Architecture and interface specification

Document information	
Project title	Baseline for airport controller tools
Project N°	12.04.01
Project Manager	NATMIG
Deliverable Name	Architecture and interface specification
Deliverable ID	D04
Edition	00.02.00

Authoring & Approval

Prepared By		
Name & company	Position / Title	Date
[redacted] NATMIG	[redacted]	2011-05-16
[redacted] NATMIG		2011-09-22

Reviewed By		
Name & company	Position / Title	Date
[redacted] NATMIG (Saab AB)	[redacted]	28/9/2011
[redacted] National Air Traffic Services (NATS), UK		22/9/2011
[redacted] Selex Systems Integration Ltd		28/9/2011

Approved By		
Name & company	Position / Title	Date
[redacted] NATMIG (Saab AB)	[redacted]	29/9/2011
[redacted] National Air Traffic Services (NATS), UK		22/9/2011
[redacted] Selex Systems Integration Ltd		29/9/2011

Document History

Edition	Date	Status	Author	Justification
00.00.01	2010-07-26	DRAFT	[redacted]	New Document
00.00.02	2011-05-16	DRAFT		Major updates to match new system requirements
00.00.03	2011-05-25	DRAFT		Update after review comments
00.01.00	2011-05-26	Initial Issue		Review complete.
00.01.01	2011-09-22	Revised document		Updates following SJU assessment
00.02.00	2011-09-28	Second Issue		Review complete

Intellectual Property Rights (foreground)

This deliverable consists of Foreground owned by one or several Members or their Affiliates.

Table of Contents

List of tables

List of figures

1 Introduction

1.1 Purpose of the document

This document describes the architecture of the prototypes developed in project 12.4.1 primarily for threads TH01 and TH03 as defined in [1] and [2].

Since TH02 requirements are not yet fully mature, there may be a need later for an update of the architecture to cover also this thread.

1.2 Intended readership

This document is mainly intended for the project members involved in prototype specification and development.

1.3 Background

The architecture described in this document aims to be compliant with “Architecture of the Technical Systems Description Document for Step 1” [5].

1.4 Acronyms and Terminology

Term	Definition
ACC	ATC Control Centre
ADDEP	Airport Departure Data Entry Panel
ATM	Air Traffic Management
APP	Approach
CFMU	(European) Central Flow management Unit
CTOT	Calculated Take-Off Time
DPI	Departure Planning Information. DPI messages are used for ATFM purposes to provide departure expectation updates to the CFMU and for other network users.
EAEA	European ATM Enterprise Architecture
E-ATMS	European Air Traffic Management System
EFPS	Electronic Flight Progress Strip system
EOBT	Estimated Off-Block Time
ER	En-Route
ETOT	Estimated Take-Off Time
FOIPS	Flight Object Interoperability Proposed Standard
NM	Network Management
NOP	Network Operations Plan

Term	Definition
NOV 5	NATO Operation View (5. Operational Activity Model)
NSV 1	NATO System View (1. System Interface Definition)
SAM	Slot Allocation Message
SESAR	Single European Sky ATM Research Programme
SJU	SESAR Joint Undertaking (Agency of the European Commission)
SJU Work Programme	The programme which addresses all activities of the SESAR Joint Undertaking Agency.
SESAR Programme	The programme which defines the Research and Development activities and Projects for the SJU.
TOBT	Target Off-Block Time
TSAT	Target Start-up Approval Time
TTA	Target Time of Arrival
TTL	Time To Lose
TTOT	Target Take-Off Time

2 Overview

The architecture in this project describes how the Aerodrome ATC System supports the requirements related to project 12.4.1. Chapter 3 describes a generic and implementation independent architecture for an Aerodrome ATC System. Chapters 4 and 5 describes how two variants of the prototype in 12.4.1 are compatible with the general architecture, where chapter 4 describes a low-cost solution for airports without an electronic flight strip system, and chapter 5 describes an extension of an electronic flight strip system. The two variants prove that the architecture and the operational concepts it supports can be implemented on all airports regardless of size.

2.1 Limitations

The document primarily describes a basic Aerodrome ATC System contributing to the Traffic Synchronization Service. The document does not cover a complete Aerodrome ATC System.

The architecture is kept at a high level in this early stage as it is assumed that a lot of refactoring will be needed when EAEA reaches a more mature state. Detailed descriptions of the information model and services are considered to be implementation specific for the time being as there are currently too many uncertainties on the higher levels of EAEA for it to be worthwhile of making detailed descriptions.

There is currently no established Architecture Framework and no supporting tools. This architecture specification is therefore written in a slightly haphazard manner with inspiration from the NATO Architecture Framework.

To date, the specification for SWIM has not been available. It is not within the scope of this project to contribute to the definition of SWIM or to define Enabling Services.

3 General Architecture

3.1 Overview

This chapter describes a generic and implementation independent architecture for an Aerodrome ATC System and how it interacts with its environment. The architecture is limited to describe two capabilities of the Aerodrome ATC System

- Provision of sending flight departure data (DPI messages) to the Regional NM (CFMU)
- Synchronize departing flights with the arrival management process for the destination airport.

The two capabilities have no direct dependencies between them except that they use, or can use, the same inputs from the Tower Controller, and therefore an Aerodrome ATC System can support either capability or both. This chapter describes the capabilities together as they are related on an architectural level.

It should be noted that the Aerodrome ATC Human Machine Interface provisions that are needed to achieve these capability may be similar to, and thus could potentially replace, the current ATC Controller paper strip provisions - for both departures and arrivals.

3.2 Operational Activity Model (NOV-5)

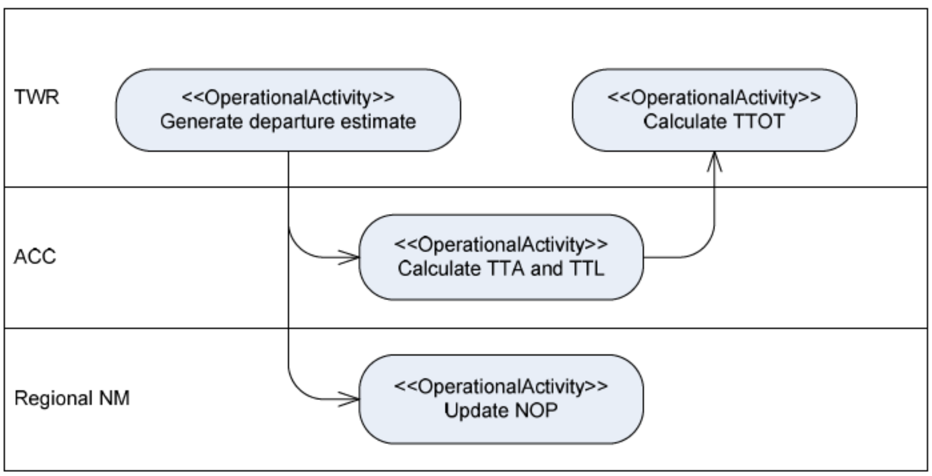

Figure 3-1 Operational activity model (NOV-5)

The TWR calculates an estimate of the take-off time, and optionally expected departure runway and departure route for each departing flight. This could either be done manually or automatically. Either way, the tower sends the departure information to the Regional NM (CFMU) and [1]relevant ACCs. Through them, information becomes available to the destination aerodromes. There may also be other subscribers, but that is out of scope for this architecture document.

The Regional NM uses the departure information to update and refine the Network Operations Plan and to inform other network management related stakeholders.

The destination airport's ACC uses the departure information to allocate a slot in the arrival sequence for the destination airport. Through the slot allocation process, the ACC can deliver a TTA (Target Time of Arrival) and a TTL (Time To Lose) advisory which are sent back to the TWR[2]. The TTL is used by the TWR to calculate a TTOT, either manually or automatically.

[1] The SESAR Architecture does not yet identify if the relationship with the destination airport arrival management will be conducted directly with the destination airport's parent ACC, or via the departure airports parent ACC.
[2] This may be directly, or via the departure airports parent ACC.

3.3 System Interface Description (NSV-1)

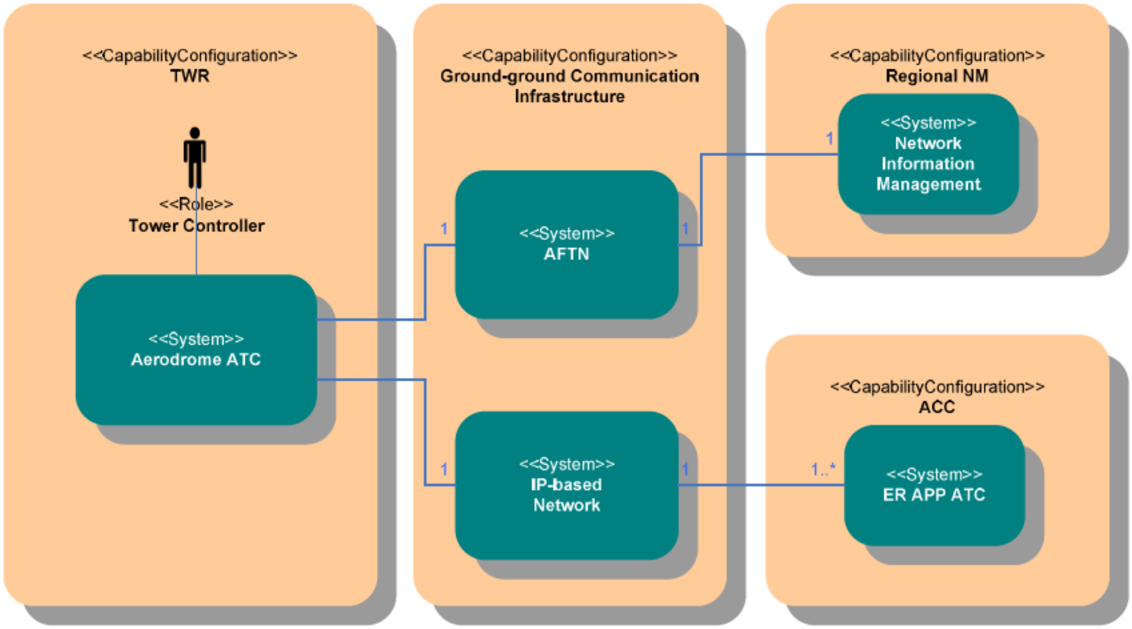

Figure 3-2 System interface description (NSV-1)

The Aerodrome ATC System sends Aerodrome ATC departure data to the Network Information Management System (CFMU) via AFTN and to one or many ER APP ATC Systems via an IP-based network. The Aerodrome ATC System receives arrival data from ER APP ATC Systems via the same IP-based network.

The Aerodrome ATC System also receives initial Flight Plan, Flight Plan Update, and Flight Plan Cancellation Information that is provided from the CFMU. This is an existing provision to aerodromes and is reflected above. However, this provides the base data for the Aerodrome provisions.

3.4 Detailed System Interface Description

The AFTN connection is used for sending DPI messages [6] to the Network Information Management system as mentioned earlier in this document. The types of DPI messages that can be expected from the TWR are implementation specific, but the TWR shall at least be able to send A-DPI messages.

The AFTN connection is also used for other purposes, such as exchanging ATS messages and SAM messages etc.

As many aerodrome ATCs are directly served by their parent ACC's system, the only standard current available for direct data exchanges between an Aerodrome ATC System and an ER APP ATC System is the OLDI standard. This could be a possible candidate where the destination aerodrome is also parented by the same ACC as the departure aerodrome. Where they are not, additional ER APP ATC to ER APP ATC OLDI exchanges, or new ACC to ACC system information exchange mechanisms would be needed to forward the information, and the data exchanges become considerably more complex.

Alternatively new mechanisms could be standardised allowing the Aerodrome ATC System to conduct exchanges directly with the destination aerodrome's ER APP ATC System (e.g. across FIR boundaries).

The proposed solution in this architecture is that the data exchange is based on a publish-subscribe pattern. The details of how data is published will be implementation specific until SWIM and appropriate standards are made available in SESAR.

This document describes the system architecture prior to having any mature inputs from the SESAR WP8 and WPB, or the operationally facing projects 6.6.1, 6.6.2 and 5.6.4. A service oriented approach has been chosen even though the WP8 ISRM has been available as input as should be expected for later stages in the SESAR developments.

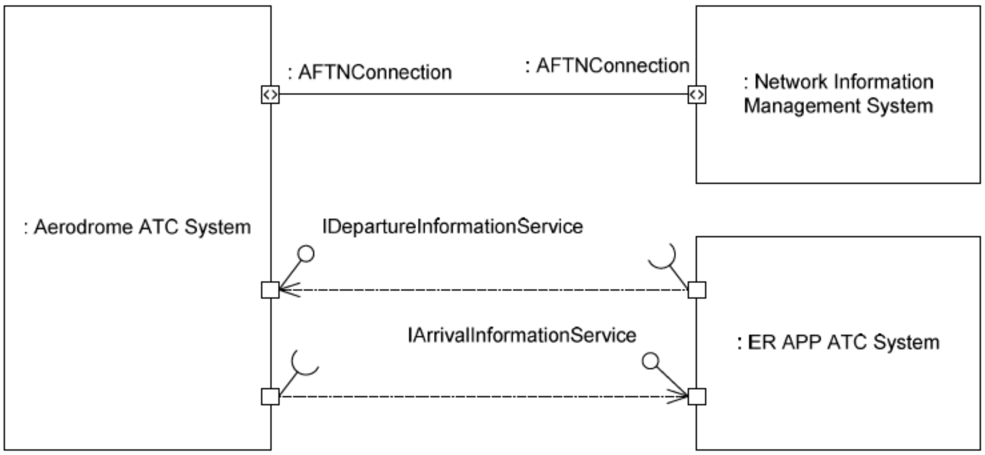

Figure 3-3 System interfaces described in SysML

Port/interface	Description
AFTNConnection	Represents an AFTN connection. The AFTN connection is used for both DPI messages and other types of messages, e.g. flight plan messages. DPI messages are sent on ADEXP format. Other message types can be on either ADEXP or ICAO format, depending on implementation and site specific or global policies.
IDepartureInformationService	An interface through which the Aerodrome ATC System publishes departure data. The ER APP ATC System is a consumer of this departure data.
IArrivalInformationService	An interface through which the ER APP ATC System publishes arrival data. The Aerodrome ATC System is a consumer of this arrival data.

Table 3-1 Port/interface description for Figure 3-3

4 Implementation with ADDEP

4.1 Overview

The Airport Departure Data Entry Panel, ADDEP, is a low cost and simple solution that implements the generic architecture as described in previous chapter. ADDEP works as an electronic complement to traditional paper strips and is not meant to be a full Aerodrome ATC System.

4.2 Operational modes

ADDEP supports two operational modes. The operational mode is configured off-line.

- Basic mode – used to send A-DPI messages to CFMU as described in the operational concept described in [3] and requirements in 12.04.01-D03 Thread 1 – Requirement Specification [1].
- Extended mode – implements the airport side of the operational concept “Handling Origin Airports inside the AMAN Horizon” described in [4] and requirements in 12.04.01-D12 Thread 3 – Requirement Specification [2] section 3.1.1 (Departures from regional airports within AMAN horizon).

In both modes, ADDEP receives flight plan data over AFTN and displays the departing flights in a strip-like fashion in the HMI.

4.2.1 Basic mode

In basic mode the Tower Controller pushes a button on the strip representing the flight when push-back clearance and take-off clearance are given. When a button is pushed, ADDEP generates an A-DPI message which is sent to CFMU (see the DPI Implementation Manual [6]). Basic mode only supports A-DPI messages.

The DPI message is filled with the following information:

Data item	Value
ARCID, EOBD, EOBT, ADEP, ADES	As filed in the flight plan
AOBD, AOBT	The time when the push back button was pressed
TAXITIME	Standard taxi time
TTOT	The time when the take-off button was pressed, or AOBT + TAXITIME when the at push-back.

Table 4-1 DPI message contents in ADDEP basic operational mode

The basic operational mode corresponds to the requirements described in 12.04.01-D03 Thread 1 – Requirement Specification [1].

4.2.2 Extended mode

In the extended mode, the Tower Controller can input runway changes, TOBT, ETOT and TTOT in addition to the inputs in basic mode. The system automatically proposes ETOT as TOBT + EXOT, where EXIT is a standard taxi time, and TTOT as ETOT + TTL. The Tower Controller can either use the automatically proposed values or manually enter other values.

When the Tower Controller enters ETOT or TTOT, the updated departure data is published to be available to AMAN. The same information is used for generating DPI messages. As the Tower Controller explicitly manages ETOT and TTOT, the quality of the data in the messages is expected to be better compared to the basic mode. The extended mode provision may use existing or new DPI messages, or may be provided in a new standard form when defined.

The extended operational mode corresponds to the requirements described in section 3.1.1 (Departures from regional airports within AMAN horizon) in 12.04.01-D12 Thread 3 – Requirement Specification [2].

4.3 Overview

The technical solution is based on a central server that hosts all services for multiple airports. Each airport will have a unique instance of each airport specific service so that each airport is handled individually. The airports share a single connection to AFTN though.

The server works as a web server and the HMI is extended to each airport through HTTPS connections. On the airport side the HMI runs on a web browser.

The rationale behind this solution is to let a large number of regional airports support this capability for a low initial cost and low maintenance cost. It also makes it easier for prototype deployment and maintenance. For airports equipped with an EFPS, there is no need for this web based solution.

ADDEP is divided into a client, which runs on the web-browser, and a server, that runs on the web-server. There may be many clients connected to the physical server. The interface between client and server is implementation specific.

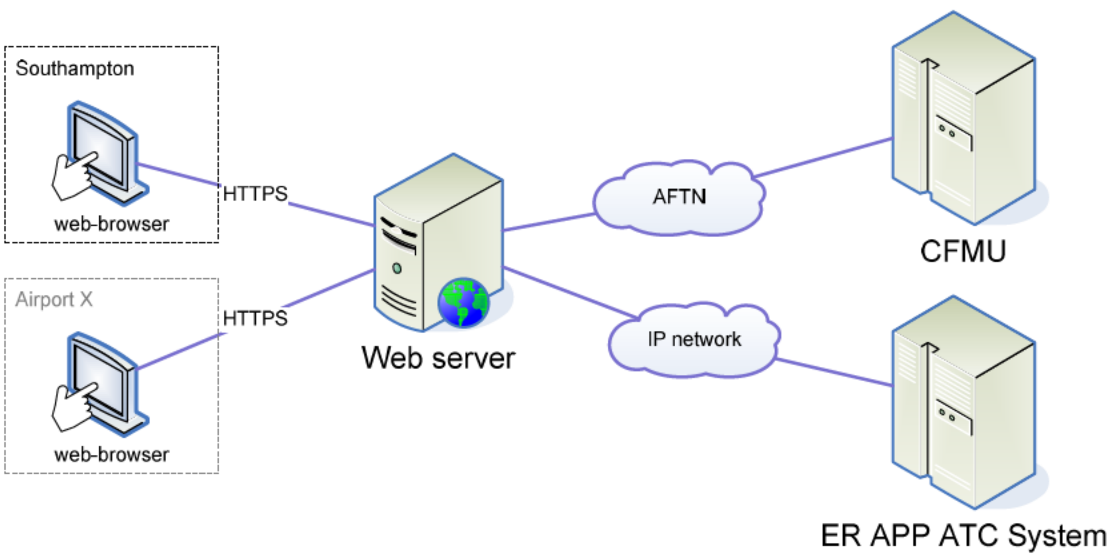

Figure 4-1 Overview of the system implementation of ADDEP

4.4 Detailed description

4.4.1 System architecture

Figure 4-2 describes how ADDEP implements the architecture in chapter 3. ADDEP can be regarded as a partial Aerodrome ATC System in this case. However, for implementation the server and connectivity to CFMU and the local ACC is likely to be hosted within the Capability Configurations of the ACC that parents the Aerodromes.

The system component architecture including internal services and their interconnections are described in the Figure 4-2 block diagram.

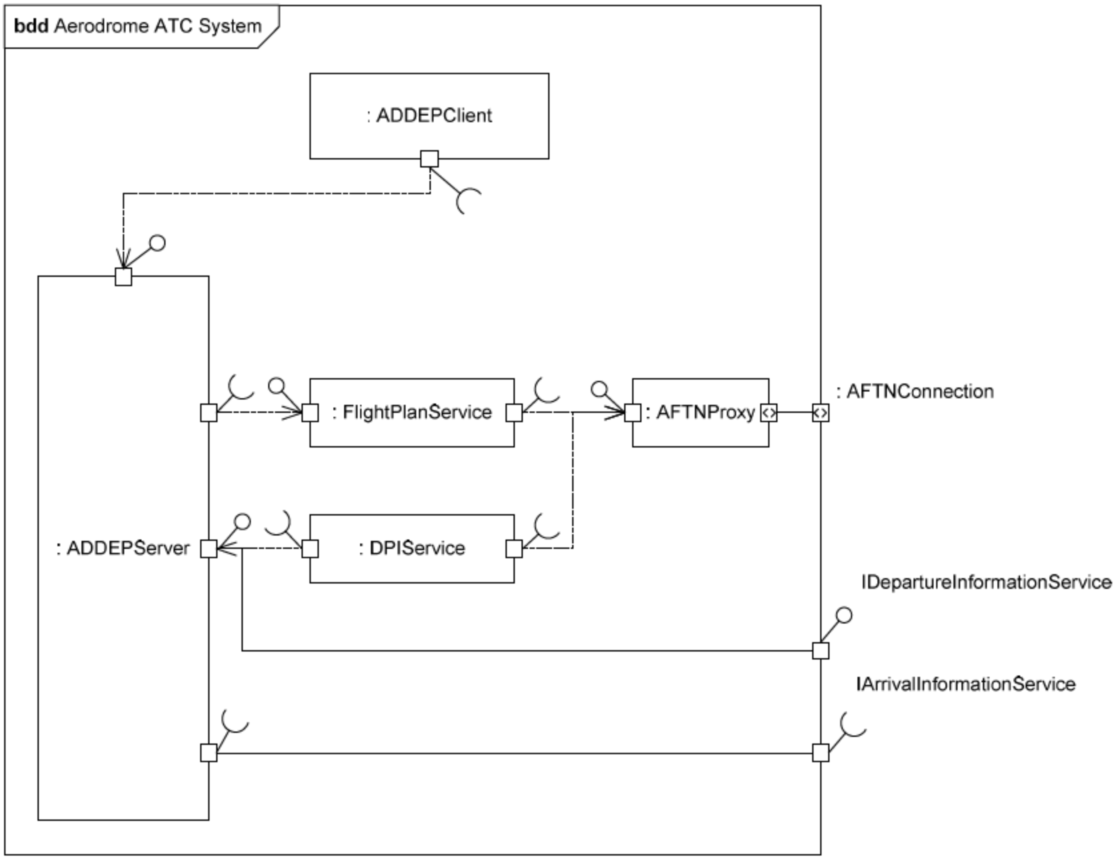

Figure 4-2 The system component architecture around ADDEP

Element	Description
ADDEPClient	ADDEPClient is the web-based HMI for the Tower Controller.
ADDEPServer	ADDEPServer implements the IDepartureInformationService. ADDEPServer subscribes on flight plans from FlightPlanService and on arrival data from IArrivalInformationService. Internally, ADDEPServer compiles information from different sources into internal flight objects which are sent to ADDEPClient.
DPITransmissionService	DPITransmissionService subscribes on departure data from ADDEPServer and creates DPI messages which the service sends to the CFMU via AFTNProxy. The DPI Transmission Service could be extended in the future to take onboard information from other services in order to improve the quality of the DPI messages.
FlightPlanService	Flight Plan Service provides flight plan data for flights that concerns the airport. The information is needed to be able to display the flights to the Tower Controllers. Flight plan data is also needed to fill in details in the DPI messages. The flight plan data exposed shall include all data from the initial flight plan any subsequent updates on that data, including cancellation status, if any.

Element	Description
AFTNProxy	AFTNProxy is a service used to simplify sharing of a single AFTN connection between the services for the multiple airports hosted on the web server. The details on how AFTNProxy communicates with AFTN are site specific.

Table 4-2 Descriptions of elements in Figure 4-2

4.4.2 User interaction

The user interacts with the tool through the touch screen display pictured in Figure 4-1. During operation, the user (ATC operator) is presented with a set of flight strips representing the current active departures from the airport. The flight strips are organized into different sections depending on which state the flight is in. The states include On-Stand, Pushback and Take-Off. The tool's HMI is specially designed for touch screen operation, allowing enough space to operate the buttons displayed on the screen.

The tool's HMI allows the operator to manage the state transition for the flights. When the flight is given the push back clearance, the user presses the Push-back button (on the strip) which moves the strip into the Push-back section. A similar action is used to move the strip from Push-back to Take-Off. The reverse operations are possible by pressing the Cancel button which moves the flight to the previous state. The transition actions cause the system to send DPI message to the CFMU.

The information available on the strip to support the operator includes Call sign, EOBT (from flight plan) and TTOT as predicted by the system.

The flight strips remain in the system until moved into the Take-Off section plus another 10 minutes before the strip is allowed to time-out.

5 Implementation with EFPS and DMAN

5.1 Overview

This chapter describes a variation of the architecture in chapter 3 when implemented with an EFPS and DMAN. The architecture describes the server components/services from the 12.4.1 prototype in a context where a DMAN and EFPS are present.

This setup will be used during validation of the basic DMAN, representing a baseline integration of flow synchronization tools with the Tower System.

The server components from the 12.4.1 prototype has been reused for this configuration and made to serve a more general airport configuration in contrast to chapter 3 which focuses on the regional airports. Whilst the ADDEP HMI is designed as a complement to paper strips, the EFPS is a full replacement.

The requirements corresponding to this configuration are described in Section 3.1.2 (Departures from DMAN equipped airports) of document D12 Thread 3 – Requirement Specification [2].

5.2 System architecture

The system architecture depicted in chapter 4 is re-used, sharing the same server components.

In comparison to chapter 4, the following differences should be noted:

- The ADDEP HMI is replaced by the EFPS
- The DMAN tool is integrated with the ADDEP server components, reusing the interfaces for arrival and departure information.

The system component architecture including internal services and their interconnections are described in the Figure 5-1 block diagram.

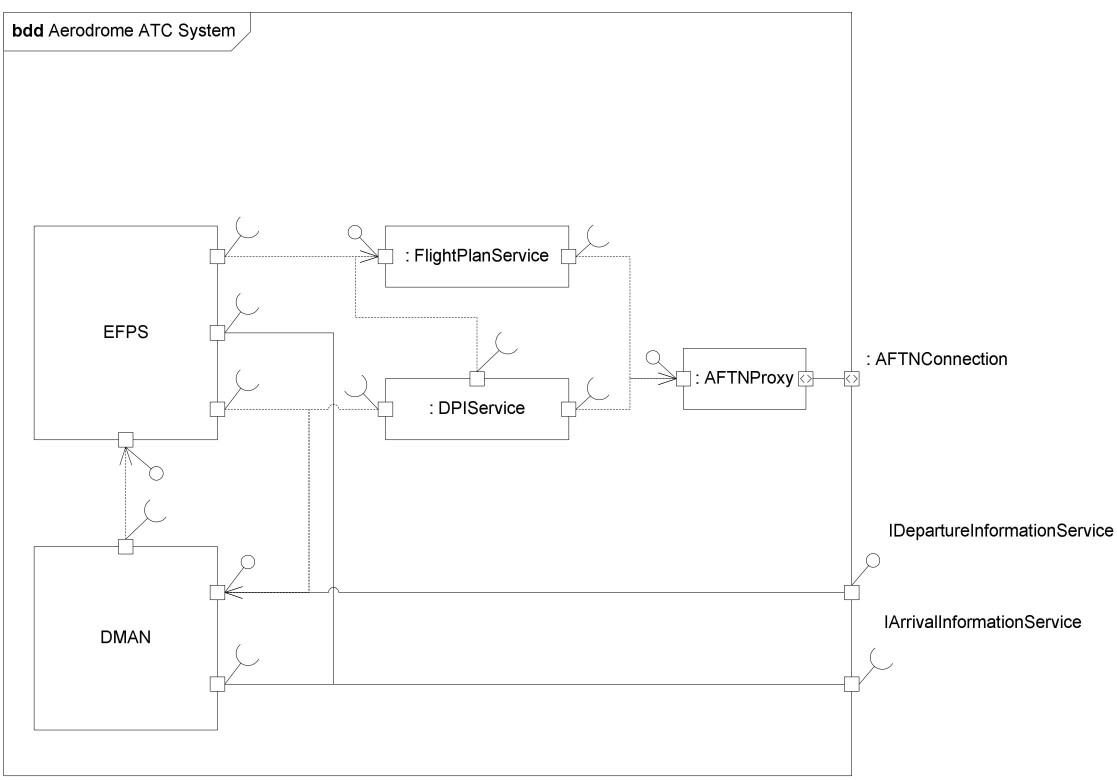

Figure 5-1 The system component architecture around the ADDEP server components together with the EFPS and DMAN tools

Element	Description
EFPS	EFPS subscribes on departure data from IDepartureInformationService and arrival data from IArrivalInformationService to be able to display data to the controllers.
DMAN	DMAN implements the IDepartureInformationService. DMAN subscribes on arrival data from IArrivalInformationService. DMAN needs information from EFPS, but the details about this is not considered to be in scope of this document.
DPITransmissionService	DPITransmissionService subscribes on departure data from ADDEPServer and creates DPI messages which the service sends to the CFMU via AFTNProxy. The DPI Transmission Service could be extended in the future to take onboard information from other services in order to improve the quality of the DPI messages.
FlightPlanService	Flight Plan Service provides flight plan data for flights that concerns the airport. The information is needed to be able to display the flights to the Tower Controllers. Flight plan data is also needed to fill in details in the DPI messages. The flight plan data exposed shall include all data from the initial flight plan any subsequent updates on that data, including cancellation status, if any.
AFTNProxy	The AFTN Proxy is a service used to simplify sharing of a single AFTN connection between the services for the multiple airports hosted on the web server.

Table 5-1 Descriptions of elements in Figure 5-1

5.3 User interaction

In this configuration the interface for user interaction is through the EFPS system which displays flight strips for both arrivals and departures. Even though the description of EFPS is out of scope for this document, some specific aspects related to the integration are described.

The information provided in the flight strip has been extended to support the ATC operator. The information includes a number of airport CDM milestones relating to the aircraft departure status which needs to be presented to the operator. These include EOBT, TOBT, TSAT as well as CTOT, ETOT and TTOT. These milestones have been identified as necessary elements for the operator to follow and manage the departure sequence. The milestones are read-only for the operator except for TOBT which the operator is allowed to change.

6 References

[1] 12.04.01-D03 Thread 1 - Requirement Specification, version 00.01.00

[2] 12.04.01-D12 Thread 3 – Requirement Specification, version 00.00.02

[3] 12 04 01-D10 Concept of Operation, version 00.00.01

[4] 05 06 04-D06-V1 Initial OSED.doc

[5] B.04.03_D03_01 Architecture of the Technical Systems Description Document for Step 1, Edition 00.01.01

[6] DPI Implementation Guide, URB/USD/DPI_Impl_Guide, Edition 1.4, 2010-01-05

-**END OF DOCUMENT -**