NASA/TM–2013-216531

Precision Departure Release Capability (PDRC) Technology Description

Shawn A. Engelland
Ames Research Center, Moffett Field, California

Alan Capps, Kevin Day, and Corissa Robinson
Mosaic ATM, Leesburg, Virginia

Jody R. Null
University of California, Santa Cruz
Ames Research Center, Moffett Field, California

June 2013

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 collections 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 creating custom thesauri, building customized databases, and organizing and publishing research results.

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 via the Internet to help@sti.nasa.gov

- Fax your question to the NASA STI Help Desk at 443-757-5803

- Phone the NASA STI Help Desk at 443-757-5802

- Write to:
  NASA STI Help Desk
  NASA Center for AeroSpace Information
  7115 Standard Drive
  Hanover, MD 21076-1320

NASA/TM–2013-216531

Precision Departure Release Capability (PDRC) Technology Description

Shawn A. Engelland
Ames Research Center, Moffett Field, California

Alan Capps, Kevin Day, and Corissa Robinson
Mosaic ATM, Leesburg, Virginia

Jody R. Null
University of California, Santa Cruz
Ames Research Center, Moffett Field, California

National Aeronautics and
Space Administration

Ames Research Center
Moffet Field, California 94035-1000

June 2013

Available from:

NASA Center for AeroSpace Information
7115 Standard Drive
Hanover, MD 21076-1320

National Technical Information Service
5301 Shawnee Road
Alexandria, VA 22312

This report is also available in electronic form at

http://www.aviationsystemsdivision.arc.nasa.gov/

Table of Contents

List of Figures

List of Tables

This page intentionally left blank.

1 Introduction

This section provides context for the PDRC-IADS research activity, a high-level overview of the PDRC concept and an overview of the material presented in this document.

1.1 Scope

Future air traffic demands are expected to require a greater degree of integration among the automation systems used to manage arrival, departure, and surface traffic. The next generation air transportation system (NextGen) envisions Integrated Arrival/Departure/Surface (IADS) operations as described in the JPDO Integrated Work Plan [38] and in the FAA's NextGen Mid-Term Concept of Operations [35]. Various NextGen concepts [28, 31, 32] describe IADS operations that feature a greater degree of automated coordination as traffic flows from one control domain to the next in the tactical air traffic control environment.

A logical step towards the NextGen vision of fully-integrated arrival/departure/surface operations is to automate tactical scheduling of departure traffic that will join a constrained en route traffic flow as depicted in Figure 1:1, which is based on Figure 1-1 of Reference 28.

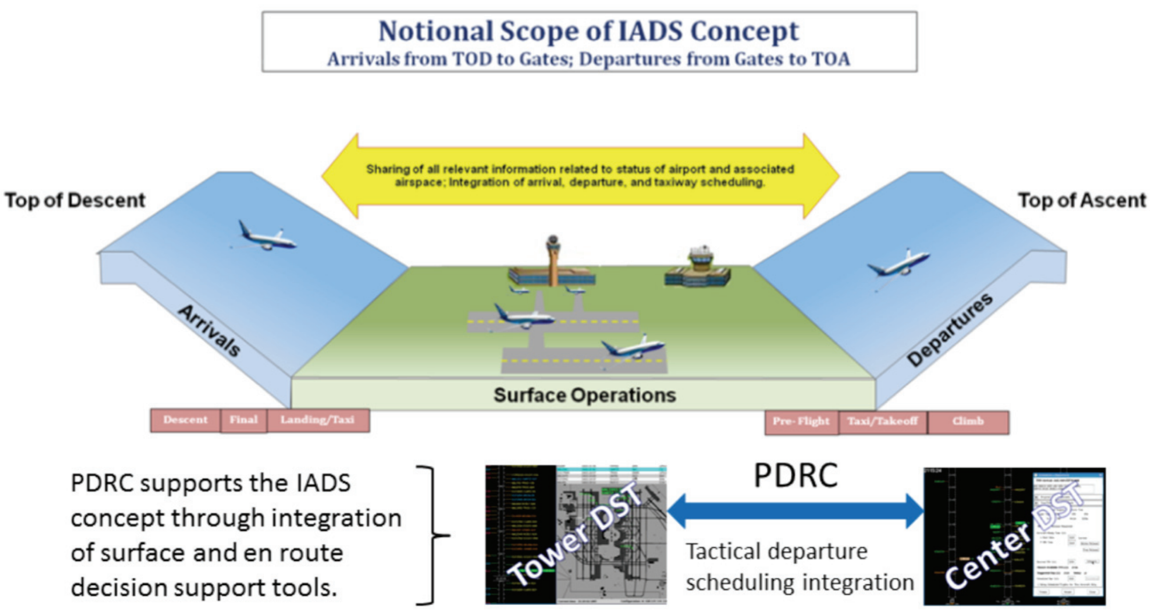

Figure 1:1 – PDRC supports IADS tactical departure scheduling.

A commonly used tactical Traffic Management Initiative (TMI) is the Call For Release (CFR) procedure, which is also known as the Approval Request (APREQ) procedure. CFR procedures vary from facility to facility; however, they generally require the Air Traffic Control Tower (Tower) to request approval from the Air Route Traffic Control Center (Center) prior to releasing departures headed to specified destinations. Earlier research [20, 21] at NASA Ames focused on automating inter-facility coordination during CFR procedures. An FAA-led effort built on this work to develop and evaluate the Departure Flow Management prototype [13].

Presently, en route tactical departure scheduling to meet CFR restrictions is often accomplished with the Traffic Management Advisor (TMA) decision support tool. Tactical departure scheduling with TMA is thoroughly described in Section 3 of the PDRC ConOps [1].

The Precision Departure Release Capability (PDRC) concept combines the automated coordination demonstrated in the previous research [13, 20, 21] with the use of surface trajectory-based takeoff (OFF) time predictions and runway assignments to improve en route tactical departure scheduling during CFR procedures.

PDRC is intended to inform development of mid-term NextGen concepts and technologies, but PDRC evaluations were conducted in the present-day National Airspace System (NAS) environment at a specific location. The scope for this document will be limited to the immediate PDRC-IADS research activity as follows:

System:	PDRC research software system developed for evaluations.
Environment:	North Texas (NTX) Research Station and associated FAA facilities: Fort Worth Center (ZFW), DFW TRACON (D10), and DFW Towers.
Time frame:	2010-2014 for development, evaluation and transition.

This document includes notes and comments on extending the PDRC research results to other NAS environments.

1.2 Identification

The PDRC-IADS research activity is an element of the Systems Analysis Integration and Evaluation (SAIE) Project of NASA's Airspace Systems Program. The Aviation Systems Division at NASA Ames Research Center is conducting the PDRC-IADS research activity based out of NTX. Some of this research activity is being accomplished under a NASA Research Announcement contract (NNA11AC17C), which was awarded on 23 Sep 2011. Mosaic ATM, Inc. is the prime contractor for this work and CSC and Veracity Engineering are subcontractors. This document is a joint effort between NASA and contractor personnel.

NASA and the FAA are coordinating NextGen technology transfer via Research Transition Teams (RTTs). The RTTs have defined Research Transition Products (RTPs), consisting of distinct concept and/or technology elements that can be transferred as a package. PDRC is one of four RTPs currently being coordinated by the IADS RTT. NASA delivered an initial PDRC RTP package to the FAA in July 2012. Formal delivery of the core PDRC RTP package is slated for the summer of 2013. This Technology Description document will be one element of the PDRC RTP package. The FAA has identified the Time Based Flow Management (TBFM) Program and the Terminal Flight Data Manager (TFDM) Program as recipients of the PDRC RTP.

1.3 Concept overview

Figure 1:2 provides a high-level overview of the PDRC operational concept. This figure is applicable to both the outbound and inbound tactical departure scheduling situations described in Section 3 of the ConOps [1]. The right side of the figure depicts departure traffic operating under the CFR procedure where departures must be merged into constrained en route traffic flows. The left side of the figure shows the PDRC decision support tools used for tactical departure scheduling.

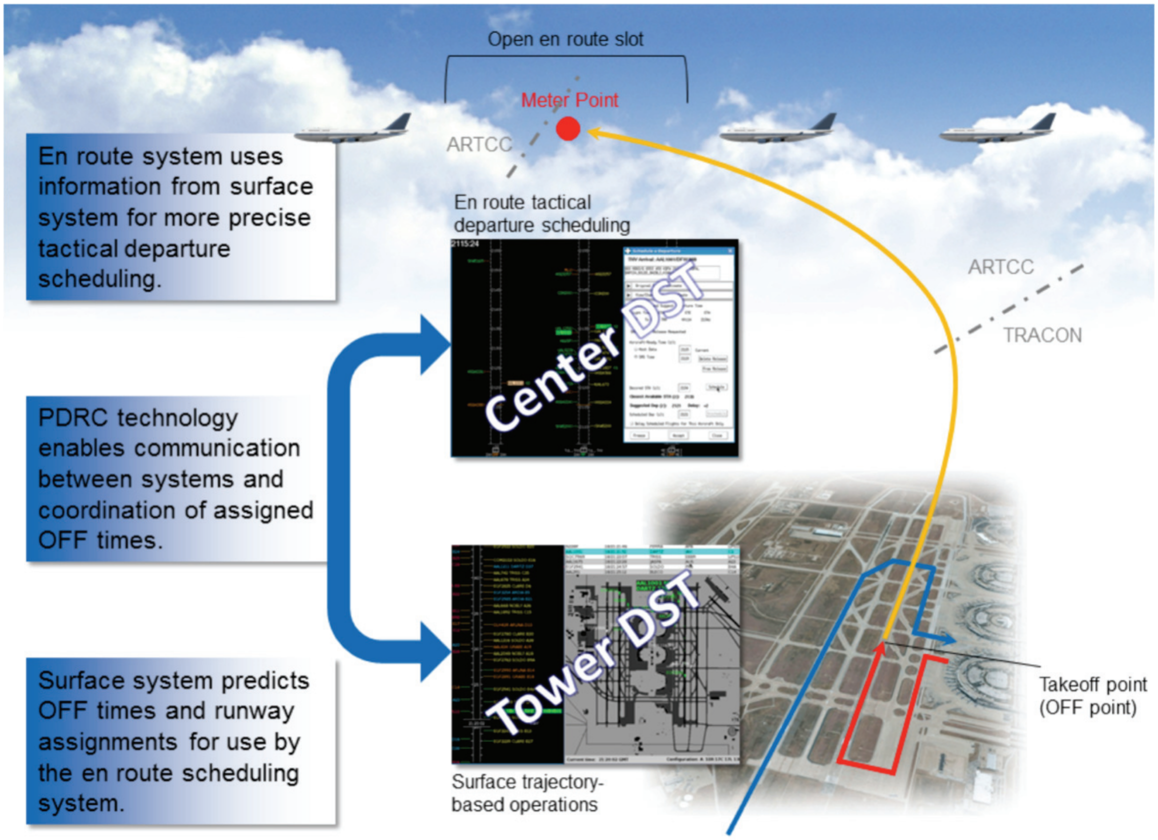

Figure 1:2 – Precision Departure Release Capability (PDRC) system overview.

The upper portion of the figure depicts a traffic stream in the en route domain that is under a CFR constraint. PDRC builds on an existing tactical departure scheduling decision support tool used by the Center to schedule departures into this constrained overhead stream. Ascent modeling in the en route decision support tool enables precise time-based scheduling and de-confliction at the meter point. The modeled ascent trajectory is illustrated by the gold line in Figure 1:2.

The lower portion of Figure 1:2 depicts the Tower environment where a NextGen surface trajectory-based decision support tool is in use. NextGen surface trajectory-based operations are enabled by a surface surveillance system and air carrier data sharing that provides intent and status information (e.g., gate assignments, estimated and actual pushback times). The surface trajectories computed and used by this decision support tool are represented by the blue and red lines in this figure.

PDRC focuses on the automated communication and use of surface trajectory-based OFF time predictions and runway assignments for tactical departure scheduling in CFR situations. In present-day operations, OFF time prediction and communication is manual. Automated PDRC communication is illustrated by the double-headed arrow on the left side of Figure 1:2. The Center decision support tool uses surface trajectory-based OFF time predictions for departure scheduling and coordinates release times with the Tower surface trajectory-based decision support tool. The Tower tool predicts OFF times and runway assignments for use by the Center tool in tactical departure scheduling and coordinates release times with the Center decision tool.

The focal point for PDRC is the OFF event in Figure 1:2 where the red trajectory joins with the gold trajectory on the departure runway. The Tower decision support tool computes surface trajectories to this point to develop OFF time estimates. The Center decision support tool computes ascent trajectories from this point to the merge point in the overhead stream for tactical departure scheduling.

1.4 Document overview

This document describes the technology developed under the PDRC-IADS research activity. Companion documents describe the concept of operations [1] and provide a preview of analytical and field evaluation results [3].

This technology description document incorporates elements of a number of industry standard system and software design documents; however, it does not adhere to a particular document specification. This document seeks to capture all available information regarding technologies (e.g., prototype software systems) developed under the PDRC-IADS research activity and used in the field evaluations at NTX. This information will be organized in a modular manner so that individual sections can be used to develop documents required to support FAA procurement and implementation activities.

The audience for this document includes the following:

- The NASA PDRC-IADS team that will use this document to organize and refine technology descriptions
- The NASA/FAA IADS RTT, which is facilitating the research transition process
- Other NASA NextGen researchers who may use this technology description to coordinate research within NASA and with external research partners
- FAA NextGen implementers who may use this technology description (and other PDRC research products) to inform development of IADS elements of the NextGen enterprise architecture

As implied by the version numbers, this is a living document that will be developed and refined throughout the PDRC-IADS research activity. An updated version will be delivered with each formal RTP delivery. Intermediate versions may also be provided between formal RTP deliveries.

This document is organized as follows:

Section 1 provides context for the PDRC-IADS research activity, a high-level overview of the PDRC concept and an overview of the material presented in this document.

Section 2 summarizes program-level requirements for PDRC-enabled tactical departure scheduling. These requirements are organized by the two domains or environments that PDRC addresses: surface and enroute.

Section 3 provides an overview of the PDRC prototype software system used for concept development and evaluation.

Section 4 provides design details for the surface software component.

Section 5 provides design details for the enroute software component.

Section 6 provides design details for the interface component.

Section 7 describes the development history for the prototype software used in PDRC.

Section 8 describes how to build the prototype software.

Section 9 describes running the prototype software.

Section 10 provides a list of references, applicable documents, and related research for the PDRC-IADS research activity. This list is intended to be common across the PDRC-IADS document family.

The **Glossary** at the end of the document presents a list of acronyms, and their definitions, that are used within the Technology Description document.

2 Program-level requirements

This section summarizes program-level requirements for PDRC-enabled tactical departure scheduling. These requirements are organized by the two domains or environments that PDRC addresses: surface and enroute. Subsequent sections will describe the specific prototype software used for concept development and validation in the PDRC-IADS research activity.

As described in Section 1.3, the PDRC concept involves two decision support tools or automation systems – one for the surface domain and one for the enroute domain. PDRC program level requirements were focused on the interface between the surface and enroute systems. Given that FAA requirements have already been specified for the surface and enroute systems, PDRC program requirements focused on the differences that were required to enable PDRC capability.

2.1 Surface system requirements

This subsection presents PDRC program-level requirements for the surface system. Figure 2:1 illustrates the requirements overview diagram with surface system input requirements shown at the top of the diagram. Surface system interface, process, store, and log requirements are summarized in the middle of the diagram. Surface system output requirements are shown at the bottom of Figure 2:1. The surface system requirement text is in Table 2:1.

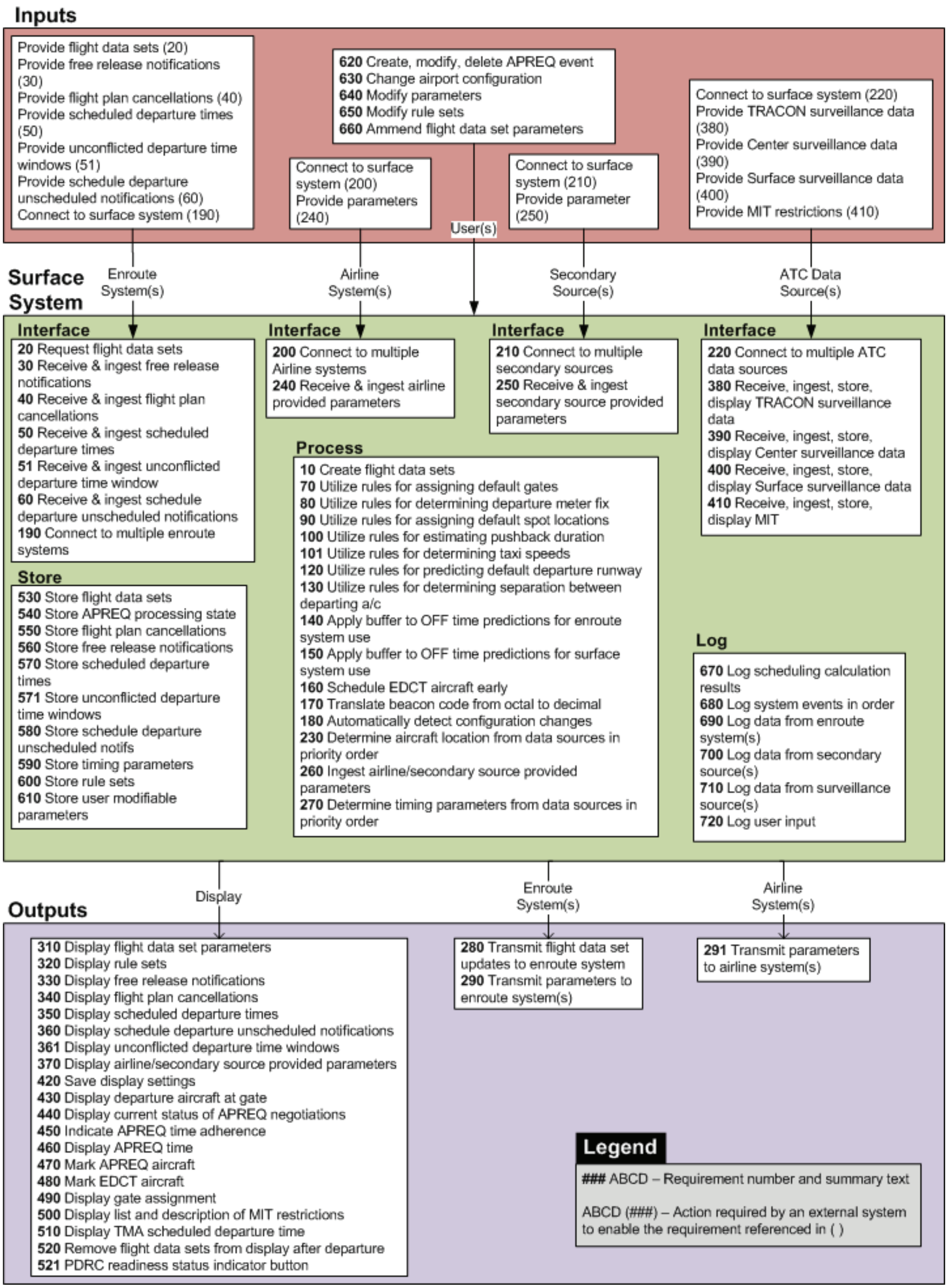

Figure 2:1 – Surface system requirements overview diagram.

Table 2:1 – PDRC program-level requirements for the surface domain.

ID	Requirement Text	Requirement Note
Surface System		
10	The surface system shall create flight data sets for aircraft of interest to the surface system. The flight data set parameters shall include: (a) Aircraft ID (b) TMA Unique ID (c) Departure Airport (d) Destination Airport (e) Beacon Code (f) Runway Name (g) Flight Status (h) Call for Release Restriction Status (i) Undelayed Coordinated Off Time (j) Predicted Coordinated Off Time (k) Actual OFF Time	
20	The surface system shall request flight data set data from the enroute system upon initial connect or upon request.	
30	The surface system shall receive and ingest free release notifications from the enroute system upon user entry of the cancellation information.	Note 1: There are times within an APREQ in which there is very low demand followed by high demand. If the enroute operator sees the likely release time and while evaluating the overhead stream sees this low demand, they can mark the aircraft as a free release. Note 2: Free release notifications are manually entered in the enroute system and are event driven (i.e., they do not happen on a periodic basis).
40	The surface system shall receive and ingest flight plan cancellations from the enroute system upon user entry of the cancellation information.	Note: Flight plan cancellations are manually updated in the enroute system and are event driven (i.e., they do not happen on a periodic basis).

ID	Requirement Text	Requirement Note
50	The surface system shall receive and ingest scheduled departure times from the enroute system.	The scheduled departure times are sent upon user entry in the enroute system. The scheduled departure time received from the enroute system takes into account the ready time information provided by the surface system plus accounts for overhead stream delay. Note: Scheduled departure times are manually updated in the enroute system and are event drive (i.e., they do not happen on a periodic basis).
51	The surface system shall receive and ingest unconflicted departure time windows from the enroute system.	
60	The surface system shall receive and ingest departure unschedule notifications from the enroute system.	The unschedule notifications are sent upon user entry in the enroute system. In this case, the flight has not been cancelled, but the TMA/EDC user needs to unschedule the agreed upon time. This could happen if something changes in the NAS that prevents the aircraft from making the agreed upon time, or it could be done as a way to start the time negotiation process over. Note: Scheduled departure unschedule notifications are manually updated in the enroute system and are event driven (i.e., they do not happen on a periodic basis).
70	The surface system shall utilize an adaptable set of rules for assigning default gates to aircraft.	
80	The surface system shall utilize an adaptable set of rules for determining the default departure fix.	
90	The surface system shall utilize an adaptable set of rules for assigning default spot locations.	
100	The surface system shall utilize an adaptable set of rules for estimating the transit time from gate pushback to Spot.	

ID	Requirement Text	Requirement Note
101	The surface system shall utilize an adaptable set of rules for determining taxi speeds for aircraft.	
120	The surface system shall utilize an adaptable set of rules for predicting the default departure runway.	
130	The surface system shall utilize an adaptable set of rules for determining separation between departing aircraft.	
140	The surface system shall apply an adaptable buffer to the OFF time predictions. The OFF time prediction plus the buffer will be used as the initial OFF time prediction by the enroute system.	
150	The surface system shall apply an adaptable buffer to the OFF time predictions.	The OFF time prediction plus the buffer will be used as the APREQ time in the surface system.
160	The surface system shall utilize the earliest time available within the constraints as the predicted time for EDCT aircraft.	The value passed to the enroute system should be the earliest available time within the EDCT window that can be met while including other surface constraints.
170	The surface system shall receive and reconcile beacon codes in formats used by different systems.	The beacon code from TMA is in octal and this must match the beacon code received from ASDE-X, which is in the decimal format.
180	The surface system shall automatically detect airport configuration changes.	
190	The surface system shall accept connections from multiple enroute systems.	
200	The surface system shall accept connections from multiple airline systems.	
210	The surface system shall accept connections from multiple secondary sources of airline data.	FlightStats is an example of a secondary source of airline data.
220	The surface system shall accept connections from multiple ATC data sources.	Center Surveillance, TRACON Surveillance, and Ground Surveillance are examples of ATC data sources.
230	If multiple sources of aircraft location are available, the surface system shall ingest, store, and display the aircraft location data in the following prioritized order: (a) Surface Surveillance (b) TRACON Surveillance (c) Center Surveillance	Only one location value for each aircraft should exist in the system at any given time. This location data is updated each time new data is received from the sources listed.

ID	Requirement Text	Requirement Note
240	The surface system shall receive the following parameters from airline systems. Timing related parameters, a, b, and d, are requested in a GMT format at the highest level of precision available from each individual airline's system. (a) Actual OUT time (b) Estimated Pushback time (c) Gate Assignment (d) Scheduled Pushback time	This requirement is not intended to levy any data precision requirements against the airlines systems, but rather accepts whatever precision their systems currently have. The airline systems are considered to be the primary source for obtaining the timing parameter values.
250	The surface system shall receive the following parameters from a secondary source for airline data. Timing related parameters, a, b, and d, are requested in a GMT format at the highest level of precision available from each source's system. (a) Actual OUT time (b) Estimated Pushback time (c) Gate Assignment (d) Scheduled Pushback time	FlightStats is an example of a secondary source of data for airline related information. This requirement is not intended to levy any data precision requirements against the secondary source systems, but rather accepts whatever precision their systems currently have.
260	The surface system shall ingest the following parameters upon receipt: (a) Actual OUT time (b) Estimated Pushback time (c) Gate Assignment (d) Scheduled Pushback time	Receipt of each parameter is event driven and the update frequency is expected to vary based on airlines, traffic levels, time of day, and weather conditions.
270	If multiple sources for the list of parameters are available, the surface system shall ingest, store, and display the parameters in the following prioritized order: (a) Manual entry by an ATCT User (b) Primary source (directly from airlines) (c) Secondary source (d) Surface system assigned default	Only one value for each of the timing parameters should exist in the system at any given time. If a parameter has input from all 4 sources listed in the requirement, then only the manual entry from ATCT will persist. If a parameter has input from only 1 source, then it is that source that will provide the data that is persisted in the system.
Output to Other Systems		
280	The surface system shall transmit periodic updates of flight data sets to the enroute system(s).	The prototype system sent updates every 10 seconds.

ID	Requirement Text	Requirement Note
290	The surface system shall transmit the following parameters to the enroute system(s): (a) Actual Wheels OFF time (b) Aircraft State (c) Predicted Wheels OFF time (d) Delete Notification (e) Departure Runway (f) Earliest Wheels OFF time	Predicted Wheels OFF Time is the best prediction available to the system including any delay expected by other surface traffic. Earliest Wheels OFF Time is the undelayed time that assumes no other surface traffic exists.
291	The surface system shall transmit the following parameters to the airline system(s) at a minimum: (a) APREQ Notice (b) Spot Time (c) Estimated OFF time (d) Estimated Meter Point Crossing Time (e) Predicted Departure Runway	Note that the estimated meter point crossing time comes from the enroute system but is being relayed by the surface system.
User Interface		
310	The surface system shall display the following parameters on the surface TMC display: (a) Departure Fix (b) Predicted OFF Time (c) Gate Assignment (d) Predicted Pushback Time (e) Assigned Runway (f) Predicted Spot (g) Electronic Negotiation of Tactical scheduling Readiness status	
320	The surface system shall display the adaptable sets of rules for the following: (a) Assigning default gates to departing aircraft (b) Determine the default departure fix (c) Assigning default spot locations for departing aircraft (d) Estimating the pushback duration for departing aircraft (e) Estimating ramp taxi speed for departing aircraft (f) Predicting the departure runway for departing aircraft (g) Determining separation between departing aircraft	
330	The surface system shall display free release notifications from the enroute system on the surface TMC display.	

ID	Requirement Text	Requirement Note
340	The surface system shall display flight plan cancellations from the enroute system on the surface TMC display.	
350	The surface system shall display scheduled departure times from the enroute system on the surface TMC display.	
351	The surface system shall display unconflicted departure time windows from the enroute system on the surface TMC display.	
360	The surface system shall display scheduled departure unschedule notifications from the enroute system on the surface TMC display.	See requirement # 60 for more details on “unschedule” functions.
370	The surface system shall display the following parameters received from an airline system and/or a secondary source on the surface TMC display. (a) Actual OUT time (b) Estimated OUT time (c) Gate Assignment (d) Scheduled OUT time	
380	The surface system shall receive, ingest, store, and display incoming TRACON surveillance data.	This data is used to determine an aircraft's location.
390	The surface system shall receive, ingest, store, and display incoming Center surveillance data.	This data is used to determine an aircraft's location.
400	The surface system shall receive, ingest, store, and display incoming surface surveillance data from ASDE-X and ADS-B.	This data is used to determine an aircraft's location.
410	The surface system shall receive, ingest, store, and display Miles in Trail values affecting departure flow at an airport of interest.	
420	The surface system shall allow user specified display setting to be saved and recalled.	
430	The surface system shall display departure aircraft while at the gate on the surface TMC display.	

ID	Requirement Text	Requirement Note
440	The surface system shall display the current status of the APREQ negotiation with the enroute system on the surface TMC display.	For instance, when the flight is "requested", the words are displayed on the same line with the aircraft to give the TMC an indication that the request has already come in. The status is also displayed in the surface flights table.
450	The surface system shall indicate on the surface TMC display if there is a difference between an aircraft's predicted time and APREQ time. The threshold of the time difference shall be configurable.	
460	The surface system shall display the APREQ time on the surface TMC display.	The desired precision of the APREQ time is seconds.
470	The surface system shall uniquely mark APREQ aircraft on the timeline with an adaptable color.	
480	The surface system shall uniquely mark EDCT aircraft on the timeline with an adaptable color.	
490	The surface system shall display the gate assignment for each aircraft on the surface TMC display.	
500	The surface system shall display a list and description of all Miles in Trail constraints that are currently active on the surface TMC display.	
510	The surface system shall display the TMA scheduled departure time on the surface TMC display.	The desired precision of the TMA scheduled departure time is seconds.
520	The surface system shall remove flight data sets from the displays after they have departed.	
521	The surface system shall display a status indicator allowing the user to specify the readiness status for electronic negotiation of tactical scheduling.	Electronic negotiation of tactical scheduling status indicates that a user is capable and willing to use electronic negotiation for tactical scheduling operations rather than phone calls. This status indicator also logs duration of electronic negotiation during operations.
Data Storage		
530	The surface system shall store flight data sets for planned and active flights within a given airspace.	

ID	Requirement Text	Requirement Note
540	The surface system shall store the state of APREQ processing.	The current states are stored on a flight by flight basis, and include: (a) the flight is/isn't affected by an APREQ (b) it is affected and a release time has been requested (c) scheduled time received (d) accepted (e) rejected (f) canceled
550	The surface system shall store flight plan cancellations from the enroute system.	
560	The surface system shall store free release notifications from the enroute system.	
570	The surface system shall store scheduled departure times from the enroute system.	
571	The surface system shall store unconflicted departure time windows from the enroute system.	
580	The surface system shall store scheduled departure unschedule notifications from the enroute system.	See requirement # 60 for more details on "unschedule" functions.
590	The surface system shall store the following airline and/or secondary source provided parameters in a GMT format: (a) Actual OUT time (b) Estimated OUT time (c) Gate Assignment (d) Scheduled OUT time	
600	The surface system shall store adaptable sets of rules for the following: (a) Assigning default gates to departing aircraft (b) Determine the default departure fix (c) Assigning default spot locations for departing aircraft (d) Estimating the pushback duration for departing aircraft (e) Estimating ramp taxi speed for departing aircraft (f) Predicting the departure runway for departing aircraft	

ID	Requirement Text	Requirement Note
	(g) Determining separation between departing aircraft	
610	The surface system shall store the following user modifiable parameters: (a) Departure Fix (b) Predicted OFF Time (c) Gate Assignment (d) Predicted Pushback Time (e) Assigned Runway (f) Predicted Spot	
User Actions		
620	The surface system shall allow a user to create, modify, and delete an APREQ event. An APREQ event includes the following information at a minimum: (a) Start Time (b) Stop Time (c) Engine Type (Jet, Turboprop, and Prop) (d) Destination Airport (if applicable) (e) Jet Airway (if applicable)	
630	The surface system shall allow the user to manually change the airport configuration.	
640	The surface system shall allow the following parameters to be modifiable by a system user: (a) Departure Fix (b) Predicted OFF Time (c) Gate Assignment (d) Predicted Pushback Time (e) Assigned Runway (f) Predicted Spot	
650	The surface system shall allow users to modify the adaptable rule sets.	
660	The surface system shall amend flight data set parameters upon receipt of user input or upon detection of system generated amendments.	
Logging		

ID	Requirement Text	Requirement Note
670	The surface system shall log results of all calculations made by the system that affect the scheduling of a flight.	
680	The surface system shall log all system events in chronological order.	
690	The surface system shall log the following from the enroute system(s): (a) Raw data (b) Time stamp (c) Source	
700	The surface system shall log the following from the secondary system(s): (a) Raw data (b) Time stamp (c) Source	
710	The surface system shall log the following from the surveillance system(s): (a) Raw data (b) Time stamp (c) Source	
720	The surface system shall log all user inputs.	

2.2 Enroute system requirements

This subsection presents PDRC program-level requirements for the enroute system. Figure 2:2 illustrates the enroute requirements overview diagram with enroute system input requirements shown at the top of the diagram. Enroute system interface, process, store, and log requirements are summarized in the middle of the diagram. Enroute system output requirements are shown at the bottom of Figure 2:2. The enroute requirement text is in Table 2:2.

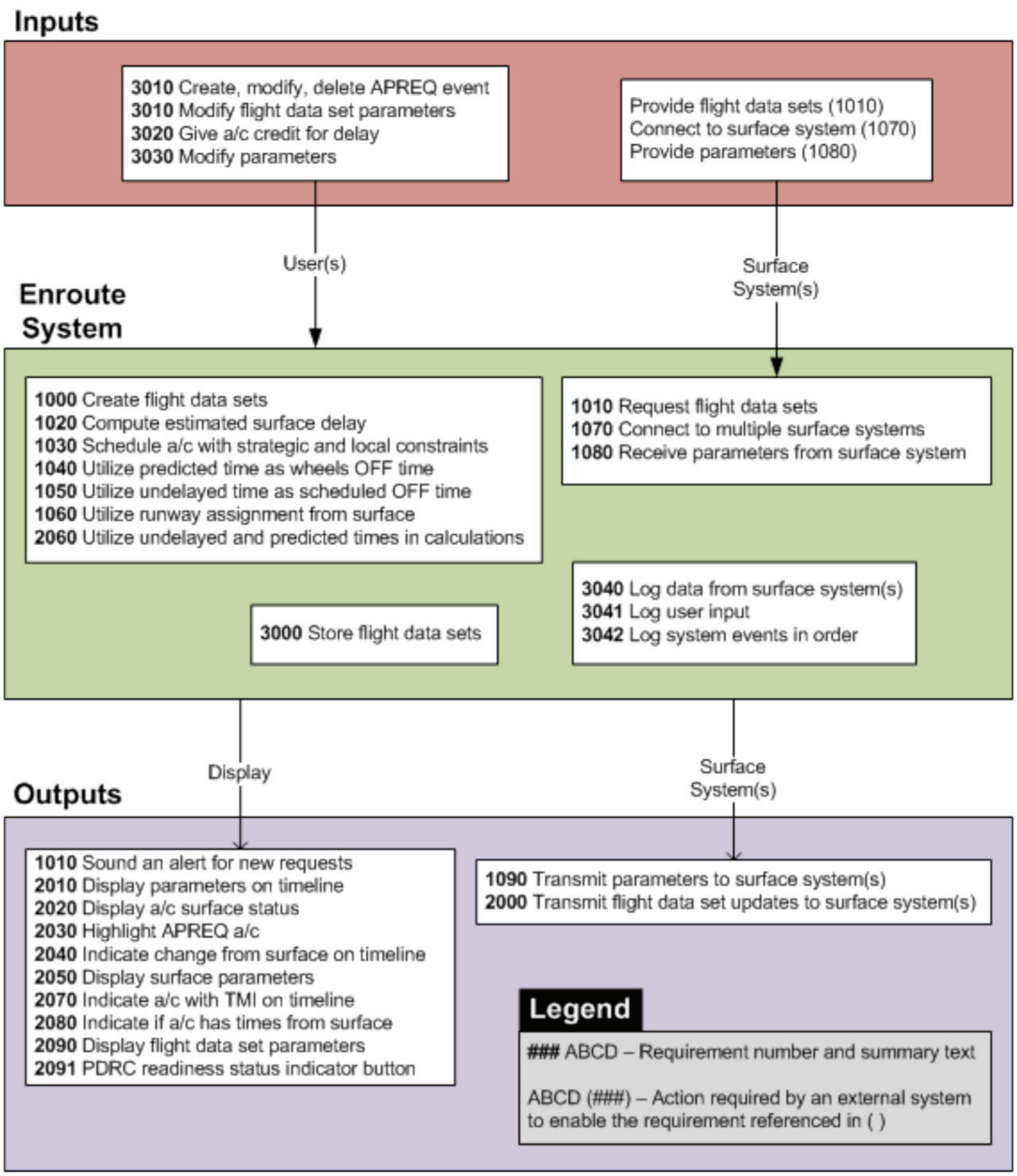

Figure 2:2 – Enroute system requirements overview diagram.

Table 2:2 – PDRC program-level requirements for the enroute system.

ID	Requirement Text	Requirement Note
Enroute System		
1000	The enroute system shall create flight data sets for aircraft of interest to the enroute system. The flight data set parameters shall include: (a) Aircraft ID (b) TMA Unique ID (c) Departure Airport (d) Destination Airport (e) Beacon Code (f) Runway Name (g) Flight Status (h) Call for Release Restriction Status (i) Undelayed Coordinated Off Time (j) Predicted Coordinated Off Time (k) Actual OFF Time	
1001	The enroute system shall request flight data set data from the surface system upon initial connect or upon request.	
1020	The enroute system shall compute the estimated surface delay.	
1030	The enroute system shall take into account strategic and local traffic management constraints when scheduling an aircraft.	
1040	The enroute system shall utilize the predicted time from the surface system as the expected wheels OFF time.	
1050	The enroute system shall utilize the undelayed time from the surface system as the earliest wheels OFF time.	
1060	The enroute system shall utilize the runway assignment assigned by the surface system.	
1070	The enroute system shall accept connections to multiple surface systems.	
1080	The enroute system shall receive the following parameters from the surface system: (a) Runway Assignment (b) Actual OFF Time	

ID	Requirement Text	Requirement Note
2060	The enroute system shall utilize the undelayed and predicted times from the surface system for delay calculation.	
Output to Surface System		
1090	The enroute system shall transmit the following to the surface system: (a) Flight Unscheduled Notification (b) Coordinated Scheduled OFF times (c) Estimated Time of Arrival at a Meter Point (d) Scheduled Time of Arrival at a Meter Point (e) Miles in Trail Separation Requirements (f) APREQ Event Data (g) Unconflicted Departure Time Windows	The unconflicted departure time window depicts the range of time ahead and behind a Coordinated Scheduled OFF time that would also be sufficient for the aircraft to be departed OFF.
2000	The enroute system shall transmit flight data sets to the surface system for all flights that it receives from its surveillance sources.	
User Interface		
1010	The enroute system shall have the ability to sound an audible alert when a new request for release time has arrived from a surface system.	
2010	The enroute system shall display the following parameters: (a) Estimated Surface Delay (b) Coordinated Scheduled Time of Departure (c) Actual OFF Times (d) Electronic Negotiation of Tactical scheduling Readiness status	The estimated surface delay is the predicted time from surface minus the undelayed.
2020	The enroute system shall provide a display panel that contains relevant information and status on all surface departure aircraft.	
2030	The enroute system shall provide aircraft specific highlighting on a display that maps to flights APREQ processing state.	
2040	The enroute system shall display an indication on the primary timeline when a change has occurred to surface scheduling.	
2050	The enroute system shall display relevant surface information on the primary scheduling panel. This includes but is not limited to: (a) Predicted Wheels OFF Time (b) Departure Runway Assignment (c) Surface Status	

ID	Requirement Text	Requirement Note
2070	The enroute system shall use an adaptable symbol to indicate when an aircraft is subject to both a strategic traffic management initiative and has times from the surface system.	
2080	The enroute system shall indicate when an aircraft has times from the surface system through the use of an adaptable symbol on the timeline display.	
2090	The enroute system shall display flight data sets on displays.	
2091	The enroute system shall display a status indicator allowing the user to specify the readiness status for electronic negotiation of tactical scheduling.	Electronic negotiation of tactical scheduling status indicates that a user is capable and willing to use electronic negotiation for tactical scheduling operations rather than phone calls. This status indicator also logs duration of electronic negotiation during operations.
Data Storage		
3000	The enroute system shall store flight data sets for planned and active flights within a given airspace.	
User Actions		
3010	The enroute system shall amend flight data set parameters upon receipt of user input.	
3020	The enroute system shall allow the user the option to give credit for delay taken on the ground when scheduling aircraft in the overhead stream.	
3030	The enroute system shall allow a user to update the following parameters: (a) Scheduled Time of Departure (b) Flight Status (i.e., Cancelled, Unscheduled) (c) Runway Assignment (d) OFF Time Estimation	The TMA/EDC system allows this today on the timeline, however, not with the surface data. The basic philosophy is that all surface provided values are advisory in nature and easily overridden by the user when required.
Logging		
3040	The enroute system shall log the raw incoming messages from the surface system.	
3041	The enroute system shall log all user inputs.	
3042	The enroute system shall log all system events in chronological order.	

3 PDRC prototype software overview

The PDRC research activity strives to be implementation neutral, meaning that the research results will be generally applicable, regardless of which specific surface or en-route tools are used. Generally applicable research results will include the overall operational concept, OFF time prediction accuracy requirements, and information exchange requirements. The prototype software used for development and evaluation may not reflect eventual implementation systems.

Figure 3:1 provides a high-level overview of the prototype software being used for PDRC concept development and evaluation. The upper portion of this figure shows the Center tactical departure scheduling decision support tool. The PDRC prototype utilizes rTMA for this component. The lower portion of this figure shows the Tower decision support tool. The PDRC prototype uses SDSS for this component. Detailed descriptions of rTMA and SDSS are provided in later sections.

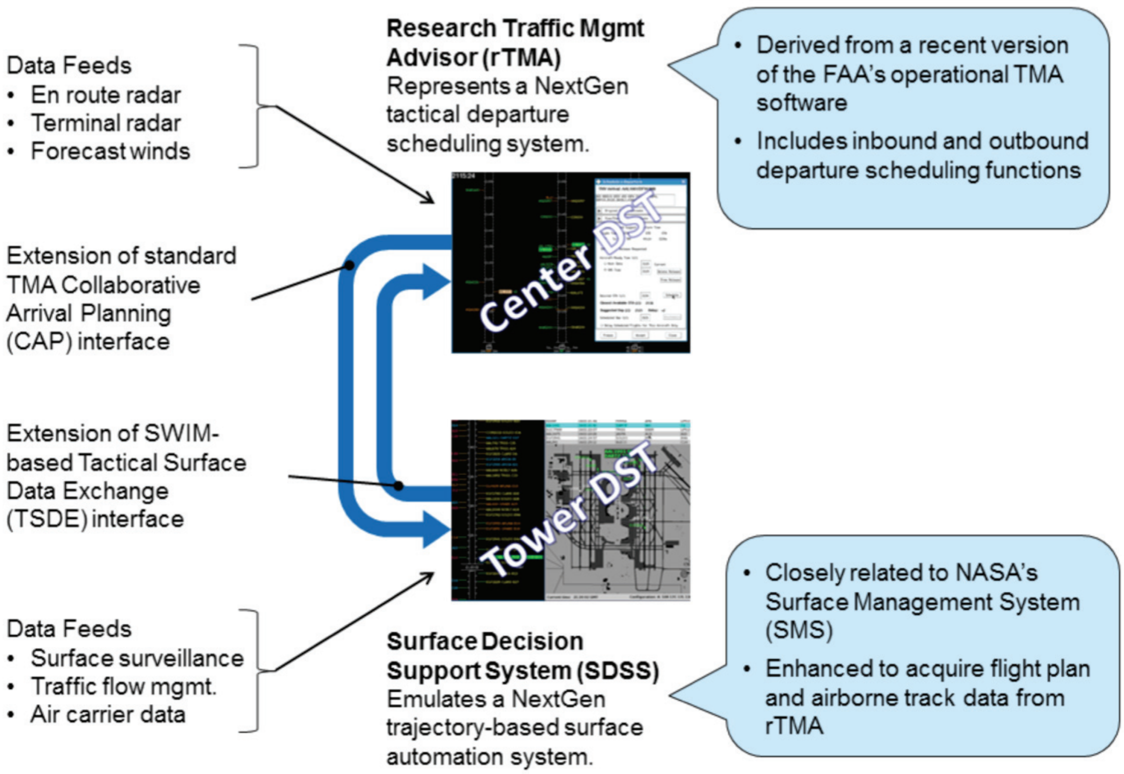

Figure 3:1 – PDRC prototype software overview.

3.1 Research TMA (rTMA) overview

In 1996, NASA successfully demonstrated a prototype TMA system at Fort Worth Center (ZFW) and DFW TRACON (D10) [26]. The NASA prototype TMA was part of the Center/TRACON Automation System (CTAS) family of decision support tools. The prototype TMA was primarily focused on arrival metering; however, it did include an internal departure scheduling capability to support departures from within the Center that were destined to the TMA-metered airport. The FAA's subsequent TMA development effort improved internal departure scheduling. Adjacent center metering was introduced in 2003 when ZFW controllers began metering Houston-bound traffic using times computed by the Houston Center (ZHU) TMA system. TMA adjacent center metering capabilities enable "internal" departure scheduling beyond the arrival airport's home ARTCC. In 2006 the FAA added the En Route Departure Capability (EDC) to TMA, which built upon NASA's Multi-Center TMA research [14], and further enhanced TMA's tactical departure scheduling capabilities. EDC is commonly used to apply miles-in-trail restrictions and to regulate departures into constrained airspace. As of January 2011 the FAA had deployed TMA to 80 operational facilities: 20 ARTCCs, 31 TRACONS, 28 ATCTs and the ATCSCC.

NASA continues to use the CTAS software baseline (including prototype TMA) for NextGen concept and technology development. The PDRC research activity's focus on operational tactical departure scheduling dictated use of a research system that was nearly identical to operational TMA. Research TMA (rTMA) was derived from a recent release of operational TMA software. rTMA was modified to build and execute in NASA's research environment (i.e., Linux operating system on Intel processors), and configured to run without the complex monitor and control system that supports the FAA's operational TMA installation.

PDRC provided the original motivation for creating rTMA; however, this research tool promptly found other applications. NASA and the FAA are both using rTMA to support various R&D efforts, and the agencies are actively collaborating to further develop rTMA capabilities. Figure 3:2 illustrates the relationship between rTMA and the FAA's operational TMA.

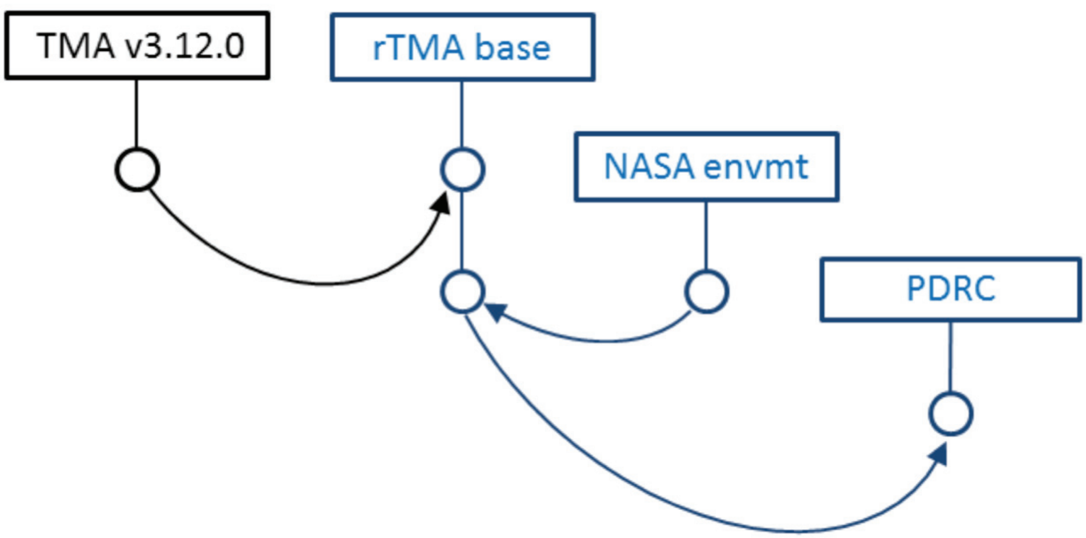

Figure 3:2 – rTMA relationship to operational TMA versions.

The black elements of Figure 3:2 represent the FAA's operational TMA software. The blue elements in the figure represent NASA rTMA development activity. As shown in the figure, current versions of rTMA are derived from FAA TMA v3.12.0. The rtma_base branch is intended to remain as close as possible to the parent FAA TMA version. A very limited number of changes are introduced to rtma_base to accommodate NASA's software development environment. After these environment-focused changes have been made, various research activities perform development on branches off of rtma_base. The PDRC-IADS research activity works in the iads_integ branch where the current rTMA software is identified as PDRC_3.0.x. Section 7 provides more details on PDRC software development history.

The rTMA development strategy maximizes commonality with the FAA operational TMA baseline to facilitate technology transfer. rTMA includes a new Surface Data Interface (SDIF) module that enables communication with SDSS via the Tactical Surface Data Exchange (TSDE) interface. rTMA also includes scheduling algorithms (i.e., Dynamic Planner and Meter Point Dynamic Planner) and user interface modifications to enable use of SDSS trajectory-based OFF time predictions. rTMA retains TMA's two trajectory-based tactical departure scheduling functions: TMA EDC for the outbound situation, and TMA arrival metering "internal" departure scheduling for the inbound situation.

The upper left portion of Figure 3:1 depicts the data feeds required by rTMA, which are the same as for operational TMA systems. Flight plans and tracks are provided by surveillance data feeds from Center (home and all adjacent facilities) Host or ERAM computers. Surveillance data from the ARTS or STARS computers at all involved TRACON facilities is also required. Finally, the system uses Rapid Refresh (RR) forecast winds aloft information provided by NOAA.

3.2 Surface management system overview

In 2003, NASA successfully demonstrated the prototype Surface Management System (SMS) decision support tool at Memphis [7]. NASA transferred the SMS technology to the FAA, which developed it into the Surface Decision Support System (SDSS) research platform. The FAA's Advanced Technology Development & Prototyping Group has used SDSS at various FAA test beds to support Surface Trajectory-Based Operations (STBO) research activities [8].

SMS continues to be NASA's primary platform for NextGen surface automation research. SMS supports various NASA research activities [17], in addition to the PDRC research activity described in this paper. These NASA research activities contribute to a common SMS software baseline. Active software development collaboration between NASA and the FAA STBO project has resulted in a high degree of commonality between SMS and SDSS – to the point where the SMS and SDSS names are often used interchangeably. The PDRC-IADS research activity has adopted the SDSS terminology, and it will be used wherever practical in PDRC documentation.

SDSS is a full-featured decision support tool designed to help controllers, traffic managers, and air carriers manage the movements of aircraft on the surface of busy airports. SDSS includes traffic management functions for use in Towers, TRACONs, and Centers. PDRC utilizes SDSS traffic management functions associated with the CFR procedure as well as the underlying surface trajectory computations that support all SDSS functions. SDSS produces trajectory-based OFF time predictions by combining air carrier intent information with surface surveillance data and departure queue projections.

The lower left portion of Figure 3:1 depicts the data feeds required by SDSS. At NTX, the surface surveillance data comes from the DFW Airport Surface Detection Equipment, Model X (ASDE-X) Data Distribution system. Air carrier data (i.e., gate assignments, equipment assignments, and arrival and departure time estimates) were obtained from a combination of a direct interface to a major air carrier as well as a secondary interface to a commercial service. rTMA is the source for flight plans and airborne surveillance data, as described below.

Figure 3:3 presents the recent history for SDSS software development. The July 2011 PDRC shadow evaluation [4] used the 8.4.4nX series of SDSS releases, which are just beyond the top of

Figure 3:3. The summer 2012 Block 1 PDRC operational evaluation used the SDSS version 9.1.2.n1, which is circled in red near the "PDRC-IADS" label near the center of the diagram. The winter 2012/2013 Block 2 evaluation used the 9.2.1 version of SDSS and is circled in red the lower half of Figure 3:3.

Figure 3:3 also illustrates the collaborative nature of SDSS software development. One can see that versions of SDSS used for NASA's SARDA work (see lower right portion of the diagram) have benefitted from earlier PDRC developments. Likewise, more recent versions of SDSS have benefitted from numerous developments from other research activities.

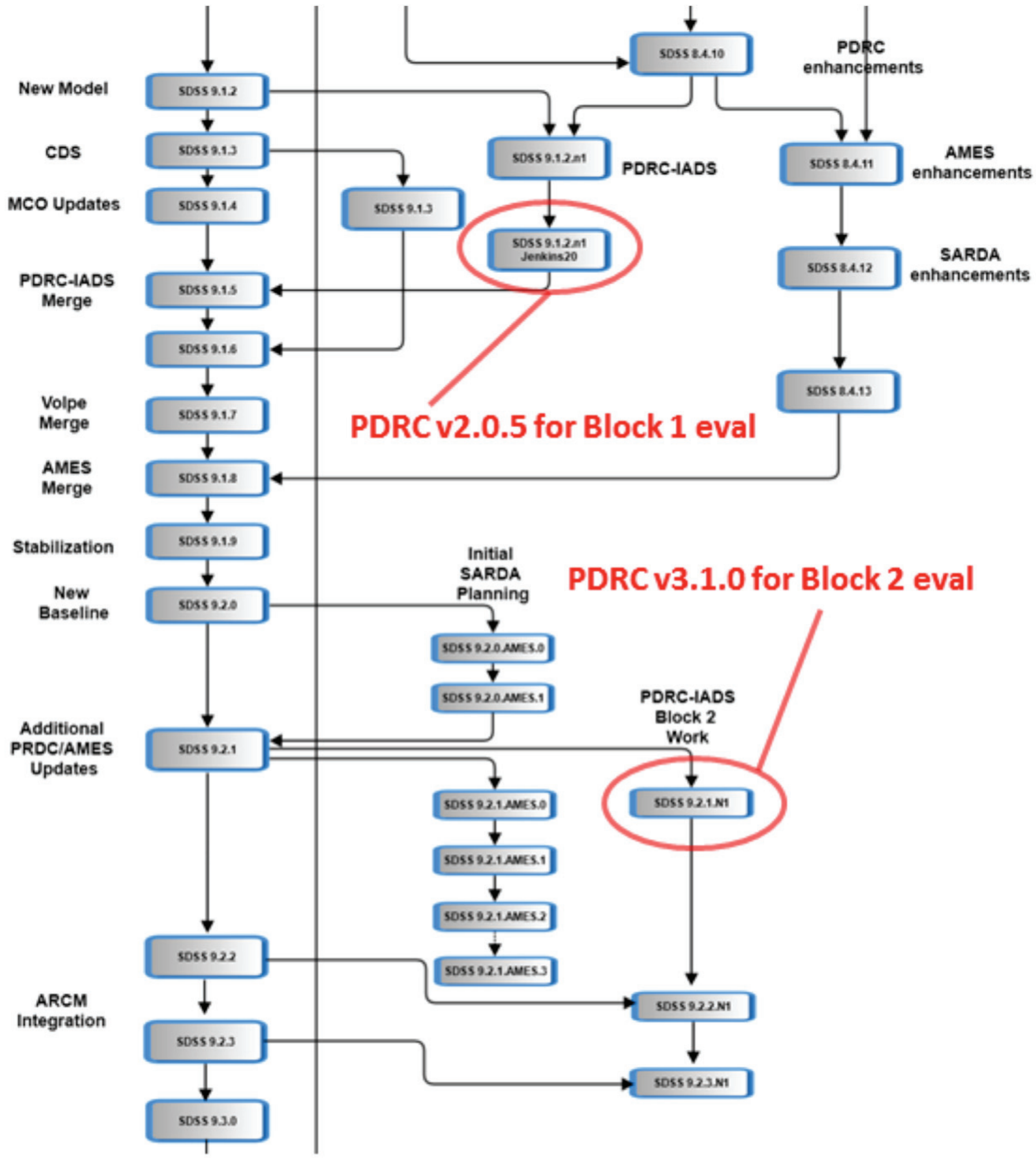

Figure 3:3 – Recent SMS/SDSS development history.

3.3 PDRC two-way interface overview

PDRC is primarily a systems integration research activity where two relatively mature decision support tools (rTMA and SDSS) have been combined to create a new capability. PDRC technology enables rTMA and SDSS to share information that reduces uncertainty in the tactical departure scheduling process. This technology includes a two-way communications interface between rTMA and SDSS, represented by the double-headed arrows on the left side of Figure 3:1.

The PDRC two-way interface leverages existing SDSS and TMA communications capabilities. The rTMA-to-SDSS interface uses the existing SDSS interface to the TMA Collaborative Arrival Planning (CAP) data feed. The CAP message set was extended to include PDRC scheduling information. The new SDSS-to-rTMA interface is an extension of SDSS' SWIM-based TSDE interface.

A primary function of this interface is to deliver the SDSS OFF time predictions and runway predictions to rTMA. Specifically, these are the Undelayed and Predicted Coordinated OFF Times (UCOT and PCOT). UCOT is the earliest time that SDSS predicts that the flight could take off if there is no congestion on the airport surface. The PCOT is the time that SDSS projects that the flight would take off based on its scheduling of aircraft actions under the active scheduling constraints. PCOT is used by rTMA in tactical departure schedule calculations. UCOT can be used to project the congestion-induced ground delay a flight will experience, which is used by PDRC to attempt to provide a "credit" for ground delay when establishing the tactical departure schedule. The runway assignments are used by the rTMA system to schedule the aircraft. Previously the runway assignment was subject to error due to being manually selected.

The PDRC two-way interface also enabled a novel solution to a flight plan and airborne surveillance data source requirement. Typically SDSS uses Traffic Flow Management System (TFMS) data as the source for flight plans and airborne surveillance data. PDRC requires the flight data in the component decision support tools (i.e., rTMA and SDSS) to correspond as closely as possible. This requirement was met by using rTMA's existing CAP interface to deliver Center and TRACON flight plan and track data to SDSS. This approach follows the example of earlier NASA work [19], providing SDSS with new, high-quality surveillance data sources and ensuring that both PDRC components are operating on identical sets of flight information. This approach also required a significant modification to EDC's Input Source Manager (ISM) process to allow information for all flights to be passed to the surface system. The operational EDC design compartmentalizes flight plan information and track data, thereby limiting the ability to exchange flight information among EDC processes. However, the surface system required all flights and all tracks (AFAT) from available flights. Therefore, the EDC logic was modified to allow AFAT to be passed through CAP into the surface system but the core scheduling of EDC to remain as is.

3.4 PDRC software architecture

A high level diagram of the architecture used in PDRC is depicted in Figure 3:4. This diagram depicts the primary PDRC components involved in a configuration in which a single surface system connects to a single rTMA system. The architecture, however, is capable of supporting multiple rTMA systems connected to a single SDSS, as well as daisy chaining data to multiple downstream instances of PDRC.

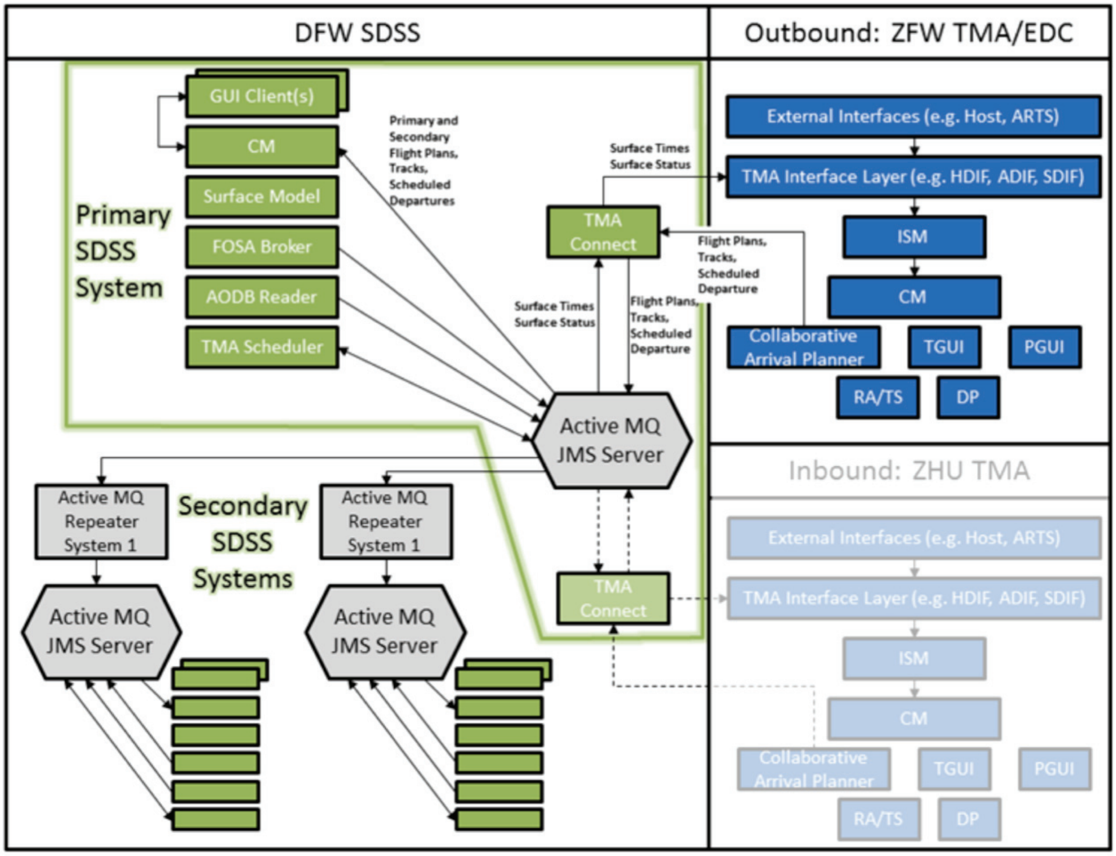

Figure 3:4 – PDRC prototype software architecture.

4 Enroute system design details

This section presents details of the rTMA component of the PDRC prototype software system. Figure 4:1 shows the primary Computer Software Configuration Items (CSCIs) for TMA/rTMA. New CSCIs developed for the PDRC-IADS research activity are shown in gray. Several other CSCIs were modified to implement PDRC functionality. The new and modified CSCIs are described in the following subsections.

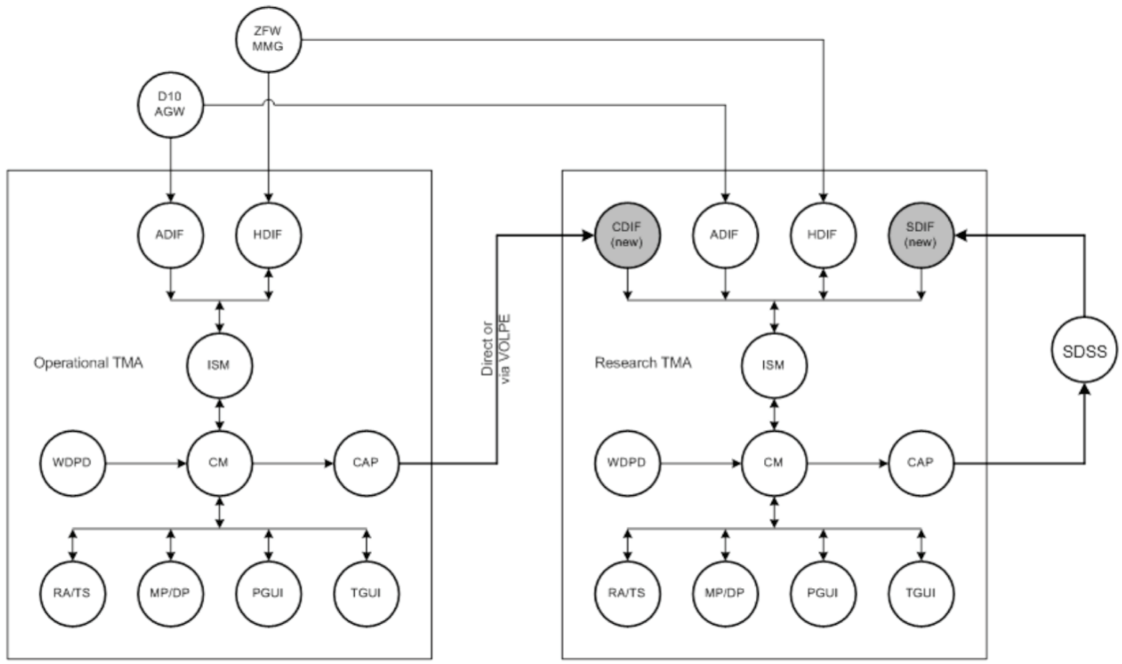

Figure 4:1 – Research TMA data interfaces for PDRC.

4.1 Operational TMA CAP to rTMA CDIF interface

A new Research TMA CSCI was created to allow synchronization with the operational TMA system at ZFW. This CSCI was patterned after other existing TMA input processes like Host Data Interface (HDIF), ARTS Data Interface (ADIF) and ETMS Data Interface (EDIF). This new process connects to an operational or research CAP XML. The new process is named CAP Data Interface (CDIF).

An adaptation was used to configure the interface, to specify login information, and to specify exactly what data is desired from the CDIF feed. After establishing a connection, each incoming XML application data message is converted into an equivalent internal TMA message and sent on to ISM. CDIF was designed to support one or more such CAP connections, although only one connection was used for PDRC.

The TMA CAP ICD was followed for this interface. The rTMA system acted as a client and requested only certain data from the server. In particular, only `<con>` data elements will be requested to obtain configuration and flow information updates from the TMA system. The `<con>` data element contains relevant configuration information, which could be used to keep rTMA in sync with the ZFW operational TMA system. The data is of an uncompressed XML format that was logged before any TMA processing is performed. No changes to the incoming CAP interface will be possible since the server could be part of an operational system.

NOTE: The CDIF interface enables rTMA to mirror or follow the state of another TMA system. This capability was considered highly desirable for PDRC development and shadow evaluations; however, it was not used due to the unavailability of a CAP data feed from ZFW's operational TMA. Also, the capability to mirror an operational system via the CAP feed is not fundamental to the PDRC operational concept or necessary for the operational evaluations.

4.2 SDSS to rTMA SDIF interface

A new rTMA Unix process (CSCI) was created to allow the rTMA system to receive information from the SDSS system at DFW. This CSCI was patterned after other TMA input processes like HDIF, ADIF and EDIF.

The SDSS side of the interface provides the rTMA system with a feed of SDSS data using an extension of the FOSA interface [46]. An adapter converts the FOSA-enabled data to XML over TCP/IP sockets, using an HTTP session layer.

Adaptation is used to configure the interface as required. After establishing a connection, each incoming SDSS XML application data message is converted into a new internal TMA message and sent to rTMA's ISM CSCI. For more information on each message, see Section 6.3.

4.3 Research TMA DP Component

A key enabling technology of TMA/EDC is the Meter Point Dynamic Planner (MPDP) component. The MPDP contains the algorithmic logic required to ensure de-confliction of aircraft over adapted metering points.

Currently, the MPDP takes a manually entered ready time (i.e., OFF time prediction) as input and uses this information to find available slots in the overhead stream and also to calculate expected ground delay. PDRC technology allows the manual ready times to be replaced with automated surface OFF time estimates. The MPDP scheduler generates the earliest departure time that is at or after the Predicted Coordinated OFF Time (PCOT).

In an effort to give a flight credit in the tactical departure scheduling process for delay experienced on the airport surface, the TMA Dynamic Planner and EDC Meter Point Dynamic Planner algorithm were modified. Figure 4:2 gives a hypothetical example of how this credit works. The left hand side of this diagram represents the "As-Is" case of how TMA/EDC scheduling works today. The right-hand side Figure 4:2 illustrates the situation when surface delay credit is applied. Both the "As-Is Today" and "With Surface Delay Credit" portions of Figure 4:2 show the same five aircraft (call signs Airborne1 through Airborne4 and Surface1) on representations of TMA timelines. As is typical for TMA, undelayed times to the meter point are shown on the left side of the timeline while scheduled times to the meter point are on the right side of the timeline.

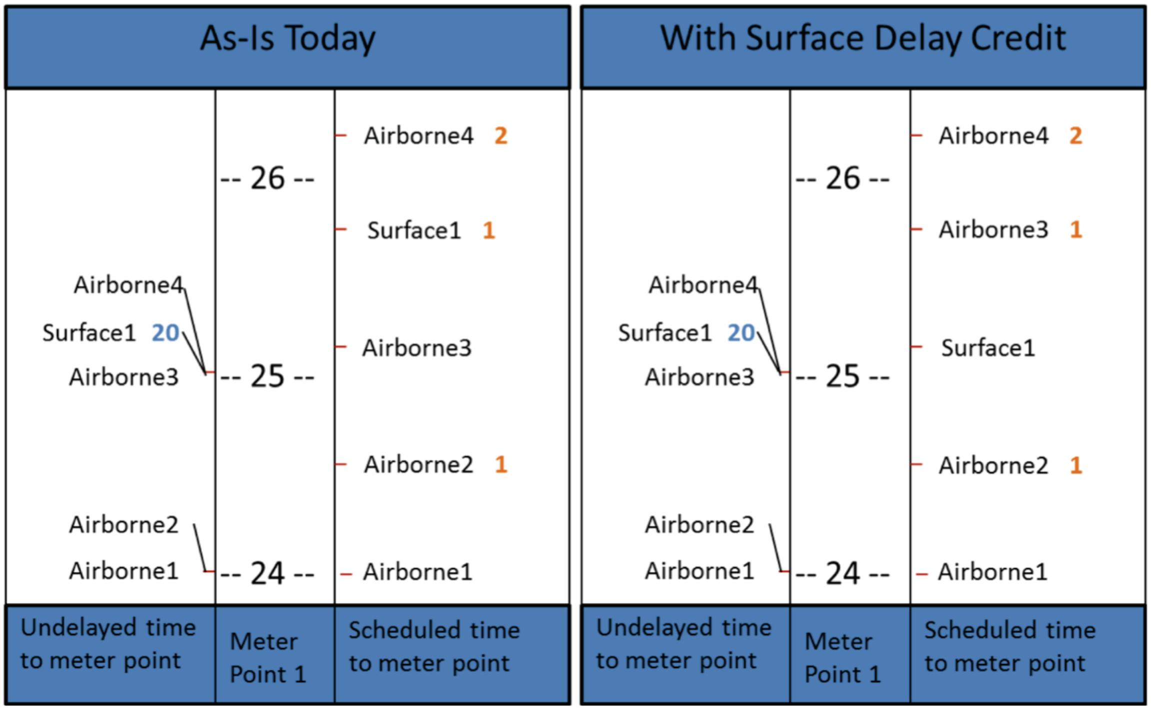

Figure 4:2 - PDRC surface delay credit.

Note that several of the aircraft have undelayed times to the meter point that are nearly identical. In this case, TMA/EDC gives priority to the aircraft with the earliest undelayed time in an effort to supply a first-come-first-serve (FCFS) schedule. The delays assigned by the scheduler are shown in orange on the right side of the timeline. These delays are required to ensure all metering constraints and wake vortex separations are met at the time of meter point crossing. The blue number that is shown on the surface aircraft on the left side of the timeline represents the estimated amount of surface delay this aircraft will incur by the time it departs. In this example, aircraft Surface1 has a 20 minute surface delay and in addition is assigned a one minute airborne delay.

In today's system it is not possible to give credit for surface delay because an objective measure of the delay is not available to the enroute scheduler. However, PDRC sends both the undelayed and predicted time to the enroute scheduler on a periodic basis. The enroute scheduler can then use this information to obtain a better picture of FCFS from a multi-domain perspective. In this case, the scheduler uses the 20 minute ground delay to prioritize Surface1 ahead of Airborne3 (see right-hand side of Figure 4:2). The scheduler does not advance Surface1 any earlier than this placement in the schedule (e.g., ahead of Airborne2) because it will not allow the aircraft to be scheduled at any time earlier than when the surface system predicts it will be OFF.

4.4 Research TMA User Interface Component

A primary mechanism that enables inter-facility collaboration via PDRC technology is the rTMA scheduling panel. New dialog panels have been added to the existing TMA/EDC system to allow the TMC to see all pending surface release time requests, to allow the TMC to select a specific request from the list, and to allow actions to be performed related to the request. Aircraft specific highlighting on the primary Timeline display allows for easy identification of aircraft that are currently requesting a release time, as well as tracking other states as they progress through the scheduling process.

Figure 4:3 illustrates the new panel that enables the TMC to view all pending release time requests sent from SDSS to rTMA. A request for release time from SDSS will change the status displayed in the last column of the list. TMCs can select the "TMA/SMS Status" button to sort the list of departures by status, which will group together all of the flights with pending release time requests.

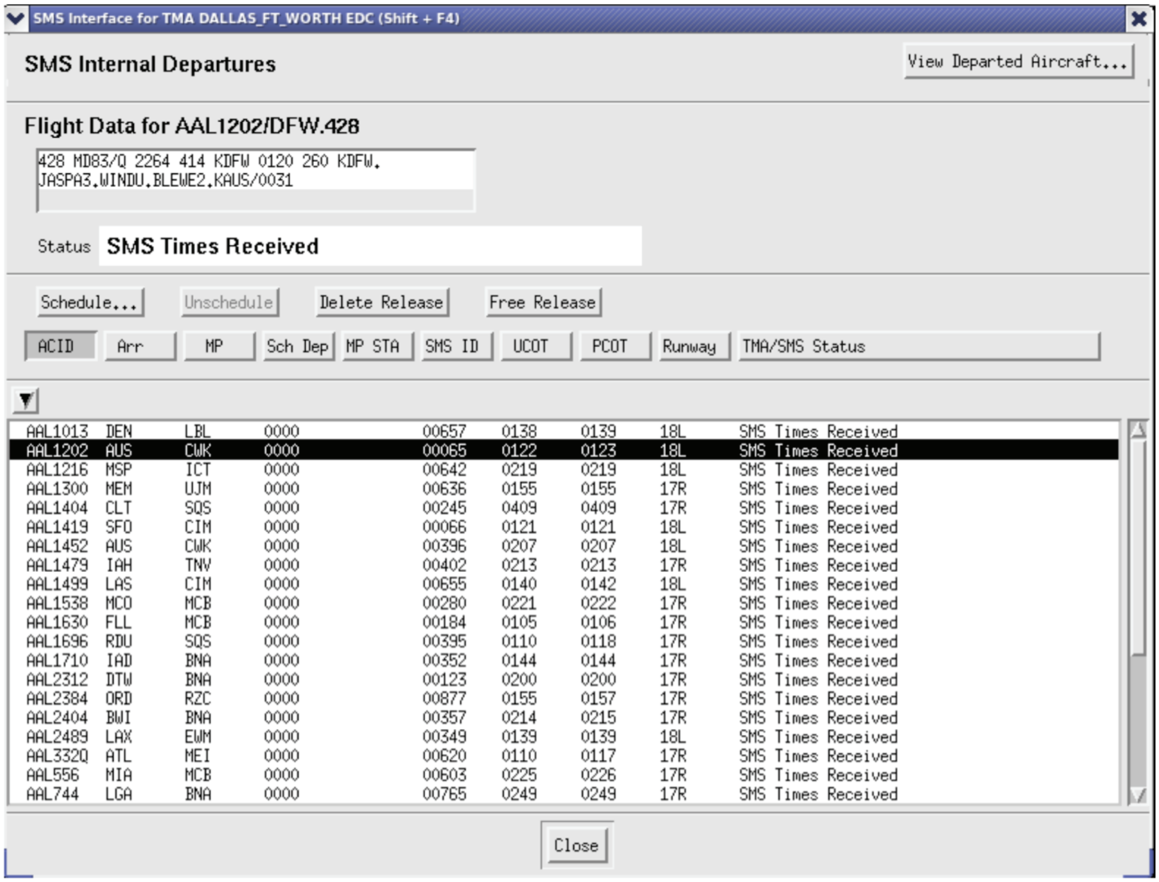

Figure 4:3 – New rTMA user interface panel allows viewing of all pending SMS release time requests.

Modifications to the existing TMA Schedule a Departure panel, which include surface OFF time estimates and PDRC status information, are shown in Figure 4:4. The primary change has been the addition of an "SMS Time" field to the "Aircraft Ready Time" section of the panel. The predicted OFF time (i.e., PCOT) computed by SDSS is automatically entered into this field so that it is ready for use when the Scheduled a Departure panel is activated. An "SMS State" indicator has also been added to the panel immediately above the "Aircraft Ready Time" area. In the example shown in Figure 4:4 the SMS state is "Release Requested."

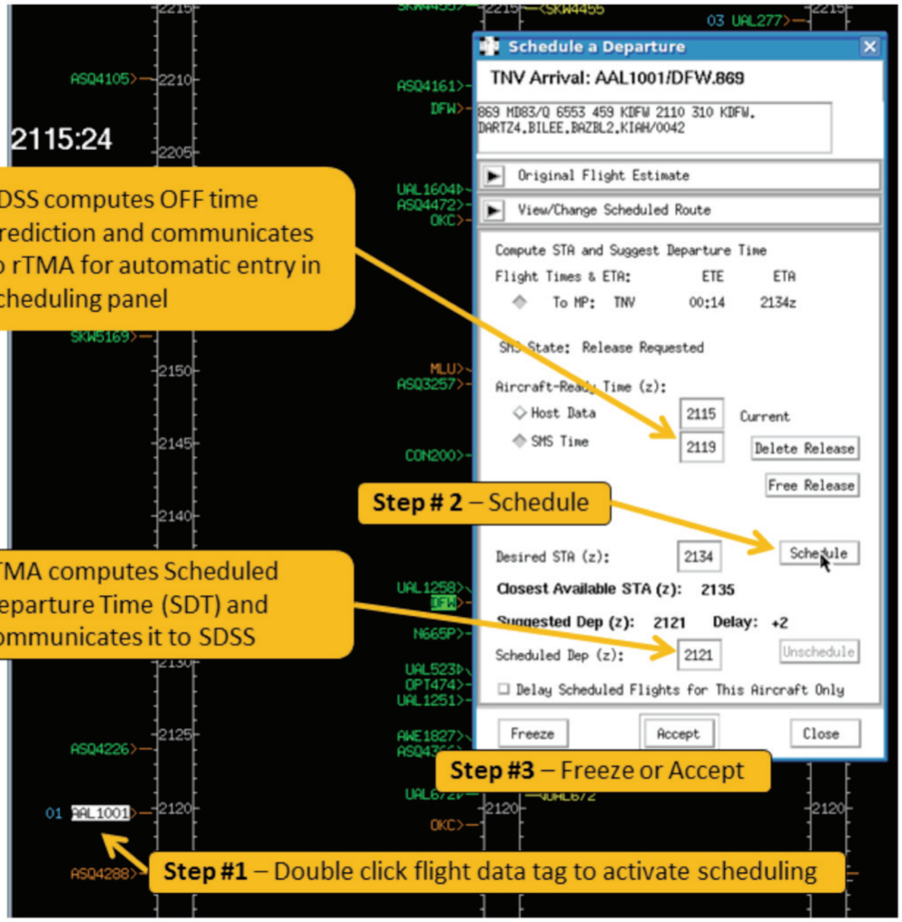

Figure 4:4 – Modifications to rTMA Schedule a Departure panel.

Annotations in Figure 4:4 show the three primary steps involved in scheduling a departure. The operational usage of this panel is described in greater detail in the ConOps [1].

5 Surface system design details

This section presents details of the surface component of the PDRC prototype software system. As noted in Section 3, NASA's SMS research system and the FAA's SDSS research system are being developed, and the names are often used interchangeably. SDSS modifications for the PDRC-IADS research activity have focused on the SDSS/rTMA two-way communications, which are handled via the Tactical Surface Data Exchange (TSDE) interface. TSDE was previously known as Flight Operator Surface Application (FOSA) and a number of the SDSS software processes still use the FOSA terminology. Figure 5:1 provides an overview of the SDSS TSDE/FOSA architecture.

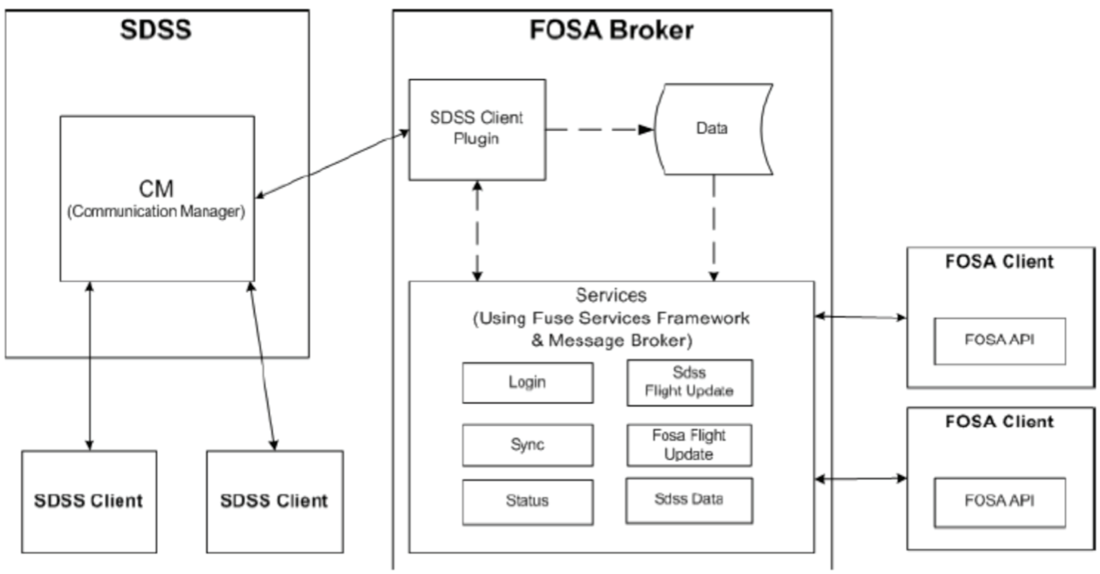

Figure 5:1 – SMS TSDE (aka FOSA) architectural overview.

5.1 Surface TMA Connect Process

A module was developed for the SDSS system to manage the flow of data from SDSS to rTMA, and from rTMA to SDSS. This module is known as 'TMA Connect'. TMA Connect accesses flight data from the SDSS Communications Manager (CM), and periodically sends relevant subsets of this data to TMA through its Surface Data Interface (SDIF), including the current OFF times (unconstrained and predicted) and runway assignments.

TMA connect is responsible for establishing the connections with each required TMA/EDC system, parsing the messages from TMA and forwarding them to the appropriate SDSS process for further processing.

The TMA connect process was originally created to obtain data from the Memphis TMA Collaborative Arrival Planner process. Given the Memphis TMA connection went through Volpe and did not have access to all TMA data, fairly substantial changes were required for PDRC development in order to enable new messages. TMA connect was also modified to handle multiple TMA connections. The processing is complicated by the fact that each TMA system has its own unique identifier generated by the TMA Input Source Manager (ISM) process (this was required with the expansion of TMA to multiple centers). So any process interfacing with multiple TMA systems must also handle flight matching between those systems.

The TMA connect process handles a significant portion of the state logic required for successful communication with TMA during the tactical scheduling process.

Lastly, the TMA connect process is responsible for logging of the raw incoming messages from TMA/EDC. This logging is very important for debugging and analysis, especially given the multi-domain nature of PDRC.

5.2 Surface TSDE Interface and ActiveMQ

The primary mechanism to obtain and relay information with the surface system for PDRC activities is via the Tactical Surface Data Exchange (TSDE) interface [46]. The TSDE interface is also used to communicate information from air carriers into the surface system at DFW. A fundamental component used by TSDE for message brokering is Apache ActiveMQ. ActiveMQ, along with components from the Fuse message broker, are used in SDSS to broker Java Messaging Service (JMS) messages between external and internal producers and consumers. SDSS does this work by utilizing a set of domain specific queues and topics.

The TSDE interface is implemented with NAS System Wide Information Management (SWIM) [37] compliance in mind. TSDE utilizes SWIM's Fuse product line as an information broker to clients.

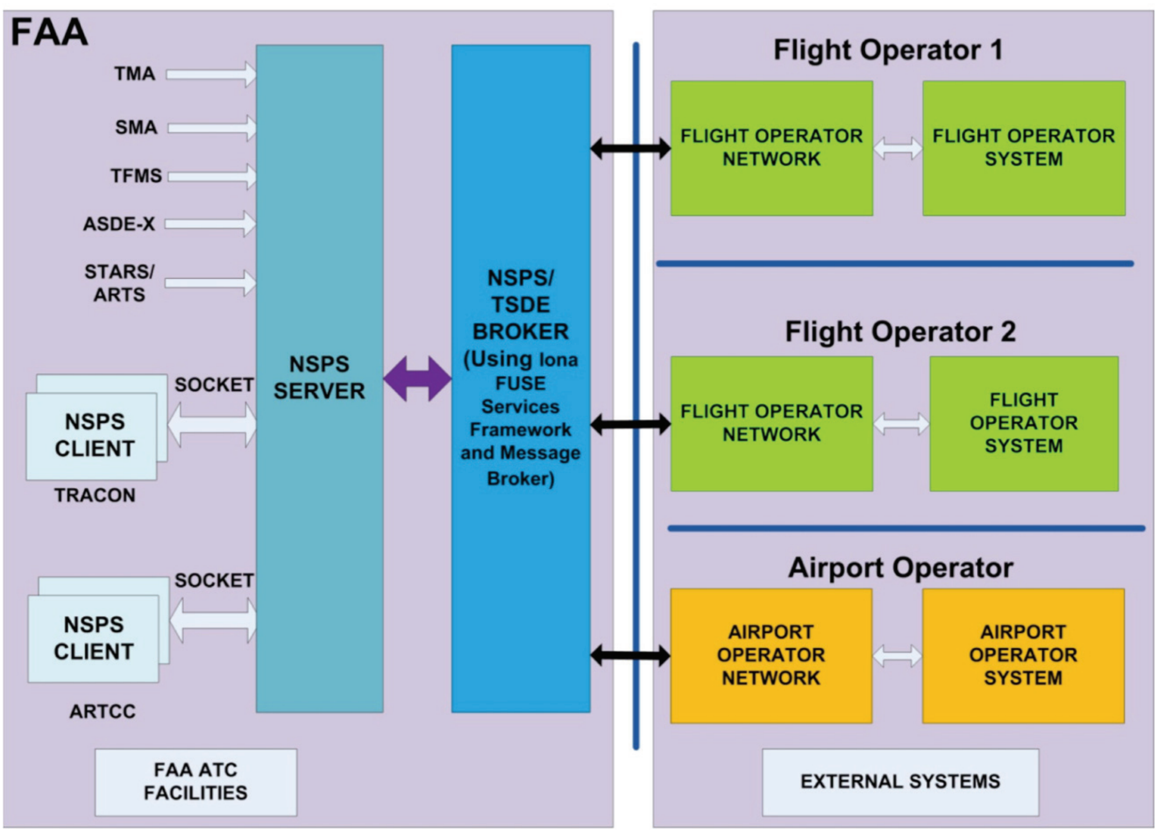

Figure 5:2 – TSDE interface to flight operator systems.

TSDE services are generally implemented at a local level to provide data on behalf of that particular airport, resulting in a multitude of instances for each service. The local TSDE Services supply information from the perspective of the Flight/Airport Operators at the specific airport at which the service is required, and the client requests the service either directly from the specific location or from a service broker that redirects the request to the appropriate location. Figure 5:2 illustrates how a typical TSDE interface might be configured. In the case of PDRC the TSDE interface is used in four areas: outgoing surface interchange with TMA, incoming processing of a direct airline interface, incoming interface with a secondary commercial provider of gates and estimated pushback times, and outgoing ActiveMQ repeaters capable of serving additional instances of PDRC.

5.3 Surface Communications Manager (CM)

Modifications have been made to the SDSS CM to manage data from the rTMA and process the state management of CFR flights.

Table 5:1 shows the actions and flight state changes of the surface system during a CFR restriction. This includes a description of each CFR state value, how that state may be reached, and the actions that are performed once that state is reached.

Table 5:1 – SDSS state changes for PDRC.

CFR State	Description	Trigger	Action
NONE	The flight is not subject to a CFR restriction.	Initial operating state. Also set when a controlling CFR restriction is removed or when a flight that was subject to a CFR restriction has departed.	If changed from another state to NULL, send rTMA a message to remove the flight from rTMA/SDSS Internal Departure Active Coordination.
REQUEST_PENDING	The flight is subject to a CFR restriction but no request for a scheduled departure time (SDT) has been made.	First state after a flight is discovered as subject to a CFR restriction (e.g., meets airport or jet airway specific in APREQ event). This state is also reached when a request is canceled by the SDSS user.	If reached because of a cancelation, send a cancelation message to rTMA.
RESPONSE_PENDING	A request for departure time has been submitted but no response has been received.	Set when the user initiates a request for release time via the surface user interface. This state is also reached when a scheduled departure time is rejected.	Send a request message to rTMA for the initial request. Send a reject message if the scheduled time was rejected.

FREE_RELEASE	A response has been received from rTMA that frees the flight to be released without regard to the CFR restriction.	Set when a response message from rTMA is received.	Schedule the flight as if no CFR restriction were in place. NOTE: This means that rTMA will not be used to schedule the flight (generally due to low traffic volume) and the ARTCC will accommodate the aircraft when departure occurs.
SDT_RECEIVED	A response (with scheduled departure time) has been received from rTMA but has not yet been accepted or rejected.	Set when a response message from rTMA is received.	Wait for the user to accept or reject the scheduled departure time or to cancel the request.
SDT_ACCEPTED	The scheduled departure time from rTMA is accepted. This step is optional.	Set when the user accepts the scheduled departure time.	Set the flight's departure schedule appropriately to meet the scheduled departure time

As described in Section 6.2.5 of the ConOps [1], acceptance of the SDT by the Tower TMC/FLM was deemed to be optional during the PDRC operational evaluations.

5.4 Surface User Interface Component

The SDSS prototype surface system includes a full-featured user interface with existing functions for managing the CFR (referred to as APREQ in SDSS software and documentation) scenario that is the focus for PDRC-IADS research. These standard SDSS functions are documented elsewhere [44, 45]. This section focuses on changes made to the SDSS user interface for the PDRC-IADS research activity.

A number of client changes were requested and implemented prior to Block 1 execution of PDRC. These were primarily oriented toward implementing a display that would allow the users to change the upcoming runway assignments to better balance traffic on the East and West side of DFW airport. These changes included allowing the display of the APREQ and EDCT times on one line, showing the current APREQ state on the timeline, allowing for display of APREQ times to seconds-level precision, and displaying EDCT and APREQ times in white on the timeline. Figure 5:3 shows the SDSS user interface as is commonly configured by the DFW ATCT.

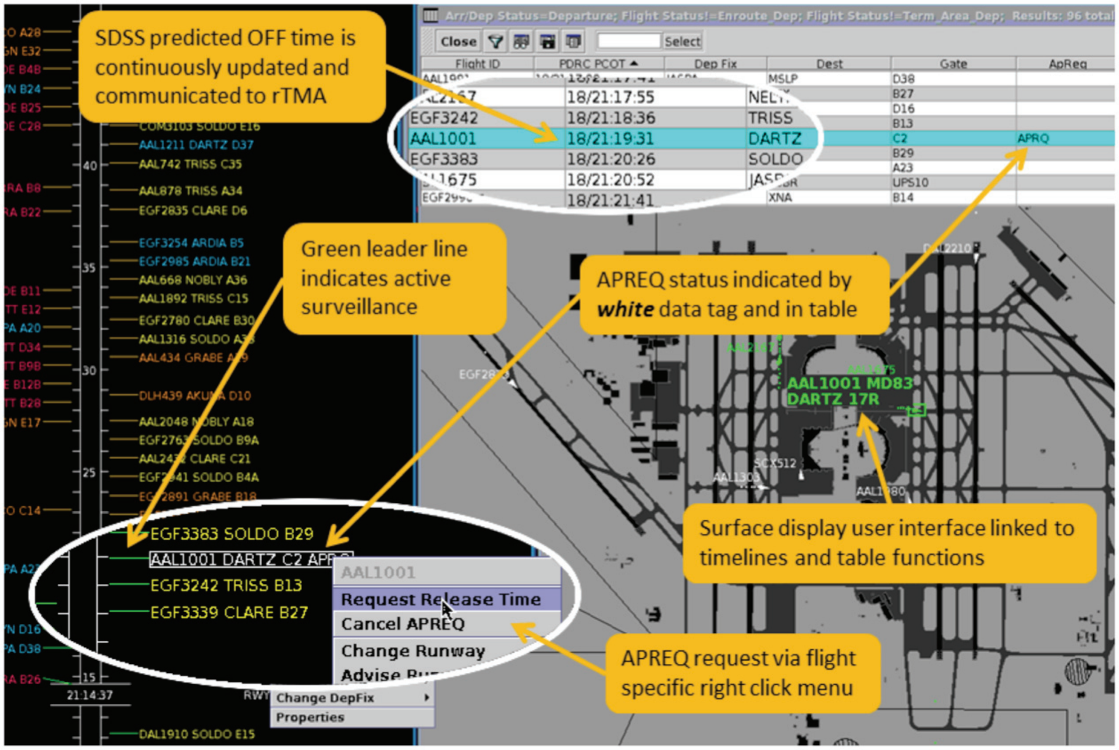

Figure 5:3 – SDSS user interface elements used in PDRC scheduling.

5.5 Surface Secondary Carrier Interface Component

The surface system has a component that is presently called the Airline Operations Database Reader (AODB reader). This component is responsible for periodically ingesting gate assignments and pushback time estimates from a secondary source. Currently this secondary source is FlightStats, but the prior source was that of DFW Airport's AODB system from whence it retained its name.

The AODB reader also uses the TSDE interface to obtain, parse and process data from a secondary source. In addition to updating gate assignments, the SDSS system also updates estimated time of pushback based upon this data. The general hierarchy of airline data sources is that the direct airline data is of higher priority that the secondary source. If no gate information is received from either a direct airline connection or the secondary source, then the SDSS system uses the default gates assignment as specified in adaptation.

6 Interface description

Section 3.3 provided an overview of the two-way interface that has been developed for PDRC.

6.1 General Characteristics

Figure 6:1 shows the line of demarcation between rTMA and SDSS.

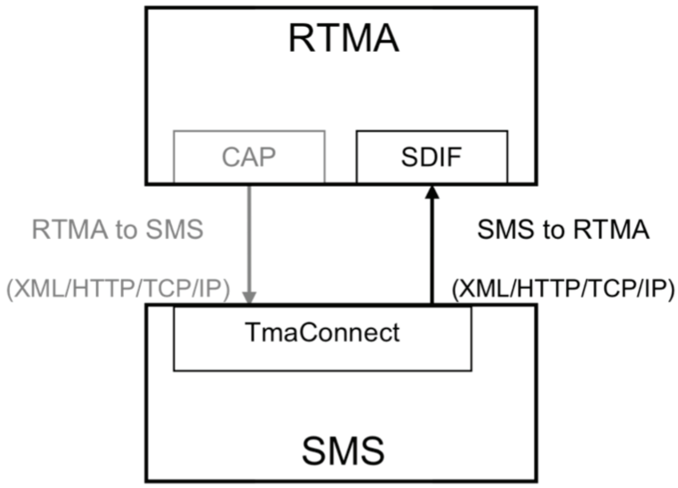

Figure 6:1 – Interface between rTMA and SMS

SDSS sends data from its TmaConnect component to the rTMA Surface Data Interface (SDIF) described in Section 4.2. This data is sent as XML-encoded messages over an HTTP connection. In a similar fashion, the rTMA to SDSS interface also sends XML-encoded messages over an HTTP connection from the Collaborative Arrival Planning (CAP) process to the SDSS TmaConnect component. Details of the two interfaces are presented in the following subsections.

6.2 rTMA to SDSS connection

Communications from rTMA to SDSS heavily leverage the Collaborative Arrival Planning (CAP) interface that is a standard feature of the FAA's operational TMA system. Since rTMA was derived from operational TMA the standard CAP interface was already in the software. Additionally, the FAA STBO Project had already developed the TmaConnect component so that SDSS could use CAP data elements.

The standard TMA CAP interface is well documented [42]. Extensions were required to implement PDRC functionality. These extensions are detailed below. The changes implemented to the native CAP message are highlighted in red below for easy identification. The changes were minimal because TMA CAP had a fairly comprehensive list of data that was already available.

```
<env envSrce="TMA.Zxx.FAA.GOV" envTime="ENVELOPE-TIME">
  <tma msgId="MESSAGE-ID" msgTime="MESSAGE-TIME">

    <!-- Aircraft Information Category -->
    <air airType="AIRCRAFT-EVENT-TYPE" tmaId="TMA-ID">

      <!-- Flight Plan Information Group -->
      <flt>
        <old>AIRCRAFT-ID</old>
        <aid>AIRCRAFT-ID</aid>
        <dap>DEPARTURE-AIRPORT-NAME</dap>
        <apt>ARRIVAL-AIRPORT-NAME</apt>
        <fps>FP-STATUS</fps>
        <acs>AIRCRAFT-STATUS</acs>
        <typ>AIRCRAFT-TYPE</typ>
        <eng>ENGINE-TYPE</eng>
        <bcn>BEACON-CODE</bcn>
        <spd>FILED-SPEED</spd>
        <ara>ASSIGNED-REQUESTED-ALTITUDE</ara>
        <ina>INTERIM-ALTITUDE</ina>
        <trw>TRACON-RUNWAY-NAME</trw>
        <drw>DEPARTURE-RUNWAY-NAME</drw>
        <tds>CURRENT-TRACK-DATA-SOURCE</tds>
        <cfx>COORDINATION-FIX-NAME</cfx>
        <ctm>COORDINATION-TIME</ctm>
        <etd>ESTIMATED-DEPARTURE-TIME</etd>
        <std>SCHEDULED-DEPARTURE-TIME</std>
        <etm>EDCT-TIME</etm>
        <ucot>UNDELAYED-COORDINATED-OFF-TIME</ucot>
        <pcot>PREDICTED-COORDINATED-OFF-TIME</pcot>
        <aot>ACTUAL-OFF-TIME</aot>
        <sfs>SMS-FLIGHT-STATUS</sfs>
        <srs>SMS-REQUEST-STATUS</srs>
        <est>EDCT-STATUS</est>
        <a10>FIELD-10A-ROUTE</a10>
        <b10>FIELD-10B-ROUTE</b10> (future)
        <c10>FIELD-10C-ROUTE</c10> (future)
        <tcr>TMA-CONVERTED-ROUTE</tcr>
      </flt>
```

<ucot>	yyyy-mm-ddThh:mm:ssZ	SMS Undelayed Coordinated Off Time
<pcot>	yyyy-mm-ddThh:mm:ssZ	SMS Predicted Coordinated Off Time
<aot>	yyyy-mm-ddThh:mm:ssZ	SMS Actual Off Time
<sfs>	[0-9]{1-2}	SMS Flight Status - as received from SMS

<**srs**>	 NONE ACTIVE PENDING SENT ACCEPTED REJECTED FREE	SMS Request Status NONE - No SMS flight data received ACTIVE - SMS flight data received PENDING - SMS Request for Release Time received, awaiting internal departure scheduling SENT - TMA SDT sent to SMS, awaiting SMS accept/reject ACCEPTED - SMS accepted the SDT REJECTED - SMS rejected the SDT FREE - not scheduled, exempted from Call for Release
<**est**>	FAA	EDC	Estimated Departure Clearance Status FAA – Using FAA coordination time or STD EDC – Using EDCT time (if ETM non-zero)

<**int**> - TMA Interface Status Information Group

	Format / Range	**Element Description**
<**ifn**>	[A-Z][A-Z0-9]{2}	Interface name Zxx – Host or TMA interface (e.g., ZFW) xxx – ARTS/STARS interface (e.g., D10) WIF – Weather interface TFM – TFMS interface (e.g., TFM) xxx - SMS interface (e.g., DFW)
<**ift**>	HOST ARTS STAR WXIF TFMS TMA SMS	Interface type

6.3 SDSS to rTMA connection

This section provides the design characteristics for an interface between the Surface Management System (SMS) and the Research Traffic Management Advisor (rTMA). This document provides data descriptions, formats, and protocols that are used by the two systems to exchange information.

6.3.1 Low level Interface Design Requirements for PDRC Software

This section states the interface requirements between an SDSS and an rTMA.

a. SDSS SHALL [0001] listen for incoming TCP connection requests on an adapted TCP port number as defined in SDSS adaptation.

b. rTMA SHALL [0002] initiate a TCP connection to the SDSS at an adapted IP address and port.

c. SDSS SHALL [0003] accept a TCP connection request from an rTMA if it is connecting from an IP address defined in SDSS adaptation.

d. SDSS SHALL [0004] allow each rTMA to make one TCP connection per IP address.

e. Upon connection establishment, rTMA SHALL [0005] send an HTTP GET request to SDSS as described in subsequent sections of this ICD.

f. Upon successful validation, SDSS SHALL [0006] send an HTTP 200 OK response to rTMA as described in subsequent sections of this ICD. If validation fails, SDSS SHALL [0007] send an HTTP 400 BAD response to rTMA and close the connection.

g. After successful connection establishment and validation, SDSS SHALL [0008] send HTTP chunked XML application data messages to rTMA as described in subsequent sections of this ICD. At this point the communication interface is in one-way mode from SDSS to rTMA.

h. rTMA SHALL [0009] log all data transmitted to and received from an SDSS.

i. SDSS SHALL [0010] log all data transmitted to and received from rTMA.

j. SDSS SHALL [0011] transmit time information in a standardized format using a subset of the ISO 8601 protocol as specified in subsequent sections of this ICD.

6.3.2 Interface Overview

The rTMA client connects to the SDSS server via an HTTP GET request with the Uniform Resource Locator (URL) containing identification information. Once authenticated, the rTMA client receives an HTTP OK response message. The SDSS will then transmit application data of interest to rTMA.

After the initial exchange of HTTP request/response information, each transmission from SDSS will contain an **<env>** envelope that will contain one <**sms**> element. Each <**env**> element is sent as a separate HTTP chunk over an established TCP/IP connection. Use of the HTTP protocol is described in Section 6.3.4.

Each <**sms**> element may contain a variety of information. Information about flights departing the airport managed by SDSS will be sent in <**air**> elements. Following the initial transmission of all information related to an aircraft (called an add), only changes will be sent (called an amendment) along with aircraft identifying information. Eventually the aircraft should be removed by SDSS when it is no longer of interest (called a delete). SDSS may also transmit request release time **<rrt>** and accept/reject release time <**art**> messages.

For ease of parsing, tag names follow a certain style. All tags are basically three characters (e.g., <**aid**>). Each element type is given a unique tag name that should be used consistently throughout. The goal is to make the XML data concise yet readable and easy to parse.

Each message must contain the <**aid**> and <**iid**> elements for identification purposes. The **<iid>** element is obtained from the <**air airType="NEW" tmaId="F12345"**> in the CAP data feed. The <**iid**> corresponds to the value of the **tmaId** attribute.

6.3.3 XML Message Overview

This is an overview of the XML message structure showing the top few levels of elements and indicating the categories or groups of information contained in each element.

6.3.3.1 XML Data Categories

There are three categories of data sent: `<air>`, `<rrt>`, and `<art>`. Each category will be sent in a separate `<sms>` element and will not be mixed.

6.3.3.2 XML Message Structure

The following pages contain the detailed view of the XML message structure. The `<sms>` element allows other systems to use this protocol (e.g., `<tma>`, `<tfms>`). White space between elements in the XML is not significant and should be ignored. SDSS messages are sent to the rTMA encapsulated in an outermost `<env>` element. Each `<env>` element can contain one `<sms>` element.

```
<env envSrce="ENVELOPE-SOURCE" envTime="ENVELOPE-TRANSMISSION-TIME">
  <sms msgId="MESSAGE-ID" msgTime="MESSAGE-CREATION-TIME">
   ...
  </sms>
</env>
```

6.3.3.3 Extensibility

In order to provide a transition to future versions of the ICD, implementations should ignore all of the following:

1. Element attributes with an unknown name
2. Elements with an unknown tag including all nested sub-elements up to and including the unknown end tag
3. Order of elements to the maximum extent possible

6.3.3.4 XML Data Messages

6.3.3.4.1 Add Aircraft Message

```
<env envSrce="SMS.DFW.FAA.GOV" envTime="ENVELOPE-TRANSMISSION-TIME">
  <sms msgId="MESSAGE-ID" msgTime="MESSAGE-CREATION-TIME">
    <air airType="NEW">
      <aid>AIRCRAFT-ID</aid>
      <iid>TMA-UNIQUE-ID</iid>
      <dep>DEPARTURE-AIRPORT</dep>
      <dst>DESTINATION-AIRPORT</dst>
      <bcn>BEACON-CODE</bcn>
      <rwy>RUNWAY-NAME</rwy>
      <fst>FLIGHT-STATUS</fst>
      <cfr>CALL-FOR-RELEASE-RESTRICTION</cfr>
      <ucot>UNDELAYED-COORDINATED-OFF-TIME</ucot>
      <pcot>PREDICTED-COORDINATED-OFF-TIME</pcot>
      <aot>ACTUAL-OFF-TIME</aot>
    </air>
  </sms>
```

```
</env>
```

6.3.3.4.2 Amend Aircraft Message

```
<env envSrce="SMS.DFW.FAA.GOV" envTime="ENVELOPE-TRANSMISSION-TIME">
  <sms msgId="MESSAGE-ID" msgTime="MESSAGE-CREATION-TIME">
    <air airType="AMD">
      <aid>AIRCRAFT-ID</aid>
      <old>AIRCRAFT-ID</old>
      <iid>TMA-UNIQUE-ID</iid>
      <dep>DEPARTURE-AIRPORT</dep>
      <dst>DESTINATION-AIRPORT</dst>
      <bcn>BEACON-CODE</bcn>
      <rwy>RUNWAY-NAME</rwy>
      <fst>FLIGHT-STATUS</fst>
      <cfr>CALL-FOR-RELEASE-RESTRICTION</cfr>
      <ucot>UNDELAYED-COORDINATED-OFF-TIME</ucot>
      <pcot>PREDICTED-COORDINATED-OFF-TIME</pcot>
      <aot>ACTUAL-OFF-TIME</aot>
    </air>
  </sms>
</env>
```

6.3.3.4.3 Delete Aircraft Message

```
<env envSrce="SMS.DFW.FAA.GOV" envTime="ENVELOPE-TRANSMISSION-TIME">
  <sms msgId="MESSAGE-ID" msgTime="MESSAGE-CREATION-TIME">
    <air airType="DEL">
      <aid>AIRCRAFT-ID</aid>
      <iid>TMA-UNIQUE-ID</iid>
    </air>
  </sms>
</env>
```

6.3.3.4.4 Request Release Time Message

```
<env envSrce="SMS.DFW.FAA.GOV" envTime="ENVELOPE-TRANSMISSION-TIME">
  <sms msgId="MESSAGE-ID" msgTime="MESSAGE-CREATION-TIME">
    <rrt>
      <aid>AIRCRAFT-ID</aid>
      <iid>TMA-UNIQUE-ID</iid>
      <rwy>RUNWAY-NAME</rwy>
      <cfr>CALL-FOR-RELEASE-RESTRICTION</cfr>
      <ucot>UNDELAYED-COORDINATED-OFF-TIME</ucot>
      <pcot>PREDICTED-COORDINATED-OFF-TIME</pcot>
    </rrt>
  </sms>
</env>
```

6.3.3.4.5 Accept / Reject Release Time Message

```
<env envSrce="SMS.DFW.FAA.GOV" envTime="ENVELOPE-TRANSMISSION-TIME">
  <sms msgId="MESSAGE-ID" msgTime="MESSAGE-CREATION-TIME">
    <art>
      <aid>AIRCRAFT-ID</aid>
      <iid>TMA-UNIQUE-ID</iid>
      <ari>ACCEPT-REJECT-INDICATOR</ari>
    </art>
  </sms>
</env>
```

6.3.3.5 XML Data Element Format

<table>
<tr><th></th><th>Format / Range</th><th>Element Description</th></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td><aid></td><td>[A-Z][A-Z0-9]{2-6}</td><td>Aircraft identifier</td></tr>
<tr><td><old></td><td>[A-Z][A-Z0-9]{2-6}</td><td>Aircraft identifier before AID amendment</td></tr>
<tr><td><dep></td><td>[A-Z][A-Z0-9]{1-4}</td><td>Departure airport name</td></tr>
<tr><td><dst></td><td>[A-Z][A-Z0-9]{1-4}</td><td>Destination airport name</td></tr>
<tr><td><fst></td><td>Integer from 0 to 13</td><td>Flight status (from SDSS-FOSA ICD)
0 - Unknown
1 - Scheduled
2 - Pushback
3 - Taxiing out in the ramp
4 - Taxiing out
5 - Returning to the ramp
6 - In the departure queue
7 - Off
8 – Enroute
9 – Within terminal airspace
10 – On final approach
11 – Taxiing in
12 – Taxiing in within the ramp
13 – In the gate</td></tr>
<tr><td><bcn></td><td>[0-7]{4}</td><td>Beacon code (octal)</td></tr>
<tr><td><rwy></td><td>[1-9][LCR] or
[1-2][0-9][LCR] or
[3][0-6][LCR]</td><td>Departure runway name</td></tr>
<tr><td><iid></td><td>[A-Z][0-9]{5}</td><td>TMA unique identifier (from TMA CAP data)</td></tr>
<tr><td><ucot></td><td>yyyy-mm-ddThh:mm:ssZ</td><td>Undelayed coordinated off-time</td></tr>
<tr><td><pcot></td><td>yyyy-mm-ddThh:mm:ssZ</td><td>Predicted coordinated off-time</td></tr>
<tr><td><aot></td><td>yyyy-mm-ddThh:mm:ssZ</td><td>Actual off-time</td></tr>
<tr><td><cfr></td><td>[A-Z0-9_-/]{0-31}</td><td>Call for release restriction
Name of CFR restriction for flight
<cfr></cfr> if flight not subject to CFR</td></tr>
<tr><td><ari></td><td>ACCEPT | REJECT</td><td>Accept / Reject indicator</td></tr>
<tr><td></td><td></td><td></td></tr>
</table>

6.3.3.6 XML Data Attribute Format

<table>
<tr><th></th><th>Format / Range</th><th>Attribute Description</th></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td>envSrce</td><td>SMS.aaa.FAA.GOV</td><td>Envelope source e.g., SMS.DFW.FAA.GOV</td></tr>
<tr><td>envTime</td><td>yyyy-mm-ddThh:mm:ssZ</td><td>Envelope transmission time</td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td>msgId</td><td>[1-9][0-9]{0-9}</td><td>Message identifier</td></tr>
<tr><td>msgTime</td><td>yyyy-mm-ddThh:mm:ssZ</td><td>Message creation time</td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td>airType</td><td>NEW | AMD | DEL</td><td>Aircraft data type
NEW – new aircraft
AMD – amended aircraft data
DEL – delete aircraft</td></tr>
</table>

6.3.3.7 Example XML Messages

```
<env envSrce="SMS.DFW.FAA.GOV" envTime="2010-04-01T11:57:56Z">
  <sms msgId="1294" msgTime="2010-04-01T11:57:56Z">
    <air airType="NEW">
      <aid>AAL2139</aid>
      <iid>F00635</iid>
      <dep>DFW</dep>
      <dst>ATL</dst>
      <bcn>0427</bcn>
      <rwy>27L</rwy>
      <fst>1</fst>
      <cfr>KATL_0600_0900</cfr>
      <ucot>2010-04-01T12:34:56Z</ucot>
      <pcot>2010-04-01T12:34:56Z</pcot>
    </air>
  </sms>
</env>
```

```
<env envSrce="SMS.DFW.FAA.GOV" envTime="2010-04-01T12:15:19Z">
  <sms msgId="1367" msgTime="2010-04-01T12:15:19Z">
    <air airType="AMD">
      <aid>AAL2139</aid>
      <iid>F00635</iid>
      <bcn>0428</bcn>
      <fst>2</fst>
    </air>
  </sms>
</env>
```

```
<env envSrce="SMS.DFW.FAA.GOV" envTime="2010-04-01T12:35:15Z">
  <sms msgId="1367" msgTime="2010-04-01T12:35:15Z">
    <air airType="AMD">
      <aid>AAL2139</aid>
      <iid>F00635</iid>
      <bcn>0428</bcn>
      <fst>7</fst>
      <aot>2010-04-01T12:34:38Z</aot>
    </air>
  </sms>
</env>
```

```
<env envSrce="SMS.DFW.FAA.GOV" envTime="2010-04-01T12:45:09Z">
  <sms msgId="1675" msgTime="2010-04-01T12:45:09Z">
    <air airType="DEL">
      <aid>AAL2139</aid>
      <iid>F00635</iid>
    </air>
  </sms>
</env>
```

6.3.4 HTTP Protocol

HTTP/1.1 will be used to encapsulate the XML application data on the TCP/IP connections. HTTP/1.1 allows for data transmission of large data sets in small pieces. At this time, no data compression will be used. HTTP headers would only be exchanged at connection establishment. Once the SDSS sends rTMA a response header, SDSS may begin transmitting XML application data to the rTMA. Each transmission would be sent using the HTTP/1.1 chunked protocol indicated in the **`"Transfer-Encoding: chunked"`** header field.

In the following descriptions, the fields enclosed in *<lf>* in italics do not represent XML data, rather they are field separators for readability in this document only. E.g., *<lf>* indicates the US-ASCII line feed character `'\n'` or the character value of 10 decimal.

6.3.4.1 HTTP Request/Response Headers

<u>Request Header</u> (from rTMA once at connect)

```
GET URL HTTP/1.1<cr><lf>
User-Agent: TMA/version<cr><lf>
Connection: keep-alive<cr><lf>
<cr><lf>
```

<u>Response Header</u> (from SDSS once)

```
HTTP/1.1 200 OK<cr><lf>
Server: SMS/version<cr><lf>
Content-Type: text<cr><lf>
Transfer-Encoding: chunked<cr><lf>
<cr><lf>
```

<u>Application Data</u> (continuous)

```
<content-length-hex><cr><lf>
<application-xml-data>
<cr><lf>
<content-length-hex><cf><lf>
<application-xml-data>
<cr><lf>
```

<u>Error Response Header</u> (from SDSS once)

```
HTTP/1.1 400 BAD REQUEST<cr><lf>
<cr><lf>
```

The `<content-length-hex>` field is a four character, hexadecimal number indicating the number of bytes to follow. This length does not include the *<cr><lf>* field following the `<content-length-hex>` field nor the *<cr><lf>* fields following the `<application-xml-data>` field. This limits the application data transfer to 65535 bytes. The URL will be used to indicate the airport managed by SDSS.

6.3.4.2 HTTP Request/Response Examples

Example Request Header (once)

```
GET /ZFW HTTP/1.1<cr><lf>
User-Agent: TMA/3.10.0<cr><lf>
Connection: Keep-Alive<cr><lf>
<cr><lf>
```

Example Response Header (once)

```
HTTP/1.1 200 OK<cr><lf>
Server: SMS/8.2<cr><lf>
Content-Type: text<cr><lf>
Transfer-Encoding: chunked
<cr><lf>
```

Example Application Data (continuous)

```
01F3<cr><lf>
<env ...><lf>
  <sms ...><lf>
    <air><lf>
     ...
    </air><lf>
  </sms><lf>
</env><lf>
<cr><lf>
045C<cr><lf>
...
<cr><lf>
```

Explicit Example

```
0033<cr><lf>
<env><lf>
  <sms><lf>
    <air><lf>
    </air><lf>
  </sms><lf>
</env><lf>
<cr><lf>
```

Note that the decimal byte count value of 51 (hex 33) includes twelve spaces used for indentation of the XML plus six `<lf>` characters.

6.3.5 Protocol Stack Implementation

The functional characteristics are implemented as shown in Figure 6:2 with respect to the seven-layer Open Systems Interconnection (OSI) reference model.

Application Layer - The Application Layer encodes application data using the W3C Extensible Markup Language (XML).

Presentation Layer - The Presentation Layer is not used.

Session Layer - The Session Layer uses RFC 2616, Hypertext Transport Protocol (HTTP).

Transport Layer - The Transport Layer uses RFC 793, Transmission Control Protocol (TCP).

Network Layer - The Network Layer uses RFC 791, Internet Protocol (IP).

Data Link Layer - The Data Link Layer uses IEEE 802.3.

Physical Layer - The Physical Layer uses Category 5 cabling.

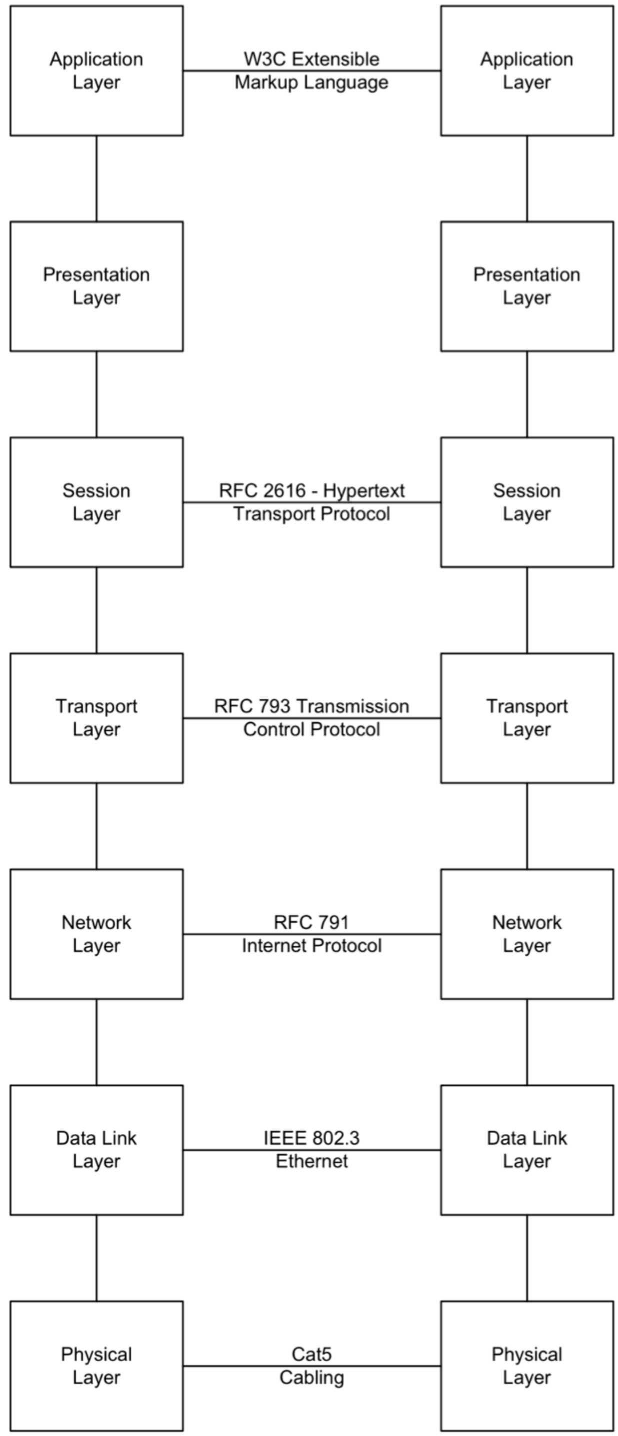

Figure 6:2 – OSI protocol mapping for the SMS to rTMA interface.

6.4 Call For Release Two-Way Status Indication

During the PDRC Block 1 operational evaluation, user feedback identified the need for TMCs to signal their readiness to use PDRC for Call For Release scheduling operations. In response, a small status panel comprised of click-activated buttons was added to the Center and Tower displays. The buttons serve as status indicators and can be clicked to toggle between states as shown in Figure 6:3.

The status panels were hosted on various displays (Center, East Tower, and West Tower) and were synchronized by sharing a common activity log. When a button was clicked to toggle status, the activity was written to a file. All status panels monitored this file to keep their indicators in sync. This approach simplified the ability of extending status panels to other displays if needed and served as a log to record durations of PDRC support.

<u>Example Log Activity</u>

```
[20130401 10:05:01] atct => red, artcc => red, committedBy =>
   run_PDRC_button_panels.sh
[20130401 11:23:29] atct => red, artcc => green, committedBy => artcc
[20130401 11:32:43] atct => green, artcc => green, committedBy => etwr
[20130401 13:25:24] atct => green, artcc => red, committedBy => artcc
[20130401 15:16:45] atct => red, artcc => red, committedBy => etwr
```

It can be seen from the example of logged activity that the Center first signaled their readiness to use PDRC at 11:23:29 UTC, followed by the East Tower approximately 9 minutes later. The Center and East Tower signaled that they were jointly supporting PDRC from 11:32:43 UTC until 13:25:24 UTC. Although this example identifies the East Tower specifically as toggling the ATCT button, the status indicator does not distinguish between East or West because their status cannot be independent of one another.

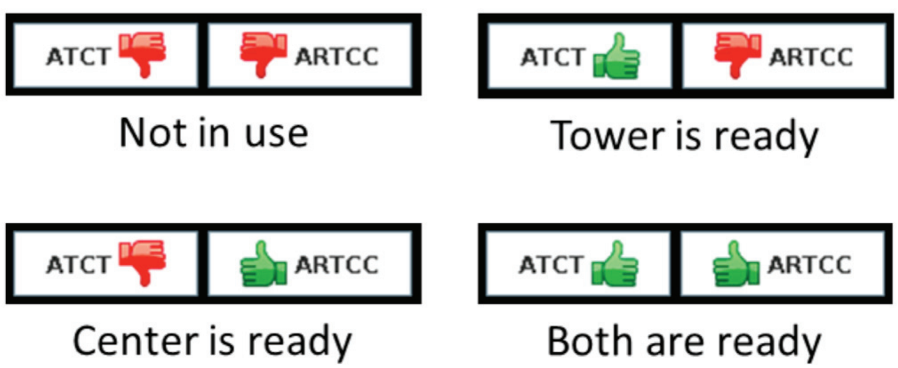

Figure 6:3 – PDRC status indicators facilitate communication between Tower and Center.

7 PDRC software development history

This section summarizes the PDRC software development history. One goal for this summary is to allow collaboration partners to trace PDRC software developments back to a common, familiar version of the software.

7.1 PDRC software releases

Since PDRC integrates the rTMA and SDSS prototype decision support tools, each version of the PDRC software consists of rTMA and SDSS versions plus the adaptation files and scripts required to successfully execute the software. This section provides a timeline of PDRC releases along with the rTMA and SDSS version numbers applicable to each release. Note that the rTMA version identifiers may be a little confusing as they contain the "PDRC" ClearCase branch label in addition to the numeric version identifier.

PDRC version 2.0.1 released for NTX use on 25 Feb 2012
rTMA version PDRC_3.0.1
SDSS version 9.1.2.N1.Jenkins14

PDRC version 2.0.2 released for NTX use on 9 Mar 2012
rTMA version PDRC_3.0.1
SDSS version 9.1.2.N1.Jenkins16

PDRC version 2.0.3 released for NTX use on 19 Mar 2012
rTMA version PDRC_3.0.2
SDSS version 9.1.2.N1.Jenkins17

PDRC version 2.0.4 released for NTX use on 5 Apr 2012
rTMA version PDRC_3.0.3
SDSS version 9.1.2.N1.Jenkins18

PDRC version 2.0.5 released for NTX use on 13 Apr 2012
rTMA version PDRC_3.0.3
SDSS version 9.1.2.N1.Jenkins20

PDRC version 3.1 released for NTX use on 23 Oct 2012
rTMA version PDRC_3.0.4
SDSS version 9.2.1.N1.10032018

PDRC version 4.0.0.1 released for NTX use on 15 Mar 2013
This release introduced a new configuration management mechanism. The "NTX Assembler" is a configuration controlled collection of all (non-rTMA) components required to run PDRC. Each of the components may have its own version number as shown below.
rTMA version PDRC_3.0.4
NtxAssembler-4.0.0.1-bin-03152013

- SDSS 9.2.2.N1
- FOSA 9.5.2
- TMA 2.6.0
- AodbReader 2.6.0
- Commander 1.2.1
- ActiveMqRepeater 1.3.1

- ActiveMq 5.4.2-fuse-02-00

7.2 SDSS change history through Block 2 field evaluation

This section describes changes to SDSS current through the Block 2 operational evaluation. The common software reference point for collaboration partners is SDSS v9.2.1. These changes are discussed at a high level and the reader is encouraged to reference specific JIRA reports where available for more details.

7.2.1 PDRC v2.0.1 / SDSS v9.1.2.N1.Jenkins14

- IADS-1. Upgrade PDRC to 9.x
  - This task upgraded the PDRC system to version 9.1.2, which is the latest released version of SDSS in use at other facilities. This upgrade incorporate substantial changes including the use of the new model, new FOSA interface and over a year's worth of bug fixes and enhancements from other SDSS projects.
- IAD-3. Regression test 9.X upgrade of PDRC
  - This task went hand-in-hand with IADS-1 and led to a number of specific problem reports being entered and completed. The main purpose of this testing was to ensure significant PDRC capability was incorporated and working properly.
- IADS-5. Default gate decision tree re-ordering
  - Modified the way matching works in the default_gate_decision_tree.xml to allow for a better default gate location based upon matching by airline before matching by ramp area.
- IADS-6. SDSS to rTMA messages not flowing
  - Encompassed work required to get primary PDRC messages added to the new FOSA interface so that rTMA⇔ SDSS message interchange could work properly.
- IADS-7 and IADS-57. AODB not working properly. Upgrade AODB reader to work with FOSA 9.1
  - Encompassed work required to get AODB working with latest FOSA interface.
- IADS-8. CM UI monitoring
  - Work required to ensure the CMUI worked with new model
- IADS-11. New startup procedure
  - Work required to incorporate latest SDSS and rTMA startup
- IADS-12. Pushback buffer not working
  - The work for this was done under STBO. This incorporated the pushback buffer logic that was used in the old model into the new model. This buffer also allows the user to specify the amount of time expected for the aircraft to pushback after receiving the OUT message from the carrier.
- IADS-13. Gate assignments not showing up in Flights table
  - Work required to show default gates in the Client Flights Table
- IADS-14. EDCTs not properly handled
  - This work was done under STBO. This incorporated the EDCT handling logic from the old model into the new model.
- IADS-15. Model failure

    - This problem report is from the June 2011 evaluation. After running the 9.1.2 model for a significant number of hours this could not be repeated, so this issue was closed.
- IADS-16. Miss-association resulting in two aircraft displayed on Client.
    - This problem report is from the June 2011 evaluation. After running the 9.1.2 system for a significant number of hours this could not be repeated, so this issue was closed.
- IADS-17. Mis-association ARRIVAL/DEPARTURE mis-match
    - This problem report is from the June 2011 evaluation. After running the 9.1.2 system for a significant number of hours this could not be repeated, so this issue was closed.
- IADS-19. SMS sending start of roll instead of OFF
    - Work required to ensure the wheels OFF time is being sent to SDSS instead of the start of roll.
- IADS-21. Client can't load SCM files
    - This problem report is from the June 2011 evaluation. Analysis revealed that this problem only occurs on highly CPU loaded machines. Given that workarounds exist this issue was closed.
- IADS-22 and IADS-40. Need ability to tell Client which .scf file to load
    - This capability simplifies the maintenance of the daily use system, which previously had to have several copies of the sdss directory in order to support all user display needs. Now the client can be passed a parameter that tells it which display file to load.
- IADS-28. TMAConnect sending beacon code as a decimal instead of octal
    - Work required to fix the sending of beacon codes as octal in new FOSA
- IADS-29. Allow scheduled times from AODB airlines to be used to update SDSS out
    - This capability allows the estimated pushback updates passed along by AODB or Flightstats to be used by SDSS to update its scheduled pushback time.
- IADS-30. Add ability for Client to display departure aircraft at gates
    - This capability allows for the optional display of departure aircraft that are still at the airport gates. This is intended to assist in testing of the data feeds prior to surface track and can also be used in PDRC request for release time processing.
- IADS-40. Incorporate CHI feedback on SDSS Client from July 2011 eval (low hanging fruit only)
    - This incorporates color changes, FDB display and map display options geared toward simplifying the user CHI based upon feedback from the July 2011 evaluation.
- IADS-46 and IADS-51. Create 9.1.2.N1 PDRC branch.
    - A development branch was created for configuration control of the SDSS system that will be used for PDRC. This branch is from the release of 9.1.2 and will be incremented as needed during Spring PDRC evaluation.
- IADS-50. Flights jumps backwards on the SDSS display
    - Changes required for the proper transition of aircraft from TMA TRACON surveillance to SDSS surveillance. This incorporated SDSS and rTMA changes.
- IADS-56. Look into using common SDSS log cleanup scripts – in use at MEM, MCO and UPS

    - Work to analyze the log cleanup scripts used at other SDSS and incorporate these (in principle, not letter of the law) at NTX.
- IADS-58, IADS-63 and IADS-64. Parking Gate Information not correct
    - Work required to add additional flight matching debug to FOSA and identify American Airlines data feed related issue. This resulted in a new/updated FOSA API being supplied to American.
- IADS-59. DFW Default Parking Gate
    - Analysis and SDSS adaptation changes geared toward better predictions of default parking gate. This included using destination to determine international parking gate D, as well as changes to default American and American Eagle to a gate closer to the middle of the airport to prevent short taxi time estimates.
- IADS-74. SDSS sending multiple deletes to rTMA
    - Work required to fix issue with latest SDSS system which did not properly register/store the delete message internally.
- IADS-75. TMA Connect not sending correct runway to TMA
    - SDSS' TMA Connect process was not properly handling the amendment which resulted in the runway not being sent to TMA.
- IADS-79. Make APREQ window configurable
    - The -/+ is now adaptable in the tma-cm-adapter properties file.
- IADS-84. New model not getting to the "In" status
    - Changes required to the new model to allow the last leg of an arrival flight to get to the "In" status. These were getting stuck in the "In Ramp" status, which was a contributing factor to aircraft later being displayed at their last airborne surveillance location- and also increased the probability that a mis-association could occur.
- IADS-85. Ignore Airborne Tracks for arrival aircraft after they have landed
    - Changes to the SDSS CM's position handling logic that ignores any airborne surveillance after the aircraft had landed ("On" status). This logic was a contributing factor to aircraft being displayed at their last airborne surveillance and potentially increased the probability of a mis-association.
- IADS-81. Departure Fix Mismatch
    - During the PDRC walkthrough operational personnel commented on the departure fix assignments being incorrect. Based upon analysis it was confirmed that the TRACON Departure Fix field in SDSS does not always match what is expected based upon the Route of Flight field. These cases are specifically for aircraft that are not yet airborne and either in a state of Scheduled_Out or Taxiing_AMA and that have filed a flight plan containing an RNAV departure. These RNAV procedures specifically contain the expected departure fix.

7.2.2 PDRC v2.0.2 / SDSS v9.1.2.N1.Jenkins16

- IADS-86. Need ability to distinguish between active/inactive on timeline with FDBs color coded by departure fix
    - The SDSS Client threshold timeline is now configured based upon departure fix color. The purpose of configuring the client this way is to allow for future East/West balance considerations. Given color is being used to see the departure fix, color cannot simultaneously be used to show active/inactive status. This fix

allows an adaptable inactive/active color to be used to shown on a flights leader line which corresponds to the time the aircraft receives surface surveillance.

- Updated to SDSS AdaptationLib 9.0.2
- Updates to SDSS file menu to remove unused actions and pull forward frequently used actions.

7.2.3 PDRC v2.0.3 / SDSS v9.1.2.N1.Jenkins17

- Software change under STBO-2423, Optional ignore spot decision tree after pushback. This was required for spot prediction improvements
- Turned on 'health' debugging to further investigate periodic SDSS CM 'Not connected' messages
- Updated params to show active/inactive leader line color by default

7.2.4 PDRC v2.0.4 / SDSS v9.1.2.N1.Jenkins18

- IADS25, IADS32, IADS35, IADS68 and IADS87 were all resolved in this release, which collectively provide the following capability:
    - Display the APREQ time in seconds level precision
    - Apply the adaptable time window to the APREQ value using seconds level precision from the APREQ window (e.g., APREQ window to seconds +/- one minute)
    - Provide an indication that a request for release time has been sent to En Route system (the words "REQUESTED" are now placed next to aircraft after action and before response)Note: We are asking Surface to forego a call but give no feedback that the request has been sent. Hopefully this change is welcome.
    - Adjust the APREQ time by subtracting the predicted takeoff roll time from the En Route coordinated wheels OFF time Note: This gives surface personnel the time at which the aircraft should be released for takeoff. Currently TMA scheduled time minus 35 seconds for all aircraft.
    - Display the TMA scheduled time in the Flights panel (without roll time subtracted as this represent OFF) Note: primarily for testing, but might become more valuable when we move to type specific roll times.
    - Allow for EDCTs and APREQs to be displayed on one line (previously these both forced you to use two lines which didn't work with East tower view)
    - When an APREQ aircraft is cancelled on surface system, return it to its default state so that it can be requested again (without bars and EXP or HOLDs)
    - When an APREQ is unscheduled or reset from TMA, it should be unscheduled from surface. Note: this was already happening to a certain degree but the aircraft was not returning to its default state similar to cancel
    - When an APREQ is rejected from the surface system, keep the aircraft in such a state that another scheduled time from TMA will allow the user to 'accept'. This

may need more thought, but this keeps the aircraft 'in play' without the need for multiple clicks

  - Change APREQ late color from Red to Orange
  - Highlight APREQ and EDCT aircraft as white on the SDSS timeline
  - Update the DFW TRACON maps to match those requested by DFW STMCs

7.2.5 PDRC v2.0.5 / SDSS v9.1.2.N1.Jenkins20

- IADS-96. Make apreq roll time and apreq buffer adaptable
  - The ATCT TMCs are currently subtracting 90 seconds from the coordinated time they get from TMA. An adaptable buffer has been added to account for this practice. The APREQ time will be displayed to the seconds and now uses the formula APREQ time = TMA Scheduled Departure Time – (adaptable) Takeoff Roll + (adaptable) APREQ buffer. The takeoff roll and buffer are set to zero by default and can be adapted in config/tma-cm-adapter/tma-cm-adapter.properties file.
- IADS-97. Automatic display of aircraft at gates when user zooms in to a certain level
  - This solution is a compromise between too much clutter on the surface display versus lack of gate assignment information. This capability addresses ATCT request for dynamic display of gate information on the surface map. It must be enabled in params by setting display_departures_at_gates_zoom to true. The zoom level was selected by trial and error and may need some adjustment based upon user feedback. To see aircraft at gates you need to zoom in 4 times from the default zoom level (reset zoom).

7.2.6 PDRC v3.1 / 9.2.1.N1.10032018

- IADS-24. Merge changes for flight dump to SDSS trunk
  - Allows for TmaCmAdapter to dump flight information to CM log when the log rolls over.
- IADS-47. Broadcast FOSA to multiple SDSS instances
  - The 'daisy chaining' of TSDE to multiple instances is necessary for integration testing as well as distribution of limited data streams. Use the ActiveMQ repeater to forward messages to other downstream instances of ActiveMQ.
- IADS-71. Adapt the pushback buffer at DFW based upon assumption of american pushback times
  - Current pushback buffer options can be specified by carrier, and it seems likely that this is needed. The idea is to do the best we can with the current available options and identify any new capability that might be needed to improve accuracy to the SPOT.
- IADS-77. Client intermittently loses flights and displays "Not Connected to CM"
  - When running 9.1.2 at DFW we were seeing an intermittent problem with the Client losing flights and displaying the "Not Connected to CM". This corrected that issue.

- IADS-90. Create a ntx_prod assembler/install that can be used for the production environment at NTX
    - There was one install assembly for NTX, ntx_dev. Custom scripts were used to copy over top of the ntx_dev install for each release. This new assembly allows for configuration control and access to the common production settings at NTX. There are 3 new options for the NtxAssembler. You can specify deployment, machine, and component. Deployment options are prod (production), dev (development) and driver. Machine options are the NTX machine names (vma3, eagle, etc). Components are sdss, tma, commander, activemq, aodb, or all.
- IADS-100. Remove certain characters from route string in Tma Connect
    - Handle "*" and "+" in route strings.
- IADS-118. Q routes handled by APREQ
    - Added additional routes to the jet_routes.xml file. There were two existing J routes. Two additional J routes added (J131, J52) along with 3 Q routes (Q100, Q102, Q105).
- IADS-119. Add HOU to APREQ airport list
    - HOU added to the pacing_airports file which now allows it to show up in the ApReq panel.
- IADS-120. Update IATA code mapping to increase American TSDE matching
    - Analysis showed that additional airports needed to be added to the IATA mapping file to allow for better matching with American Airlines TSDE feeds.
- IADS-123. Investigate low ramp taxi time issues
    - Initial indications were that the ramp taxi time SDSS was predicting prior to OFF were too low (e.g., on the order of 20 seconds). After analysis, new speeds were added to the departure_taxi_speed_decision_tree.
- IADS-131. Need way to modify ramp taxi speed by aircraft type (and possibly carrier). Taxi decision tree?
    - Modeled taxi speed in the ramp area was higher than actual. A new adaptation schema was created for departure taxi speed decision tree categories. The categories include aircraft type, carrier, and gate values.
- IADS-138. Flight state decision tree criterion
    - The decision trees need a new criterion type that allows filtering based on the flight state and this added it to the code.
- IADS-140. Update taxi speed categories file
    - The categories file needed to include entries for the ramp speed and ama speed and this ticket added it to the code.
- IADS-141. Update trajectory modeling to account for varying speed
    - The trajectory code assumed constant taxi speed. The logic was updated to use the new ramp and ama speeds in the decision tree categories. Taxi time logic

modified to apply either the ramp taxi speed or ama taxi speed based on waypoint type.

- IADS-143. Add arrival/departure airport to TmaMeteredFlight object
  - Added arrival/departure airports to the TMAMeteredFlight object.
- IADS-147. Move meter point service to TmaCmAdapter
  - Several changes made to move the meter point service from TMAConnect into the TmaCmAdapter. The adapter project now hosts the service to which the model will get the meter point data. The MeteredData object was made part of the SDSS BaseFlight class so any new fields should be immediately available to SDSS without further changes to the SDSS code.
- IADS-148, IADS-150, IADS-151, IADS-152, IADS-153, and IADS-158. Enhanced meter point scheduling
  - This incorporated several updates to the PDRC SDSS. These included the ability to display in the flights table meter data times in seconds, MIT for each SCN, and the leading/trailing flight keys for the meter window.
  - The TMACMAdapter processing logic was updated to assume/use a MIT restriction of 5 if no other MIT value has been received.
  - Update the logic so that timeline datablock only displays metering information if the flight is an ApReq.
  - Timeline logic updated to always show the meter delay when the field is enabled. If no meter delay data is available then '0' is shown.
  - Suspended aircraft logic updated so that suspended aircraft are not immediately deleted since doing so negatively affect the scheduling of downstream flights. The logic now bundle delete messges into 1 minute groups and then deletes those aircraft approximately 60 minutes after receiving the deletion message.
- IADS-156. Allow ActiveMqRepeater to capture repeated messages
  - The ActiveMqRepeater now saves the messages it has repeated to a log file.
- IADS-157. APREQ buffer
  - A bias buffer has been added that provides the ability to add an adaptable number of seconds to the actual OFF time estimate prior to passing it along to TMA. This value is then removed when the time comes back from TMA.
- IADS-161. Meter window times are showing really big invalid numbers
  - Corrected a divide by zero.
- IADS-162. Update metering window to always display
  - Updated to always show the metering window for any flight that is an ApReq departure.
- IADS-163. Command line argument for displayfile
  - Corrected an unexpected change; the argument was added back in.
- IADS-164. TmaConnect: Remove unused fields from filtered flights
  - Clean up.

7.2.7 PDRC v4.0.0.1 / 9.2.2.N1

As the version number suggest, v4.0.0.1 was a major release for PDRC. SDSS v9.2.2.N1 contains numerous new features to support a PDRC follow-on research activity known as TRACON Departure Scheduling. The following list of change tracking items covers only the changes relevant to PDRC Core and not those specific to TRACON Departure Scheduling. Many of the PDRC Core software changes in v9.2.2.N1 were implemented to expose PDRC enhanced scheduling information to the two-way air carrier (i.e., TSDE/FOSA) interface.

PDRC Core improvements

- IADS-189 Aggressive OFF scheduling
- IADS-223 Fix log4j configuration
- IADS-224 Runway closures, fix closures, etc. messages not following pattern and being recorded
- IADS-229 Setup Daisy Chaining configuration
- IADS-234 Properties file updates
- IADS-237 Make runway roll time configurable

TSDE/FOSA updates

- IADS-205 Add message flow to FosaBroker to support TMAScheduler data
- IADS-206 Implement transformation logic to convert TMAScheduler data into Fosa data
- IADS-214 Create FOSA object structure for TMA Scheduler metered data
- IADS-215 Create metered data handler in FOSA API
- IADS-216 Add MeteredData sync capability to FosaAPI
- IADS-225 Add ability for FOSAAPI to do a sync on a single flight

Enhanced scheduling updates

- IADS-192 TMA Scheduler not sending leading/trailing times

7.3 rTMA change history through Block 2 field evaluation

This section describes changes to rTMA current through the Block 2 field evaluation. The common software reference point for collaboration partners is TMA v3.12.0. These changes are discussed at a high level and the reader is encouraged to reference specific PR reports where available for more details.

The changes covered the following areas:

- Upleveling of the PDRC system to TBFM release 3.12
- Enhancements to provide new capabilities
- Fixes to problems in the PDRC rTMA system

7.3.1 PDRC v2.0.1 / rTMA vPDRC_3.0.1

- AFPRS00012947. Initial seeding of the rTMA baseline with FAA V3.12.0-P2
    - Record covered work for merging the V3.12.0-P2 labeled code in as the baseline for rTMA. The new baseline was created off the FAA TBFM TMA code base as

opposed to the older FAA TMA code base. This new baseline has a totally different directory structure from the older baseline. All elements were labeled with the "rTMA_BASELINE" label.

- AFPRS00012948. rTMA PDRC Linux Port up-level
  - Record covered work for merging the original PDRC Linux port and related work into the new rTMA baseline. Most functionality involved was originally implemented via the below records. Various other changes were made due to merging/re-implementation of code. The original PDRC code was developed off of FAA TMA V3.10.0. Code was labeled and released as "rTMA_B1.0.0". In addition to supporting PDRC development this code base allows for other NASA projects to make use of the FAA TBFM TMA baseline. Project specific work is performed on integration branches off the new rTMA baseline.
    - CSCnj14660 - PDRC Linux Port
    - CSCnj14740 - PDRC analysis tools changes
    - CSCnj15048 - Analysis tools enhancements
    - AFPRS12314 - Run rTMA under ARC Clearcase
- AFPRS00012949. rTMA PDRC General up-level
  - Record covered work for merging the original PDRC specific work into a PDRC integration branch off the new rTMA baseline. Most functionality involved was originally implemented via the below records. Various other changes were made due to merging/re-implementation of code.
    - CSCnj14661 - PDRC CAP Synchronization
    - CSCnj14662 - PDRC SMS Interface
    - CSCnj14740 - PDRC analysis tools changes
    - CSCnj15037 - PDRC playback and analysis
    - AFPRS12213 - Implement rTMA AFAT processing
    - AFPRS12314 - Run rTMA under ARC Clearcase
- AFPRS00013032. rTMA PDRC Linux Port up-level (TGUI)
  - Record covered work for merging the original TGUI PDRC Linux port work into the new rTMA baseline. Most functionality involved was originally implemented via the records identified below. Various other changes were made due to merging/re-implementation in the new baseline. Resolved the vertical sizing problems of the TGUI timelines. This record is a companion to AFPRS12948.
    - CSCwb08204 - PDRC Linux Port
    - CSCnj14660 - PDRC Linux Port
- AFPRS00013070. rTMA PDRC General Uplevel (TGUI)
  - Record covered work for merging the original PDRC TGUI specific work (CTSrv02169) into a PDRC integration branch off the new rTMA baseline. Various other changes were made due to merging/re-implementation in the new baseline. This record is a companion to AFPRS12949.
- AFPRS00013111. rTMA PDRC TRACON scratch pad uplevel
  - This record covered work for merging the original PDRC TRACON scratchpad entry modifications (AFPRS12518) into the new PDRC integration branch. In the original work, CM was modified to optionally allow the assignment of runways based on the TRACON scratch pad entries.
- AFPRS00013187. rTMA PDRC Vertical Sizing Problems On TGUI Dialogs

    - Provided fixes for various TGUI dialog issues which were not covered in original Linux port work. These included items such as vertical sizing issues for the Internal Departures and Graph Setup panels and various other text field sizing/display issues (problem with entering file names when selecting display, graph, and timeline setup files, data column line up on the Meter Point dialog, garbage text being shown on graph plot dialogs). Implemented only on PDRC integration branch.
- AFPRS00013224. rTMA PDRC TGUI SMS related problems fixed
    - Work covered various SMS related TGUI issues, namely, non-SMS aircraft having SMS data shown on Aircraft Data dialog, the list of scheduled aircraft on the SMS Internal Departures dialog flickering when updated, and the print button not working on the View Internal Departures Report dialog. Implemented on PDRC integration branch.
- AFPRS00013233. rTMA linux port ctas_free() memory leak
    - Record corrected a legacy Linux port introduced issue related to large-size chunks of memory not being de-allocated. The change had been introduced to prevent shared memory from being de-allocated. The change attempted to use sbrk() in deletion eligibility determination but this was problematic due to how GNU "glibc" malloc() allocates memory. Only weather file usage processes (i.e., WDPD, RA, TS) were noticed as being affected. Implemented in rTMA baseline.
- AFPRS00013246. rTMA PDRC TGUI adapt SMS indicator symbols
    - When running with SMS, TGUI timeline aircraft tags with an assigned PCOT always had a hardcoded "p" symbol displayed next to them. It was requested by the field that the symbols not be shown during operational evaluations. Therefore, processing was added to handle configuring both the PCOT character value and display via adaptation. Additionally, the configuring of EDCT character display was added as well as displaying variations of either PCOT or EDCT, or both characters. Implemented on PDRC integration branch.
- AFPRS00013250. rTMA weather related issues
    - Record corrected a raw weather file copy which had made the live weather copy script un-usable, made another live weather script change which allowed for weather to become old when rendezvous file wasn't updated, modified CM to properly extract the weather filename date/time values so the correct date/time values would appear in the PGUI weather panel, and modified WDPD so that it may be run from a directory pathname which contains a '-'. The WDPD item was a legacy TMA issue. Implemented in rTMA baseline.
- AFPRS00013282. rTMA send CAP all AFAT tracks
    - Cases had been seen in SDSS (SMS) where temporary surface surveillance dropouts and other anomalies caused track usage to revert back to the CAP received TRACON tracks. This switch caused aircraft position jumps on the SDSS GUI. Changes in this record defaulted CM track processing to not send AFAT tracks to CAP for flights whose data was not ready (e.g., flight tagged as landed). Also, a command line option was added to override the default behavior. Implemented in rTMA baseline.
- AFPRS00013315. rTMA CAP recording suppression

- Provided developer command line options for suppressing creation of the xml, message, and debug recording files. This work was originally requested since the CAP AFAT xml files grow quite large and most systems do not need to record. Record also corrected a legacy TMA issue which disallowed the user from modifying the testpoints recording configuration via the command line. Implemented in rTMA baseline.

7.3.2 PDRC v2.0.2 / rTMA vPDRC_3.0.1

No rTMA changes from PDRC v2.0.1.

7.3.3 PDRC v2.0.3 / rTMA vPDRC_3.0.2

- AFPRS00013285. rTMA truncated TGUI dialog titles
- AFPRS00013311. rTMA NASA/Ames dialogs crash TGUI
- AFPRS00013345. rTMA TGUI departure configuration inconsistencies
- AFPRS00013368. PDRC Walkthrough SMS Mods for TGUI
- AFPRS00013374. rTMA AFAT ISM sends initial amd instead of add
- AFPRS00013433. rTMA CAP reset aircraft std clear fix
- AFPRS00013434. Merge missed CAP recording suppression file to iads_integ
- Upgraded adaptations for ZFW (ZFW_T3.12.0_7.1_P1.0) and its adjacent facilities, including back merged pref file changes from ZFW operational users

7.3.4 PDRC v2.0.4 / rTMA vPDRC_3.0.3

- AFPRS00013345. rTMA TGUI departure configuration inconsistencies (additional fix)
- AFPRS00013474. PDRC PCOT indicator sometimes erroneously displayed
- AFPRS00013478. PDRC TGUI SMS status name changes
- Adaptation releases ZFW_T3.12.0_7.1_P1.2. Updates to local and adjacent center adaptations

7.3.5 PDRC v2.0.5 / rTMA vPDRC_3.0.3

No rTMA changes from PDRC v2.0.4.

7.3.6 PDRC v3.1 / rTMA vPDRC_3.0.4

- AFPRS00013974. PDRC TGUI SADD disappears before schedule completion

7.3.7 PDRC v4.0.0.1 / rTMA vPDRC_3.0.4

No rTMA changes from PDRC v3.1

8 Build Process

8.1 SDSS

To build the SDSS system at NTX, complete the following steps:

1) cd to /casa/pdrc/sdss_src/Ntx
2) ./mvn_install

This will pull in all the latest source from the local machine (which is obtained through the external repository), determine and resolve the dependencies (by downloading any required packages from Nexus), and produce a single zipped package containing SDSS, AODB reader, TMA Connect and Commander.

8.2 rTMA

It is suggested that directories be configured as follows in order to support the below build and configuration procedures.

rtma
- adaptation
- software
- weather

8.2.1 Build procedure

The following procedure may be used to build an rTMA system. Examples are ksh commands.

- Create a top-level source directory (e.g., rtma/software/pdrc_3.0.1) and unpackage source in it.
- Set CTAS_ROOT environment variable to point to top-level directory and change into that directory. Note that relative pathnames listed throughout are relative to this directory.
  - export CTAS_ROOT=/<path>/pdrc_3.0.1
  - cd $CTAS_ROOT
- Use the ./build_system script to build software. Note that a makefile script overwrites the version id in the ctas_version_id.h file. Need to make sure to write access the file or if a rebuild must change version id back to "STUB_VERSION". Also note that in the build version no dashes should be used as added via makefile scripting (e.g., PDRC_3.0.1-P1 should be specified as PDRC_3.0.1P1).
  - mv ./common/api/ctas_version_id.h ./common/api/ctas_version_id.h.org (only if initial build)
  - cp ./common/api/ctas_version_id.h.org ./common/api/ctas_version_id.h
  - chmod 666 ./common/api/ctas_version_id.h
  - export TMA_VER=PDRC_3.0.1
  - export TMA_MAK=gmake
  - ./build_system clean (only if rebuild)
  - script BUILD_SCRIPT
  - ./build_system all

8.2.2 Initial Weather configuration

The following procedures may be used to set up weather-related directories. Additional configuration will be handled as part of the script run in the Initial System configuration section.

- Create a directory to house the weather and support files (e.g., rtma/weather). Create sub-directories as needed per below information.
  nws_pbk
    20110101/
    20110102/
    ...
    where user stores default NOAA files for playback
  raw_pbk
    20110101/
    20110102/
    ...
    where user stores default raw files for playback
  nws/
    where playback NOAA files are copied for use by this script
  raw/
    where all below facility raw files are moved for WDPD use
  ZFW/
    where grib_read created ZFW raw files are stored
  ZHU/
    where grib_read created ZHU raw files are stored
  ...
- Have the system administrator modify system weather script so that the NOAA weather file and rendezvous file are copied into the above directory area. The actual locations are detailed in the ./tools/misc/src/wthr_nws2raw_live script.
- Obtain copies of grib_read and the National.config file. Make sure only desired facilities are uncommented in National.config. Note that the grib_read executable and the National config file are not maintained in TBFM V3.12.0. If unable to obtain the user must use rTMA_1.0.0 vob label, rebuild system via buildctas, and obtain from /vobs/ctas/realtime_procs/ctas_remote_wthr_srvc/ grib_read/grib_read and remote_tc/National.config
- Copy in weather scripts from ./tools/misc/src (wthr_nws2raw_live, wthr_nws2raw_pbk, wthr_raw_pbk, and wthr_cleanup) and adjust weather base path, etc. as necessary.
- The weather is scp'ed to any boxes running either PGUI or RA (scp'ed even if on same box as WDPD). Need to have system set up to allow scp without password otherwise will have to enter password at least once per hour when weather syncs occur.
- Start a cron job (crontab –e) that executes the weather script at some cyclic minutes past each hour in order to handle rendezvous file updates near to when received. Make sure the weather script has permissions of 755 so cron can execute. After starting cron job manually run script once so that the initial weather data is populated (assumes system weather script has already provided files).
  - add: */10 * * * * <script location>/wthr_nws2raw_live

8.2.3 Initial System configuration

The following assumes that an adaptation directory has been created (e.g., rtma/adaptation) and pertinent adaptations installed in it.

- Use the configure_system script. Note that rTMA looks for adaptation in the ./apps/adaptation directory. This directory must contain the physical adaptation directories or links to the directories containing the adaptation. It is preferable to maintain the adaptation separate from the build and provides links. The configure_system script uses the configure_system_adp script to set up the links. Note that for running the ZFW system only preference file storage is needed for the D10, D11, and EDC tracon groups.
    - ./tools/misc/src/configure_system

9 Running the System

9.1 PDRC

To support running the PDRC system in a production manner, the start scripts from the prior field evaluation were tailored. This tailoring allows for the use of the SDSS ‘commander’ to start up the SDSS system given this is the standard way the system is initialized at other facilities.
To start the system, simply cd to the appropriate machine specific directory and type
`./run_PDRC_ZFW.sh`

More information can be found in the PDRC Development System Configuration notice that was sent to the ntx_pdrc email distribution. The procedures will be refined as needed during the evaluation.

9.2 SDSS

See Section 9.1

9.3 rTMA

Both the rTMA and SDSS systems are run via a common startup script. For rTMA this script manages loading adaptation into shared memory via SAM and running the system via EMU and its associated config file. Note that the script assumes that there is an “rtma” link in the same directory as the script that points to the rTMA version directory being used.

The **`$CTAS_ROOT/tools/misc/src/configure_system_adp`** and **`$CTAS_ROOT/tools/misc/src/load_system_adp`** scripts may be run again if the need arises for changing what adaptation the system is currently being run with.

The **`$CTAS_ROOT/tools/misc/src/pref_install`** script may be run again if the need arises for changing what facility the rTMA system is being run for.

Reference documents

This section provides a list of references, applicable documents, and related research for the PDRC-IADS research activity. To facilitate document tracking, identical reference lists are being maintained across the PDRC-IADS document family [1, 2, 3]. Consequently, this list should be treated as a bibliography of relevant documents instead of a strict list of references. Documents are listed by category rather than in order of citation, and some documents in the bibliography may not be directly referenced in this document.

PDRC documents:

1. Engelland, S.A., Capps, A., and Day, K., "Precision Departure Release Capability (PDRC) Concept of Operations," NASA/TM-2013-216534, June 2013.
2. Engelland, S.A., Capps, A., Day, K., Robinson, C., and Null, J.R., "Precision Departure Release Capability (PDRC) Technology Description," NASA/TM-2013-216531, June 2013.
3. Engelland, S.A., Capps, A., Day, K., Kistler, M., Gaither, F., and Juro, G., "Precision Departure Release Capability (PDRC) Final Report," NASA/TM-2013-216533, June 2013.
4. Engelland, S.A. and Capps, A., "Trajectory-Based Takeoff Time Predictions Applied to Tactical Departure Scheduling: Concept Description, System Design, and Initial Observations," AIAA-2011-6875, *11th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Virginia Beach, VA, September 20–22, 2011.
5. Capps, A. and Engelland, S.A., "Characterization of Tactical Departure Scheduling in the National Airspace System," AIAA-2011-6835, *11th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Virginia Beach, VA, September 20–22, 2011.
6. Capps, A., Day, K., Walenciak, E., and Engelland, S.A., "Impact of Departure Prediction Uncertainty on Tactical Departure Scheduling System Performance," AIAA-2012-5674, *12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Indianapolis, IN, September 17–19, 2012.

Research and technical papers:

7. Atkins, S., Jung, Y., Brinton, C., Stell, L., and Rogowski, S., "Surface Management System Field Trial Results," AIAA 2004-6241, *4th AIAA Aviation Technology, Integration, and Operations (ATIO) Forum*, Chicago, Illinois, September 20–22, 2004.
8. Brinton, C., Lent, S., and Provan, C., "Field Test Results of Collaborative Departure Queue Management," *29th Digital Avionics Systems Conference*, Salt Lake City, Utah, October 3–7, 2010.
9. Cook, L., Atkins, S., and Jung Y., "Improved Prediction of Gate Departure Times Using Pre-Departure Events," AIAA 2008-8919, *26th Congress of International Council of the Aeronautical Sciences (ICAS) including the 8th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Anchorage, AK, September 14–19, 2008.
10. Coppenbarger, R., Dyer, G., Hayashi, M., Lanier, R., Stell, L., Sweet, D., "Development and Testing of Automation for Efficient Arrivals in Constrained Airspace," *27th*

International Congress of the Aeronautical Sciences (ICAS), Nice, France, September 19–24, 2010.

11. de Jonge, H., Tuinstra, E., and Seljée, R., "Outbound Punctuality Sequencing by Collaborative Departure Planning," *6th USA/Europe ATM 2005 Seminar*, Baltimore, Maryland, June 2005.
12. de Jonge, H., Tuinstra, E., and Seljée, R., "Outbound Punctuality Sequencing by Collaborative Planning," NLR-TP-2005-013, National Aerospace Laboratory of the Netherlands (NLR), July 2005.
13. Doble, N., Timmerman, J., Carniol, T., Klopfenstein, M., Tanino, M., and Sud, V., "Linking Traffic Management to the Airport Surface: Departure Flow Management and Beyond," *Eighth USA/Europe Air Traffic Management Research and Development Seminar (ATM2009)*, Napa, CA, June 29 – July 2, 2009.
14. Farley, T. C., Landry, S. J., Hoang, T., Nickelson, M., Levin, K. M., Rowe, D., and Welch, J. D., "Multi-Center Traffic Management Advisor: Operational Test Results," AIAA-2005-7300, *5th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Arlington, VA, September 26–28, 2005.
15. Futato, S., McMillan, K., Callon, S., "TMA En Route Departure Capability Jacksonville ARTCC and Orlando ATCT Usage Data", April 8, 2008.
16. Grabbe, S., "Traffic Management Advisor Flow Programs: an Atlanta Case Study", *AIAA Guidance, Navigation, and Control Conference*, Portland, Oregon, August 8–11, 2011.
17. Hoang, T., Jung, Y., Holbrook, J., and Malik, W., "Tower Controllers' Assessment of the Spot and Runway Departure Advisor (SARDA) Concept," *9th USA/Europe ATM R&D Seminar (ATM2011)*, Berlin, Germany, June 14–17, 2011.
18. Jung, Y. C., and Isaacson, D. R., "Design Concept and Development Plan of the Expedite Departure Path (EDP)," *AIAA Aircraft Technology, Integration, and Operations (ATIO) Conference*, Los Angeles, CA, October 1–3, 2002.
19. Jung, Y., and Monroe, G., "Development of Surface Management System Integrated with CTAS Arrival Tool," AIAA 2005-7334, *5th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference,* Arlington, Virginia, September 26–28, 2005.
20. Kopardekar, P., Green, S., Brinkman, C., Thompson, P., Evans, M., and Davis, D., "Making Internal Departure Release Operations More Efficient," AIAA 2004-6346, *4th AIAA Aircraft Technology, Integration, and Operations (ATIO) Forum*, Chicago, Illinois, September 20–22, 2004.
21. Kopardekar, P., Green, S., and Thompson, P. "Improving Efficiency of Departure Release Communications for En Route Overhead Traffic Flow Management," *23rd Digital Avionics Systems Conference*, Salt Lake City, Utah, October 24–28, 2004.
22. Krozel, J., Rosman, D., and Grabbe, S., "Analysis Of En Route Sector Demand Error Sources", AIAA-2002-5016, *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Monterey, California, August 5–8, 2002.
23. Landry, S. J., and Villanueva, A., AIAA-2007-7713, "Mitigating the Effect of Demand Uncertainty Due to Departures in a National Time-Based Metering System". *7th AIAA Aviation Technology, Integration, and Operations (ATIO) Technical Forum*, Belfast, Northern Ireland, September 18–20, 2007.

24. Palopo, K., Lee, H, and Chatterji, G., "Benefit Assessment of the Precision Departure Release Capability Concept", *11th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Virginia Beach, VA., September 20–22, 2011.
25. Stein, B., Ceniccola, D., and Vincent, D., "TBFM Coupled Scheduling," *56th Annual ATCA Conference and Exposition*, National Harbor, Maryland, October 2–5, 2011.
26. Swenson, H. N., Hoang, T., Engelland, S., Vincent, D., Sanders, T., Sanford, B., and Heere, K., "Design and Operational Evaluation of the Traffic Management Advisor at the Ft. Worth Air Route Traffic Control Center," *First USA/Europe Air Traffic Management R&D Seminar*, Saclay, France, June 1997.
27. Thipphavong, J., and Landry, S. J., "Effects of the uncertainty of departures on multi-center traffic management advisor scheduling.", AIAA-2005-7301, *5th AIAA Aviation Technology, Integration, and Operations (ATIO) Technical Forum*, Arlington, VA., September 26–28, 2005.

Related NextGen initiatives:

28. MITRE CAASD MTR110240R1 A Concept for Integrated Arrival, Departure, and Surface (IADS) Operations for the Mid-Term, January 2012.
29. MITRE CAASD MTR100349R1 A Functional Analysis of Integrated Arrival, Departure, and Surface (IADS) Operations, March 2012.
30. MITRE CAASD MTR110156R1 Integrated Arrival, Departure, and Surface (IADS): Research Gaps and Recommendations, January 2012.
31. MITRE CAASD MP090230 Surface Trajectory-Based Operations (STBO) Mid-Term Concept of Operations Overview and Scenarios, September 2009.
32. MITRE CAASD MP090169 A Mid-Term Concept of Operations for a Tower Flight Data Manager (TFDM), September 2009.
33. FAA ATO System Operations Services, Operational Concept for Departure Flow Management, June 29, 2007.
34. FAA ATO Surface Operations Office, U.S. Airport Surface Collaborative Decision Making (CDM) Concept of Operations (ConOps) in the Near-Term – Application of Surface CDM at United States Airports, June 15, 2012.
35. FAA, "NextGen Mid-Term Concept of Operations for the National Airspace System," version 2.1, September 2010.
36. DFM System Requirements, Version 1.0, September 15, 2008.
37. FAA System Wide Information Management (SWIM) http://www.swim.gov/
38. JPDO, "Integrated Work Plan for the Next Generation Air Transportation System," version FY13, http://jpe.jpdo.gov/ee/request/home

Software documents, specifications and standards:

39. FAA NOTICE N JO 7110.612 amends FAA Order JO 7110.65 concerning Traffic Management Advisor (TMA) Departure Restrictions, Clearance Void Times, Hold for Release, and Release Times, January 30, 2013, http://www.faa.gov/documentLibrary/media/Notice/N7110.612.pdf
40. TMA/EDC System/Subsystem Design Description, September 2008.

41. Computer Sciences Corporation, “Traffic Management Advisor (TMA) Interface Control Document (ICD) for the TMA Collaborative Arrival Planning (CAP) Tool,” Contract No. DTFA03-02-D-00001, May 31, 2007.
42. FAA ICD NAS-IC-82392403-04, TMA CAP to Volpe's TMA Collector ICD.
43. VNTSC-TFM-07-01, ICD: Flow of TMA Data from Volpe to CDM Participants, V1.6 Draft A.
44. Raytheon, “CTO-05 Surface Management System, CTOD 5-24-1 Final Report,” 2004.
45. Louisville SMS User’s Guide, Version 7.0, October 16, 2006.
46. FAA IRD NAS-IR-XXXXX, NextGen Surface Prototype System (NSPS) Tactical Surface Data Exchange (TSDE) Interface Requirements Document (IRD), draft version 5.1, January 3, 2013.
47. FAA ConUse, Flight Operator Surface Application (FOSA) Concept of Use, version 1.0, November 30, 2010, draft.
48. FOSA API Overview, Version 1.0, undated.
49. Surface Operations Data Analysis and Adaptation (SODAA) User’s Guide, Version 2.10.0, April 30, 2013.
50. ISO 8601, Data Elements and Interchange Formats - Representation of Dates and Times, 2004.
51. RFC 2616, Hypertext Transport Protocol -- HTTP/1.1, June 1999.
52. IEEE-STD 1362-1998, Guide for Information Technology, System Definition – Concept of Operations, 1998.
53. FAA, National Airspace System (NAS) System Engineering Manual, Version 3.1, June 6, 2006.
54. NASA SP-2007-6105, Systems Engineering Handbook, Rev 1, December 2007.

Glossary

This section provides a list of acronyms and terms relevant to the PDRC-IADS research activity. Identical glossaries are being maintained across the PDRC-IADS document family [1, 2, 3].

ADIF	ARTS Data Interface
APREQ	Approval Request – see CFR
ARTCC	Air Route Traffic Control Center – one of twenty FAA facilities responsible for En Route ATC in the NAS
ATC	Air Traffic Control
ATCT	Air Traffic Control Tower
CAP	Collaborative Arrival Planning
CDIF	CAP Data Interface
CDQM	Collaborative Departure Queue Management
CFR	Call For Release – a TMI used to regulate the flow of departures into a constrained overhead stream (also known as APREQ).
ConOps	Concept of Operations
CRT	Coordinated Release Time – the target release time negotiated between Tower and Center during CFR operations (see SDT).
CTD	Concept and Technology Development (NASA Project)
DFM	Departure Flow Management
EDC	En Route Departure Capability
EDIF	ETMS Data Interface
ETMS	Enhanced Traffic Management System – see TFMS
FLM	Frontline Manager
FOSA	Flight Operator Surface Application – see TSDE
GUI	Graphical User Interface
HDIF	Host Data Interface
IADS	Integrated Arrival/Departure/Surface
NAS	National Airspace System
NextGen	The next generation air transportation system
OFF	Aircraft takeoff time
OTC	OFF Time Compliance
PCOT	Predicted Coordinated OFF Time

PDRC	Precision Departure Release Capability
PGUI	Planview GUI
RFRT	Request For Release Time
RMP	Research Management Plan
rTMA	Research TMA
RTT	Research Transition Team – a joint NASA/FAA activity to facilitate NextGen technology transfer
SADD	Schedule a Departure Dialog
SAIE	System Analysis Integration and Evaluation (NASA Project)
SDIF	Surface Data Interface
SDSS	Surface Decision Support System – often used interchangeably with the original SMS name
SDT	Scheduled Departure Time – proposed Coordinated Release Time (see CRT) computed by TMA and communicated to SDSS by PDRC.
SMS	Surface Management System – see SDSS
STBO	Surface Trajectory-Based Operations
TBFM	Time Based Flow Management
TFMS	Traffic Flow Management System – replaces ETMS
TGUI	Timeline GUI
TMA	Traffic Management Advisor
TMC	Traffic Management Coordinator
TMI	Traffic Management Initiative
TMU	Traffic Management Unit
TRACON	Terminal RADAR Approach Control
TSDE	Tactical Surface Data Exchange – replaces FOSA