All about CBTA- How to create Test scripts, SDC and TCE; How test script runs..!!

**CBTA (Component Based Test Automation)** is a functionality of SAP Solution Manager where we can create test cases in modular structure. As the name suggests,CBTA is component based testing and there are 2 types of components namely,

                                              1. Screen Components

                                              2. Default Components

Screen components are nothing but the activity done while testing the system under test such as mouse click, entering transactions etc. Default components are the data entered at the time of recording such as values.

Some terminologies to be known are,

1. **SUT (System Under Test)**:

          SUT is the managed system which is to be tested. In SUT Management, the following details are maintained

* Technical RFC destination (looks like TST\_<SID>\_<CLNT>)
* Business user (i.e. Tester Profile)
* Base URL / HTTP destination in case of non-ABAP systems.

2. **SDC (System Data Container)**

          SDC defines the systems on which the automated test cases are recorded or executed. A system is defined as follows

* Operational function (eg: CRM,ERP)
* Technical role (eg: Development system, Production system)

3. **TDC (Test Data Container)**

          TDC is a central repository for all test data. It contains the information of parameters, attributes and variants.

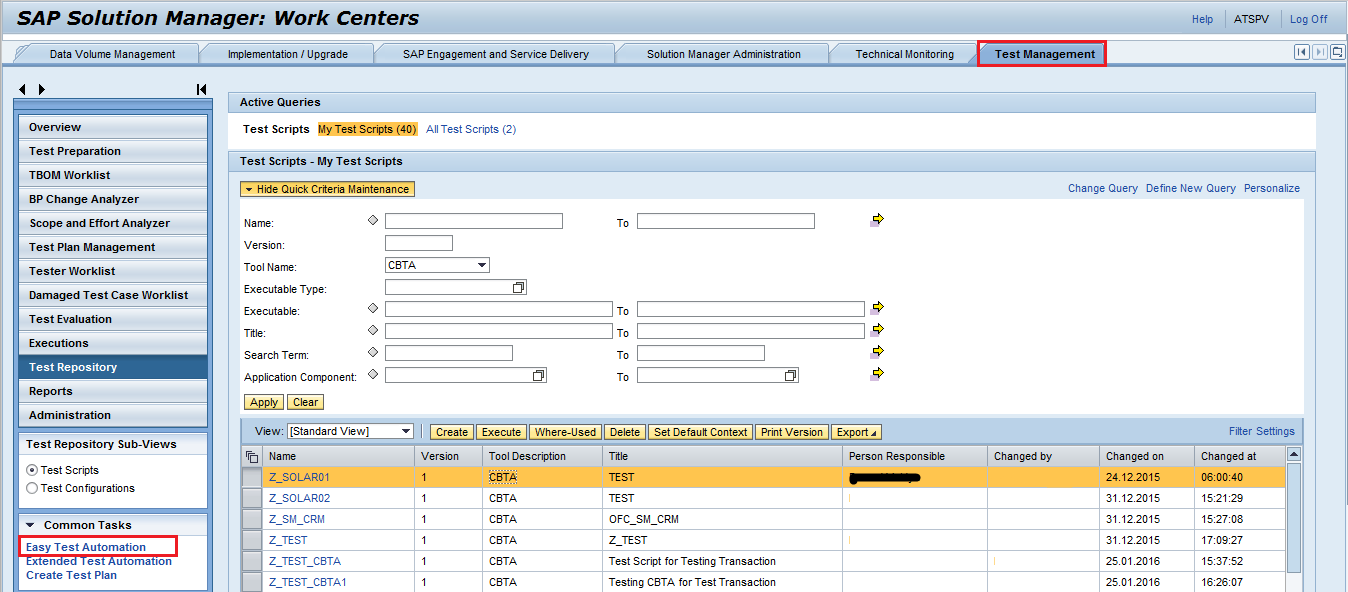
Now we will look how test scripts, SDC are created and see how automated testing is done with the help of SAP Solution Manager

**Prerequisites:**

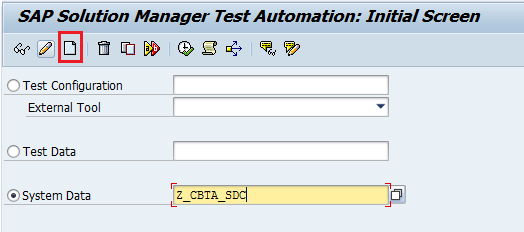
Perform system configuration steps as given in the guided procedure SOLMAN\_SETUP prior to create test script

**Note: Test cases can be created in configuration phase of each project via SOLAR02 transaction. But I am explaining how CBTA works via transaction SM\_WORKCENTER**

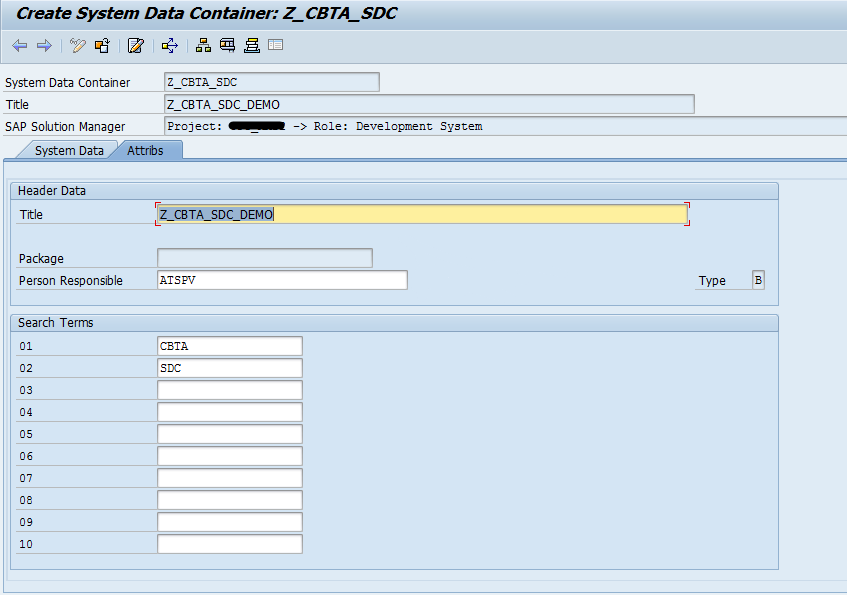
**Step 1:**Login into SAP Solution Manager system and run transaction SM\_WORKCENTER. Under Test Management workcenter, click Easy Test Automation as shown below

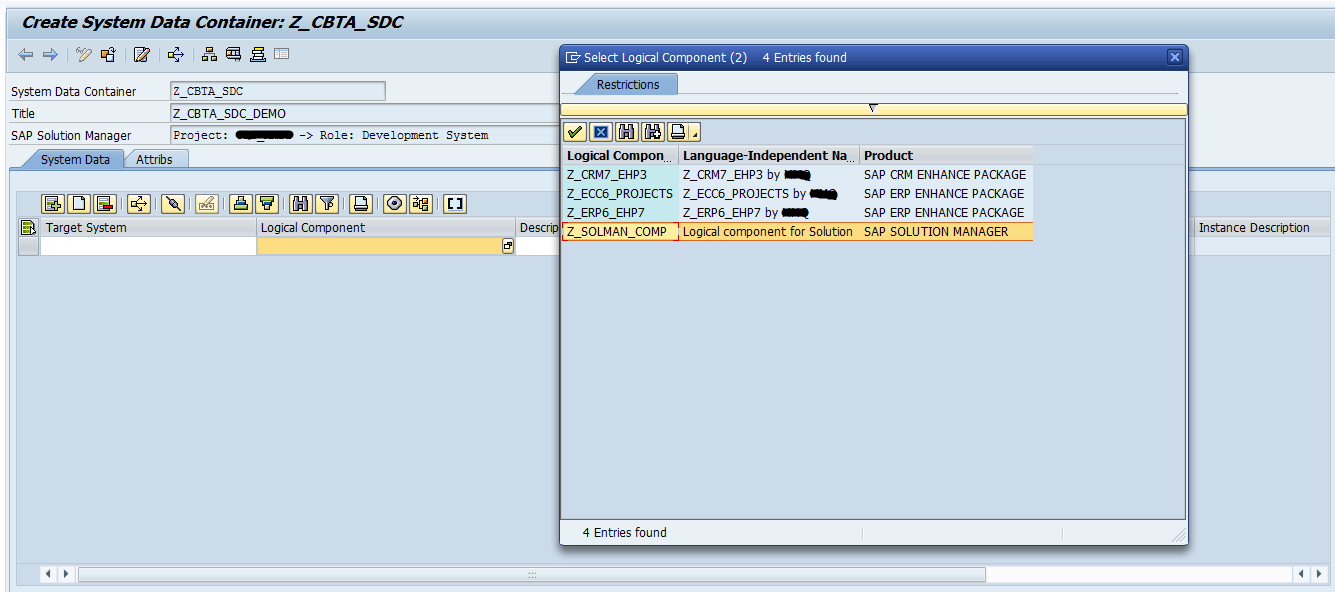
[](https://blogs.sap.com/wp-content/uploads/2016/07/1_996131.png)

**Step 2:**A pop up opens where you can create new SDC or use existing SDC. To create new SDC, select **System Data** and enter the SDC name and click create as shown below

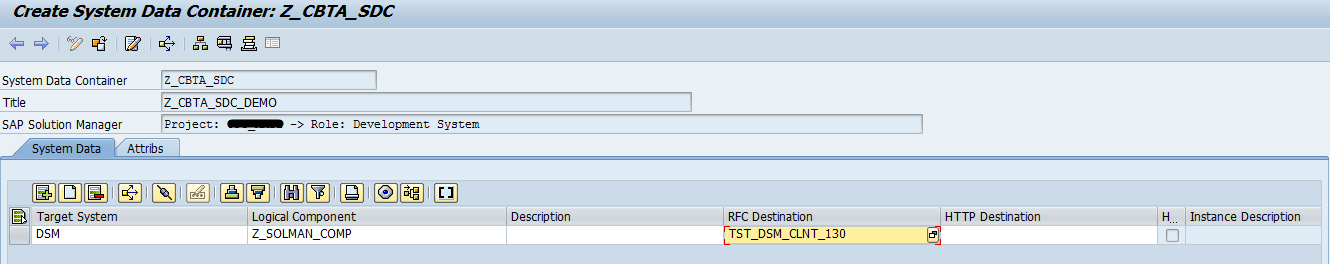
[](https://blogs.sap.com/wp-content/uploads/2016/07/2_996043.png)

**Step 3:**Then you will be directed to SDC creation window where we need to give the name of the System Data Container and specify the logical component, target system as shown in the below screenshots.

[](https://blogs.sap.com/wp-content/uploads/2016/07/3_996133.png)

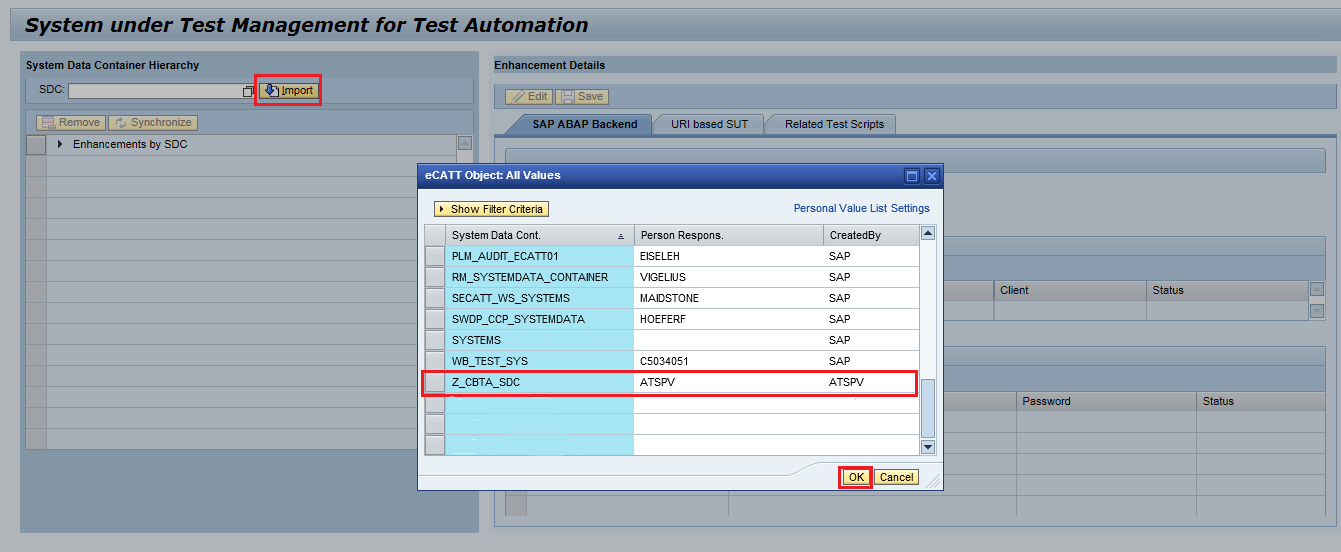
[](https://blogs.sap.com/wp-content/uploads/2016/07/4_996140.png)

After entering the details, upon saving the SDC contains the following information

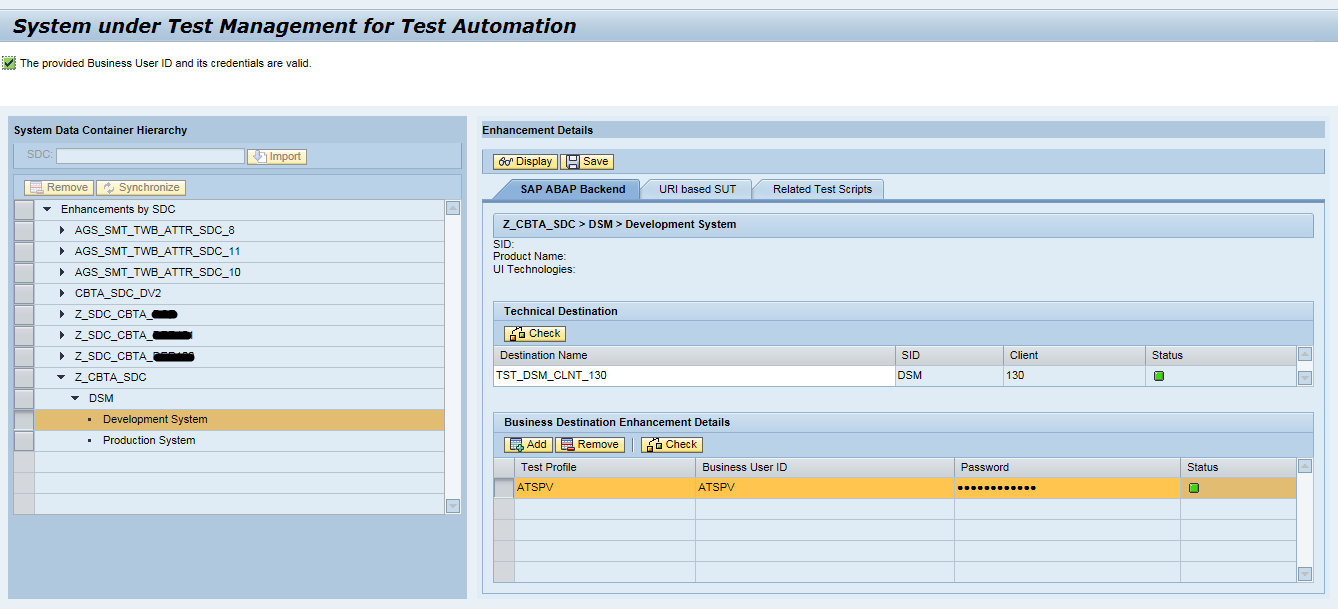
[](https://blogs.sap.com/wp-content/uploads/2016/07/5_996141.png)

Save these configuration in transport request.

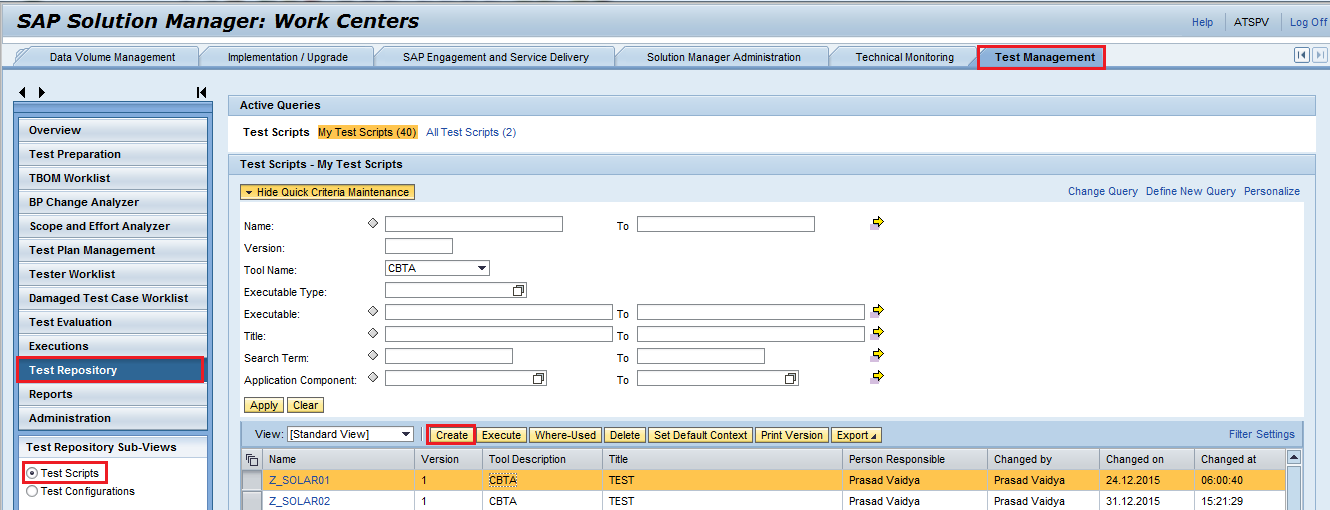
**Step 4:**Now, in Test Management workcenter, go to CBTA Settings-> Maintain SUT. Here, you need to import the SDC created as shown below

[](https://blogs.sap.com/wp-content/uploads/2016/07/6_996152.png)

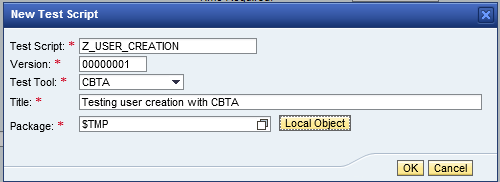
**Step 5:**After selecting and importing the SDC, now you need to specify the RFC that communicates with the SUT which is configured as prerequisite and also assign the tester profile so that it enables logon to managed system while the script runs.

[](https://blogs.sap.com/wp-content/uploads/2016/07/7_996154.png)

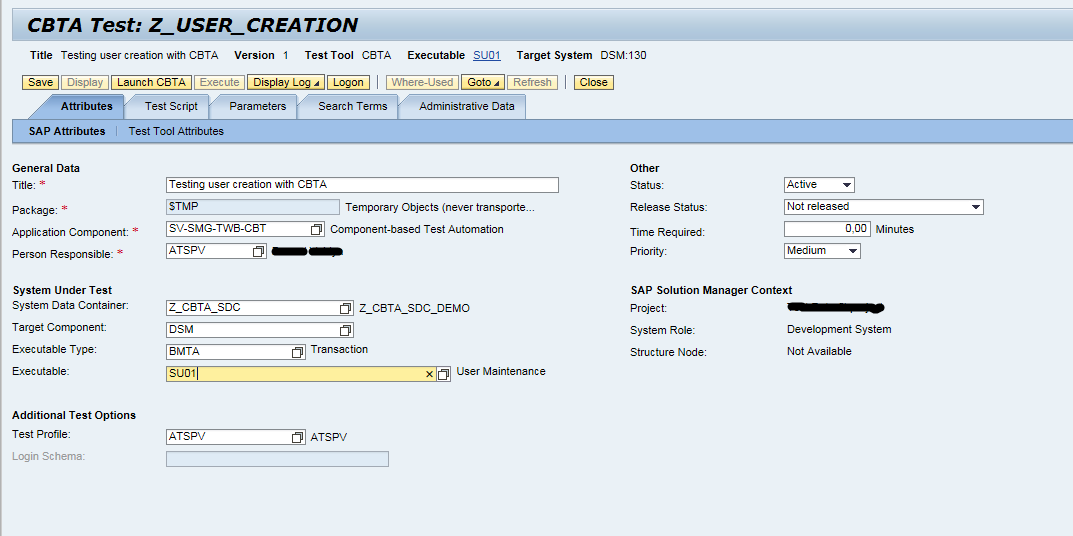
**Step 6:**Now, to create Test scripts, in Test Management workcenter-> Test Repository-> Test Scripts

[](https://blogs.sap.com/wp-content/uploads/2016/07/8_996167.png)

When you click create Test Scripts, a pop up will appear where you need to give the name and title of the script, test tool would be CBTA as shown below

[](https://blogs.sap.com/wp-content/uploads/2016/07/9_996170.png)

**Step 7:**Now, in the attributes tab of the test script, you need to specify the Application Component, here it will be CBTA component, SDC for this test script, Target system, Executable type and the Transaction as shown below

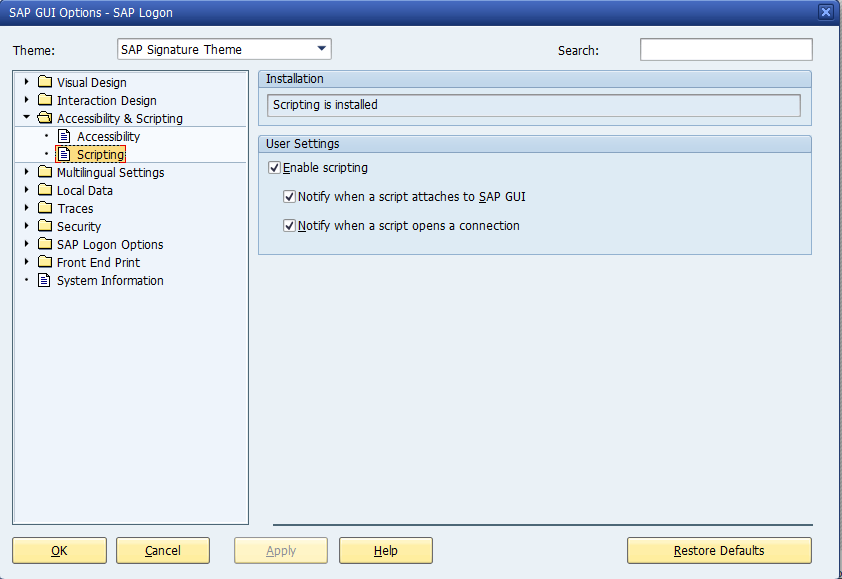
[](https://blogs.sap.com/wp-content/uploads/2016/07/10_996172.png)

**Note: Executable can be a transaction, BSP Application, CRM Webclient, Webdynpro application**

**Step 8:**Once the details are filled in step 7, now click Launch button as that the script runs in the managed system remotely.

**Note: In order to run the scripts remotely, settings in the managed system should be done. For this, run transaction SCC4 in corresponding managed system and keep the settings as changes are allowed.**

**Also you must enable scripts to run by making the changes in SAP Logon settings are shown below**

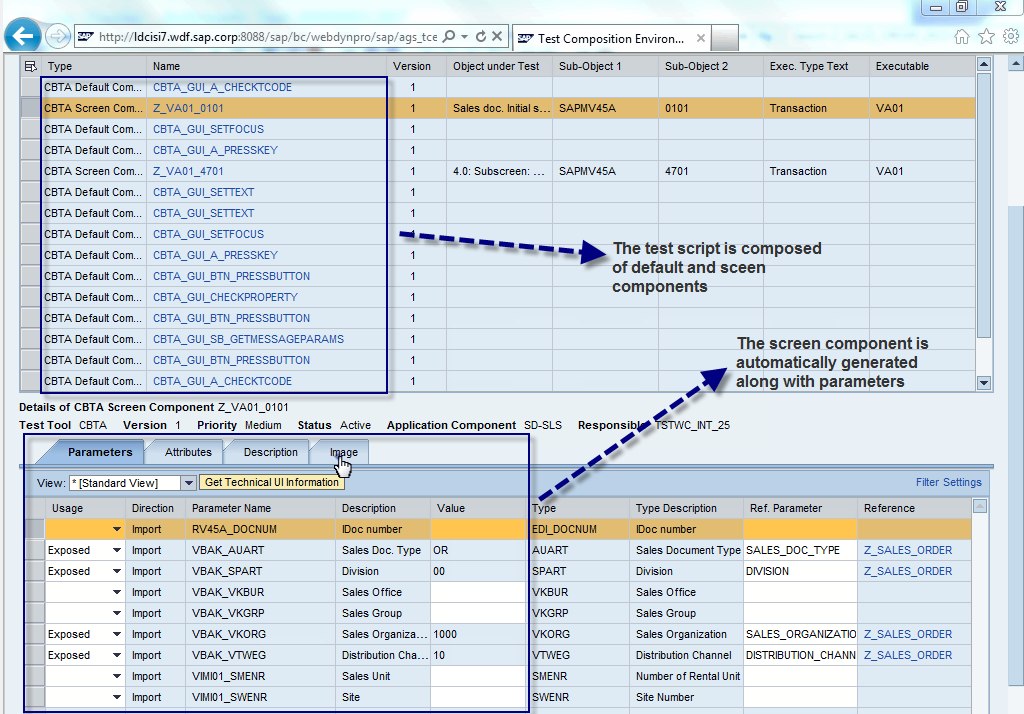
[](https://blogs.sap.com/wp-content/uploads/2016/07/11_996187.png)

So finally when you click on Launch CBTA, the scripts runs in the SUT and the test log is stored in SAP Solution Manager system.

# Component Based Test Automation (CBTA) in SAP Solution Manager 7.1

CBTA is a test automation tool present in SAP Solution Manager (SAP SolMan), and the usage right is available for all enterprise support customers. The tool helps in creation of modular test cases which are made of components. The user can record a transaction or a SAP web application as a normal business user, subsequently, CBTA would create a modular test in SAP Solution Manager.

So, what’s big deal? – The CBTA generated tests are made of default and screen components. The default components typically signify user actions e.g. pressing of enter key, navigation to menu item, etc while the screen components represent a screen or sub-screen which does the input for application under test. The screen components are easier to repair or recreate if the application under test changes, which leads to much easier maintenance.

[](https://blogs.sap.com/wp-content/uploads/2014/09/26_08_544645.png)

Many default components are pre-shipped by SAP, and it’s available in SAP Solution Manager, while the screen components are created automatically, when a particular screen is accessed while doing the recording. It’s also interesting to note that the screen components are created by accessing the DDIC from the System Under Test, and also the input fields are automatically parameterized. The process of the screen component creation is called Inspection.

The CBTA generated tests would have the following:

1. The script is made of components e.g. default and screen components

The CBTA generated test scripts can be seen in an environment called Test Composition Environment (TCE). TCE provides powerful environment not only to trigger test script creation, but makes edit of the test script easier in the following manner:

1. New default or screen components can be easily inserted
2. Sequence of the components can changed easily
3. Screen component can generated
4. Usage of parameters can be changed
5. Test Data Container assignment, etc.

CBTA was launched in 2012 in SAP SolMan 7.1 SP07, and it was supporting following UI technologies:

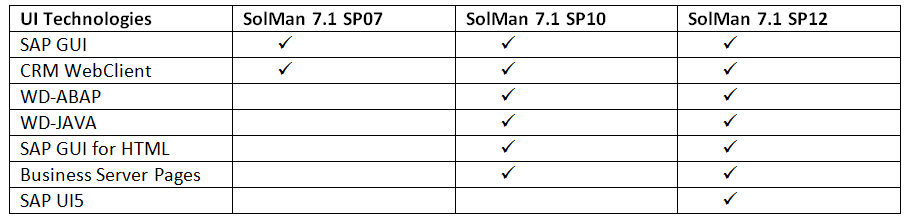
1. SAP GUI
2. CRM Web Client (Sales, Service and Marketing)

With SAP SolMan 7.1 SP10, CBTA started supporting automation of the following UI technologies over and above the ones already mentioned before:

1. WebDynPro ABAP (WD-ABAP)
2. WebDynPro JAVA (WD-JAVA)
3. SAP NetWeaver Portal (SAP NW Portal)
4. SAP GUI for HTML
5. Business Server Pages (HTML B controls)

With SAP SolMan 7.1 SP12, CBTA also started supporting automation of SAP UI5.  CBTA generated test for SAP UI 5 application is currently using only default components.

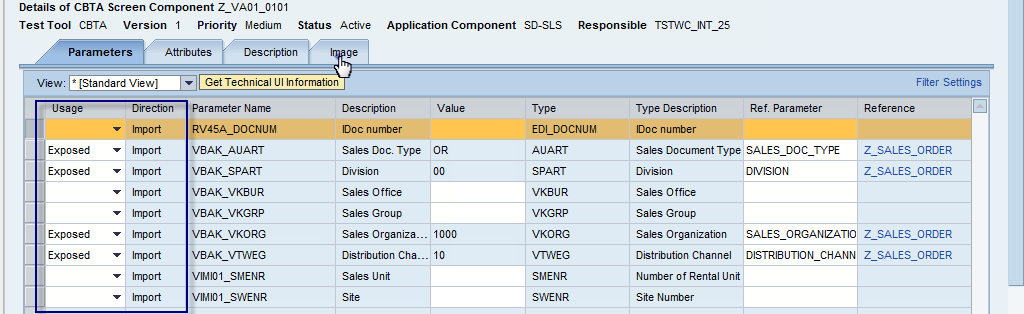
The above mentioned information can be simply put in the following table:

[](https://blogs.sap.com/wp-content/uploads/2014/09/19_09_544802.png)

The component used by CBTA has following parameters which are classified in the following:

1. Import parameter
2. Export parameter

The parameters are also classified on the basis of usage:

1. Exposed: The parameter can accept value from TDC
2. Fixed: As the name suggests,[](https://blogs.sap.com/wp-content/uploads/2014/09/26_08_02_544803.png)the value does not change.
3. Local: The parameter is used with in the script to send or receive values
4. Blank: not used at all e.g. many default components has parameter name: Option, which is reserved for future use, and not currently used by CBTA.

Generally, a suitable Export parameter is exposed, so that end-to-end test can be created.In the next post, I will explain several of ways of executing a single CBTA test and passing of multiple data iterations via a Test Data Container.