

272098 v2.2

Aptean Ltd
Copyright © 2011-2026.

Contents

- 1 272098.....1**
 - 1.1 Client Requirement.....2
 - 1.2 Solution.....2
 - 1.3 Scope.....5
- 2 Set-up.....6**
 - 2.1 Pre-requisites.....6
 - 2.2 Menu Structure.....6
 - 2.3 Data.....6
- 3 Functional Description.....7**
 - 3.1 EDI Interface Setup.....7
 - 3.2 File Processing - ?SAL? Message.....8
 - 3.3 File Processing - ?SAU? Message.....9
 - 3.4 File Processing - Interface Tracking.....10
 - 3.5 Interface Errors.....12
 - 3.6 File Resending.....13
 - 3.7 Interface Cross Docking Enquiry.....14
- 4 AUTHORISED BY.....18**

1 272098



DHL MTS

Update Trip Info on S&L Receipt

FUNCTIONAL SPECIFICATION - 10.4.7

18/01/2011 - 2.2
Reference: 272098 SA-7Y5ARG



1.1 Client Requirement

Change Request Summary:

MTS to update the Trip information when receives a Scan n Load (Scan n Unload) file.

Change Request Details:

In Exel MDCs/MSCs we create a Trip using Scan n Load, the receiver also uses scan n load application to confirm the quantity of each material he has received. MTS only receives a file when user creates a Trip but does not receive any information when material actually received by next cross dock location. We have done this change already in a Scan n Load application to send another file, named Scan n Unload, when user receives a material at the delivered cross dock location, file format is attached. MTS when receives a file updates the information. One most possible error on this if Trip is still not updated by user (Trailer information) means it is in Planned status. Please advise how we can overcome on this possible error.

Benefits identified as a result of the change:

Business requirements to minimise the human errors.

1.2 Solution

A new inbound interface message ?SAU? will be created for the ?Scan & Unload? process to compliment the existing inbound interface message ?SAL? for the ?Scan & Load? process.

A new standard directory structure will be required for the upload of the inbound ?SAU? files for files processed, acknowledged and failed; a new database job will be required to process the file; new procedures and functions in the package ?DP_INT_MSG? will be required to read and process the ?SAU? files.

A file may only be reprocessed if the user has access to the function ?SAP_REPROCESS_BOOKINGS?; the reprocessing of the ?SAU? files will follow the logic of the reprocessing of the existing ?SAL? files so that the file is obtained from the archive directory and placed in the inbound directory to be reprocessed by the database job.

It will be possible to edit the ?SAU? detail lines in the ?Interface Errors? screen if the status of the header record is ?FAILURE?; it will then be possible to save the amendments and the status of the header and detail records will be updated to ?FAILURE_RR? to indicate that the file has been re-sent for processing and is present in the inbound directory; the amended data will then be used to create a new file, in the same format, to upload.

New tables called ?INT_SAU_HEADER_RESEND? and ?INT_SAU_DETAIL_RESEND? will be required for the edited data; these new tables will include columns for all of the relevant data contained in the ?SAU? files plus columns to identify the records and to enable reprocessing:

HEADER

INT_RECORD_ID (the ?SAU? filename)
MTS_OPERATION
MSG_TYPE
FROM_LOCATION (received in ?SAU? file)
CREATED_DATE
CREATED_BY
COMMENTS
RESEND_ERROR_MESSAGE
RESEND_PROCESSED

DETAIL

INT_RECORD_ID (the ?SAU? filename)
REC_NUM
RECORD_STATUS
SCHED_NAME (of trip found for the booking reference)
TRIP_ID (of trip found for the booking reference)
OMS_REF (found for the booking reference)
START_UNLOADING_DATE (received in ?SAU? file)
START_UNLOADING_TIME (received in ?SAU? file)
END_UNLOADING_TIME (received in ?SAU? file)
BOOKING_REFERENCE (received in ?SAU? file)
QTY_RECEIVED (received in ?SAU? file)
CURRENT_DEPOT (received in ?SAU? file)



COMMENTS

If the status of the header record is not ?FAILURE? or ?PARTIAL? then the reprocessing will not be allowed because there will be no data to reprocess; the reprocessing will only process the detail lines that exist at status ?FAILURE? and the header status will be updated accordingly.

Therefore only files that failed to upload fully may be reprocessed and the files in the directory for failed uploads will be those files that have at least one detail line that failed to upload (i.e. were updated to status ?FAILURE?).

New tables called ?INT_SAU_HEADER? and ?INT_SAU_DETAIL? will be required to store the header and detail sections of the ?SAU? file with multiple detail lines possible per header record; these new tables will include columns for all of the data contained in the ?SAU? files plus columns to identify the records and to enable reprocessing:

HEADER

INT_RECORD_ID (the ?SAU? filename)
 MTS_OPERATION
 MSG_TYPE
 RECORD_STATUS
 NUM_SCANS
 NUM_SUCCESS
 NUM_FAILURE
 FROM_LOCATION (received in ?SAU? file)
 VALIDATION_ERROR
 CREATED_DATE
 CREATED_BY
 UPDATED_DATE
 UPDATED_BY
 RAW_FILE

DETAIL

INT_RECORD_ID (the ?SAU? filename)
 REC_NUM
 RECORD_STATUS
 SCHED_NAME (of trip found for the booking reference)
 TRIP_ID (of trip found for the booking reference)
 OMS_REF (found for the booking reference)
 START_UNLOADING_DATE (received in ?SAU? file)
 START_UNLOADING_TIME (received in ?SAU? file)
 END_UNLOADING_TIME (received in ?SAU? file)
 BOOKING_REFERENCE (received in ?SAU? file)
 QTY_RECEIVED (received in ?SAU? file)
 CURRENT_DEPOT (received in ?SAU? file)
 VALIDATION_ERROR
 ACTION
 SOURCE_SYSTEM

The files will have the name format of ?SAUXXXXXXX.csv? and will have lines for the header and details sections as follows:

HEADER

Opening Tag (i.e. <RECHEADERSTART>)
 Database
 From Location
 Closing Tag (i.e. <RECHEADEREND>)

DETAIL

Opening Tag (i.e. <XDOCKSTART>)
 Start Unloading Date
 Start Unloading Time
 End Unloading Time
 Booking Reference
 Quantity Received
 Current Depot (i.e. Unloading Location)
 Closing Tag (i.e. <XDOCKEND>)

A new page of the ?Interface Errors? screen called ?SAU? will be created to display the details from the ?SAU? file processed; this page will contain header and detail sections to correspond with the file format and a ?Reprocess? button will be available for files that failed to upload correctly. A ?Search? button will also be provided to allow the user to select records in a popup screen based on the following criteria, for example:

- Record ID
- Message Type
- Status
- From Location
- Created Date Range
- Trip ID
- OMS Reference
- Booking Reference
- Current Depot



It will not be possible to modify the data in the ?Interface Errors? screen as the detail lines in the ?SAU? file may be reprocessed until they are processed successfully.

Each item will be validated to ensure that it is populated and that it exists where appropriate (e.g. a valid location, booking reference, date/time format, non-negative quantity), an order and a trip can be found for the booking reference and that the order is due to be delivered to the current depot from the ?from location? on a trip stop.

Whereas the ?SAL? message will create a trip and add the current order for the booking reference to it, the ?SAU? message will use the booking reference and locations provided to obtain the current order (i.e. OMS reference) and the current trip ID for the order between the locations.

If a detail line of the ?SAU? file is deemed invalid then it will be marked as a ?FAILURE?; the header record status will be marked as ?FAILURE? or ?PARTIAL? as appropriate (i.e. a header record will have a status of ?PARTIAL? if it has at least one detail line that has failed to upload and at least one detail line that has uploaded successfully); the number of successful and failed detail lines will be counted and displayed in the ?Interface Errors? screen like the ?SAL? message; if the header record status is ?FAILURE? or ?PARTIAL? then it may be reprocessed because the ?SAU? file will have been placed in the failed directory.

If the header line and the detail lines are valid then the header status will be marked as ?PROCESSED? and the detail status will be marked as ?SUCCESS?.

If the validation of the file format and items is successful then the data will be uploaded into a new table called ?SCH_ORD_XDOCK? via a function in a new package called ?DP_INT_TRK? (i.e. ?Database Package Interface Tracking?); this package may be used by other processes to maintain the new table at different stages of the order processing to provide a comparison between quantities loaded and unloaded.

This new table will contain columns as follows:

TABLE ?SCH_ORD_XDOCK?

BOOKING_REF
FROM_LOCATION
OMS_REF
ACTIVITY (i.e. ?LOAD? or ?UNLOAD?)
TRIP_ID
DU_QTY
WEIGHT
VOLUME
DU_TYPE
STATUS (i.e. ?SUCCESS?, ?FAILURE? or ?WARNING?)
ERROR_TEXT
REASON_CODE (i.e. for)
CREATED_DATE
CREATED_BY
UPDATED_DATE
UPDATED_BY
ACTIONED
ACTIONED_DATE
ACTIONED_BY
FREE_TEXT

The ?FROM_LOCATION?, ?OMS_REF?, ?ACTIVITY? and ?TRIP_ID? will form a unique primary key on the table.

This table may be used to store information about the loading and unloading of materials at different stages; therefore the ?SAL? message will be changed to update this table so that a comparison may be made with the data provided by the ?SAU? message.

Further validation will then be required to determine whether the details for the trip stop have any discrepancies, for example, the order status, trip status and trip stop type will be checked for particular values and, if not as expected, they will create a warning; the status of the table will then correspond to the validation performed.



It would be expected that a trip for the ?SAU? message will have a status of ?EN-ROUTE? and not ?PLANNED?, for example, and that the trip stop type would be for cross-docking; however, this status would not stop uploads/updates to the ?SCH_ORD_XDOCK? table but it will generate a warning message.

N.B. No trip details will be updated in MTS by the ?SAU? message.

A number of checks will be performed when the ?SAU? message is uploaded for comparison with previous stages of the order found:

- ?Missing from the Trip? (i.e. the order is on the inbound trip but is not in the ?SAU? file)
- ?Difference in Quantity? (i.e. the previous quantity for the order is different to the quantity received)
- ?Extra Quantity on the Trip? (i.e. the order is in the ?SAU? file but is not on an inbound trip)

The previous quantity of the order may be taken as follows:

1. Quantity in the ?SAL? message
2. ?POC? quantity
3. ?Planned? quantity (i.e. the original quantity ordered)

The previous quantity found will be recorded in the column ?ERROR_TEXT? for reference.

The ?DU_QTY?, ?WEIGHT?, ?VOLUME? and ?DU_TYPE? will be variables that will be set per trip event if possible.

A new form called ?INT_XDOCK_ENQ? (i.e. ?Interface Cross Docking Enquiry?) will be created to display the data contained on table ?SCH_ORD_XDOCK? and to allow the user to record whether any discrepancies recorded have been assessed, the user may set the flag ?ACTIONED? to ?Y? enter a description of the action in the column ?FREE_TEXT?.

Breakdown:

- New ?SAU? file processing: dev-3 days; test-1 day
- New ?SAU? interface errors page: dev-4 days; test-1.25 days
- New ?SAU? reprocessing: dev-1 day; test-0.5 day
- New ?DP_INT_TRK? package: dev-1.5 days; test-0.5 days
- New ?INT_XDOCK_ENQ? form: dev-3 days; test-1 day

N.B. Implementation assistance will be charged on a T&M basis at the relevant rate.

1.3 Scope

This change will be applied to system version 10.4.7 on SARTST and once approved SARPRD.



2 Set-up

2.1 Pre-requisites

System parameters will need to be set.

2.2 Menu Structure

New screens will be added to the menus.

2.3 Data

New tables will be introduced as described in Appendix A.



3 Functional Description

3.1 EDI Interface Setup

A new inbound interface message ?SAU? will be created for the ?Scan & Unload? process to compliment the existing inbound interface message ?SAL? for the ?Scan & Load? process.

New directories will need to be setup to store the ?SAU? file information, these will be:

/webint/\${DB}/interface/SAU/INBOUND/processed

/webint/\${DB}/interface/SAU/INBOUND/ack

/webint/\${DB}/interface/SAU/INBOUND/failed

Where \${DB} represents the database name.

The following system parameters will be created to process the ?SAU? files:

System Parameter	Value
SAU_PATH	/webint/\${DB}/interface/SAU/INBOUND
SAU_ACK_PATH	/webint/\${DB}/interface/SAU/INBOUND/ack
SAU_FAILED_PATH	/webint/\${DB}/interface/SAU/INBOUND/failed
SAU_PROCESSED_PATH	/webint/\${DB}/interface/SAU/INBOUND/processed
SAU_IGNORE_HEADER_FAILURES	N
SAU_INTERFACE_USER	MTS_INTERFACE
SAU_IDENTIFIER	SAU*
SAU_LISTING_NAME	sau_files.1st
SAU_LISTING_SCRIPT_NAME	sau_files.ksh
SAU_REPROCESS_TIME_LIMIT	0.020833
SAU_UNIX_FILENAME	SAU_upload
SAU_UNIX_FILENAME_PATH	/usr/bin/hgp

The file to be uploaded is expected to have the file name format ?SAUXXXXXXXXXX.csv? where ?SAU? is the abbreviation of ?Scan and Unload? and ?XXXXXXXXXX? represents a sequence number.

For example:

	A	B	C	D	E	F	G	H
1	<RECH	MDCDMM	<RECH	HEADEREND>				
2	<XDOCK	15:56:36	15:56:54	20006457700001300	1	MSCABQ	<XDOCK	END>
3	<XDOCK	15:56:36	15:56:54	20006458410001300	1	MSCABQ	<XDOCK	END>
4	<EOF>							

Each ?Header? section will contain multiple records in the ?Detail? section for how many bookings were unloaded.

The header line of the file will contain the following information delimited with a comma:

Name	Type	Nullable	Position	Comments
OPENING_TAG	VARCHAR2(20)	N	1	i.e. <RECHEADERSTART>
DATABASE	VARCHAR2(10)	N	2	
FROM_LOCATION	VARCHAR2(25)	N	3	
CLOSING_TAG	VARCHAR2(20)	N	4	i.e. <RECHEADEREND>



The detail line(s) of the file will contain the following information delimited with a comma:

Name	Type	Nullable	Position	Comments
OPENING_TAG	VARCHAR2(20)	N	1	i.e. <XDOCKSTART>
START_UNLOADING_DATE	DATE	N	2	
START_UNLOADING_TIME	VARCHAR2(8)	N	3	
END_UNLOADING_TIME	VARCHAR2(8)	N	4	
BOOKING_REF	VARCHAR2(20)	N	5	
QTY_RECEIVED	NUMBER	N	6	
CURRENT_DEPOT	VARCHAR2(25)	N	7	
CLOSING_TAG	VARCHAR2(20)	N	8	i.e. <XDOCKEND>

3.2 File Processing - ?SAL? Message

The processing of the ?SAL? files will be changed to update the new table ?SCH_ORD_XDOCK? for cross docking for loading activities so that a comparison may be made with the data provided by the ?SAU? message. This table may be used to store information about the loading and unloading of materials at the cross-docking depot at different stages.

If the validation of the file format and items is successful then the data will be uploaded into a new table called ?SCH_ORD_XDOCK? via a function titled ?FN_ADD_SCH_ORD_XDOCK? in a new package titled ?DP_INT_TRK? (i.e. ?Database Package Interface Tracking?); the record ID and booking reference for the successful detail lines will be passed to this function so that the data may be read from the ?INT_SAL_DETAIL? table.

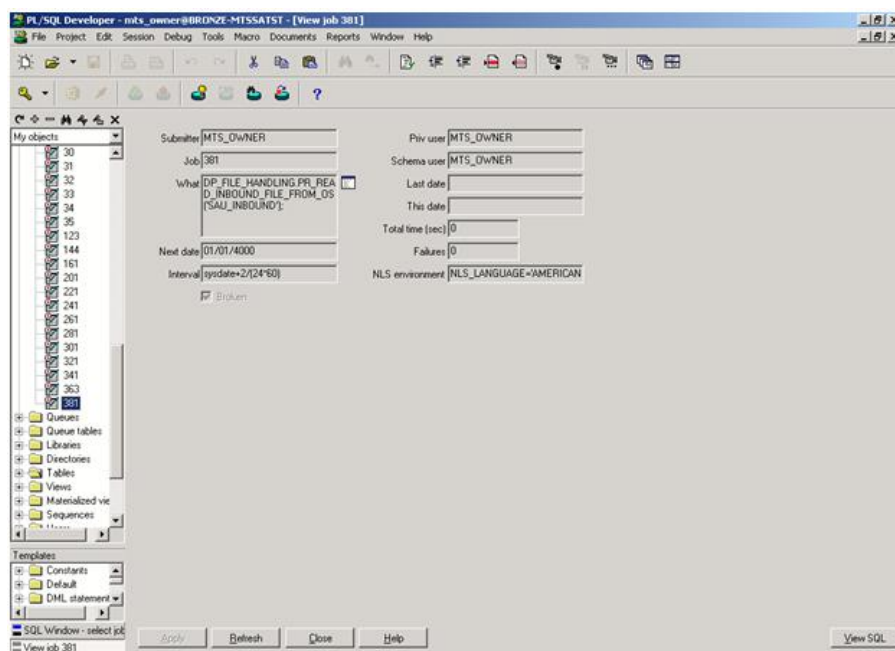
The details will be included on the new table ?SCH_ORD_XDOCK? as follows:

SCH_ORD_XDOCK	Column / Value	Table
BOOKING_REF	BOOKING_REF	INT_SAL_DETAIL
FROM_LOCATION	FROM_LOCATION	INT_SAL_DETAIL
OMS_REF	OMS_REF	INT_SAL_DETAIL
ACTIVITY	?L?	
TRIP_ID	TRIP_ID	INT_SAL_DETAIL
SCHED_NAME		
START_UNLOADING_DATE		
START_UNLOADING_TIME		
END_UNLOADING_TIME		
DU_QTY	DU_QTY	INT_SAL_DETAIL
WEIGHT		
VOLUME		
DU_TYPE		
CURRENT_DEPOT	NEXT_LOCATION	INT_SAL_DETAIL
STATUS	?SUCCESS?	
UPLOAD_TEXT	?Successful Load?	
REASON_CODE		
REASON_TEXT		
CREATED_DATE	System Date	
CREATED_BY	System User	
UPDATED_DATE		
UPDATED_BY		
ACTIONED	?N?	
ACTIONED_DATE		
ACTIONED_BY		



3.3 File Processing - ?SAU? Message

A database job will be created to process the ?SAU? files received, for example:



The procedure ?PR_READ_INBOUND_FILE? in the package ?DP_FILE_HANDLING? will be changed to process the ?SAU_INBOUND? file as required by the database job.

New procedures and functions in the package ?DP_INT_MSG? will be required to read and process the ?SAU? files received.

A procedure called ?READ_SAU_FILE? will be written to process the ?SAU? file and this procedure will call functions ?READ_SAU_HEADER? and ?READ_SAU_DETAIL?.

The procedure ?READ_SAU_HEADER? will validate the header line:

1. The file name must not be a duplicate
2. The file must have a header line
3. The file must be for the system database operation
4. The ?From Location? must be a valid active location

If an error has been detected then the status of the header record will be set to ?FAILURE? and the detail lines will not be read; otherwise, if the validation has passed then the status will be updated to ?NEW? and the detail lines read.

The procedure ?READ_SAU_DETAIL? will validate the detail lines:

1. The file must have a detail line
2. The booking reference must be valid
3. The booking reference must not be a duplicate in the file
4. The booking reference must have an order and the latest order for the booking reference must be present on a trip stop from the ?From Location? to the current depot
5. The start and end dates and times must be valid in the format ?YYMMDD? or ?HH24:MI:SS?
6. The quantity received must be a number greater than 0
7. The current depot must be a valid active location

If an error has been detected then the status of the detail record will be set to ?FAILURE?; otherwise, if the validation has passed then the status will be updated to ?SUCCESS?. Once all of the detail lines have been processed then the status of the header record will be updated to ?FAILURE?, ?PARTIAL? or ?PROCESSED? as appropriate for the detail lines. The number of detail lines that were processed successfully or that failed to process will be calculated and stored respectively in the columns ?NUM_SUCCESS? or ?NUM_FAILURE? on the table ?INT_SAU_HEADER?; the total number of detail lines scanned will be stored in the column ?NUM_SCANS? and should be the total of the other two



columns.

Appropriate error messages will be written at the header and detail levels for the validation performed.

The data received will enable order and trip details to be found for the booking reference provided and included on table ?INT_SAU_DETAIL?:

1. ?OMS_REF? will contain the latest order (i.e. the maximum value of ?OMS_REF? from table ?SCH_ORD?)
2. ?SCHED_NAME? will contain the schedule of the latest order
3. ?TRIP_ID? will contain the outbound trip from the ?From Location? to the current depot of the latest order
4. ?ACTION? will contain the text ?U? for ?Unload?
5. ?SOURCE_SYSTEM? will contain the text ?SAP?

The raw file details will be stored for display in the ?Interface Errors? screen, for this the procedure ?PR_READ_INBOUND_FILE? will call procedure ?PR_LOAD_FILE_INTO_ISH? for external directory ?SAU_INBOUND?; the raw file data will then be updated in column ?RAW_FILE? on table ?INT_SAU_HEADER? in procedure ?PR_LOAD_FILE_INTO_ISH? if the external directory is ?SAU_INBOUND?.

Once the ?SAU? file has been processed, the record ID and booking reference for the successful detail lines will be passed to this function so that the data may be read from the ?INT_SAU_DETAIL? table.

The details will be included on the new table ?SCH_ORD_XDOCK? as follows:

SCH_ORD_XDOCK	Column / Value	Table
BOOKING_REF	BOOKING_REF	INT_SAU_DETAIL
FROM_LOCATION	FROM_LOCATION	INT_SAU_DETAIL
OMS_REF	OMS_REF	INT_SAU_DETAIL
ACTIVITY	?U?	
TRIP_ID	TRIP_ID	INT_SAU_DETAIL
SCHED_NAME	SCHED_NAME	INT_SAU_DETAIL
START_UNLOADING_DATE	START_UNLOADING_DATE	INT_SAU_DETAIL
START_UNLOADING_TIME	START_UNLOADING_TIME	INT_SAU_DETAIL
END_UNLOADING_TIME	END_UNLOADING_TIME	INT_SAU_DETAIL
DU_QTY	QTY_RECEIVED	INT_SAU_DETAIL
WEIGHT		
VOLUME		
DU_TYPE		
CURRENT_DEPOT	CURRENT_DEPOT	INT_SAU_DETAIL
STATUS	?SUCCESS?, ?FAILURE? or ?WARNING?	
UPLOAD_TEXT		
REASON_CODE		
REASON_TEXT		
CREATED_DATE	System Date	
CREATED_BY	System User	
UPDATED_DATE		
UPDATED_BY		
ACTIONED	?N?	
ACTIONED_DATE		
ACTIONED_BY		

3.4 File Processing - Interface Tracking

A new package title ?DP_INT_TRK? (i.e. ?Database Package Interface Tracking?) will be created to write to the new table ?SCH_ORD_XDOCK? and to assess discrepancies for the ?Unload? process; the function ?FN_ADD_SCH_ORD_XDOCK? will be created for this purpose.



If a booking is being loaded via the ?SAL? message then no validation will be required; the ?Status? will be ?SUCCESS? and the ?Upload Text? will be ?Successful Load?.

If a booking is being unloaded via the ?SAU? message then validation will be required to determine whether the details for the unloading at the trip stop have any discrepancies with the loading of the trip for each order; for example, the order status, trip status and trip stop type will be checked for particular values and, if not as expected, they will create a warning; the status of the table ?SCH_ORD_XDOCK? will then correspond to the validation performed.

It will be expected that the status of the trip being unloaded at the cross-dock depot will be ?EN-ROUTE? because the trip is in progress, if the trip has a different status such as ?PLANNED? then a warning message will be issued because it cannot be guaranteed that the trip information is complete; however, this status would not prevent updates to the ?SCH_ORD_XDOCK? table but it will generate a warning message.

It will be expected that the trip stop is for cross-docking so the order will be unloaded but not at the delivery location; however, all booking references will be processed as unloaded at the cross-dock depot as advised.

A number of checks will be performed when the ?SAU? message is uploaded for comparison with previous stages of the order found and the following ?Upload Text? set:

1. ?Status is X and not EN-ROUTE for Trip? (i.e. the trip is not en route to the current depot)
2. ?Difference in Quantity for Order - Previous Quantity is X? (i.e. the previous quantity presumed loaded for the order is different to the quantity received)
3. ?Extra Quantity on the Trip for unknown Order X? (i.e. the order is in the ?SAU? file but is not on an inbound trip to the current depot)

Checks 1 and 2 will set the status to ?WARNING? and check 3 will set the status to ?FAILURE?; if no errors have occurred then the status will be set to ?SUCCESS?.

The previous quantity of the order may be taken in order of preference as follows:

1. Quantity in the ?SAL? message (?INT_SAL_DETAIL.DU_QTY?)
2. ?POC? quantity if the ?POC? flag has been set to ?Y?
(?SCH_ORDER_LINE.ACTUAL_DESPATCHED_QUANTITY?)
3. ?Planned? quantity (?SCH_ORDER_LINE.QUANTITY?)

How the previous quantity was found will be recorded in the text written to column ?UPLOAD_TEXT? for reference.

The ?WEIGHT?, ?VOLUME? and ?DU_TYPE? will be variables that will be set per trip event if possible:

1. ?WEIGHT? will be set to the planned weight (?SCH_ORDER_LINE.WEIGHT?)
2. ?VOLUME? will be set to the planned volume (?SCH_ORDER_LINE.CUBE?)
3. ?DU_TYPE? will be set to the DU type (?SCH_ORDER_LINE.DU_TYPE?)

Once the file has been validated and no errors have been detected then the trip stop found for the unloading will be updated with the start and end dates and times for the unloading.

The items ?START_UNLOADING_DATE?, ?START_UNLOADING_TIME? and ?END_UNLOADING_TIME? will be used to update the following items on table ?SCH_TRIP_STOP? for the trip stop:

Item	Column
START_UNLOADING_DATE +	ACTUAL_ARRIVE
START_UNLOADING_TIME	
START_UNLOADING_DATE +	ACTUAL_DEPART
END_UNLOADING_TIME	

An order audit message will be written when these dates and times are set.

N.B. If these dates and times are set then they will be fixed and the planned dates and times may not be recalculated for the trip stop.



3.5 Interface Errors

The ?Interface Errors? screen will be changed to include a new page titled ?SAU? located after the page titled ?SAL?.

The new page will include data at the header and detail levels as uploaded in the ?SAU? file and the data will be displayed in separate sections as follows:

Header (single record):

Name	Table	Column	Comments
Record ID	INT_SAU_HEADER	INT_RECORD_ID	
Msg Type	INT_SAU_HEADER	MSG_TYPE	
Status	INT_SAU_HEADER	RECORD_STATUS	
Success	INT_SAU_HEADER	NUM_SUCCESS	
Failure	INT_SAU_HEADER	NUM_FAILURE	
Created Date	INT_SAU_HEADER	CREATED_DATE	Format DD-MMM-YY HH24:MI:SS
Validation Error	INT_SAU_HEADER	VALIDATION_ERROR	

Detail (multiple records):

Name	Table	Column	Comments
Status	INT_SAU_DETAIL	RECORD_STATUS	
Action	INT_SAU_DETAIL	ACTION	
Schedule	INT_SAU_DETAIL	SCHED_NAME	
Trip	INT_SAU_DETAIL	TRIP_ID	
Order	INT_SAU_DETAIL	OMS_REF	
Start Date	INT_SAU_DETAIL	START_UNLOADING_DATE	Format DD-MMM-YY
Start Time	INT_SAU_DETAIL	START_UNLOADING_TIME	Format HH24:MI:SS
End Time	INT_SAU_DETAIL	END_UNLOADING_TIME	Format HH24:MI:SS
Booking Ref	INT_SAU_DETAIL	BOOKING_REF	
Qty Rec	INT_SAU_DETAIL	QTY_RECEIVED	
From Location	INT_SAU_DETAIL	FROM_LOCATION	
Current Depot	INT_SAU_DETAIL	CURRENT_DEPOT	
Validation Error	INT_SAU_DETAIL	VALIDATION_ERROR	

The raw file details will be obtained from the column ?RAW_FILE? on table ?INT_SAU_HEADER? via the function ?FN_READ_CLOB_ISH? in package ?DP_FILE_HANDLING? for when the record name contains the text ?SAU?; if the record name contains the text ?SAL? then the raw file details will be obtained from the table ?INT_SAL_HEADER?. The same logic will apply to the procedures ?PR_CLEAR_CLOB_ISH? and ?PR_APPEND_CLOB_ISH? in package ?DP_FILE_HANDLING?.

A button titled ?Search? will be available in the ?Header? section to allow the user to select records in a popup screen using the following criteria:

- Record ID
- Message Type
- Header Status
- From Location
- Created Date From
- Created Date To
- Booking Reference
- Order
- Trip ID
- Detail Status

For the ?Header Status? and ?Detail Status? fields there will be a drop-down list of the appropriate values (e.g. ?FAILURE?, ?FAILURE_RR?, ?SUCCESS?, ?REPROCESSD?, ?ACTIONED?, ?NEW? or ?ALL?).

The created date fields will be in the format ?DD/MM/YY?.



A button titled ?Reprocess? will be available in the ?Header? section for reprocessing, however, a file may only be reprocessed if the user has access to the function ?SAP_REPROCESS_BOOKINGS?.

If the file is reprocessed then the status will be updated to ?REPROCESSD? to prevent further amendments to the file. The reprocessed record will have the same file name format but with a suffix (_rpn) where ?n? represents a sequence number for the number of times the file has been reprocessed. The file will be obtained from the archive directory and placed in the inbound directory to be reprocessed by the database job. See ?Section 3.6? for further details regarding file resending.

A double click in the ?Record ID? field in the ?header? section will display the raw file data plus messages related to the upload; the raw file data will be displayed on the canvas ?CAN_RAW_FILE? for new data block ?INT_SAU_HEADER? in the ?INT_ERR? form.

The details displayed in the ?Interface Errors? screen may not be changed, however, if the status of the record in the ?Header? section is ?Failure? or ?Partial? then the record may be edited via a right-click option.

A screen titled ?Edit Scan & Unload Detail? will be called in which details uploaded may be amended for reprocessing; a new canvas titled ?CAN_SAU_EDIT? will be created in the ?INT_ERR? form; the changes may then be saved or cancelled via buttons titled ?Save, Exit? or ?Cancel, Exit?.

The edit screen will display the following items in the ?Header? section:

Name	Table	Column	Editable
Record ID	INT_SAU_HEADER	INT_RECORD_ID	N

The edit screen will display the following items in the ?Detail? section:

Name	Table	Column	Editable
Status	INT_SAU_DETAIL	RECORD_STATUS	N
Action	INT_SAU_DETAIL	ACTION	N
Booking Ref	INT_SAU_DETAIL	BOOKING_REF	N
Schedule	INT_SAU_DETAIL	SCHED_NAME	N
Trip	INT_SAU_DETAIL	TRIP_ID	N
Order	INT_SAU_DETAIL	OMS_REF	N
Start Date	INT_SAU_DETAIL	START_UNLOADING_DATE	Y
Start Time	INT_SAU_DETAIL	START_UNLOADING_TIME	Y
End Time	INT_SAU_DETAIL	END_UNLOADING_TIME	Y
Qty Rec	INT_SAU_DETAIL	QTY_RECEIVED	Y
Current Depot	INT_SAU_DETAIL	CURRENT_DEPOT	N

Validation will exist to ensure that the dates and times entered are valid and that the quantity received is a number greater than 0.

If the button ?Save, Exit? is pressed then any changes made will be saved and an updated file created for reprocessing by the database job. The changes made may then be viewed in the ?Detail? section of the ?Interface Errors? screen. The action performed will be recorded in the ?COMMENTS? column on the ?RESEND? tables.

The status of the record will be updated to ?FAILURE_RR? to indicate that the file has been re-sent for processing and is present in the inbound directory; the amended data will then be used to create a new file in the same format to upload

The reprocessed record will have the same file name format but with a suffix (_n) where ?n? represents a sequence number for reprocessing.

3.6 File Resending

New tables called ?INT_SAU_HEADER_RESEND? and ?INT_SAU_DETAIL_RESEND? will be required for the edited data; these new tables will include columns for all of the relevant data contained in the ?SAU? files plus columns to identify the records and to enable reprocessing.

If the status of the header record is not ?FAILURE? or ?PARTIAL? then the reprocessing will not be allowed because there will be no data to reprocess; the reprocessing will only process the detail lines that exist at status ?FAILURE? and the header status will be updated accordingly.



Therefore only files that failed to upload fully may be reprocessed and the files in the directory for failed uploads will be those files that have at least one detail line that failed to upload.

3.7 Interface Cross Docking Enquiry

If the validation of the file format and items is successful then the data will be uploaded into a new table called ?SCH_ORD_XDOCK?.

The ?BOOKING_REF?, ?FROM_LOCATION?, ?OMS_REF?, ?TRIP_ID? and ?ACTIVITY? will form a unique primary key on the table.

A new form titled ?INT_XDOCK_ENQ? (i.e. screen ?Interface Cross Docking Enquiry?) will be created to display the data contained on table ?SCH_ORD_XDOCK? and to allow the user to record whether any discrepancies recorded have been assessed; the user may set the flag ?ACTIONED? to ?Y? and enter a description of the action in the column ?Reason Text? and a ?Reason Code?; once the record has been ?ACTIONED? then no more changes will be allowed in the screen.

A button titled ?Search? will exist to enter selection criteria to access the data for the unloading performed via the ?SAU? message. Buttons titled ?Save? and ?Close? will exist to save any changes made and to close the form; if the form is closed after data has been changed but not saved then a prompt will be issued to confirm whether the changes should be saved or cancelled.

The data may be obtained via the optional entry of the following fields: ?Current Depot?, ?Schedule?, ?From Location?, ?Booking Ref?, ?Order?, ?Trip?, ?Activity?, ?Status?, ?Actioned?, ?From Created Date? and ?To Created Date?; a search screen will be available to specify these values; the ?Status? will be limited to ?SUCCESS?, ?FAILURE? or ?WARNING?; the ?Action? will be limited to ?Y?, ?N? or ?ALL?.

A list of values will be available to select the following fields: ?Current Depot?, ?Schedule?, ?From Location?, ?Booking Ref? (for prefixes, e.g. ?45?) and ?Trip? (for a specific schedule).

The screen will display the following fields in two adjacent columns with the ?Load? activity on the left and the ?Unload? activity on the right for the combination of ?Booking Ref?, ?From Location?, ?Order? and ?Trip?:

Name	Table	Column	Editable
Current Depot	SCH_ORD_XDOCK	CURRENT_DEPOT	N
Schedule	SCH_ORD_XDOCK	SCHED_NAME	N
Booking Ref	SCH_ORD_XDOCK	BOOKING_REF	N
From Location	SCH_ORD_XDOCK	FROM_LOCATION	N
Order	SCH_ORD_XDOCK	OMS_REF	N
Trip	SCH_ORD_XDOCK	TRIP_ID	N
Activity Start	SCH_ORD_XDOCK	START_UNLOADING_DATE/TIME	N
Activity End	SCH_ORD_XDOCK	END_UNLOADING_TIME	N
DU Qty	SCH_ORD_XDOCK	DU_QTY	Y
Weight	SCH_ORD_XDOCK	WEIGHT	Y
Volume	SCH_ORD_XDOCK	VOLUME	Y
DU Type	SCH_ORD_XDOCK	DU_TYPE	N
Status	SCH_ORD_XDOCK	STATUS	N
Upload Text	SCH_ORD_XDOCK	UPLOAD_TEXT	N
Reason Code	SCH_ORD_XDOCK	REASON_CODE	Y
Reason Text	SCH_ORD_XDOCK	REASON_TEXT	Y
Created Date	SCH_ORD_XDOCK	CREATED_DATE	N
Created By	SCH_ORD_XDOCK	CREATED_BY	N
Updated Date	SCH_ORD_XDOCK	UPDATED_DATE	N
Updated By	SCH_ORD_XDOCK	UPDATED_BY	N
Actioned	SCH_ORD_XDOCK	ACTIONED	Y
Actioned Date	SCH_ORD_XDOCK	ACTIONED_DATE	N
Actioned By	SCH_ORD_XDOCK	ACTIONED_BY	N

The ?Upload Text? and ?Reason Text? fields will require a scrollbar to display the data due to their lengths.



If the ?Actioned? flag is set to ?Y? then the ?Actioned By? field will be set to the system user and the ?Actioned Date? field will be set to the system date and time when the changes are saved.

The ?DU Qty?, ?Weight? and ?Volume? recorded for the items unloaded may be entered in this screen.

A ?Reason Code? and ?Reason Text? may be entered to record why any changes were made or if the details were accepted.

Any changes made and saved will update the ?Updated By? field to the system user and the ?Updated Date? field to the system date and time.

Table Updates Required

The following tables will be added to the database:

INT_SAU_HEADER:

Name	Type	Nullable	Default	Storage	Comments
INT_RECORD_ID	VARCHAR2(100)	N			
MTS_OPERATION	VARCHAR2(10)	Y			
MSG_TYPE	VARCHAR2(12)	Y			
RECORD_STATUS	VARCHAR2(10)	Y			
NUM_SCANS	NUMBER	Y			
NUM_SUCCESS	NUMBER	Y			
NUM_FAILURE	NUMBER	Y			
VALIDATION_ERROR	VARCHAR2(2000)	Y			
CREATED_DATE	DATE	Y			
CREATED_BY	VARCHAR2(40)	Y			
UPDATED_DATE	DATE	Y			
UPDATED_BY	VARCHAR2(40)	Y			
RAW_FILE	CLOB	Y			

INT_SAU_DETAIL:

Name	Type	Nullable	Default	Storage	Comments
INT_RECORD_ID	VARCHAR2(100)	Y			
REC_NUM	NUMBER	Y			
RECORD_STATUS	VARCHAR2(10)	Y			
SCHED_NAME	VARCHAR2(12)	Y			
FROM_LOCATION	VARCHAR2(25)	Y			
TRIP_ID	VARCHAR2(12)	Y			
OMS_REF	VARCHAR2(12)	Y			
START_UNLOADING_DATE	DATE	Y			
START_UNLOADING_TIME	VARCHAR2(8)	Y			
END_UNLOADING_TIME	VARCHAR2(8)	Y			
BOOKING_REF	VARCHAR2(20)	Y			
QTY_RECEIVED	NUMBER	Y			
CURRENT_DEPOT	VARCHAR2(25)	Y			
VALIDATION_ERROR	VARCHAR2(2000)	Y			
ACTION	VARCHAR2(1)	Y			
SOURCE_SYSTEM	VARCHAR2(25)	Y			
CREATED_DATE	DATE	Y			
CREATED_BY	VARCHAR2(40)	Y			
UPDATED_DATE	DATE	Y			
UPDATED_BY	VARCHAR2(40)	Y			

INT_SAU_HEADER_RESEND:



Name	Type	Nullable	Default	Storage	Comments
INT_RECORD_ID	VARCHAR2(100)	N			
MTS_OPERATION	VARCHAR2(10)	Y			
MSG_TYPE	VARCHAR2(12)	Y			
FROM_LOCATION	VARCHAR2(25)	Y			
CREATED_DATE	NUMBER	Y			
CREATED_BY	VARCHAR2(40)	Y			
COMMENTS	VARCHAR2(2000)	Y			
RESEND_ERROR_MESSAGE	VARCHAR2(2000)	Y			
RESEND_PROCESSED	VARCHAR2(1)	Y			?N?

INT_SAU_DETAIL_RESEND:

Name	Type	Nullable	Default	Storage	Comments
INT_RECORD_ID	VARCHAR2(100)	Y			
REC_NUM	NUMBER	Y			
RECORD_STATUS	VARCHAR2(10)	Y			
SCHED_NAME	VARCHAR2(12)	Y			
TRIP_ID	VARCHAR2(12)	Y			
OMS_REF	VARCHAR2(12)	Y			
START_UNLOADING_DATE	DATE	Y			
START_UNLOADING_TIME	VARCHAR2(8)	Y			
END_UNLOADING_TIME	VARCHAR2(8)	Y			
BOOKING_REF	VARCHAR2(20)	Y			
QTY_RECEIVED	NUMBER	Y			
CURRENT_DEPOT	VARCHAR2(25)	Y			
COMMENTS	VARCHAR2(2000)	Y			
CREATED_DATE	NUMBER	Y			
CREATED_BY	VARCHAR2(40)	Y			

SCH_ORD_XDOCK:

Name	Type	Nullable	Default	Storage	Comments
BOOKING_REF	VARCHAR2(20)	N			
FROM_LOCATION	VARCHAR2(25)	N			
OMS_REF	VARCHAR2(12)	N			
ACTIVITY	VARCHAR2(1)	N			
TRIP_ID	VARCHAR2(12)	N			
SCHED_NAME	VARCHAR2(12)	N			
START_UNLOADING_DATE	DATE	Y			
START_UNLOADING_TIME	DATE	Y			
END_UNLOADING_TIME	DATE	Y			
DU_QTY	NUMBER(24,4)	Y			
WEIGHT	NUMBER(12,2)	Y			
VOLUME	NUMBER(20,5)	Y			
DU_TYPE	VARCHAR2(12)	Y			
CURRENT_DEPOT	VARCHAR2(25)	N			
STATUS	VARCHAR2(7)	Y			
UPLOAD_TEXT	VARCHAR2(2000)	Y			
REASON_CODE	VARCHAR2(2)	Y			
REASON_TEXT	VARCHAR2(2000)	Y			
CREATED_DATE	DATE	Y			
CREATED_BY	VARCHAR2(40)	Y			
UPDATED_DATE	DATE	Y			
UPDATED_BY	VARCHAR2(40)	Y			



ACTIONED	VARCHAR2(1)	Y
ACTIONED_DATE	DATE	Y
ACTIONED_BY	VARCHAR2(40)	Y

References

Ref No	Document Title & ID	Version	Date
1	EST-272098 SA-7Y5ARG Update Trip Info on S&L Receipt v5.0.doc	5.0	29/06/10

Document History

Version	Date	Status	Reason	Initials
0.1	05/08/10	Draft	Initial version	PDR
1.0	18/08/10	Issue	Reviewed and Issued	MJC
1.1	18/08/10	Issue	Reviewed and Issued	PJG
1.2	27/09/10	Draft	Changes made as requested by Arif Marufi	PDR
2.0	01/11/10	Draft	Included trip update of stop unloading times in Section 3.4 as requested by Arif Marufi and updated quote for extra development	PDR
2.2	18/01/11	Draft	Change to start date format in SAU message to ?YYMMDD? as advised by Arif Marufi	PDR



4 AUTHORISED BY

<i>Matt Crisford</i>	Development Manager
<i>Peter Greer</i>	TMSCC MTS Product Manager

