

SSM Flex Rewrite
13.3 Agent Takeover Checklist
Screen Details

REVISION HISTORY

Version	Date Updated	Revision Author	Brief Description of Changes
1.0	8/11/2011	Paul Medcalf	Creation
1.1	11/8/2011	Linda Reyes	Updated message copy and screenshots
1.2	03/13/2012	Amar Mattey	Defect fixes 7898, graphic updates, ICE clean up notes
1.3	4/29/13	KC Painter	Updated screenshot and message properties

NOTES

OPEN QUESTIONS

1. None

ART STATUS

1. Art Approved

CHANGES FROM LEGACY SSM

1. Creative and copy changes

DEVELOPER NOTES

1. The Flex table needs to be modified to accommodate a table that's height is based on a specific number instead of a specific number of rows. Add an attribute called maxRowGridHeight, in configureScrollPolicy ignore row calculations when maxRowGridHeight is specified and use it as the height.
2. A function for created a hash where each element contains a collection needs to be created in the ConversionUtility. A function that creates a collection where each element contains a hash already exists.
3. Flex will sort the passengers by Last Name, First Name.
4. Java will supply the tasks per passenger sorted in the payload to Flex according to the Task Order shown in the Task Table below.

PAYOUT FROM JAVA

NEW_STATE = AgentTakeoverChecklist

SELECTED_PASSENGER = <PNRId>

This parameter is used to auto-scroll the list of passengers. Defines the PNRid of passenger to show at top of screen. Only applicable when scroll bar is present.

If attribute is not provided (e.g., on initial display) or if scroll bar is not present, no special positioning is needed.

This is used to simplify agent processing of tasks for each passenger as they work down the list.

ATC_PAX_<PNR_ID>_TASK_<N> = <Agent Task>

Possible values of <Agent Task> are listed below in the Task Table.

Java will supply these tasks in the order they are to be displayed per passenger.
Specifying this name-value pair by itself causes all of the buttons in the associated passenger row for the specified action to be enabled and unselected, which is the default button behavior.

PNR_ID is the ID of the affected passenger

N a 0 based number representing the index of this action.

Examples:

```
ATC_PAX_01.01_TASK_0 = VIEW_RESIDENT_CARD  
ATC_PAX_01.01_TASK_1 = VIEW_PASSPORT_NAME
```

ATC_PAX_<PNR_ID>_PARAM_<N> = <parameter value>

An optional attribute that provides a text value to be substituted into certain agent task descriptions.

PNR_ID is the ID of the passenger

N is a 0 based number representing the index of the associated action.

For example the VERIFY_DATE_OF_BIRTH task displays the date of birth via the following:

```
ATC_PAX_01.01_TASK_0 = VERIFY_DATE_OF_BIRTH  
ATC_PAX_01.01_PARAM_0 = 01/01/1980
```

ATC_PAX_<PNR_ID>_STATE_<N> = <State>

An attribute that is used to either select the valid button, select the not valid button, disable both the valid and not valid buttons, or show all buttons as enabled but none selected. The values are as follows:

- passed – selects the valid button
- failed – selects the not valid button
- disabled – disabled both the valid and not valid button
- enabled – enable all but select none

Example:

```
ATC_PAX_01.01_TASK_0 = VERIFY_DATE_OF_BIRTH  
ATC_PAX_01.01_STATE_0 = passed
```

Task Table

Agent Task	Task Order	Task Description	Button 1	Button 2	Button 3
VERIFY_DATE_OF_BIRTH	0	...446 plus date of birth	Valid	Not Valid	View Data
VERIFY_PASSPORT_EXP_DATE	1	...373	Valid	Not Valid	View Timatic
VERIFY_PASSPORT_NAME	2	... 382	Valid	Not Valid	<none>
VERIFY_MANUAL_PASSPORT_NAME	4	... 374	Valid	Not Valid	View Data
VERIFY_RESIDENT_CARD_EXP_DATE	5	... 390	Valid	Not Valid	View Timatic
VERIFY_RESIDENT_CARD_NAME	6	... 391	Valid	Not Valid	<none>
VERIFY_REQD_TRAVEL_DOCS	8	... 365	Valid	Not Valid	View Timatic
ENTER_VISA_INFO	9	... 375	Valid	Not Valid	Enter Data
ENTER_VISA_ISSUE_DATE	10	... 376	Valid	Not Valid	Enter Data

Example:

```
ATC_PAX_01.01_TASK_0 = VERIFY_PASSPORT_EXP_DATE  
ATC_PAX_01.01_TASK_1 = VERIFY_PASSPORT_NAME
```

This payload would mean that there is one passenger that will have two lines in their row as shown below. Notice that both rows of buttons are enabled but neither Valid nor Not Valid is selected. This is because no ATC_PAX_<PNR_ID>_STATE_<N> attribute was specified for either task.

EVENTS AND PAYLOAD TO JAVA

ANY BUTTON IN TASK TABLE

EVENT_NAME = atc_action

SELECTED_PAX = <PNR ID>

The ID of the passenger who's button was pressed.

ATC_TASK = <Agent Task>

This attribute indicates the agent task (row in the table) corresponding to a button that was pressed.

All possible <Agent Task> values are listed above in the Task Table.

ATC_ACTION = <Action>

This attribute indicates the button in the agent task row that was pressed by the user.

Possible <Action> values (from the Task Table) are:

"valid", "not_valid", "view_tematic", "view_data", and "enter_data"

NOTE: Any time a button is pressed within the table an atc_action event is sent to Java. For example if a Valid button is pressed, the payload is sent containing ATC_ACTION=valid. In the case that Java specifies a valid or not valid button to start being selected using the ATC_PAX_<PNR_ID>_STATE_<N> name-value pair, an event is not sent to Java unless the user presses another button for that task. This is because Java already knows whether a particular passenger action is passed or failed if they are specifying it in the initial payload to Flex.

For example if there is one passenger with two available tasks and Java specifies that the first task has a state of "passed," then that means that the first line in the passenger row will start with the "Valid" button as selected. The only time a payload would be sent to Java regarding that task would be if the user pressed the "Not Valid" button for that same task.

DISPLAY RULES AND FUNCTIONALITY

INITIAL SCREEN DISPLAY

The following components are displayed on this screen:

1. Each passenger has a row in the table, and each of those passenger rows can have multiple lines within it.
2. Each line in a passenger row is used to list the passenger name, the agent task to be performed, and a list of buttons.
3. The action values for a passenger determine what text and what buttons are displayed on the lines within a passenger's row, as specified in the Task Table.
4. The sets of passenger rows are sorted according to passenger last name and then passenger first name (if necessary). (Flex does this sorting.)
5. The tasks for a passenger are sorted according to the Task Order listed in the table above. (Java does this sorting.)

6. When a Not Valid button is pressed in a passenger's row, all of the other tasks in that row are marked as invalid ("Not Valid" buttons are selected). The payload is only sent for the Not Valid button that was pressed. Payloads do not need to be sent for the other not valid buttons that were automatically selected. Java needs to mark other tasks for the same passenger as Not Valid if one Not Valid button payload is sent.
7. When a Valid button is pressed, only the button that was pressed changes to selected state. Other buttons (Valid or Not Valid) retain their selected state.
8. When a View Data, Enter Data, or ViewTimatic button is pressed, the corresponding screen is displayed with information to be reviewed and/or changed for the corresponding passenger and agent task.

SELECTED PASSENGER DISPLAY

The SELECTED_PASSENGER attribute will be included in the payload for this screen upon re-display of the screen following display of secondary screens (via View Data, View Timatic, etc.). If the screen includes a scroll bar, the view area should be scrolled to show the passenger identified by the SELECTED_PASSENGER attribute. The SELECTED_PASSENGER should be displayed as close to the top of the view area as possible.

MESSAGE COPY

STATIC LABELS AND COPY

13_03_00_1	Document Verification
13_03_00_365	Verify all required travel documents
13_03_00_373	Verify passport expiration date
13_03_00_374	Verify that the name entered matches the name on the passport
13_03_00_375	Enter visa information
13_03_00_376	Enter visa expiration date
13_03_00_382	Verify name on passport
13_03_00_390	Verify U.S. Permanent Resident Card expiration date
13_03_00_391	Verify that the name on the U.S. Permanent Resident Card matches the passport
13_03_00_446	Verify date of birth

TABLE COLUMN LABELS

13_03_00_2257	Task List
13_03_00_2364	Task
13_03_00_2365	Action
13_03_00_1086	Passenger

BUTTON LABELS

13_03_00_2358	Valid
13_03_00_2359	Not Valid
13_03_00_2360	View Data
13_03_00_2361	View Timatic
13_03_00_2362	Enter Data

SCREEN DISPLAY UTILITY CONTEXTS

An SDU context must be defined to expose each unique screen layout and each unique string (for translation purposes) that can appear on the screen. The following are unique contexts that should be defined in the Screen Display Utility for this screen.

1. 3 Pax, Single and Multi Line
2. 1 Pax, 5 Actions, Some Disabled
3. 2 Pax, 3 and 2 Lines, Some Disabled
4. 3 Pax, Single and Multi Line, Some Disabled
5. DWC - 2 Pax, Verify date of birth
6. 9 pax, all possible tasks (scroll bar)

SCREEN DESIGN



Current Time
6:10 pm

Start Over

Document Verification

Task List

Passenger	Task	Action	
Bryan Helm	Verify name on passport Passport Number: Expires:	Valid	Not Valid
		Valid	Not Valid
Execplat Testfenix	Verify name on passport Passport Number: Expires:	Valid	Not Valid
Explatnum Testfenix	Verify that the name on the U.S. Permanent Resident Card matches the passport	Valid	Not Valid
	Verify name on passport Passport Number: Expires:	Valid	Not Valid

Current Time
10:37 am

Start Over

Document Verification

Task List

Passenger	Task	Action		
Alexandar Alvardo	Verify name on passport Passport Number: Expires:	<input checked="" type="checkbox"/> Valid	Not Valid	
	Verify that the name on the U.S. Permanent Resident Card matches the passport	Valid	Not Valid	
Tester Eight	Verify name on passport Passport Number: Expires:	Valid	<input checked="" type="checkbox"/> Not Valid	
	Verify that the name on the U.S. Permanent Resident Card matches the passport	Valid	Not Valid	
Tester Five	Enter visa expiration date	<input checked="" type="checkbox"/> Valid	Not Valid	Enter Data

