

TAMS

Simulation Concept Document - Iteration 3 and 4

R. Suikat, DLR



Supported by:



Federal Ministry
of Economics
and Technology

Version: 1-0-0
Status: Released
Date: 2012-11-16
Filename: TAMS_SCD-v-1-0-0.doc

on the basis of a decision
by the German Bundestag

Copyright Note:

Privileged Business Information – Do Not Release

This document contains Trade Secrets of the TAMS Project Member Organisation. With the receipt of this document, the receiver agrees to keep strictly confidential, all Trade Secrets of the TAMS Project Member Organisations contained herein.

Copying this document, giving it to others, and the use or communication of the contents thereof, is forbidden without express authority.

Offenders are liable to the payment of damages. All rights are reserved especially in the event of the grant of a patent or the registration of a utility model or design.

Revision History

Version	Chapter	Page(s)	Date	Remark	Author
0-1-0	All	All	2011-03-09	Initial version	R. Suikat
0-9-0	All	All	2012-06-26	Merged input from FW	R. Suikat
0-9-1	All	All	2012-07-27	Final Draft	R. Suikat
0-9-1	All	All	2012-07-31	DLR internal Review	F. Piekert, F. Rudolph
0-9-1	All	All	2012-09-05	Final DLR review	R. Suikat
0-9-9	All	All	2012-09-10	Version for partner review	R. Suikat
1-0-0			2012-09-24	Release of Final Version	F. Piekert



Table of Contents

Revision History	3
Table of Contents	4
1 Introduction	6
1.1 Purpose	6
1.2 Writing process	6
2 TAMS Simulation Requirements	7
2.1 Simulation time.....	7
2.2 Aircraft simulation.....	7
2.2.1 Arriving aircraft (airborne):	7
2.2.2 Arriving traffic (on ground):.....	8
2.2.3 Turnaround process:	9
2.2.4 Departing traffic (on ground):	9
2.2.5 Departing traffic (airborne):	10
2.3 Terminal Processes	10
2.3.1 Passenger Attributes	11
2.3.2 Departing Passengers.....	11
2.3.3 Check-In Process.....	11
2.3.4 Security Process	12
2.3.5 Passport Control Process	12
2.3.6 Boarding Process.....	12
2.3.7 Arriving passengers	12
2.4 Interfaces from simulation to TAM systems	13
2.4.1 Timetick <i>[b, f, j, t, n, q]</i>	13
2.4.2 Radar data <i>[b, f, q]</i>	14
2.4.3 Initial ATC flightplan <i>[q]</i>	17
2.4.4 ATC Flightplan updates <i>[b, f, q]</i>	18
2.4.5 Airport Flightplan updates <i>[b, f, q]</i>	19
2.4.6 Turnaround process data <i>[j]</i>	19
2.4.7 Terminal process data <i>[h, t, x]</i>	19
2.4.8 Static data	24
2.5 Interfaces from TAM systems to simulation	25
2.5.1 Control of arriving flights via AMAN <i>[d]</i>	25



2.5.2	Control of taxiing and departing flights (SMAN) [e].....	25
2.5.3	Control of turnaround process via TMAN [g].....	26
2.5.4	Control of terminal processes via PAXMAN [x, u].....	26
3	References	27
4	Glossary / Abbreviations	28



1 Introduction

1.1 Purpose

This document describes the Simulation Concept for the third and fourth TAMS project iteration. The main step from iteration-2 to Iteration-3/-4 with respect to the simulation is that the simulation shall implement the planning results of the connected tactical tools (AMAN/DMAN/SMAN/TMAN). This requires a feedback loop to be closed from the tools to the simulation.

The document describes the functional requirements and the interfaces between the simulation and the TAM-suite, so that the development and implementation of the software can proceed.

1.2 Writing process

The agreed quality process for the TAMS project does not define a standard procedure for the creation phase of documents. For this document, the following procedure shall be applied:

- All partners are welcome and encouraged to write comments into the document. If you refer to other comments, please do not refer to them by the comment number assigned by Word, as this changes as soon as another comment is inserted.
- Only the document owner shall directly modify the text, unless others are assigned to contribute to specific sections.
- Whenever changes shall be made to the document (including comments), the document shall be checked out from the team site and checked in after adding the comment.
- Version numbers shall only be incremented by the document owner.
- Open questions/issues are highlighted in yellow and may be written in German.
- Sections not relevant to the current iteration are greyed out.

2 TAMS Simulation Requirements

This document lists the requirements for the TAMS total airport simulation suite. The requirements are structured in functional and data-interfacing requirements. The functional requirements list, which processes need to be simulated in which way, the interface requirements describe the data and formats that need to be made available to the TAM system.

2.1 Simulation time

1. The simulation shall generate the current simulation time and distribute it to the TAM suite. The simulation time shall be sent once per second in xml DateTime format.
2. The simulation shall be able to run faster than real-time. When the simulation is running at a different speed than real-time, the current simulation time shall still be sent once per second.
3. The simulation shall be able to pause on request. When the simulation is paused, the (unchanged) simulation time shall also be sent once per second.

2.2 Aircraft simulation

2.2.1 Arriving aircraft (airborne):

1. Simulation must be possible with and without input from planning systems.
2. Dynamic simulation of aircraft within tactical range, i.e. aircraft “known” to AMAN. For these aircraft radar tracks shall be generated. The range is defined by FIR entry.
3. An Individual Flight Plan message (IFPL) shall be generated and sent 3 hours before departure at outstation according to ADEXP standard. The content of the message is defined in section 3.7.
4. *Flightplan* update messages (FUMs) shall be generated and sent according to ADEXP standard. A FUM will be sent at take-off from outstation and when a significant change in ELDT occurs. The content of the FUM is specified in section 3.7. Departure delays at the remote airport shall be defined in the scenario event file and FUMs shall be sent at times corresponding to such departure delays.
5. Aircraft shall follow standard routes, the default route shall be defined in scenario data (ATC *flightplan*). For simulation without feedback from planning systems the runway and STAR shall be defined in the scenario file.
The route shall be specified within the region of interest (FIR Bremen). Route points outside the FIR are not specified and the flight is only actually simulated within the FIR.
6. ATOOT (Actual Take-off time from Outstation) shall be freely adjustable (before departure message (FUM) has been generated) to allow deviations from schedule. Adjustments may be defined in scenario event file or should be possible manually and/or via interface.
7. Time of flight shall be adjustable within certain bounds for airborne flights (before FIR entry) to allow deviations from schedule (mainly for long haul flights). Adjustments may be defined in scenario event file or should be possible manually.



8. Simulated aircraft (within FIR range) shall be controllable via commands (AMAN), to allow planning systems to control the arrival sequence. The simulation shall respect the following inputs from AMAN:
 - 8.1. Holding: AMAN can request a holding pattern to be executed by the aircraft at the default holding position defined in the AIP for the respective STAR. The command from AMAN shall contain the duration to be spent in the holding. The relevant message format is defined in 2.5.1.
 - 8.2. Estimated (requested) Landing Time: AMAN shall specify a requested landing time, which is the ELDT according to the sequence planned by AMAN. The simulation shall ensure that the actual landing occurs within +/- 30 seconds of this requested time.
 - 8.3. Arrival Runway: On initial radar contact AMAN shall send a message to the simulation specifying the arrival runway for this flight. The simulation will select the corresponding STAR and route the flight to the selected arrival runway. The arrival runway may be changed later on by AMAN and the simulation shall then route the flight to the new runway. Runway changes are not possible once the STAR entry has been reached, as no defined routings exist to a different runway after passing that point.
9. If no commands are sent from AMAN, simulated flights shall follow arrival route and STAR and land on runway defined in scenario file.

2.2.2 Arriving traffic (on ground):

1. Simulate taxiing traffic from touchdown to parking position. Ground radar data shall be generated for taxiing aircraft.
2. Taxi clearances shall be given from external tool (SMAN) via interface. SMAN shall be able to control the following parameters:
 - 2.1. Taxi route: SMAN will send taxi routes to the simulation. For this purpose the airport topology (routing network) is matched between the simulation and SMAN. SMAN will send a list of nodes on this network to the simulation. The final node will be the node corresponding to the selected stand/parking position of the flight, ensuring that the taxi route will lead to the correct parking position. The simulation shall route the aircraft along the nodes specified by SMAN. The message format is defined in 2.5.2. When the aircraft reaches its parking position the simulation shall send an actual inblock event (AIBT) to the main database and initiate the turnaround process. Rerouting shall be possible. The simulation shall then follow the new list of nodes sent by SMAN from the current position of the aircraft.
 - 2.2. Clearances: The simulation shall respect clearances issued by SMAN to control the traffic on the ground. For arriving flights, only taxi clearances will be issued.
 - 2.3. Stop bars: Stop bars are defined at a number of places in the node network to allow proper control of the ground traffic by SMAN. Stop bars shall be controllable (on/off) by messages from SMAN and flights shall stop in front of active stop bars.
3. If planning systems are not connected (SGMAN/SMAN/DMAN; fully automatic simulation) the parking position shall be taken from the scenario (schedule).
4. If planning systems are not connected (SGMAN/SMAN/DMAN; fully automatic simulation) taxi route shall be taken from the scenario (extended schedule data) or assigned by the simulation.

2.2.3 Turnaround process:

1. The simulation shall generate events upon beginning and end of defined turnaround steps. Standard durations for these processes are defined depending on aircraft type and the dependencies among sub processes are modeled. The model is defined to match the underlying turnaround process model of the TMAN. The main sub processes are: Toilet; Water; Deboarding; Cabin Board Supply; Offload Catering; Load Catering; Cleaning; Fueling; Unloading; Loading; Security Check; Cabin Check; Boarding Bus; Bus Transfer; Actual Boarding.
2. The turnaround process shall be simulated in real time. Start and end times of all sub processes are communicated directly to TMAN via xml-messages.
3. The turnaround process shall also be simulated in fast-time. Start and end times of all processes are also communicated to TMAN via xml-messages. This fast-time simulation provides estimate/plan data for the turnaround process to TMAN, thereby simulating the underlying resource management systems. The fast-time simulation shall be executed periodically (every five minutes) with the latest available estimated inblock times.
4. The turnaround simulator shall wait for a boarding-completed message from TMAN before closing the process chain. This allows TMAN to control "closing the gate", simulating the airline decision maker in case of late passengers. Thus, a TMAN operator can decide to wait for passengers or close the gate, and the simulation will account for this decision.

2.2.4 Departing traffic (on ground):

1. Simulate taxiing traffic including start-up and push back from parking position to assigned runway. Ground radar data shall be generated for taxiing aircraft.
2. Runway for take-off and taxi route shall be controllable via interface from SMAN (see also section 2.2.2). Note that in the TAMS project, SMAN and DMAN are closely coupled and SMAN will also send information originating from DMAN to the simulation.
3. Clearances shall be given from external tool (SMAN) and status messages from the simulation will be send back to SMAN. The following sequence of events shall occur for each departing flight:
 - 3.1. At TSAT SMAN will send a startup clearance to the simulation. The simulation shall initiate the startup of the flight. For twin engine flights startup takes three minutes.
 - 3.2. When the startup is completed the aircraft is ready for pushback/taxi depending on the parking position. If the aircraft is parked at a pushback position, SMAN will issue a pushback clearance via xml message. The simulation shall then perform a pushback. The direction and end position of the pushback shall be defined in the simulation base data for all runways used in the simulation runs, so that the simulation will execute a pushback to the proper position.
 - 3.3. When the pushback is finished (or after start-up on a taxi-out position) the simulation shall send a taxi-request message to SMAN.
 - 3.4. After receiving the taxi-request message SMAN will issue a taxi-route and a taxi clearance up to the line-up position on the departure runway. The simulation shall then taxi following this route.
 - 3.5. When the aircraft reaches the line-up position on the departure runway the simulation shall send a ready for take-off message to SMAN.
 - 3.6. After receiving the ready for take-off message SMAN will send a take-off clearance to the simulation, which shall then execute the take-off. Note that SMAN will not



issue this clearance before TTOT.

4. If planning systems are not connected (fully automatic simulation) take-off runway shall be taken from the scenario file.
5. If planning systems are not connected (fully automatic simulation), taxi route shall be taken from the scenario (extended schedule data) or assigned by the simulation.
6. Conflict detection for taxiing traffic shall cause aircraft to stop before colliding.
7. Traffic flow shall again be controlled via stop bars switched by SMAN. This allows to control the following aspects:
 - 7.1. Conflict avoidance: A number of stop bars are defined along the taxiways to allow SMAN to control the traffic flow and thus prevent conflicts on the ground.
 - 7.2. Runway crossings: Stop bars in front of each runway crossing are switched on by SMAN when a runway is in use for landing/take-off. The simulation will stop taxiing aircraft in front of active stop bars, thus preventing crossing a runway that is currently in use.
 - 7.3. Holding at runway holding position: While the taxi clearance of SMAN is all the way through to the lineup position on the runway, a stop bar at the runway holding position allows to properly queue the flights at the holding position. The conflict detection within the simulation shall ensure that queuing aircraft will stop behind the aircraft in front. SMAN will open the stop bar for just one flight when a lineup shall occur. The simulation shall then execute the lineup for the first aircraft in queue.

2.2.5 Departing traffic (airborne):

1. Simulate departing traffic from take-off to a defined range after. Radar data shall be generated for simulated flights up to FIR exit
2. Aircraft shall follow standard routings, default route to be defined in scenario data (scenario file).
3. If the runway configuration is changed, the departure runway and SIDs shall be selected accordingly.

2.3 Terminal Processes

Passenger movements are simulated in a microscopic fast-time simulation environment developed by DLR institute FW to simulate detailed passenger movements. The passenger simulation is based on a permanent time step related calculation of direction and velocity as well as assessment and resolution of potential conflicts. The modeling follows a strict order of the separate process steps.

The general procedure for passengers originally starting from the airport is as following (aligned to Stuttgart Airport):

Entry -> check-in -> bulk luggage -> security control -> passport control -> boarding

The general procedure for transit passengers is as following (aligned to Stuttgart Airport):

Entry -> passport control -> 2nd passport control -> boarding

According to passenger and flight attributes some of the process steps can be skipped.

2.3.1 Passenger Attributes

The passenger attributes set in the simulation determine the walking paths of the passengers and their terminal departure process.

Attribute	Type
Entry	Source number
Luggage	standard, none, bulk
Check-In	Standard counter, business / first counter, self, online, late night
Passport Control	None, Schengen, Non-Schengen, Critical
Gate	Gate number
Destination	Schengen, Non-Schengen
Origin	Schengen, Non-Schengen
Individual parameters	Walking speed, group size, entry time

As the flight attributes needed for the landside simulation are included in the passenger attributes (see table above), they will not be further explained in this context.

2.3.2 Departing Passengers

1. Departing passengers are generated on defined entry points. The arrival time at the airport depends on previously defined passenger and flight attributes in combination with scheduled take-off times.
2. The mode of transportation to the airport (car, bus, train) is modeled in form of passenger attributes and a restriction to certain entry points.
3. Transit passengers are generated on defined entry points with the arriving of the respective feeder (AIBT).

2.3.3 Check-In Process

1. In the simulation the Check-In counters are assigned according to defined opening hours, specified airlines (airline alliances) and passenger attributes (Check-In type, booking class).
2. The Check-In process is simulated using a standard service time for the passenger/document verification, the issuing of the boarding passes and the optional baggage drop off.
3. The duration of the service time depends on the Check-In type (counter, online, late night, with or without baggage) set in the passenger attributes.
4. The online Check-In is simulated by a service time for baggage drop off.
5. The late night Check-In is simulated by a direct path to security control.



6. Bulk luggage drop off is only possible after Check-In and is simulated by an additional service time for the specified bulk luggage counters.

2.3.4 Security Process

1. The security process is simulated by a service time needed for passenger and hand luggage screens and general checks according to aviation regulations.
2. The passengers' choice of the security control station is independent of the location of the Check-In counter and the boarding gate.

2.3.5 Passport Control Process

1. The passport control process is simulated by a service time representing the time needed for passenger identification.
2. The service time depends on the destination (Schengen / Non-Schengen, Critical) which is part of the passengers attributes.
3. For passengers on Schengen Flights a passport control is optional.
4. The location of the passport control follows from the actual gate of the flight which is set in the passenger attributes.

2.3.6 Boarding Process

1. In the simulation the gates are assigned to opening hours defined by the Estimated Start Boarding Time (ESBT) and a buffer time before Estimated Off-Block Time (EOBT).
2. The boarding process is simulated by a service time representing the time needed for passenger identification and boarding pass check.
3. The duration of the service time is set to a standard parameter defined according to the specific boarding procedure.
4. Each flight has a different gate that is set in the passenger attributes.
5. Estimated Off-Block Time (EOBT)

2.3.7 Arriving passengers

1. The TAMS concept aims on integrating land- and airside processes. Due to the fact, that the processes of arriving passengers at the airport are strictly separated from the processes of departing and transfer passengers and there are likely no impacts on the airside, arriving passengers are not included. Thus, in the simulation only departing and transit passengers are considered.

2.4 Interfaces from simulation to TAMS systems

Note: the letter [x] after the subsection title indicates the data path in the functional TAMS system architecture diagram [2], as can be seen in the following figure. All of those links directly interfacing a TAMS system with a simulator will be described below in chapters 2.4 [b, f, j, t, n, q] and 2.5 [d, e, g, x]. Links between TAMS systems are not described herein; these are part of the TAMS SYRD [2].

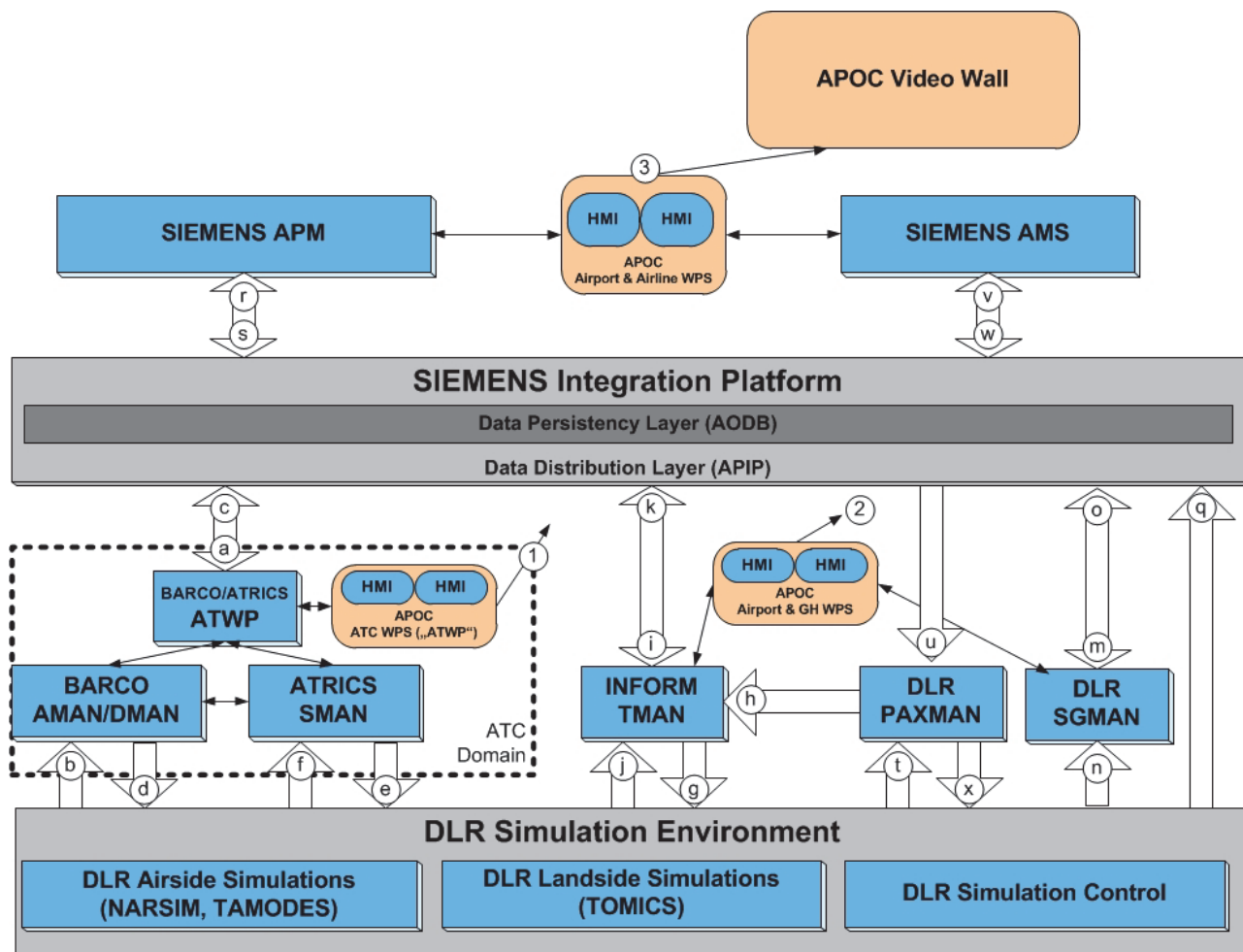


Figure 2-1: TAMS Interfaces and Logical Data Flow - Iteration 3 and 4

2.4.1 Timetick [b, f, j, t, n, q]

A timetick message shall be sent every second. The message shall be sent as an xml string using the xml-datetime format via udp-broadcast (port 1404). When the simulation is paused, the same time will be sent repeatedly. The broadcast message is part of each simulation – application interface.

Example time message:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?> <simtime>2010-07-06T14:04:00Z</simtime>\eot
```

In addition the time will also be sent to the APIP via WSDL [q] as this is necessary for the APM system.



2.4.2 Radar data [b, f, q]

2.4.2.1 Airborne traffic

1. The simulation shall provide radar data in ASTERIX Cat 62 (Ref 5) format to the TAM suite (specifically AMAN). Radar data are sent for all aircraft within the Bremen FIR at a rate of one message every four seconds. The messages shall be sent via udp. The aircraft is identified in the message by its callsign, which is transmitted in Data Item I062/390 "Flight Plan Related Data", Subfield CSN. A unique trackID (integer number) is also generated by the radar server.

The following table lists the data field that are sent by the simulation:

Data Item Ref. No.	Supported	Description	Value
I062/010	y	Data Source Identifier	
		SAC	SAC
		SIC	SIC
I062/015	n	Service Identification	
I062/040	y	Track Number	
		TRACK NUMBER	CUST_TR_NO
I062/060	y	Track Mode 3/A Code	
		CH	false
		Mode3/A reply in octal	MODE_A
I062/070	y	Time Of Track Information	
		Time Of Track Information	DATA_TIME
I062/080	y	Track Status	
		MON	0
		SPI	false
		MRH	0
		SRC	0
		CNF	0
		SIM	true
		TSE	0
		TSB	0
		FPC	false
		AFF	0
		STP	0
		KOS	0
		AMA	false
		MD4	0
		ME	false
		MI	false
		MD5	0
		CST	false
		PSR	false
SSR	false		
MDS	false		
ADS	false		
SUC	false		
AAC	false		
I062/100	n	Calculated Track Position (Cartesian)	
I062/105	y	Calculated Track Position (WGS-84)	
		Latitude in WGS-84	LATITUDE
		Longitude in WGS-84	LONGITUDE
I062/110	n	Mode 5 Data reports & Extended Mode 1 Code	



I062/120	n	Track Mode 2 Code	
I062/130	n	Calculated Track Geometric Altitude	
I062/135	n	Calculated Track Barometric Altitude	
I062/136	y	Measured Flight Level	
		Measured Flight Level	MODE_C
I062/185	y	Calculated Track Velocity (Cartesian)	
		Vx	X_VEL
		Vy	Y_VEL
I062/200	n	Mode Of Movement	
I062/210	n	Calculated Acceleration (Cartesian)	
I062/220	y	Calculated Rate Of Climb/Descent	
		Calculated Rate Of Climb/Descent	CD_RATE
I062/245	y	Target Identification	
		STI	1
		Callsign 6 bits each	CALLSIGN
I062/270	n	Target Size & Orientation	
I062/290	n	System Track Update Ages	
I062/295	n	Track Data Ages	
I062/300	n	Vehicle Fleet Identification	
I062/340	n	Measured Information	
I062/380	y	Mode-S / ADS-B Related Data	
		ADR	1
		ID	-
		MHG	-
		IAS	-
		TAS	-
		SAL	-
		FSS	-
		TIS	-
		TID	-
		COM	-
		SAB	-
		ACS	-
		BCR	-
		GVR	-
		RAN	-
		TAR	-
		TAN	HEADING
		GSP	VEL
		VUN	-
		MET	-
		EMC	-
		POS	-
		GAL	-
		RUN	-
		MB	-
		IAR	-
		MAC	-
		MPS	-
		BPS	-
I062/390	y	Flight Plan Related Data	
		TAG SAC	SAC
		TAG SIC	SIC
		CSN in Ascii	CALLSIGN



		IFI TYP	1
		IFI NBR	CUST_TR_NO
		FCT FR	FL_RULES
		FCT RVSM	1
		FCT HPR	1
		TAC	AIRCRAFT_TYPE
		WTC	WAKE_TURBULENCE
		DEP	DEPARTURE
		DST	DESTINATION
		RDS	DES_RWY
		CFL	-
		CTL	-
		TOD	-
		AST	-
		STS	-
		STD	-
		STA	-
		PEM	-
		PEC	-
I062/500	n	Estimated Accuracies	
I062/510	n	Composed Track Number	

2.4.2.2 Taxi traffic (ground radar)

1. The simulation shall provide ground radar data in ASTERIX Cat 10 (Ref 4) format to the TAM suite (specifically SMAN). These data shall be sent for aircraft flying below 400ft or moving on ground at a rate of one message every 4 seconds. The messages shall be sent via udp-multicast.

The aircraft is identified in the message by its callsign, which is transmitted in the Target Identification field. A unique trackID (integer number) is also generated by the radar server. Note that the track IDs of ground and airborne radar are NOT identical.

The following table lists the data fields that are sent by the simulation:

I010/200	y	Calculated Track Velocity in Polar Co-ordinates	
		Ground Speed	VEL
		Track Angle	HEADING
I010/202	y	Calculated Track Velocity in Cartesian Coord.	
		Vx	X_VEL
		Vy	Y_VEL
I010/210	n	Calculated Acceleration	
I010/220	y	Target Address	
		Target Address	MODE_S
I010/245	y	Target Identification	
		STI	1
		Target Identification 6 bits each	CALLSIGN
I010/250	y	Mode S MB Data	
		Mode SMB Data	MB_DATA
		Mode BDS1	MB_DATA_2
		Mode BDS2	MB_DATA_2
I010/270	y	Target Size & Orientation	
		LENGTH	TARGET_LENGTH

	ORIENTATION	TARGET_ORIENTATION
	WIDTH	TARGET_WIDTH
I010/280	n Presence	
I010/300	n Vehicle Fleet Identification	
I010/310	n Pre-programmed Message	
I010/500	n Standard Deviation of Position	
I010/550	y System Status	
	NOGO	0
	OVL	0
	TSV	0
	DIV	0
	TTF	0

2.4.3 Initial ATC flightplan [q]

At 3hrs before take-off (both for arriving and departing flights) the simulation will issue an ATC Individual Flight Plan (IFPL) message as defined in the ADEXP standard. The message shall be transmitted to the AODB via WSDL and contain the following fields:

Field Name	Description	Sample
addr	Address field	We always use
adep	Departure airport, ICAO	CFMUTACT here
Ades	Destination airport, ICAO	
arctyp	ICAO aircraft type	
ceqpt	Communication equipment, always set to „S“ in simulation (not really being used)	
seqpt	Surveillance equipment, always set to „C“ in simulation	
Eobd	EOBD, format yymmdd	
eobt	EOBT, format hhmm	
filtim	filing time for <i>flightplan</i> , format ddhhmm	
lflid	<i>flightplan</i> -ID (2char followed by 8 digits)	LH10101505
	Simulation will use airline code (IATA) followed by flight number padded to four digits followed by SOBT. Note that operationally this identifier can be anything, only formal requirement is 2 characters followed by 8 digits.	(LH1010, SOBT 15:05)
wktrc	wake turb. Categ.	
fittyp	ICAO flighttype, (always use „S“ for scheduled flight)	
fitrul	ICAO flightrule (always use I for instrument flight)	
ttleet	total elapsed time (flighttime), format hhmm	
Route	ICAO field 15 information containing speed, requested flight level and route, probably not used by any system but transmitted for completeness	
Rfl	requested flight level	F300
Speed	requested speed, knots or Mach, will be given in knots	N400
Sid	5-character identification of selected SID (Standard Instrument Departure) for departing flights	
Star	5-character identification of selected STAR (Standard Arrival Route) for arriving flights	
Rtepts	List of routepoints. Each point is given by its pt-ID (-PTID). Simulation will transmit flight level and estimated time over for the first point that is within FIR Bremen as well as the	



	STAR Entry point. For arriving flights.
Arcid	aircraft-ID, using callsign in simulation. Note that according to the standard this can also contain the aircraft registration.

An example message is shown below showing the IFPL for the single flight selected for the Quality Gate test.

```
-TITLE IFPL -BEGIN ADDR -FAC CFMUTACT -END ADDR -ADEP EDDF -ADES EDDX -ARCTYP A30B -CEQPT
S -SEQPT C -EOBD 100930 -EOBT 0810 -FILTIM 300510 -IFPLID LH00800905 -SRC RPL -WKTRC H -
TTLEET 0055 -FLTRUL I -FLTTYP S -ROUTE N0400F330 MARUN Y153 WRB N850 NIE T901 NOLGO -
BEGIN RTEPTS -PT -PTID TOLGI -FL F300 -ETO 090525083443 -PT -PTID NOLGO -ETO 090525084828
-END RTEPTS -STAR NOL23 -RFL F300 -SPEED N0400 -ARCID DLH008
```

2.4.4 ATC Flightplan updates [b, f, q]

Flightplan update messages (FUMs) as defined in Ref 7 will be sent at specific events during the simulation. The messages will also be sent to the AODB using WSDL. These events are:

1. Take-off (from outstation or own airport): A flight has departed from an airport and a *flightplan* activation (FUM) is sent and distributed by the CFMU.
2. 3hrs before estimated landing time.
3. Delay of an arriving flight: A significant delay occurs and a FUM is sent to update the ELDT.

Field Name	Description	Sample
addr	compare IFPL	
ifplid	"	
arcid	"	
arctyp	"	
adep	"	
ades	"	
eobd	"	
eobt	"	
eldt	estimated landing time	
fltstate		AA
estdata	est. Time and flight level at coordination point (STAR entry point)	

An example FUM message indicating a delay for an airborne flight is shown below:

```
-TITLE FUM -BEGIN ADDR -FAC CFMUTACT -END ADDR -IFPLID LH00800905 -ARCID DLH008 -ARCTYP
A30B -ADEP EDDF -ADES EDDX -EOBD 100930 -EOBT 0840 -ELDT 100930093000 -FLTSTATE AA -
ESTDATA -PTID NOLGO -ETO 090525084828 -FL F300
```

2.4.5 Airport Flightplan updates [b, f, q]

The simulation shall provide *flightplans* and updates to the TAM suite. The following update messages shall be sent:

4. Additional Flight: The simulation shall send a “SubmitNewFlightplanMsg” via the “DLR Event Service” to the AODB. The content of this message is defined in the TAMS Interface Description Document [3].
1. Cancelled Flight: The simulation shall send a “CancelFlightMsg” via the “DLR Event Service” to the AODB. The content of this message is defined in the TAMS Interface Description Document [3].
2. Stand/Gate update: The simulation shall send a “ChangeStandMsg” via the “DLR Event Service” to the AODB. The content of this message is defined in the TAMS Interface Description Document [3]. This message would normally be sent by the SGMAN, but for Iteration-1 the simulation will provide this message.

2.4.6 Turnaround process data [j]

The simulation shall send turnaround process data directly to the TMAN system. The information will be sent as a formatted xml-string via a TCP-connection. The interface and the message format is described in: GS RealTime, Process Interface Description, Version 2.0, 8.7.2010.

As soon as a connection is established between the simulation and TMAN a bulk message containing all available information is being sent. There is hence no need for a specific bulk request.

The turnaround process data shall include estimated/predicted times as well as actual times for all defined turnaround sub processes.

2.4.7 Terminal process data [h, t, x]

The terminal process data will be provided by the DLR Landside Simulation to the TAMS system in Iteration 3 and 4. This data includes actual landside information about the current pax processing (at CKI, Security and gate) and the Estimated Passenger at Gate Time EPGT.

The landside system uses the microscopic simulation tool TOMICS to simulate passengers in real-time. TOMICS is based on static terminal data as described in chapter 2.4.8.2 and uses data from the actual flight plan and the terminal entrances as initial values for the simulation.

TOMICS simulation results are provided to the airside system and the PaxMan. The Passenger Manager (PaxMan) is able to predict the EPGT rather accurately on base of the actual situation throughout the terminal by monitoring the different passenger processes and general activities at different service facilities. This allows for considerably improved accuracy as well as timeliness of passenger status information and thus consolidate landside and airside planning for Total Airport Management. The result consists of current and predicted data sets.

2.4.7.1 Required Simulation Input Data [t, x]

The following chapter lists the requirements for the Terminal Simulation. The requirements are structured in Passenger Service Times, in Flight Schedules, in a Source Allocation Plan and in Passenger Properties.



2.4.7.1.1 Passenger Service Times

A service time will be generated on each process step to simulate the time needed for communication between passenger and employee as well as the respective workload. The service time is based on empirical analysis of real airport processes and will be realized with different distributions.

- Distributed pax service time at the check-in counter
- Distributed pax service time at the passport control
- Distributed pax service time at the security control
- Distributed pax service time at the boarding pass control

2.4.7.1.2 Flight Schedule (list of flight plans) [t]

An initial Flight schedule is taken from table 'sim_flightplan' of datapool 'EADF' (a related arrow is missing in Figure 2-1). All updates on these flights will be taken from the SIAMOS-Platform via polling. The following table lists the required data fields of the initial *flightplan*:

	Data Field	Data Type	Description
1	sobt	date	Scheduled Off Block Time
2	eobt	date	Estimated Off Block Time
3	callsign	varchar	Callsign ICAO-Code
4	AIRLINE	varchar(3)	Airline ICAO-Code.
5	orig_airport	varchar(4)	Airport ICAO-Code.
6	dest_airport	varchar(4)	Airport ICAO-Code.
7	ac_icao	varchar(4)	Aircraft ICAO-Code.
8	FLIGHT_FUNCTION	char	J: scheduled flight passengers F: scheduled flight cargo M: scheduled flight airmail C: charter flight passengers P: positioning flight
9	GATE	varchar	Gate identifier.
10	PAX	int	Number of booked passengers
11	id	varchar(40)	UUID (Primary Key)
12	ts	timestamp	CURRENT_TIMESTAMP
13	direction	char	A: arrival D: departure
14	flight_type	char	Line domestic Line/charter continental Line/charter intercontinental unknown

2.4.7.1.3 Airport Resources Source Allocation Plan [x]

A source allocation plan for airport resources (e.g. check-in counter, boarding gate) is not distributed in iteration 3 or 4. PaxMan manages the resources himself based on some LOS (Level Of Service Constrains). The simulation runs with a sample allocation plan.

For the simulation, a source allocation plan must provide data as follows:

- A flight allocates one or more airport resources.
- For each resource and flight a start and an end time for allocation is defined.

2.4.7.1.4 Passenger Properties

The individual passenger characteristics are presented according to passenger type (e.g. tourist, businesspeople). This concerns the distribution of walking speeds, the individual service area and the orientation time. These properties are given for many different models and does not depend on the traffic scenario.

At the beginning of the simulation, the passenger properties closely follow the expanded flight schedule (e.g. number of PAX according to booking classes, accepted pieces of luggage according to flight type or the modal split according to flight type).

During the simulation, various properties are added or lost:

- Passenger Type (direct/transfer, private(e.g. tourist), businesspeople)
- Number of checked bags and handbags
- Walking speed
- Route Definition for Passengers
- Group Type
- Entry time

2.4.7.2 Simulation Output Data [t, h]

There are two different kinds of output data. On one hand the PaxMan needs a lot of information to serve internal tools like Queue Observer, TeMaDis Localizer, Security agent. On other hand there must be a connection to airside operations.

It was decided that passenger output data is routed over TMAN, as TMAN needs this information as well and also because this allows TMAN to terminate the boarding process when passengers would arrive too late. When the aircraft is ready for boarding, TAMODES sends a boarding start event to TMAN. This event is forwarded to the passenger simulation as well, and the passengers will move from the gate area into the airplane. The passenger simulation continuously informs TMAN of the number of boarded and missing passengers. When all passengers have boarded, TMAN will send a boarding completed event to TAMODES, which will then complete the turnaround process and send an aircraft ready event back to TMAN. The following defined output data will be transferred to the TAM System TMAN.

A tool called TAMS Realsim Connector was established to fulfill these needs and puts the basic information to PaxMan database.



When a simulated passenger passes a process step, a check-in counter, a security lane or a terminal gate during simulation, the tool reports such events back to the PaxMan database. There are 3 tables with structure as follow:

	Data Field	Data Type	Description
1	tomics_pax_id	INTEGER	Reference to TOMICS person ID
2	attributes	VARCHAR(255)	delimited attributes of each single person
3	simtime	DATETIME	Current simulation time person passed by
4	connector_id	INTEGER	Reference to task station

These raw data are the basis of further calculations like the filling of global_dep_flights_status or global_dep_flight_status_first_last. See next chapters. The TOMICS integrated "SensorSim" is used to simulate state of the art sensors to get information about traffic flow and also emulating the imprecise information provided by today's sensors in a real environment. The produced data are the base for PaxMan tools Queue Observer and Security agent. The structure of output for defined camera observation system is as follow:

	Data Field	Data Type	Description
1	x	float	Determined x coordinate
2	y	float	Determined y coordinate
3	roomID	INTEGER	Reference to TOMICS room ID
4	timestamp	float	Current simulation time of recognition

2.4.7.2.1 global_dep_flights_status

The global_dep_flights_status Table contains actual information from the terminal simulation and generated values from the forecast simulation. These values are used from pax_radar and TMAN.

	Data Field	Data Type	Description
1	Callsign	VARCHAR(7)	Callsign
2	SOBT	DATETIME	Scheduled Off Block Time
3	TOBT	DATETIME	TARGET Off Block Time
4	PAX	INTEGER	Number of booked Pax

5	Airline	VARCHAR(2)	Airline Code
6	Checked_In	INTEGER	Number of checked in Passengers
7	Security_Checked	INTEGER	Number of security checked Passengers
8	Boarded	INTEGER	Number of On-Board Passengers
9	Boarded_Transfer	INTEGER	Number of On-Board Transfer Passengers
15	forecast_boarded_percent	DATETIME	Forecast - Percentage of On-Board Passengers
16	forecast_last_passenger_boarded	DATETIME	Forecast - Last Passenger On-Board (EPGT)

2.4.7.2.2 global_dep_flights_status_first_last

The global_dep_flights_status Table contains actual information from the terminal simulation. These values are used from pax_radar and TMAN.

	Data Field	Data Type	Description
1	Callsign	VARCHAR(7)	Callsign
2	SOBT	TIME	Scheduled Off Block Time
4	First_Checked_In	DATETIME	First Passenger Checked In
5	Last_Checked_In	DATETIME	Last Passenger Checked In
6	First_Security_Checked	DATETIME	First Passenger Security Checked
7	Last_Security_Checked	DATETIME	Last Passenger Security Checked
8	First_Boarded	DATETIME	First Passenger On-Board
9	Last_Boarded	DATETIME	Last Passenger On-Board

2.4.7.2.3 QueueObserver

The queue observer determines the queue length at each open task station. The algorithm takes the coordinates of passengers as input. The coordinates are retrieved from the simulation by a sensor simulation for cameras.



2.4.8 Static data

There are some static data required by the simulation as well as the TAM tool suite. These include the flight schedule and airport data.

2.4.8.1 Flight schedule data [q]

The flight schedules define the baseline scenario for each experiment. These schedules comprise all flights scheduled for the simulated day of operation, and include those data for each flight that are commonly defined in the long term planning phase. In SESAR terms these *flightplans* are called Shared Business Trajectories (SBT).

The simulation shall send the initial flight schedules to the AODB, when the experiment is started. This will be a one-time block-send of all schedules in the scenario. The format and content is defined in the TAMS Interface Description Document [3]. The data transmission will be manually triggered by operator input.

2.4.8.2 Airport data

Airport static data can comprise numerous things: Airport geometry, stand and gate descriptions and parameters, taxi-ways and usages, possible runway configurations, etc. It will be part of the integration process of the TAM suite at a target airport to collect these data and make them available to the tools contained. To this end a standard may have to be defined for storage of these data in the AODB. This will then imply that all tools must be able to read their relevant configuration data from the AODB.

The simulation is in this respect just a specific target airport. This means that it is part of the integration of the TAM suit with the simulation that the relevant parameters as specified above be collected for the SIMULATED airport and also stored inside the AODB. It is extremely unlikely that any target airport will provide such data in an AODB conforming standard to be simply uploaded into the AODB when the TAM suite is integrated, and neither will the simulation. Instead, it will be necessary to manually (or supported by some conversion tools) transfer the available data into the agreed formats and then store the resulting database permanently in the AODB.

2.4.8.2.1 Terminal Layout

The terminal layout shows a generic international airport, which is modeled in TOMICS. The proportions between facility quantities are based on real construction drawings. The model defines

- Rooms
- Entries
- Exits
- Obstacles and
- Handling facilities.

The definition for the handling facilities' allocation is based on the source allocation plan.

The simulation does not include any kind of defined routes for the passenger movement, only a strict temporal order which depends on the terminal process (2.4.7).

2.5 Interfaces from TAM systems to simulation

An important feature of the simulation will be the ability of fully automated runs including planning systems (XMAN) controlling the simulation. For this purpose, interfaces/protocols must be available. These interfaces are required only for the automatic control of the simulation by the tools and replace the manual control with human controllers and pilots. Hence, these interfaces do not exist in reality, and therefore protocols/procedures can be defined and agreed freely between XMAN and simulation. These interfaces are represented by the links d, e, g, and x.

2.5.1 Control of arriving flights via AMAN [d]

Arriving flights shall be controlled by the simulation such that the ELDT computed by the AMAN will be achieved. To this end AMAN sends a message to the simulation whenever there is a significant change in the ELDT or the runway for any given flight. This occurs when AMAN receives the *flightplan* from the APIP, when radar contact is established (flight enters the FIR) and also whenever an ELDT gets changed to do a sequence change. The message contains the following information:

- Callsign
- Runway
- ELDT
- Holding pattern and holding duration (optional)

The interface from AMAN to the simulation uses a simple xml-format transmitted via a tcp connection. Times are given in UNIX time.

Sample AMAN message:

```
<AtwpMsg>
  <AmanMsg timestamp="1243235726">
    <Callsign>DLH002</Callsign>
    <Rwy>23</Rwy>
    <ELDT>1243236985</ELDT>
  </AmanMsg>
</AtwpMsg>
```

2.5.2 Control of taxiing and departing flights (SMAN) [e]

Taxiing and departing flights are controlled by SMAN. For simplicity in the current setup SMAN will also send the departure clearances, which are generated by DMAN. SMAN and DMAN are closely coupled in the TAMS suite and hence SMAN gets the clearances from DMAN anyway. The flights are controlled by

- Clearances, i.e. taxi-, hold-, startup-, pushback-, lineup-, take-off-clearance.
- Routes, a series of link sections defined on the airport ground layout provided by the simulation.
- Stopbar, stopbars may be switched on/off to control the traffic on ground. This includes stopbars at the runway holding positions, stopbars to prevent crossing of the runway and additional stopbars to separate sections on the ground so that conflicts can be avoided.

The messages are also sent in xml format via a tcp connection. The format is defined in [8].



2.5.3 Control of turnaround process via TMAN [g]

The only real control information that is received by the turnaround simulation from the TAMS tools is the “boarding completed” message from the TMAN. TMAN is generating this flag based on the number of boarded passengers and the expected arrival time of the passengers which it receives from the landside simulation in relation to the aircraft’s delay.

2.5.4 Control of terminal processes via PAXMAN [x, u]

All external updates for the landside are received over the Passenger and Airport Data Exchange Unit (PAXU). The PAXU is used to manage the data exchange between landside and airside systems and simulations in TAMS. The central TAMS integration platform provides the PAXU with updates on the initial *flightplan*. These updates include information about actual and estimated inblock and offblock times (EIBT, AIBT, AOBT, TOBT) and also information about stand changes. Based on these changes the PAXU generates updates events for the tools inside the PaxMan Suite and the landside simulation environment TOMICS. All landside PaxMan changes within the simulation environment are managed via the TAMS Realsim Connector. This tools scans for new updates for persons and task stations.

3 References

The following documents are referenced in this Operational Concept Description:

Nr	Document Title	Version / Status	Author / Company
1	TAMS OCD_v1-0-1	1-0-1	DLR
2	Tams System Requirements Description I3	1-3-0	Siemens
3	Tams SAD/IDD	2-12-0	Siemens
4	EUROCONTROL STANDARD DOCUMENT FOR SURVEILLANCE DATA EXCHANGE Part 7, Cat 10	1.1, March 2007	Eurocontrol
5	EUROCONTROL STANDARD DOCUMENT FOR SURVEILLANCE DATA EXCHANGE Part 9, Cat 62	1.1, Dec 2009	Eurocontrol
6	EUROCONTROL STANDARD DOCUMENT for ATS Data Exchange Presentation (ADEXP)	2.1, Dec 2001	Eurocontrol
7	IFPS and RPL Dictionary of Messages	1.0, Mar 1998	Eurocontrol
8	SMAN - Technical Documentation - Simulator Interface	1.1, Jun 2011	ATRICS



4 Glossary / Abbreviations

A-CDM	Airport – Collaborative Decision Making
ADEXP	ATS Data Exchange Presentation
AIBT	Actual In-Block Time
AMAN	Arrival Manager
ANSP	Air Navigation Service Provider
AODB	Airport Operations DataBase
APIP	Airport Process Integration Platform
ASTERIX	All Purpose ST ructured E urocontrol Su Rveillance Information EX change
ATC	Air Traffic Control
ATOOT	Actual Take-off time from Outstation
CFMU	Central Flow Management Unit
DMAN	Departure Manager
DPI	Departure Information Message (ADEXP)
EADF	Eingangs- und Ausgangs-DatenFusion
ELDT	Estimated Landing Time
ESBT	Estimated Start Boarding Time
EOBT	Estimated Off-Block Time
EPGT	Estimated Passenger at Gate Time
FIR	Flight Information Region
FUM	<i>Flightplan</i> Update Message (ADEXP)
IFPL	Individual <i>Flightplan</i> Message (ADEXP)
LOS	Level Of Service
OCD	Operational Concept Document
PaxMan	Passenger Manager
PAXU	Passenger and Airport Data Exchange Unit
RBT	Reference Business Trajectory
SBT	Standard Business Trajectory
SCD	Simulation Concept Document
SESAR	Single European Skies ATM Research
SGMAN	Stand/Gate Manager
SID	Standard Instrument Departure (Route)
SMAN	Surface Manager
STAR	Standard Arrival Route



SYRD	System Requirements Document
TAM	Total Airport Management
TAMS	TAM Suite
TMAN	Turn around Manager
TOBT	Target Off-Block Time
TSAT	Target Startup Approval Time
TTOT	Target Take-Off Time
WSDL	Web Services Description Language
XMAN	tactical planning tool (AMAN/DMAN/SMAN, etc.)