Forms-APEX Workflow
A SoftBASE A/S White Paper, August 2019

Introduction 2

Architecture Overview 2

Fig. 1 - Forms Value-upgrade integration functions - MM_USER_BUTTON_APEX 4
Fig. 2 - MM_USER_BUTTON_APEX - Code snippet. 4
Fig. 3 - User Client Machine - processes. 5
Fig. 4 - APEX Application Process INIT_P0_WJSI - After Authentication 5
Fig. 5 - APEX Application Process INIT_P0_WJSI - After Authentication 6

Integration Functions 6

Single Sign On - Forms and APEX. 6

Fig. 6 - Forms-APEX workflow - Single Sign On - Authentication 7

Forms-APEX workflow - Forms menu and navigation - after authentication 7

Fig. 7 - Forms-APEX workflow application navigation in both APEX and Forms. 8
Fig. 8 - Forms-APEX workflow application navigation - Selecting and entry and the form stack reacts immediately. 9

Single Sign On-Off - Forms and APEX. 9

Forms-APEX Deep linking. 9

Fig. 9 - The Right Click API - Deep Link to Apex 9
Fig. 10 - The Right Click API - Deep link to Apex - QA page opened in APEX on the order, which is passed from Forms to APEX 10
Fig. 11 - The Right Click API - Setup screen 10

Security 11

Fig. 12 - Forms-APEX workflow - Users and groups in Forms above and APEX below 11

Summary 11
Introduction

Forms-APEX Workflow is a ready made turnkey APEX application, library (pll), an object reference library (olb) for Forms. This enables a four-way integration between an existing Forms application and APEX. Four-way and not only bidirectional integration, delivers both push and pull of data, and events is supported both from Forms to APEX and APEX to Forms.

In this way, the Forms-APEX workflow adds value to your existing and future Forms applications as a client application. There is no need for any code change in your existing Forms application, but a pll and an olb will be added to your application.

The integration between Forms and APEX supports the FSAL deployment of Forms. It is based on the websocketjsi, which is a new javascript interface, that was released first time through Forms release 12.2.1.3.

You get the best of both worlds, the client usage in Forms, and the web usage in APEX. You can run them separately or together.

Architecture Overview

Forms Websocket Java Script Integration (WJSI) This feature requires the use of Eclipse/Jetty version 9.4.5, a third-party Java jar file. This Java jar file must be signed with a trusted and known certificate.

Using the Eclipse/Jetty server, a running Forms application can communicate to an HTML page through the Jetty server. The Jetty server is an extremely lightweight application server that has built-in support for Websocket technology and hosts a special intermediate application that assists in the communication between browser (APEX) and Forms application. This lightweight server is delivered to the user’s PC during application startup.

The Form is extended with a When-Custom-Javascript-Event trigger for handling all push-pull events from APEX.

The code is..

```sql
Begin
    SbS_Javascript_Event;
End;
```
In order to handle push-pull events to APEX a package in a library is provided, the SBS_WJSI_API, which has functions for handling the websocketjsi server from Forms. This library is added to the Forms application.

PACKAGE SBS_WJSI_API IS
/* The SBS_PORT is called upon application start, typically a FSAL launching of a forms Module in order to find the first available PORT on the users PC for the websocket Jetty server */
/* This is totally transparent for the user */
/* A separate Turnkey Form MMSAPEXLOGIN is provided for this purpose */
Function SBS_PORT(
    wf_mm_low in Number default 7000,
    wf_mm_high in Number default 7999) return Number;

/* A start of the websocket server, initiated from MMSAPEXLOGIN */
Procedure SBS_START_WJSI(wf_mm_port in Number);

/* A stop of the websocket server, initiated from MMSAPEXLOGIN */
Procedure SBS_STOP_WJSI;

/* A Function, that returns the status of the Websocket server, i.e. RUNNING or not RUNNING, initiated from MMSAPEXLOGIN */
Function SBS_STATUS_WJSI Return varchar2;

/* On a running server a peer to peer session is started. Every APEX page connects itself to this Session upon rendering. */
Procedure SBS_BEGIN_SESSION(
    wf_mm_port in Number,
    wf_mm_session in Varchar2 Default 'SBSwebsocketJsi');

/* MMSAPEXLOGIN can stop the session, typically upon on Logoff */
Procedure SBS_STOP_SESSION;

/* MMSAPEXLOGIN checks if a session is ACTIVE or NOT ACTIVE */
Function SBS_STATUS_SESSION return varchar2;

/* The right click API can initiate this SBS_Event from any form, and this provides a call to APEX with execution of a javascript function */
Procedure SBS_Event(
    wf_mm_return in Varchar2,
    wf_mm_event in Varchar2,
    wf_mm_payload in Varchar2);

END;

Additional procedure is added to the Forms application in order to support the Forms-APEX integration, for instance the MM_USER_BUTTON_APEX, which is a part of the Value-upgrade integration functions. However, no Forms code is changed.
The overall architecture is that the Forms application communicates to the Jetty websocket server (locally deployed on the PC) through a java package call the frmwebsocketjsi.jar. This requires no installation on the PC, not even the java client, as the client functionality is delivered through a setup of the Forms Standalone Launcher (FSAL). The Fsal is launched from a Desktop shortcut, which links to a script installed once on your LAN, servicing all PC’s.
The APEX client communicates with the Jetty websocket server through raising events with or without payload. The form either performs a function or returns data through its functionality and the eval-javascript function.

The components of the integration makeup a toolset, which is easily build into your APEX and Forms applications. In Forms, merely by applying the pll’s and the olb. In APEX by applying the js to your workspace, and to add 1 application process after login.

**Fig. 3 - User Client Machine - processes.**

**Fig. 4 - APEX Application Process INIT_P0_WJSI - After Authentication**
Thereto 1 Dynamic Action on Page 0 (Global Action) SBSRECONNECT in order to initiate the socket session on each page.

Integration Functions

The Forms-APEX Workflow consists of number of integration functions and possibilities, which are listed below.

Single Sign On - Forms and APEX.

The Forms-APEX Workflow is typically started from your desktop by a double click on a FSAL icon for your application. This FSAL icon will however start a FSAL setup which initially calls a form MMSAPEXLOGIN. This form performs the following;

1. Initiates the frmwebsocketjsi server on the user’s PC/Macbook/Linux PC.
2. Opens the Forms-APEX workflow application (1220), where you make a normal APEX login, using Forms-APEX workflow custom authentication scheme.
3. The Forms-APEX workflow application (1220) raises an event with the username and password as payload (hashed), for the MMSAPEXLOGIN form.
4. The MMSAPEXLOGIN form catches the event, resolves the payload, performs a form login and opens the starting form of your application.
Forms-APEX workflow - Forms menu and navigation - after authentication

The APEX application contains the total authorized navigation of the Forms application behind it. Thus, you can navigate to any form from APEX. A multiple choice in your navigation is a key element in a user-friendly workflow between Forms and APEX. Being able to do anything or everything from both APEX as well as Forms, makes the next step easy and intuitive.
Fig. 7 - Forms-APEX workflow application navigation in both APEX and Forms.

When selecting a menu entry in either Forms or APEX the form reacts immediately.
Single Sign On-Off - Forms and APEX.

The Single Sign On-Off function is supported out of APEX (where you logged in) and planned to be supported out of the form as well. In APEX, i.e. the Forms-APEX workflow application, which can be any application, when the work is done you perform a normal logout. The application performs:

1. An adjusted logout url in APEX
   javascript: sbs_apex_logout('&LOGOUT_URL.'
   The sbs_apex_logout js will recognise if you are running the application as standalone or together with Forms.
2. When running together with Forms, event is raised to the Forms stack, in order to perform a logout in Forms.
3. The APEX application performs a normal logout by issuing the logout url. The logout url is a part of the APEX MasterApp from SoftBASE and it redirects to the initial app as specified in the login URL.
   f?p=&CTR_BACKURL_APP.:LOGIN_DESKTOP:&SESSION.

Forms-APEX Deep linking.

You can perform a deep linking navigation from any form in your application, to any page in APEX, including passing parameters from the form application, like value of screen items. In this way you can use APEX for analysis or reporting as well as direct extension to your Forms application. This is done, by using the Right click API in your Forms application. The Right click API is added to your Forms application as a part of the Forms Value-upgrade.
Fig. 10 - The Right Click API - Deep link to Apex - QA page opened in APEX on the order, which is passed from Forms to APEX

The Right Click deep linking can be accessed from any screen and easily setup to your needs, also with multiple links from one screen.

Fig. 11 - The Right Click API - Setup screen

Figure 11 illustrates the Right click API to APEX. In command the application and the page are specified. Path specifies the URL (:::RIR,1:) means Reset Interactive Report (or Grid), clear cache on page 1-. IREQ_MM_ORDER_ID / :MM_ACTIVITIES.MM_ORDER_ID means
pass the content of the screen item :MM_ACTIVITIES.MM_ORDER_ID into Interactive Report (or Grid) filter for column MM_ORDER_ID.

Security

The Forms-APEX workflow share a common authentication and authorization, which makes user and group administration, including authorization as easy as it was in your Forms application. The security administration of your Forms application is kept unchanged.

Fig. 12 - Forms-APEX workflow - Users and groups in Forms above and APEX below

Summary

Forms-APEX workflow extends your existing well working Forms application with the best of modern tools, APEX. In your back office you will be able to use Forms-APEX workflow combined with APEX, and out of office you can use APEX directly on mobile devices, especially when combined with APEX MasterApp from SoftBASE (freeware).
Adding Forms-APEX workflow to your Forms application pays off immediately as the project is secure and maintains the already proven functionality of the Forms application. No code is changed. At the same time the best of the web is made directly available. Using the APEX MasterApp, you can decide to replicate some functionality in APEX, or to get an APEX version of that functionality in order to use it, in front office or out of office on mobile devices.