator sends to the system includes flight path coordination response, flight path proposal, and flight path status request. The data that the aerodrome coordinator receives from the system includes flight path coordination request, flight path proposal response, flight path status, and relevant alerts.

Airspace coordinator interacts with the ground-based DPP automation system during the preflight phase via Ethernet, and with the onboard DPP automation system during the flight via coordination data link. The data that the airspace coordinator sends to the system includes flight path coordination response, proposed flight path proposal, and flight path status request. The data that the airspace coordinator receives from the system includes flight path coordination request, proposed flight path proposal response, flight path status, and relevant alerts.

Aerodrome data provider interacts with the ground-based DPP automation system during the preflight phase via Ethernet, and with the onboard DPP automation system during the flight via service data link. The data that the aerodrome data provider sends to the system includes complete aerodrome data upon request and any changes thereafter. The data that the aerodrome data provider receives from the system includes a list of relevant aerodromes for which data is needed.

Airspace data provider interacts with the ground-based DPP automation system during the preflight phase via Ethernet, and with the onboard DPP automation system during the flight via service data link. The data that the airspace data provider sends to the system includes complete airspace data upon request and any changes thereafter. The data that the airspace data provider receives from the system includes a list of relevant airspace regions for which data is needed.

Atmosphere data provider interacts with the ground-based DPP automation system during the preflight phase via Ethernet, and with the onboard DPP automation

system during the flight via service data link. The data that the atmosphere data provider sends to the system includes complete atmosphere data upon request and any changes thereafter. The data that the atmosphere data provider receives from the system includes a list of relevant airspace regions for which atmosphere data is needed.

Obstacle data provider interacts with the ground-based DPP automation system during the preflight phase via Ethernet, and with the onboard DPP automation system during the flight via service data link. The data that the obstacle data provider sends to the system includes complete obstacle data upon request and any changes thereafter. The data that the obstacle data provider receives from the system includes a list of relevant airspace regions for which obstacle data is needed.

Traffic data provider interacts with the ground-based DPP automation system during the preflight phase via Ethernet, and with the onboard DPP automation system during the flight via service data link. The data that the traffic data provider sends to the system includes complete traffic data upon request and any changes thereafter. The data that the traffic data provider receives from the system includes a list of relevant airspace regions for which traffic data is needed.

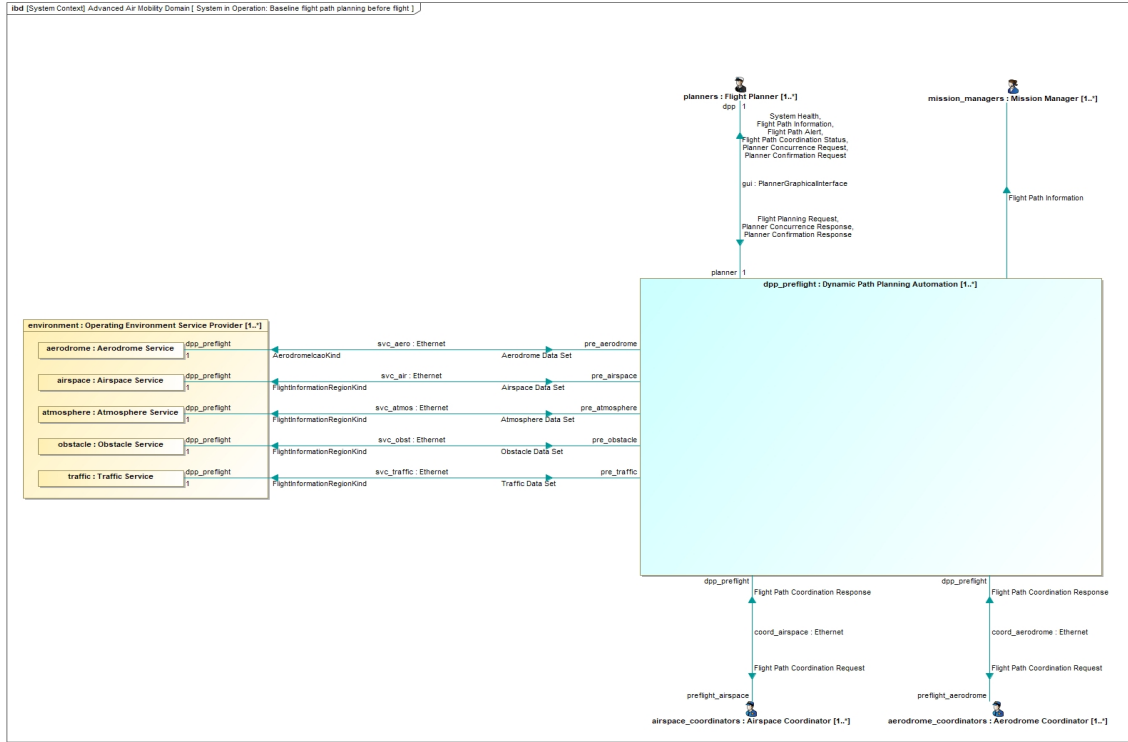


Figure 7: DPP automation system in AAM operational context during preflight phase.

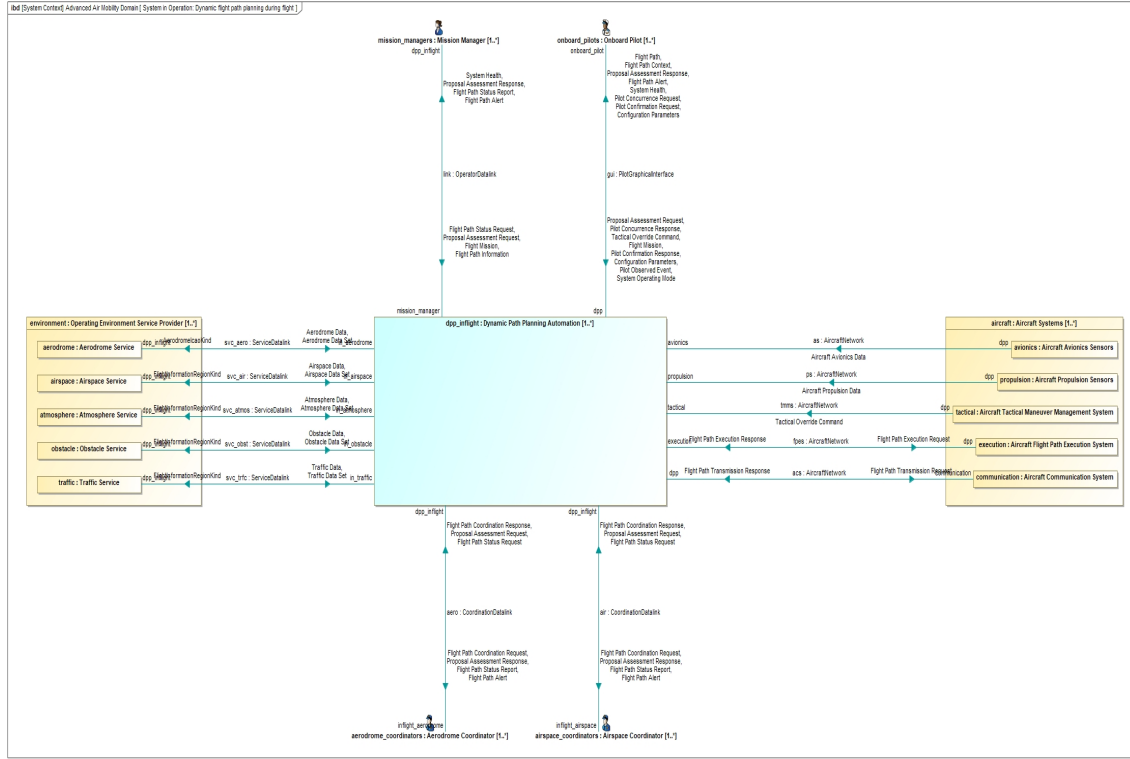


Figure 8: DPP automation system in AAM operational context during the flight.

3.5 System Usage

The functional goals of the DPP automation system are refined by a set of use cases. The use cases describe the manner in which the primary actors use the DPP automation system to achieve their goals. Figure 9 and 10 present the DPP automation system usage before and during the flight, respectively. The following is a complete list of use cases:

1. Activate Baseline Flight Path
2. Adapt Flight Path to Mission Change
3. Adapt Flight Path to Pilot Observed Event
4. Adapt Flight Path to Sensed Information
5. Adapt Flight Path to Shared Operating Environment
6. Adapt Flight Path to System Configuration Change
7. Assess Proposed Flight Path Change
8. Build Baseline Flight Path
9. Comply with Tactical Maneuver Command
10. Coordinate Baseline Flight Path

11. Coordinate Candidate Flight Path
12. Mitigate Flight Path Contingency
13. Monitor Flight Path Status
14. Monitor System Health
15. Monitor System Performance
16. Perform Postflight Analysis
17. Refresh Flight Path Periodically
18. Synchronize Baseline Flight Path
19. Update Flight Path

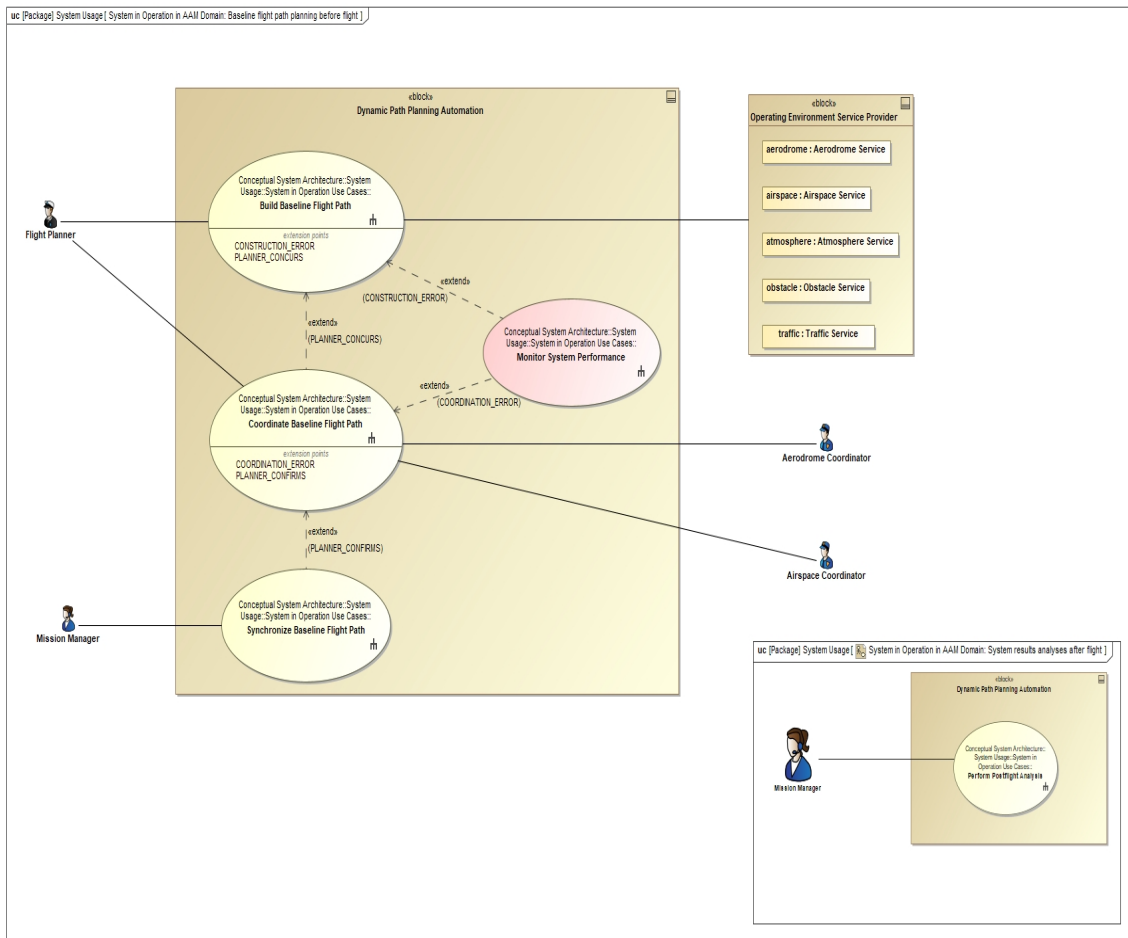


Figure 9: DPP automation system usage before (and after) the flight.

The use cases are described graphically using activity diagrams, and as narratives using the following convention [5].

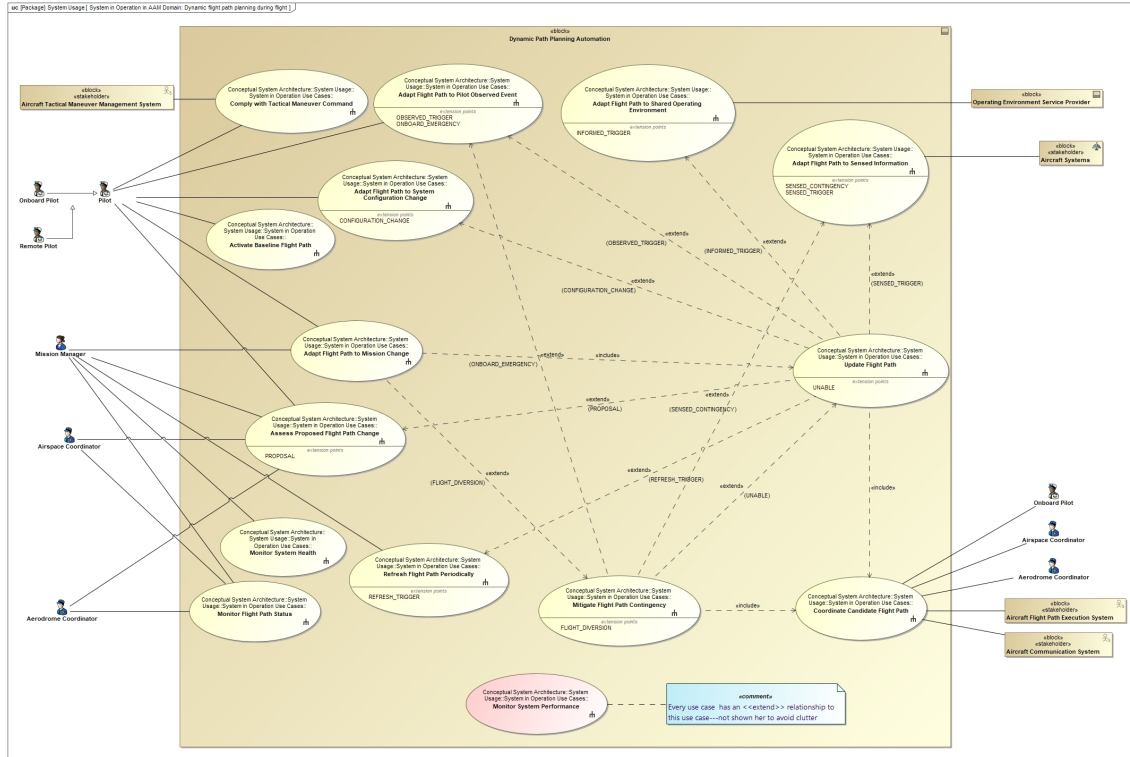


Figure 10: DPP automation system usage during the flight.

Use Case: name of the use case.

Scope: the system that enables the use case.

Primary Actor: the actor that invokes the use case.

Secondary Actor: the actor that the system relies on to accomplish the use case.

Preconditions: the conditions that must be met for the use case to be executed.

Trigger: the event that prompts the use case to begin execution.

Nominal Scenario: the primary flow of interactions between the system and actors.

Alternate Scenarios: the branches from the nominal flow to handle special conditions.

Postconditions: the conditions that must be true upon completion of the use case.

3.5.1 Build Baseline Flight Path

The flight planner uses the DPP automation system to build a baseline flight path during preflight planning. The following narrative describes this use case, with Figure 11 presenting the corresponding SysML representation.

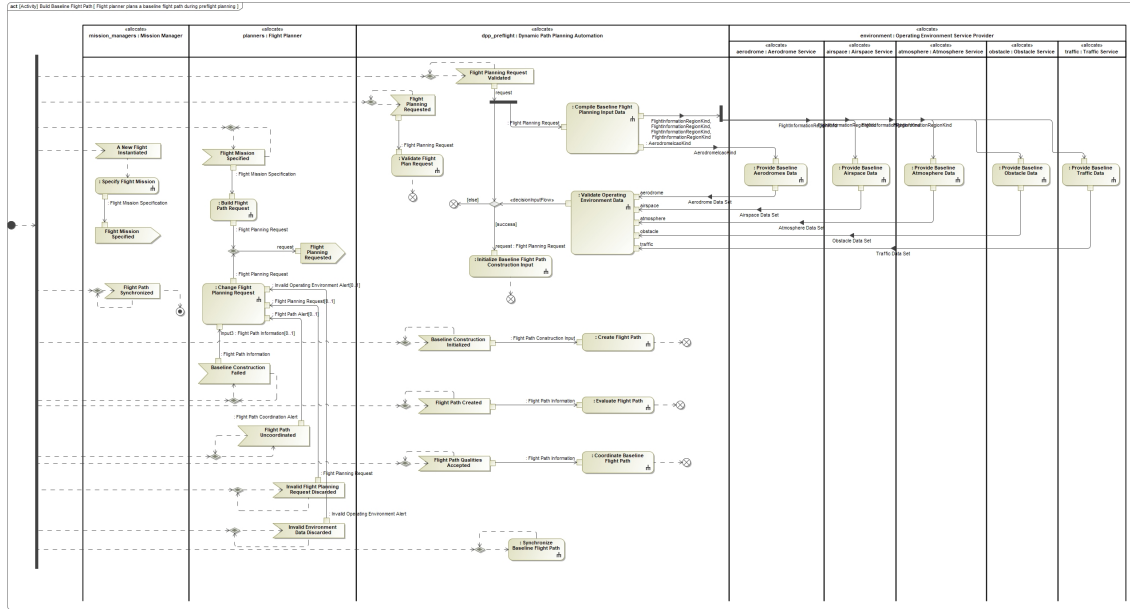


Figure 11: Flight planner builds a baseline flight path during preflight planning.

Use Case: Build baseline flight path

Scope: Ground-based DPP automation system before the flight

Primary Actor: Flight planner

Secondary Actors: Shared operating environment data providers

Preconditions: The ground-based DPP automation system is configured to interface with selected flight path coordinators and operating environment data providers.

Trigger: Flight planner receives the flight mission specification from mission manager.

Nominal Scenario:

1. The flight planner builds a flight planning request based on the mission specification and planning preferences.
2. The flight planner enters the flight planning request into the ground-based DPP automation system via the flight planner graphical user interface.
3. The DPP automation system validates the request for accuracy and completeness.

4. If the request is valid, the DPP automation system initializes the flight path planning functions.
5. The DPP automation system transmits an information request to the configured operating environment data providers indicating the aerodromes and airspace for which shared information is needed, via the configured secure Ethernet connections.
6. The operating environment data providers share the relevant aerodrome, airspace, atmosphere, obstacle, and traffic information with the DPP automation system via the configured secure Ethernet connections.
7. The DPP automation system validates the received operating environment data for timeliness, relevance and completeness.
8. Upon receipt of a complete set of valid operating environment data, the DPP automation system builds the flight path construction input based on the flight planning request and operating environment data.
9. The DPP automation system validates the flight path construction input for completeness.
10. If the flight path construction input is valid, the DPP automation system constructs a flight path.
11. The DPP automation system evaluates the feasibility⁵ of the computed flight path.
12. If the computed flight path is feasible, the DPP automation system determines if the flight path conflicts with any of the known operating environment hazards.
13. If the computed flight path is feasible and safe, i.e., not in conflict with any known hazard, the DPP automation system determines if the flight path cost meets the flight planning optimality criterion.
14. The DPP automation system displays the feasible, safe, and optimal flight path on the flight planner graphical user interface.
15. The flight planner determines if the constructed flight path is acceptable.
16. If the constructed flight path is unacceptable, the flight planner changes the flight planning request. [➡ Step 2]
17. The flight planner coordinates the constructed flight path.
18. Upon flight planner's confirmation, the DPP automation system informs the mission manager of availability of a baseline flight path via a secure Ethernet connection to the mission manager station.

Alternate Scenarios/Extensions:

- 3a. If the request is invalid:
 - 3a.1 The DPP automation system generates an "Invalid Flight Planning Request" alert.
 - 3a.2 The DPP automation system displays the alert on the flight planner graphical user interface.

⁵A feasible path is one that conforms to the aircraft performance and range capabilities; complies with the airspace and aerodrome structure, rules, and constraints; and avoids the terrain and charted obstacles.

- 3a.3 The flight planner changes the flight planning request by updating the flight planning parameters to comply with the alert. The flight planner may need to coordinate with the mission manager if a change in flight mission is warranted.
[➡ Step 2]
- 7a. If the operating environment data is invalid:
 - 7a.1 The DPP automation system generates an “Invalid Operating Environment” alert.
 - 7a.2 The DPP automation system displays the alert on the flight planner graphical user interface.
 - 7a.3 The flight planner changes the flight planning request by updating the flight planning parameters to comply with the alert. The flight planner may need to coordinate with the mission manager if a change in flight mission is warranted.
[➡ Step 2]
- 9a. If the flight path construction input is invalid:
 - 9a.1 The DPP automation system generates an “Invalid Flight Path Construction Input” alert.
 - 9a.2 The DPP automation system displays the alert on the flight planner graphical user interface.
 - 9a.3 The flight planner changes the flight planning request by updating the flight planning parameters to comply with the alert. The flight planner may need to coordinate with the mission manager if a change in flight mission is warranted.
[➡ Step 2]
- 11a. If the computed flight path is infeasible:
 - 11a.1 The DPP automation system generates an “Infeasible Flight Path” alert.
 - 11a.2 The DPP automation system displays the alert on the flight planner graphical user interface.
 - 11a.3 The flight planner changes the flight planning request by updating the flight planning parameters to comply with the alert. The flight planner may need to coordinate with the mission manager if a change in flight mission is warranted.
[➡ Step 2]
- 12a. If the computed flight path is in conflict with a known hazard:
 - 12a.1 The DPP automation system generates an “Unsafe Flight Path” alert.
 - 12a.2 The DPP automation system displays the alert on the flight planner graphical user interface.
 - 12a.3 The flight planner changes the flight planning request by updating the flight planning parameters to comply with the alert. The flight planner may need to coordinate with the mission manager if a change in flight mission is warranted.
[➡ Step 2]
- 13a. If the computed flight path is less optimal than the current:

- 13a.1 The DPP automation system generates an “Suboptimal Flight Path” alert.
- 13a.2 The DPP automation system displays the alert on the flight planner graphical user interface.
- 13a.3 The flight planner changes the flight planning request by updating the flight planning parameters to comply with the alert. The flight planner may need to coordinate with the mission manager if a change in flight mission is warranted.
[➡ Step 2]

Postconditions: A flight path which executes the specified flight mission is available.

3.5.2 Coordinate Baseline Flight Path

The flight planner uses the DPP automation system to coordinate the baseline flight path during preflight planning. The following narrative describes this use case, with Figure 12 presenting the corresponding SysML representation.

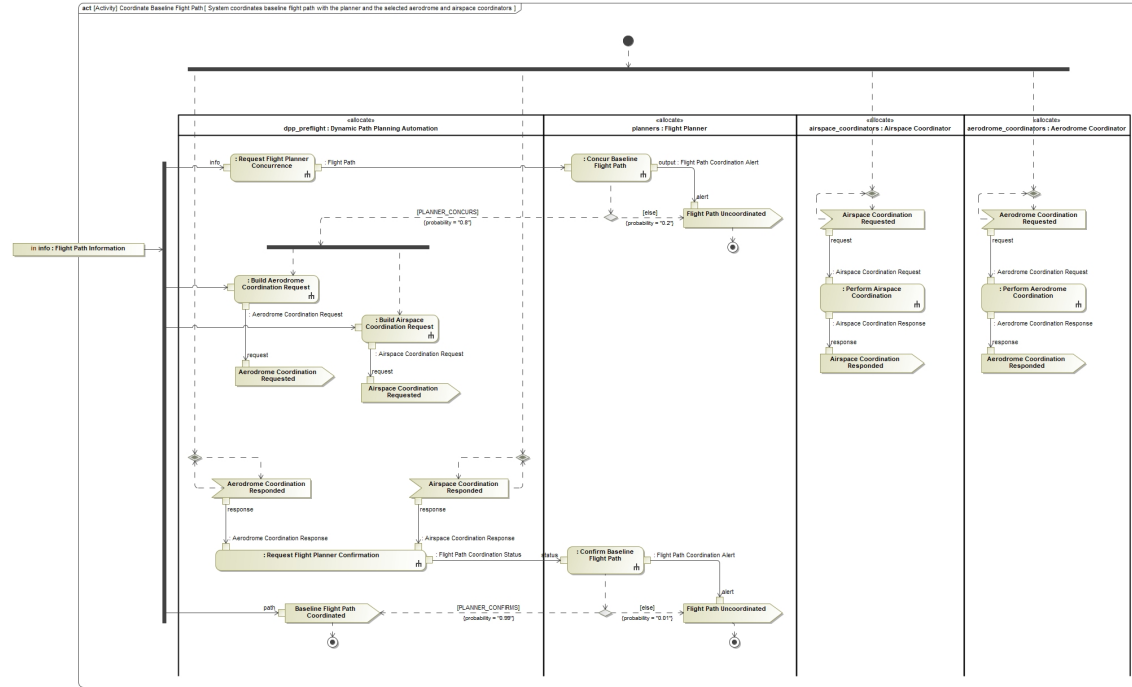


Figure 12: Flight planner coordinates a baseline flight path during preflight planning.

Use Case: Coordinate baseline flight path

Scope: Ground-based DPP automation system before the flight

Primary Actor: Flight planner

Secondary Actors: Aerodrome and airspace coordinators

Preconditions: The ground-based DPP automation system contains a flight path that is safe and acceptable to the flight planner.

Trigger: Flight planner concurs with the flight path built by the DPP automation system.

Nominal Scenario:

1. The flight planner instructs the DPP automation system to coordinate an acceptable flight path with the relevant participants in the operational domain.
2. The DPP automation system compiles an aerodrome coordination request based on the computed flight path.

3. The DPP automation system transmits the aerodrome coordination request to the selected aerodrome coordinator via a configured secure Ethernet connection.
4. The DPP automation system compiles an airspace coordination request based on the computed flight path.
5. The DPP automation system transmits the airspace coordination request to the selected airspace coordinator via a configured secure Ethernet connection.
6. The DPP automation system receives the aerodrome coordinator response via a configured secure Ethernet connection.
7. The DPP automation system receives the airspace coordinator response via a configured secure Ethernet connection.
8. The DPP automation system analyzes the responses from the aerodrome and airspace coordinators.
9. If the flight path coordination is successful, the DPP automation system displays the coordinated flight path on the flight planner graphical user interface.
10. The flight planner determines if the coordinated flight path is acceptable.
11. If the coordinated flight path is unacceptable, the flight planner may change the flight planning request. [➡ Step 2]

Alternate Scenarios/Extensions:

- 8a. If the computed flight path coordination fails:
 - 8a.1 The DPP automation system generates an “Unharmonized Flight Path” alert.
 - 8a.2 The DPP automation system displays the alert on the flight planner graphical user interface.
 - 8a.3 The flight planner changes the flight planning request by updating the flight planning parameters to comply with the alert. The flight planner may need to coordinate with the mission manager if a change in flight mission is warranted. [➡ Step 2]

Postconditions: A flight path which complies with the specified flight mission is available.

3.5.3 Synchronize Baseline Flight Path

The mission manager uses the DPP automation system to synchronize the baseline flight path between the ground-based and onboard DPP automation systems during preflight planning. The following narrative describes this use case, with Figure 13 presenting the corresponding SysML representation.

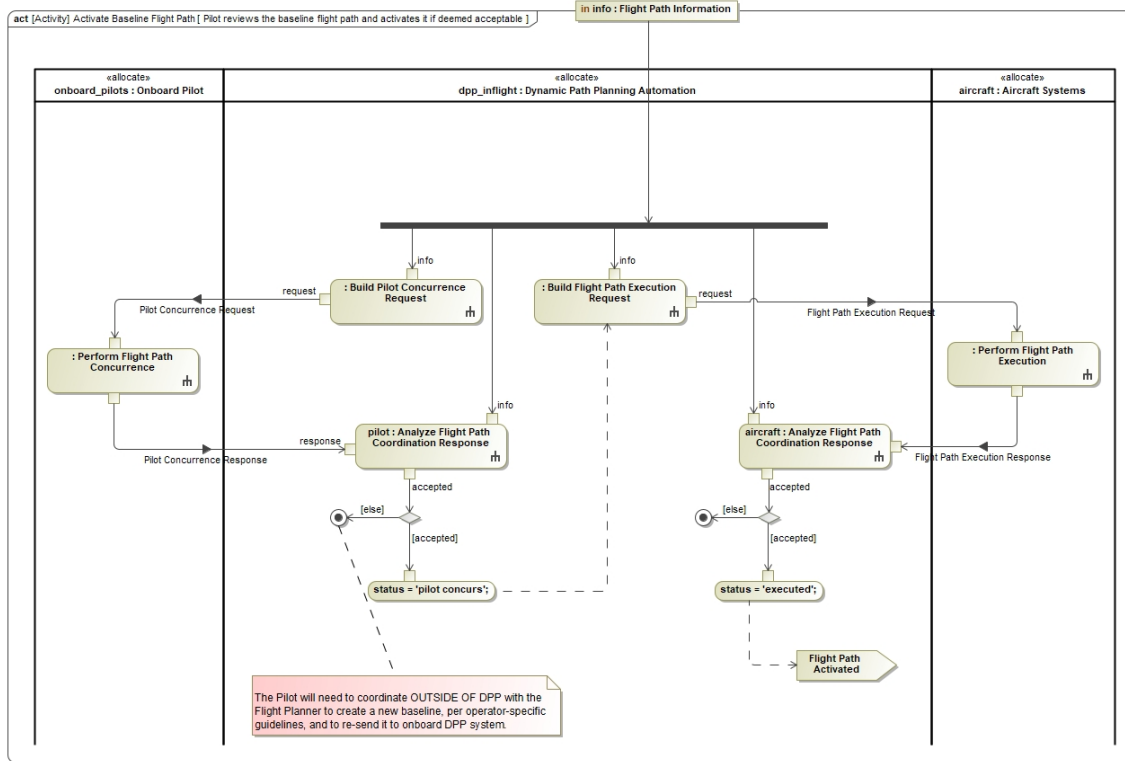


Figure 13: Mission manager synchronizes the path between ground and onboard systems.

Use Case: Synchronize baseline flight Path

Scope: Ground-based and onboard DPP automation systems before the flight

Primary Actors: Mission manager, Pilot

Secondary Actor: Aircraft

Preconditions:

- A flight path which executes the specified flight mission is available.
- The ground-to-aircraft and aircraft-to-ground data communication links are operational.

Trigger: The mission manager instructs the ground-based DPP automation system to transmit the planned flight path to the onboard DPP automation system.

Nominal Scenario:

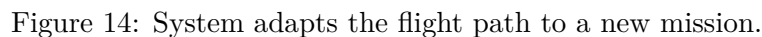
1. The ground-based DPP automation transmits the planned flight path to the on-board DPP automation system via ground-to-aircraft data communication link.
2. The onboard DPP automation acknowledges the receipt of the planned flight path via aircraft-to-ground data communication link.
3. The onboard DPP automation system presents the received flight path on the pilot graphical user interface.
4. The pilot reviews the flight path for completeness and acceptability.
5. If the flight path is acceptable, the DPP automation system compiles a flight path execution request.
6. The DPP automation system transmits the flight path execution request to the aircraft guidance system via the onboard aircraft network.
7. The DPP automation system receives the flight path execution response from the aircraft guidance system via the onboard aircraft network.
8. The DPP automation system analyzes the aircraft guidance system response.
9. The DPP automation system saves the successfully executed flight path as the current flight path.
10. The DPP automation system informs the mission manager of successful activation of the planned flight path.

Alternate Scenarios/Extensions:

- 4a. If the flight path is not acceptable to the pilot:
 - 4a.1 The pilot alerts the mission manager of unsuitable flight path.
 - 4a.2 The ground-based DPP automation system executes use case 3.5.1.
 - 4a.3 The onboard DPP automation system receives a new flight path from the ground-based DPP automation system. [➡ Step 2]
- 8a. If the aircraft is unable to execute the flight path:
 - 8a.1 The onboard DPP automation system generates an “Unexecutable Flight Path” alert.
 - 8a.2 The onboard DPP automation system displays the alert on the pilot graphical user interface.
 - 8a.3 The pilot alerts the mission manager of unexecutable flight path.
 - 8a.4 The ground-based DPP automation system executes use case 3.5.1.
 - 8a.5 The onboard DPP automation system receives a new flight path from the ground-based DPP automation system. [➡ Step 2]

Postconditions: A safe, efficient, and operationally acceptable flight path that executes the specified flight mission is available.

The pilot-in-command uses the DPP automation system to adapt the flight path in response to a change in flight mission. The following narrative describes this use case, with Figure 14 presenting the corresponding SysML representation.



Preconditions:

- ### Nominal Scenario:

- 36

3. If the mission change is likely to impact the flight path, the DPP automation system compiles a flight path update trigger.
4. The DPP automation system updates the flight path. [➡ 3.5.12]
5. If the flight path is updated successfully, the DPP automation system replaces the stored flight mission with the new mission.

Alternate Scenarios/Extensions:

- 1a. If the mission change request is invalid:
 - 1a.1 The DPP automation system generates an “Invalid Flight Mission Request” alert.
 - 1a.2 The DPP automation system displays the alert on the pilot graphical user interface.
 - 1a.3 The pilot may decide to revise the flight mission request. [➡ Step 1]
- 4a. If the flight path is not updated successfully:
 - 4a.1 The DPP automation system displays “Flight Path Update Unavailable: Mission Change Request” on the pilot graphical user interface.
 - 4a.2 The pilot may decide to revise the flight mission request. [➡ Step 1]

Postconditions: The aircraft is executing the new flight mission.

3.5.5 Adapt Flight Path to Shared Operating Environment

The DPP automation system adapts the flight path autonomously to changes in the operating environment shared by external data providers. The data providers may share changes to the aerodrome, airspace, atmosphere, obstacles, and traffic data. The following narrative describes this use case, with Figure 15 presenting the corresponding SysML representation.

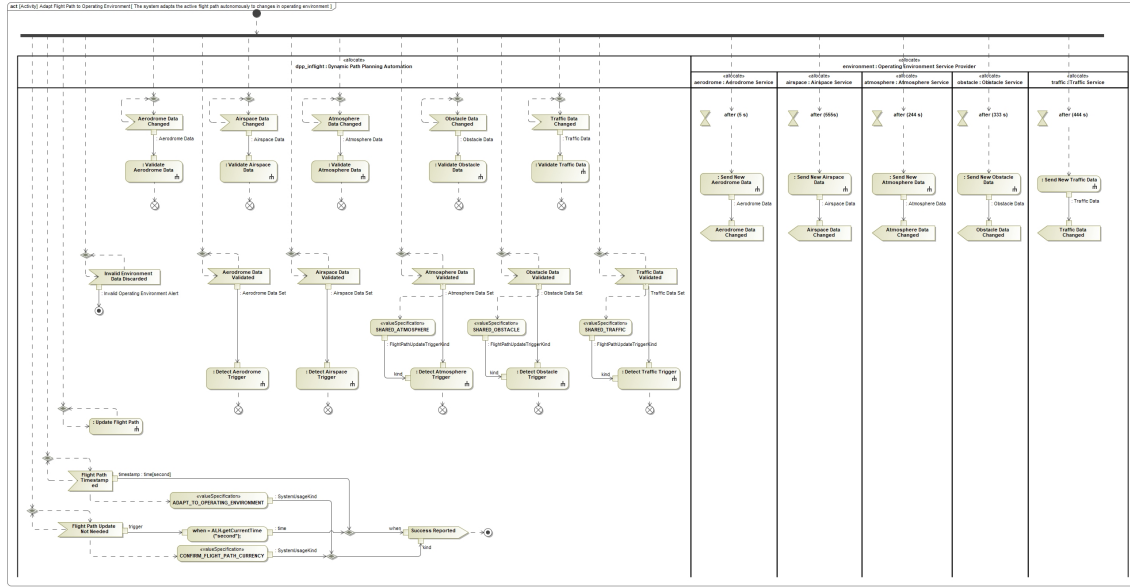


Figure 15: System adapts the flight path autonomously to operating environment.

Use Case: Adapt flight path to shared operating environment

Scope: Onboard DPP automation system during the flight

Primary Actor: Mission manager

Secondary Actors: Shared operating environment data providers

Preconditions:

- The DPP automation system is operating in nominal state.
- The mission manager has configured the DPP automation system to respond autonomously to changes in shared operating environment.

Trigger: The DPP automation system receives data from a shared operating environment data provider.

Nominal Scenario:

1. The DPP automation system determines the type of the operating environment data received: aerodrome, airspace, atmosphere, obstacle, or traffic.

2. The DPP automation system validates the received data for accuracy and completeness.
3. If the data is valid, the DPP automation system updates the stored operating environment data.
4. The DPP automation system determines the impact of the new data on the flight path.
5. If the data is likely to impact the flight path, the DPP automation system compiles a flight path update trigger.
6. The DPP automation system updates the flight path. [➡ 3.5.12]

Alternate Scenarios/Extensions:

- 2a. If the received operating environment data is invalid:
 - 2a.1 The DPP automation system informs the pilot-in-command (onboard or remote) of the discarded invalid data.
 - 2a.2 The DPP automation system discards the data.
- 6a. If the flight path is not updated successfully:
 - 6a.1 The system transitions to contingent state.
 - 6a.2 The DPP automation system generates a flight path contingency trigger. [➡ 19]

Postconditions: The flight path accounts for the shared operating environment change.

3.5.6 Adapt Flight Path to Sensed Information

The DPP automation system adapts the flight path autonomously to relevant information received from aircraft sensors. The sensed information includes the aircraft's position, velocity, and propulsion data, and may include hazardous atmosphere, obstacle, and traffic data. The following narrative describes this use case, with Figure 16 presenting the corresponding SysML representation.

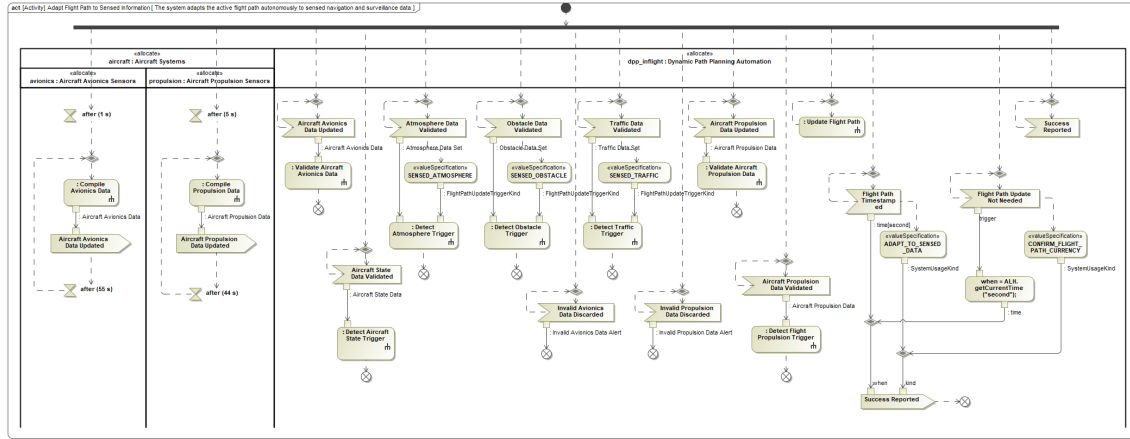


Figure 16: System adapts the flight path autonomously to aircraft sensed data.

Use Case: Adapt flight path to sensed information

Scope: Onboard DPP automation system during the flight

Primary Actor: Mission manager

Secondary Actor: Aircraft

Preconditions:

- The DPP automation system is operating in nominal state.
- The mission manager has configured the DPP automation system to respond autonomously to selected aircraft sensed data.

Trigger: The DPP automation system receives data from aircraft sensors.

Nominal Scenario:

1. The DPP automation system updates the aircraft state based on the sensed information.
2. The DPP automation system updates the aircraft propulsion data based on the sensed information.
3. The DPP automation system determines if the propulsion data (available fuel/energy and consumption rate) is in conformance with the energy profile.

4. The DPP automation system determines if the aircraft state (position and velocity) is in conformance with the flight path.
5. The DPP automation system determines if the anticipated contingency mitigation landing site is still valid.
6. The DPP automation system determines if the aircraft has reached a specified refresh event.
7. The DPP automation system generates an event-specific flight path update trigger.
8. The DPP automation system determines the type of sensed operating environment data: sensed atmosphere, sensed obstacle, or sensed traffic.
9. The DPP automation system validates the sensed operating environment data for accuracy and completeness.
10. If the data is valid, the DPP automation system updates the corresponding operating environment data.
11. The DPP automation system determines the impact of the sensed operating environment data on the flight path.
12. If the data is likely to impact the flight path, the DPP automation system compiles a flight path update trigger.
13. The DPP automation system updates the flight path. [➡ 3.5.12]

Alternate Scenarios/Extensions:

- 3a. If the propulsion data is out of conformance with the energy profile:
 - 3a.1 The system transitions to contingent state.
 - 3a.2 The DPP automation system generates a flight path contingency trigger. [➡ 19]
- 4a. If the aircraft state is out of conformance with the flight path:
 - 4a.1 The DPP automation system generates an out-of-conformance flight path update trigger. [➡ Step 13]
- 5a. If the anticipated contingency mitigation landing site is no longer viable
 - 5a.1 The DPP automation system generates a flight path update trigger with a revised mission. [➡ Step 13]
- 9a. If the received operating environment data is invalid:
 - 9a.1 The DPP automation system informs the pilot-in-command (onboard or remote) of the discarded invalid data.
 - 9a.2 The DPP automation system discards the data.
- 13a. If the flight path is not updated successfully:
 - 13a.1 The system transitions to contingent state.
 - 13a.2 The DPP automation system generates a flight path contingency trigger. [➡ 19]

Postconditions: The flight path accounts for the aircraft sensed data.

3.5.7 Adapt Flight Path to Pilot Observed Event

The DPP automation system adapts the flight path to pilot reported events. These include events which are not sensed by the aircraft—crew or passenger emergency for instance—and observed hazardous atmosphere, obstacle, or traffic aircraft. The following narrative describes this use case, with Figure 17 presenting the corresponding SysML representation.

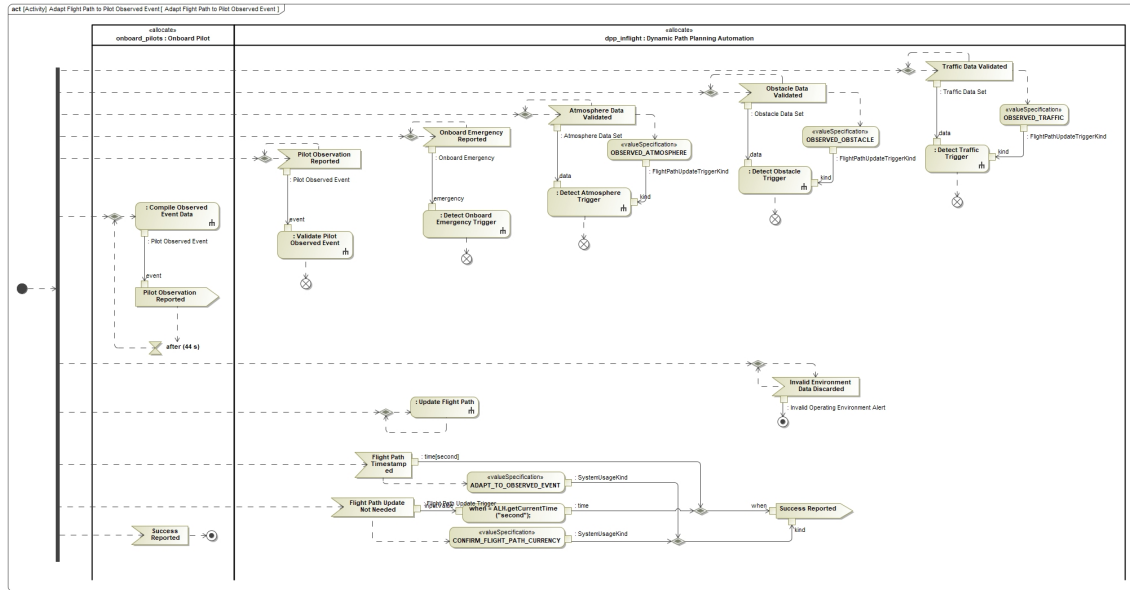


Figure 17: System adapts the flight path to observed onboard events.

Use Case: Adapt flight path to pilot observed event

Scope: Onboard DPP automation system during the flight

Primary Actor: Pilot

Secondary Actors: None

Preconditions:

- The DPP automation system is operating in nominal state.

Trigger: The pilot informs the DPP automation system of an observed event.

Nominal Scenario:

1. The DPP automation system determines the type of event: onboard emergency, observed atmosphere, observed obstacle, or observed traffic.
2. If the event is an onboard emergency, the DPP automation system determines its impact on the flight path.

3. If the event is likely to impact the flight path, the DPP automation system compiles a flight path update trigger. [➡ Step 9]
4. If the event is likely to result in a contingency, the DPP automation system compiles a flight path contingency trigger. [➡ 19]
5. The DPP automation system validates the observed operating environment data for accuracy and completeness.
6. If the data is valid, the DPP automation system updates the corresponding operating environment data.
7. The DPP automation system determines the impact of the observed operating environment data on the flight path.
8. If the data is likely to impact the flight path, the DPP automation system compiles a flight path update trigger.
9. The DPP automation system updates the flight path. [➡ 3.5.12]

Alternate Scenarios/Extensions:

- 5a. If the observed operating environment data is invalid:
 - 5a.1 The DPP automation system informs the pilot-in-command (onboard or remote) of the discarded invalid data.
 - 5a.2 The DPP automation system discards the data.
- 9a. If the flight path is not updated successfully:
 - 9a.1 The system transitions to contingent state.
 - 9a.2 The DPP automation system generates a flight path contingency trigger. [➡ 19]

Postconditions: The flight path accounts for the pilot reported changes.

3.5.8 Assess Proposed Flight Path Change

An authorized user may propose a change to the current flight path. The following narrative describes this use case, with Figure 18 presenting the corresponding SysML representation.

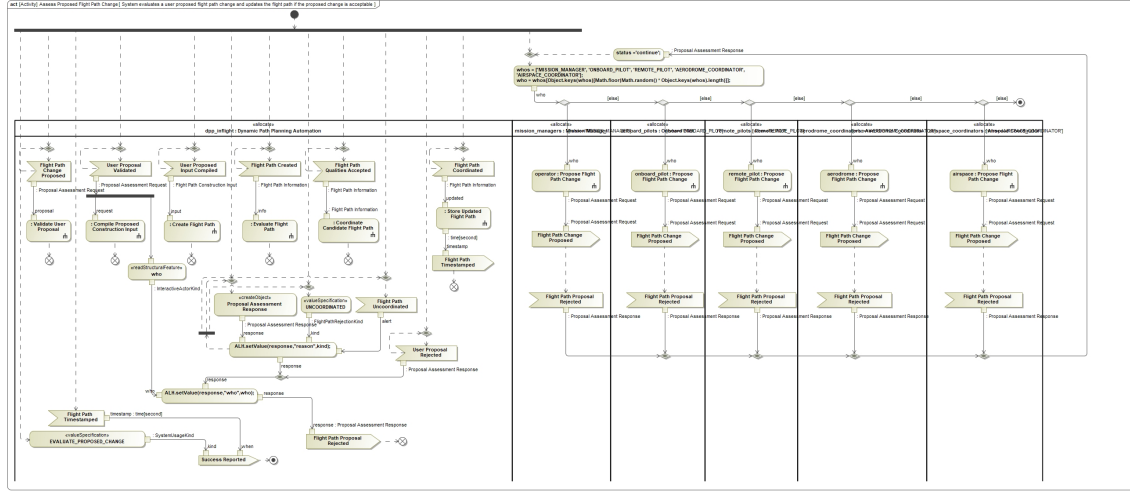


Figure 18: System evaluates a user-proposed flight path change.

Use Case: Assess proposed flight path change

Scope: Onboard DPP automation system during the flight

Primary Actors: Mission manager, Pilot, Aerodrome coordinator, Airspace coordinator

Secondary Actors: None

Preconditions:

- The DPP automation system is operating in nominal state.
- The user is authorized to propose a flight path change.

Trigger: The system receives a flight path change proposal from an authorized user.

Nominal Scenario:

1. The DPP automation system validates the proposed change request for accuracy and completeness.
2. If the proposed change request is valid, the DPP automation system determines the impact of the new mission on the flight path.
3. If the mission change is likely to impact the flight path, the DPP automation system compiles a flight path update trigger.
4. The DPP automation system updates the flight path. [➡ 3.5.12]

5. If the flight path is updated successfully, the DPP automation system replaces the stored flight mission with the new mission.

Alternate Scenarios/Extensions:

- 1a. If the proposed change request is invalid:
 - 1a.1 The DPP automation system generates an “Invalid Proposal” alert.
 - 1a.2 The DPP automation system transmits the alert to the requestor.
- 4a. If the flight path is not updated successfully:
 - 4a.1 The DPP automation system generates a “Proposal Rejected” alert.
 - 4a.2 The DPP automation system transmits the alert to the requestor.

Postconditions: The system either accepts the proposed flight path change or informs the user of the reason behind the rejection.

3.5.9 Mitigate Flight Path Contingency

The DPP automation system must be able to mitigate an identified flight path contingency autonomously. Figure 19 presents a SysML representation of this use case⁶.

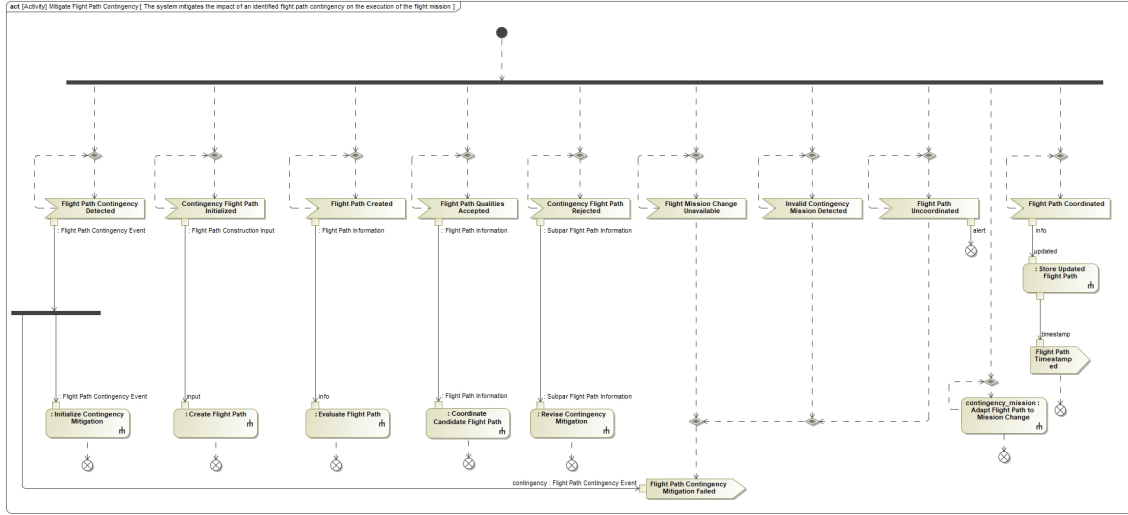


Figure 19: System mitigates the impact of contingencies on the flight path.

⁶Details of the use case are pending as the contingency management concept of the DPP automation system evolves.

3.5.10 Comply with Tactical Maneuver Command

A safety-critical system on the aircraft, as well as the pilot-in-command, may command the DPP automation system to suspend dynamic flight path planning. The following narrative describes a preliminary version of this use case⁷.

Use Case: Comply with tactical maneuver command

Scope: Onboard DPP automation system during the flight

Primary Actors: Aircraft, Pilot

Secondary Actors: None

Preconditions:

- The DPP automation system is operating in nominal state.

Trigger: Either the pilot or an aircraft commands a tactical maneuver.

Nominal Scenario:

1. The DPP automation system inhibits the display of in-progress computation results.
2. The DPP automation system continues to monitor the shared and sensed operating environment.
3. The DPP automation system stores all requests to change the flight mission.
4. The DPP automation system rejects all proposals to change the flight path.

Alternate Scenarios/Extensions:

Postconditions: The DPP automation system transitions to passive state.

⁷Details of the use case are pending as the concept of the DPP automation system integration with the safety-critical aircraft systems evolves.

3.5.11 Refresh Flight Path Periodically

The DPP automation system refreshes the flight path periodically. The following narrative describes this use case.

Use Case: Refresh flight path periodically

Scope: Onboard DPP automation system during the flight

Primary Actor: Mission manager

Secondary Actors: None

Preconditions:

- The DPP automation system is operating in nominal state.
- The mission manager has configured the DPP automation system to refresh the flight path periodically.

Trigger: A built-in timer informs the DPP automation system of the time to refresh the flight path.

Nominal Scenario:

1. The DPP automation system compiles a flight path update trigger with the latest operating environment and aircraft state information.
2. The DPP automation system updates the flight path. [➡ 3.5.12]

Alternate Scenarios/Extensions:

- 2a. If the flight path is not updated successfully:
 - 2a.1 The DPP automation system attempts refreshing the flight path again. [➡ Step 1]

Postconditions: The flight path accounts for the latest operating environment.

3.5.12 Update Flight Path

The DPP automation system updates the flight path in response to each flight path update trigger identified in other use cases. The following narrative describes this use case, with Figure 20 presenting the corresponding SysML representation.

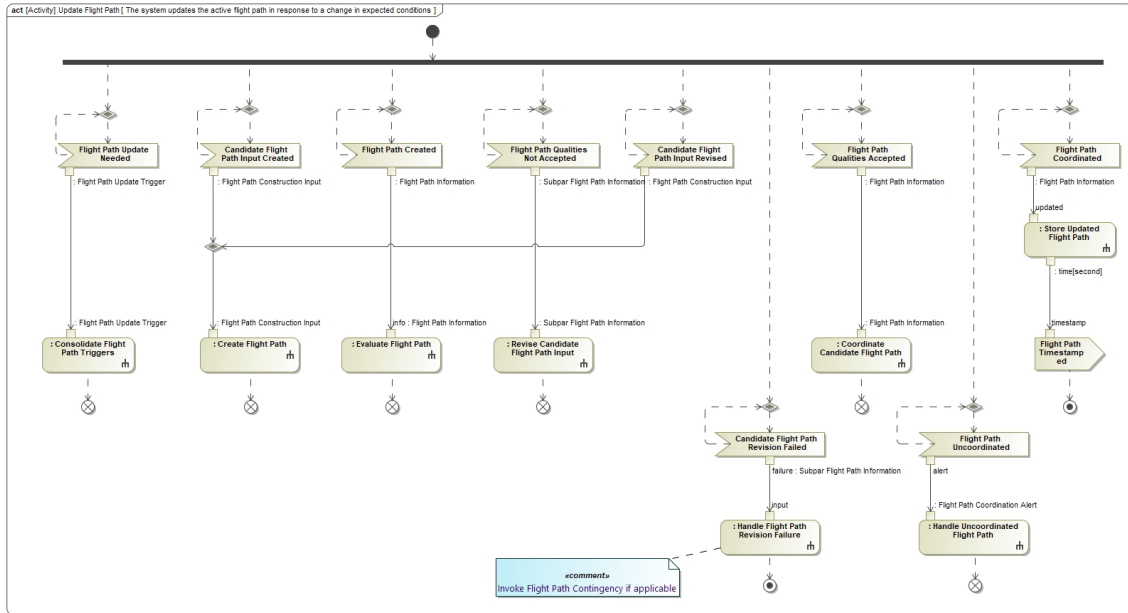


Figure 20: System updates the flight path.

Use Case: Update flight path

Scope: Onboard DPP automation system during the flight

Primary Actors: None

Secondary Actors: Shared operating environment data providers

Preconditions:

- The DPP automation system is operating in nominal state.

Trigger: A need to update the flight path is identified.

Nominal Scenario:

1. The DPP automation system consolidates flight path update triggers.
2. The DPP automation system retrieves the input parameters which were used to construct the current flight path.
3. The DPP automation system updates the input parameters based on the consolidated trigger.

4. The DPP automation system validates the flight path construction input for completeness.
5. If the flight path construction input is valid, the DPP automation system constructs a candidate flight path.
6. The DPP automation system evaluates the feasibility of the candidate flight path.
7. If the candidate flight path is feasible, the DPP automation system determines if the flight path conflicts with any of the known operating environment hazards.
8. If the candidate flight path is feasible and safe, i.e., not in conflict with any known hazard, the DPP automation system evaluates the flight path cost.
9. The DPP automation system coordinates the feasible, safe, and optimal candidate flight path with configured entities. [➡ 3.5.13: Step 1]
10. The DPP automation system stores the updated flight path, and the input used to generate it, as the new baseline data for potential future updates.

Alternate Scenarios/Extensions:

- 4a. If the flight path construction input is invalid:
 - 4a.1 If the flight path update was triggered by a user-proposed flight path change, the DPP automation system rejects the user proposal due to “Invalid Flight Path Construction Input”. [➡ 3.5.8: Step 4a.]
 - 4a.2 Otherwise, the DPP automation system applies the relevant flight path revision rule to update the flight path construction input. [➡ Step 4]
- 6a. If the candidate flight path is infeasible:
 - 6a.1 If the flight path update was triggered by a user-proposed flight path change, the DPP automation system rejects the user proposal due to “Infeasible Flight Path”. [➡ 3.5.8: Step 4a.]
 - 6a.2 Otherwise, the DPP automation system applies the relevant flight path revision rule to update the flight path construction input. [➡ Step 4]
- 7a. If the candidate flight path is in conflict with at least one known hazard:
 - 7a.1 If the flight path update was triggered by a user-proposed flight path change, the DPP automation system rejects the user proposal due to “Unsafe Flight Path”. [➡ 3.5.8: Step 4a.]
 - 7a.2 Otherwise, the DPP automation system applies the relevant flight path revision rule to update the flight path construction input. [➡ Step 4]
- 8a. If the candidate flight path cost is greater than that of the current flight path:
 - 8a.1 If the flight path update was triggered by a user-proposed flight path change, the DPP automation system rejects the user proposal due to “Suboptimal Flight Path”. [➡ 3.5.8: Step 4a.]
 - 8a.2 Otherwise, the DPP automation system applies the relevant flight path revision rule to update the flight path construction input. [➡ Step 4]

Postconditions: The flight path accounts for the triggered change.

3.5.13 Coordinate Candidate Flight Path

The DPP automation system coordinates each candidate flight path with the selected entities—per AAM operator concept—to ensure that the flight path is harmonized, executable, and acceptable. Every flight path must be harmonized with respect to the relevant airspace and aerodrome participants, executable by the aircraft, and acceptable to the pilot. The following narrative describes this use case, with Figure 21 presenting the corresponding SysML representation.

Use Case: Coordinate candidate flight path

Scope: Onboard DPP automation system during the flight

Primary Actors: None

Secondary Actors: Pilot, Aerodrome coordinator, Airspace coordinator, Aircraft

Preconditions:

- The communication link between the DPP automation system and the aerodrome coordinator is operational.
- The communication link between the DPP automation system and the airspace coordinator is operational.
- The communication link between the DPP automation system and the aircraft guidance system is operational.
- The pilot graphical user interface is operational.

Trigger: A candidate flight path is constructed in response to a flight path update trigger.

Nominal Scenario:

1. The DPP automation system displays the feasible, safe, and optimal candidate flight path on the pilot graphical user interface.
2. The pilot determines if the candidate flight path is acceptable to harmonize with other participants in the operating domain.
3. If the pilot concurs:
 - (a) The DPP automation system compiles an aerodrome coordination request based on the computed flight path.
 - (b) The DPP automation system compiles an airspace coordination request based on the computed flight path.
 - (c) The DPP automation system transmits the aerodrome coordination request to the selected aerodrome coordinator via air-to-ground coordination connection.
 - (d) The DPP automation system transmits the airspace coordination request to the selected airspace coordinator via air-to-ground coordination connection.
4. The DPP automation system receives the aerodrome coordinator response via ground-to-air coordination connection.

5. The DPP automation system receives the airspace coordinator response via ground-to-air coordination connection.
6. The DPP automation system analyzes the responses from the aerodrome coordinator.
7. The DPP automation system analyzes the responses from the airspace coordinator.
8. If the aerodrome and airspace flight path coordination are successful, the DPP automation system displays the coordinated flight path on the pilot graphical user interface.
9. The pilot confirms acceptability of the coordinated flight path.
10. Upon the pilot's acceptance, the DPP automation system compiles a flight path execution request based on the coordinated flight path.
11. The DPP automation system transmits the flight path execution request to the aircraft guidance system via the onboard aircraft network.
12. The DPP automation system receives the flight path execution response from the aircraft guidance system via the onboard aircraft network.
13. The DPP automation system analyzes the aircraft guidance system response.
14. If the flight path is executed successfully, the DPP automation system compiles a flight path transmission request, based on the executed flight path, for the purpose of sharing operational intent with other operators.
15. The DPP automation system transmits the flight path transmission request to the aircraft communication system via the onboard aircraft network.
16. The DPP automation system receives the flight path transmission response from the aircraft communication system via the onboard aircraft network.
17. The DPP automation system analyzes the aircraft communication system response.
18. If the flight path is transmitted successfully, the DPP automation system updates the flight path status on the pilot graphical user interface.

Alternate Scenarios/Extensions:

- 2a. If the pilot does not accept the candidate flight path:
 - 2a.1 The DPP automation system generates an "Uncoordinated Flight Path: Pilot Concurrence" alert.
 - 2a.2 The DPP automation system creates a flight path update trigger due to uncoordinated candidate flight path. [➡ 3.5.12]
- 6a. If the aerodrome coordinator rejects the candidate flight path:
 - 6a.1 The DPP automation system generates an "Uncoordinated Flight Path: Aerodrome Coordination" alert.
 - 6a.2 The DPP automation system creates a flight path update trigger due to uncoordinated candidate flight path. [➡ 3.5.12]
- 7a. If the airspace coordinator rejects the candidate flight path:

- 7a.1 The DPP automation system generates an “Uncoordinated Flight Path: Airspace Coordination” alert.
 - 7a.2 The DPP automation system creates a flight path update trigger due to uncoordinated candidate flight path. [➡ 3.5.12]
 - 9a. If the pilot does not accept the coordinated flight path:
 - 9a.1 The DPP automation system generates an “Uncoordinated Flight Path: Pilot Confirmation” alert.
 - 9a.2 The DPP automation system creates a flight path update trigger due to uncoordinated candidate flight path. [➡ 3.5.12]
 - 13a. If the aircraft is unable to execute the new flight path:
 - 13a.1 The DPP automation system generates an “Uncoordinated Flight Path: Flight Path Execution” alert.
 - 13a.2 The DPP automation system creates a flight path update trigger due to uncoordinated candidate flight path. [➡ 3.5.12]
 - 17a. If the aircraft fails to transmit the executed flight path:
 - 17a.1 The DPP automation system generates an “Uncoordinated Flight Path: Flight Path Transmission” alert.
 - 17a.2 The DPP automation system tags the flight as “noncooperative” on the pilot graphical display.
- Postconditions:** The flight path is acceptable, executable, and harmonious with users in the selected aerodrome and airspace.
-

3.6 Measures of Effectiveness

The nonfunctional goals of the DPP automation system are refined by measures of effectiveness (MOEs). The MOEs are the standards against which the capability of a system to achieve its intended purpose may be judged. The standards are specific properties that any potential solution must exhibit to some extent [11]. The MOEs are independent of any solution and do not specify the system performance criteria. Identifying correct MOEs requires frequent engagements with stakeholders, primarily because the MOEs include not only the quantitative metrics but also the qualitative subjective measures.

An initial development of the quantitative⁸ MOEs for the DPP automation system is conducted in preparation for stakeholder reviews, which have not occurred yet. Recall that the stakeholder concerns are consolidated into four types of system goals: functional, performance, interface, and design (§3.3). Of these, the interface goals do not translate directly to MOEs. The MOEs corresponding to the design goals will benefit from the design standards for the DPP automation system, which have yet to be established. Therefore, this version of the architecture focuses specifically on the performance goals and the performance aspects of the functional goals. These MOEs are derived by analyzing the system availability, system integrity, and pilot workload in the context of the system’s ability to achieve its functional goals. An initial set of MOEs, for each use case corresponding to the functional goals, includes:

- Probability of the system to execute the use case successfully.
- Probability of the system to provide a successful result within a specified time.
- Probability of the system to provide a timely alert of failure to execute the use case.
- Probability of the system to fail to alert (different from late alerting).
- Probability of the system to alert falsely.
- Probability of the system to produce nuisance alerts.

Further research and coordination with stakeholders is needed in order to establish a comprehensive set of MOEs for the DPP automation system.

3.7 Top-Level System Requirements

This section derives the top-level system requirements from the use cases, system context, and system measures of effectiveness. Specifically:

- Table 1 lists the functional requirements derived from the use cases and MOEs.
- Table 2 lists the usability requirements derived from the measures of effectiveness.
- Table 3 lists the physical requirements derived from stakeholder concerns.
- Table 4 lists the design constraints derived from stakeholder concerns.
- Table 5 lists the interface requirements derived from the system context.

⁸Qualitative aspects of the stakeholder concerns, related to the pilot workload for instance, will be addressed in a future version.

Table 1: Top-Level Functional Requirements

Name	Text
Sys-F-1: Build Initial Flight Path	The Dynamic Path Planning Automation System shall build a flight path that meets the mission objectives and operator preferences, subject to the aircraft performance and operating environment constraints, upon receipt of a flight planning request from the flight planner via the flight planner graphical user interface, within a specified timeframe, upon receiving the request.
Sys-F-2: Coordinate Planned Flight Path	The Dynamic Path Planning Automation System shall coordinate the planned flight path with the selected flight path coordinators upon successful creation of a valid baseline flight path.
Sys-F-3: Synchronize Planned Flight Path	The Dynamic Path Planning Automation System shall initialize the onboard baseline flight path with the planned flight path upon receipt of a flight path synchronization command from the mission manager via the operator communication link.
Sys-F-4: Activate Planned Flight Path	The Dynamic Path Planning Automation System shall activate the baseline flight path upon receipt of a flight path activation command from the pilot via the pilot graphical user interface.
Sys-F-5: Adapt Active Flight Path to Observed Anomaly	The Dynamic Path Planning Automation System shall adapt the active flight path to pilot observed anomalies received via pilot graphical user interface, within a specified timeframe, upon receiving the pilot instruction.
Sys-F-6: Adapt Active Flight Path to Operating Environment	The Dynamic Path Planning Automation System shall adapt the active flight path to changes in the operating environment data received via ground-to-air service communication link, within a specified timeframe, upon receiving the data.
Sys-F-7: Execute Revised Mission	The Dynamic Path Planning Automation System shall update the active flight path to comply with a command to revise the flight mission, within a specified timeframe, upon receiving the revised mission.
Sys-F-8: Comply with Tactical Override Command	The Dynamic Path Planning Automation System shall comply with all tactical commands to pause and resume dynamic flight path planning, within a specified timeframe, upon receiving the command.
Sys-F-9: Coordinate Candidate Flight Path	The Dynamic Path Planning Automation System shall coordinate every successfully built candidate flight path with the selected entities.
Sys-F-10: Evaluate Flight Path Proposal	The Dynamic Path Planning Automation System shall evaluate the viability of all user proposed flight path changes within a specified timeframe upon receiving the proposal.
Sys-F-11: Mitigate Flight Path Contingency	The Dynamic Path Planning Automation System shall mitigate the impact of all identified contingencies on the execution of the flight mission, within a specified timeframe of identifying the contingency.
Sys-F-12: Monitor Flight Path Conformance	The Dynamic Path Planning Automation System shall update the active flight path when the aircraft is out of conformance with the planned flight path, within a specified timeframe after identifying the change in conformance.
Sys-F-13: Monitor Flight Events	The Dynamic Path Planning Automation System shall update the active flight path at a set of predefined flight events during the flight, within a specified timeframe, upon receiving the event notification.
Continued on next page	

Table 1 – continued from previous page

Name	Text
Sys-F-14: Refresh Active Flight Path	The Dynamic Path Planning Automation System shall refresh the active flight path periodically in search of a more optimal and/or flexible path than the current path at a predefined duration since the active flight path creation.
Sys-F-15: Report Flight Path Status	The Dynamic Path Planning Automation System shall report the flight path status to the authorized users at a specified periodic rate.
Sys-F-16: Report System Health Status	The Dynamic Path Planning Automation System shall report the automation system health status to the authorized users at a specified periodic rate.
Sys-F-17: Update Active Flight Path	The Dynamic Path Planning Automation System shall compute a viable candidate flight path each time it recognizes a need to update the flight path, within a specified timeframe, upon receiving the update trigger.
Sys-F-18: Update System Configuration	The Dynamic Path Planning Automation System shall update the flight path in response to relevant changes to the system configuration, within a specified timeframe, once a configuration change has been identified.

Table 2: Top-Level Usability Requirements

Name	Text
Sys-U-1: Nuisance Alerts	The Dynamic Path Planning Automation System shall minimize providing repetitious alerts.
Sys-U-2: False Alerts	The Dynamic Path Planning Automation System shall minimize providing false alerts.
Sys-U-3: Missed Alerts	The Dynamic Path Planning Automation System shall minimize failures to provide necessary alerts.
Sys-U-4: Timely Response	The Dynamic Path Planning Automation System shall respond to user input within a specified timeframe.
Sys-U-5: System Availability	The Dynamic Path Planning Automation System shall be available to all authorized users through the life of the flight in at least a specified percent of instances.
Sys-U-6: Input Customization	The Dynamic Path Planning Automation System shall allow authorized users to customize relevant input configuration parameters.
Sys-U-7: Output Customization	The Dynamic Path Planning Automation System shall allow authorized users to customize relevant output configuration parameters.

Table 3: Top-Level Physical Requirements

Name	Text
Sys-Ph-1: Onboard Installation	The Dynamic Path Planning Automation System shall support onboard installation on operator-selected types of aircraft.
Continued on next page	

Table 3 – continued from previous page

Name	Text
Sys-Ph-2: Operator Installation	The Dynamic Path Planning Automation System shall support installation at an authorized flight operations center.
Sys-Ph-3: Aerodrome Coordinator Installation	The Dynamic Path Planning Automation System shall support installation at an authorized aerodrome flight coordination center.
Sys-Ph-4: Airspace Coordinator Installation	The Dynamic Path Planning Automation System shall support installation at an authorized airspace flight coordination center.

Table 4: Top-Level Design Constraints

Name	Text
Sys-D-1: Regulatory Compliance	The Dynamic Path Planning Automation System must comply with all applicable regulatory standards.
Sys-D-2: ANSP Compliance	The Dynamic Path Planning Automation System must comply with the relevant Air Navigation Service Provider guidelines and constraints.
Sys-D-3: Access to System	The Dynamic Path Planning Automation System must be accessible only by authorized users with valid credentials.
Sys-D-4: System Access	The Dynamic Path Planning Automation System must be authorized to access all necessary external interfaces.

Table 5: Top-Level Interface Requirements

Name	Text
Sys-I-1: From Flight Planner	The Dynamic Path Planning Automation System shall receive data from the flight planner via a ground-based flight planner graphical user interface.
Sys-I-2: To Flight Planner	The Dynamic Path Planning Automation System shall provide data to the flight planner via a ground-based flight planner graphical user interface.
Sys-I-3: From Onboard Pilot	The Dynamic Path Planning Automation System shall receive data from the onboard pilot via an onboard pilot graphical user interface.
Sys-I-4: To Onboard Pilot	The Dynamic Path Planning Automation System shall provide data to the onboard pilot via an onboard pilot graphical user interface.
Sys-I-5: From Mission Manager	The Dynamic Path Planning Automation System shall receive data from the mission manager via a ground-to-air operator communication interface.
Sys-I-6: To Mission Manager	The Dynamic Path Planning Automation System shall provide data to the mission manager via an air-to-ground operator communication interface.
Sys-I-7: From Aircraft Avionics	The Dynamic Path Planning Automation System shall receive data from the aircraft avionics system via the aircraft avionics bus interface.
Sys-I-8: From Aircraft Propulsion	The Dynamic Path Planning Automation System shall receive data from the aircraft propulsion system via the aircraft avionics bus interface.
Continued on next page	

Table 5 – continued from previous page

Name	Text
Sys-I-9: From Aircraft Tactical System	The Dynamic Path Planning Automation System shall receive data from the aircraft tactical maneuver management system via the aircraft avionics bus interface.
Sys-I-10: From Aircraft Guidance System	The Dynamic Path Planning Automation System shall receive data from the aircraft guidance system via the aircraft avionics bus interface.
Sys-I-11: To Aircraft Guidance System	The Dynamic Path Planning Automation System shall provide data to the aircraft guidance system via the aircraft avionics bus interface.
Sys-I-12: From Aerodrome Coordinator	The Dynamic Path Planning Automation System shall receive data from a selected aerodrome coordinator via a ground-to-air coordinator communication interface.
Sys-I-13: To Aerodrome Coordinator	The Dynamic Path Planning Automation System shall provide data to a selected aerodrome coordinator via an air-to-ground coordinator communication interface.
Sys-I-14: From Airspace Coordinator	The Dynamic Path Planning Automation System shall receive data from a selected airspace coordinator via a ground-to-air coordinator communication interface.
Sys-I-15: To Airspace Coordinator	The Dynamic Path Planning Automation System shall provide data to a selected airspace coordinator via an air-to-ground coordinator communication interface.
Sys-I-16: From Aerodrome Data Provider	The Dynamic Path Planning Automation System shall receive data from a selected aerodrome data service provider via a ground-to-air data communication interface.
Sys-I-17: To Aerodrome Data Provider	The Dynamic Path Planning Automation System shall provide data to a selected aerodrome data service provider via an air-to-ground data communication interface.
Sys-I-18: From Airspace Data Provider	The Dynamic Path Planning Automation System shall receive data from a selected airspace data service provider via a ground-to-air data communication interface.
Sys-I-19: To Airspace Data Provider	The Dynamic Path Planning Automation System shall provide data to a selected airspace data service provider via an air-to-ground data communication interface.
Sys-I-20: From Atmosphere Data Provider	The Dynamic Path Planning Automation System shall receive data from a selected atmosphere data service provider via a ground-to-air data communication interface.
Sys-I-21: To Atmosphere Data Provider	The Dynamic Path Planning Automation System shall provide data to a selected atmosphere data service provider via an air-to-ground data communication interface.
Sys-I-22: From Obstacle Data Provider	The Dynamic Path Planning Automation System shall receive data from a selected obstacle data service provider via a ground-to-air data communication interface.
Sys-I-23: To Obstacle Data Provider	The Dynamic Path Planning Automation System shall provide data to a selected obstacle data service provider via an air-to-ground data communication interface.
Continued on next page	

Table 5 – continued from previous page

Name	Text
Sys-I-24: From Traffic Data Provider	The Dynamic Path Planning Automation System shall receive data from a selected traffic data service provider via a ground-to-air data communication interface.
Sys-I-25: To Traffic Data Provider	The Dynamic Path Planning Automation System shall provide data to a selected traffic data service provider via an air-to-ground data communication interface.
Sys-I-26: From System Administrator	The Dynamic Path Planning Automation System shall receive data from the automation system administrator via a ground-to-air data communication interface.
Sys-I-27: To System Administrator	The Dynamic Path Planning Automation System shall provide data to the automation system administrator via an air-to-ground data communication interface.
Sys-I-28: Preflight From Airspace Coordinator	The Dynamic Path Planning Automation System shall receive data from a selected airspace coordinator via Ethernet interface.
Sys-I-29: Preflight To Airspace Coordinator	The Dynamic Path Planning Automation System shall provide data to a selected airspace coordinator via Ethernet interface.
Sys-I-30: Preflight From Aerodrome Coordinator	The Dynamic Path Planning Automation System shall receive data from a selected aerodrome coordinator via Ethernet interface.
Sys-I-31: Preflight To Aerodrome Coordinator	The Dynamic Path Planning Automation System shall provide data to a selected aerodrome coordinator via Ethernet interface.
Sys-I-32: Preflight From Aerodrome Data Provider	The Dynamic Path Planning Automation System shall receive data from a selected aerodrome data service provider via Ethernet interface.
Sys-I-33: Preflight To Aerodrome Data Provider	The Dynamic Path Planning Automation System shall provide data to a selected aerodrome data service provider via Ethernet interface.
Sys-I-34: Preflight From Airspace Data Provider	The Dynamic Path Planning Automation System shall receive data from a selected airspace data service provider via Ethernet interface.
Sys-I-35: Preflight To Airspace Data Provider	The Dynamic Path Planning Automation System shall provide data to a selected airspace data service provider via Ethernet interface.
Sys-I-36: Preflight From Atmosphere Data Provider	The Dynamic Path Planning Automation System shall receive data from a selected atmosphere data service provider via Ethernet interface.
Sys-I-37: Preflight To Atmosphere Data Provider	The Dynamic Path Planning Automation System shall provide data to a selected atmosphere data service provider via Ethernet interface.
Sys-I-38: Preflight From Obstacle Data Provider	The Dynamic Path Planning Automation System shall receive data from a selected obstacle data service provider via Ethernet interface.
Sys-I-39: Preflight To Obstacle Data Provider	The Dynamic Path Planning Automation System shall provide data to a selected obstacle data service provider via Ethernet interface.
Sys-I-40: Preflight From Traffic Data Provider	The Dynamic Path Planning Automation System shall receive data from a selected traffic data service provider via Ethernet interface.
Sys-I-41: Preflight To Traffic Data Provider	The Dynamic Path Planning Automation System shall provide data to a selected traffic data service provider via Ethernet interface.

4 Functional System Architecture

This section describes the functional architecture of the DPP automation system in an AAM operating environment. The purpose of the functional architecture is to specify the system requirements at a sufficient level of detail to be able to build a logical system architecture.



For a relatively simple system (or during the first pass for a complex system) it is acceptable to skip the functional architecture step entirely, provided the top-level system requirements defined previously in the conceptual architecture are sufficient to develop the logical architecture of the system.

4.1 Synopsis

As shown in Figure 1, the steps in developing the functional architecture are :

1. Perform behavioral decomposition of the use cases into detailed activities.
2. Perform structural decomposition of the system into a set of interrelated functions.
3. Define interfaces and data communication among the system functions.
4. Elaborate system requirements by analyzing the system activities and interfaces.

Section 4.2 describes, in detail, the behavioral decomposition of the DPP automation system use cases, defined in the conceptual architecture. Figures 22–32 depict these decompositions, which result in the following list of activities that the system must perform in order to execute the underlying use cases:

- Analyze flight path coordination response
- Build aerodrome coordination request
- Build airspace coordination request
- Build flight path execution request
- Build flight path transmission request
- Build pilot concurrence request
- Build pilot confirmation request
- Build system health report
- Change system operating mode
- Compile baseline flight planning input
- Compile proposed construction input
- Comply with tactical command
- Consolidate flight path triggers
- Create flight path
- Detect aerodrome trigger
- Detect aircraft state trigger
- Detect airspace trigger
- Detect atmosphere trigger
- Detect configuration change trigger
- Detect flight path refresh trigger
- Detect flight propulsion trigger
- Detect mission change trigger
- Detect obstacle trigger
- Detect onboard emergency trigger
- Detect traffic trigger
- Evaluate flight path
- Generate flight path status report
- Handle flight path revision failure

- Handle uncoordinated flight path
- Initialize flight path construction input
- Initialize contingency mitigation
- Manage system alerts
- Monitor system operation
- Provide system configuration parameters
- Request flight planner concurrence
- Request flight planner confirmation
- Revise candidate flight path input
- Revise contingency mitigation
- Store updated flight path
- Validate aerodrome data
- Validate aircraft avionics data
- Validate aircraft propulsion data
- Validate airspace data
- Validate atmosphere data
- Validate flight mission
- Validate flight plan request
- Validate obstacle data
- Validate operating environment data
- Validate pilot observed event
- Validate traffic data
- Validate user proposal

Section 4.3 describes the structural decomposition of the DPP automation system into a set of interconnected functional units (referred to as functions), each of which is defined by the actions it performs. Figure 33 shows that the structure of the DPP automation system is made up of the following functions:

1. Data Validation function validates incoming data into the DPP automation system.
2. Initialization function builds the initial flight path before the flight.
3. Flight Path Monitoring function monitors the flight path during the flight.
4. Flight Path Creation function creates a feasible flight path.
5. Flight Path Evaluation function evaluates the flight path safety and optimality.
6. Flight Path Revision function updates the flight path in response to changes.
7. Flight Path Coordination function coordinates the flight path with selected entities.
8. Flight Contingency Mitigation function mitigates identified flight path contingencies.
9. Output Data Processing function processes outgoing data per interface requirements.
10. System Operation function manages the system operation, health, and alerts.

Section 4.4 defines the DPP automation system interfaces: the system black-box Interface Control Document (ICD) is a formalization of the connections defined earlier in the system context diagram (Figures 7–8), while the system white-box ICD defines a set of interfaces necessary to communicate data between the system functions in order to perform the required actions. Figure 34 lists the external interfaces between the DPP automation system and the AAM domain elements. Figure 35 defines the internal interfaces between the DPP automation system functions. Figure 36 depicts the flow of data across the DPP automation system boundary via the external interfaces, and the flow of data between the DPP automation system functions via the internal interfaces.

Section 4.5 elaborates on the top-level functional requirements of the DPP automation system by taking advantage of the information revealed during the behavioral decomposition of the system.

4.2 Behavioral Decomposition

The activities that constitute the use cases identified in the DPP automation system conceptual architecture are decomposed further in order to derive unambiguous and verifiable system functional requirements. A decomposition of the key use case behaviors is described below.

4.2.1 Build Baseline Flight Path

The system behavior of building a baseline flight path, as depicted in the “Dynamic Path Planning Automation” swimlane of Figure 11, constitutes the following activities:

- Validate flight path request
- Compile baseline flight planning input data
- Validate operating environment data
- Initialize baseline flight path coordination input
- Create flight path
- Evaluate flight path

Figure 22 depicts a decomposition of these activities into more granular ones.

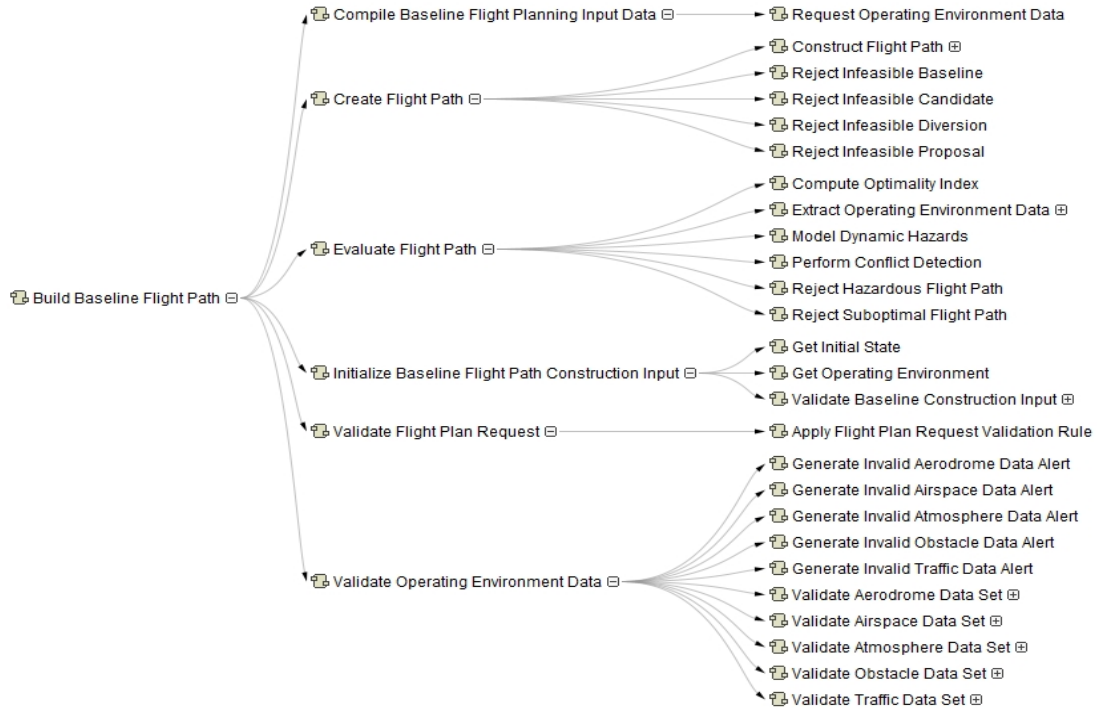


Figure 22: Build Baseline Flight Path.

4.2.2 Coordinate Baseline Flight Path

The system behavior of coordinating a baseline flight path, as depicted in the “Dynamic Path Planning Automation” swimlane of Figure 12, constitutes the following activities:

- Request flight planner concurrence
- Build aerodrome coordination request
- Build airspace coordination request
- Request flight planner confirmation

Figure 23 depicts that these activities are sufficiently granular for deriving the flight path coordination requirements.

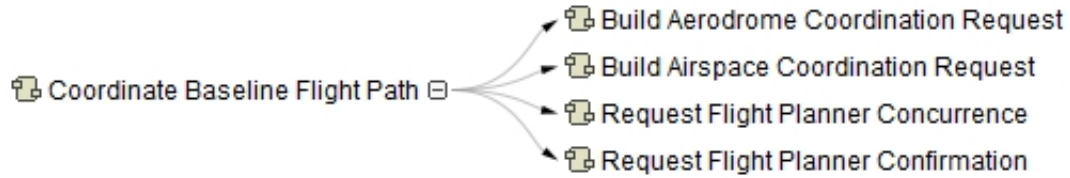


Figure 23: Coordinate Baseline Flight Path.

4.2.3 Synchronize Baseline Flight Path

The system behavior of synchronizing a baseline flight path, as depicted in the “Dynamic Path Planning Automation” swimlane of Figure 13, constitutes the following activities:

- Activate baseline flight path

Figure 24 depicts a decomposition of this activity into more granular ones.



Figure 24: Synchronize Baseline Flight Path.

4.2.4 Adapt Flight Path to Mission Change

The system behavior of adapting the flight path to a change in mission, as depicted in the “Dynamic Path Planning Automation” swimlane of Figure 14, constitutes the following activities:

- Validate flight mission

- Detect mission change trigger
- Update flight path

Figure 25 depicts a decomposition of these activities into more granular ones.

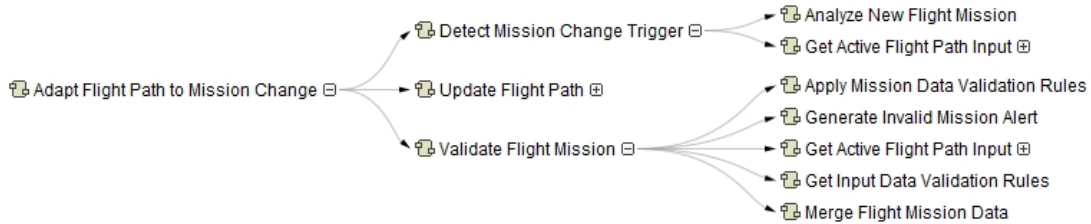


Figure 25: Adapt Flight Path to Mission Change.

4.2.5 Adapt Flight Path to Shared Operating Environment

The system behavior of adapting the flight path to a change in operating environment, as depicted in the “Dynamic Path Planning Automation” swimlane of Figure 15, constitutes the following activities:

- Validate aerodrome data
- Detect aerodrome trigger
- Validate airspace data
- Detect airspace trigger
- Validate atmosphere data
- Detect atmosphere trigger
- Validate obstacle data
- Detect obstacle trigger
- Validate traffic data
- Detect traffic trigger
- Update flight path

Figure 26 depicts a decomposition of these activities into more granular ones.

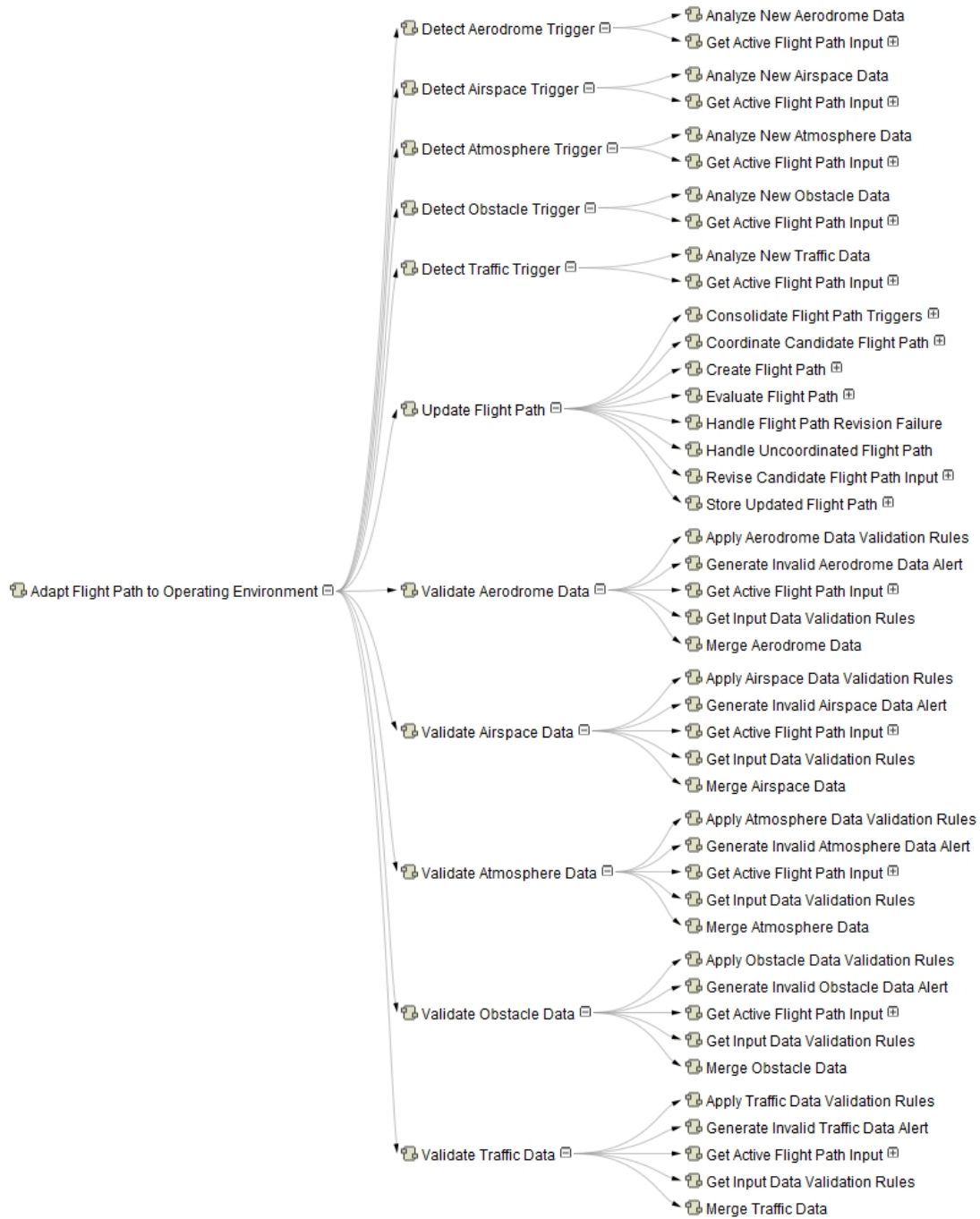


Figure 26: Adapt Flight Path to Shared Operating Environment.

4.2.6 Adapt Flight Path to Sensed Information

The system behavior of adapting the flight path to aircraft sensed changes, as depicted in the “Dynamic Path Planning Automation” swimlane of Figure 16, constitutes the following activities:

- Validate aircraft avionics data
- Detect aircraft state trigger
- Detect atmosphere trigger
- Detect obstacle trigger
- Detect traffic trigger
- Validate aircraft propulsion data
- Detect flight propulsion trigger
- Update flight path

Figure 27 depicts a decomposition of these activities into more granular ones.

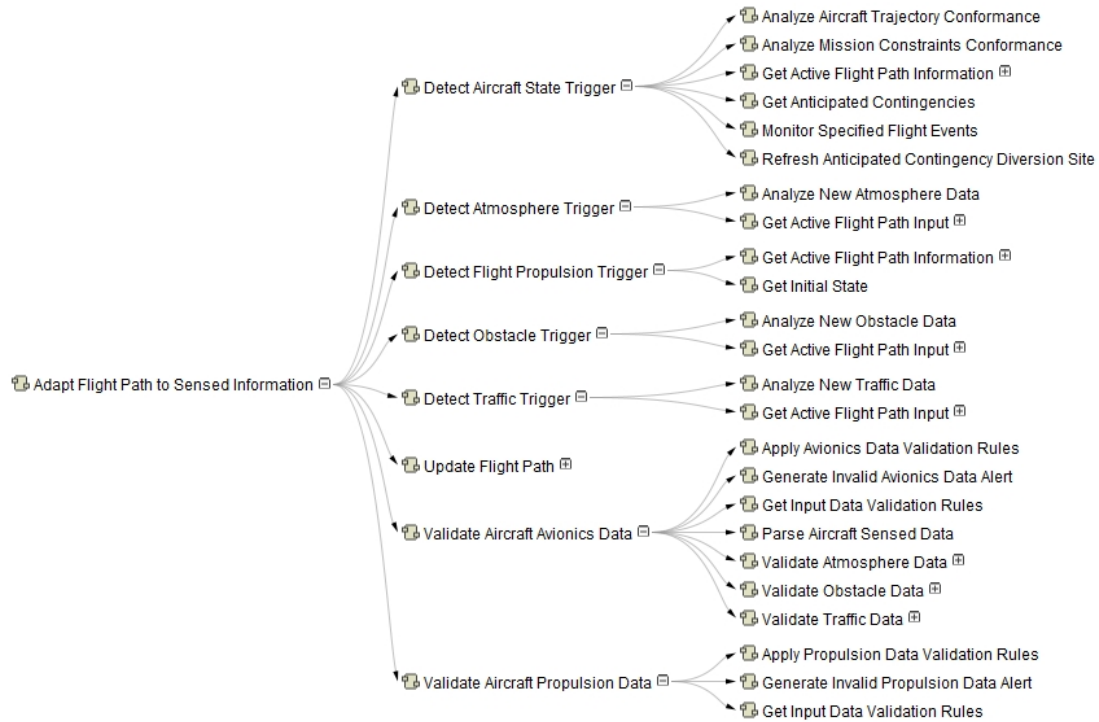


Figure 27: Adapt Flight Path to Sensed Information.

4.2.7 Adapt Flight Path to Pilot Observed Event

The system behavior of adapting the flight path to pilot observed events, as depicted in the “Dynamic Path Planning Automation” swimlane of Figure 17, constitutes the following activities:

- Validate pilot observed event
- Detect onboard emergency trigger
- Detect atmosphere trigger
- Detect obstacle trigger
- Detect traffic trigger
- Update flight path

Figure 28 depicts a decomposition of these activities into more granular ones.

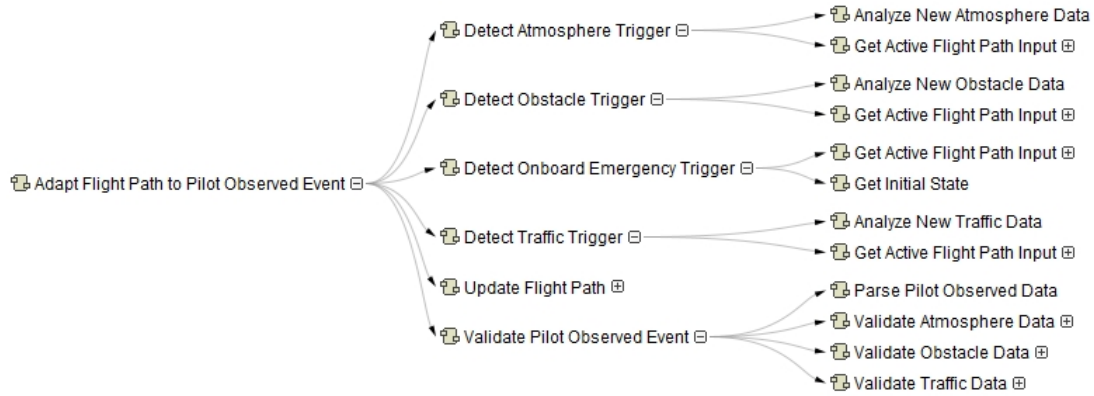


Figure 28: Adapt Flight Path to Pilot Observed Event.

4.2.8 Assess Proposed Flight Path Change

The system behavior of assessing a user-proposed flight path change, as depicted in the “Dynamic Path Planning Automation” swimlane of Figure 18, constitutes the following activities:

- Validate user proposal
- Compile proposed construction input
- Create flight path
- Evaluate flight path
- Coordinate candidate flight path
- Store updated flight path

Figure 29 depicts a decomposition of these activities into more granular ones.

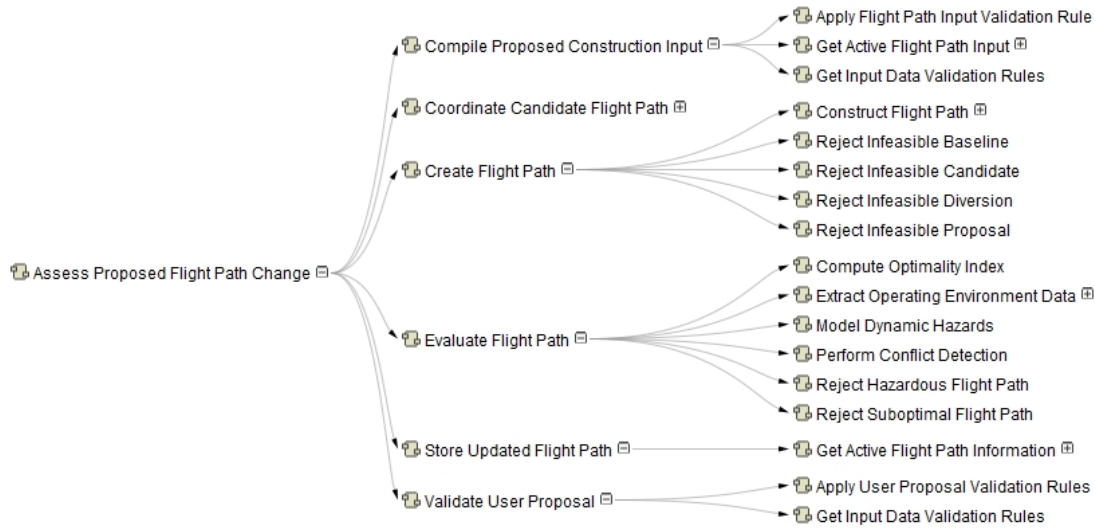


Figure 29: Assess Proposed Flight Path Change.

4.2.9 Update Flight Path

The system behavior of updating the flight path, as depicted in the “Dynamic Path Planning Automation” swimlane of Figure 20, constitutes the following activities:

- Consolidate flight path triggers
- Create flight path
- Evaluate flight path
- Coordinate candidate flight path
- Revise candidate flight path input
- Handle flight path revision failure
- Handle uncoordinated flight path
- Store updated flight path

Figure 30 depicts a decomposition of these activities into more granular ones.

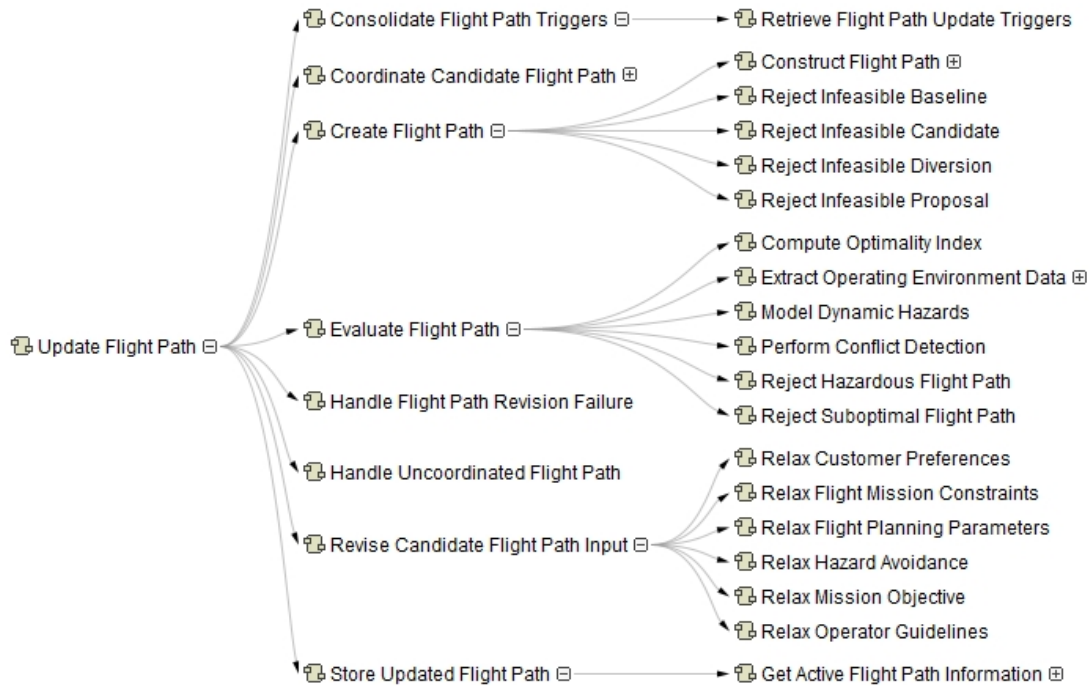


Figure 30: Update Flight Path.

4.2.10 Coordinate Candidate Flight Path

The system behavior of coordinating a candidate flight path, as depicted in the “Dynamic Path Planning Automation” swimlane of Figure 21, constitutes the following activities:

- Build pilot concurrence request
- Build aerodrome coordination request
- Build airspace coordination request
- Build pilot confirmation request
- Build flight path execution request
- Build flight path transmission request
- Analyze flight path coordination response

Figure 31 depicts that these activities are sufficiently granular for deriving the flight path coordination requirements.

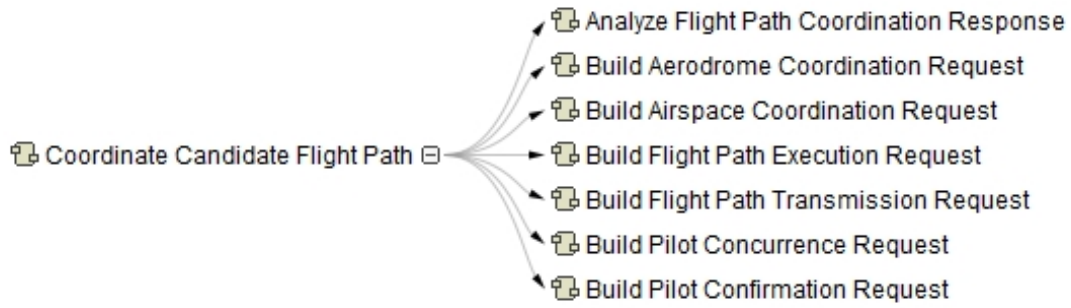


Figure 31: Coordinate Candidate Flight Path.

4.2.11 Mitigate Flight Path Contingency

The system behavior of mitigating an observed or identified flight path contingency event, as depicted in the “Dynamic Path Planning Automation” swimlane of Figure 19, constitutes the following activities:

- Initialize contingency mitigation
- Create flight path
- Evaluate flight path
- Coordinate candidate flight path
- Revise contingency mitigation
- Adapt flight path to mission change
- Store updated flight path

Figure 32 depicts a decomposition of these activities into more granular ones. A unique feature of this behavior, not shown here, is that it alters the state of the DPP automation system between “Nominal” and “Contingency” states.

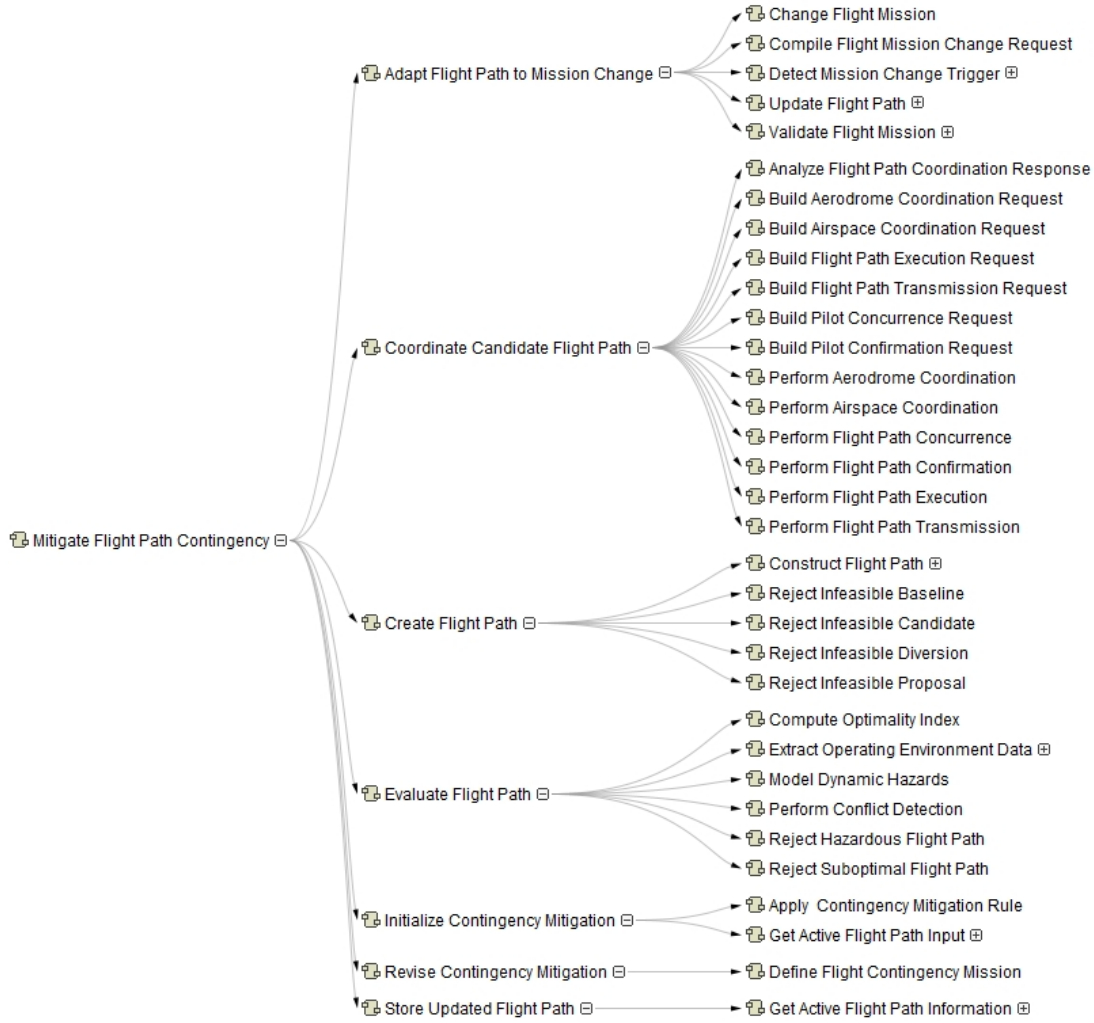


Figure 32: Mitigate Flight Path Contingency.

4.3 Structural Decomposition

The outcome of the DPP automation system behavioral decomposition is a set of system activities which enable the system to be organized into following functional units:

Data Validation function performs validation of all incoming data to the system as well as internal data that needs to be validated. This function is responsible for the following actions:

- Compile proposed construction input
- Validate aerodrome data
- Validate aircraft avionics data
- Validate aircraft propulsion data

- Validate airspace data
- Validate atmosphere data
- Validate flight mission
- Validate flight plan request
- Validate obstacle data
- Validate operating environment data
- Validate pilot observed event
- Validate traffic data
- Validate user proposal

Initialization function initializes the system during preflight planning. This function is responsible for the following actions:

- Compile baseline flight planning input
- Initialize flight path construction input

Flight Path Monitoring function monitors the flight path and the operating environment during the flight. This function is responsible for the following actions:

- Consolidate flight path triggers
- Detect aerodrome trigger
- Detect aircraft state trigger
- Detect airspace trigger
- Detect atmosphere trigger
- Detect configuration change trigger
- Detect flight path refresh trigger
- Detect flight propulsion trigger
- Detect mission change trigger
- Detect obstacle trigger
- Detect onboard emergency trigger
- Detect traffic trigger

Flight Path Creation function builds a feasible flight path for a given flight path construction input. This function is responsible for the following action:

- Create flight path

Flight Path Evaluation function evaluates a candidate flight path for safety, flexibility, and optimality. This function is responsible for the following action:

- Evaluate flight path

Flight Path Revision function revises a candidate flight path to improve the flight path qualities. This function is responsible for the following actions:

- Handle flight path revision failure
- Handle uncoordinated flight path
- Revise candidate flight path input
- Store updated flight path

Flight Path Coordination function coordinates a candidate flight path with the relevant entities—flight planner, pilot, aerodrome and airspace coordinators, aircraft—to ensure that the flight path is acceptable and executable. This function is responsible for the following actions:

- Analyze flight path coordination response
- Build aerodrome coordination request
- Build airspace coordination request
- Build flight path execution request
- Build flight path transmission request
- Build pilot concurrence request
- Build pilot confirmation request
- Request flight planner concurrence
- Request flight planner confirmation

Flight Contingency Mitigation function mitigates the impact of an identified flight path contingency event on successful execution of the flight mission. This function is responsible for the following actions:

- Initialize contingency mitigation
- Revise contingency mitigation

Output Data Processing function provides relevant results to the authorized external entities. This function is responsible for the following actions:

- Manage system alerts
- Provide system configuration parameters

System Operation function is responsible for monitoring system health, and for ensuring that the system performs the appropriate functions per its operating mode and state. This function is responsible for the following actions:

- Build system health report
- Change system operating mode
- Comply with tactical command
- Generate flight path status report
- Monitor system operation

Figure 33 depicts the structural decomposition of the system into these functional units.

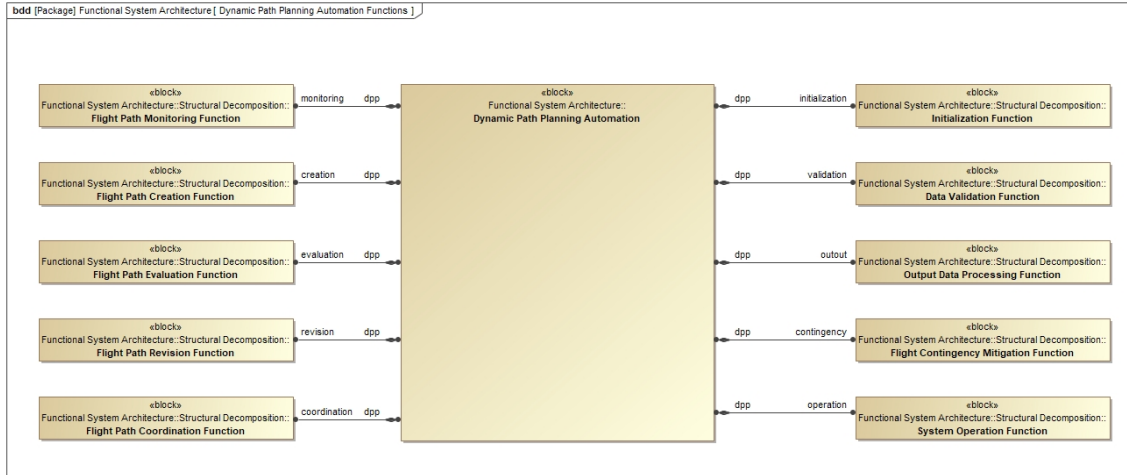


Figure 33: A structural decomposition of the DPP automation system.

4.4 Functional Interfaces

The activities identified during the system behavioral decomposition (§4.2) provide an insight into the internal communication among the system functions identified during the system structural decomposition (§4.3). In addition, the context of the DPP automation system (§3.4) defines the external interactions between the system and the domain participant actors. These internal and external interfaces provide a conduit for the communication of the data needed to accomplish the system goals.

Figure 34 lists the external interfaces of the DPP automation system, and the data that flows through these interfaces. The figure uses the SysML constructs to present the same information as seen in the system context diagrams (Figures 7–8), in an ICD format. The figure shows the name and type of the port⁹ and the types of data which may flow through the port in the direction shown. As observed in the system context diagram, the majority of the interfaces between the system and the external actors are bidirectional; the exception being the avionics, propulsion, and tactical aircraft systems which provide data to the DPP automation system but do not receive data from the DPP automation system.

Figure 35 lists the internal interfaces between the functions identified during the structural decomposition of the DPP automation system. The interfaces define the types and directions of the data which may flow across these interfaces.

Figure 36 depicts the flow of data across the DPP automation system boundary via the external interfaces, and the data communication between the DPP automation system functions via the internal interfaces.

⁹A proxy port in SysML represents an owning block's external interface.

#	Port Name	Port Type	Type Features	△ Direction
1	proxy_avionics	Avionics Functional Interface	in data : Aircraft Avionics Data	in
2	proxy_propulsion	Aircraft Propulsion Functional Interface	in data : Aircraft Propulsion Data	in
3	proxy_tactical	Aircraft Tactical Functional Interface	in command : Tactical Override Command	in
4	proxy_aerodrome	Aerodrome Data Provider Functional Interface	in data : Aerodrome Data out aerodromes : AerodromeIcaoKind [1..*]	inout
5	proxy_airspace	Airspace Data Provider Functional Interface	in data : Airspace Data out firs : FlightInformationRegionKind [1..*]	inout
6	proxy_atmosphere	Atmosphere Data Provider Functional Interface	in data : Atmosphere Data out firs : FlightInformationRegionKind	inout
7	proxy_coord_aero	Aerodrome Coordinator Functional Interface	in coord_res : Aerodrome Coordination Response in proposal : Proposal Assessment Request in status_req : Flight Path Status Request out alert : Flight Path Alert out coord_req : Aerodrome Coordination Request out response : Proposal Assessment Response out status_res : Flight Path Status Report	inout
8	proxy_coord_air	Airspace Coordinator Functional Interface	in coord_res : Airspace Coordination Response in proposal : Proposal Assessment Request in status_req : Flight Path Status Request out alert : Flight Path Alert out coord_req : Airspace Coordination Request out response : Proposal Assessment Response out status_res : Flight Path Status Report	inout
9	proxy_guidance	Aircraft Guidance Functional Interface	in response : Flight Path Execution Response out request : Flight Path Execution Request	inout
10	proxy_obstacle	Obstacle Data Provider Functional Interface	in data : Obstacle Data out firs : FlightInformationRegionKind	inout
11	proxy_operator	Mission Manager Functional Interface	in baseline : Flight Path Information in mission : Flight Mission in proposal : Proposal Assessment Request in status_request : Flight Path Status Request out alerts : Flight Path Alert [0..*] out health : System Health out response : Proposal Assessment Response out status : Flight Path Status Report	inout
12	proxy_pilot	Pilot Functional Interface	in anomaly : Pilot Observed Event in concur : Pilot Concurrence Response in confirm : Pilot Confirmation Response in mission : Flight Mission in proposal : Proposal Assessment Request in tactical : Tactical Override Command out alerts : Flight Path Alert out concur_req : Pilot Concurrence Request out confirm_req : Pilot Confirmation Request out context : Flight Path Context out health : System Health out path : Flight Path out response : Proposal Assessment Response config : Configuration Parameters	inout
13	proxy_planner	Flight Planner Functional Interface	in concur : Planner Concurrence Response in confirm : Planner Confirmation Response in request : Flight Planning Request out alerts : Flight Path Alert [0..*] out concur_req : Planner Concurrence Request out confirm_req : Planner Confirmation Request out context : Flight Path Context out health : System Health out path : Flight Path out status : Flight Path Coordination Status config : Configuration Parameters	inout
14	proxy_sync	Baseline Synchronization Functional Interface	inout context : Flight Path Context inout path : Flight Path	inout
15	proxy_traffic	Traffic Data Provider Functional Interface	in data : Traffic Data out firs : FlightInformationRegionKind	inout

Figure 34: DPP automation system interfaces with AAM domain elements.

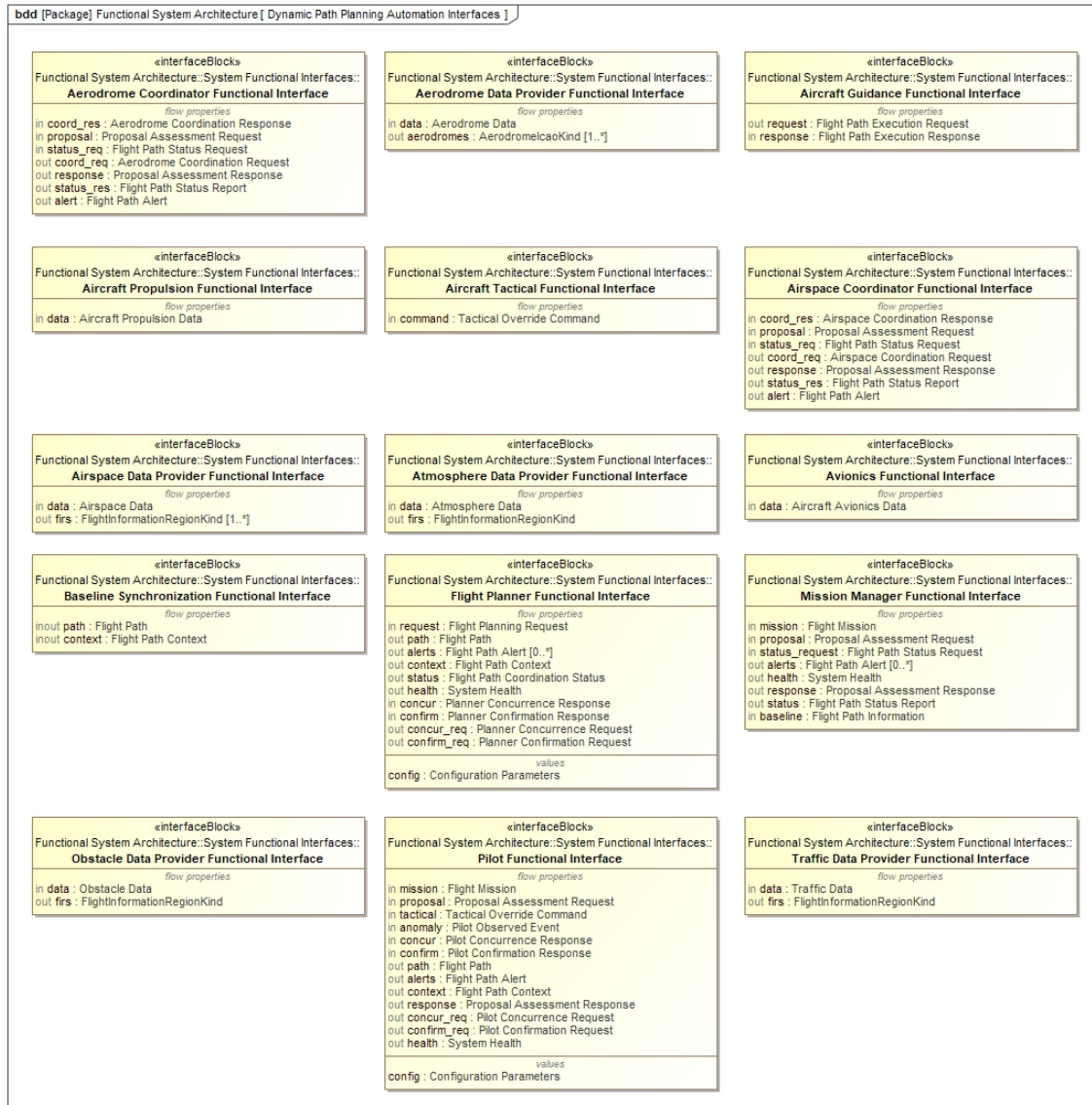


Figure 35: DPP automation system internal functional interfaces.

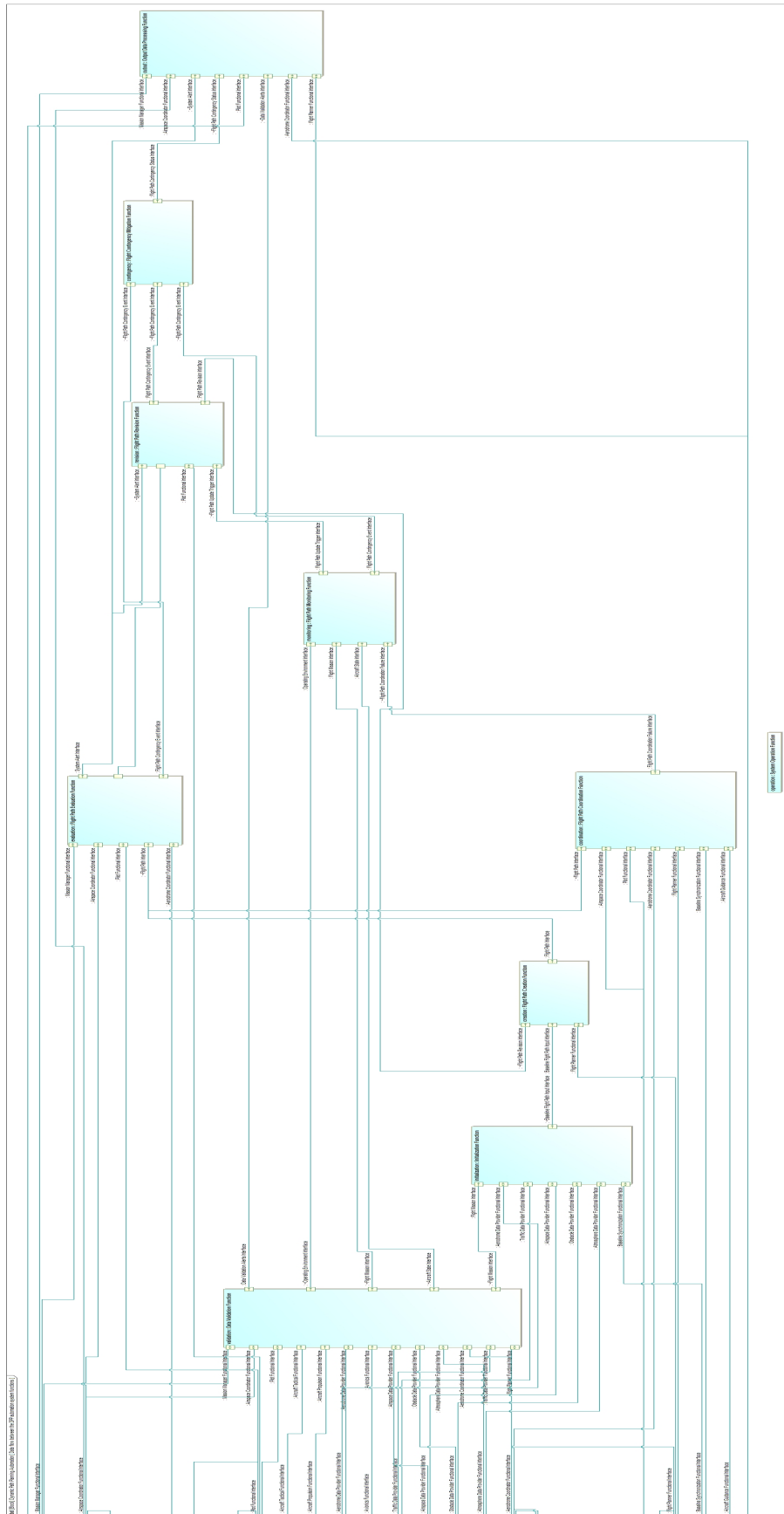


Figure 36: DPP automation system interfunctional communication.

4.5 Functional Requirements Elaboration

This section elaborates on the top-level system functional requirements, defined in §3.7, based on the system activities identified as part of the system behavioral decomposition §4.2.

An example of the DPP automation system functional requirements, derived from the actions of the “Build Baseline Flight Path” use case, results in the following expanded list of requirements:

SR-F.1 Build Baseline Flight Path

SR-F.1.1 Receive Flight Plan Request

SR-F.1.2 Validate Flight Planning Request

SR-F.1.2.1 Apply Request Validation Rule

SR-F.1.2.2 Alert of Invalid Flight Plan Request

SR-F.1.3 Compile Baseline Flight Planning Input

SR-F.1.3.1 Parse Flight Mission Specification

SR-F.1.3.2 Select Aircraft Performance Model

SR-F.1.3.3 Request Aerodrome Data

SR-F.1.3.4 Request Airspace Data

SR-F.1.3.5 Request Atmosphere Data

SR-F.1.3.6 Request Obstacle Data

SR-F.1.3.7 Request Traffic Data

SR-F.1.3.8 Determine Shared Data Need

SR-F.1.4 Compile Shared Operating Environment Data

SR-F.1.4.1 Receive Aerodrome Data

SR-F.1.4.2 Receive Airspace Data

SR-F.1.4.3 Receive Atmosphere Data

SR-F.1.4.4 Receive Obstacle Data

SR-F.1.4.5 Receive Traffic Data

SR-F.1.4.6 Validate Operating Environment Data

SR-F.1.4.7 Alert of Invalid Operating Environment Data

SR-F.1.4.8 Store Valid Operating Environment

SR-F.1.5 Initialize Flight Path Construction

SR-F.1.5.1 Compile Flight Path Construction Input

SR-F.1.5.2 Validate Flight Path Construction Input

SR-F.1.5.3 Alert of Unachievable Mission

SR-F.1.6 Create Flight Path

SR-F.1.6.1 Construct Flight Path

SR-F.1.6.2 Alert of Infeasible Flight Path

SR-F.1.7 Evaluate Flight Path

- SR-F.1.7.1 Evaluate Flight Path Safety
- SR-F.1.7.2 Evaluate Flight Path Optimality
- SR-F.1.7.3 Alert of Unmet Flight Path Qualities
- SR-F.1.7.4 Provide Flight Path Information

Another example of the DPP automation system functional requirements, derived from the actions of the “Adapt Flight Path to Mission Change” use case, results in the following expanded list of requirements:

SR-F.7 Adapt Flight Path to Mission Change

- SR-F.7.1 Receive New Mission Command
- SR-F.7.2 Validate Mission Data
 - SR-F.7.2.1 Merge Mission Data
 - SR-F.7.2.2 Apply Mission Validation Rule
 - SR-F.7.2.3 Generate Invalid Mission Alert
 - SR-F.7.2.4 Discard Invalid Mission
 - SR-F.7.2.5 Store Valid Mission Data
- SR-F.7.3 Ensure Operating Environment Availability
 - SR-F.7.3.1 Identify Missing Operating Environment Data
 - SR-F.7.3.2 Request Missing Aerodrome Data
 - SR-F.7.3.3 Request Missing Airspace Data
 - SR-F.7.3.4 Request Missing Atmosphere Data
 - SR-F.7.3.5 Request Missing Obstacle Data
 - SR-F.7.3.6 Request Missing Traffic Data
 - SR-F.7.3.7 Receive Missing Aerodrome Data
 - SR-F.7.3.8 Receive Missing Airspace Data
 - SR-F.7.3.9 Receive Missing Atmosphere Data
 - SR-F.7.3.10 Receive Missing Obstacle Data
 - SR-F.7.3.11 Receive Missing Traffic Data
 - SR-F.7.3.12 Validate Aerodrome Data
 - SR-F.7.3.13 Generate Invalid Aerodrome Data Alert
 - SR-F.7.3.14 Discard Invalid Aerodrome Data
 - SR-F.7.3.15 Store Valid Aerodrome Data
 - SR-F.7.3.16 Validate Airspace Data
 - SR-F.7.3.17 Generate Invalid Airspace Data Alert
 - SR-F.7.3.18 Discard Invalid Airspace Data
 - SR-F.7.3.19 Store Valid Airspace Data
 - SR-F.7.3.20 Validate Atmosphere Data
 - SR-F.7.3.21 Generate Invalid Atmosphere Data Alert
 - SR-F.7.3.22 Discard Invalid Atmosphere Data
 - SR-F.7.3.23 Store Valid Atmosphere Data

- SR-F.7.3.24 Validate Obstacle Data
- SR-F.7.3.25 Generate Invalid Obstacle Data Alert
- SR-F.7.3.26 Discard Invalid Obstacle Data
- SR-F.7.3.27 Store Valid Obstacle Data
- SR-F.7.3.28 Validate Traffic Data
- SR-F.7.3.29 Generate Invalid Traffic Data Alert
- SR-F.7.3.30 Discard Invalid Traffic Data
- SR-F.7.3.31 Store Valid Traffic Data
- SR-F.7.4 Detect New Mission Impact
 - SR-F.7.4.1 Confirm Unimpacted Flight Path
 - SR-F.7.4.2 Analyze New Flight Mission
 - SR-F.7.4.3 Build Mission Change Trigger
- SR-F.7.5 Update Flight Path
- SR-F.7.6 Report Successful Flight Path Adaptation
- SR-F.7.7 Report Unimpacted Flight Path
- SR-F.7.8 Report Failure

Section 5 specifies the complete set of requirements of the DPP automation system.

5 System Requirements Specification

This section specifies a detailed, verifiable, complete, and consistent set of the DPP automation system requirements in an AAM operating environment. The purpose of specifying the system requirements is to lay an unambiguous foundation in preparation for developing the logical architecture of the system. As observed during the conceptual architecture development, the system requirements can be categorized as:

Functional requirements derived from the behavioral decomposition.

Usability requirements derived from the measures of effectiveness.

Design constraints based on the corresponding stakeholder concerns.

Physical requirements based on the corresponding stakeholder concerns.

Interface requirements derived from the system context.

5.1 Synopsis

Section 5.2 lists the functional requirements of the DPP automation system. These requirements are specified in congruence with the 17 distinct use cases of the system:

1. Build baseline flight path
2. Coordinate baseline flight path
3. Synchronize baseline flight path
4. Activate baseline flight path
5. Adapt flight path to pilot observed event
6. Adapt flight path to shared operating environment
7. Adapt flight path to mission change
8. Adapt flight path to sensed information
9. Adapt flight path to system configuration change
10. Assess flight path change proposal
11. Refresh flight path periodically
12. Coordinate candidate flight path
13. Update candidate flight path
14. Mitigate flight path contingency
15. Comply with tactical override command
16. Monitor system performance
17. Perform postflight analysis

Section 5.3 lists the usability requirements of the DPP automation system. These nonfunctional requirements indicate the qualities of the DPP automation system related to alerting, response time, availability, and customizability. The usability requirements

are expressed in terms of the nuisance alerts, false alerts, missed alerts, system response time, availability of interfaces, and customization of inputs and outputs.

Section 5.4 lists the design constraints of the DPP automation system derived from the stakeholder concerns about regulations, ANSP rules, and system accessibility.

Section 5.5 lists an initial draft of the physical requirements of the DPP automation system related to system installation onboard the aircraft, on the ground, and at aerodrome and airspace coordinator stations.

Section 5.6 lists the interface requirements of the DPP automation system. These nonfunctional requirements define the types of interfaces and the data flowing through these interfaces. The DPP automation system requires bidirectional data exchange with the mission manager, flight planner, pilot-in-command, aerodrome and airspace coordinators, aircraft guidance and communication systems, shared operating environment data providers, and system administrator. The DPP automation system also requires sensed flight data from the aircraft avionics and propulsion systems, and tactical maneuver data from the aircraft tactical maneuver management system.

5.2 Functional Requirements

The list of system functional requirements includes:

SR-F.1 Build Baseline Flight Path

The Dynamic Path Planning Automation System shall build a flight path that meets the mission objectives and operator preferences, subject to the aircraft performance and operating environment constraints, within a specified timeframe, upon receiving the flight planning request from the flight planner via the flight planner graphical user interface.

SR-F.1.1 Receive Flight Plan Request

The Dynamic Path Planning Automation System shall receive a flight planning request from the flight planner via the flight planner graphical user interface.

SR-F.1.2 Validate Flight Planning Request

Each time it receives a new flight planning request, the Dynamic Path Planning Automation System shall validate the request to ensure its completeness and correctness.

SR-F.1.2.1 Apply Request Validation Rule

The Dynamic Path Planning Automation System shall apply the appropriate criteria to determine the validity of the flight plan request.

SR-F.1.2.2 Alert of Invalid Flight Plan Request

If the flight planning request is deemed invalid, the Dynamic Path Planning Automation System shall alert the planner of discrepancies in the flight planning request within a specified timeframe, upon receiving the request.

SR-F.1.3 Compile Baseline Flight Planning Input

Each time it receives a valid flight planning request, the Dynamic Path Planning Automation System shall compile the data necessary to construct a valid baseline flight path.

- SR-F.1.3.1 Parse Flight Mission Specification
The Dynamic Path Planning Automation System shall parse the flight planning request to determine the initial and terminal conditions of the flight.
- SR-F.1.3.2 Select Aircraft Performance Model
The Dynamic Path Planning Automation System shall parse the flight planning request to identify the type of aircraft to be used for the flight.
- SR-F.1.3.3 Request Aerodrome Data
The Dynamic Path Planning Automation System shall transmit an aerodrome data request containing the list of aerodrome identifiers to the selected aerodrome service provider via a ground-to-ground connection.
- SR-F.1.3.4 Request Airspace Data
The Dynamic Path Planning Automation System shall transmit an airspace data request containing the list of flight information region identifiers to the selected airspace service provider via a ground-to-ground connection.
- SR-F.1.3.5 Request Atmosphere Data
The Dynamic Path Planning Automation System shall transmit an atmosphere data request containing the list of flight information region identifiers to the selected atmosphere service provider via a ground-to-ground connection.
- SR-F.1.3.6 Request Obstacle Data
The Dynamic Path Planning Automation System shall transmit an obstacle data request containing the list of flight information region identifiers to the selected obstacle service provider via a ground-to-ground connection.
- SR-F.1.3.7 Request Traffic Data
The Dynamic Path Planning Automation System shall transmit a traffic data request containing the list of flight information region identifiers to the selected traffic service provider via a ground-to-ground connection.
- SR-F.1.3.8 Determine Shared Data Need
The Dynamic Path Planning Automation System shall parse the flight plan request to determine the list of aerodromes and airspace regions for which to request the necessary operating environment data.
- SR-F.1.4 Compile Shared Operating Environment Data
The Dynamic Path Planning Automation System shall synchronously receive the requested operating environment data from applicable data service providers via a ground-to-ground connection.
- SR-F.1.4.1 Receive Aerodrome Data
The Dynamic Path Planning Automation System shall synchronously receive requested aerodrome data sent by aerodrome data service provider via a ground-to-ground connection.
- SR-F.1.4.2 Receive Airspace Data
The Dynamic Path Planning Automation System shall synchronously receive requested airspace data sent by airspace data service provider via a ground-to-ground connection.

- SR-F.1.4.3 Receive Atmosphere Data**
The Dynamic Path Planning Automation System shall synchronously receive requested atmosphere data sent by atmosphere data service provider via a ground-to-ground connection.
- SR-F.1.4.4 Receive Obstacle Data**
The Dynamic Path Planning Automation System shall synchronously receive requested obstacle data sent by obstacle data service provider via a ground-to-ground connection.
- SR-F.1.4.5 Receive Traffic Data**
The Dynamic Path Planning Automation System shall synchronously receive requested traffic data sent by traffic data service provider via a ground-to-ground connection.
- SR-F.1.4.6 Validate Operating Environment Data**
Upon receipt of operating environment data from aerodrome, atmosphere, airspace, obstacle, and traffic data providers, the Dynamic Path Planning Automation System shall validate data for sufficiency and correctness.
- SR-F.1.4.7 Alert of Invalid Operating Environment Data**
If the operating environment data is deemed invalid, the Dynamic Path Planning Automation System shall alert the planner of the discrepancy in data.
- SR-F.1.4.8 Store Valid Operating Environment**
Upon receipt of valid operating environment data from aerodrome, atmosphere, airspace, obstacle, and traffic data providers, the Dynamic Path Planning Automation System shall store the data set as the baseline operating environment data.
- SR-F.1.5 Initialize Flight Path Construction**
The Dynamic Path Planning Automation System shall initialize the flight path construction input in less than a specified time.
- SR-F.1.5.1 Compile Flight Path Construction Input**
The Dynamic Path Planning Automation System shall compile the flight path construction input based on the flight mission specifications, flight planning parameters, aircraft initial state, aircraft type, and the baseline operating environment data.
- SR-F.1.5.2 Validate Flight Path Construction Input**
Each time it compiles a new flight path construction input, the Dynamic Path Planning Automation System shall validate the input to comply with the flight path input validation rules.
- SR-F.1.5.3 Alert of Unachievable Mission**
If the baseline flight path construction input is deemed invalid, the Dynamic Path Planning Automation System shall alert the planner of unachievable mission specifications within a specified timeframe, upon receiving the flight planning request.
- SR-F.1.6 Create Flight Path**
Each time the flight path construction input is deemed valid, the Dynamic Path Planning Automation System shall create a candidate flight path.

SR-F.1.6.1 Construct Flight Path

The Dynamic Path Planning Automation System shall construct a flight path in response to a valid flight path construction input in less than a specified time.

SR-F.1.6.2 Alert of Infeasible Flight Path

If it fails to build a feasible baseline flight path, the Dynamic Path Planning Automation System shall alert the planner of flight path construction error containing infeasible flight planning request elements within a specified timeframe, upon receiving the flight path construction input.

SR-F.1.7 Evaluate Flight Path

Each time it builds a feasible baseline flight path, the Dynamic Path Planning Automation System shall evaluate the qualities of the flight path in less than a specified time of receipt of the flight path.

SR-F.1.7.1 Evaluate Flight Path Safety

Each time it builds a feasible baseline flight path, the Dynamic Path Planning Automation System shall detect any conflict with the known hazards.

SR-F.1.7.2 Evaluate Flight Path Optimality

Each time it builds a feasible baseline flight path without any conflict with the known hazards, the Dynamic Path Planning Automation System shall compute the optimality index of the flight path.

SR-F.1.7.3 Alert of Unmet Flight Path Qualities

If it fails to build a conflict free flight path with acceptable level of optimality specified in the flight planning request, the Dynamic Path Planning Automation System shall inform the flight planner of unmet flight path qualities.

SR-F.1.7.4 Provide Flight Path Information

Each time it builds a flight path with acceptable flight path qualities, the Dynamic Path Planning Automation System shall transmit the flight path information to the flight planner via flight planner graphical user interface within a specified timeframe, upon receiving the mission specifications.

SR-F.2 Coordinate Baseline Flight Path

The Dynamic Path Planning Automation System shall coordinate the planned flight path with the selected flight path coordinators upon successful creation of a valid baseline flight path.

SR-F.2.1 Request Flight Planner Concurrence

The Dynamic Path Planning Automation System shall request the flight planner's concurrence, via flight planner graphical user interface, to coordinate the planned flight path with the relevant flight path coordinators.

SR-F.2.2 Receive Flight Planner Concurrence

The Dynamic Path Planning Automation System shall receive the flight planner's concurrence via flight planner graphical user interface.

- SR-F.2.3 Request Aerodrome Coordination**
Upon the flight planner's concurrence, the Dynamic Path Planning Automation System shall coordinate the flight plan with the selected aerodrome coordinator via ground-to-ground connection.
- SR-F.2.4 Receive Aerodrome Coordination Response**
The Dynamic Path Planning Automation System shall receive the flight plan coordination response from the selected aerodrome coordinator via ground-to-ground connection.
- SR-F.2.5 Request Airspace Coordination**
Upon the flight planner's concurrence, the Dynamic Path Planning Automation System shall coordinate the flight plan with the selected airspace coordinator via ground-to-ground connection.
- SR-F.2.6 Receive Airspace Coordination Response**
The Dynamic Path Planning Automation System shall receive the flight plan coordination response from the selected airspace coordinator via ground-to-ground connection.
- SR-F.2.7 Request Flight Planner Confirmation**
Upon receipt of the aerodrome and airspace coordination responses, the Dynamic Path Planning Automation System shall provide the consolidated flight path coordination status to the flight planner via the flight planner graphical user interface.
- SR-F.2.8 Notify Successful Baseline Coordination**
Upon the flight planner's confirmation, the Dynamic Path Planning Automation System shall notify the mission manager of successful computation of the baseline flight path.
- SR-F.3 Synchronize Baseline Flight Path**
The Dynamic Path Planning Automation System shall initialize the onboard baseline flight path with the planned flight path upon receipt of a flight path synchronization command from the mission manager via the operator communication link.
- SR-F.3.1 Receive Planned Flight Path**
The Dynamic Path Planning Automation System shall asynchronously receive the baseline flight path information from the mission manager after establishing a connection between the mission operations center and the aircraft.
- SR-F.3.2 Receive Baseline Activation Confirmation**
The Dynamic Path Planning Automation System shall asynchronously receive a notification of successful activation of the baseline flight path.
- SR-F.3.3 Notify Flight Path Synchronization**
Upon successful activation of the baseline flight path, the Dynamic Path Planning Automation System shall notify the mission manager, via operator communication link, of successful synchronization of the flight path between the ground and the onboard installations of the system.

SR-F.4 Activate Baseline Flight Path

The Dynamic Path Planning Automation System shall activate the baseline flight path upon receipt of baseline flight path.

SR-F.4.1 Request Pilot Concurrence

The Dynamic Path Planning Automation System shall provide flight path information to the onboard pilot via the pilot graphical user interface.

SR-F.4.2 Receive Pilot Concurrence

The Dynamic Path Planning Automation System shall receive the pilot concurrence response from the onboard pilot via the pilot graphical user interface.

SR-F.4.3 Analyze Pilot Concurrence

Upon receipt of the pilot concurrence response, the Dynamic Path Planning Automation System shall determine if the pilot concurs with the planned flight path.

SR-F.4.4 Alert Pilot Nonconcurrence

The Dynamic Path Planning Automation System shall alert of pilot nonconcurrence if the pilot does not concur with the planned flight path.

SR-F.4.5 Request Flight Path Execution

Upon pilot's acceptance of the flight path, the Dynamic Path Planning Automation System shall transmit flight path execution request to aircraft guidance system via aircraft avionics bus.

SR-F.4.6 Receive Flight Path Execution Response

The Dynamic Path Planning Automation System shall receive the flight path execution response from the aircraft guidance system via aircraft avionics bus.

SR-F.4.7 Analyze Flight Path Execution Response

Upon receipt of the flight path execution response, the Dynamic Path Planning Automation System shall determine if the aircraft guidance system has executed the planned flight path.

SR-F.4.8 Alert Flight Path Execution Failure

The Dynamic Path Planning Automation System shall alert of nonexecutable flight path if the aircraft fails to execute the planned flight path.

SR-F.4.9 Confirm Flight Path Activation

The Dynamic Path Planning Automation System shall confirm successful activation of the planned flight path.

SR-F.5 Adapt Flight Path to Pilot Observed Event

The Dynamic Path Planning Automation System shall adapt the active flight path to pilot observed anomalies received via pilot graphical user interface within a specified timeframe, upon receiving the pilot instruction.

SR-F.5.1 Receive Observed Atmosphere

The Dynamic Path Planning Automation System shall asynchronously receive

all observed hazardous atmosphere data reported by the pilot via the pilot graphical user interface.

SR-F.5.2 Validate Observed Atmosphere Data

Each time it receives pilot observed atmosphere data, the Dynamic Path Planning Automation System shall validate it per the atmosphere data validation rules.

SR-F.5.2.1 Merge Atmosphere Data

The Dynamic Path Planning Automation System shall merge the observed atmosphere data with the stored data to form a complete picture of the atmosphere operating environment.

SR-F.5.2.2 Apply Atmosphere Validation Rule

The Dynamic Path Planning Automation System shall apply the appropriate criteria to determine the validity of the observed atmosphere data.

SR-F.5.2.3 Generate Invalid Atmosphere Data Alert

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall generate an invalid atmosphere data alert within a specified timeframe, upon receiving the data.

SR-F.5.2.4 Discard Invalid Atmosphere Data

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid atmosphere data.

SR-F.5.2.5 Store Valid Atmosphere Data

Each time valid atmosphere data is received, the Dynamic Path Planning Automation System shall update the stored atmosphere data.

SR-F.5.3 Detect Atmosphere Data Impact

Each time it receives valid atmosphere data, the Dynamic Path Planning Automation System shall determine its impact on the active flight path within a specified timeframe, upon receiving the data.

SR-F.5.3.1 Analyze Atmosphere Data

Each time it receives valid atmosphere data, the Dynamic Path Planning Automation System shall determine if the active flight path needs to be updated in response to the new data.

SR-F.5.3.2 Build Aerodrome Induced Trigger

Each time it detects a need to update the active flight path in response to the new atmosphere data, the Dynamic Path Planning Automation System shall generate a new flight path revision trigger.

SR-F.5.3.3 Confirm Nonimpact of Aerodrome Data

Each time it determines the active flight path to be unaffected by the new atmosphere data, the Dynamic Path Planning Automation System shall generate a message indicating continued validity of the active flight path.

SR-F.5.4 Receive Observed Obstacle

The Dynamic Path Planning Automation System shall asynchronously receive all observed obstacle data reported by the pilot via the pilot graphical user interface.

SR-F.5.5 Validate Obstacle Data

Each time it receives pilot observed obstacle data, the Dynamic Path Planning Automation System shall validate it per the obstacle data validation rules.

SR-F.5.5.1 Merge Obstacle Data

The Dynamic Path Planning Automation System shall merge the observed obstacle data with the stored data to form a complete picture of the obstacle operating environment.

SR-F.5.5.2 Apply Obstacle Validation Rule

The Dynamic Path Planning Automation System shall apply the appropriate criteria to determine the validity of the observed obstacle data.

SR-F.5.5.3 Generate Invalid Obstacle Data Alert

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall generate an invalid obstacle data alert within a specified timeframe, upon receiving the data.

SR-F.5.5.4 Discard Invalid Obstacle Data

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid obstacle data.

SR-F.5.5.5 Store Valid Obstacle Data

Each time valid obstacle data is received, the Dynamic Path Planning Automation System shall update the stored obstacle data.

SR-F.5.6 Detect Obstacle Data Impact

Each time it receives valid obstacle data, the Dynamic Path Planning Automation System shall determine its impact on the active flight path within a specified timeframe, upon receiving the data.

SR-F.5.6.1 Analyze Obstacle Data

Each time it receives valid obstacle data, the Dynamic Path Planning Automation System shall determine if the active flight path needs to be updated in response to the new data.

SR-F.5.6.2 Build Obstacle Induced Trigger

Each time it detects a need to update the active flight path in response to the new obstacle data, the Dynamic Path Planning Automation System shall generate a new flight path revision trigger.

SR-F.5.6.3 Confirm Nonimpact of Obstacle Data

Each time it determines the active flight path to be unaffected by the new obstacle data, the Dynamic Path Planning Automation System shall generate a message indicating continued validity of the active flight path.

SR-F.5.7 Receive Observed Traffic

The Dynamic Path Planning Automation System shall asynchronously receive all observed noncooperative or nonbroadcasting or otherwise anomalous behavior of traffic aircraft reported by the pilot via the pilot graphical user interface.

SR-F.5.8 Validate Observed Traffic Data

Each time it receives pilot observed traffic data, the Dynamic Path Planning Automation System shall validate it per the traffic data validation rules.

- SR-F.5.8.1 Merge Traffic Data
The Dynamic Path Planning Automation System shall merge the observed traffic data with the stored data to form a complete picture of the traffic operating environment.
- SR-F.5.8.2 Apply Traffic Validation Rule
The Dynamic Path Planning Automation System shall apply the appropriate criteria to determine the validity of the observed traffic data.
- SR-F.5.8.3 Generate Invalid Traffic Data Alert
Each time it deems the data invalid, the Dynamic Path Planning Automation System shall generate an invalid traffic data alert within a specified timeframe, upon receiving the data.
- SR-F.5.8.4 Discard Invalid Traffic Data
Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid traffic data.
- SR-F.5.8.5 Store Valid Traffic Data
Each time valid traffic data is received, the Dynamic Path Planning Automation System shall update the stored traffic data.
- SR-F.5.9 Detect Traffic Data Impact
Each time it receives valid traffic data, the Dynamic Path Planning Automation System shall determine its impact on the active flight path within a specified timeframe, upon receiving the data.
 - SR-F.5.9.1 Analyze Traffic Data
Each time the traffic data is updated, the Dynamic Path Planning Automation System shall determine if the active flight path needs to be updated in response to the new data.
 - SR-F.5.9.2 Build Traffic Induced Trigger
Each time it detects a need to update the active flight path in response to the new traffic data, the Dynamic Path Planning Automation System shall generate a new flight path revision trigger.
 - SR-F.5.9.3 Confirm Nonimpact of Traffic Data
Each time it determines the active flight path to be unaffected by the new traffic data, the Dynamic Path Planning Automation System shall generate a message indicating continued validity of the active flight path.
- SR-F.5.10 Receive Reported Onboard Emergency
The Dynamic Path Planning Automation System shall asynchronously receive all onboard emergency messages reported by the pilot via the pilot graphical user interface.
- SR-F.5.11 Detect Onboard Emergency Impact
Each time an onboard emergency is reported, the Dynamic Path Planning Automation System shall determine its impact on the active flight path within a specified timeframe, upon receiving the data.
 - SR-F.5.11.1 Investigate Reported Emergency
Each time an onboard emergency is reported, the Dynamic Path Planning Automation System shall determine if it results in a flight path contingency.

SR-F.5.11.2 Generate Reported Contingency Alert

The Dynamic Path Planning Automation System shall generate a flight path contingency alert if it detects that the reported onboard emergency results in a flight path contingency.

SR-F.5.11.3 Evaluate Reported Emergency

Each time it determines that the reported emergency does not result in a flight path contingency, the Dynamic Path Planning Automation System shall determine if the active flight path needs to be updated in response to the new data.

SR-F.5.11.4 Build Onboard Emergency Induced Trigger

Each time it detects a need to update the active flight path in response to the noncontingency reported emergency, the Dynamic Path Planning Automation System shall generate a new flight path revision trigger.

SR-F.5.12 Update Flight Path

The Dynamic Path Planning Automation System shall compute a viable candidate flight path each time it recognizes a need to update the flight path, within a specified timeframe, upon receiving the flight-path-update trigger.

SR-F.5.13 Mitigate Flight Path Contingency

The Dynamic Path Planning Automation System shall mitigate the impact of all identified contingencies on the execution of the flight mission, in less than a specified time of contingency identification.

SR-F.5.14 Report Successful Flight Path Adaptation

The Dynamic Path Planning Automation System shall report every successful adaptation of the active flight path in response to the pilot observed event, within a specified timeframe, upon receiving the data, via the pilot graphical user interface.

SR-F.5.15 Report Unimpacted Flight Path

The Dynamic Path Planning Automation System shall report each time it identifies no impact of the pilot observed event on the active flight path, within a specified timeframe, upon receiving the data via the pilot graphical user interface.

SR-F.5.16 Report Failure

The Dynamic Path Planning Automation System shall report every unsuccessful attempt to adapt the active flight path in response to the pilot observed event, within a specified timeframe, upon receiving the data via the pilot graphical user interface.

SR-F.6 Adapt Flight Path to Shared Operating Environment

The Dynamic Path Planning Automation System shall adapt the active flight path to changes in the operating environment data within a specified timeframe, upon receiving the data via the ground-to-air service communication link.

SR-F.6.1 Receive Aerodrome Data

The Dynamic Path Planning Automation System shall asynchronously receive

all aerodrome data sent by a selected aerodrome data service provider when the ground-to-air data communication link is operational.

SR-F.6.2 Validate Aerodrome Data

Each time it receives the aerodrome data, the Dynamic Path Planning Automation System shall validate it per the aerodrome data validation rules.

SR-F.6.2.1 Merge Aerodrome Data

The Dynamic Path Planning Automation System shall merge the received aerodrome data with the stored data to form a complete picture of the aerodrome operating environment.

SR-F.6.2.2 Apply Aerodrome Validation Rule

The Dynamic Path Planning Automation System shall apply the appropriate criteria to determine the validity of the received aerodrome data.

SR-F.6.2.3 Generate Invalid Aerodrome Data Alert

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall generate an invalid aerodrome data alert within a specified timeframe, upon receiving the data.

SR-F.6.2.4 Discard Invalid Aerodrome Data

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid aerodrome data.

SR-F.6.2.5 Store Valid Aerodrome Data

Each time valid aerodrome data is received, the Dynamic Path Planning Automation System shall update the stored aerodrome data.

SR-F.6.3 Detect Aerodrome Data Impact

Each time it receives valid aerodrome data, the Dynamic Path Planning Automation System shall determine its impact on the active flight path within a specified timeframe, upon receiving the data.

SR-F.6.3.1 Analyze Aerodrome Data

Each time it receives valid aerodrome data, the Dynamic Path Planning Automation System shall determine if the active flight path needs to be updated in response to the new data.

SR-F.6.3.2 Build Aerodrome Induced Trigger

Each time it determines that the new aerodrome data may impact the active flight path, the Dynamic Path Planning Automation System shall generate a new flight path revision trigger.

SR-F.6.3.3 Confirm Unimpacted Flight Path

Each time it determines that the new aerodrome data does not impact the active flight path, the Dynamic Path Planning Automation System shall confirm continued viability of the active flight path.

SR-F.6.4 Receive Airspace Data

The Dynamic Path Planning Automation System shall asynchronously receive all airspace data sent by a selected airspace data service provider when the ground-to-air data communication link is operational.

SR-F.6.5 Validate Airspace Data

Each time it receives the airspace data, the Dynamic Path Planning Automation System shall validate it per the airspace data validation rules.

SR-F.6.5.1 Merge Airspace Data

The Dynamic Path Planning Automation System shall merge the received airspace data with the stored data to form a complete picture of the airspace operating environment.

SR-F.6.5.2 Apply Airspace Validation Rule

The Dynamic Path Planning Automation System shall apply the appropriate criteria to determine the validity of the received airspace data.

SR-F.6.5.3 Generate Invalid Airspace Data Alert

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall generate an invalid airspace data alert within a specified timeframe, upon receiving the data.

SR-F.6.5.4 Discard Invalid Airspace Data

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid airspace data.

SR-F.6.5.5 Store Valid Airspace Data

Each time valid airspace data is received, the Dynamic Path Planning Automation System shall update the stored airspace data.

SR-F.6.6 Detect Airspace Data Impact

Each time it receives valid airspace data, the Dynamic Path Planning Automation System shall determine its impact on the active flight path within a specified timeframe, upon receiving the data.

SR-F.6.6.1 Analyze Airspace Data

Each time it receives valid airspace data, the Dynamic Path Planning Automation System shall determine if the active flight path needs to be updated in response to the new data.

SR-F.6.6.2 Build Airspace Induced Trigger

Each time it determines that the new airspace data may impact the active flight path, the Dynamic Path Planning Automation System shall generate a new flight path revision trigger.

SR-F.6.6.3 Confirm Unimpacted Flight Path

Each time it determines that the new airspace data does not impact the active flight path, the Dynamic Path Planning Automation System shall confirm continued viability of the active flight path.

SR-F.6.7 Receive Atmosphere Data

The Dynamic Path Planning Automation System shall asynchronously receive all atmosphere data sent by a selected atmosphere data service provider when the ground-to-air data communication link is operational.

SR-F.6.8 Validate Atmosphere Data

Each time it receives the atmosphere data, the Dynamic Path Planning Automation System shall validate it per the atmosphere data validation rules.

SR-F.6.8.1 Merge Atmosphere Data

The Dynamic Path Planning Automation System shall merge the received

atmosphere data with the stored data to form a complete picture of the atmosphere operating environment.

SR-F.6.8.2 Apply Atmosphere Validation Rule

The Dynamic Path Planning Automation System shall apply the appropriate criteria to determine the validity of the received atmosphere data.

SR-F.6.8.3 Generate Invalid Atmosphere Data Alert

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall generate an invalid atmosphere data alert within a specified timeframe, upon receiving the data.

SR-F.6.8.4 Discard Invalid Atmosphere Data

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid atmosphere data.

SR-F.6.8.5 Store Valid Atmosphere Data

Each time valid atmosphere data is received, the Dynamic Path Planning Automation System shall update the stored atmosphere data.

SR-F.6.9 Detect Atmosphere Data Impact

Each time it receives valid atmosphere data, the Dynamic Path Planning Automation System shall determine its impact on the active flight path within a specified timeframe, upon receiving the data.

SR-F.6.9.1 Analyze Atmosphere Data

Each time it receives valid atmosphere data, the Dynamic Path Planning Automation System shall determine if the active flight path needs to be updated in response to the new data.

SR-F.6.9.2 Build Atmosphere Induced Trigger

Each time it determines that the new atmosphere data may impact the active flight path, the Dynamic Path Planning Automation System shall generate a new flight path revision trigger.

SR-F.6.9.3 Confirm Unimpacted Flight Path

Each time it determines that the new atmosphere data does not impact the active flight path, the Dynamic Path Planning Automation System shall confirm continued viability of the active flight path.

SR-F.6.10 Receive Obstacle Data

The Dynamic Path Planning Automation System shall asynchronously receive all obstacle data sent by a selected obstacle data service provider when the ground-to-air data communication link is operational.

SR-F.6.11 Validate Obstacle Data

Each time it receives the obstacle data, the Dynamic Path Planning Automation System shall validate it per the obstacle data validation rules.

SR-F.6.11.1 Merge Obstacle Data

The Dynamic Path Planning Automation System shall merge the received obstacle data with the stored data to form a complete picture of the obstacle operating environment.

SR-F.6.11.2 Apply Obstacle Validation Rule

The Dynamic Path Planning Automation System shall apply the appropriate criteria to determine the validity of the received obstacle data.

SR-F.6.11.3 Generate Invalid Obstacle Data Alert

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall generate an invalid obstacle data alert within a specified timeframe, upon receiving the data.

SR-F.6.11.4 Discard Invalid Obstacle Data

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid obstacle data.

SR-F.6.11.5 Store Valid Obstacle Data

Each time valid obstacle data is received, the Dynamic Path Planning Automation System shall update the stored obstacle data.

SR-F.6.12 Detect Obstacle Data Impact

Each time it receives valid obstacle data, the Dynamic Path Planning Automation System shall determine its impact on the active flight path within a specified timeframe, upon receiving the data.

SR-F.6.12.1 Analyze Obstacle Data

Each time it receives valid obstacle data, the Dynamic Path Planning Automation System shall determine if the active flight path needs to be updated in response to the new data.

SR-F.6.12.2 Build Obstacle Induced Trigger

Each time it determines that the new obstacle data may impact the active flight path, the Dynamic Path Planning Automation System shall generate a new flight path revision trigger.

SR-F.6.12.3 Confirm Unimpacted Flight Path

Each time it determines that the new obstacle data does not impact the active flight path, the Dynamic Path Planning Automation System shall confirm continued viability of the active flight path.

SR-F.6.13 Receive Traffic Data

The Dynamic Path Planning Automation System shall asynchronously receive all traffic data sent by a selected traffic data service provider when the ground-to-air data communication link is operational.

SR-F.6.14 Validate Traffic Data

Each time it receives the traffic data, the Dynamic Path Planning Automation System shall validate it per the traffic data validation rules.

SR-F.6.14.1 Merge Traffic Data

The Dynamic Path Planning Automation System shall merge the received traffic data with the stored data to form a complete picture of the traffic operating environment.

SR-F.6.14.2 Apply Traffic Validation Rule

The Dynamic Path Planning Automation System shall apply the appropriate criteria to determine the validity of the received traffic data.

SR-F.6.14.3 Generate Invalid Traffic Data Alert

Each time it deems the data invalid, the Dynamic Path Planning Automa-

- tion System shall generate an invalid traffic data alert within a specified timeframe, upon receiving the data.
- SR-F.6.14.4 Discard Invalid Traffic Data**
Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid traffic data.
- SR-F.6.14.5 Store Valid Traffic Data**
Each time valid traffic data is received, the Dynamic Path Planning Automation System shall update the stored traffic data.
- SR-F.6.15 Detect Traffic Data Impact**
Each time it receives valid traffic data, the Dynamic Path Planning Automation System shall determine its impact on the active flight path within a specified timeframe, upon receiving the data.
- SR-F.6.15.1 Analyze Traffic Data**
Each time it receives valid traffic data, the Dynamic Path Planning Automation System shall determine if the active flight path needs to be updated in response to the new data.
- SR-F.6.15.2 Build Traffic Induced Trigger**
Each time it determines that the new traffic data may impact the active flight path, the Dynamic Path Planning Automation System shall generate a new flight path revision trigger.
- SR-F.6.15.3 Confirm Unimpacted Flight Path**
Each time it determines that the new traffic data does not impact the active flight path, the Dynamic Path Planning Automation System shall confirm continued viability of the active flight path.
- SR-F.6.16 Update Flight Path**
The Dynamic Path Planning Automation System shall compute a viable candidate flight path each time it recognizes a need to update the flight path within a specified timeframe, upon receiving the flight-path-update trigger.
- SR-F.6.17 Report Successful Flight Path Adaptation**
The Dynamic Path Planning Automation System shall report every successful adaptation of the active flight path in response to the shared operating environment data within a specified timeframe, upon receiving the data, via the pilot graphical user interface.
- SR-F.6.18 Report Unimpacted Flight Path**
The Dynamic Path Planning Automation System shall report each time it identifies no impact of the shared operating environment data on the active flight path within a specified timeframe, upon receiving the data, via the pilot graphical user interface.
- SR-F.6.19 Report Failure**
The Dynamic Path Planning Automation System shall report every unsuccessful attempt to adapt the active flight path in response to the shared operating environment data within a specified timeframe, upon receiving the data, via the pilot graphical user interface.

SR-F.7 Adapt Flight Path to Mission Change

The Dynamic Path Planning Automation System shall update the active flight path to comply with a command to revise the flight mission within a specified timeframe, upon receiving the revised mission.

SR-F.7.1 Receive New Mission Command

The Dynamic Path Planning Automation System shall asynchronously receive all requests to execute a new flight mission.

SR-F.7.2 Validate Mission Data

Each time it receives the new mission data, the Dynamic Path Planning Automation System shall validate it per the mission data validation rules.

SR-F.7.2.1 Merge Mission Data

The Dynamic Path Planning Automation System shall merge the new mission data with the stored data to compile a complete set of mission specifications.

SR-F.7.2.2 Apply Mission Validation Rule

The Dynamic Path Planning Automation System shall apply the appropriate criteria to determine the validity of the new mission specifications.

SR-F.7.2.3 Generate Invalid Mission Alert

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall generate an invalid mission alert.

SR-F.7.2.4 Discard Invalid Mission

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid mission data.

SR-F.7.2.5 Store Valid Mission Data

Each time valid new mission data is executed successfully, the Dynamic Path Planning Automation System shall update the stored mission data.

SR-F.7.3 Ensure Operating Environment Availability

Each time it receives new mission data, the Dynamic Path Planning Automation System shall ensure that the necessary shared operating environment data is available.

SR-F.7.3.1 Identify Missing Operating Environment Data

Each time a valid mission data is received, the Dynamic Path Planning Automation System shall determine if the system contains all the operating environment data needed to execute the flight mission.

SR-F.7.3.2 Request Missing Aerodrome Data

If it detects missing operating environment data, the Dynamic Path Planning Automation System shall transmit an aerodrome data request containing the list of aerodrome identifiers to the selected aerodrome service provider if air-to-ground service communication link is operational.

SR-F.7.3.3 Request Missing Airspace Data

If it detects missing operating environment data, the Dynamic Path Planning Automation System shall transmit an airspace data request containing

the list of flight information region identifiers to the selected airspace service provider if air-to-ground service communication link is operational.

SR-F.7.3.4 Request Missing Atmosphere Data

If it detects missing operating environment data, the Dynamic Path Planning Automation System shall transmit an atmosphere data request containing the list of flight information region identifiers to the selected atmosphere service provider if air-to-ground service communication link is operational.

SR-F.7.3.5 Request Missing Obstacle Data

If it detects missing operating environment data, the Dynamic Path Planning Automation System shall transmit an obstacle data request containing the list of flight information region identifiers to the selected obstacle service provider if air-to-ground service communication link is operational.

SR-F.7.3.6 Request Missing Traffic Data

If it detects missing operating environment data, the Dynamic Path Planning Automation System shall transmit a traffic data request containing the list of flight information region identifiers to the selected traffic service provider if air-to-ground service communication link is operational.

SR-F.7.3.7 Receive Missing Aerodrome Data

The Dynamic Path Planning Automation System shall synchronously receive requested aerodrome data sent by aerodrome data service provider if ground-to-air service communication link is operational.

SR-F.7.3.8 Receive Missing Airspace Data

The Dynamic Path Planning Automation System shall synchronously receive requested airspace data sent by airspace data service provider if ground-to-air service communication link is operational.

SR-F.7.3.9 Receive Missing Atmosphere Data

The Dynamic Path Planning Automation System shall synchronously receive requested atmosphere data sent by atmosphere data service provider if ground-to-air service communication link is operational.

SR-F.7.3.10 Receive Missing Obstacle Data

The Dynamic Path Planning Automation System shall synchronously receive requested obstacle data sent by obstacle data service provider if ground-to-air service communication link is operational.

SR-F.7.3.11 Receive Missing Traffic Data

The Dynamic Path Planning Automation System shall synchronously receive requested traffic data sent by traffic data service provider if ground-to-air service communication link is operational.

SR-F.7.3.12 Validate Aerodrome Data

Each time it receives the aerodrome data, the Dynamic Path Planning Automation System shall validate it per the aerodrome data validation rules.

SR-F.7.3.13 Generate Invalid Aerodrome Data Alert

Each time it deems the data invalid, the Dynamic Path Planning Au-

- tomation System shall generate an invalid aerodrome data alert within a specified timeframe, upon receiving the data.
- SR-F.7.3.14 Discard Invalid Aerodrome Data**
Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid aerodrome data.
- SR-F.7.3.15 Store Valid Aerodrome Data**
Each time valid aerodrome data is received, the Dynamic Path Planning Automation System shall update the stored aerodrome data.
- SR-F.7.3.16 Validate Airspace Data**
Each time it receives the airspace data, the Dynamic Path Planning Automation System shall validate it per the airspace data validation rules.
- SR-F.7.3.17 Generate Invalid Airspace Data Alert**
Each time it deems the data invalid, the Dynamic Path Planning Automation System shall generate an invalid airspace data alert within a specified timeframe, upon receiving the data.
- SR-F.7.3.18 Discard Invalid Airspace Data**
Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid airspace data.
- SR-F.7.3.19 Store Valid Airspace Data**
Each time valid airspace data is received, the Dynamic Path Planning Automation System shall update the stored airspace data.
- SR-F.7.3.20 Validate Atmosphere Data**
Each time it receives the atmosphere data, the Dynamic Path Planning Automation System shall validate it per the atmosphere data validation rules.
- SR-F.7.3.21 Generate Invalid Atmosphere Data Alert**
Each time it deems the data invalid, the Dynamic Path Planning Automation System shall generate an invalid atmosphere data alert within a specified timeframe, upon receiving the data.
- SR-F.7.3.22 Discard Invalid Atmosphere Data**
Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid atmosphere data.
- SR-F.7.3.23 Store Valid Atmosphere Data**
Each time valid atmosphere data is received, the Dynamic Path Planning Automation System shall update the stored atmosphere data.
- SR-F.7.3.24 Validate Obstacle Data**
Each time it receives the obstacle data, the Dynamic Path Planning Automation System shall validate it per the obstacle data validation rules.
- SR-F.7.3.25 Generate Invalid Obstacle Data Alert**
Each time it deems the data invalid, the Dynamic Path Planning Automation System shall generate an invalid obstacle data alert within a specified timeframe, upon receiving the data.
- SR-F.7.3.26 Discard Invalid Obstacle Data**
Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid obstacle data.

- SR-F.7.3.27 Store Valid Obstacle Data**
Each time valid obstacle data is received, the Dynamic Path Planning Automation System shall update the stored obstacle data.
- SR-F.7.3.28 Validate Traffic Data**
Each time it receives the traffic data, the Dynamic Path Planning Automation System shall validate it per the traffic data validation rules.
- SR-F.7.3.29 Generate Invalid Traffic Data Alert**
Each time it deems the data invalid, the Dynamic Path Planning Automation System shall generate an invalid traffic data alert within a specified timeframe, upon receiving the data.
- SR-F.7.3.30 Discard Invalid Traffic Data**
Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid traffic data.
- SR-F.7.3.31 Store Valid Traffic Data**
Each time valid traffic data is received, the Dynamic Path Planning Automation System shall update the stored traffic data.
- SR-F.7.4 Detect New Mission Impact**
Each time it receives valid mission data, the Dynamic Path Planning Automation System shall determine its impact on the active flight path within a specified timeframe, upon receiving the data.
- SR-F.7.4.1 Confirm Unimpacted Flight Path**
Each time it determines that the new mission data does not impact the active flight path, the Dynamic Path Planning Automation System shall confirm continued viability of the active flight path.
- SR-F.7.4.2 Analyze New Flight Mission**
Each time a valid mission data and all necessary operating environment data are received, the Dynamic Path Planning Automation System shall determine if the active flight path needs to be updated in response to the new data.
- SR-F.7.4.3 Build Mission Change Trigger**
Each time it detects a need to update the active flight path in response to the new mission data, the Dynamic Path Planning Automation System shall generate a new flight path revision trigger.
- SR-F.7.5 Update Flight Path**
The Dynamic Path Planning Automation System shall compute a viable candidate flight path each time it recognizes a need to update the flight path, within a specified timeframe, upon receiving the flight-path-update trigger.
- SR-F.7.6 Report Successful Flight Path Adaptation**
The Dynamic Path Planning Automation System shall report every successful adaptation of the active flight path in response to the shared operating environment data within a specified timeframe, upon receiving the data via the pilot graphical user interface.
- SR-F.7.7 Report Unimpacted Flight Path**
The Dynamic Path Planning Automation System shall report each time it

identifies no impact of the shared operating environment data on the active flight path within a specified timeframe, upon receiving the data via the pilot graphical user interface.

SR-F.7.8 Report Failure

The Dynamic Path Planning Automation System shall report every unsuccessful attempt to adapt the active flight path in response to the shared operating environment data within a specified timeframe, upon receiving the data via the pilot graphical user interface.

SR-F.8 Adapt Flight Path to Sensed Information

The Dynamic Path Planning Automation System shall adapt the active flight path to aircraft sensors data received via aircraft network, within a specified timeframe, upon receiving the sensed data.

SR-F.8.1 Receive Avionics Data

The Dynamic Path Planning Automation System shall asynchronously receive avionics data each time it is sent by the aircraft via the aircraft network interface.

SR-F.8.2 Validate Avionics Data

Upon receipt of new avionics data, the Dynamic Path Planning Automation System shall validate the data per the avionics data validation rules specifying the expected data frequency, latency, format, and completeness.

SR-F.8.2.1 Apply Avionics Validation Rule

The Dynamic Path Planning Automation System shall apply the appropriate criteria to determine the validity of the received avionics data.

SR-F.8.2.2 Generate Invalid Avionics Data Alert

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall generate an alert identifying invalid avionics data elements within a specified timeframe, upon receiving the data.

SR-F.8.2.3 Discard Invalid Avionics Data

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid avionics data.

SR-F.8.2.4 Store Valid Avionics Data

Each time it deems the data valid, the Dynamic Path Planning Automation System shall update the stored avionics data.

SR-F.8.3 Process Avionics Data

Each time it receives valid avionics data, the Dynamic Path Planning Automation System shall determine the type of sensed data.

SR-F.8.3.1 Update Aircraft State

Each time it receives valid avionics data, the Dynamic Path Planning Automation System shall update the stored aircraft state.

SR-F.8.3.2 Identify Atmosphere Data

Each time it receives valid avionics data from aircraft atmosphere sensors,

the Dynamic Path Planning Automation System shall compile the atmosphere data.

SR-F.8.3.3 Identify Obstacle Data

Each time it receives valid avionics data from aircraft obstacle sensors, the Dynamic Path Planning Automation System shall compile the obstacle data.

SR-F.8.3.4 Identify Traffic Data

Each time it receives valid avionics data from aircraft traffic sensors, the Dynamic Path Planning Automation System shall compile the traffic data.

SR-F.8.4 Validate Sensed Atmosphere Data

Each time it receives sensed atmosphere data, the Dynamic Path Planning Automation System shall validate it per the atmosphere data validation rules.

SR-F.8.4.1 Merge Atmosphere Data

The Dynamic Path Planning Automation System shall merge the sensed atmosphere data with the stored data to form a complete picture of the atmosphere operating environment.

SR-F.8.4.2 Apply Atmosphere Validation Rule

The Dynamic Path Planning Automation System shall apply the appropriate criteria to determine the validity of the observed atmosphere data.

SR-F.8.4.3 Generate Invalid Atmosphere Data Alert

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall generate an invalid atmosphere data alert within a specified timeframe, upon receiving the data.

SR-F.8.4.4 Discard Invalid Atmosphere Data

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid atmosphere data.

SR-F.8.4.5 Store Valid Atmosphere Data

Each time valid atmosphere data is received, the Dynamic Path Planning Automation System shall update the stored atmosphere data.

SR-F.8.5 Validate Sensed Obstacle Data

Each time it receives sensed obstacle data, the Dynamic Path Planning Automation System shall validate it per the obstacle data validation rules.

SR-F.8.5.1 Merge Obstacle Data

The Dynamic Path Planning Automation System shall merge the sensed obstacle data with the stored data to form a complete picture of the obstacle operating environment.

SR-F.8.5.2 Apply Obstacle Validation Rule

The Dynamic Path Planning Automation System shall apply the appropriate criteria to determine the validity of the observed obstacle data.

SR-F.8.5.3 Generate Invalid Obstacle Data Alert

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall generate an invalid obstacle data alert within a specified timeframe, upon receiving the data.

- SR-F.8.5.4 Discard Invalid Obstacle Data
Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid obstacle data.
- SR-F.8.5.5 Store Valid Obstacle Data
Each time valid obstacle data is received, the Dynamic Path Planning Automation System shall update the stored obstacle data.
- SR-F.8.6 Validate Sensed Traffic Data
Each time it receives sensed traffic data, the Dynamic Path Planning Automation System shall validate it per the traffic data validation rules.
 - SR-F.8.6.1 Merge Traffic Data
The Dynamic Path Planning Automation System shall merge the sensed traffic data with the stored data to form a complete picture of the traffic operating environment.
 - SR-F.8.6.2 Apply Traffic Validation Rule
The Dynamic Path Planning Automation System shall apply the appropriate criteria to determine the validity of the observed traffic data.
 - SR-F.8.6.3 Generate Invalid Traffic Data Alert
Each time it deems the data invalid, the Dynamic Path Planning Automation System shall generate an invalid traffic data alert within a specified timeframe, upon receiving the data.
 - SR-F.8.6.4 Discard Invalid Traffic Data
Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid traffic data.
 - SR-F.8.6.5 Store Valid Traffic Data
Each time valid traffic data is received, the Dynamic Path Planning Automation System shall update the stored traffic data.
- SR-F.8.7 Detect Sensed Atmosphere Data Impact
Each time it receives valid sensed atmosphere data, the Dynamic Path Planning Automation System shall determine its impact on the active flight path within a specified timeframe, upon receiving the data.
 - SR-F.8.7.1 Analyze Atmosphere Data
Each time it receives valid atmosphere data, the Dynamic Path Planning Automation System shall determine if the active flight path needs to be updated in response to the sensed data.
 - SR-F.8.7.2 Build Aerodrome Induced Trigger
Each time it detects a need to update the active flight path in response to the new atmosphere data, the Dynamic Path Planning Automation System shall generate a new flight path revision trigger.
 - SR-F.8.7.3 Confirm Nonimpact of Aerodrome Data
Each time it determines the active flight path to be unaffected by the new atmosphere data, the Dynamic Path Planning Automation System shall generate a message indicating continued validity of the active flight path.
- SR-F.8.8 Detect Sensed Obstacle Data Impact
Each time it receives valid sensed obstacle data, the Dynamic Path Planning

Automation System shall determine its impact on the active flight path within a specified timeframe, upon receiving the data.

SR-F.8.8.1 Analyze Obstacle Data

Each time it receives valid obstacle data, the Dynamic Path Planning Automation System shall determine if the active flight path needs to be updated in response to the sensed data.

SR-F.8.8.2 Build Obstacle Induced Trigger

Each time it detects a need to update the active flight path in response to the new obstacle data, the Dynamic Path Planning Automation System shall generate a new flight path revision trigger.

SR-F.8.8.3 Confirm Nonimpact of Obstacle Data

Each time it determines the active flight path to be unaffected by the new obstacle data, the Dynamic Path Planning Automation System shall generate a message indicating continued validity of the active flight path.

SR-F.8.9 Detect Sensed Traffic Data Impact

Each time it receives valid sensed traffic data, the Dynamic Path Planning Automation System shall determine its impact on the active flight path within a specified timeframe, upon receiving the data.

SR-F.8.9.1 Analyze Traffic Data

Each time the traffic data is updated, the Dynamic Path Planning Automation System shall determine if the active flight path needs to be updated in response to the sensed data.

SR-F.8.9.2 Build Traffic Induced Trigger

Each time it detects a need to update the active flight path in response to the new traffic data, the Dynamic Path Planning Automation System shall generate a new flight path revision trigger.

SR-F.8.9.3 Confirm Nonimpact of Traffic Data

Each time it determines the active flight path to be unaffected by the new traffic data, the Dynamic Path Planning Automation System shall generate a message indicating continued validity of the active flight path.

SR-F.8.10 Detect Trajectory Conformance Change

The Dynamic Path Planning Automation System shall update the active flight path when the aircraft is out of conformance with the planned flight trajectory, in less than a specified time of identifying the change in conformance.

SR-F.8.10.1 Determine Aircraft Position Conformance

Each time a valid new aircraft state is received, the Dynamic Path Planning Automation System shall determine if the aircraft position is in conformance with the planned trajectory.

SR-F.8.10.2 Determine Aircraft Velocity Conformance

Each time a valid new aircraft state is received, the Dynamic Path Planning Automation System shall determine if the aircraft velocity is in conformance with the planned trajectory.

SR-F.8.10.3 Determine Mission Constraint Conformance

Each time a valid new aircraft state is received and it is determined to be

in conformance with the planned trajectory, the Dynamic Path Planning Automation System shall determine if the aircraft state is in compliance with all relevant mission constraints.

SR-F.8.10.4 Alert of Flight Trajectory Conformance Change

Each time it detects a change in flight trajectory conformance, the Dynamic Path Planning Automation System shall generate an alert identifying the element whose conformance to the planned flight trajectory has changed.

SR-F.8.10.5 Build Trajectory Conformance Induced Trigger

Each time it detects the aircraft to be out of conformance with the active flight trajectory, the Dynamic Path Planning Automation System shall generate a new flight path revision trigger.

SR-F.8.11 Receive Propulsion Data

The Dynamic Path Planning Automation System shall asynchronously receive propulsion data each time it is sent by the aircraft via the aircraft network interface.

SR-F.8.12 Validate Propulsion Data

Upon receipt of new propulsion data, the Dynamic Path Planning Automation System shall validate the data per the propulsion data validation rules specifying the expected data frequency, latency, format, and completeness.

SR-F.8.12.1 Apply Propulsion Validation Rule

The Dynamic Path Planning Automation System shall apply the appropriate criteria to determine the validity of the received propulsion data.

SR-F.8.12.2 Generate Invalid Propulsion Data Alert

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall generate an alert identifying invalid propulsion data elements, within a specified timeframe, upon receiving the data.

SR-F.8.12.3 Discard Invalid Propulsion Data

Each time it deems the data invalid, the Dynamic Path Planning Automation System shall discard the invalid propulsion data.

SR-F.8.12.4 Store Valid Propulsion Data

Each time it deems the data valid, the Dynamic Path Planning Automation System shall update the stored propulsion data.

SR-F.8.13 Detect Propulsion Data Impact

Each time an onboard emergency is reported, the Dynamic Path Planning Automation System shall determine its impact on the active flight path, within a specified timeframe, upon receiving the data.

SR-F.8.13.1 Determine Aircraft Energy Conformance

Each time a valid new propulsion data is received, the Dynamic Path Planning Automation System shall determine if the aircraft energy and energy consumption rate are in conformance with the planned energy profile.

SR-F.8.13.2 Alert of Energy Conformance Change

Each time it detects a change in energy consumption conformance, the Dynamic Path Planning Automation System shall generate an alert iden-

tifying the element whose conformance to the planned energy profile has changed.

SR-F.8.13.3 Evaluate Energy Contingency Criterion

If it determines the aircraft to be out of conformance with respect to the planned energy profile, the Dynamic Path Planning Automation System shall determine if the aircraft propulsion state results in a potential flight path contingency.

SR-F.8.13.4 Alert of Flight Path Contingency

If the aircraft propulsion data meets the flight path contingency criterion, the Dynamic Path Planning Automation System shall generate a flight path contingency alert.

SR-F.8.13.5 Build Energy Conformance Induced Trigger

Each time it detects the aircraft to be out of conformance with the active flight trajectory and a flight path contingency has not been detected, the Dynamic Path Planning Automation System shall generate a new flight path revision trigger.

SR-F.8.14 Refresh Flight Path at Specific Events

The Dynamic Path Planning Automation System shall update the active flight path at a set of specified flight events during the flight, within a specified timeframe, upon receiving the event notification.

SR-F.8.14.1 Identify Flight Trajectory Event

Each time the updated aircraft state is determined to be in conformance with the planned flight trajectory, the Dynamic Path Planning Automation System shall determine if the aircraft state satisfies the criterion of one or more of a predefined set of flight trajectory events.

SR-F.8.14.2 Identify Flight Energy Event

Each time the updated aircraft state is determined to be in conformance with the planned flight trajectory and energy profile, the Dynamic Path Planning Automation System shall determine if the aircraft state satisfies the criterion of one or more of a predefined set of aircraft propulsion events.

SR-F.8.14.3 Build Flight Event Induced Trigger

Each time it detects a need to update the active flight path due to the occurrence of one or more of the predefined set of flight events, the Dynamic Path Planning Automation System shall generate a new flight path revision trigger.

SR-F.8.15 Mitigate Anticipated Contingency

Each time the aircraft state is updated, the Dynamic Path Planning Automation System shall ensure continued mitigation of anticipated contingencies.

SR-F.8.15.1 Validate Contingency Mitigation Plan

Each time the aircraft state is updated, the Dynamic Path Planning Automation System shall determine if the planned mitigation of anticipated contingencies is still valid.

SR-F.8.15.2 Refresh Contingency Mitigation Plan

Each time it determines that the planned contingency mitigation option is

invalid, the Dynamic Path Planning Automation System shall select a new mitigation plan per specified operator preference.

SR-F.8.15.3 Identify Flight Path Contingency

The Dynamic Path Planning Automation System shall declare a flight path contingency if none of the operator preferred mitigation options are viable.

SR-F.8.16 Mitigate Flight Path Contingency

The Dynamic Path Planning Automation System shall mitigate the impact of all identified contingencies on successful execution of the flight mission, in less than a specified time of contingency identification.

SR-F.8.17 Update Flight Path

The Dynamic Path Planning Automation System shall compute a viable candidate flight path each time it recognizes a need to update the flight path, within a specified timeframe, upon receiving the flight-path-update trigger.

SR-F.8.18 Report Successful Flight Path Adaptation

The Dynamic Path Planning Automation System shall report every successful adaptation of the active flight path in response to the sensed data, within a specified timeframe, upon receiving the data via the pilot graphical user interface.

SR-F.8.19 Report Failure

The Dynamic Path Planning Automation System shall report every unsuccessful attempt to adapt the active flight path in response to the sensed data, within a specified timeframe, upon receiving the data via the pilot graphical user interface.

SR-F.9 Adapt Flight Path to System Configuration Change

The Dynamic Path Planning Automation System shall update the flight path in response to relevant changes to the system configuration, in less than a specified time of configuration change.

SR-F.9.1 Receive Configuration Change Request

The Dynamic Path Planning Automation System shall asynchronously receive a request to provide system configuration parameters from the pilot via the pilot graphical user interface.

SR-F.9.2 Provide Mutable Configuration Parameters

Each time it receives a request to provide configuration parameters, the Dynamic Path Planning Automation System shall provide the list of mutable configuration parameters to the pilot via the pilot graphical user interface.

SR-F.9.3 Receive New Configuration Parameters

The Dynamic Path Planning Automation System shall asynchronously receive configuration parameter values from the pilot via the pilot graphical user interface.

SR-F.9.4 Validate Configuration Data

Each time it receives the configuration parameters, the Dynamic Path Planning Automation System shall validate those per configuration validation rules.

- SR-F.9.4.1 Generate Invalid Configuration Data Alert
Each time it deems a configuration parameter value invalid, the Dynamic Path Planning Automation System shall generate an invalid configuration parameter alert.
- SR-F.9.4.2 Discard Invalid Configuration Data
Each time it deems a configuration parameter value invalid, the Dynamic Path Planning Automation System shall discard the invalid configuration parameter.
- SR-F.9.4.3 Store Valid Configuration Data
Each time valid configuration parameter values are received, the Dynamic Path Planning Automation System shall update the stored configuration parameter values.
- SR-F.9.5 Detect Configuration Change Impact
Each time it receives valid configuration parameters, the Dynamic Path Planning Automation System shall determine its impact on the active flight path within a specified timeframe, upon receiving the data.
 - SR-F.9.5.1 Analyze Configuration Data
Each time it receives valid configuration parameters, the Dynamic Path Planning Automation System shall determine if the active flight path needs to be updated in response to the new configuration parameter values.
 - SR-F.9.5.2 Build Configuration Change Trigger
Each time it detects a need to update the active flight path in response to the new configuration parameters, the Dynamic Path Planning Automation System shall generate a new flight path revision trigger.
 - SR-F.9.5.3 Change System Operating Mode
If it determines the new configuration parameter is to command a different operating mode, then the Dynamic Path Planning Automation System shall change the system operating mode to its commanded value.
- SR-F.9.6 Update Flight Path
The Dynamic Path Planning Automation System shall compute a viable candidate flight path each time it recognizes a need to update the flight path, within a specified timeframe, upon receiving a flight-path-update trigger.
- SR-F.9.7 Report Successful Flight Path Adaptation
The Dynamic Path Planning Automation System shall report every successful adaptation of the active flight path in response to a change in system configuration data, within a specified timeframe, upon receiving the data via the pilot graphical user interface.
- SR-F.9.8 Report Failure
The Dynamic Path Planning Automation System shall report every unsuccessful attempt to adapt the active flight path in response to a change in system configuration data, within a specified timeframe, upon receiving the data via the pilot graphical user interface.

SR-F.10 Assess Flight Path Change Proposal

The Dynamic Path Planning Automation System shall evaluate the viability of all user proposed flight path changes, within a specified timeframe, upon receiving the proposal.

SR-F.10.1 Receive Flight Path Evaluation Request

The Dynamic Path Planning Automation System shall asynchronously receive flight path evaluation proposal from authorized users via an applicable interface.

SR-F.10.2 Validate Proposed Change Request

Each time it receives a proposal to evaluate a flight path change, the Dynamic Path Planning Automation System shall validate it per the proposal validation rules.

SR-F.10.2.1 Identify Proposer

The Dynamic Path Planning Automation System shall identify the source of each proposal.

SR-F.10.2.2 Apply Proposal Validation Rule

The Dynamic Path Planning Automation System shall apply the appropriate criteria to determine the validity of the flight path proposal based on the proposer authorization.

SR-F.10.2.3 Compile Proposed Flight Path Input

The Dynamic Path Planning Automation System shall compile flight path construction input based on the proposed flight path change.

SR-F.10.3 Build Proposed Flight Path

Each time it receives a valid proposal, the Dynamic Path Planning Automation System shall build a corresponding proposed flight path.

SR-F.10.3.1 Create Proposed Flight Path

The Dynamic Path Planning Automation System shall create a flight path based on the proposed flight path construction input.

SR-F.10.3.2 Evaluate Proposed Flight Path

Each time it creates a feasible flight path, the Dynamic Path Planning Automation System shall evaluate its flight path qualities.

SR-F.10.3.3 Coordinate Proposed Flight Path

Each time it creates a feasible flight path with acceptable flight path qualities, the Dynamic Path Planning Automation System shall coordinate the flight path with the selected entities.

SR-F.10.3.4 Receive Flight Path Coordination Response

The Dynamic Path Planning Automation System shall asynchronously receive the response to flight path coordination request.

SR-F.10.3.5 Analyze Flight Path Coordination Response

The Dynamic Path Planning Automation System shall analyze the flight path coordination response.

SR-F.10.3.6 Store Updated Flight Path Information

If the flight path coordination is successful, then the Dynamic Path Planning Automation System shall set the proposed flight path as the new baseline flight path for all future adaptations.

SR-F.10.4 Respond to Flight Path Evaluation Proposal

The Dynamic Path Planning Automation System shall respond to every flight path evaluation request.

SR-F.10.4.1 Reject Invalid Flight Path Proposal

The Dynamic Path Planning Automation System shall reject the proposal each time it deems the flight path evaluation request to be invalid.

SR-F.10.4.2 Reject Infeasible Flight Path Proposal

The Dynamic Path Planning Automation System shall reject the proposal each time it determines that a feasible flight path could not be computed for the proposed flight path construction input.

SR-F.10.4.3 Reject Unacceptable Flight Path Proposal

The Dynamic Path Planning Automation System shall reject the proposal each time it determines the qualities of the proposed flight path to be unacceptable.

SR-F.10.4.4 Reject Uncoordinated Flight Path Proposal

The Dynamic Path Planning Automation System shall reject the proposal each time it determines that the computed flight path could not be coordinated with the selected entities.

SR-F.10.4.5 Accept Proposed Flight Path

The Dynamic Path Planning Automation System shall accept the proposal upon successful coordination of the flight path with the selected entities..

SR-F.10.4.6 Compile Proposal Response

The Dynamic Path Planning Automation System shall tailor the response to each flight path evaluation request based on the proposer.

SR-F.10.4.7 Transmit Proposal Response

The Dynamic Path Planning Automation System shall transmit the tailored response to flight path evaluation proposal to appropriate proposer via an applicable interface.

SR-F.10.5 Report Successful Flight Path Adaptation

The Dynamic Path Planning Automation System shall report every successful adaptation of the active flight path in response to a proposed flight path change, within a specified timeframe, upon receiving the flight path evaluation proposal.

SR-F.10.6 Report Failure to Evaluate Proposal

The Dynamic Path Planning Automation System shall report every unsuccessful attempt to assess a proposed flight path evaluation request, within a specified timeframe, upon receiving the flight path evaluation proposal

SR-F.11 Refresh Flight Path Periodically

The Dynamic Path Planning Automation System shall refresh the active flight path periodically in search of a more optimal and/or flexible path than the current path at a predefined duration since the active flight path creation.

SR-F.11.1 Update Latest Update Time

The Dynamic Path Planning Automation System shall keep track of the time at which the active flight path is updated.

SR-F.11.2 Compute Next Update Time

Each time the active flight path is updated, the Dynamic Path Planning Automation System shall set the next flight path refresh time based upon the latest update time plus the specified refresh period.

SR-F.11.3 Build Flight Path Refresh Trigger

The Dynamic Path Planning Automation System shall generate a new flight path revision trigger when the current time is greater than or equal to the flight path refresh time.

SR-F.11.4 Update Flight Path

The Dynamic Path Planning Automation System shall compute a viable candidate flight path each time it recognizes a need to update the flight path, within a specified timeframe, upon receiving the flight-path-update trigger.

SR-F.11.5 Report Successful Flight Path Adaptation

The Dynamic Path Planning Automation System shall report every successful refresh of the active flight path, in less than a specified time of the computed refresh time value.

SR-F.11.6 Report Failure to Refresh

The Dynamic Path Planning Automation System shall report every unsuccessful attempt to refresh the active flight path, in less than a specified time of the computed refresh time value.

SR-F.12 Coordinate Candidate Flight Path

The Dynamic Path Planning Automation System shall coordinate every successfully built candidate flight path with the selected entities.

SR-F.12.1 Receive Candidate Flight Path

The Dynamic Path Planning Automation System shall receive the flight path to be coordinated with the selected entities.

SR-F.12.2 Establish Pilot Concurrence

Each time it determines the need to coordinate a flight path, the Dynamic Path Planning Automation System shall seek the pilot's concurrence if so configured.

SR-F.12.2.1 Build Pilot Concurrence Request

The Dynamic Path Planning Automation System shall build a request for pilot concurrence each time a computed flight path is received and the system is in nonautonomous mode.

SR-F.12.2.2 Request Pilot Concurrence

The Dynamic Path Planning Automation System shall transmit pilot concurrence request to the pilot via pilot graphical user interface.

SR-F.12.2.3 Receive Pilot Concurrence Response

The Dynamic Path Planning Automation System shall asynchronously receive pilot concurrence response from the pilot via pilot graphical user interface if the system is in nonautonomous mode.

- SR-F.12.2.4 Analyze Pilot Concurrence**
Each time it receives the pilot concurrence response, the Dynamic Path Planning Automation System shall determine if the pilot concurs with the computed flight path.
- SR-F.12.2.5 Alert Pilot Nonconcurrence**
The Dynamic Path Planning Automation System shall alert of pilot non-concurrence each time the pilot does not concur with the computed flight path.
- SR-F.12.3 Establish Aerodrome Coordination**
Each time the pilot concurs with the flight path, the Dynamic Path Planning Automation System shall coordinate the flight path with the specified aerodrome coordinator.
- SR-F.12.3.1 Build Aerodrome Coordination Request**
The Dynamic Path Planning Automation System shall build a request to coordinate the flight path with the selected aerodrome coordinator each time a computed flight path is received and the pilot concurrence is successful.
- SR-F.12.3.2 Request Aerodrome Coordination**
The Dynamic Path Planning Automation System shall transmit aerodrome coordination request to the selected aerodrome coordinator via air-to-ground coordination link.
- SR-F.12.3.3 Receive Aerodrome Coordination Response**
The Dynamic Path Planning Automation System shall asynchronously receive aerodrome coordination response from the selected aerodrome coordinator via ground-to-air coordination link.
- SR-F.12.3.4 Analyze Aerodrome Coordination**
Each time it receives the aerodrome coordination response, the Dynamic Path Planning Automation System shall determine if the computed flight path is harmonized with the relevant aerodromes.
- SR-F.12.3.5 Alert Aerodrome Coordination Failure**
The Dynamic Path Planning Automation System shall alert of uncoordinated flight path each time the aerodrome coordination is unsuccessful.
- SR-F.12.4 Establish Airspace Coordination**
Each time the pilot concurs with the flight path, the Dynamic Path Planning Automation System shall coordinate the flight path with the specified airspace coordinator.
- SR-F.12.4.1 Build Airspace Coordination Request**
The Dynamic Path Planning Automation System shall build a request to coordinate the flight path with the selected airspace coordinator each time a computed flight path is received and the pilot concurrence is successful.
- SR-F.12.4.2 Request Airspace Coordination**
The Dynamic Path Planning Automation System shall transmit airspace coordination request to the selected airspace coordinator via air-to-ground coordination link.

- SR-F.12.4.3 Receive Airspace Coordination Response**
The Dynamic Path Planning Automation System shall asynchronously receive airspace coordination response from the selected airspace coordinator via ground-to-air coordination link.
- SR-F.12.4.4 Analyze Airspace Coordination**
Each time it receives the airspace coordination response, the Dynamic Path Planning Automation System shall determine if the computed flight path is harmonized with the relevant airspace.
- SR-F.12.4.5 Alert Airspace Coordination Failure**
The Dynamic Path Planning Automation System shall alert of uncoordinated flight path each time the airspace coordination is unsuccessful.
- SR-F.12.5 Establish Pilot Confirmation**
Each time the flight path is coordinated successfully with the aerodrome and airspace coordinators, the Dynamic Path Planning Automation System shall seek pilot's confirmation prior to flight path execution, if so configured.
- SR-F.12.5.1 Build Pilot Confirmation Request**
The Dynamic Path Planning Automation System shall build a request for pilot confirmation of the flight path each time the computed flight path is coordinated successfully with relevant aerodromes and airspace.
- SR-F.12.5.2 Request Pilot Confirmation**
The Dynamic Path Planning Automation System shall transmit pilot confirmation request to the pilot via pilot graphical user interface if the system is in nonautonomous mode.
- SR-F.12.5.3 Receive Pilot Confirmation Response**
The Dynamic Path Planning Automation System shall asynchronously receive pilot confirmation/acceptance response from the pilot via pilot graphical user interface if the system is in nonautonomous mode.
- SR-F.12.5.4 Analyze Pilot Confirmation**
Each time it receives the pilot confirmation response, the Dynamic Path Planning Automation System shall determine if the pilot confirms the computed flight path.
- SR-F.12.5.5 Alert Pilot Nonconfirmation**
The Dynamic Path Planning Automation System shall alert of unconfirmed flight path each time the pilot does not confirm/accept the computed flight path.
- SR-F.12.6 Establish Flight Path Execution**
Each time the pilot confirms the flight path, the Dynamic Path Planning Automation System shall coordinate the flight path with the aircraft flight path execution system.
- SR-F.12.6.1 Build Flight Path Execution Request**
The Dynamic Path Planning Automation System shall build a request to execute the flight path each time the computed flight path is confirmed/accepted by the pilot.

- SR-F.12.6.2 Request Flight Path Execution**
The Dynamic Path Planning Automation System shall transmit flight path execution request to the aircraft guidance system via the avionics bus interface.
- SR-F.12.6.3 Receive Flight Path Execution Response**
The Dynamic Path Planning Automation System shall asynchronously receive flight path execution response from the aircraft guidance system via the avionics bus interface.
- SR-F.12.6.4 Analyze Flight Path Execution**
Each time it receives the flight path execution response, the Dynamic Path Planning Automation System shall determine if the aircraft guidance system has executed the computed flight path.
- SR-F.12.6.5 Alert Flight Path Execution Failure**
The Dynamic Path Planning Automation System shall alert of nonexecutable flight path each time the aircraft fails to execute the computed flight path.
- SR-F.12.7 Establish Flight Path Transmission**
Each time the aircraft executes the flight path successfully, the Dynamic Path Planning Automation System shall coordinate the flight path with the aircraft communication system.
- SR-F.12.7.1 Build Flight Path Transmission Request**
The Dynamic Path Planning Automation System shall build a request to transmit the flight path each time the computed flight path is executed successfully by the aircraft guidance system.
- SR-F.12.7.2 Request Flight Path Transmission**
The Dynamic Path Planning Automation System shall provide flight path transmission request to the aircraft communication system via the avionics bus interface.
- SR-F.12.7.3 Receive Flight Path Transmission Response**
The Dynamic Path Planning Automation System shall asynchronously receive flight path transmission response from the aircraft communication system via the avionics bus interface.
- SR-F.12.7.4 Analyze Flight Path Transmission**
Each time it receives the flight path transmission response, the Dynamic Path Planning Automation System shall determine if the aircraft communication system has transmitted the computed flight path.
- SR-F.12.7.5 Alert Flight Path Transmission Failure**
The Dynamic Path Planning Automation System shall alert of flight path transmission failure each time the aircraft communication system fails to transmit the computed flight path.
- SR-F.12.8 Provide Flight Path Coordination Status**
The Dynamic Path Planning Automation System shall confirm successful coordination of the candidate flight path with the selected entities.

SR-F.13 Update Candidate Flight Path

The Dynamic Path Planning Automation System shall compute a viable candidate flight path each time it recognizes a need to update the flight path, within a specified timeframe, upon receiving the flight-path-update trigger.

SR-F.13.1 Process Flight Path Trigger

Each time it identifies a new flight path revision trigger, the Dynamic Path Planning Automation System shall collate it with pending triggers to determine the highest priority consolidated flight path revision trigger, in less than a specified time of trigger identification.

SR-F.13.1.1 Collate Flight Path Triggers

The Dynamic Path Planning Automation System shall collate identified trigger with pending triggers to determine the highest priority consolidated flight path revision trigger.

SR-F.13.1.2 Compile Consolidated Flight Path Trigger

The Dynamic Path Planning Automation System shall consolidate elements of the highest priority flight path revision trigger with the latest data elements of all other pending triggers.

SR-F.13.1.3 Extract Candidate Flight Path Input

The Dynamic Path Planning Automation System shall extract the candidate flight path construction input from the consolidated flight path revision trigger.

SR-F.13.2 Create Candidate Flight Path

Each time a candidate flight path construction input is generated, the Dynamic Path Planning Automation System shall create a corresponding candidate flight path.

SR-F.13.2.1 Construct Candidate Flight Path

The Dynamic Path Planning Automation System shall construct a flight path for a specified flight path construction input.

SR-F.13.2.2 Determine Candidate Flight Path Feasibility

The Dynamic Path Planning Automation System shall determine if the computed candidate flight path is feasible with respect to the aircraft and static operating environment constraints.

SR-F.13.2.3 Report Infeasible Flight Path

The Dynamic Path Planning Automation System shall alert each time it determines that a feasible flight path could not be created for the candidate flight path construction input.

SR-F.13.2.4 Report Feasible Flight Path Creation

The Dynamic Path Planning Automation System shall report successful creation of a feasible candidate flight path.

SR-F.13.3 Evaluate Candidate Flight Path

Each time it creates a feasible candidate flight path, the Dynamic Path Planning Automation System shall evaluate its flight path qualities.

SR-F.13.3.1 Evaluate Candidate Flight Path Safety

The Dynamic Path Planning Automation System shall detect all conflicts

between the candidate flight path and known operating environment hazards each time a feasible flight path is constructed.

SR-F.13.3.2 Report Hazardous Flight Path

The Dynamic Path Planning Automation System shall alert each time it detects a conflict between the candidate flight path and one or more operating environment hazards.

SR-F.13.3.3 Evaluate Candidate Flight Path Optimality

The Dynamic Path Planning Automation System shall compute the optimization performance index of the computed conflict-free candidate flight path.

SR-F.13.3.4 Report Suboptimal Flight Path

The Dynamic Path Planning Automation System shall alert each time it detects the candidate flight path to be less optimal than the active flight path.

SR-F.13.3.5 Report Acceptable Flight Path

The Dynamic Path Planning Automation System shall report acceptable flight path qualities of the candidate flight path.

SR-F.13.4 Revise Candidate Flight Path

The Dynamic Path Planning Automation System shall revise each revisable candidate flight path that possesses unacceptable flight path qualities.

SR-F.13.4.1 Determine Flight Path Revisability

The Dynamic Path Planning Automation System shall determine if a candidate flight path with unacceptable flight path qualities can be revised, based on the type of flight path revision trigger.

SR-F.13.4.2 Confirm Unacceptable Flight Path

If it determines that a candidate flight path with unacceptable flight path qualities cannot be revised, then the Dynamic Path Planning Automation System shall report unacceptability of the candidate flight path.

SR-F.13.4.3 Update Candidate Flight Path Construction Input

If it determines that a candidate flight path with unacceptable flight path qualities can be revised, then the Dynamic Path Planning Automation System shall apply flight path revision rules to compile a new candidate flight path construction input.

SR-F.13.4.4 Relax Operator Guidelines

The Dynamic Path Planning Automation System shall relax operator guidelines, when viable, to revise the candidate flight path input that resulted in a flight path with unacceptable qualities.

SR-F.13.4.5 Relax Customer Preferences

The Dynamic Path Planning Automation System shall relax customer preferences, when viable, to revise the candidate flight path input that resulted in a flight path with unacceptable qualities.

SR-F.13.4.6 Relax Planning Parameters

The Dynamic Path Planning Automation System shall relax flight planning parameters, when viable, to revise the candidate flight path input that resulted in a flight path with unacceptable qualities.

- SR-F.13.4.7 Relax Mission Constraints

The Dynamic Path Planning Automation System shall relax flight mission constraints, when viable, to revise the candidate flight path input that resulted in a flight path with unacceptable qualities.
- SR-F.13.4.8 Relax Hazard Avoidance

The Dynamic Path Planning Automation System shall relax hazard avoidance criteria, when viable, to revise the candidate flight path input that resulted in a flight path with unacceptable qualities.
- SR-F.13.4.9 Relax Mission Objective

The Dynamic Path Planning Automation System shall relax flight mission objective, when viable, to revise the candidate flight path input that resulted in a flight path with unacceptable qualities.
- SR-F.13.4.10 Generate Unavailable Flight Path Construction Input

If it has exhausted the list of specified flight path revision rules, then the Dynamic Path Planning Automation System shall report its inability to revise the candidate flight path.
- SR-F.13.4.11 Create Revised Candidate Flight Path

Each time it revises the candidate flight path construction input, the Dynamic Path Planning Automation System shall create a new candidate flight path.
- SR-F.13.5 Coordinate Candidate Flight Path

Each time it creates a feasible candidate flight path with acceptable flight path qualities, the Dynamic Path Planning Automation System shall coordinate the flight path with the selected entities.
- SR-F.13.6 Handle Flight Path Revision Failure

The Dynamic Path Planning Automation System shall respond to its inability to revise the flight path per flight path revision rules.
- SR-F.13.6.1 Identify Flight Path Revision Contingency

Each time it fails to revise the candidate flight path per specified flight path revision rules, the Dynamic Path Planning Automation System shall determine if the failed revision results in a flight path contingency.
- SR-F.13.7 Handle Uncoordinated Flight Path

The Dynamic Path Planning Automation System shall handle uncoordinated candidate flight path based on specified flight path coordination rules.
- SR-F.13.7.1 Analyze Uncoordinated Flight Path Notification

Each time it receives a flight path coordination failure notification, the Dynamic Path Planning Automation System shall determine if the flight path coordination failure needs to be resolved.
- SR-F.13.7.2 Build Uncoordinated Flight Path Trigger

Each time it determines the need to resolve the flight path coordination failure, the Dynamic Path Planning Automation System shall generate a flight path revision trigger.
- SR-F.13.8 Store Active Flight Path Information

Each time it updates the flight path successfully, the Dynamic Path Planning

Automation System shall replace the active flight path information with the candidate flight path information.

SR-F.13.9 Provide Updated Flight Path Information

Each time it updates the flight path successfully, the Dynamic Path Planning Automation System shall provide the flight path information to the pilot via the pilot graphical user interface.

SR-F.14 Mitigate Flight Path Contingency

The Dynamic Path Planning Automation System shall mitigate the impact of all identified contingencies on the execution of the flight mission, in less than a specified time of contingency identification.

SR-F.14.1 Receive Contingency Notification

The Dynamic Path Planning Automation System shall asynchronously receive a notification each time a flight path contingency is detected.

SR-F.14.2 Identify Contingency Type

Each time a potential flight path contingency is detected, the Dynamic Path Planning Automation System shall determine the type of the contingency.

SR-F.14.3 Select Contingency Mitigation Option

The Dynamic Path Planning Automation System shall apply the contingency mitigation rules based on the type of the contingency to select one of the predefined contingency mitigation plans.

SR-F.14.4 Transition to Contingency State

Each time a potential flight path contingency is detected, the Dynamic Path Planning Automation System shall transition from its current (either Nominal or Shadow) state to Contingency state.

SR-F.14.5 Identify Missing Operating Environment Data

Each time a valid mission data is received, the Dynamic Path Planning Automation System shall determine if the system contains all the operating environment data needed to execute the flight mission.

SR-F.14.6 Request Missing Aerodrome Data

If it detects missing operating environment data, the Dynamic Path Planning Automation System shall transmit an aerodrome data request containing the list of aerodrome identifiers to the selected aerodrome service provider if air-to-ground service communication link is operational.

SR-F.14.7 Request Missing Airspace Data

If it detects missing operating environment data, the Dynamic Path Planning Automation System shall transmit an airspace data request containing the list of flight information region identifiers to the selected airspace service provider if air-to-ground service communication link is operational.

SR-F.14.8 Request Missing Atmosphere Data

If it detects missing operating environment data, the Dynamic Path Planning Automation System shall transmit an atmosphere data request containing the

list of flight information region identifiers to the selected atmosphere service provider if air-to-ground service communication link is operational.

SR-F.14.9 Request Missing Obstacle Data

If it detects missing operating environment data, the Dynamic Path Planning Automation System shall transmit an obstacle data request containing the list of flight information region identifiers to the selected obstacle service provider if air-to-ground service communication link is operational.

SR-F.14.10 Request Missing Traffic Data

If it detects missing operating environment data, the Dynamic Path Planning Automation System shall transmit a traffic data request containing the list of flight information region identifiers to the selected traffic service provider if air-to-ground service communication link is operational.

SR-F.14.11 Receive Missing Aerodrome Data

The Dynamic Path Planning Automation System shall synchronously receive requested aerodrome data sent by aerodrome data service provider if ground-to-air service communication link is operational.

SR-F.14.12 Receive Missing Airspace Data

The Dynamic Path Planning Automation System shall synchronously receive requested airspace data sent by airspace data service provider if ground-to-air service communication link is operational.

SR-F.14.13 Receive Missing Atmosphere Data

The Dynamic Path Planning Automation System shall synchronously receive requested atmosphere data sent by atmosphere data service provider if ground-to-air service communication link is operational.

SR-F.14.14 Receive Missing Obstacle Data

The Dynamic Path Planning Automation System shall synchronously receive requested obstacle data sent by obstacle data service provider if ground-to-air service communication link is operational.

SR-F.14.15 Receive Missing Traffic Data

The Dynamic Path Planning Automation System shall synchronously receive requested traffic data sent by traffic data service provider if ground-to-air service communication link is operational.

SR-F.14.16 Build Contingency Flight Path Construction Input

If the selected contingency mitigation plan does not require a diversion, then the Dynamic Path Planning Automation System shall build a new candidate flight path construction input in compliance with the selected contingency mitigation plan.

SR-F.14.17 Construct Contingency Flight Path

The Dynamic Path Planning Automation System shall construct a contingency candidate flight path each time a new flight path construction input is built.

SR-F.14.18 Evaluate Contingency Flight Path Qualities

The Dynamic Path Planning Automation System shall evaluate the flight path qualities of the contingency candidate flight path.

- SR-F.14.19 Identify Flight Diversion Parameters**
If the selected contingency mitigation plan requires a diversion, then the Dynamic Path Planning Automation System shall identify the aerodrome and airspace information needed to execute flight diversion mission to the selected landing site.
- SR-F.14.20 Execute Flight Diversion Mission**
The Dynamic Path Planning Automation System shall update the active flight path to comply with a command to revise the flight mission, within a specified timeframe, upon receiving the revised mission.
- SR-F.14.21 Update Contingency Flight Path Construction Input**
The Dynamic Path Planning Automation System shall update the mutable elements of the flight path construction input per specified rules.
- SR-F.14.22 Evaluate Contingency Criteria**
Each time it builds a contingency candidate flight path, the Dynamic Path Planning Automation System shall evaluate contingency criteria.
- SR-F.14.23 Coordinate Contingency Mitigated Flight Path**
The Dynamic Path Planning Automation System shall coordinate every candidate flight path that mitigates the detected contingency with the selected entities.
- SR-F.14.24 Alert of Unmitigated Contingency**
Each time the contingency candidate flight path exhibits persistence of one or more contingencies, the Dynamic Path Planning Automation System shall alert users of unmitigated contingencies.
- SR-F.14.25 Report Contingency Mitigation**
Each time the contingency candidate flight path exhibits resolution of all detected contingencies, the Dynamic Path Planning Automation System shall alert users of successful contingency mitigation.
- SR-F.14.26 Transition to Nominal State**
Upon resolution of all detected flight path contingencies, the Dynamic Path Planning Automation System shall transition from Contingency state to the prior (either Nominal or Shadow) state.
- SR-F.15 Comply with Tactical Override Command**
The Dynamic Path Planning Automation System shall comply with all tactical commands to pause and resume dynamic flight path planning, within a specified timeframe, upon receiving the command.
- SR-F.15.1 Receive Tactical System Command**
The Dynamic Path Planning Automation System shall asynchronously receive a tactical command sent by the aircraft tactical maneuver management system via the avionics bus interface.
- SR-F.15.2 Receive Tactical Pilot Command**
The Dynamic Path Planning Automation System shall asynchronously receive a tactical command sent by the pilot via the pilot graphical user interface.

SR-F.15.3 Identify Tactical Command Type

The Dynamic Path Planning Automation System shall identify if the type of tactical command is to pause, resume, or stop dynamic flight path planning.

SR-F.15.4 Respond to Pause Command

Upon receipt of a pause command while the system is in Active state, the Dynamic Path Planning Automation System shall transition to Passive state.

SR-F.15.5 Respond to Resume Command

Upon receipt of a resume command while the system is in Passive state, the Dynamic Path Planning Automation System shall transition to Active state.

SR-F.15.6 Respond to Stop Command

Upon receipt of a stop command, the Dynamic Path Planning Automation System shall transition to Idle state.

SR-F.15.7 Disregard Pause Command

The Dynamic Path Planning Automation System shall disregard a pause command if it is in Passive state.

SR-F.15.8 Disregard Resume Command

The Dynamic Path Planning Automation System shall disregard a resume command if it is in Active state.

SR-F.15.9 Monitor Flight Passively

The Dynamic Path Planning Automation System in passive state shall continue to monitor flight path progress.

SR-F.15.10 Inhibit Flight Path Alerts

The Dynamic Path Planning Automation System in passive state shall inhibit display and transmission of all flight path alerts.

SR-F.16 Monitor System Performance

The Dynamic Path Planning Automation System shall monitor system performance continually through the flight.

SR-F.16.1 Report Flight Path Status

The Dynamic Path Planning Automation System shall report the flight path status to the authorized users at a specified periodic rate.

SR-F.16.1.1 Receive Mission Manager Flight Status Request

The Dynamic Path Planning Automation System shall asynchronously receive all requests to report the flight status sent by the mission manager when the ground-to-air operator communication link is operational.

SR-F.16.1.2 Receive Aerodrome Coordinator Flight Status Request

The Dynamic Path Planning Automation System shall asynchronously receive all requests to report the flight status sent by the aerodrome coordinator when the ground-to-air data communication link is operational.

SR-F.16.1.3 Receive Airspace Coordinator Flight Status Request

The Dynamic Path Planning Automation System shall asynchronously receive all requests to report the flight status sent by the airspace coordinator when the ground-to-air data communication link is operational.

- SR-F.16.1.4 Compile Flight Path Status
The Dynamic Path Planning Automation System shall compile the active flight path status each time it changes.
- SR-F.16.1.5 Provide Flight Status to Pilot
The Dynamic Path Planning Automation System shall provide the active flight path status periodically at a predefined frequency to the pilot via the pilot graphical user interface.
- SR-F.16.1.6 Provide Flight Status to Mission Manager
The Dynamic Path Planning Automation System shall provide the active flight path status to the mission manager in response to each request via the air-to-ground operator communication link.
- SR-F.16.1.7 Provide Flight Status to Aerodrome Coordinator
The Dynamic Path Planning Automation System shall provide the active flight path status to the aerodrome coordinator in response to each request via the air-to-ground data communication link.
- SR-F.16.1.8 Provide Flight Status to Airspace Coordinator
The Dynamic Path Planning Automation System shall provide the active flight path status to the airspace coordinator in response to each request via the air-to-ground data communication link.
- SR-F.16.1.9 Compile Flight Path Alerts
The Dynamic Path Planning Automation System shall compile all alerts related to the active flight path.
- SR-F.16.1.10 Provide Flight Alerts to Pilot
The Dynamic Path Planning Automation System shall provide all flight path alerts to the pilot via the pilot graphical user interface, in less than a specified time of identifying a flight path planning anomaly.
- SR-F.16.1.11 Provide Flight Alerts to Mission Manager
The Dynamic Path Planning Automation System shall provide relevant flight path alerts via the air-to-ground operator communication link, in less than a specified time of identifying a flight path planning anomaly.
- SR-F.16.1.12 Provide Flight Alerts to Aerodrome Coordinator
The Dynamic Path Planning Automation System shall provide relevant flight path alerts to the aerodrome coordinator in response to each request via the air-to-ground data communication link, in less than a specified time of identifying a flight path planning anomaly.
- SR-F.16.1.13 Provide Flight Alerts to Airspace Coordinator
The Dynamic Path Planning Automation System shall provide relevant flight path alerts to the airspace coordinator in response to each request via the air-to-ground data communication link, in less than a specified time of identifying a flight path planning anomaly.
- SR-F.16.2 Report System Health Status
The Dynamic Path Planning Automation System shall report the automation system health status to the authorized users at a specified periodic rate.
- SR-F.16.2.1 Receive Mission Manager Health Status Request
The Dynamic Path Planning Automation System shall asynchronously

receive all requests to report the system health sent by the mission manager when the ground-to-air operator communication link is operational.

SR-F.16.2.2 Compile System Health Status

The Dynamic Path Planning Automation System shall compile health status of its components and interfaces each time it changes.

SR-F.16.2.3 Provide Health Status to Pilot

The Dynamic Path Planning Automation System shall provide the system health status periodically at a predefined frequency to the pilot via the pilot graphical user interface.

SR-F.16.2.4 Provide Health Status to Mission Manager

The Dynamic Path Planning Automation System shall provide the system health status to the mission manager in response to each request via the air-to-ground operator communication link.

SR-F.16.2.5 Compile System Health Alerts

The Dynamic Path Planning Automation System shall compile all alerts related to the health of the system components and interfaces.

SR-F.16.2.6 Provide Health Alerts to Pilot

The Dynamic Path Planning Automation System shall provide all system health alerts to the pilot via the pilot graphical user interface, in less than a specified time of identifying a system health anomaly.

SR-F.16.2.7 Provide Health Alerts to Mission Manager

The Dynamic Path Planning Automation System shall provide relevant system health alerts via the air-to-ground operator communication link, in less than a specified time of identifying a system health anomaly.

SR-F.17 Perform Postflight Analysis

Upon the completion of each flight, the Dynamic Path Planning Automation System shall report the results of a specified series of analyses about the completed flight.

SR-F.17.1 Analyze Flight Path Flexibility

For each flight path revision during the flight, the Dynamic Path Planning Automation System shall analyze every degradation of the flight path qualities leading up to the next revision.

SR-F.17.2 Analyze Pilot Workload

For each flight path revision during the flight, the Dynamic Path Planning Automation System shall determine the frequency and duration of each interaction with the pilot during the flight.

SR-F.17.3 Analyze System Performance

The Dynamic Path Planning Automation System shall compare the system performance monitored during the flight against the expected performance requirements per system design.

5.3 Usability Requirements

The list of system usability requirements includes:

SR-U.1 Nuisance Alerts

The Dynamic Path Planning Automation System shall minimize providing repetitious alerts.

SR-U.1.1 Input Validation

The Dynamic Path Planning Automation System shall minimize providing repetitious alerts related to input data validation error.

SR-U.1.2 Path Creation

The Dynamic Path Planning Automation System shall minimize providing repetitious alerts when it is unable to create a valid flight path for the specified input parameters.

SR-U.1.3 Flight Path Feasibility

The Dynamic Path Planning Automation System shall minimize providing repetitious alerts when the computed flight path is not feasible per flight planning rules.

SR-U.1.4 Flight Path Qualities

The Dynamic Path Planning Automation System shall minimize providing repetitious alerts when the computed flight path does not meet desired flight path qualities.

SR-U.1.5 Flight Path Evaluation

The Dynamic Path Planning Automation System shall minimize providing repetitious alerts when it is unable to evaluate the flight path qualities of a specified flight path.

SR-U.1.6 Flight Path Coordination

The Dynamic Path Planning Automation System shall minimize providing repetitious flight path coordination failure alerts.

SR-U.1.7 Flight Path Revision

The Dynamic Path Planning Automation System shall minimize providing repetitious alerts when it is unable to find a valid flight path construction input due to unavailable degrees of freedom with respect to the mutable elements of construction input.

SR-U.1.8 Flight Path Conformance

The Dynamic Path Planning Automation System shall minimize providing repetitious alerts related to the aircraft's out of conformance with respect to the active flight path.

SR-U.1.9 Flight Path Contingency

The Dynamic Path Planning Automation System shall minimize providing repetitious flight path contingency identification alerts.

SR-U.1.10 Missing Input

The Dynamic Path Planning Automation System shall minimize providing repetitious alerts when it determines the expected data to be missing.

SR-U.1.11 System Health

The Dynamic Path Planning Automation System shall minimize providing repetitious health anomaly alerts.

SR-U.2 False Alerts

The Dynamic Path Planning Automation System shall minimize providing false alerts.

SR-U.2.1 Input Validation

The Dynamic Path Planning Automation System shall minimize alerting about valid input data.

SR-U.2.2 Path Creation

The Dynamic Path Planning Automation System shall minimize alerting about a valid computed flight path.

SR-U.2.3 Flight Path Feasibility

The Dynamic Path Planning Automation System shall minimize alerting incorrectly about flight path feasibility.

SR-U.2.4 Flight Path Qualities

The Dynamic Path Planning Automation System shall minimize alerting incorrectly about the satisfaction of desired flight path qualities.

SR-U.2.5 Flight Path Evaluation

The Dynamic Path Planning Automation System shall minimize alerting incorrectly about its ability to evaluate a flight path successfully.

SR-U.2.6 Flight Path Coordination

The Dynamic Path Planning Automation System shall minimize alerting incorrectly about flight path coordination results.

SR-U.2.7 Flight Path Revision

The Dynamic Path Planning Automation System shall minimize alerting incorrectly about its ability to compile a valid flight path construction input.

SR-U.2.8 Flight Path Conformance

The Dynamic Path Planning Automation System shall minimize alerting incorrectly about aircraft's conformance to the active flight path.

SR-U.2.9 Flight Path Contingency

The Dynamic Path Planning Automation System shall minimize alerting incorrectly about identifying flight path contingency.

SR-U.2.10 Missing Input

The Dynamic Path Planning Automation System shall minimize alerting incorrectly about missing data.

SR-U.2.11 System Health

The Dynamic Path Planning Automation System shall minimize alerting incorrectly about system health.

SR-U.3 Missed Alerts

The Dynamic Path Planning Automation System shall minimize failures to provide necessary alerts.

SR-U.3.1 Input Validation

The Dynamic Path Planning Automation System shall minimize failure to alert of input data validation error.

SR-U.3.2 Path Creation

The Dynamic Path Planning Automation System shall minimize failure to alert when it is unable to create a valid flight path for the specified input parameters.

SR-U.3.3 Flight Path Feasibility

The Dynamic Path Planning Automation System shall minimize failure to alert when the computed flight path is not feasible per flight planning rules.

SR-U.3.4 Flight Path Qualities

The Dynamic Path Planning Automation System shall minimize failure to alert when the computed flight path does not meet desired flight path qualities.

SR-U.3.5 Flight Path Evaluation

The Dynamic Path Planning Automation System shall minimize failure to alert when it is unable to evaluate the flight path qualities of a specified flight path.

SR-U.3.6 Flight Path Coordination

The Dynamic Path Planning Automation System shall minimize failure to alert of flight path coordination failure.

SR-U.3.7 Flight Path Revision

The Dynamic Path Planning Automation System shall minimize failure to alert when it is unable to find a valid flight path construction input due to unavailable degrees of freedom with respect to the mutable elements of construction input.

SR-U.3.8 Flight Path Conformance

The Dynamic Path Planning Automation System shall minimize failure to alert when the aircraft is out of conformance with respect to the active flight path.

SR-U.3.9 Flight Path Contingency

The Dynamic Path Planning Automation System shall minimize failure to alert of identified flight path contingencies.

SR-U.3.10 Missing Input

The Dynamic Path Planning Automation System shall minimize failure to alert when it determines the expected data to be missing.

SR-U.3.11 System Health

The Dynamic Path Planning Automation System shall minimize failure to alert of system health anomaly.

SR-U.4 Response Time

The Dynamic Path Planning Automation System shall respond to user input within a specified timeframe.

SR-U.4.1 Pilot Interaction

The Dynamic Path Planning Automation System shall respond to pilot input within a specified timeframe.

SR-U.4.2 Flight Planner Interaction

The Dynamic Path Planning Automation System shall respond to flight planner input within a specified timeframe.

SR-U.5 Interface Availability

The Dynamic Path Planning Automation System shall be available to all authorized users through the life of the flight in at least a specified percent of instances.

SR-U.5.1 Graphical Pilot Interface

The Dynamic Path Planning Automation System shall exchange information with an authorized pilot via a graphical interface through the life of the flight in at least a specified percent of instances.

SR-U.5.2 Graphical Flight Planner Interface

The Dynamic Path Planning Automation System shall exchange information with an authorized flight planner via a graphical interface through the life of the flight in at least a specified percent of instances.

SR-U.5.3 Mission Manager Interface

The Dynamic Path Planning Automation System shall exchange information with an authorized mission manager via an air-ground data communication interface through the life of the flight in at least a specified percent of instances.

SR-U.5.4 Aerodrome Coordinator Interface

The Dynamic Path Planning Automation System shall exchange information with an authorized aerodrome coordinator via an air-ground data communication interface through the life of the flight in at least a specified percent of instances.

SR-U.5.5 Airspace Coordinator Interface

The Dynamic Path Planning Automation System shall exchange information with an authorized airspace coordinator via an air-ground data communication interface through the life of the flight in at least a specified percent of instances.

SR-U.5.6 Shared Operating Environment Interface

The Dynamic Path Planning Automation System shall exchange information with selected operating environment data providers via an air-ground data communication interface through the life of the flight in at least a specified percent of instances.

SR-U.5.7 Aircraft Systems Interface

The Dynamic Path Planning Automation System shall exchange information with aircraft systems via an onboard network communication interface through the life of the flight in at least a specified percent of instances.

SR-U.6 Input Customization

The Dynamic Path Planning Automation System shall allow authorized users to customize relevant input configuration parameters.

SR-U.6.1 Pilot Display Configuration

The Dynamic Path Planning Automation System shall allow an authorized pilot to select configuration parameters to customize pilot input.

SR-U.6.2 Flight Planner Display Configuration

The Dynamic Path Planning Automation System shall allow an authorized flight planner to select configuration parameters to customize planning input.

SR-U.7 Output Customization

The Dynamic Path Planning Automation System shall allow authorized users to customize relevant output configuration parameters.

SR-U.7.1 Pilot Display Configuration

The Dynamic Path Planning Automation System shall allow an authorized pilot to select configuration parameters to customize pilot display.

SR-U.7.2 Flight Planner Display Configuration

The Dynamic Path Planning Automation System shall allow an authorized pilot to select configuration parameters to customize flight planner display.

SR-U.7.3 User-specific Tailored Information

The Dynamic Path Planning Automation System shall apply configuration parameters to provide results tailored to the recipient.

5.4 Design Constraints

The list of system requirements derived from stakeholder design constraints includes:

SR-D.1 Regulatory Compliance

The Dynamic Path Planning Automation System must comply with all applicable regulatory standards.

SR-D.1.1 Availability Standard

The Dynamic Path Planning Automation System must meet system availability requirements as set in the applicable standards.

SR-D.1.2 Performance Standard

The Dynamic Path Planning Automation System must meet system operational performance requirements as set in the applicable standards.

SR-D.1.3 Safety Standard

The Dynamic Path Planning Automation System must meet system operational safety requirements as set in the applicable standards.

SR-D.1.4 Flight Planning Standard

The Dynamic Path Planning Automation System must comply with the applicable flight plan content and formatting standards.

SR-D.2 Air Navigation Service Provider Compliance

The Dynamic Path Planning Automation System must comply with the relevant ANSP guidelines and constraints.

SR-D.2.1 Special Use Airspace Avoidance

The Dynamic Path Planning Automation System must avoid restricted airspace as defined by the relevant Air Navigation Service Provider.

SR-D.2.2 Air Traffic Controller Instructions

The Dynamic Path Planning Automation System must comply with the relevant Air Traffic Controller instructions when flying in controlled airspace.

SR-D.3 Access to System

The Dynamic Path Planning Automation System must be accessible only by authorized users with valid credentials.

SR-D.3.1 Flight Planner

The Dynamic Path Planning Automation System must be accessible to an authorized flight planner with valid credentials.

SR-D.3.2 Onboard Pilot

The Dynamic Path Planning Automation System must be accessible to the authorized onboard pilot with valid credentials.

SR-D.3.3 Mission Manager

The Dynamic Path Planning Automation System must be accessible to the authorized flight mission manager with valid credentials.

SR-D.3.4 Aerodrome Coordinator

The Dynamic Path Planning Automation System must be accessible to an authorized aerodrome coordinator with valid credentials.

SR-D.3.5 Airspace Coordinator

The Dynamic Path Planning Automation System must be accessible to an authorized airspace coordinator with valid credentials.

SR-D.3.6 System Administrator

The Dynamic Path Planning Automation System must be accessible to an authorized system administrator with valid credentials.

SR-D.4 System Access to External Systems

The Dynamic Path Planning Automation System must be authorized to access all necessary external interfaces.

SR-D.4.1 Aerodrome Data Provider

The Dynamic Path Planning Automation System must be authorized to access all selected aerodrome data providers.

SR-D.4.2 Airspace Data Provider

The Dynamic Path Planning Automation System must be authorized to access all selected airspace data providers.

SR-D.4.3 Atmosphere Data Provider

The Dynamic Path Planning Automation System must be authorized to access all selected atmosphere data providers.

SR-D.4.4 Obstacle Data Provider

The Dynamic Path Planning Automation System must be authorized to access all selected obstacle data providers.

SR-D.4.5 Traffic Data Provider

The Dynamic Path Planning Automation System must be authorized to access all selected traffic data providers.

SR-D.4.6 Aircraft

The Dynamic Path Planning Automation System must be authorized to access the onboard aircraft data network and relevant aircraft sensors, as applicable for the type of the aircraft.

5.5 Physical Requirements

The list of system physical requirements includes:

SR-P.1 Onboard Installation

The Dynamic Path Planning Automation System shall support onboard installation on operator-selected types of aircraft.

SR-P.2 Ground Installation

The Dynamic Path Planning Automation System shall support installation at an authorized flight operations center.

SR-P.3 Aerodrome Coordinator Installation

The Dynamic Path Planning Automation System shall support installation at an authorized aerodrome flight coordination center.

SR-P.4 Airspace Coordinator Installation

The Dynamic Path Planning Automation System shall support installation at an authorized airspace flight coordination center.

5.6 Interface Requirements

The list of system interface requirements includes:

SR-I.1 Flight Planner To System

The Dynamic Path Planning Automation System shall receive data from the flight planner via a ground-based flight planner graphical user interface.

SR-I.1.1 Flight Planning Request

The Dynamic Path Planning Automation System shall receive flight path planning request from the flight planner via a ground-based flight planner graphical user interface.

SR-I.1.2 Updated System Configuration

The Dynamic Path Planning Automation System shall receive updated configuration parameter values from the flight planner via a ground-based flight planner graphical user interface.

SR-I.1.3 Baseline Concurrence Response

The Dynamic Path Planning Automation System shall receive concurrence to the baseline flight path from the flight planner via a ground-based flight planner graphical user interface.

SR-I.1.4 Baseline Confirmation Response

The Dynamic Path Planning Automation System shall receive confirmation of the baseline flight path from the flight planner via a ground-based flight planner graphical user interface.

SR-I.2 System To Flight Planner

The Dynamic Path Planning Automation System shall provide data to the flight planner via a ground-based flight planner graphical user interface.

SR-I.2.1 Flight Path

The Dynamic Path Planning Automation System shall provide the computed flight path to the flight planner via a ground-based flight planner graphical user interface.

SR-I.2.2 Flight Path Context

The Dynamic Path Planning Automation System shall provide relevant flight path contextual data to the flight planner via a ground-based flight planner graphical user interface.

SR-I.2.3 Flight Path Alerts

The Dynamic Path Planning Automation System shall provide all preflight flight path alerts to the flight planner via a ground-based flight planner graphical user interface.

SR-I.2.4 Flight Path Coordination Status

The Dynamic Path Planning Automation System shall provide the status of preflight flight path coordination with aerodrome and airspace coordinators to the flight planner via a ground-based flight planner graphical user interface.

SR-I.2.5 System Configuration Parameters

The Dynamic Path Planning Automation System shall provide selected system configuration parameters to the flight planner via a ground-based flight planner graphical user interface.

SR-I.2.6 System Health

The Dynamic Path Planning Automation System shall provide the system health during preflight planning to the flight planner via a ground-based flight planner graphical user interface.

SR-I.2.7 Baseline Concurrence Request

The Dynamic Path Planning Automation System shall request flight planner's

concurrence to proceed with the baseline flight path coordination via a ground-based flight planner graphical user interface.

SR-I.2.8 Baseline Confirmation Request

The Dynamic Path Planning Automation System shall request pilot's confirmation of the coordinated baseline flight path via a ground-based flight planner graphical user interface.

SR-I.3 Onboard Pilot to System

The Dynamic Path Planning Automation System shall receive data from the onboard pilot via an onboard pilot graphical user interface.

SR-I.3.1 Revised Mission

The Dynamic Path Planning Automation System shall receive the revised flight mission data from the onboard pilot via an onboard pilot graphical user interface.

SR-I.3.2 Observed Changes

The Dynamic Path Planning Automation System shall receive all changes to the onboard and surrounding environment observed by the onboard pilot via an onboard pilot graphical user interface.

SR-I.3.3 Flight Path Proposal

The Dynamic Path Planning Automation System shall receive new flight path proposed by the pilot via an onboard pilot graphical user interface.

SR-I.3.4 Flight Path Concurrence Response

The Dynamic Path Planning Automation System shall receive concurrence to a candidate flight path from the onboard pilot via an onboard pilot graphical user interface.

SR-I.3.5 Flight Path Confirmation Response

The Dynamic Path Planning Automation System shall receive confirmation of the candidate flight path from the onboard pilot via an onboard pilot graphical user interface.

SR-I.3.6 Tactical Override Command

The Dynamic Path Planning Automation System shall receive all tactical commands potentially overriding dynamic flight path planning from the onboard pilot via an onboard pilot graphical user interface.

SR-I.3.7 Updated Configuration Parameters

The Dynamic Path Planning Automation System shall receive updated system configuration parameters from the onboard pilot via an onboard pilot graphical user interface.

SR-I.4 System To Onboard Pilot

The Dynamic Path Planning Automation System shall provide data to the onboard pilot via an onboard pilot graphical user interface.

SR-I.4.1 Flight Path

The Dynamic Path Planning Automation System shall provide the revised flight path to the onboard pilot via an onboard pilot graphical user interface.

SR-I.4.2 Flight Path Context

The Dynamic Path Planning Automation System shall provide relevant flight path contextual data to the onboard pilot via an onboard pilot graphical user interface.

SR-I.4.3 Flight Path Alerts

The Dynamic Path Planning Automation System shall provide all alerts related to the revised flight path to the onboard pilot via an onboard pilot graphical user interface.

SR-I.4.4 Flight Path Concurrence Request

The Dynamic Path Planning Automation System shall request pilot's concurrence to proceed with a candidate flight path coordination via an onboard pilot graphical user interface.

SR-I.4.5 Flight Path Confirmation Request

The Dynamic Path Planning Automation System shall request pilot's confirmation of a coordinated candidate flight path via an onboard pilot graphical user interface.

SR-I.4.6 Proposal Evaluation Response

The Dynamic Path Planning Automation System shall provide response to all pilot proposed flight paths via an onboard pilot graphical user interface.

SR-I.4.7 System Configuration Parameters

The Dynamic Path Planning Automation System shall provide selected system configuration parameters to the pilot via an onboard pilot graphical user interface.

SR-I.4.8 System Health

The Dynamic Path Planning Automation System shall provide periodic system health status to the pilot via an onboard pilot graphical user interface.

SR-I.5 Mission Manager To System

The Dynamic Path Planning Automation System shall receive data from the mission manager via a ground-to-air operator communication interface.

SR-I.5.1 Baseline Flight Path

The Dynamic Path Planning Automation System shall receive the baseline flight path from the mission manager via a ground-to-air operator communication interface.

SR-I.5.2 Revised Mission

The Dynamic Path Planning Automation System shall receive the revised flight mission data from the mission manager via a ground-to-air operator communication interface.

SR-I.5.3 Flight Path Proposal

The Dynamic Path Planning Automation System shall receive new flight path proposed by the mission manager via a ground-to-air operator communication interface.

SR-I.5.4 Flight Path Status Request

The Dynamic Path Planning Automation System shall receive flight path status request from the mission manager via a ground-to-air operator communication interface.

SR-I.6 System To Mission Manager

The Dynamic Path Planning Automation System shall provide data to the mission manager via an air-to-ground operator communication interface.

SR-I.6.1 Flight Path Alerts

The Dynamic Path Planning Automation System shall provide a preselected set of flight path alerts to the mission manager via an air-to-ground operator communication interface.

SR-I.6.2 Proposal Evaluation Response

The Dynamic Path Planning Automation System shall provide response to all mission manager proposed flight paths via an air-to-ground operator communication interface.

SR-I.6.3 System Health

The Dynamic Path Planning Automation System shall provide periodic system health status to the mission manager via an air-to-ground operator communication interface.

SR-I.6.4 Flight Path Status

The Dynamic Path Planning Automation System shall provide periodic flight path status to the mission manager via an air-to-ground operator communication interface.

SR-I.7 Aircraft Avionics To System

The Dynamic Path Planning Automation System shall receive data from the aircraft avionics system via the aircraft avionics bus interface.

SR-I.7.1 Sensed Atmosphere Data

The Dynamic Path Planning Automation System shall receive atmosphere data sensed by the aircraft sensors via the aircraft avionics bus interface.

SR-I.7.2 Sensed Obstacle Data

The Dynamic Path Planning Automation System shall receive obstacle data sensed by the aircraft sensors via the aircraft avionics bus interface.

SR-I.7.3 Sensed Traffic Data

The Dynamic Path Planning Automation System shall receive traffic aircraft data sensed by the aircraft sensors via the aircraft avionics bus interface.

SR-I.7.4 Sensed Terrain Data

The Dynamic Path Planning Automation System shall receive terrain data sensed by the aircraft sensors via the aircraft avionics bus interface.

SR-I.8 Aircraft Propulsion To System

The Dynamic Path Planning Automation System shall receive data from the aircraft propulsion system via the aircraft avionics bus interface.

SR-I.8.1 Fuel Consumption Rate

The Dynamic Path Planning Automation System shall receive aircraft fuel consumption rate from the aircraft propulsion system via the aircraft avionics bus interface.

SR-I.8.2 Fuel Remaining

The Dynamic Path Planning Automation System shall receive the amount of fuel onboard from the aircraft propulsion system via the aircraft avionics bus interface.

SR-I.9 Aircraft Tactical System To System

The Dynamic Path Planning Automation System shall receive data from the aircraft tactical maneuver management system via the aircraft avionics bus interface.

SR-I.9.1 Pause Command

The Dynamic Path Planning Automation System shall receive all commands to pause dynamic path planning from the aircraft tactical maneuver management system via the aircraft avionics bus interface.

SR-I.9.2 Resume Command

The Dynamic Path Planning Automation System shall receive all commands to resume dynamic path planning from the aircraft tactical maneuver management system via the aircraft avionics bus interface.

SR-I.9.3 Stop Command

The Dynamic Path Planning Automation System shall receive all commands to stop dynamic path planning from the aircraft tactical maneuver management system via the aircraft avionics bus interface.

SR-I.10 Aircraft Guidance To System

The Dynamic Path Planning Automation System shall receive data from the aircraft guidance system via the aircraft avionics bus interface.

SR-I.10.1 Flight Path Execution Status

The Dynamic Path Planning Automation System shall receive flight path execution status from the aircraft guidance system via the aircraft avionics bus interface.

SR-I.11 System To Aircraft Guidance

The Dynamic Path Planning Automation System shall provide data to the aircraft guidance system via the aircraft avionics bus interface.

SR-I.11.1 Flight Path Execution Request

The Dynamic Path Planning Automation System shall provide flight path execution request to the aircraft guidance system via the aircraft avionics bus interface.

SR-I.12 Aerodrome Coordinator To System

The Dynamic Path Planning Automation System shall receive data from a selected aerodrome coordinator via a ground-to-air coordinator communication interface.

SR-I.12.1 Flight Path Coordination Response

The Dynamic Path Planning Automation System shall receive flight path coordination response from a selected aerodrome coordinator via a ground-to-air coordinator communication interface.

SR-I.12.2 Flight Path Proposal

The Dynamic Path Planning Automation System shall receive new flight path proposed by an authorized aerodrome coordinator via a ground-to-air coordinator communication interface.

SR-I.12.3 Flight Path Status Request

The Dynamic Path Planning Automation System shall receive flight path status request from an authorized aerodrome coordinator via a ground-to-air coordinator communication interface.

SR-I.13 System To Aerodrome Coordinator

The Dynamic Path Planning Automation System shall provide data to a selected aerodrome coordinator via an air-to-ground coordinator communication interface.

SR-I.13.1 Flight Path Coordination Request

The Dynamic Path Planning Automation System shall provide flight path coordination request to a selected aerodrome coordinator via an air-to-ground coordinator communication interface.

SR-I.13.2 Proposal Evaluation Response

The Dynamic Path Planning Automation System shall provide response to all aerodrome coordinator proposed flight paths via an air-to-ground coordinator communication interface.

SR-I.13.3 Flight Path Alert

The Dynamic Path Planning Automation System shall provide selected flight path alerts to a selected aerodrome coordinator via an air-to-ground coordinator communication interface.

SR-I.13.4 Flight Path Status

The Dynamic Path Planning Automation System shall provide flight path status when requested by an authorized aerodrome coordinator via an air-to-ground coordinator communication interface.

SR-I.14 Airspace Coordinator To System

The Dynamic Path Planning Automation System shall receive data from a selected airspace coordinator via a ground-to-air coordinator communication interface.

SR-I.14.1 Flight Path Coordination Response

The Dynamic Path Planning Automation System shall receive flight path coordination response from a selected airspace coordinator via a ground-to-air coordinator communication interface.

SR-I.14.2 Flight Path Proposal

The Dynamic Path Planning Automation System shall receive new flight path proposed by an authorized airspace coordinator via a ground-to-air coordinator communication interface.

SR-I.14.3 Flight Path Status Request

The Dynamic Path Planning Automation System shall receive flight path status request from an authorized airspace coordinator via a ground-to-air coordinator communication interface.

SR-I.15 System To Airspace Coordinator

The Dynamic Path Planning Automation System shall provide data to a selected airspace coordinator via an air-to-ground coordinator communication interface.

SR-I.15.1 Flight Path Coordination Request

The Dynamic Path Planning Automation System shall provide flight path coordination request to a selected aerodrome coordinator via an air-to-ground coordinator communication interface.

SR-I.15.2 Proposal Evaluation Response

The Dynamic Path Planning Automation System shall provide response to all airspace coordinator proposed flight paths via an air-to-ground coordinator communication interface.

SR-I.15.3 Flight Path Alert

The Dynamic Path Planning Automation System shall provide selected flight path alerts to a selected airspace coordinator via an air-to-ground coordinator communication interface.

SR-I.15.4 Flight Path Status

The Dynamic Path Planning Automation System shall provide flight path status when requested by an authorized airspace coordinator via an air-to-ground coordinator communication interface.

SR-I.16 Aerodrome Data Provider To System

The Dynamic Path Planning Automation System shall receive data from a selected aerodrome data service provider via a ground-to-air data communication interface.

SR-I.16.1 Baseline Aerodrome Data

The Dynamic Path Planning Automation System shall receive a complete set of aerodrome data for the requested aerodromes from a selected aerodrome data service provider via a ground-to-air data communication interface.

SR-I.16.2 Changed Aerodrome Data

The Dynamic Path Planning Automation System shall receive changes to the aerodrome data for the requested aerodromes from a selected aerodrome data service provider via a ground-to-air data communication interface.

SR-I.17 System To Aerodrome Data Provider

The Dynamic Path Planning Automation System shall provide data to a selected aerodrome data service provider via an air-to-ground data communication interface.

SR-I.17.1 Aerodrome Identifiers

The Dynamic Path Planning Automation System shall provide a list of aerodromes to a selected aerodrome data service provider via an air-to-ground data communication interface.

SR-I.18 Airspace Data Provider To System

The Dynamic Path Planning Automation System shall receive data from a selected airspace data service provider via a ground-to-air data communication interface.

SR-I.18.1 Baseline Airspace Data

The Dynamic Path Planning Automation System shall receive a complete set of airspace data for the requested airspace regions from a selected airspace data service provider via a ground-to-air data communication interface.

SR-I.18.2 Changed Airspace Data

The Dynamic Path Planning Automation System shall receive changes to the airspace data for the requested airspace regions from a selected airspace data service provider via a ground-to-air data communication interface.

SR-I.19 System To Airspace Data Provider

The Dynamic Path Planning Automation System shall provide data to a selected airspace data service provider via an air-to-ground data communication interface.

SR-I.19.1 Airspace Identifiers

The Dynamic Path Planning Automation System shall provide a list of airspace

flight information region identifiers to a selected airspace data service provider via an air-to-ground data communication interface.

SR-I.20 Atmosphere Data Provider To System

The Dynamic Path Planning Automation System shall receive data from a selected atmosphere data service provider via a ground-to-air data communication interface.

SR-I.20.1 Baseline Atmosphere Data

The Dynamic Path Planning Automation System shall receive a complete set of atmosphere data for the requested airspace regions from a selected atmosphere data service provider via a ground-to-air data communication interface.

SR-I.20.2 Changed Atmosphere Data

The Dynamic Path Planning Automation System shall receive all available atmosphere data for the requested airspace regions from a selected atmosphere data service provider via a ground-to-air data communication interface.

SR-I.21 System To Atmosphere Data Provider

The Dynamic Path Planning Automation System shall provide data to a selected atmosphere data service provider via an air-to-ground data communication interface.

SR-I.21.1 Airspace Identifiers

The Dynamic Path Planning Automation System shall provide a list of airspace flight information region identifiers to a selected atmosphere data service provider via an air-to-ground data communication interface.

SR-I.22 Obstacle Data Provider To System

The Dynamic Path Planning Automation System shall receive data from a selected obstacle data service provider via a ground-to-air data communication interface.

SR-I.22.1 Baseline Obstacle Data

The Dynamic Path Planning Automation System shall receive a complete set of obstacle data for the requested airspace regions from a selected atmosphere data service provider via a ground-to-air data communication interface.

SR-I.22.2 Changed Obstacle Data

The Dynamic Path Planning Automation System shall receive changes to obstacle data for the requested airspace regions from a selected obstacle data service provider via a ground-to-air data communication interface.

SR-I.23 System To Obstacle Data Provider

The Dynamic Path Planning Automation System shall provide data to a selected obstacle data service provider via an air-to-ground data communication interface.

SR-I.23.1 Airspace Identifiers

The Dynamic Path Planning Automation System shall provide a list of airspace flight information region identifiers to a selected obstacle data service provider via an air-to-ground data communication interface.

SR-I.24 Traffic Data Provider To System

The Dynamic Path Planning Automation System shall receive data from a selected traffic data service provider via a ground-to-air data communication interface.

SR-I.24.1 Baseline Traffic Data

The Dynamic Path Planning Automation System shall receive a complete set of traffic data for the requested airspace regions from a selected traffic data service provider via a ground-to-air data communication interface.

SR-I.24.2 Changed Traffic Data

The Dynamic Path Planning Automation System shall receive changes to traffic data for the requested airspace regions from a selected traffic data service provider via a ground-to-air data communication interface.

SR-I.25 System To Traffic Data Provider

The Dynamic Path Planning Automation System shall provide data to a selected traffic data service provider via an air-to-ground data communication interface.

SR-I.25.1 Airspace Identifiers

The Dynamic Path Planning Automation System shall provide a list of airspace flight information region identifiers to a selected traffic data service provider via an air-to-ground data communication interface.

SR-I.26 Administrator To System

The Dynamic Path Planning Automation System shall receive data from the automation system administrator via a ground-to-air data communication interface.

SR-I.27 System To Administrator

The Dynamic Path Planning Automation System shall provide data to the automation system administrator via an air-to-ground data communication interface.

SR-I.28 Preflight Aerodrome Coordinator To System

The Dynamic Path Planning Automation System shall receive data from a selected aerodrome coordinator via Ethernet interface.

SR-I.28.1 Flight Path Coordination Response

The Dynamic Path Planning Automation System shall receive flight path coordination response from a selected aerodrome coordinator via Ethernet interface.

SR-I.29 Preflight System To Aerodrome Coordinator

The Dynamic Path Planning Automation System shall provide data to a selected aerodrome coordinator via Ethernet interface.

SR-I.29.1 Flight Path Coordination Request

The Dynamic Path Planning Automation System shall provide flight path coordination request to a selected aerodrome coordinator via Ethernet interface.

SR-I.30 Preflight Airspace Coordinator To System

The Dynamic Path Planning Automation System shall receive data from a selected airspace coordinator via Ethernet interface.

SR-I.30.1 Flight Path Coordination Response

The Dynamic Path Planning Automation System shall receive flight path coordination response from a selected airspace coordinator via Ethernet interface.

SR-I.31 Preflight System To Airspace Coordinator

The Dynamic Path Planning Automation System shall provide data to a selected airspace coordinator via Ethernet interface.

SR-I.31.1 Flight Path Coordination Request

The Dynamic Path Planning Automation System shall provide flight path coordination request to a selected aerodrome coordinator via Ethernet interface.

SR-I.32 Preflight Aerodrome Data Provider To System

The Dynamic Path Planning Automation System shall receive data from a selected aerodrome data service provider via Ethernet interface.

SR-I.32.1 Baseline Aerodrome Data

The Dynamic Path Planning Automation System shall receive a complete set of aerodrome data for the requested aerodromes from a selected aerodrome data service provider via Ethernet interface.

SR-I.33 Preflight System To Aerodrome Data Provider

The Dynamic Path Planning Automation System shall provide data to a selected aerodrome data service provider via Ethernet interface.

SR-I.33.1 Aerodrome Identifiers

The Dynamic Path Planning Automation System shall provide a list of aerodromes to a selected aerodrome data service provider via Ethernet interface.

SR-I.34 Preflight Airspace Data Provider To System

The Dynamic Path Planning Automation System shall receive data from a selected airspace data service provider via Ethernet interface.

SR-I.34.1 Baseline Airspace Data

The Dynamic Path Planning Automation System shall receive a complete set of airspace data for the requested airspace regions from a selected airspace data service provider via Ethernet interface.

SR-I.35 Preflight System To Airspace Data Provider

The Dynamic Path Planning Automation System shall provide data to a selected airspace data service provider via Ethernet interface.

SR-I.35.1 Airspace Identifiers

The Dynamic Path Planning Automation System shall provide a list of airspace flight information region identifiers to a selected airspace data service provider via Ethernet interface.

SR-I.36 Preflight Atmosphere Data Provider To System

The Dynamic Path Planning Automation System shall receive data from a selected atmosphere data service provider via Ethernet interface.

SR-I.36.1 Baseline Atmosphere Data

The Dynamic Path Planning Automation System shall receive a complete set of atmosphere data for the requested airspace regions from a selected atmosphere data service provider via Ethernet interface.

SR-I.37 Preflight System To Atmosphere Data Provider

The Dynamic Path Planning Automation System shall provide data to a selected atmosphere data service provider via Ethernet interface.

SR-I.37.1 Airspace Identifiers

The Dynamic Path Planning Automation System shall provide a list of airspace flight information region identifiers to a selected atmosphere data service provider via Ethernet interface.

SR-I.38 Preflight Obstacle Data Provider To System

The Dynamic Path Planning Automation System shall receive data from a selected obstacle data service provider via Ethernet interface.

SR-I.38.1 Baseline Obstacle Data

The Dynamic Path Planning Automation System shall receive a complete set of obstacle data for the requested airspace regions from a selected atmosphere data service provider via Ethernet interface.

SR-I.39 Preflight System To Obstacle Data Provider

The Dynamic Path Planning Automation System shall provide data to a selected obstacle data service provider via Ethernet interface.

SR-I.39.1 Airspace Identifiers

The Dynamic Path Planning Automation System shall provide a list of airspace flight information region identifiers to a selected obstacle data service provider via Ethernet interface.

SR-I.40 Preflight Traffic Data Provider To System

The Dynamic Path Planning Automation System shall receive data from a selected traffic data service provider via Ethernet interface.

SR-I.40.1 Baseline Traffic Data

The Dynamic Path Planning Automation System shall receive a complete set of traffic data for the requested airspace regions from a selected traffic data service provider via Ethernet interface.

SR-I.41 Preflight System To Traffic Data Provider

The Dynamic Path Planning Automation System shall provide data to a selected traffic data service provider via Ethernet interface.

SR-I.41.1 Airspace Identifiers

The Dynamic Path Planning Automation System shall provide a list of airspace flight information region identifiers to a selected traffic data service provider via Ethernet interface.

6 Logical System Architecture

This section describes a logical architecture of the DPP automation system in an AAM operating environment. The purpose of the logical architecture is to define an implementation-agnostic structure of the system that performs the system behavior to satisfy the specified system requirements (§5). A logical architecture represents the first step toward solving the problem formulated in the conceptual and functional architectures.



Since a number of logical architectures of a system could potentially satisfy a given set of system requirements, a trade study is often performed to select the optimal logical architecture of the system. The performance criterion of such a trade study is determined by the system measures of effectiveness corresponding to the highest priority stakeholder concerns.

6.1 Synopsis

As shown in Figure 1, the steps in developing a logical architecture are:

1. Define the operational domain elements to satisfy the system interface requirements.
2. Define the system behavior at its highest level.
3. Build a system structure to organize and manage logical parts of the system.
4. Define the system measures of performance to evaluate the system capabilities.
5. Allocate system requirements to the system parts, i.e., subsystems.

Section 6.2 describes the structural and behavioral elements of the AAM domain in which the DPP automation system is expected to operate. The AAM domain structure consists of an interconnected set of elements that trace back to the domain participants defined in the conceptual architecture (Figure 6). These elements are: mission manager, flight planner, onboard pilot, remote pilot, aircraft, aerodrome coordinator, airspace coordinator, operating environment data providers, and onboard and ground-based instances of the DPP automation system. Figure 37 depicts the information exchange between the AAM domain elements in the context of the onboard and ground-based instances of the DPP automation system. The interfaces between the DPP automation system and the domain elements are designed to satisfy the system interface requirements. Figures 38 depicts the activities which the AAM domain participants must perform in support of the DPP automation system operation, while Figures 39–40 show example actions of the pilot and the aircraft during the flight.

Section 6.3 outlines the top-level behavior of the DPP automation system in terms of its states. Specifically, the onboard DPP automation system powers on and remains in “Idle” state until it receives a baseline flight path at which time it transitions to “Initialized”

state. Upon activation of the baseline flight path by the pilot-in-command, the system transitions to “Active-Nominal” state. During nominal operation, the system remains in this state until the successful completion of the mission. In a nonnominal case, the system transitions from “Active-Nominal” state to “Active-Contingent” state upon detection of a contingency, and transitions back upon mitigation of the contingency. When in “Active” state, the system may receive a command from the aircraft or the pilot to pause dynamic flight path planning. The system transitions to “Passive” state and only transitions back to “Active” state upon receiving a command to resume active flight path planning. While in either “Active” or “Passive” state, the system transitions to “Idle” state upon receipt of a command to stop dynamic flight path planning.

Section 6.4 describes the structure of the DPP automation system needed to accomplish its expected behavior. Establishing the system structure entails defining the system parts (§6.4.1), the interfaces between these parts (§6.4.3), the interfaces between the parts and the external domain elements (§6.4.2), and the types of data needed to flow through these interfaces (§6.4.4). Figure 42 presents the DPP automation system as a collection of following parts which facilitate in accomplishing the required system behavior efficiently:

1. Flight path initialization subsystem,
2. Flight path creation subsystem,
3. Flight path monitoring subsystem,
4. Flight path evaluation subsystem,
5. Flight path revision subsystem,
6. Flight path coordination subsystem,
7. Flight contingency mitigation subsystem,
8. Input data processing subsystem,
9. Output data processing subsystem,
10. Health monitoring subsystem,
11. Alert management subsystem,
12. Operation management subsystem, and
13. Data repository subsystem.

Figures 43–44 present the external and internal interfaces of the DPP automation system. Figure 45 depicts the synthesis of the DPP automation system structure based on the exchange of data through these system interfaces between the system parts.

Section 6.5 defines a representative set of measures of performance of the DPP automation system. The measures of performance provide an indication of how well the system meets its requirements. Further research, stakeholder coordination, and system design iterations are needed in order to establish a comprehensive set of these measures.

Section 6.6 allocates the DPP automation system requirements to each of its subsystems. These subsystem requirements serve as the foundation of the logical architecture of the DPP automation subsystems (§7).

6.2 System Domain Elements

The DPP automation system is being designed to operate in an AAM operational environment. A precise representation of the AAM domain elements is necessary in order to provide a context to the DPP automation system logical architecture. In other words, a description of the DPP automation system architecture requires a glimpse into the AAM domain architecture. The scope of the AAM domain architecture is limited to defining its structure and behavior necessary to build a DPP automation system.

6.2.1 Domain Structure

As defined in §3.4, the AAM domain participants of interest to the DPP automation system are: mission manager, flight planner, onboard pilot, remote pilot, aircraft, aerodrome coordinator, airspace coordinator, and operating environment data providers. The interfaces and data exchange between the domain participants and the DPP automation system define the structure of the AAM domain architecture. Figure 37 depicts the data exchange between the DPP automation system and the AAM participants.

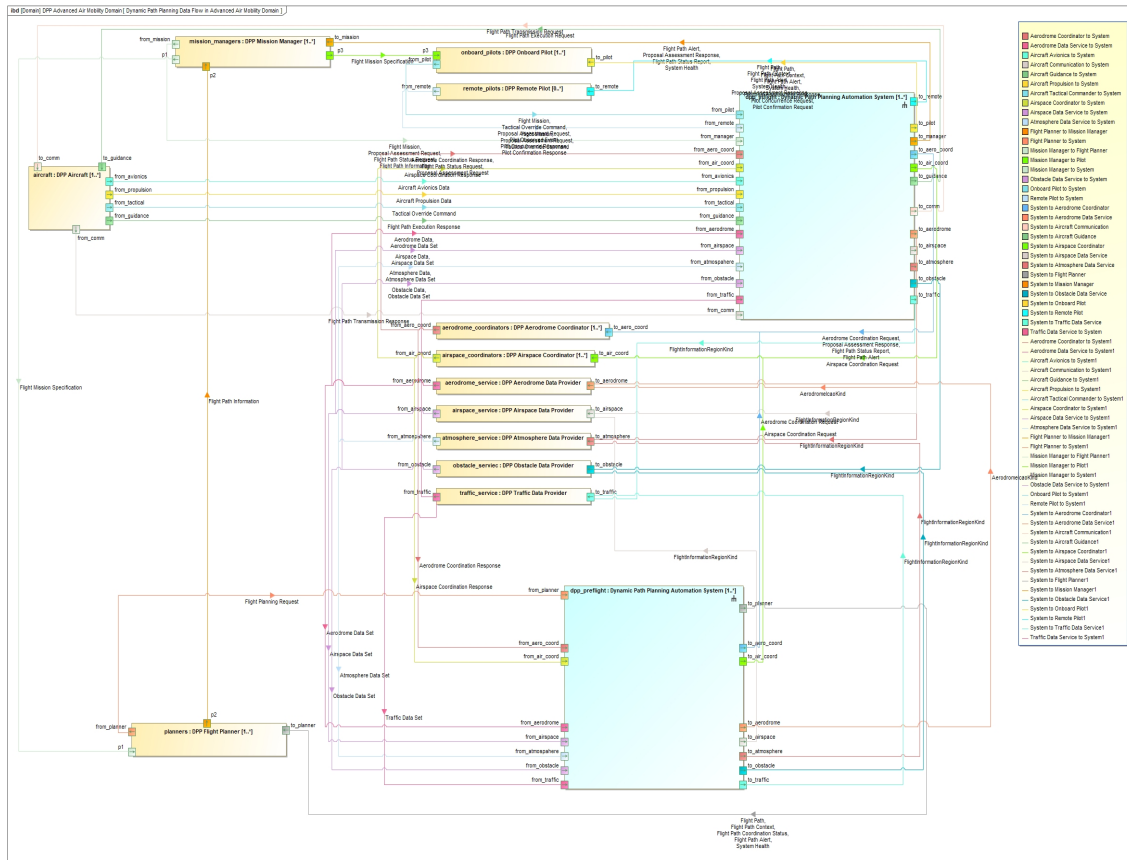


Figure 37: Advanced Air Mobility Domain Structure

6.2.2 Domain Behavior

As identified in §3.5, the DPP automation system interacts with the AAM domain participants in order to accomplish system use cases. The actions performed by these participants, collectively, define the behavior the AAM domain architecture. Figure 38 provides the list of activities that the AAM domain participants must perform in support of the DPP automation system operation. Figures 39 and 40 show example actions of an onboard pilot and the aircraft during the flight.

Legend		Domain Structure													
Allocate		DPP Aerodrome Coordinator DPP Aerodrome Data Provider DPP Aircraft DPP Airspace Coordinator DPP Airspace Data Provider DPP Atmosphere Data Provider DPP Flight Planner DPP Mission Manager DPP Obstacle Data Provider DPP Onboard Pilot DPP Remote Pilot DPP System Administrator													
Domain Behavior		4	1	5	3	2	2	8	5	2	12	4	1	2	
Build Flight Path Request(context DPP Flight Planner)	1														
Build Tactical Command	2														
Change Flight Mission	3														
Change Flight Planning Request(context DPP Flight Planner)	1														
Change System Configuration Parameters	4														
Compile Avionics Data(context DPP Aircraft)	1														
Compile Flight Mission Change Request	2														
Compile Flight Mission Elements(context DPP Mission Manager)	1														
Compile Observed Event Data(context DPP Onboard Pilot)	1														
Compile Propulsion Data(context DPP Aircraft)	1														
Concur Baseline Flight Path(context DPP Flight Planner)	1														
Confirm Baseline Flight Path(context DPP Flight Planner)	1														
Create Flight Path Status Request	3														
Perform Aerodrome Coordination(context DPP Aerodrome Coordinator)	1														
Perform Airspace Coordination(context DPP Airspace Coordinator)	1														
Perform Baseline Concurrence(context DPP Flight Planner)	1														
Perform Baseline Confirmation(context DPP Flight Planner)	1														
Perform Flight Path Concurrence(context DPP Onboard Pilot)	1														
Perform Flight Path Confirmation(context DPP Onboard Pilot)	1														
Perform Flight Path Execution(context DPP Aircraft)	1														
Perform Flight Path Transmission(context DPP Aircraft)	1														
Propose Flight Path Change	6														
Provide Baseline Aerodromes Data(context DPP Aerodrome Coordinator)	1														
Provide Baseline Airspace Data(context DPP Airspace Data Provider)	1														
Provide Baseline Atmosphere Data(context DPP Atmosphere Data Provider)	1														
Provide Baseline Obstacle Data(context DPP Obstacle Data Provider)	1														
Provide Baseline Traffic Data(context DPP Traffic Data Provider)	1														
Report Observed Atmosphere(context DPP Onboard Pilot)	1														
Report Observed Obstacle(context DPP Onboard Pilot)	1														
Report Observed Traffic(context DPP Onboard Pilot)	1														
Report Onboard Emergency(context DPP Onboard Pilot)	1														
Send New Aerodrome Data(context DPP Aerodrome Coordinator)	1														
Send New Airspace Data(context DPP Airspace Data Provider)	1														
Send New Atmosphere Data(context DPP Atmosphere Data Provider)	1														
Send New Obstacle Data(context DPP Obstacle Data Provider)	1														
Send New Traffic Data(context DPP Traffic Data Provider)	1														
Specify Flight Mission(context DPP Mission Manager)	1														

Figure 38: Advanced Air Mobility Domain Behavior

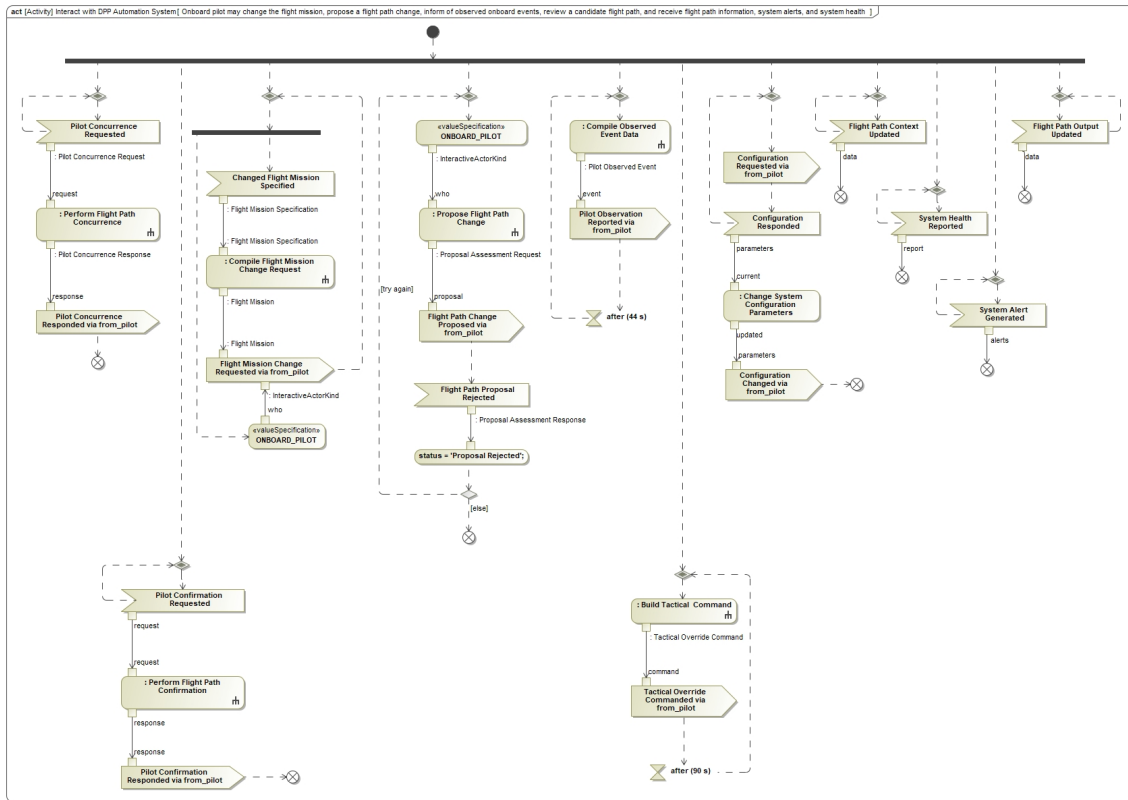


Figure 39: Onboard Pilot Actions

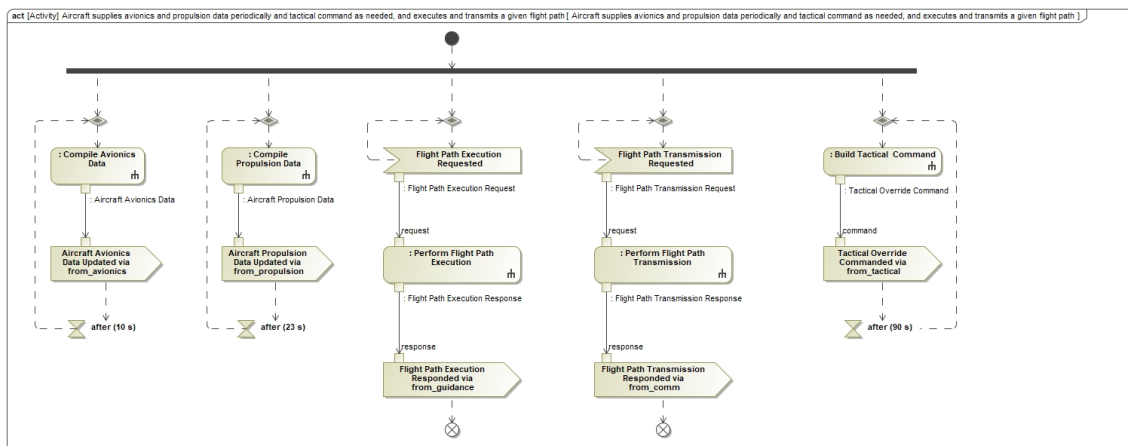


Figure 40: Aircraft Actions

6.3 Logical System Behavior

The logical architecture of the DPP automation system generally refrains from specifying detailed behaviors because, for all but the simplest of systems, they are delegated to the specific parts of the system. However, a top-level behavior of the system like the DPP automation system can be specified as a state machine. Figure 41 depicts the DPP automation system state transitions.

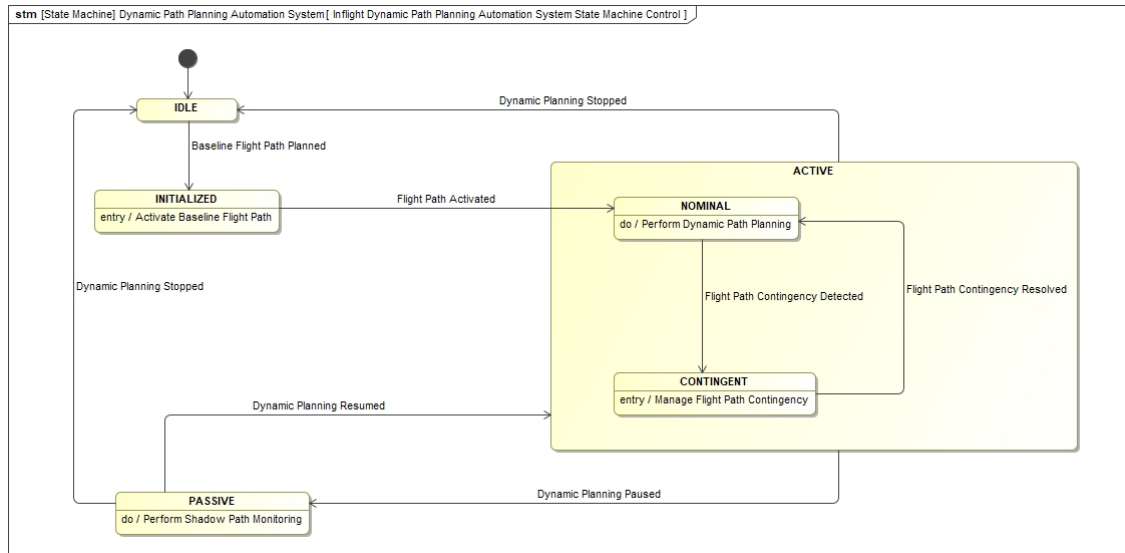


Figure 41: DPP automation system state machine behavior.

6.4 Logical System Structure

The logical architecture of the DPP automation system comprises the interconnections between the system parts as well as those between the system and the domain elements.

6.4.1 System Parts

The DPP automation system is organized as a set of logical (implementation-agnostic) parts, each dedicated to perform a specialized behavior of the system. The structural decomposition of the system, defined as part of the system functional architecture (§4.3), offers guidance on organizing the system. Figure 42 presents the logical system parts that make up the DPP automation systems, namely:

1. Input Data Processing Subsystem
2. Flight Path Initialization Subsystem
3. Flight Path Creation Subsystem
4. Flight Path Monitoring Subsystem
5. Flight Path Evaluation Subsystem
6. Flight Path Revision Subsystem
7. Flight Path Coordination Subsystem
8. Flight Contingency Mitigation Subsystem
9. Alert Management Subsystem
10. Health Monitoring Subsystem
11. Data Repository Subsystem
12. Operation Management Subsystem
13. Output Data Processing Subsystem

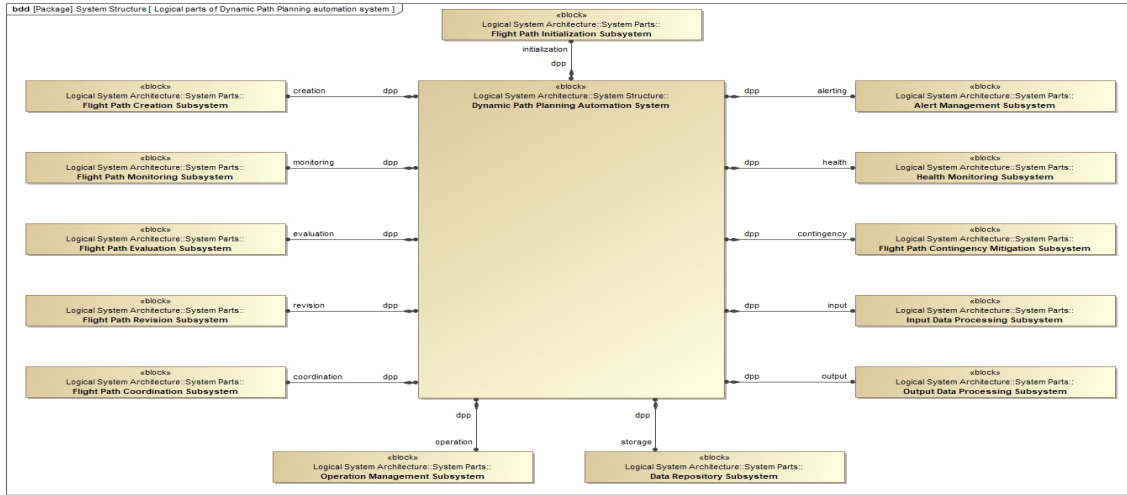


Figure 42: Logical System Parts

6.4.2 External Interfaces

The DPP automation system needs to interact with the AAM domain elements in order to satisfy the system interface requirements (§5.6). Figure 43 presents the external interfaces—16 incoming and 13 outgoing—between the DPP automation system and the AAM domain elements. These interfaces allow the DPP automation system to exchange data with the flight planner, onboard pilot, remote pilot, mission manager, aerodrome coordinator, airspace coordinator, aircraft systems, and operating environment data providers.

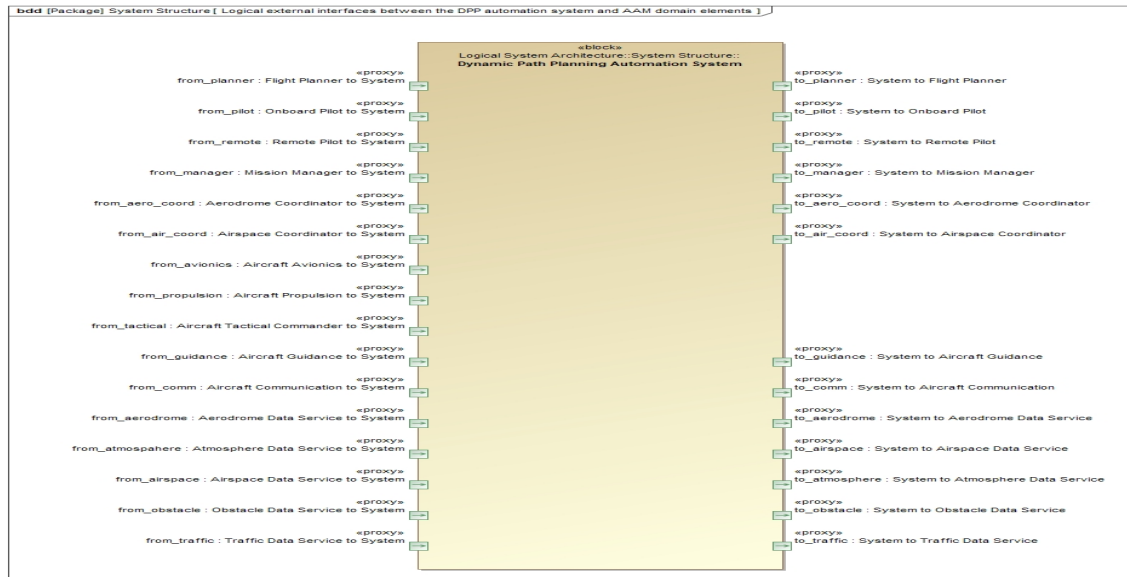


Figure 43: External Logical Interfaces

6.4.3 Internal Interfaces

Figure 44 presents the complete list of incoming and outgoing data for each of the DPP automation system parts.

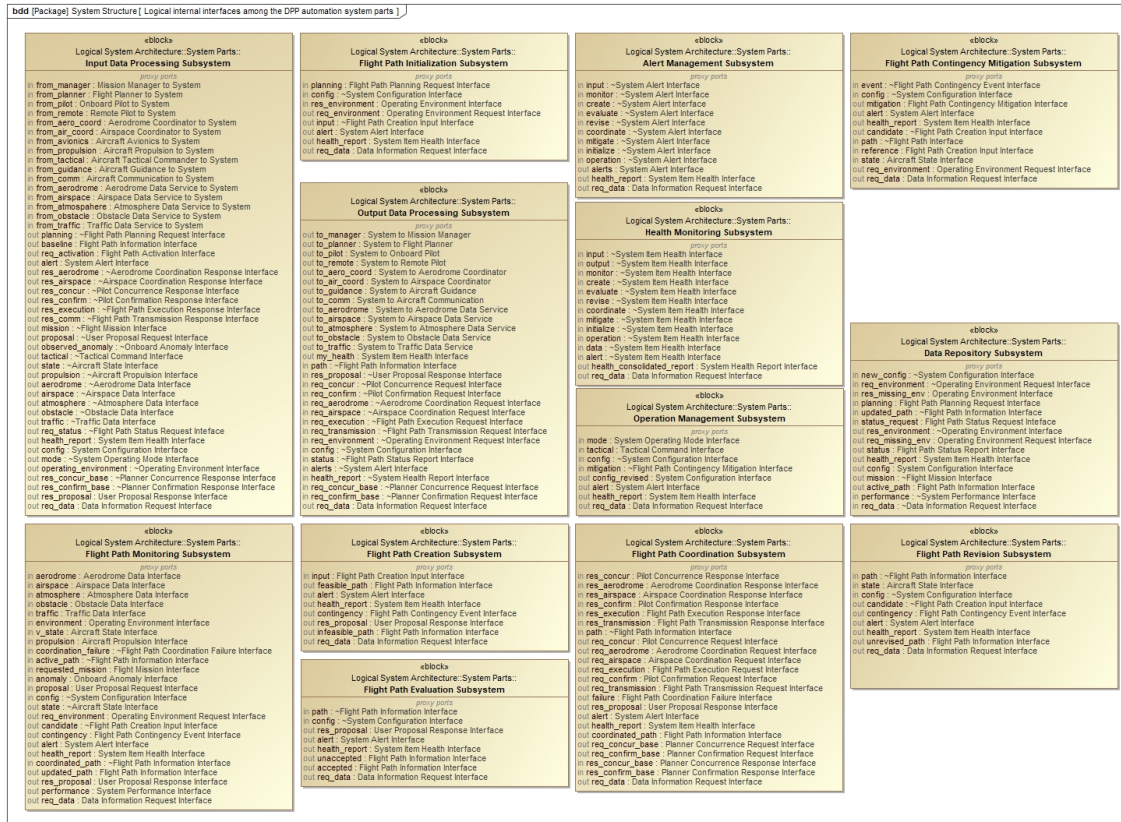


Figure 44: Internal Logical Interfaces

6.4.4 Data Exchange

The final step in building the logical structure of the DPP automation system is to identify the types of information exchanged between the system parts (including the system boundary) through the abovementioned system interfaces.

Figure 45 presents the information flow across the interfaces between the DPP automation system parts. The Flight Path Coordination Subsystem, responsible for the coordination of a candidate flight path with external entities, is highlighted as a way of explaining the contents of this figure. Specifically:

- The flight path coordination subsystem receives flight path information via (**in path**) from the flight path creation subsystem.
- The flight path coordination subsystem receives baseline flight path concurrence response (**in res_concur_base**) from the input data processing subsystem.
- The flight path coordination subsystem receives baseline flight path confirmation response (**in res_confirm_base**) from the input data processing subsystem.

- The flight path coordination subsystem receives aerodrome coordination response (**in res_aerodrome**) from the input data processing subsystem.
- The flight path coordination subsystem receives flight path concurrence response (**in res_concur**) from the input data processing subsystem.
- The flight path coordination subsystem receives airspace coordination response (**in res_airspace**) from the input data processing subsystem.
- The flight path coordination subsystem receives flight path confirmation response (**in res_confirm**) from the input data processing subsystem.
- The flight path coordination subsystem receives flight path execution response (**in res_execution**) from the input data processing subsystem.
- The flight path coordination subsystem receives flight path transmission response (**in res_transmission**) from the input data processing subsystem.
- The flight path coordination subsystem transmits baseline flight path concurrence request (**out req_concur_base**) to the output data processing subsystem.
- The flight path coordination subsystem transmits baseline flight path confirmation request (**out req_confirm_base**) to the output data processing subsystem.
- The flight path coordination subsystem transmits flight path concurrence request (**out req_concur**) to the output data processing subsystem.
- The flight path coordination subsystem transmits aerodrome coordination request (**out req_aerodrome**) to the output data processing subsystem.
- The flight path coordination subsystem transmits airspace coordination request (**out req_airspace**) to the output data processing subsystem.
- The flight path coordination subsystem transmits flight path confirmation request (**out req_confirm**) to the output data processing subsystem.
- The flight path coordination subsystem transmits flight path execution request (**out req_execution**) to the output data processing subsystem.
- The flight path coordination subsystem transmits flight path transmission request (**out req_transmission**) to the output data processing subsystem.
- The flight path coordination subsystem transmits coordinated flight path information (**out coordinated_path**) to the flight path monitoring subsystem.
- The flight path coordination subsystem transmits reason to reject a user proposed change (**out res_proposal**) to the output data processing subsystem.
- The flight path coordination subsystem transmits coordination failure status (**out failure**) to the flight path monitoring subsystem.
- The flight path coordination subsystem transmits subsystem anomaly alert (**out alert**) to the alert management subsystem.
- The flight path coordination subsystem transmits subsystem health report (**out health**) to the health monitoring subsystem.
- The flight path coordination subsystem transmits requested data item (**out req_data**) to the requesting subsystem.

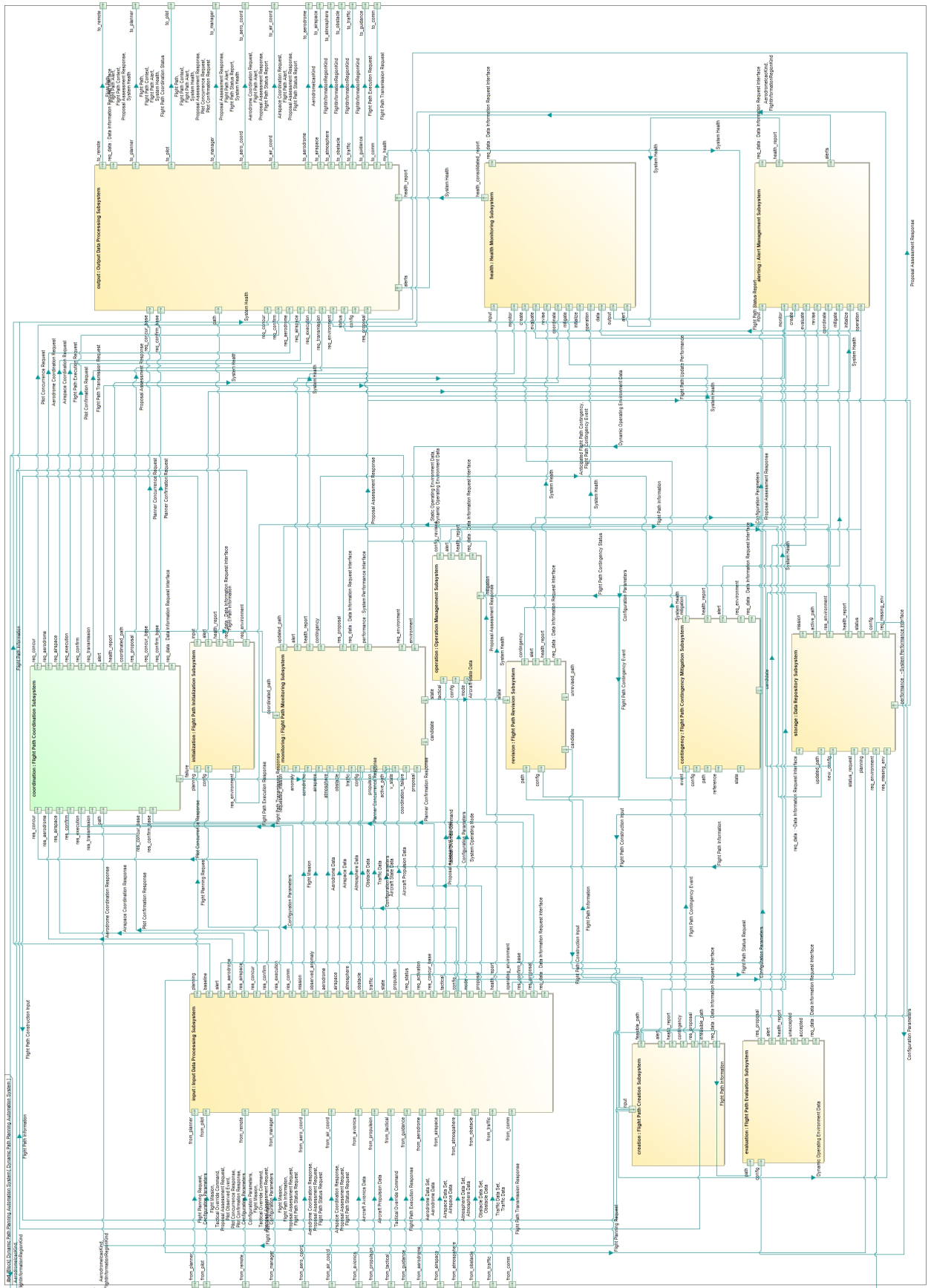


Figure 45: DPP Automation System Structure

6.5 System Measures of Performance

The measures of performance (MOPs) represent the performance characteristics of the system that indicate how well the system meets its requirements. While the MOEs are used to validate that the conceptual system satisfies the stakeholder needs, the MOPs are used to verify that the built system meets the specified requirements. An initial version of the DPP automation system measures of performance is extracted from the performance components of the functional requirements:

- Time to validate incoming data.
- Time to plan an acceptable initial flight path.
- Time to alert of invalid flight path creation request.
- Time to alert of unachievable mission.
- Time to create a flight path.
- Time to evaluate a flight path.
- Time to revise a flight path.
- Time to coordinate a flight path.
- Time to detect a flight-path-update trigger.
- Time to detect a flight path conflict with a known hazard.
- Time to deconflict the flight path against detected conflicts.
- Time to adapt the flight path in response to new aerodrome data.
- Time to adapt the flight path in response to new airspace data.
- Time to adapt the flight path in response to new atmosphere data.
- Time to adapt the flight path in response to new obstacle data.
- Time to adapt the flight path in response to new traffic data.
- Time to adapt the flight path in response to a pilot reported event.
- Time to adapt the flight path in response to a new mission.
- Time to adapt the flight path in response to a change in flight conformance.
- Time to detect a flight path contingency condition.
- Time to mitigate the impact of a flight path contingency on system operation.
- Time to evaluate a user proposal.
- Time to respond to a flight path coordination failure.
- Frequency of recognizing external inputs correctly.
- Frequency of missed alerts.
- Frequency of false alerts due to the incorrect application of a valid rule.
- Frequency of false alerts due to the correct application of an invalid rule.
- Frequency of flight path creation failures.
- Frequency of flight path evaluation failures.
- Frequency of flight path coordination failures.
- Frequency of flight path revision failures.

The MOPs of the DPP automation system will continue to be refined during the iterative systems engineering cycles of developing the logical architecture of the system and its parts.

6.6 System Requirements Allocation to Subsystems

The final step in the logical architecture development of the DPP automation system is to allocate the system requirements to the identified logical subsystems. Figure 46–48 provide examples of the allocation of the system functional requirements to the DPP automation subsystems.

alm [Package] System Requirements Allocation [System Requirements Allocated to Subsystems 1]											
Legend											
Allocate											
Dynamic Path Planning Automation System											
	Alerting	Contingency	Coordination	Creation	Flight	Health	Initialization	Input	Monitoring	Operation	Output
	5	21	78	29	28	3	13	163	82	19	38
	4	6	6	6	6	6	6	6	6	6	6
System Requirements	5	21	78	29	28	3	13	163	82	19	38
Design Constraints											
Functional Requirements	2	18	75	20	22		13	154	79	16	34
SR-F1 Build Baseline Flight Path											
SR-F1.1 Receive Flight Plan Request	1										
SR-F1.2 Validate Flight Planning Request	1										
SR-F1.2.1 Apply Request Validation Rule	1										
SR-F1.2.2 Alert of Invalid Flight Plan Request	1										
SR-F1.3 Compile Baseline Flight Planning Input	1										
SR-F1.3.1 Parse Flight Mission Specification	1										
SR-F1.3.2 Select Aircraft Performance Model	1										
SR-F1.3.3 Request Aerodrome Data	2										
SR-F1.3.4 Request Airspace Data	2										
SR-F1.3.5 Request Atmosphere Data	2										
SR-F1.3.6 Request Obstacle Data	2										
SR-F1.3.7 Request Traffic Data	2										
SR-F1.3.8 Determine Shared Data Need	1										
SR-F1.4 Compile Shared Operating Environment Data	1										
SR-F1.4.1 Receive Aerodrome Data	1										
SR-F1.4.2 Receive Airspace Data	1										
SR-F1.4.3 Receive Atmosphere Data	1										
SR-F1.4.4 Receive Obstacle Data	1										
SR-F1.4.5 Receive Traffic Data	1										
SR-F1.4.6 Validate Operating Environment Data	1										
SR-F1.4.7 Alert of Invalid Operating Environment Data	1										
SR-F1.4.8 Store Valid Operating Environment	1										
SR-F1.5 Initialize Flight Path Construction	1										
SR-F1.5.1 Compile Flight Path Construction Input	1										
SR-F1.5.2 Validate Flight Path Construction Input	1										
SR-F1.5.3 Alert of Unachievable Mission	1										
SR-F1.6 Create Flight Path	1										
SR-F1.6.1 Construct Flight Path	1										
SR-F1.6.2 Alert of Infeasible Flight Path	1										
SR-F1.7 Evaluate Flight Path	1										
SR-F1.7.1 Evaluate Flight Path Safety	1										
SR-F1.7.2 Evaluate Flight Path Optimality	1										
SR-F1.7.3 Alert of Unmet Flight Path Qualities	1										
SR-F1.7.4 Provide Flight Path Information	1										
SR-F2 Coordinate Baseline Flight Path	1										
SR-F3 Synchronize Baseline Flight Path	1										
SR-F4 Activate Baseline Flight Path											
SR-F5 Adapt Flight Path to Pilot Observed Event											
SR-F6 Adapt Flight Path to Shared Operating Environment											
SR-F7 Adapt Flight Path to Mission Change											
SR-F8 Adapt Flight Path to Sensed Information											
SR-F9 Adapt Flight Path to System Configuration Change											
SR-F10 Assess Flight Path Change Proposal											
SR-F11 Refresh Flight Path Periodically											
SR-F12 Coordinate Candidate Flight Path	1										
SR-F13 Update Candidate Flight Path	4										
SR-F14 Mitigate Flight Path Contingency	1										
SR-F15 Comply with Tactical Override Command											
SR-F16 Monitor System Performance											
SR-F17 Perform Postflight Analysis											
Interface Requirements											
Physical Requirements											
Usability Requirements	3	3	3	9	6	3		9	3	3	4

alm [Package] System Requirements Allocation [System Requirements Allocated to Subsystems 2]											
Legend											
Allocate											
Dynamic Path Planning Automation System											
	Alerting	Contingency	Coordination	Creation	Flight	Health	Initialization	Input	Monitoring	Operation	Output
	5	21	78	29	28	3	13	163	82	19	38
	4	6	6	6	6	6	6	6	6	6	6
System Requirements	5	21	78	29	28	3	13	163	82	19	38
Design Constraints											
Functional Requirements	2	18	75	20	22		13	154	79	16	34
SR-F1 Build Baseline Flight Path											
SR-F2 Coordinate Baseline Flight Path	1										
SR-F2.1 Request Flight Planner Concurrence	2										
SR-F2.2 Receive Flight Planner Concurrence	2										
SR-F2.3 Request Aerodrome Coordination	2										
SR-F2.4 Receive Aerodrome Coordination Response	2										
SR-F2.5 Request Airspace Coordination	2										
SR-F2.6 Receive Airspace Coordination Response	2										
SR-F2.7 Request Flight Planner Confirmation	2										
SR-F2.8 Notify Successful Baseline Coordination	1										
SR-F3 Synchronize Baseline Flight Path	1										
SR-F3.1 Receive Planned Flight Path	2										
SR-F3.2 Receive Baseline Activation Confirmation	3										
SR-F3.3 Notify Flight Path Synchronization	1										
SR-F4 Activate Baseline Flight Path											
SR-F4.1 Request Pilot Concurrence	2										
SR-F4.2 Receive Pilot Concurrence	2										
SR-F4.3 Analyze Pilot Concurrence	1										
SR-F4.4 Alert Pilot Nonconcurrence	1										
SR-F4.5 Request Flight Path Execution	2										
SR-F4.6 Receive Flight Path Execution Response	2										
SR-F4.7 Analyze Flight Path Execution Response	1										
SR-F4.8 Alert Flight Path Execution Failure	1										
SR-F4.9 Confirm Flight Path Activation	1										
SR-F5 Adapt Flight Path to Pilot Observed Event											
SR-F6 Adapt Flight Path to Shared Operating Environment											
SR-F7 Adapt Flight Path to Mission Change											
SR-F8 Adapt Flight Path to Sensed Information											
SR-F9 Adapt Flight Path to System Configuration Change											
SR-F10 Assess Flight Path Change Proposal											
SR-F11 Refresh Flight Path Periodically											
SR-F12 Coordinate Candidate Flight Path	1										
SR-F13 Update Candidate Flight Path	4										
SR-F14 Mitigate Flight Path Contingency	1										
SR-F15 Comply with Tactical Override Command											
SR-F16 Monitor System Performance											
SR-F17 Perform Postflight Analysis											
Interface Requirements											
Physical Requirements											
Usability Requirements	3	3	3	9	6	3		9	3	3	4

Figure 46: Subsystem Requirements Derivation-1

pkg [Package] System Requirements Allocation [System Requirements Allocated to Subsystems 5]

Legend

Allocate

Dynamic Path Planning Automation System

alerting : Alerting : contingency : contingency : coordination : coordination : flight creation : flight creation : flight evaluation : flight evaluation : flight health : flight health : flight initialization : flight initialization : flight input : flight input : flight monitoring : flight monitoring : flight operation : flight operation : flight output : flight output : flight revision : flight revision : flight storage : flight storage : data

System Requirements

5217829283131638219384634

Design Constraints

Functional Requirements

218752022131547916344334

SR-F1 Build Baseline Flight Path

SR-F2 Coordinate Baseline Flight Path

1

</

Figure 48: Subsystem Requirements Derivation-3

7 Logical Subsystems Architecture

This section describes logical architectures of the DPP automation system parts, i.e., subsystems. The purpose of building a logical architecture of a subsystem is to define an implementation-agnostic structure of the subsystem that is dedicated to performing a specified behavior in order to meet the allocated requirements.

7.1 Synopsis

As shown in Figure 1, the steps in developing a logical architecture of a subsystem parallel those used in developing the logical architecture of the system. Specifically, each subsystem becomes the system of interest and the system plays the role of the domain. This process is applied recursively until the system is decomposed to a sufficient level of detail to satisfy the specified system requirements (§5) and, thus, the underlying stakeholder concerns. Adapting the logical system architecture steps listed in §6.1 to a subsystem results in the following steps:

1. Define the domain architecture of the subsystem.
2. Define the behavior that the subsystem is tasked to perform by the system.
3. Build a subsystem structure to organize and manage the logical parts of the subsystem.
4. Define measures of performance of each subsystem to evaluate its capabilities.
5. Allocate and expand the requirements of a subsystem to its parts, i.e., subsubsystems.

Section 6 has already defined the domain—the DPP automation system—architecture of the subsystems. Sections 7.2–7.14 describe the behavior and structure of the DPP automation subsystems at varying levels of depth. This version of the architecture presents the behavior and structure of the following subsystems:

1. Flight Path Initialization Subsystem
2. Flight Path Creation Subsystem
3. Flight Path Monitoring Subsystem
4. Flight Path Evaluation Subsystem
5. Flight Path Revision Subsystem
6. Flight Path Coordination Subsystem
7. Input Data Processing Subsystem
8. Output Data Processing Subsystem
9. Data Repository Subsystem

A logical architecture of the following subsystems is deferred to a future version because a complete and accurate concept of the respective subsystems is evolving:

1. Flight Contingency Mitigation Subsystem
2. Alert Management Subsystem
3. Health Monitoring Subsystem
4. Operation Management Subsystem

Further research, stakeholder coordination, and system design iterations are needed in order to establish a comprehensive set of subsystem MOPs.

7.2 Flight Path Initializer

The flight path initializer is an implementation-agnostic representation of the flight path initialization subsystem of the DPP automation system. The basis of the behavioral and structural elements of the flight path initializer logical architecture is the set of system requirements allocated to the flight path initialization subsystem. Figure 49 shows an abridged version of these requirements (see §5 for the complete requirement statements).

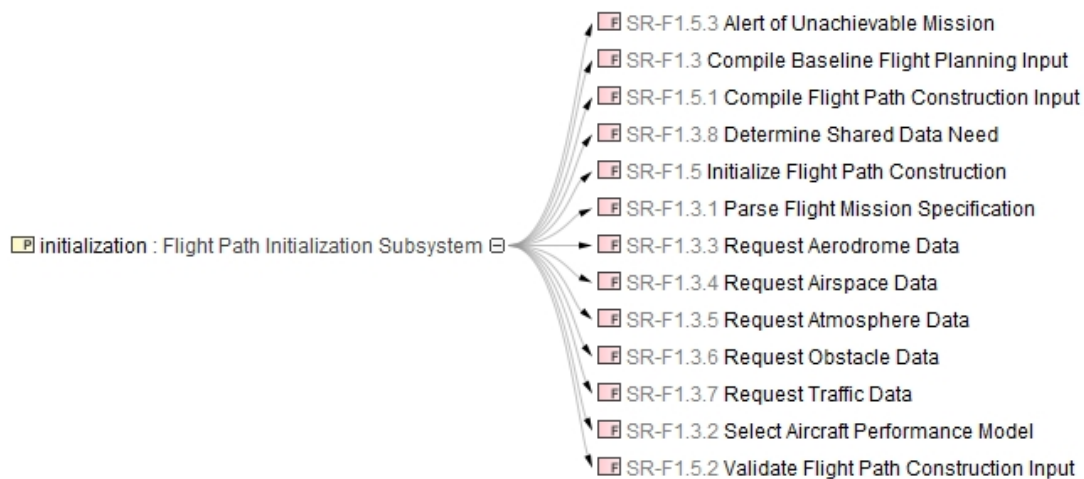


Figure 49: Flight Path Initializer Requirements

7.2.1 Subsystem Behavior

The behavior of the flight path initializer includes:

- (a) parsing the flight planning request to extract the mission specification and planning parameters,
- (b) compiling the baseline flight path construction input based on the flight planning request and the necessary operating environment data, and
- (c) reporting self-health periodically.

Figure 50 depicts an example behavior of the flight path initializer.

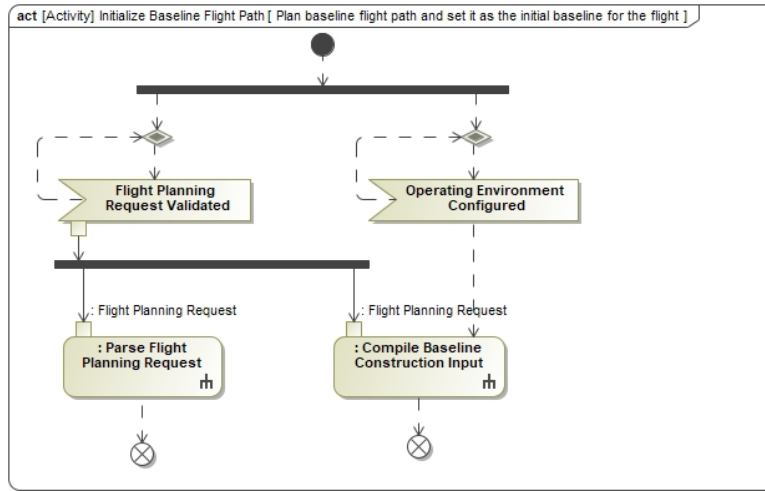


Figure 50: Flight Path Initializer Behavior

7.2.2 Subsystem Structure

The structure of the flight path initializer is defined by the information flow between its parts:

1. Baseline input compiler
2. Operating environment configurator
3. Health reporter

Figure 51 depicts the internal structure of the flight path initializer.

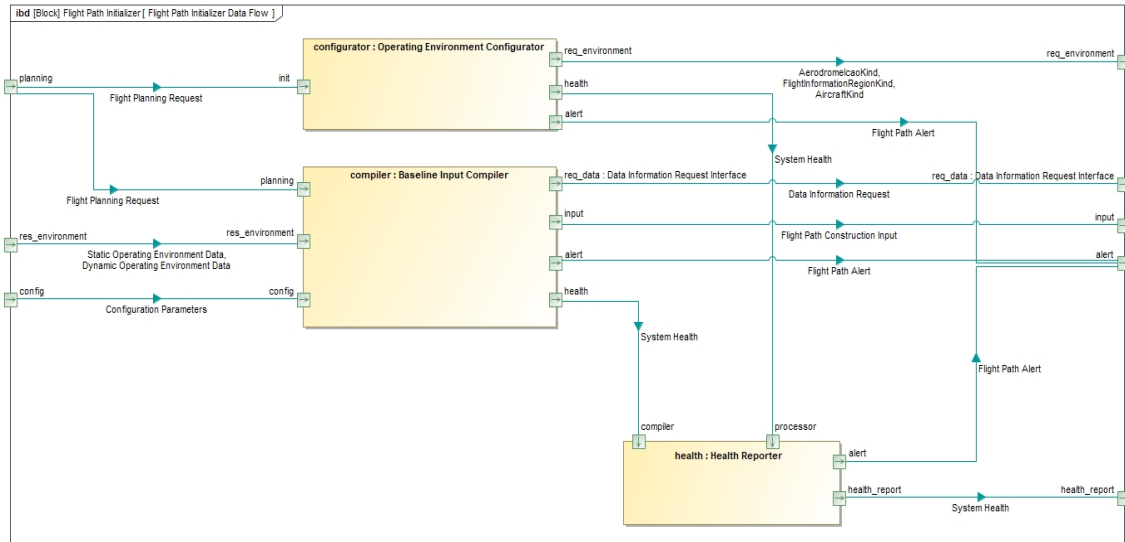


Figure 51: Flight Path Initializer Structure

7.3 Flight Path Creator

The flight path creator is an implementation-agnostic representation of the flight path creation subsystem of the DPP automation system. The basis of the behavioral and structural elements of the flight path creator logical architecture is the set of system requirements allocated to the flight path creation subsystem. Figure 52 shows an abridged version of these requirements (see §5 for the complete requirement statements).

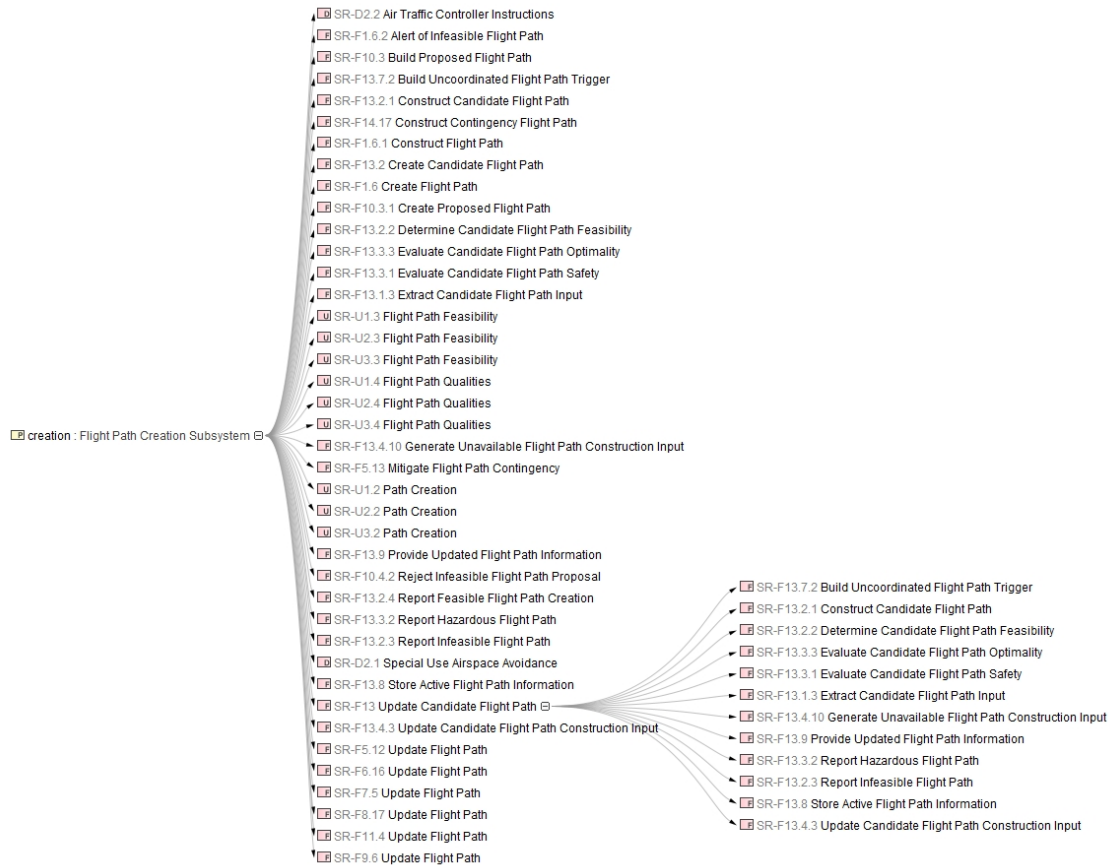


Figure 52: Flight Path Creator Requirements

7.3.1 Subsystem Behavior

The behavior of the flight path creator includes:

- constructing a flight path for the specified flight path construction input,
- informing the requestor of availability of a feasible flight path,
- alerting the requestor of unavailability of a feasible flight path and,
- reporting self-health periodically.

Figure 53 depicts an example behavior of the flight path creator.

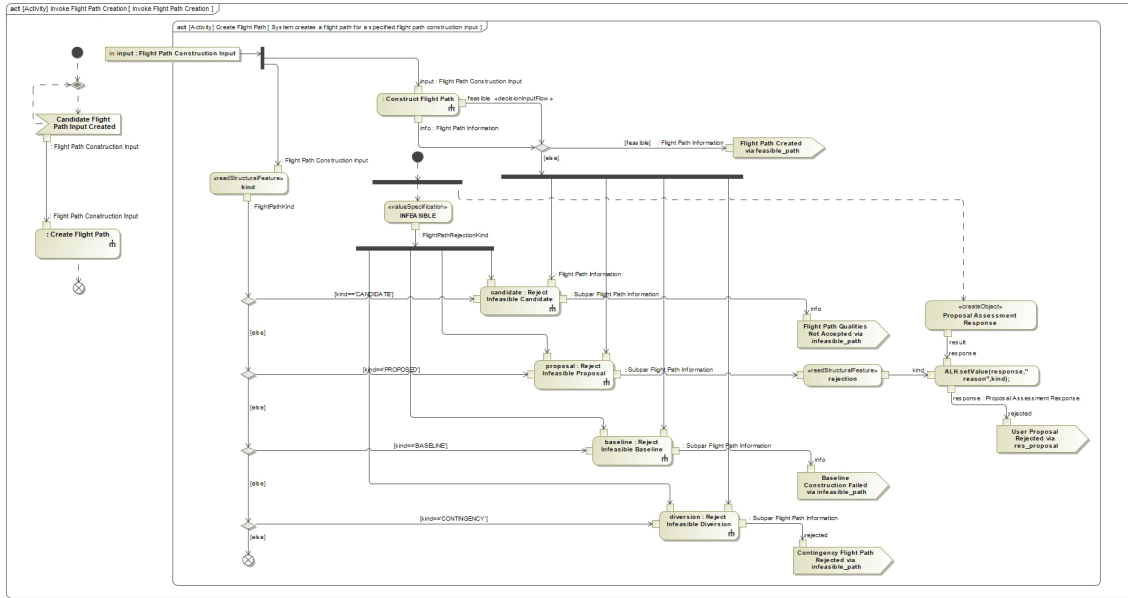


Figure 53: Flight Path Creator Behavior

7.3.2 Subsystem Structure

The structure of the flight path creator is defined by the information flow between its parts:

1. Flight path constructor
2. Health reporter

Figure 54 depicts the internal structure of the flight path creator.

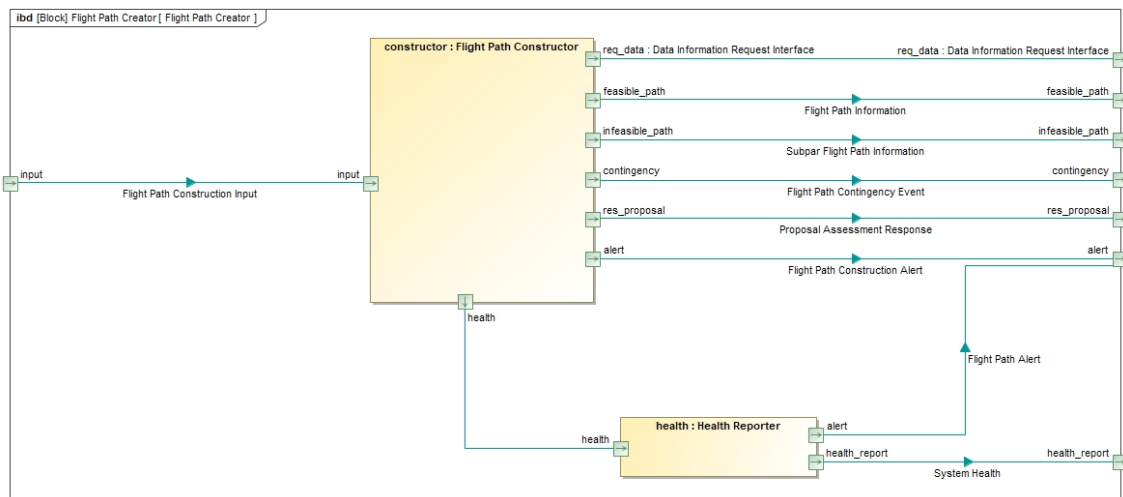


Figure 54: Flight Path Creator Structure

7.4 Flight Path Monitor

The flight path monitor is an implementation-agnostic representation of the flight path monitoring subsystem of the DPP automation system. The basis of the behavioral and structural elements of the flight path monitor logical architecture is the set of system requirements allocated to the flight path monitoring subsystem. Figure 55 shows an abridged version of these requirements (see §5 for the complete requirement statements).

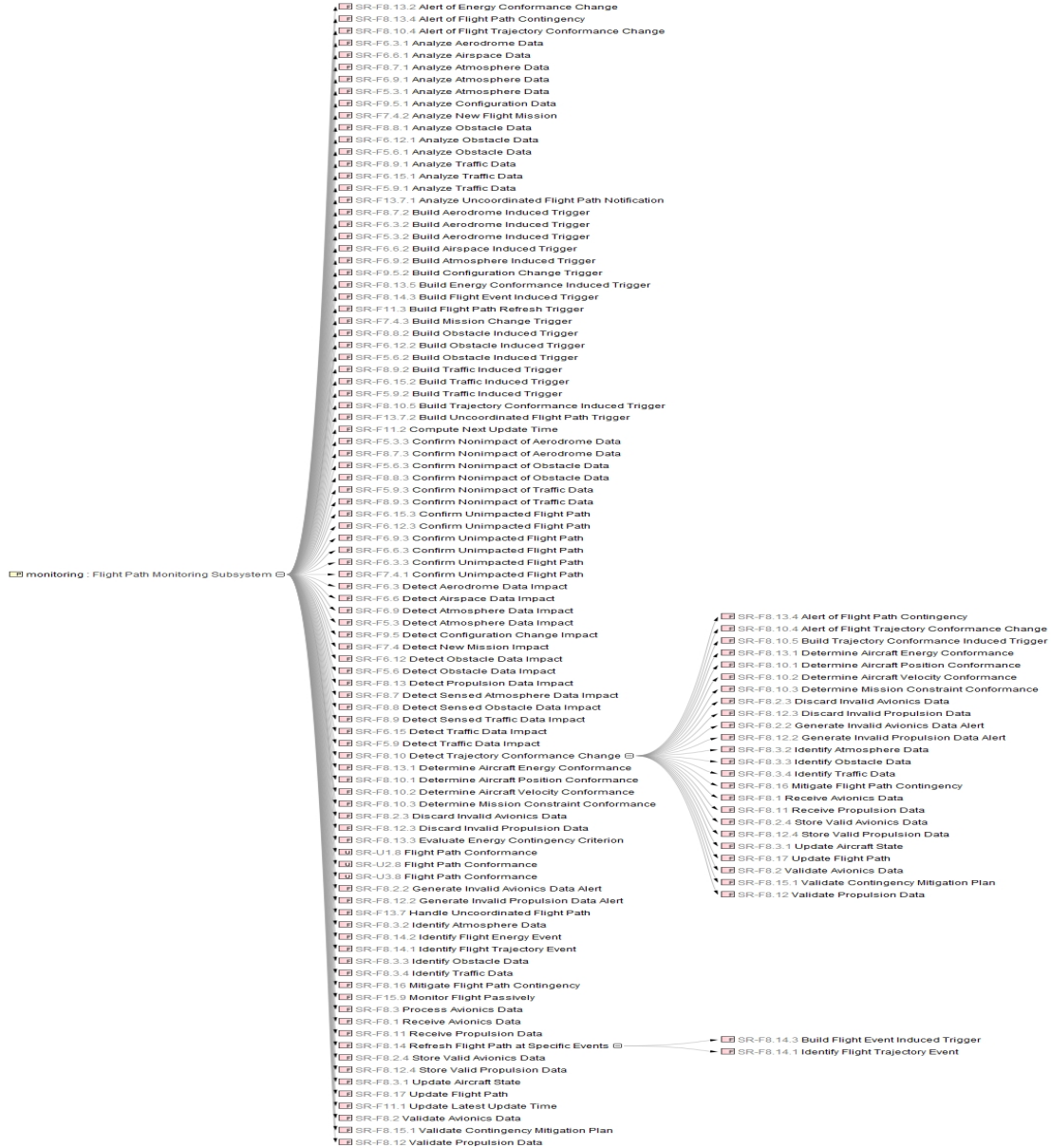


Figure 55: Flight Path Monitor Requirements

7.4.1 Subsystem Behavior

The behavior of the flight path monitor includes:

- (a) monitoring changes to flight mission, operating environment, sensed information, and system configuration;
- (b) monitoring onboard events and user-proposed flight path changes;
- (c) monitoring the currency of the current flight path;
- (d) detecting a need to update the current flight path;
- (e) consolidating flight path update triggers;
- (f) handling uncoordinated flight path;
- (g) evaluating flight path update performance, and
- (h) reporting self-health periodically.

Figure 56 depicts an example behavior of the flight path monitor.

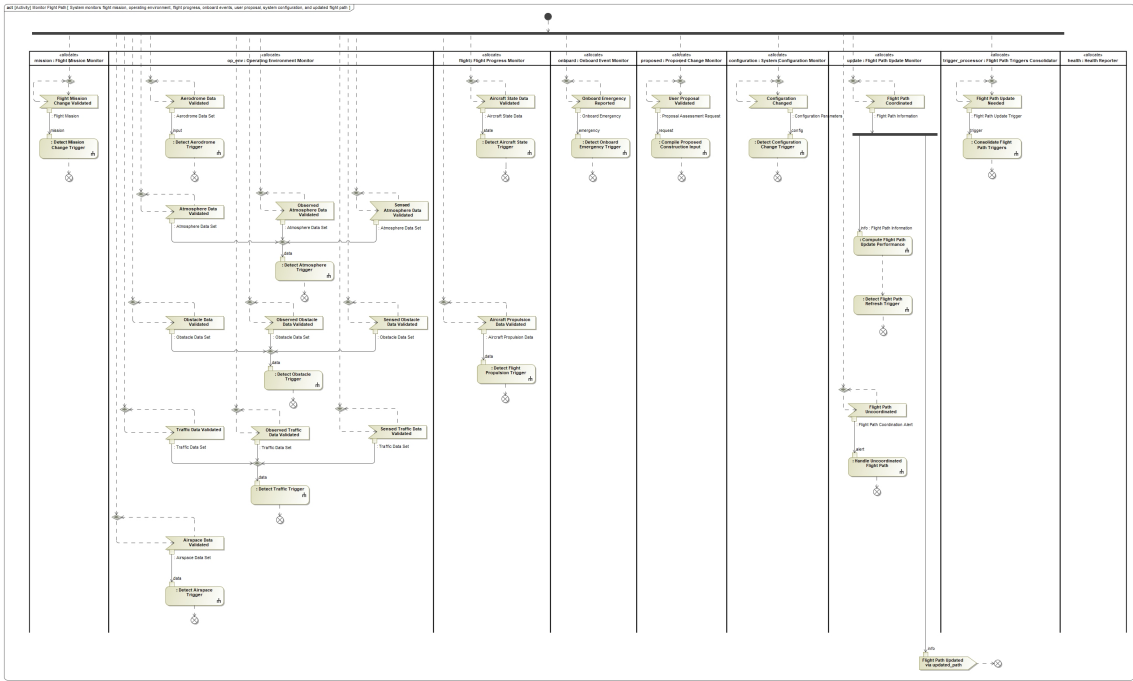


Figure 56: Flight Path Monitor Behavior

7.4.2 Subsystem Structure

The structure of the flight path monitor is defined by the information flow between its parts:

1. Flight mission monitor
2. Operating environment monitor
3. Flight progress monitor
4. Onboard event monitor

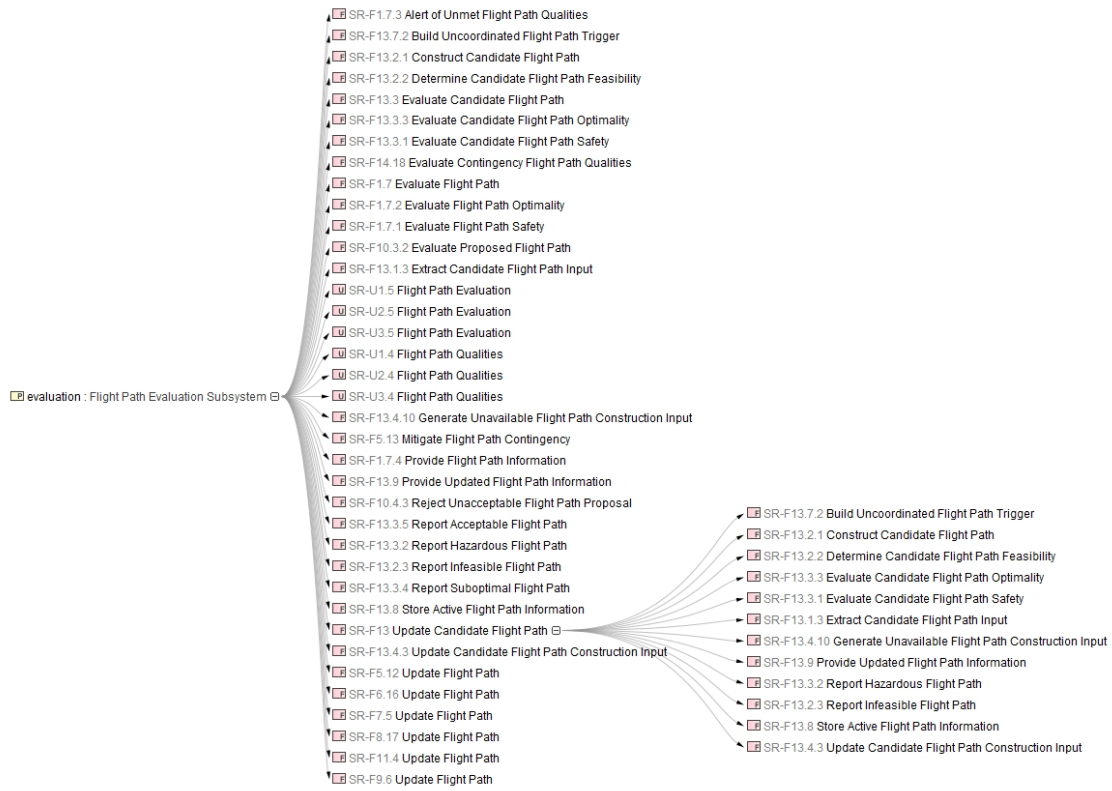


Figure 58: Flight Path Evaluator Requirements

7.5.1 Subsystem Behavior

The behavior of the flight path evaluator includes:

- modeling operating environment hazards,
- detecting conflicts against known hazards,
- alerting the requestor of flight path conflicts,
- computing optimality index of a conflict-free and noncontingent flight path,
- alerting the requestor of a suboptimal flight path, and
- reporting self-health periodically.

Figure 59 depicts an example behavior of the flight path evaluator.

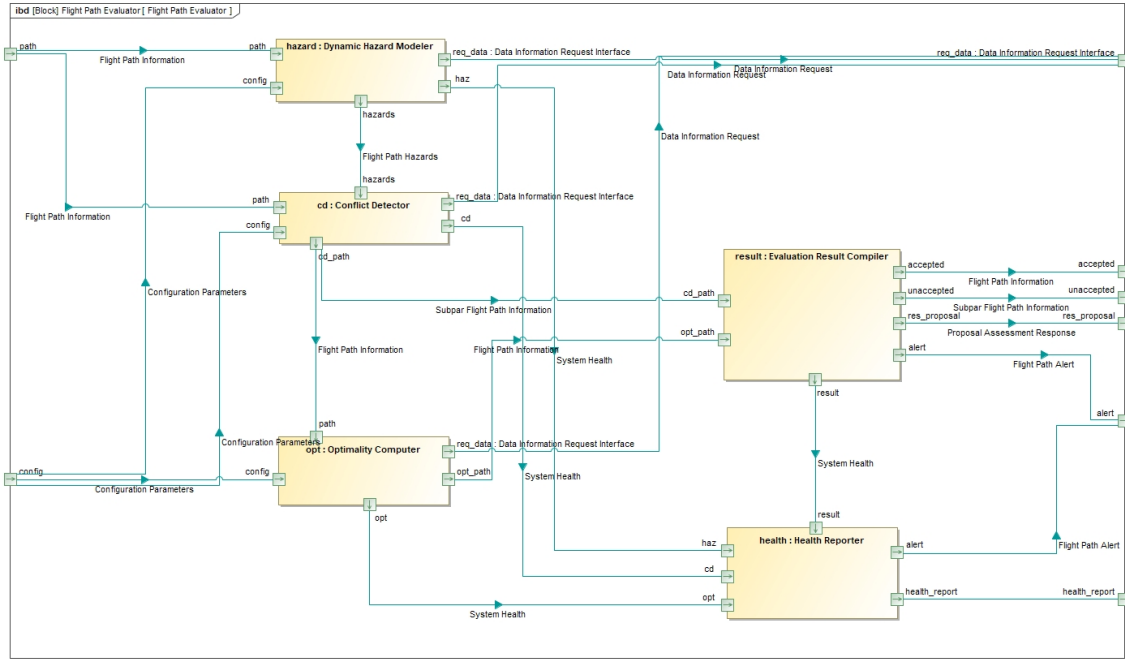


Figure 60: Flight Path Evaluator Structure

7.6 Flight Path Revisor

The flight path revisor is an implementation-agnostic representation of the flight path revision subsystem of the DPP automation system. The basis of the behavioral and structural elements of the flight path revisor logical architecture is the set of system requirements allocated to the flight path revision subsystem. Figure 61 shows an abridged version of these requirements (see §5 for the complete requirement statements).

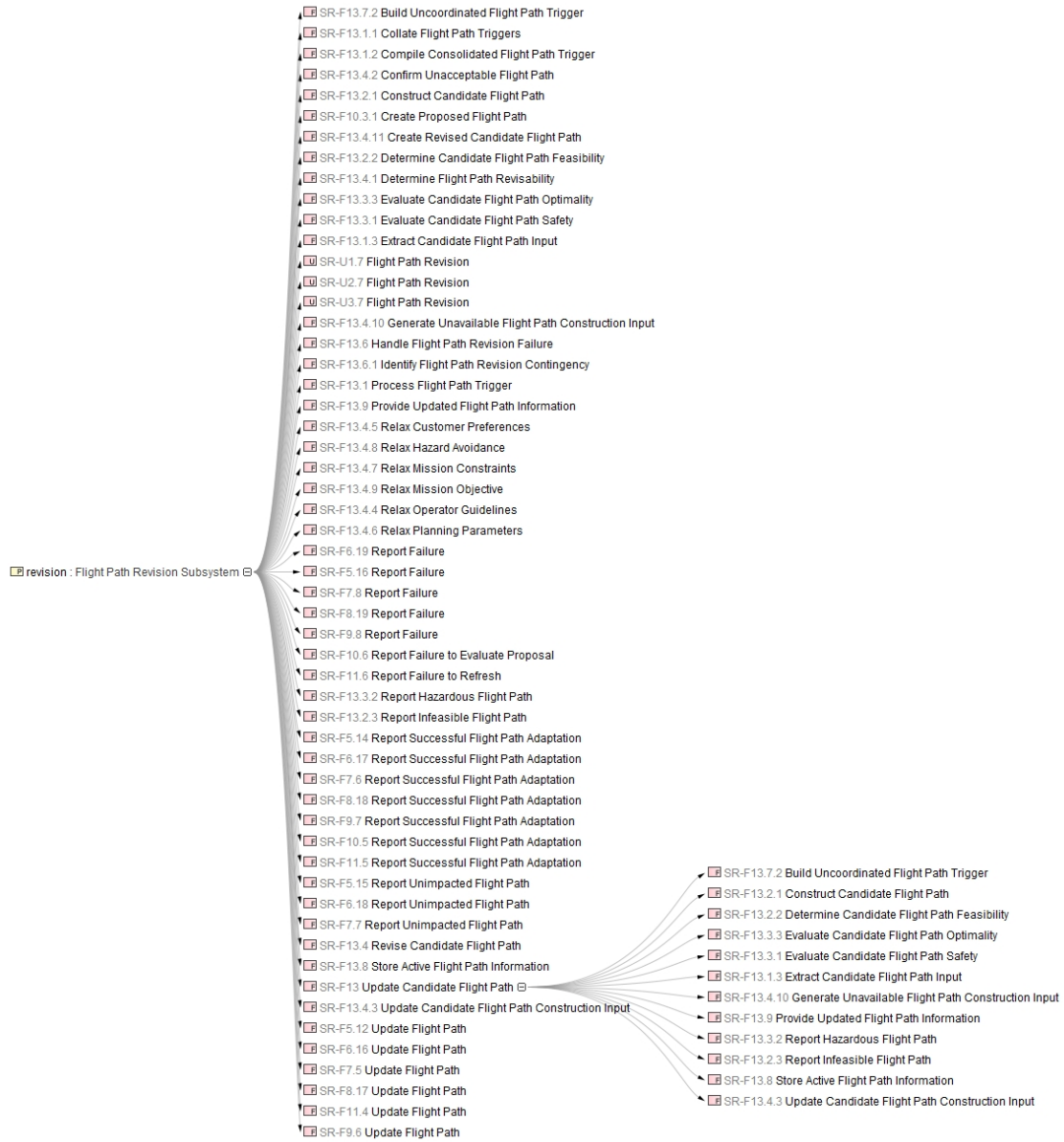


Figure 61: Flight Path Revisor Requirements

7.6.1 Subsystem Behavior

The behavior of the flight path revisor includes:

- (a) revising the candidate flight path construction input in order to improve flight path qualities,
- (b) creating a revised candidate flight path,
- (c) handling failure to revise a flight path due to unavailability of mutable parameters, and
- (d) reporting self-health periodically.

Figure 62 depicts an example behavior of the flight path revisor.

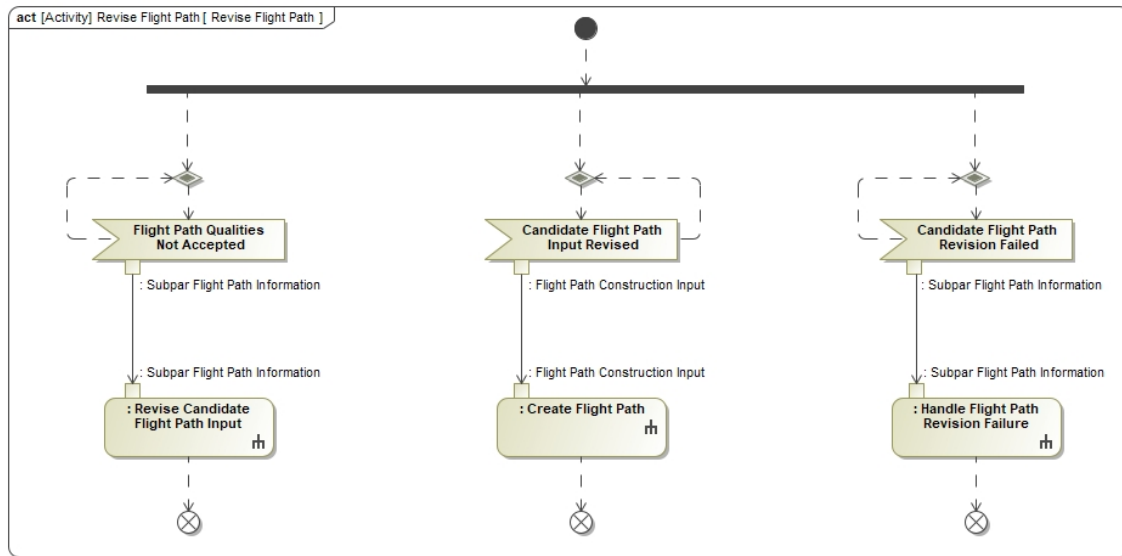


Figure 62: Flight Path Revisor Behavior

7.6.2 Subsystem Structure

The structure of the flight path revisor is defined by the information flow between its parts:

1. User preference relaxation manager
2. Flight path constraint relaxation manager
3. Health reporter

Figure 63 depicts the internal structure of the flight path revisor.

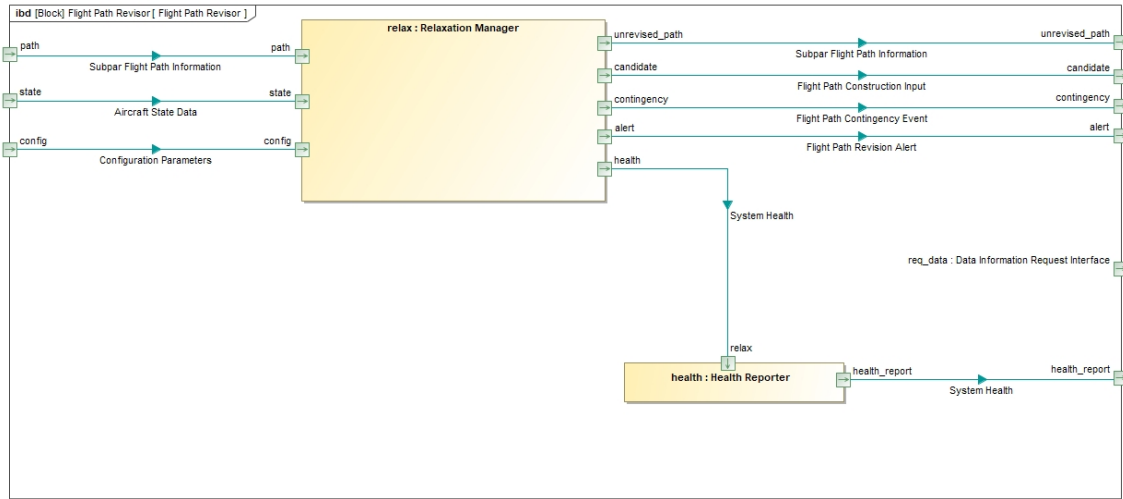


Figure 63: Flight Path Revisor Structure

7.7 Flight Path Coordinator

The flight path coordinator is an implementation-agnostic representation of the flight path coordination subsystem of the DPP automation system. The basis of the behavioral and structural elements of the flight path coordinator logical architecture is the set of system requirements allocated to the flight path coordination subsystem. Figure 64 shows an abridged version of these requirements (see §5 for the complete requirement statements).

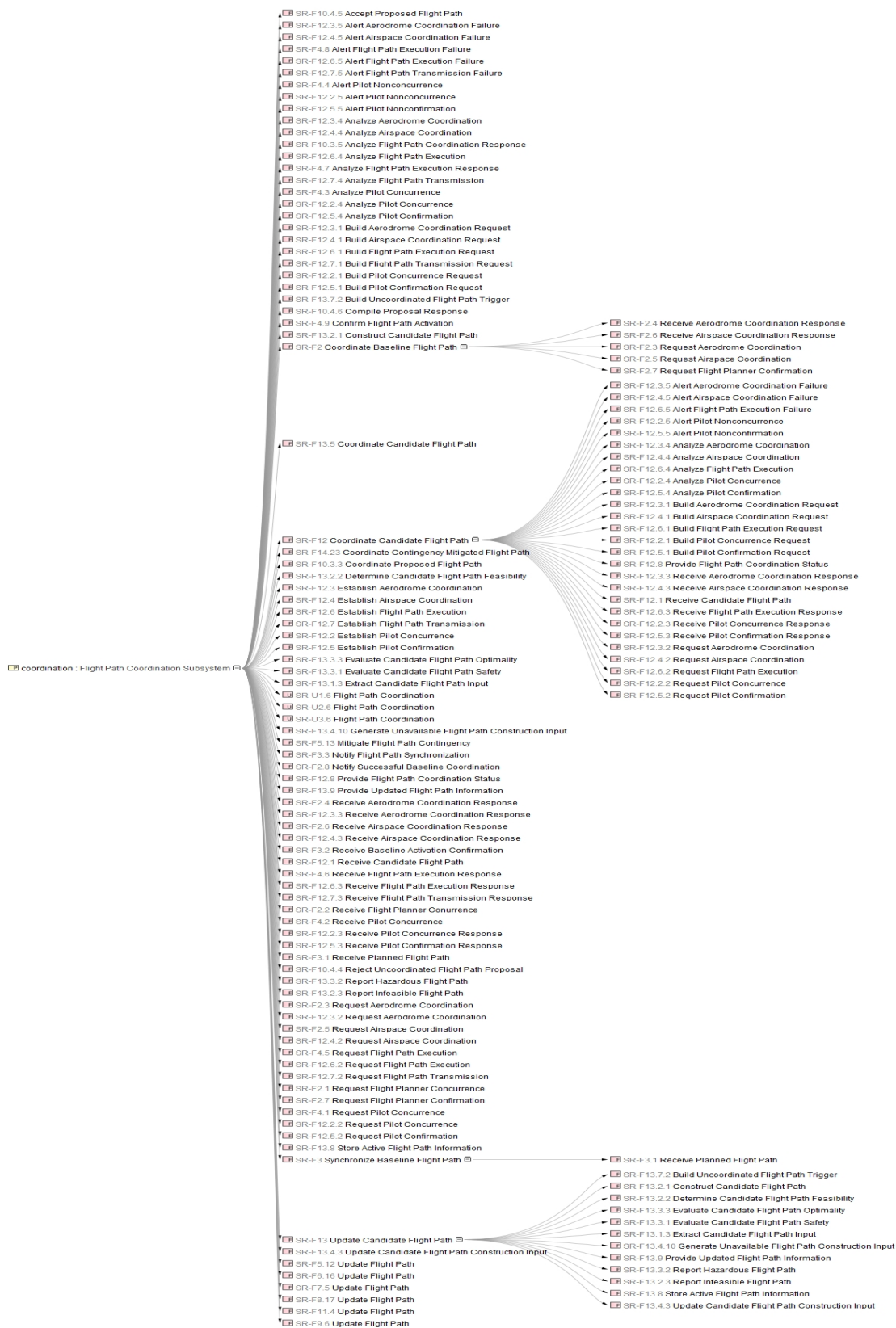


Figure 64: Flight Path Coordinator Requirements

7.7.2 Subsystem Structure

The structure of the flight path coordinator is defined by the information flow between its parts:

1. Flight planner request generator
2. Onboard pilot request generator
3. Aerodrome coordination request generator
4. Airspace coordination request generator
5. Aircraft systems request generator
6. Coordination response analyzer
7. Health reporter

Figure 66 depicts the internal structure of the flight path coordinator.

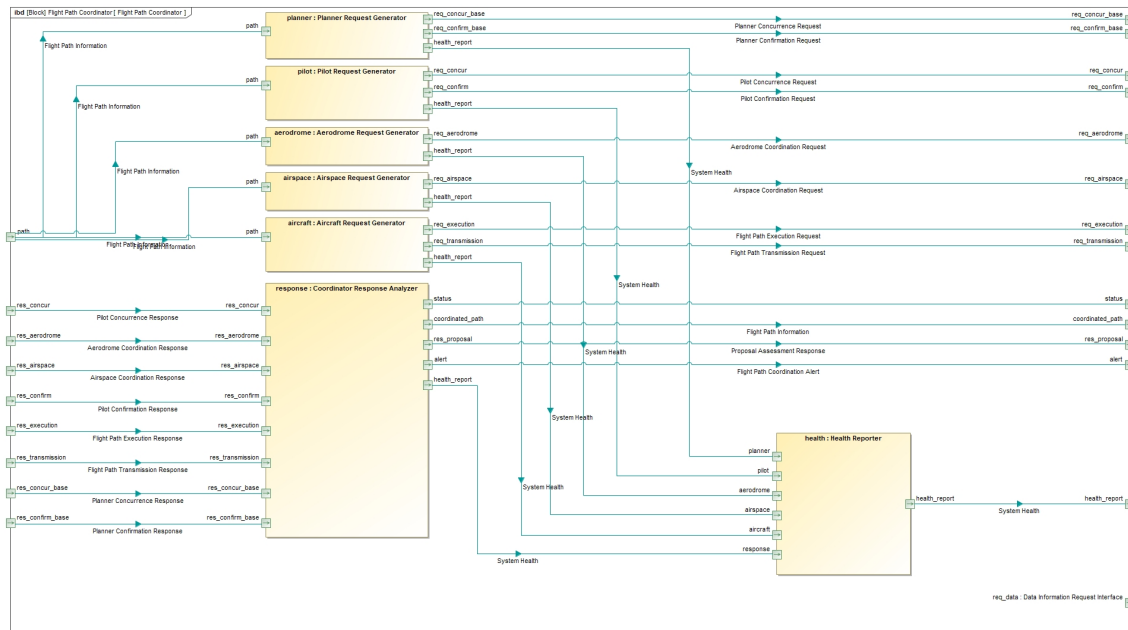


Figure 66: Flight Path Coordinator Structure

7.8 Input Data Processor

The input data processor is an implementation-agnostic representation of the input data processing subsystem of the DPP automation system. The basis of the behavioral and structural elements of the input data processor logical architecture is the set of system requirements allocated to the input data processing subsystem. Figure 67 shows an abridged version, separated into two columns for readability, of these requirements (see §5 for the complete requirement statements).

7.8.1 Subsystem Behavior

The behavior of the input data processor includes:

- (a) validating the incoming user requests and operating environment data,
- (b) informing the appropriate subsystems of valid input, and
- (c) reporting self-health and the health of the system interfaces periodically.

Figure 68 depicts an example behavior of the input data processor.

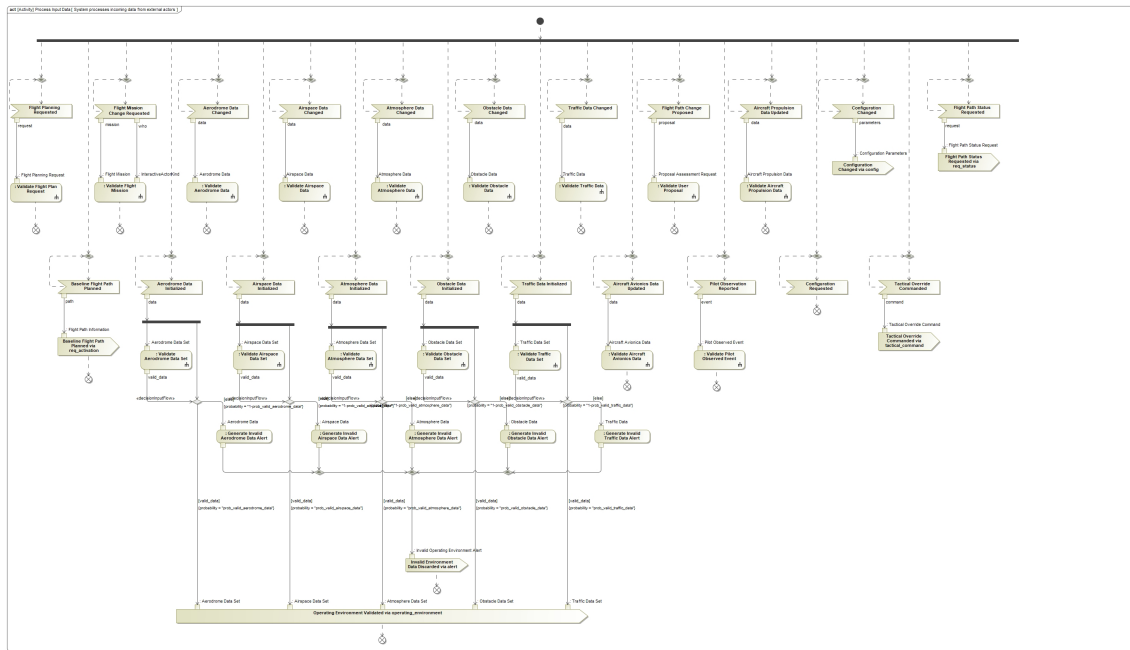


Figure 68: **Input Data Processor** Behavior

7.8.2 Subsystem Structure

The structure of the input data processor is defined by the information flow between its parts:

1. Mission manager input processor
2. Flight planner input processor
3. Onboard pilot input processor
4. Remote pilot input processor
5. Aerodrome coordinator input processor
6. Airspace coordinator input processor
7. Aircraft avionics input processor
8. Aircraft propulsion input processor
9. Aircraft tactical input processor
10. Aircraft guidance input processor
11. Aircraft communication input processor

- Figure 69 depicts the internal structure of the input data processor.



The output data processor is an implementation-agnostic representation of the output data processing subsystem of the DPP automation system. The basis of the behavioral and structural elements

of the output data processor logical architecture is the set of system requirements allocated to the output data processing subsystem. Figure 70 shows an abridged version of these requirements (see §5 for the complete requirement statements).

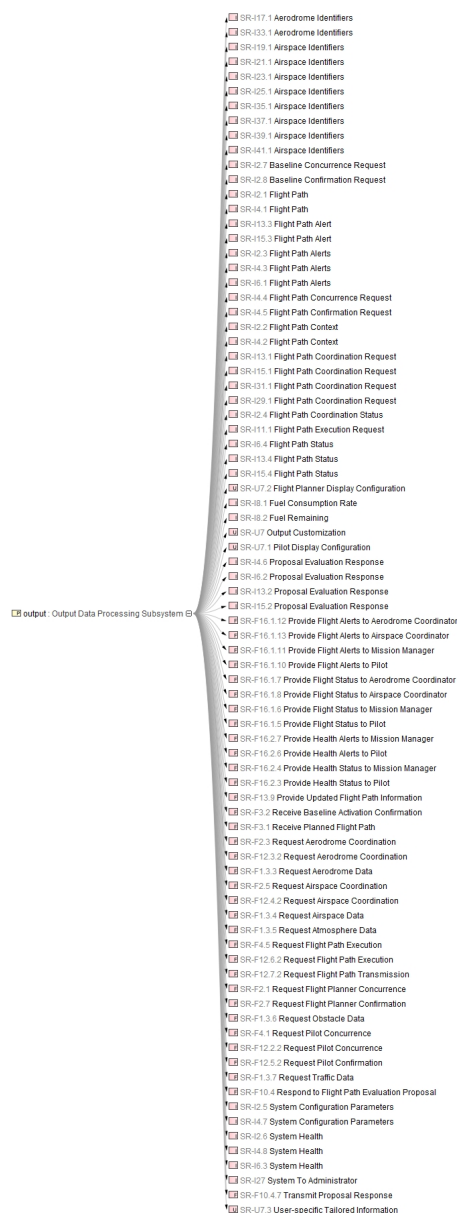


Figure 70: **Output Data Processor** Requirements

7.9.1 Subsystem Structure

The structure of the output data processor is defined by the information flow between its parts:

1. Output target selector
2. Mission manager output compiler

3. Flight planner output compiler
4. Onboard pilot output compiler
5. Remote pilot output compiler
6. Aerodrome coordinator output compiler
7. Airspace coordinator output compiler
8. Aircraft guidance output compiler
9. Aircraft communication output compiler
10. Aerodrome output compiler
11. Airspace output compiler
12. Atmosphere output compiler
13. Obstacle output compiler
14. Traffic output compiler
15. Health reporter

Figure 71 depicts the internal structure of the output data processor.

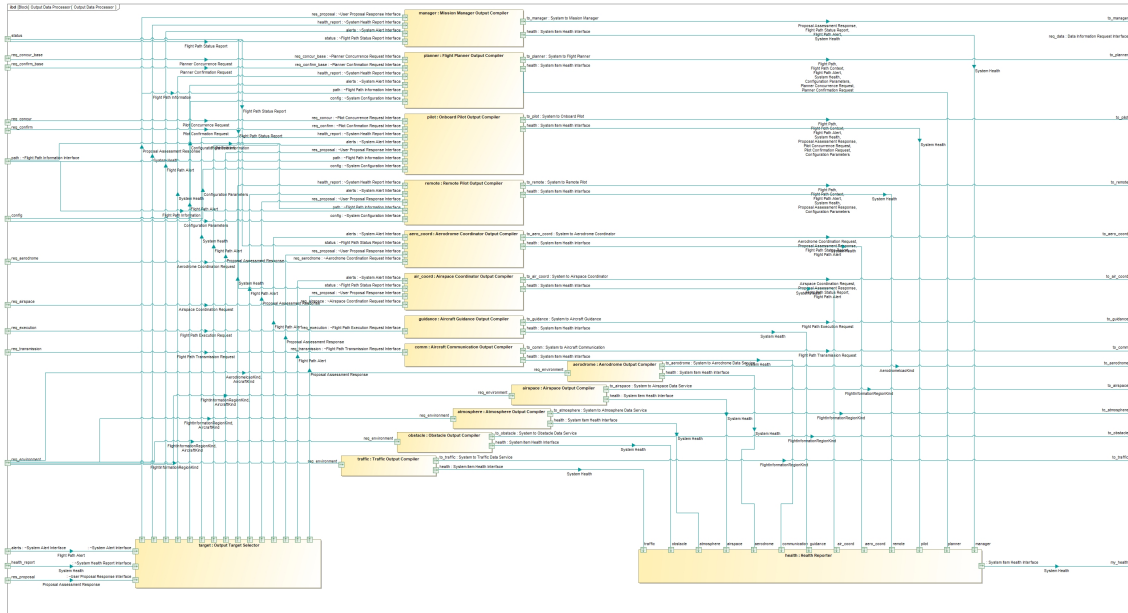


Figure 71: Output Data Processor Structure

7.10 Data Repository Manager

The data repository manager is an implementation-agnostic representation of the data repository subsystem of the DPP automation system. The basis of the behavioral and structural elements of the data repository manager logical architecture is the set of system requirements allocated to the data repository subsystem. Figure 72 shows an abridged version of these requirements (see §5 for the complete requirement statements).

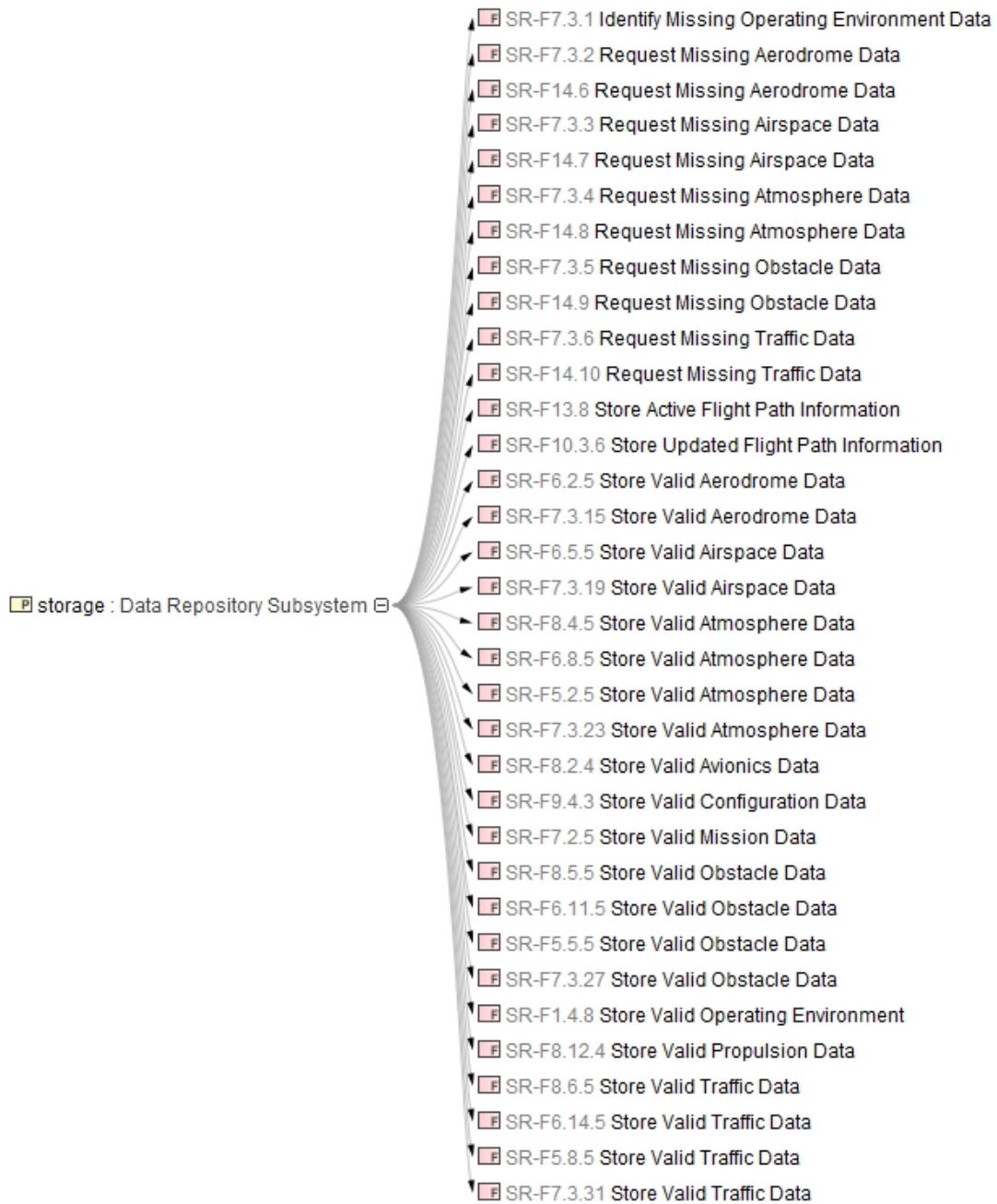


Figure 72: Data Repository Manager Requirements

7.10.1 Subsystem Structure

The structure of the data repository manager is defined by the information flow between its parts:

1. Flight data manager
2. Operating environment data manager
3. Static data manager
4. Flight planning data manager
5. System status reporter
6. Health reporter

Figure 73 depicts the internal structure of the data repository manager.

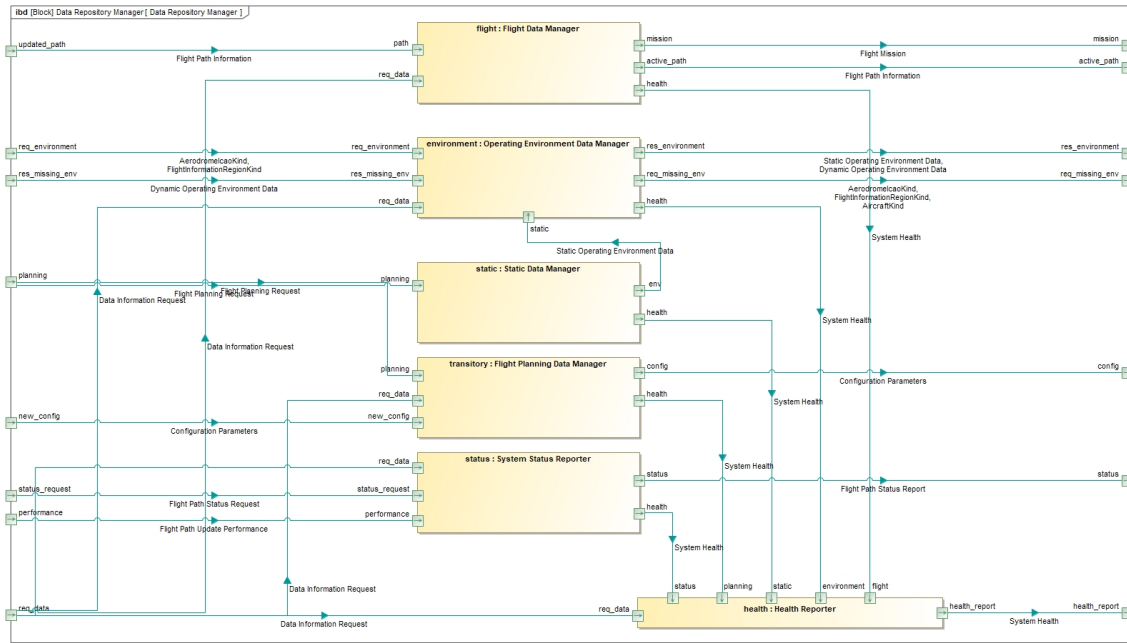


Figure 73: Data Repository Manager Structure

7.11 Flight Contingency Manager

The flight contingency manager is an implementation-agnostic representation of the flight contingency mitigation subsystem of the DPP automation system. The basis of the behavioral and structural elements of the flight path monitor logical architecture is the set of system requirements allocated to the flight path monitoring subsystem. Figure 74 shows an abridged version of these requirements (see §5 for the complete requirement statements). A logical architecture of the flight contingency manager requires a complete and accurate concept of the flight contingency mitigation subsystem, which is a work in progress.

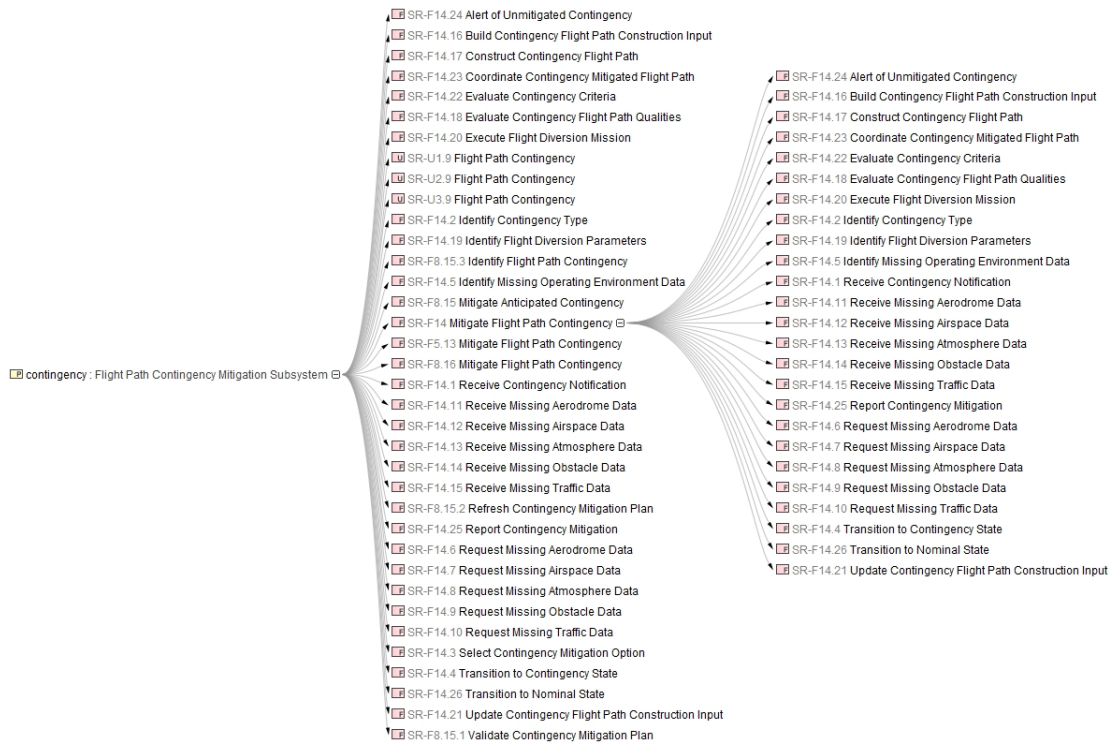


Figure 74: Flight Contingency Manager Requirements

7.12 System Alerts Manager

The system alert manager is an implementation-agnostic representation of the alert management subsystem of the DPP automation system. The basis of the behavioral and structural elements of the system alert manager logical architecture is the set of system requirements allocated to the alert management subsystem. Figure 75 shows an abridged version of these requirements (see §5 for the complete requirement statements). A logical architecture of the system alert Manager requires a complete and accurate concept of the alert management subsystem, which is a work in progress.

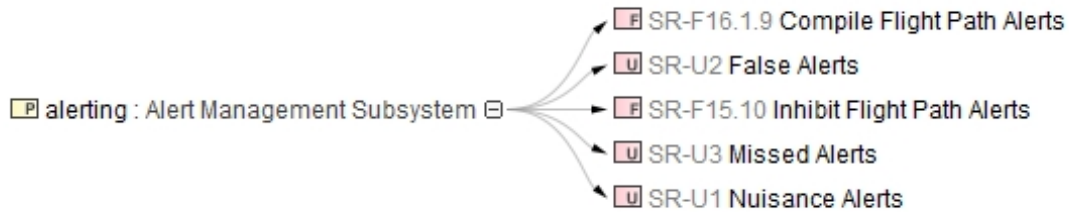


Figure 75: System Alerts Manager Requirements

7.13 System Health Monitor

The system health monitor is an implementation-agnostic representation of the health monitoring subsystem of the DPP automation system. A logical architecture of the system health monitor requires a complete and accurate concept of the health monitoring subsystem, which is a work in progress.

7.14 System Operation Manager

The system operation manager is an implementation-agnostic representation of the operation management subsystem of the DPP automation system. The basis of the behavioral and structural elements of the system operation manager logical architecture is the set of system requirements allocated to the operation management subsystem. Figure 76 shows an abridged version of these requirements (see §5 for the complete requirement statements). A logical architecture of the system operation manager requires a complete and accurate concept of the operation management subsystem, which is a work in progress.



Figure 76: System Operation Manager Requirements

8 Future Work

The DPP automation system architecture presented in this document is a work in progress. This section suggests a few important next steps to refine and complete the architecture.

Conceptual System Architecture: The current document identifies the following additional use cases, based on an analysis of the stakeholder concerns, whose details were not paramount to an initial version of the system architecture and thus have been deferred to future work:

- Comply with tactical maneuver commands
- Monitor system health
- Monitor flight path status
- Monitor system performance
- Perform postflight analysis

A revised version of this document will also need to expand on the contingency mitigation architecture of the DPP automation system as the underlying concept evolves. Further research and coordination with the DPP automation system stakeholders will need to be conducted in order to validate the current measures of effectiveness and to establish additional ones as needed.

Functional System Architecture: The revisions to the conceptual system architecture will spawn corresponding revisions to the functional and structural decompositions, subsequently producing an updated set of DPP automation system requirements.

Logical System Architecture: The current document identifies the following subsystems that are necessary to meet the DPP automation system requirements, whose details were not paramount to an initial version of the system architecture and thus have been deferred to future work:

- Flight Contingency Manager
- System Alerts Manager
- System Health Monitor
- System Operation Manager

A revised version of this document will also need to address the revisions to the DPP automation system requirements based on the updated conceptual and functional architectures. The logical architectures of the DPP automation subsystems will need to be refined until their structural and behavior elements satisfy every system requirement; this includes: defining detailed subsystem requirements; defining subsystem measures of performance; establishing traceability to the system requirements.

Physical System Architecture: A physical architecture represents an implementation-specific structure of the system that meets the requirements defined in the logical system and subsystem architectures. Since it straddles the boundary between the system architecture and the system design phases of the systems engineering process, the details of a physical architecture are considered outside the scope of this document. However, a revised version of this document should specify the system implementation requirements, derived from the logical architecture of the subsystems, which will ensure that the implemented system meets the system requirements, and transitively the stakeholder needs.

System Architecture Development Methodology: The DPP automation system architecture exemplified a successful application of the new MBSE approach initially developed by the author for this project. The versatility of the modeling approach needs to be established by architecting systems and operations beyond the aviation domain. The methodology, though agnostic to the modeling language and modeling tool, must continue to evolve in order to take advantage of the upcoming new features and improvements in the language (SysML) and the tool used in this project (MagicDraw). Specifically, the approach needs to exploit the automated verification and validation capabilities of the modeling tools to showcase the phenomenal value of MBSE.

Research: NASA aims to leverage the presented architecture to design a reference DPP automation system, in support of the verification and validation of the AAM concepts related to flight path management. This architecture document, as well as the ensuing design documents, will also further the AAM operators' understanding of the automation capabilities that will be required to operate safely, cooperatively, and efficiently in the increasingly complex AAM operating environment. The DPP automation system architecture will offer guidance to the industry, certification authorities, and service providers in establishing best design practices, community standards, and means of compliance recommendations for industry-wide adoption of dynamic path planning automation systems. The presented system architecture is extensible to flight operations beyond the AAM operational environment, and will be used to study the impact of automation on such operations.

9 Summary

This document presented an architecture of the dynamic path planning automation system previously conceptualized and documented in the context of an AAM domain. The purpose of the DPP automation system is to build, coordinate, and maintain a safe, efficient, and operationally acceptable flight path throughout the flight. To meet the challenges of AAM operations, the DPP automation system must not only adapt autonomously to the dynamic operating environment, but also respond to potential mission changes, aircraft commands, and user instructions. A comprehensive, rigorous, and flexible architecture is the key to meeting these complex requirements in the design and development of future DPP automation systems. A new, versatile, model-based approach was developed and employed to architect such a system from first principles. The key steps in this approach are the development of (a) conceptual, (b) functional, (c) logical, and (d) physical architectures of the DPP automation system to a level of depth that is suitable for designing the system.

A conceptual architecture represents the formalization of a system concept in a selected operational environment. The conceptual architecture of the DPP automation system was developed in the context of AAM operations. The key stakeholders of the DPP automation system were identified to be the pilots, mission managers, flight planners, flight path coordinators, aircraft, passengers, cargo shippers, impacted communities, standards development organizations, and regulatory authorities. Their primary concerns were related to the desired capabilities and performance of the system while complying with the operational and design constraints. The DPP automation system goals were derived from the stakeholder concerns. A set of use cases and measures of effectiveness were identified to refine the functional and nonfunctional goals of the system, respectively, while the system context was defined to meet the interface goals. The use case specifications, the system context, the measures of effectiveness, and the design goals provided a list of top-level requirements of the DPP automation system.

A functional architecture represents an elaboration of the conceptual architecture because the top-level system requirements do not generally provide the level of depth needed to develop a logical system architecture. The development of the DPP automation system functional architecture constituted: decomposing the use case behaviors into system activities; defining a set of interrelated functions which are responsible for performing these activities; identifying the interfaces and data communication between these functions; and deriving the detailed system requirements from these behavioral and structural elements.

A logical architecture represents the first step toward solving the problem formulated in the conceptual and functional architectures. It defines an implementation-agnostic structure of the system that performs the system behaviors in order to satisfy the specified system requirements. The development of the DPP automation system logical architecture constituted: defining the AAM domain architecture of relevance to dynamic path planning; outlining the system behavior at its highest level; defining a system structure to manage its logical parts; identifying the system measures of performance to evaluate the system capabilities; and allocating the system requirements to the logical parts, i.e., subsystems. The logical architecture of the DPP automation system was elaborated with the logical architectures of the system parts, namely, the flight path initialization, flight path creation, flight path monitoring, flight path revision, flight path coordination, input data processing, output data processing, flight contingency management, and supporting subsystems.

The DPP automation system architecture presented here is a work in progress. The document offered suggestions to refine and complete the system architecture.

References

- [1] *Advanced Air Mobility | Air Taxis*. FAA, June 2021. <https://www.faa.gov/air-taxis>.
- [2] *Advanced Air Mobility Mission*. NASA, Oct. 2024. <https://www.nasa.gov/mission/aam/>.
- [3] *Advancing Aerial Mobility: A National Blueprint*. National Academies of Sciences, Engineering, and Medicine 2020. <https://doi.org/10.17226/25646>.
- [4] Aleksandravičienė, Aistė and Morkevičius, Aurelijus. *A Practical Guide to Systems Modeling using MagicGrid from Dassault Systèmes*. Nov. 2021. ISBN: 978-609-454-554-2.
- [5] Cockburn, Alistair. *Writing Effective Use Cases*. Addison-Wesley Longman Publishing Co., Inc., Oct. 2000. ISBN: 978-0-201-70225-5.
- [6] *Dassault Systèmes CATIA MagicDraw*. 2022. <https://www.3ds.com/products/catia/no-magic/magicdraw>.
- [7] Delligatti, Lenny. *SysML Distilled: A Brief Guide to the Systems Modeling Language*. 2nd ed. Nov. 2013. ISBN: 978-0-321-92786-6.
- [8] *INCOSE Object-Oriented Systems Engineering Method (OOSEM)*. OMG Standards Development Organization, Apr. 2011. <https://www.omgwiki.org/MBSE/doku.php?id=mbse:incoseoosem>.
- [9] *OMG Systems Modeling Language (SysML)*. 2025. <https://www.omgsysml.org/>.
- [10] Sharma, Vivek and Wing, David J. *Dynamic Path Planning Automation Concept for Advanced Air Mobility*. NASA/TM-20220009974, NASA Langley Research Center, Hampton, VA, Sept. 2022.
- [11] Sproles, Noel. “Formulating measures of effectiveness”. *Systems Engineering*. Vol. 5(4). Wiley Periodicals, Inc., May 2002, pp. 253–263.
- [12] Wing, David J. et al. *The Viability of Digital Flight, Part 2: Requiring Performance for Conflict Management*. NASA/TM-20250007129, NASA Langley Research Center, Hampton, VA, Aug. 2025.

Appendix A

Abbreviations

AAM	Advanced Air Mobility
ANSP	Air Navigation Service Provider
DPP	Dynamic Path Planning
FAA	Federal Aviation Administration
ICD	Interface Control Document
INCOSE	International Council on Systems Engineering
MBSE	Model-Based Systems Engineering
MOE	Measure of Effectiveness
MOP	Measure of Performance
NASA	National Aeronautics and Space Administration
OOSEM	Object-Oriented Systems Engineering Method

Appendix B

Glossary

Aerodrome Coordinator An entity, either human or automated, which provides coordination services among operators and aircraft at one or more aerodromes and other potential takeoff and landing sites.

Airspace Coordinator An entity, either human or automated, which provides coordination services among operators and aircraft in a geographic or otherwise operationally defined region.

Dynamic Path Planning The process of building and maintaining a safe, efficient, and operationally acceptable flight path throughout the flight.

Flight Planner The entity responsible for the planning of a flight path to execute a specified flight mission.

Mission Manager The entity responsible for the management of scheduled and on-demand flight missions of one or more aircraft.

Stakeholder The entity with a vested interest in the definition, design, implementation, and operation of a system.

Stakeholder Actor A type of stakeholder, either human or automation, who interacts with the system of interest.

Supporting Actor An entity, either human or automation, which interacts with the system of interest and is not a stakeholder.

Measures of Effectiveness The system attributes that indicate how well the system achieves its intended purpose. They are tied to the overall objectives and mission of the system, often reflecting stakeholder expectations and needs.

Measures of Performance The system attributes that indicate how well the system performs its specific functions. They are concrete, technical measures that relate to the system's functionality and performance.

System Conceptual Architecture A formalization of the system concept in a selected operational environment. It constitutes: identifying the primary stakeholders and their concerns; analyzing the stakeholder concerns to derive system goals; defining the system boundary and interfaces in the AAM domain; refining the system goals via use cases and measures of effectiveness; and extracting the top-level system requirements from the identified use cases and measures of effectiveness.

System Functional Architecture An elaboration of the conceptual architecture because the top-level requirements of a complex system do not generally provide the level of depth needed to develop a logical system architecture. It constitutes: decomposing the system behavior; defining the system functions which are responsible for performing these activities; identifying the interfunctional interfaces; and deriving detailed system requirements.

System Logical Architecture The first step toward solving the problem formulated in the conceptual and functional architectures. It defines an implementation-agnostic structure of the system that performs the system behaviors in order to satisfy the specified system requirements.