



SAP Testing

tutorialspoint

SIMPLY EASY LEARNING

www.tutorialspoint.com



<https://www.facebook.com/tutorialspointindia>



<https://twitter.com/tutorialspoint>

About the Tutorial

Many Organizations implement SAP ERP to manage their business operations efficiently. SAP Testing is about testing the functionality of various SAP modules and to ensure that they perform as per the configuration. This tutorial explains the basics of SAP testing.

Audience

This tutorial is meant for all those readers who want to learn complete end-to-end testing and validation of all SAP modules in SAP ERP environment.

Prerequisites

You need to have a basic understanding of the concepts of enterprise resource planning in order to make the most of this tutorial.

Disclaimer & Copyright

© Copyright 2016 by Tutorials Point (I) Pvt. Ltd.

All the content and graphics published in this e-book are the property of Tutorials Point (I) Pvt. Ltd. The user of this e-book is prohibited to reuse, retain, copy, distribute or republish any contents or a part of contents of this e-book in any manner without written consent of the publisher.

We strive to update the contents of our website and tutorials as timely and as precisely as possible, however, the contents may contain inaccuracies or errors. Tutorials Point (I) Pvt. Ltd. provides no guarantee regarding the accuracy, timeliness or completeness of our website or its contents including this tutorial. If you discover any errors on our website or in this tutorial, please notify us at contact@tutorialspoint.com

Table of Contents

About the Tutorial.....	i
Audience	i
Prerequisites	i
Disclaimer & Copyright.....	i
Table of Contents	ii
1. SAP TESTING – OVERVIEW.....	1
2. SAP TESTING – TYPES OF TESTING	4
3. SAP TESTING – MANUAL AND AUTOMATION TESTING.....	5
4. SAP TESTING – SDLC	6
Software Testing Life Cycle.....	7
5. SAP TESTING – TESTING TYPES IN SAP	11
Unit Testing.....	11
System Testing	11
Scenario-based Testing	11
Integration Testing	12
SAP Interface Testing	12
SAP User Acceptance Testing	12
SAP System Performance Testing	12
SAP Load Testing	12
SAP Security and Authorizations Testing	13
SAP Cutover Testing	13
SAP Regression Testing	13

6.	SAP TESTING – TESTING PROCESS	14
	Test Planning.....	14
	Test System Setup	14
	Test Execution and Evaluation	14
	SAP Testing – Critical Process	14
7.	SAP TESTING – NAVIGATION.....	16
8.	SAP TESTING – SCREEN FLOW	19
	Creating Screen Flow.....	19
	Testing Screens	20
9.	SAP TESTING – MODULES.....	21
10.	SAP TESTING – INTERFACES.....	23
11.	SAP TESTING – TEST-CASES.....	24
12.	SAP TESTING – TAO	27
	SAP TAO 2.0	27
	SAP TAO Process Architecture	28
	Prerequisites for Installing SAP TAO	29
13.	SAP TAO – FOLDER STRUCTURE.....	30
14.	SAP TESTING – CONFIGURE SAP QUALITY CENTER	33
	HP Server Attributes	33
	HP Project	33
15.	SAP TAO – RESULTS ANALYSIS	38
16.	SAP TAO – BUILD & EXECUTE TEST SCRIPT	40
17.	SAP TAO – CONSOLIDATE	41

18. SAP TAO – UI SCANNER	42
19. SAP TESTING – AUTOMATION TOOLS	43
20. SAP TESTING – SOLUTION MANAGER	44
21. SAP TESTING – HP BUSINESS PROCESS TESTING	46
HP Quality Center (ALM)	47
22. SAP TESTING – QTP	49
23. SAP TESTING – CASE STUDY	51
24. SAP TESTING – INTERVIEW QUESTIONS	52

1. SAP Testing – Overview

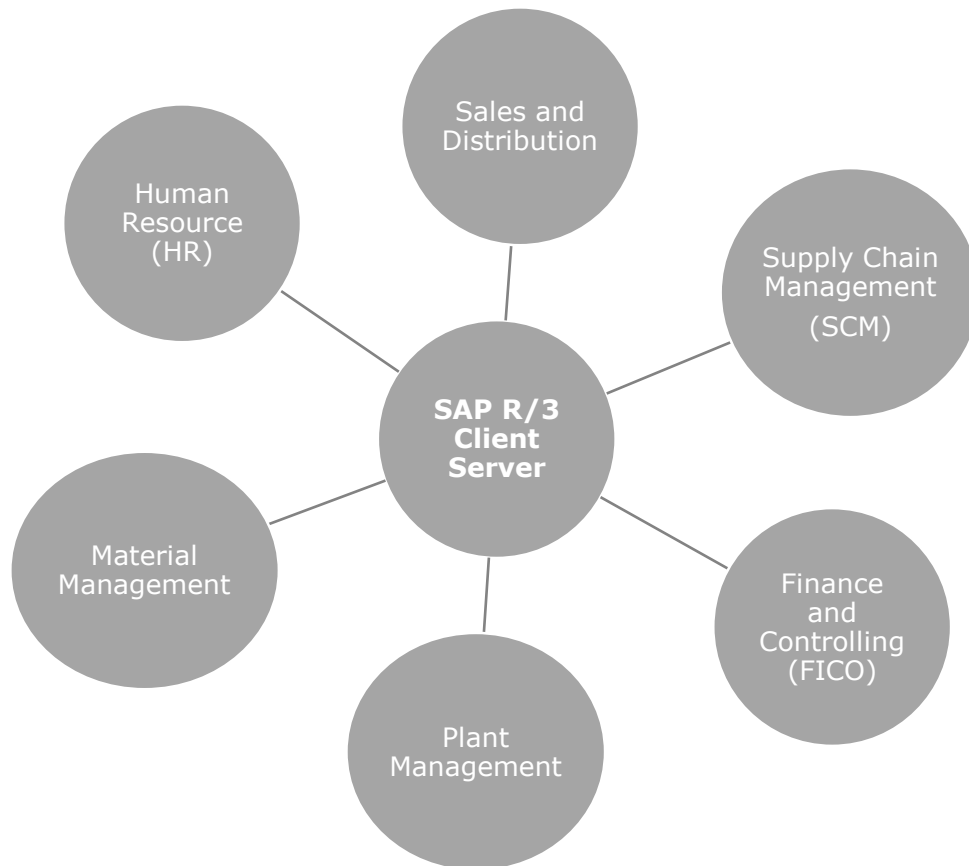
Many organizations implement SAP ERP (Enterprise Resource Planning) to manage their business operations and adapt according to new market challenges. SAP R/3 is an integrated ERP software that allows organizations to manage their business efficiently. Organizations can reduce the cost to run their operations by using SAP R/3 ERP packages.

SAP R/3 also allows customers to interact with different databases to run different applications with the help of a user-friendly GUI. The SAP R/3 system is divided into different modules to cover the functionality of different business operations in an organization.

The most common SAP R/3 modules are:

- SAP Material Management
- SAP Financial Accounting and Controlling
- SAP Sales and Distribution
- SAP Human Resource
- SAP Supply Chain Management
- SAP Plant Management

SAP Testing is about testing the functionality of these modules and to ensure that they perform as per the configuration.



A SAP system undergoes various changes like patch management and fixes, new module implementations, and various other configuration changes. All these modifications raise a need for Regression testing that is to be performed in SAP environments. SAP testing automation tools like SAP Test Acceleration and Optimization tools can be used for this purpose.

SAP TAO is an automation tool to generate test cases for end-to-end scenarios for SAP applications. Apart from this, there are various other Automation testing tools for SAP testing like HP QTP, and ECATT, etc. that can be used.

Importance of Testing

Here is a list of key reasons why SAP testing is performed and why it is an important function in the growth of an organization:

- **System Validation:** SAP Testing involves complete end-to-end testing and validation of all SAP modules in SAP ERP environment.
- **Quality and Revenue:** SAP Testing is an output-based testing and not like conventional testing methods which are input-based. It ensures the quality of SAP system and also focuses on revenue and cost of the organization.
- **Cost and Predictability:** SAP Testing involves reducing the SAP development costs and improve predictability.

- **Compliance Requirement:** SAP Testing ensures that the SAP implementation is meeting the new compliance requirements in a specific organization and all the modules are working as per the expected configuration.
- **New Implementation and Configuration Changes:** There are different types of changes implemented in a SAP system, like patches and fixes, new implementation, configurational changes. Therefore, SAP testing ensures that all the modules are performing as per requirement in this dynamic system environment.
- **Integration:** SAP testing is performed to test various reports, data flows and work flows, GUI forms, etc. It is used to check system integration between different modules. For example, if an order posting is done that requires an action in Sales and Distribution, MM and FICO, then SAP testing checks the integration between these systems.
- **Performance:** It is also used to ensure if the system will be able to meet the Service Level agreements, time taken by system to perform specific actions, performance of the system, etc.

2. SAP Testing – Types of Testing

There are different testing methods that can be used to test the functionality of a software, system, or an application.

The most common testing techniques are:

- **Unit Testing:** It is a type of white-box testing that involves testing a single unit or group of units.
- **Integration Testing:** In this testing, multiple systems are combined together to test the output of the integrated system.
- **Functional Testing:** It checks the functionality of each module as per the desired result.
- **Usability Testing:** It checks the ease of use of an application or a system. It checks how easy it would be for a new user to use an application or to understand the system.
- **Acceptance Testing:** Acceptance testing is performed to test if a system meets the user requirement and whether to accept the application or system.
- **System Testing:** Entire system is tested as per the requirement and specification.
- **Stress Testing:** In this testing, the system is put into stress beyond its specification to check when it fails.
- **Performance Testing:** This testing is performed to check if the system meets the performance requirement.
- **Regression Testing:** It includes testing the full application or system for the modifications.
- **Beta Testing:** The aim of beta testing is to cover unexpected errors. It falls under the class of black-box testing. It is performed by releasing the pre-version of the final product, called Beta.
- **Database Testing:** Database testing is used to test the data in the database. It is performed using SQL statements.
- **ETL Testing:** ETL testing is performed to ensure if data is correctly extracted, transformed, and loaded from a source system to a target system.

3. SAP Testing – Manual and Automation Testing

Manual Testing

Manual testing means you are testing a software manually without using any automated tools or any script. In this type of testing, the tester takes over the role of an end-user and tests the software to identify bugs or any unexpected behavior.

There are different stages of a manual testing. They are: unit testing, integration testing, system testing, and user acceptance testing.

Various test plans, test cases, or test scenarios are used by a manual tester to ensure the completeness of testing. Manual testing can also be called **exploratory testing** because the testers explore the software to identify errors in it manually.

Automation Testing

In Automation testing, the tester writes the scripts and uses software tools to test the product. This process involves the automation of a manual process. Automation testing includes re-running the test-cases multiple times that were performed manually.

Automation testing is also used to test the application from load, performance, and stress purpose. It is used to increase the coverage of test. Automation testing improves the accuracy and saves time and money in comparison to manual testing.

Software Testing Tools

The following tools can be used for Automation testing:

- HP Quick Test Professional (QTP)
- Selenium
- SAP TAO
- ECATT
- IBM Rational Functional Tester
- SilkTest
- TestComplete
- Testing Anywhere
- WinRunner
- LoadRunner
- Visual Studio Test Professional
- WATIR

4. SAP Testing – SDLC

Software Development Life Cycle determines the series of steps to be performed to develop an application or the efficiency of a software. In this chapter, we will discuss the phases defined in SDLC. Each phase has its own process and deliverables that goes into the next phase.

Feasibility Study

The first stage of SDLC is **requirement gathering**. After the requirements are gathered, the team comes up with a rough plan of software process. At this step, the team analyzes if a software can be made to fulfill all the requirements of the user. It is found out if the project is financially, practically, and technologically feasible for the organization to take up. There are many algorithms available, which help the developers to conclude the feasibility of a software project.

System Analysis

At this step, the developers decide a roadmap of their plan and try to bring up the best software model suitable for the project. System analysis includes understanding of software product limitations, learning system-related problems or changes to be done in the existing systems, identifying and addressing the impact of the project on the organization and personnel etc. The project team analyzes the scope of the project and plans the schedule and resources accordingly.

Software Design

The next step is to bring the whole knowledge of requirements and analysis on to the desk and design the software product. The inputs from the users and the information gathered in the requirement gathering phase are the inputs of this step. The output of this step comes in the form of two designs; **logical design** and **physical design**. Engineers produce meta-data and data dictionaries, logical diagrams, data-flow diagrams and in some cases pseudocodes.

Coding

This step is also known as **programming phase**. The implementation of software design starts in terms of writing the program code in a suitable programming language and developing error-free executable programs efficiently.

Testing

An estimate says that 50% of the whole software development process should be tested. Errors may ruin the software from critical level to its own removal. Software testing is done while coding by the developers and thorough testing is conducted by testing experts at various levels of code such as module testing, program testing, product testing, in-house testing and testing the product at user's end. Early discovery of errors and their remedy is the key to developing a reliable software.

Integration

Software may need to be integrated with the libraries, databases, and other program(s). This stage of SDLC deals with the integration of the software with outer world entities.

Implementation

Implementation or deployment means installing the software on user machines. At times, the software needs post-installation configurations at the user's end. Software is tested for portability and adaptability and integration related issues are solved during implementation.

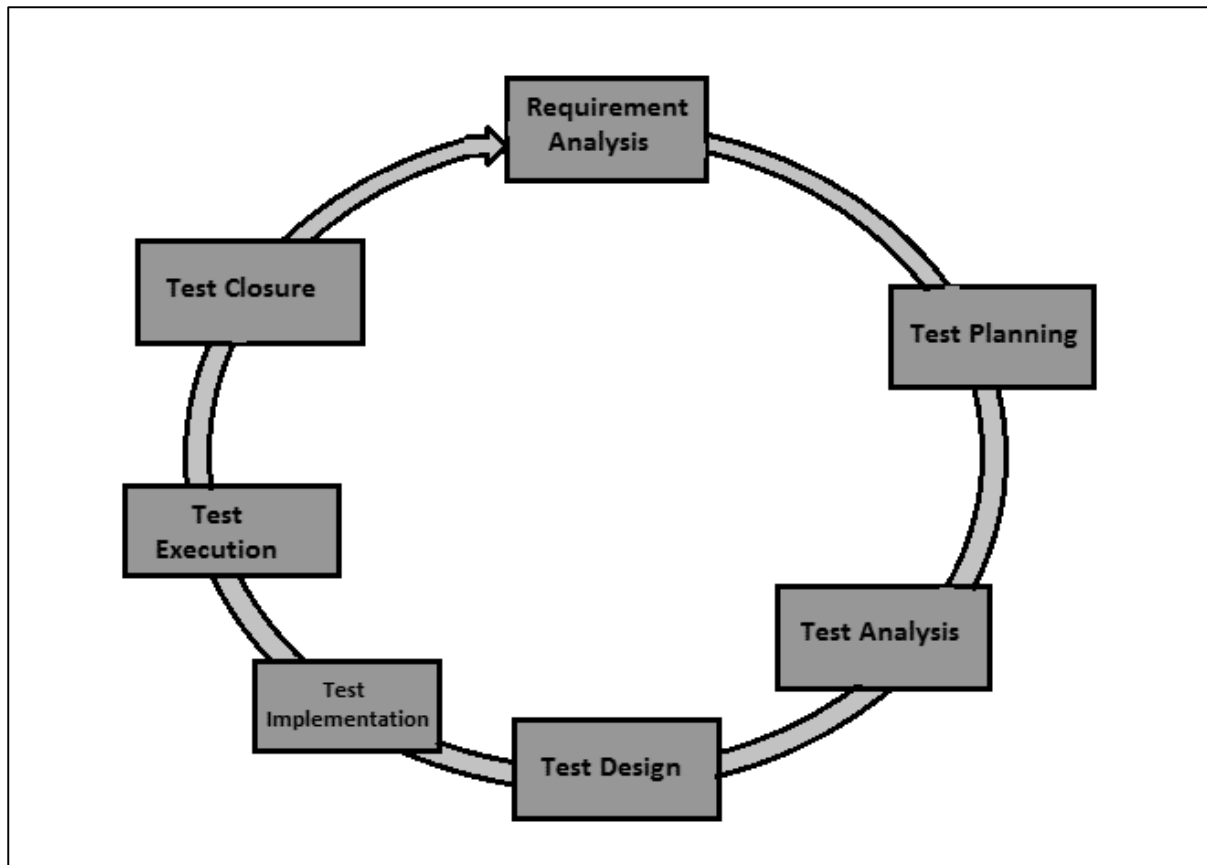
Software Testing Life Cycle

Software Testing Life Cycle (STLC) consists of all the steps that are performed in a specific way to ensure that quality goals are met and each step has specific goals and deliverables.

STLC is used to improve the quality of a software product and to make it capable to meet the business requirements to achieve certain goals.

The different stages that come under Software Testing Life Cycle are as follows:

- Requirements phase
- Test Planning
- Test Analysis
- Test Design Phase
- Test Implementation
- Test Execution Phase
- Test Closure Phase



Requirement Phase

This is the first phase of Software Testing Life Cycle. During this phase, the tester's job is to analyze the requirements. There are various methods for Requirement Analysis like conducting brainstorming sessions with business people, team members, and try to find out whether the requirements are testable or not.

This phase determines the scope of the testing. If a testing team finds any features that can't be tested, then that should be communicated to the client.

Test Planning

In this phase, the tester identifies the activities and resources which would help to meet the testing objectives.

Various metrics are defined and there are methods available to determine and track those metrics. Test planning also includes identifying key performance indicators for testing evaluation.

Test Analysis

This phase determines the guidelines that has to be tested. It includes identifying the test conditions using the requirements document, any risks involved, and other test criteria.

Various factors are used to find out the test conditions:

- Product Complexity
- Depth of Testing
- Risk Involved
- Skills Required
- Knowledge of testing team members
- Test management
- Availability of the stakeholders

Test conditions should be written in a detailed way.

Let us take an **example**. For a website selling products online, a test condition is that a customer should be able to make an online payment. You can add detailed conditions like, payment should be feasible using Credit card, NEFT transfer, debit card or net banking.

The advantage of writing the detailed test condition is that it increases the scope of testing because test-cases are normally written on the basis of the test condition. It allows to write more detailed test cases. It also helps in determining the condition of when to stop the testing of a software product.

Test Design Phase

This phase determines how the tests are performed.

- Break down the test conditions into multiple sub-conditions to increase its coverage.
- Get the test data.
- Set up the test environment.
- Get the requirement traceability metrics.
- Create the test coverage metrics.

Test Implementation Phase

This phase includes the creation of detailed test-cases as per the test conditions and metrics defined.

- Prioritize the test case.
- Test-case to be used for Regression.
- Ensure the correctness of the test-cases.
- Sign off of the test-cases before the actual execution starts.

Test Execution Phase

This phase of Software Testing Life Cycle involves actual execution of test-cases.

- Execute the test-cases.
- Log the defects.
- Check traceability metrics to track progress.

Test Closure

This phase includes checking for the completion of the test.

- Check if all the test-cases are executed and opened defects.
- Note down the lessons learnt.
- Close the Testing phase.

5. SAP Testing – Testing Types in SAP

There are different types of testing methods available that can be used to perform SAP testing.

Unit Testing

Unit testing is used to test the functionality of a SAP system and its various components. It is performed by domain and configuration experts who know the functionality of each unit in a system.

Example

Suppose the task is to create a sales order and save it. To perform unit testing for this task, the tester should know that the sales order can be saved using the SAP organization elements like customer master data, partner functions, material master data, company code, credit control area, sales organization, etc.

Example

In ABAP development, Unit testing can be performed to check if a report can be created from developer-generated data. It requires assistance from the domain expert.

System Testing

System Testing involves the integration of elements of a SAP system to ensure that related SAP functionality are linked together in the development environment.

Example

If you say a cash flow for a quotation in an organization would show that a quote can be used to create a sales order, a delivery can be created and processed from the order, the delivery can be billed, the billing released to accounting, and a customer payment applied against the accounting invoice.

Each unit is tested like this and then the test results are combined using system testing.

Scenario-based Testing

Scenario testing, as the name suggests, is performed as per specific business cases.

Example

Suppose there are a few tasks that are specific to a customer segment or a given product line or a set of services. For these specific line of target, you have different scenarios that you need to test. This testing is also performed in the development environment.

Integration Testing

In this testing, testing data comes from a real data extraction source. Data is known to business end-users.

Integration testing is used to present that the business process, as designed and configured in SAP, runs using real-world data. In addition the testing shows that the interface triggers, reports, workflows are working.

SAP Interface Testing

Interface testing ensures that a business process on a SAP system runs automatically, the events are triggered, and the results are transferred to the receiver system. Interface testing involves execution on the sending system followed by automatic generation of the interface output, and then the receiving system consuming that file and proving that a business process continues on the receiver.

Ideally, interface testing involves larger testing activities as a project progresses. Interface testing shows that triggering works, the data selection is accurate and complete, data transfer is successful, and the receiver is able to consume the sent data.

SAP User Acceptance Testing

SAP UAT is used to ensure that the end-users are able to perform the assigned job functions with the new system. The important aspect of this testing is to understand the business requirement and to ensure that the expected features, functions and capabilities are available.

SAP System Performance Testing

Performance testing checks the following aspects:

- Whether the system response time is acceptable as per the business requirement
- Whether periodic processes are running within permissible time,
- Whether the expected concurrent user load can be supported.

Performance testing identifies bottlenecks and coding inefficiencies in the SAP system. It is not likely that system performance tuning is perfectly set up and the program is running with optimized code.

SAP Load Testing

In Load Testing, the tester applies maximum load on a system, either online users or periodic batch processing, and identifies whether the system is capable enough to handle the load. If not, it finds out the steps needed to improve performance.

SAP Security and Authorizations Testing

Security and Authorizations Testing is used to ensure that users are only able to execute transactions and access appropriate data that is relevant to their project.

As with the implementation of Security standards, this is really important to test if security and authorization is placed in a system. Test IDs for job roles are created and used to both confirm what a user can do and what a user cannot do.

SAP Cutover Testing

Cutover testing is usually performed once in a project lifecycle. Here a full-scale execution is done of all the tasks involved to extract data from legacy systems. Then, to perform any kind of data conversion, load the results into the SAP system and fully validate the results, including a user sign-off.

SAP Regression Testing

Regression testing is used to find new functionalities and to test previous functionalities in a system when it is upgraded or a new system is set up. The key role of regression testing is to test the existing functionality and newly updated configuration and codebase.

When you upgrade your SAP system or apply a patch, it shouldn't affect the functionality that is expected to be performed by the users. In addition, it should not affect the new features that are supposed to be introduced in a new release.

6. SAP Testing – Testing Process

SAP testing process is usually divided into three phases:

- Test Planning
- Test System setup
- Test Execution and evaluation

Test Planning

Test planning includes the steps that are involved in the initial phase of testing.

- Gathering the requirement. What needs to be tested? Functional requirements to be collected for system and application testing.
- Test-case development for manual and automation testing. In automation testing, various tools can be used for creating test-cases.
- Reviewing the test-cases

Test System Setup

Test system setup involves setting up the test environment to run the test-cases. Here, the tester needs to define key metrics for reporting.

Test Execution and Evaluation

Test execution and evaluation involves executing the test-cases and noting down the output. It includes the following activities:

- Defect handling and reporting
- Assessment of Test plans as per results
- Documentation of all defects and compare the results with key metrics

SAP Testing – Critical Process

In SAP system testing, you need to identify critical business processes that will be affected by implementing changes in a SAP system. It mostly comes under Regression testing, where you apply a patch or implement a new system.

The first step before applying these changes is to perform **change impact analysis**. This covers critical processes that will be affected by implementing the change.

Example

Let us take an example. Suppose the planned changes are as follows:

Planned changes:

- SAP support pack 7
- Custom code
- SAP enhance package

System effected:

- Sales Order
- Delivery of goods
- Payment method

Impact Analysis for Critical Processes:

Here, the following activities are to be carried out in Impact Analysis:

- Identification of the critical business processes impacted by change implementation.
- Business justification to be provided as to why this change has to be implemented.
- Creating the test plan to monitor critical processes for SAP testing while performing the change.
- Evaluation of impact of change on critical processes and the purpose to implement the change.

7. SAP Testing – Navigation

SAP testing navigation ensures that you cover each module of your SAP system and perform at least one test for each functionality. It also reduces the manual testing effort and covers most of the testing paths in a SAP system.

OPA tests can be performed to check SAP Testing Navigation. OPA is known as **Open Source Programming** language and it is mostly used for developing web applications. For compilation of OPA program, you can use Node.js on the server and JavaScript on the client side.

Creating a Test using OPA

OPA allows you to use three objects in **Qunit**. These functions should be defined in a test so that OPA knows what actions to be taken.

- **Given** – to pass arrangements
- **When** – actions to be taken
- **Then** – assertion

Example

The following example shows how to use all the 3 objects in Qunit:

```
jQuery.sap.require("sap.ui.test.Opa");
jQuery.sap.require("sap.ui.test.opaQunit");
opaTest("press a Button", function (Given, When, Then) {
  // Arrangements
  Given.iStartMyApp();
  //Actions
  When.iPressOnTheButton();
  // Assertions
  Then.theButtonShouldHaveADifferentText();
});
```

The next step is to define the three functions.

Defining the Functions

```
var arrangements = new sap.ui.test.Opa
({
  iStartMyApp : function ()
  {
    return this.iStartMyAppInAFrame("../index.html");
  }
});
```

```

    }
  });

```

In the above function, we have assumed that the app runs in a page called index.html. Our OPA test is located in the test/opa.html folder.

Defining Arrangements

```

var actions = new sap.ui.test.Opa
({
  iPressOnTheButton : function ()
  {
    return this.waitFor
    ({
      viewName : "Main",
      id : "pressMeButton",
      success : function (oButton)
      {
        oButton.$().trigger("tap");
      },
      errorMessage : "No Button found"
    });
  }
});

```

Defining Assertions

```

var assertions = new sap.ui.test.Opa({
  theButtonShouldHaveADifferentText : function () {
    return this.waitFor({
      viewName : "Main",
      id : "pressMeButton",
      matchers : new sap.ui.test.matchers.PropertyStrictEquals({
        name : "text",
        value : "got pressed"
      }),
      success : function (oButton)
      {

```

```
                Opa.assert.ok(true, "The button's text changed to: " +  
oButton.getText());  
            },  
            errorMessage : "No change in Button's text"
```

Running the OPA test

```
sap.ui.test.Opa.extendConfig  
(  
    arrangements : arrangements,  
    actions : actions,  
    assertions : assertions,  
    viewNamespace : "view."  
));
```

8. SAP Testing – Screen Flow

Screen flow logic in SAP Testing is like an ABAP code and it is used to contain the processing blocks. It contains the procedural part of the screen. It is created in screen painter and this screen painter is similar to an ABAP editor.

Screen flow logic involves no external data declaration and each processing block is defined with a prefix 'PROCESS'. For example,

- PROCESS AFTER INPUT
- PROCESS BEFORE OUTPUT
- PROCESS ON HELP-REQUEST
- PROCESS ON VALUE-REQUEST

Each screen flow logic should contain PROCESS AFTER INPUT and PROCESS BEFORE OUTPUT keywords.

In an event block, you can use keywords like MODULE, FIELD, ON, VALUES, CALL, etc.

Screen Flow Logic Keyword	Description
MODULE	Calls a dialog module in an ABAP program
FIELD	Specifies the point at which the contents of a screen field should be transported
ON	Used in conjunction with FIELD
VALUES	Used in conjunction with FIELD
CHAIN	Starts a processing chain.
ENDCHAIN	Ends a processing chain.
CALL	Calls a subscreen.
LOOP	Starts processing a screen table
ENDLOOP	Ends processing a screen table

Creating Screen Flow

In the Repository browser, double-click on the name of a screen and it will display the flow logic of the screen.

The Flow Logic Editor of the Screen Painter will open and you edit the screen flow logic. You can use any of the available ABAP source code editors to define the flow logic.

You can use the following keywords to create the screen flow logic:

CALL	Calls a subscreen
CHAIN	Starts a processing chain
ENDCHAIN	Ends a processing chain
ENDLOOP	Ends loop processing
FIELD	Refers to a field. Can be combined with the keywords MODULE and SELECT
LOOP	Starts loop processing
MODIFY	Modifies a table
MODULE	Identifies a processing module
ON	Used with FIELD assignments
PROCESS	Defines a processing event
SELECT	Checks an entry in a table

Testing Screens

Testing screens are used to test the appearance of a screen as it will appear at runtime. If you have already programmed the flow logic, you can choose whether to simulate the screen with or without it.

To perform Screen test, follow the steps given below:

- Select Screen -> Test.
- The system will display a dialog box for the runtime simulation.
- You can change the window coordinates, if it is required.
- Next, define the scope of simulation.
- To include the flow logic, click 'choose complete flow logic'.
- Click Continue and the screen simulation will appear.

9. SAP Testing – Modules

There are different SAP modules implemented in an organization that can be tested using various testing tools like HP Quick Test Professional (QTP), IBM Rational Functional Tester (RFT), and SAP Test Acceleration and Optimization (TAO) tool.

Common SAP Modules

The common SAP modules are listed below:

- **Financial Modules:** Finance Accounting and Controlling (FICO), Treasure (TR), and Enterprise Control (EC).
- **Logistics Modules:** Material Management (MM), Plant Maintenance (PM), Sales and Distribution (SD), Quality Management (QM), etc.
- **Human Resource Management:** Accounting Payroll, Time Management, Training and Event Management.

All these modules are inter-dependent and the functionality of one module affects the functionality of other modules.

Example

Suppose you have to create a Sales Order in Sales and Distribution (SD) module. Here, you first need to enter the **transaction code** (e.g., Transaction Code VA01). Next, check the stock of the item in Inventory module and check the credit limit available on Customer profile in Customer Relationship Module. It shows that all these modules are interdependent; if you customize any of these modules, it will affect the related ERP system.

To perform SAP testing, you need to understand the features, functionalities, and how the workflow takes place in these SAP modules. Most of the common reasons of failure of ERP implementation project is incorrect test planning and the use of wrong test-cases.

Non SAP ERP systems like PeopleSoft, Edwards, Oracle E business suite have different customers and capabilities. The testing team needs to understand the functionality of complete system.

Types of Testing Team

There are normally two types of testers available in SAP projects:

- **Core Testers** who are responsible to perform basic testing of ERP system and modules.
- **Implementation Testers** who work on implementation project and cover the customization functionalities of SAP modules.

Customization requests from clients can impact the modules of a SAP system. The testing team should be able to record each customization request and its impact on the other SAP modules.

ERP systems are large systems and therefore the testing process should ideally be automated. It is always advisable to perform automated testing for ERP systems, as manual testing is a very time-consuming and lengthy process. Without testing each component of the SAP system, it is really tough to achieve 100% quality and successful implementation of SAP project.

Performing SAP testing for the above example using QTP

To perform SAP testing for the above example, follow the steps given below:

- The first step is to install HP QTP tool and to install necessary plug-ins within QTP to make it compatible to connect to ERP system.
- The next step is to use HP Quality Center (QC) to develop the test plan and then to convert test plan design to Test Scripts. These test plans can be converted to scripts using HP QTP.
- Next, record the R/3 GUI screen of SAP system for SD module while creating Sales Order or creating PR in MM.
- After you are done with the recording in QTP tool, create a script in VB.

```
SAPGuiSession("Session").SAPGuiWindow("SAP Easy Access -  
User").SAPGuiOKCode("OKCode").Set "/nVA01"  
  
SAPGuiSession("Session").SAPGuiWindow("SAP Easy Access - User").  
SendKey ENTER
```

You can also add different parameters and customizations as per your requirement.

10. SAP Testing – Interfaces

An Interface in an ERP system is known as a tool that is responsible to get data from one system and move that data to another system. For example, assume you have a program that produces a report in XML format and then this program reads the XML file to provide input to another system. You can also manipulate while passing the information from one system to other.

Example

Consider a vendor tool that takes care of employee attendance. Now, the system interface will take this information and populate it in the SAP HR system.

Interface testing ensures that this job is running successfully to ensure that the data is transferred completely and there is no error while transferring the data to the SAP HR system.

Points to be considered while performing SAP Interface Testing

SAP Interface testing is purely dependent on the operations and organizational processes. While performing SAP Interface Testing, you need to consider the following key points:

- What is the purpose of using SAP Interface and what business scenarios are processed by the interface?
- Check if the Interface is processing all business scenarios accurately as per the test strategy.
- To perform Interface testing, the best practice is to start with performing Unit testing.
- You have to check if the outbound interface is alright, which means that it ensures to perform file meeting specifications in terms of layout, etc.
- You have to check if the inbound interface is okay. Check if it is reading the file correctly and if it is performing the correct steps in the target system.

11. SAP Testing – Test-Cases

SAP test-cases are required to perform a check on the installation and configuration of the SAP system, any new implementation, multi-language and device testing, intranet testing, real-time testing, etc.

An ERP system is a common centralized system and is used by multiple users simultaneously in real time. Hence it creates a need to write the test-cases with lot of effort and dedication.

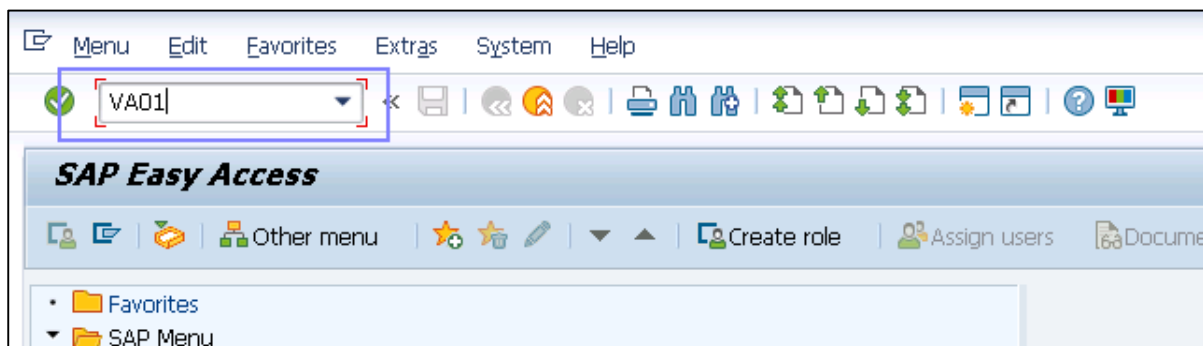
An ERP system also involves various FI transactions, so each test-case should cover the scope of all the configuration and implementation part. Test data should be passed carefully and each test should have a column with name output data.

Example

Test Case ID: XYZ_ERP_SD_A20301

Module: SAP Sales & Distribution SD



Let us check the transaction **VA01** to create a sales order in the Sales and Distribution (SD) system.



Fields to be entered while creating an Order:

- Order Date
- Order Type
- Expiry Date
- Customer ID
- Shipping Id, Shipping Details, etc.

Create Sales Order: Initial Screen

☐ Create with Reference
  Sales
  Item overview
  Ordering party



Order Type AEBO order

Organizational Data

Sales Organization	0001	Sales Org. Germany
Distribution Channel	C1	Direct Sales
Division		
Sales Office		
Sales Group		

Once you enter the details, press Enter and add all the details in the Sales Order.

Create AEBO order: Overview

 Orders
  Document

AEBO order Net value

Sold-To Party

Ship-To Party

PO Number PO date

Sales | Item overview | Item detail | Ordering party | Procurement | Shipping | Reason for rejection

Req. deliv.date Deliver.Plant

☐ Complete dlv. Total Weight

Delivery block Volume

To create a SAP test-case, you can pass input data (correct and Incorrect and see the outcome):

Correct Input Data

- Order Date 01/01/2016, Order Type Sales Order
- Expiry Date 15/01/2016, Shipping Date 10/01/2016,
- Select Payment Due Date 10/01/2015, Item Qty 10, etc.

Incorrect Input Data

- Order Date 01/01/2017, Order Type Sales Order
- Expiry Date 15/01/2017, Shipping Date 10/01/2017,
- Select Payment Due Date 10/01/2017, Item Qty 0, etc.

Output Data: With Correct Data

- Order is successfully saved in SAP module and invoiced.

- Next is Packing slip number.
- Next is Shipping Order number, etc.

Output Data: With incorrect Data

- Error message for incorrect data input. Text message for incorrect input data should be defined in the system.

Expected Output

- Sales Order#
- Sales Invoice#
- Packing List#, etc.

12. SAP Testing – TAO

SAP Testing TAO (Test Acceleration and Optimization) is a tool used to perform automated testing of SAP systems. SAP TAO helps customers to fasten the process of creating automated test cases for SAP systems. Automation testing using TAO is performed by creating test components for various transactions in SAP modules.

Test components using TAO are uploaded to HP Quality Center. Test components created like this are normally for the single transactions and can be later used to create test scenarios. This tool can be easily integrated with SAP Solution Manager to maintain the different test components.

In short, you can say that SAP TAO enables SAP clients in automating the business needs by automatically creating draft test-cases and test components.

Key Features of TAO

SAP TAO enables customers to breakdown a single software into multiple parts which can be integrated to test cases using a simple interface by dragging and dropping each part.

- TAO supports flexible reuse of test cases and data.
- If there is any functional change in the system, it is easy to maintain test-cases because of this change.

The common versions of SAP TAO are TAO 2.0 and TAO 3.0.

SAP TAO Integration with other Tools

One of the key features of SAP TAO is that it can be easily integrated with various tools to create automated test-cases. Some common tools which can be integrated with TAO are:

- SAP Solution Manager (Solman)
- HP Quality Center QC
- HP Quality Test Professional QTP

SAP TAO 2.0

SAP TAO 2.0 is a tool that is used to create automatic test-cases during Regression testing of a system. It helps SAP customers to create different test components from the screens of a transaction and parameterize them.

These test components are created normally for one transaction code and later they can be combined to test scenarios. It can be easily integrated with the Business Process Change Analyzer in SAP Solution Manager.

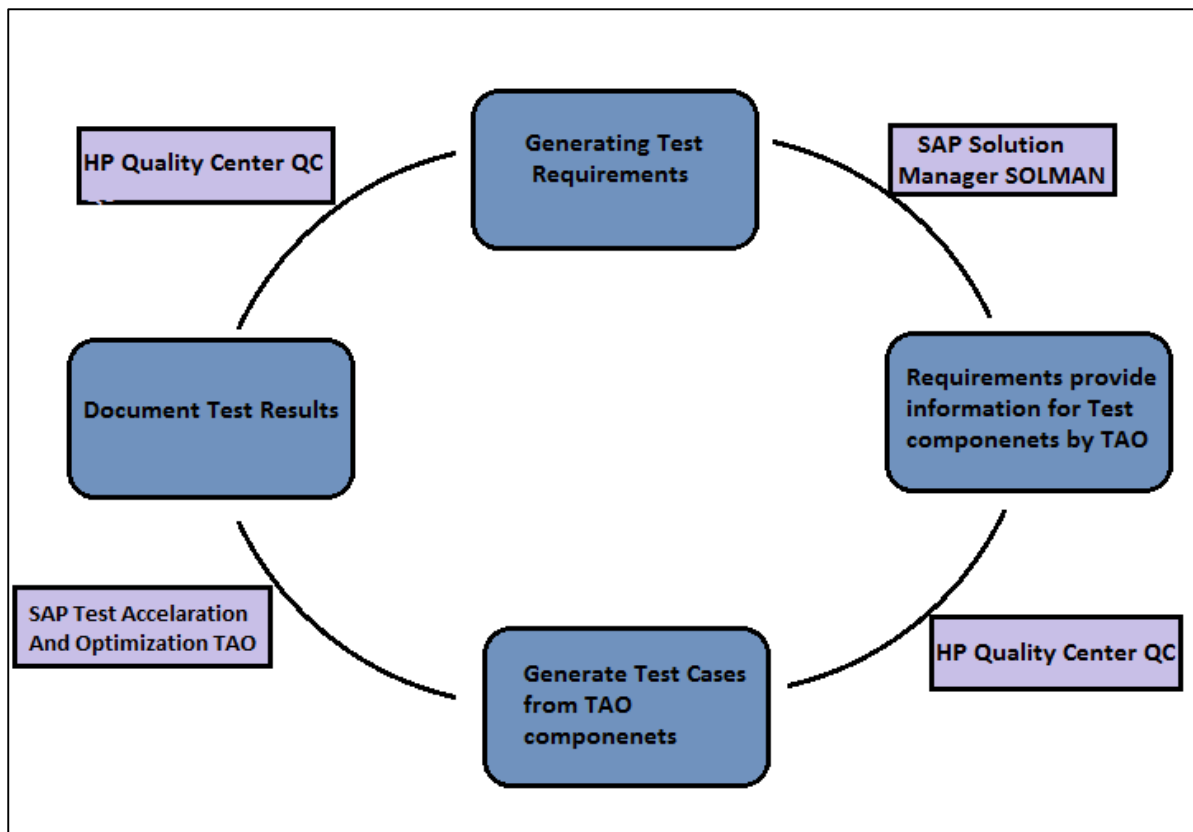
SAP TAO 2.0 comes with different Service pack SP 02, SP 04, and SP 06.

SAP TAO performs the following tasks in SAP Testing Lifecycle:

- Test Cases creation
- Regeneration of affected components
- Test Cases Consolidation
- Test Execution

SAP TAO Process Architecture

The following illustration shows the process architecture of SAP TAO.



The steps are as follows:

- The first step is to get the requirements for SAP Testing. This includes understanding the functionality of SAP modules and identifying the requirement for testing to be performed.
- Next, get the information from the requirement analysis to generate the Test components.
- The next step is to generate test-cases and components using SAP TAO tool.
- The last step is to document the test results and update the analysis with test results.

Prerequisites for Installing SAP TAO

Downloaded SAP Solution Manager Solman compatibility Matrix, Quality Center, and SAP TAO from this link – <http://service.sap.com/testing>

Go to Test Management and Additional Information > Test Management. It will show all the compatible components supported by SAP TAO.

Check the prerequisites for components on SAP systems in your system landscape:

- Version of SAP Solution Manager SOLMAN, after checking the compatibility matrix as above.
- Version of ST-PI on all systems under test.
- Version of ST-A/PI on systems under test of type CRM.

For SAP Solution Manager Configuration work center, ensure that the systems on which testing has to be done are connected to SOLMAN.

SAP TAO client installation requires the following configuration on local system:

- RAM: 4 GB
- Free disk space: 500 MB
- LAN connection to SAP Quality Center QC server and SOLMAN
- Administration Rights
- SAPGUI with the latest patch level
- Microsoft Excel 97 or higher.
- Microsoft Internet Explorer for CRM UI support

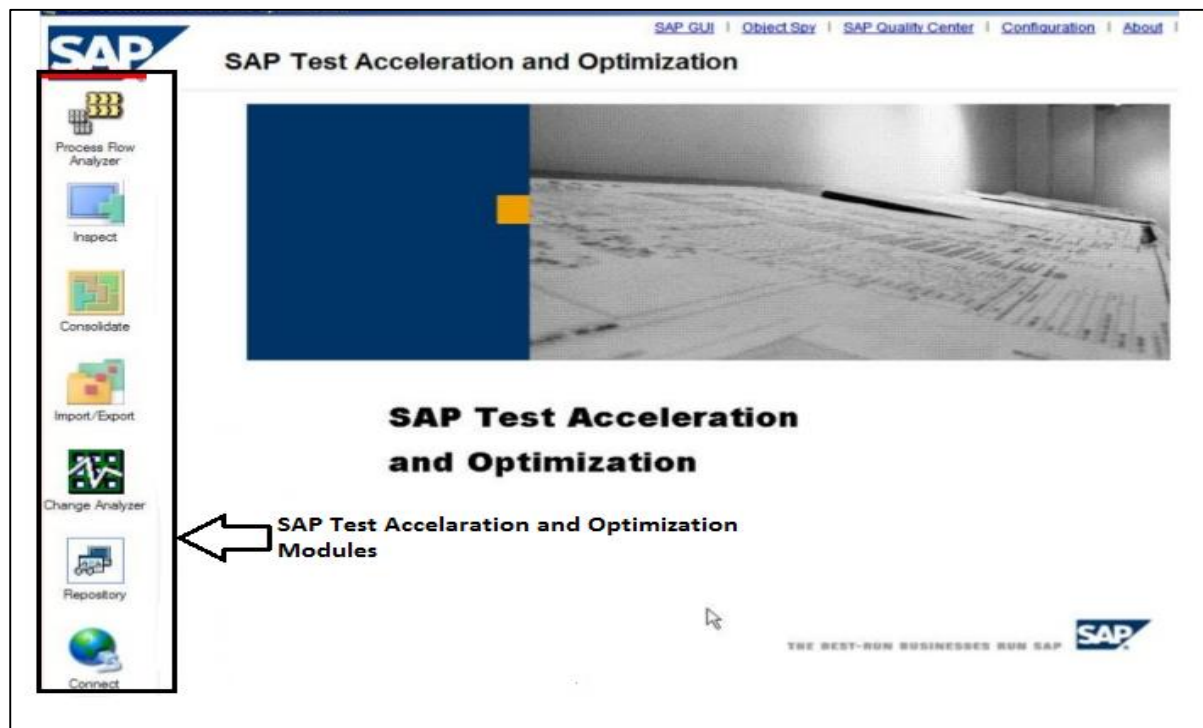
13. SAP TAO – Folder Structure

There are various modules inside the SAP TAO tool. We have discussed the key components in this chapter.

Process Flow Analyzer

The first component is Process Flow Analyzer which is used to automatically find out the user interfaces used in transaction codes executed in the SAP system. It automatically creates the test components and uploads them to Quality Center.

Process Flow Analyzer is also used to identify the sequence of test components as per user actions and creation of spreadsheet values.

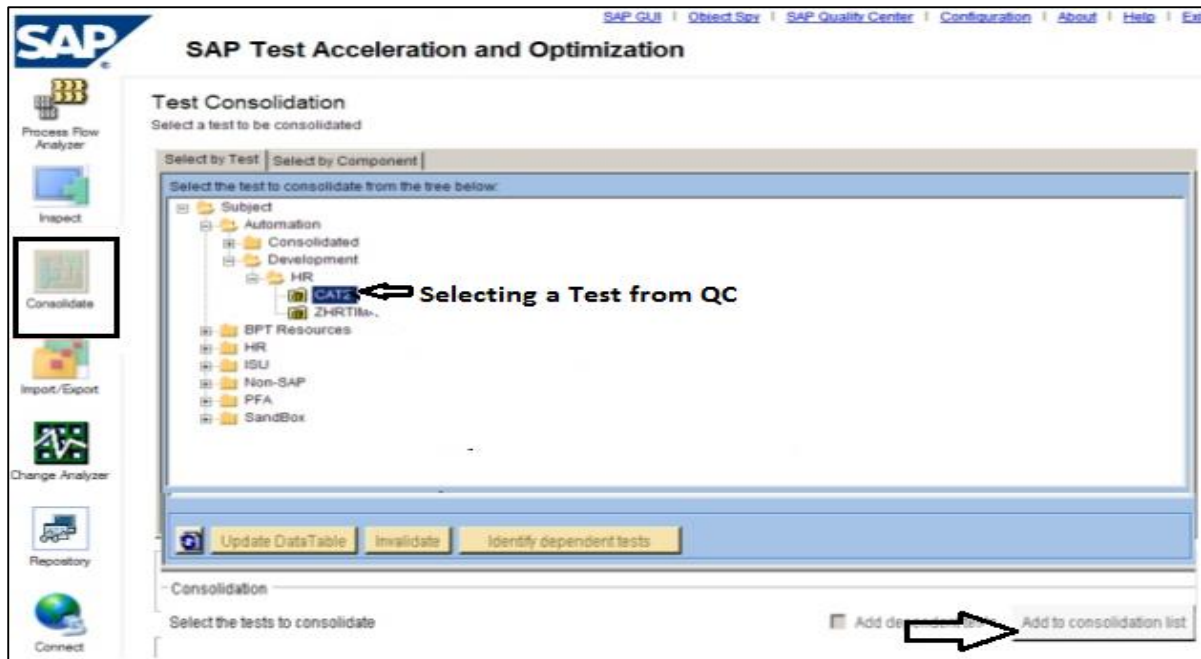


Inspect

You have to add Transaction codes to the list. SAP TAO will return all the screens associated with a particular transaction. You can choose any screen and click on Inspect option.

Consolidate

It allows you to collect multiple test components into one test. You need to find the test in Quality Center from QC tree and click on Consolidate.



Select the Transaction code you want to consolidate and click **Add to Consolidate list** at the bottom to create a test scenario.

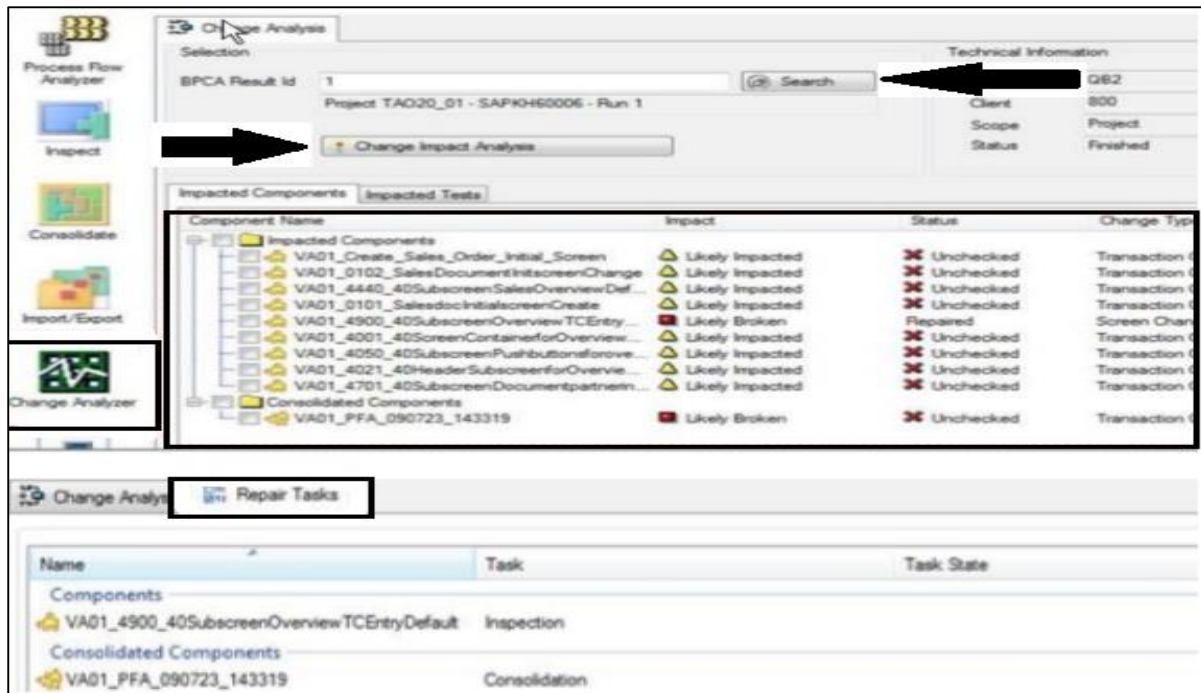
Import and Export

Import and Export options are used to export to Quality Center and import from Quality Center. You need to select the components to be imported and exported and click on the required button.

Change Analyzer

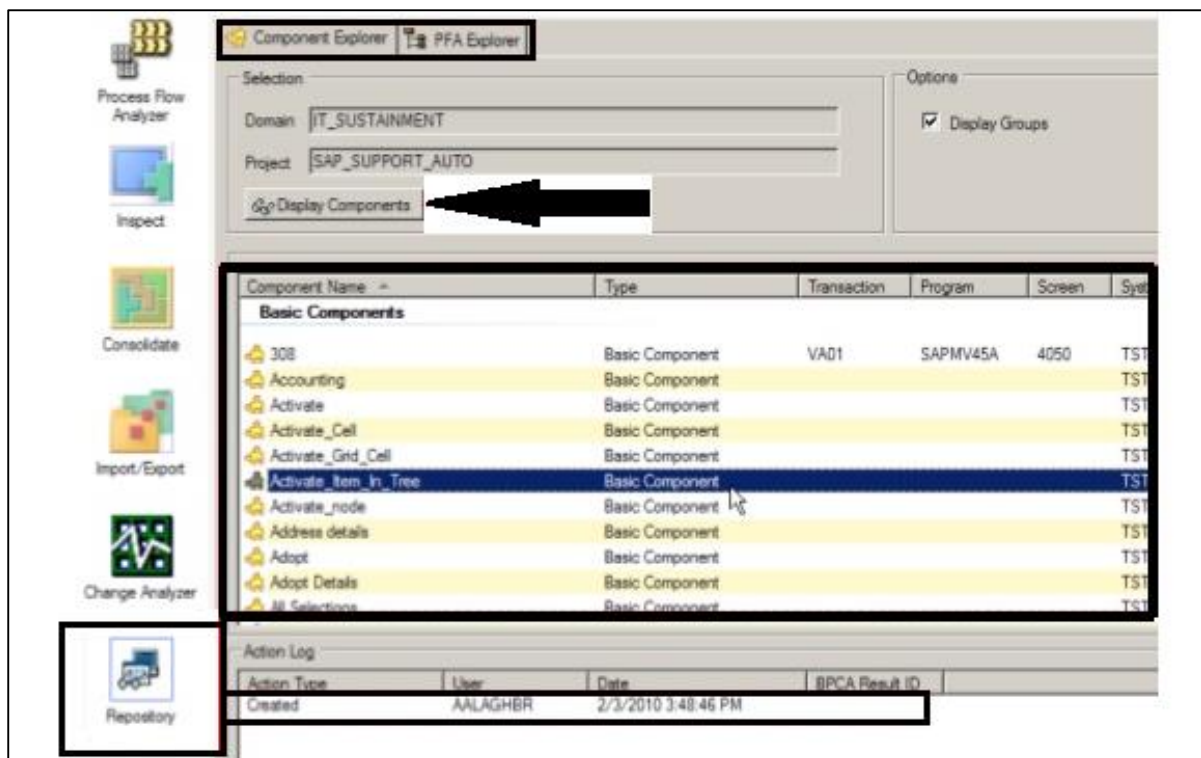
Change Analyzer is used to track the changes and impact on the SAP system. When you make any change to a SAP system, it identifies the affected business processes.

To use Change Analyzer, select the specific project and click the **change impact analysis** option. It will allow you to review and repair the impacted components in the SAP system because of this change.



Repository

Repository is used to contain information about all the test components and flow in a project. To review information about any test component and to check the process flow, you can click **Component Explorer** or **PFA Explorer**.



14. SAP Testing – Configure SAP Quality Center

Configuration of SAP Quality Center can be done in two ways:

- HP Server Attributes
- HP Project

HP Server Attributes

Step 1: In SAP TAO, click SAP Quality Center. You should have SAP Quality Center URL in the connection panel of SAP TAO. Next, enter the URL in the web browser.

Step 2: Enter the following values in SAP Quality Center by HP Site Administration and click **apply to entire**.

DISABLE_EXTENDED_STORAGE=N BACKWARD_SUPPORT_ALL_DOMAINS_PROJECTS=Y

Step 3: For a new project, perform the following steps:

- Create new domain and project in SAP QC.
- Create new user to assign that project to new user.
- Allocate Project Administrator role to newly created user.

Step 4: Log off from SAP Quality Center by HP Site Administration.

HP Project

In SAP TAO, click SAP Quality Center. You should have SAP Quality Center URL in connection panel of SAP TAO.

- **Step 1:** Enter the URL in your web browser.
- **Step 2:** The next step is to login to domain/project.
- **Step 3:** Navigate to the Test Plan.
- **Step 4:** Under Subject folder, create a folder with the name BPT Resources -> create a folder library.
- **Step 5:** SAP TAO notes information about test Consolidation into a user field of the test entities in QC and by default it is TS_USER_01. Note that this is correct when connected to SOLMAN for all the tests created by SAP TAO.
- **Step 6:** SAP QC users shouldn't use the similar user field for other purposes in their Quality Center projects.

Creating Application Area in QTP

Application area is required by SAP TAO components to contain HP QTP and web add-ons. To create an application area in QTP, you need to perform the following steps:

- Run QTP by HP as Administrator. It varies as per the operating system.
- You can select Web addins and SAP.
- Next, connect automatically or manually to Quality Center project. This step is automatic if it is configured.
- Next, create an application area with the name_SAP_Doc. The name of application area is entered when the Save button is clicked.

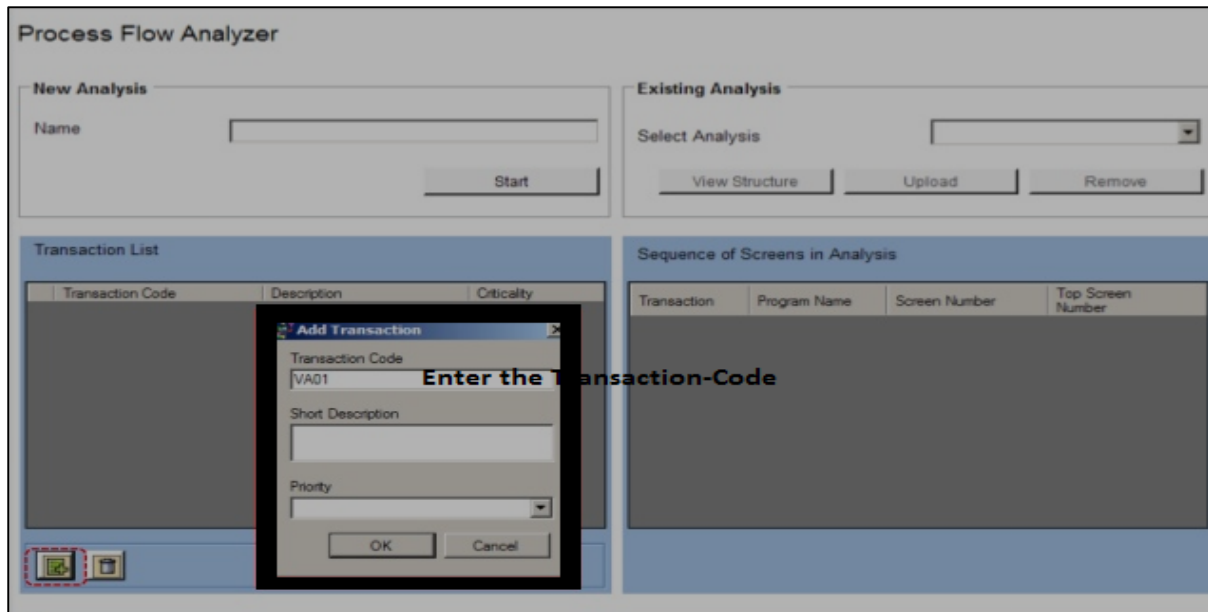
Process Flow Analyzer (PFA)

Process Flow Analyzer is used to automatically find out the user interfaces used in transactions codes executed in SAP system. It automatically creates the test components and upload them to Quality Center.

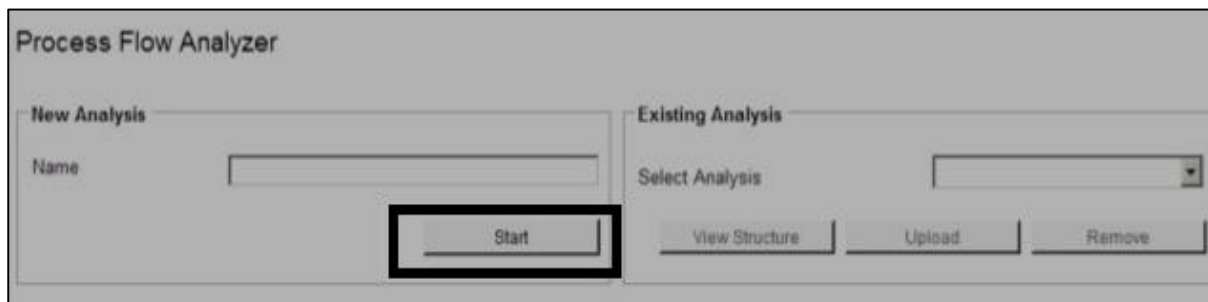
It is also used to identify sequence of test components as per user actions and creation of spreadsheet values.



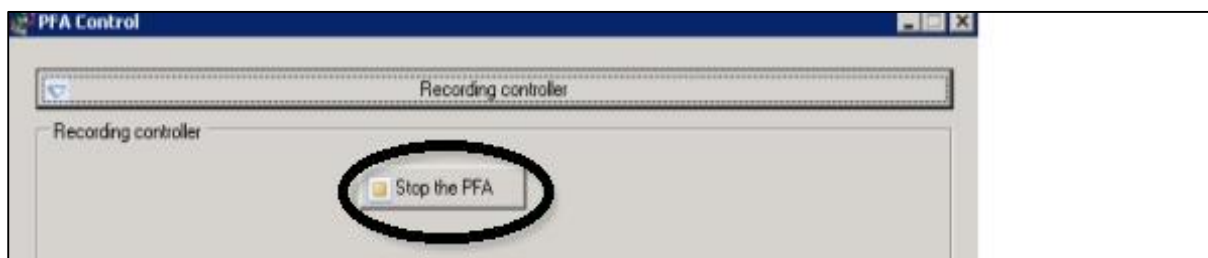
Click Add Transaction button -> Enter the Transaction-code and click OK.



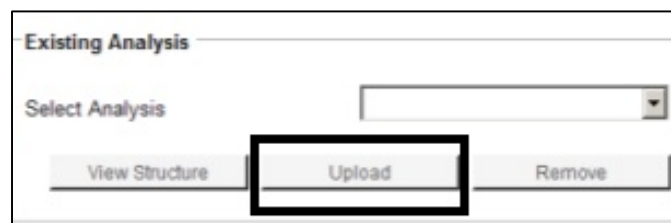
The next step is to click the **Start** button.



TAO will launch SAP and will log the Transaction entered by you. Once the transaction process is completed, click the **Stop** button.

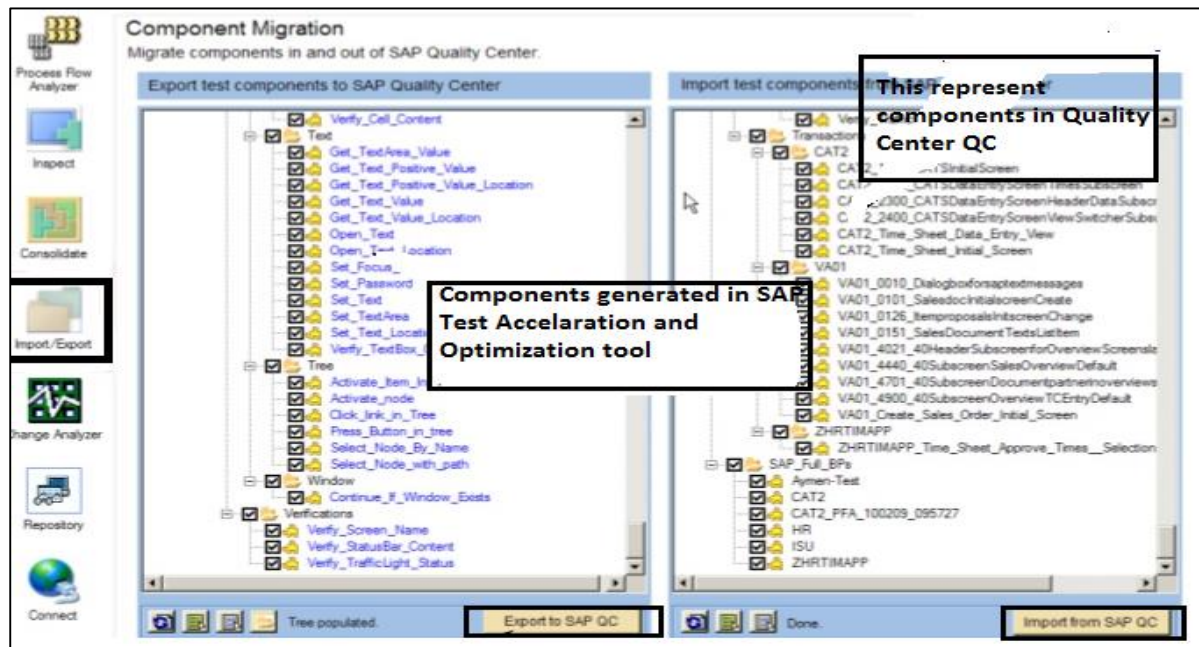


TAO will return all the screens that are used to create the process flow. Next, upload the analysis to QC. This can be done by clicking the **Upload** button.



Import/Export

Import and Export options are used to export to Quality Center and import from Quality Center. You need to select the components to be imported and exported and click the required button.

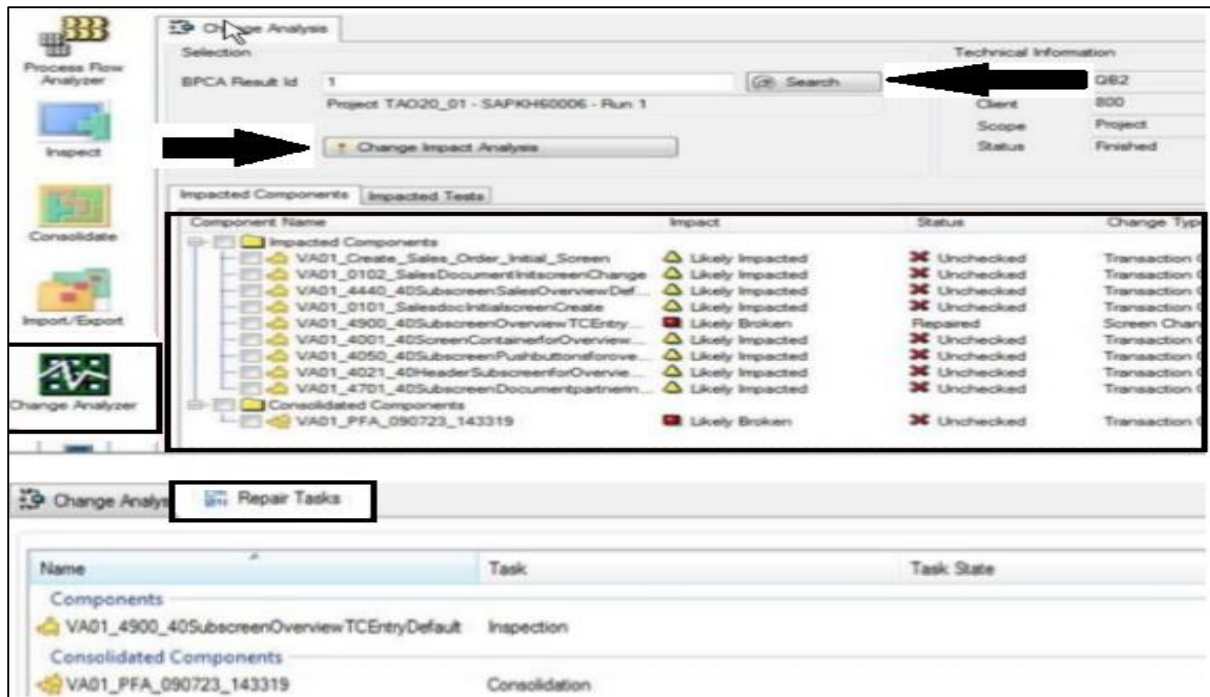


First, identify the components to be imported or exported. Click **Export** to SAP QC or **Import** from SAP QC to import/export the required components.

Change Analyzer

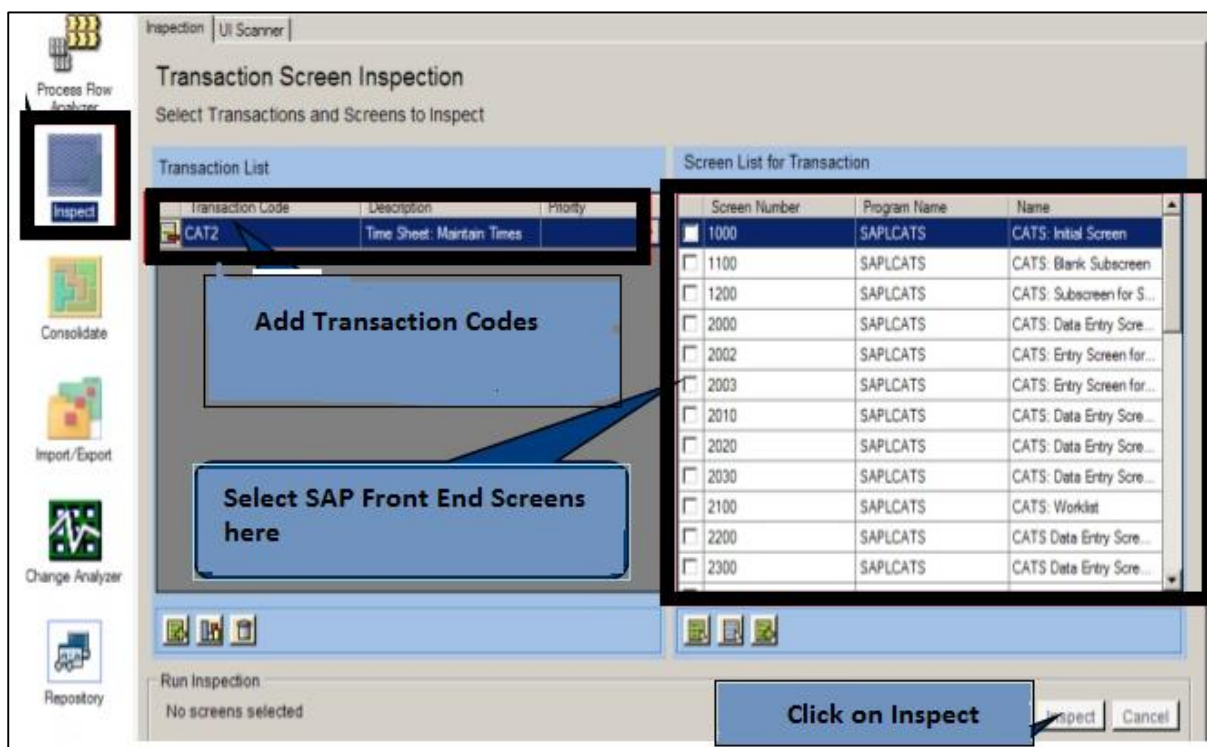
Change Analyzer is used to track the changes and impact on the SAP system. When you make any change to a SAP system, it identifies the affected business processes.

To use Change Analyzer, select the specific project and click the **change impact analysis** option. It will allow you to review and repair the impacted components in the SAP system.



Inspect

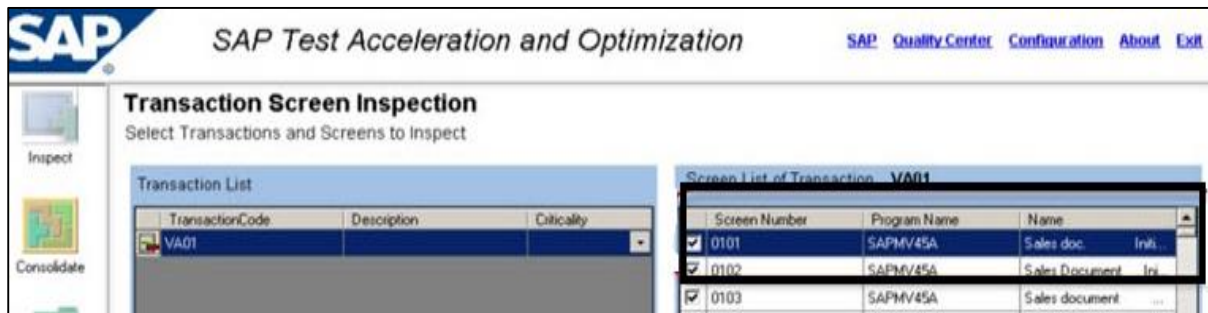
You have to add Transaction codes to the list. SAP TAO will return all the screens associated with a particular transaction. You can choose any screen and click the **Inspect** option.



15. SAP TAO – Results Analysis

To analyze the results of SAP TAO, perform the following steps:

Step 1: Go to the Tests list and in the tree, click the test for the analysis. It will take you to list of reports and the status of running tests will be displayed.



Step 2: Analyze a test, click the **View Report** option. It will open a new window with a detailed report.



There is an option to adjust the columns to be displayed by clicking the mouse icon. You can also see the HTML format of the test report by clicking the HTML Report. To investigate the test results further, you can go to the log folder from the HTML report.

Transaction Screen Inspection									
Select Transactions and Screens to Inspect									
Inspection Results									
Number	TCODE	PROG	SCREEN	Name	Description	Inspection	ComponentCreation	Uploaded	
1	ME21	SAFM06E	0101	Header Data: Purchase Order		OK	OK	Not Performed	

16. SAP TAO – Build & Execute Test Script

Test building is done in SAP Quality Center using SAP TAO. You need to consolidate test components to create test scenarios. You can execute a **single or multiple tests** in SAP TAO using Technical Bill of Material TBOM.

Single Test Execution

Technical Bill of Material (TBOM) is used to contain the objects in an executable form. Change Analyzer makes use of this to tell if an executable object is affected when a change is performed.

In case you want to use the Business Process Change Analyzer, you need to generate a TBOM for each executable object in test scenarios and processes.

Note: If you need to run a single test and you have to update TBOM, click **Execute and update TBOM**.

You can check the details of update on the TBOM page. You can modify the run list that is created in SAP TAO.

Multiple Tests Execution

In Business Process Change Analyzer, to execute multiple tests, you can select a folder and add it to the run list. You can also select TBOM creation at the time of execution.

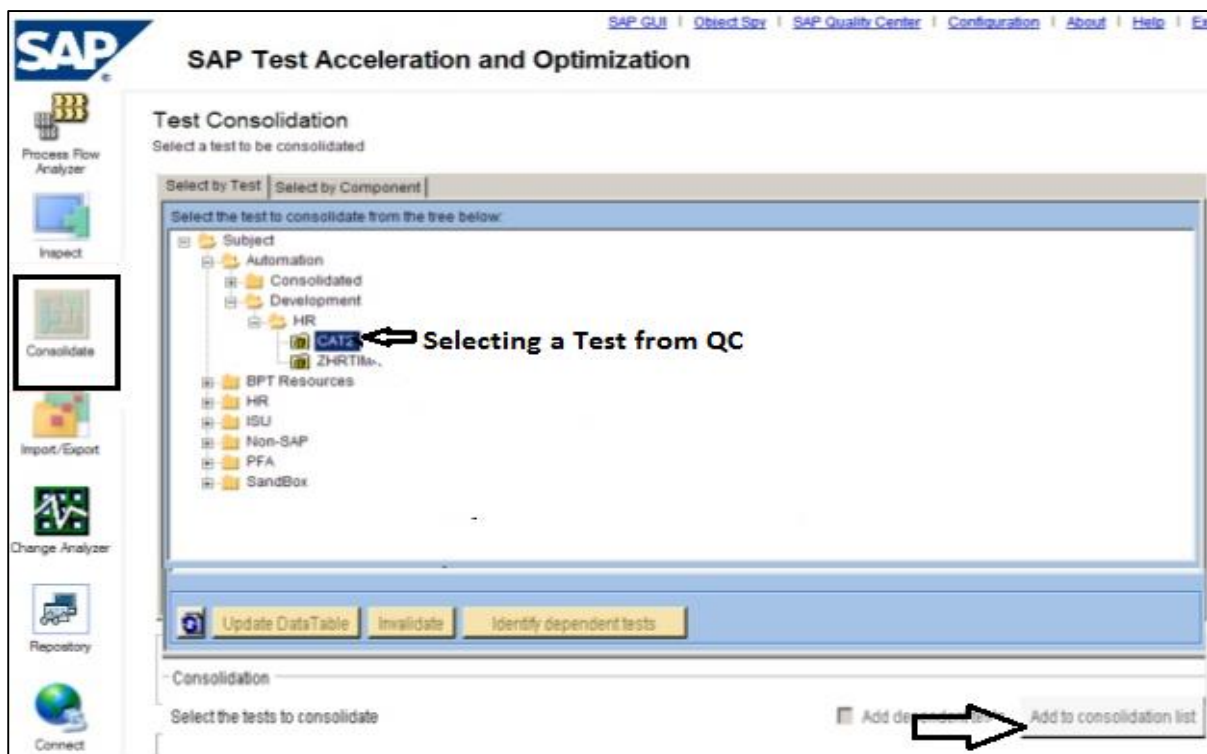
Note: If TBOM already exists at time of execution, it will only update the existing TBOM.

17. SAP TAO – Consolidate

Consolidate is known as a process to combine SAP TAO components with inbuilt components to create test scenarios as single transactional business components. It allows you to collect multiple test components into one test.

It happens when transactional components are gathered. The following screenshot shows the transaction components in QC.

You need to find the test in Quality Center from QC tree and click the Consolidate option.



Select the Transaction code you want to consolidate and click **Add to Consolidate list** at the bottom to create a test scenario.

As covered in the previous chapters, you need to follow the steps given below:

- **Step 1:** Find out the transaction you want to consolidate in QC.
- **Step 2:** Add the transactions to consolidate list.
- **Step 3:** Press the consolidate button.

18. SAP TAO – UI Scanner

UI scanner is used to create new screen components with existing components. It is a plugin for QTP tool. You should try to use inspect tab over UI scanner. The standard UI scanner works only with GUI front-end client. You can also use third-party UI scanner for capturing the screen components.

If you have to use UI scanner, you need to activate it in the Inspection tab of SAP TAO tool. UI scanner is used to get the information from one screen in one go and transfer these screen objects to QC as a screen component.

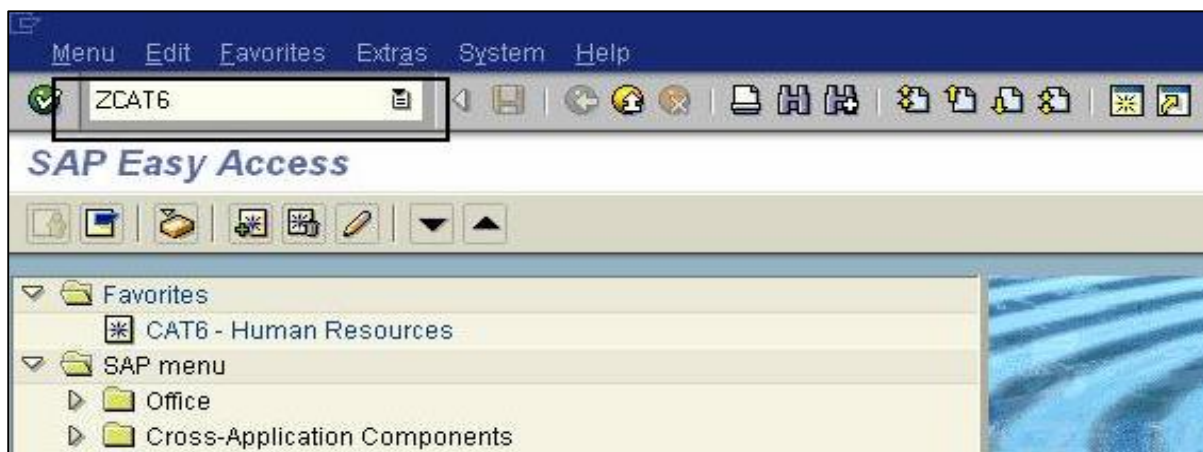
UI scanner allows you to create components from SAP GUI screen which are not supported by Process Flow Analyzer or Inspection tab.

How to use UI Scanner with SAP TAO?

When you login to SAP TAO, click the **Inspect** tab. It will show an option to use UI Scanner.



Login to the SAP system. Enter the transaction code and go to the screen to be scanned and log off.



You can use the default UI Scanner option with QTP tool, by going to UI Scanning tab under Inspect option in SAP TAO. Otherwise, you can use custom QTP test with UI Scanner process by going to UI Scanner and selecting external option.

19. SAP Testing – Automation Tools

In Automation testing, the testers write the scripts and use other software tools to test the product. This process involves automation of a manual process. In comparison, manual testing is time-consuming and requires a team of experience test professionals, subject matter experts, and effective communication between the team members.

Automation Testing includes re-running the test-cases multiple times that were performed manually.

In addition to Regression testing, you can say that Automation testing is also used to test the application from load, performance, and stress purpose. It is used to increase the coverage of test, improves accuracy, and saves time and money in comparison to manual testing.

Software Testing Tools

The following tools can be used for Automation testing:

- HP Quick Test Professional (QTP)
- Selenium
- SAP TAO
- ECATT
- IBM Rational Functional Tester
- SilkTest
- TestComplete
- Testing Anywhere
- WinRunner
- LoadRunner
- Visual Studio Test Professional
- WATIR

20. SAP Testing – Solution Manager

To perform SAP Automation testing, there is a need to set up a communication between SAP TAO, SAP Solution Manager, and the system to be tested.

To set up this configuration, you need an administration authorization on SAP SOLMAN. This is required to access and modify data table content.

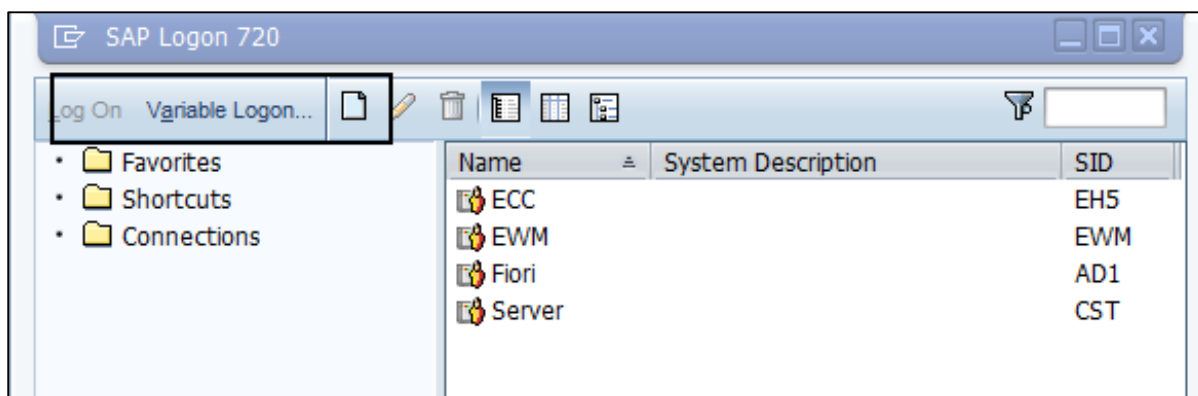
T-Code: SE16

You need to display the table name: AGS_TAO_SETTING and change the value for AGS_TAO_ENABLE_SM_SETUP to ON.

Next, check the compatibility matrix for SAP TAO and Solution Manager. SAP TAO 3.0 contains a Process Flow Analyzer recording wizard which can be used to ease the test recording.

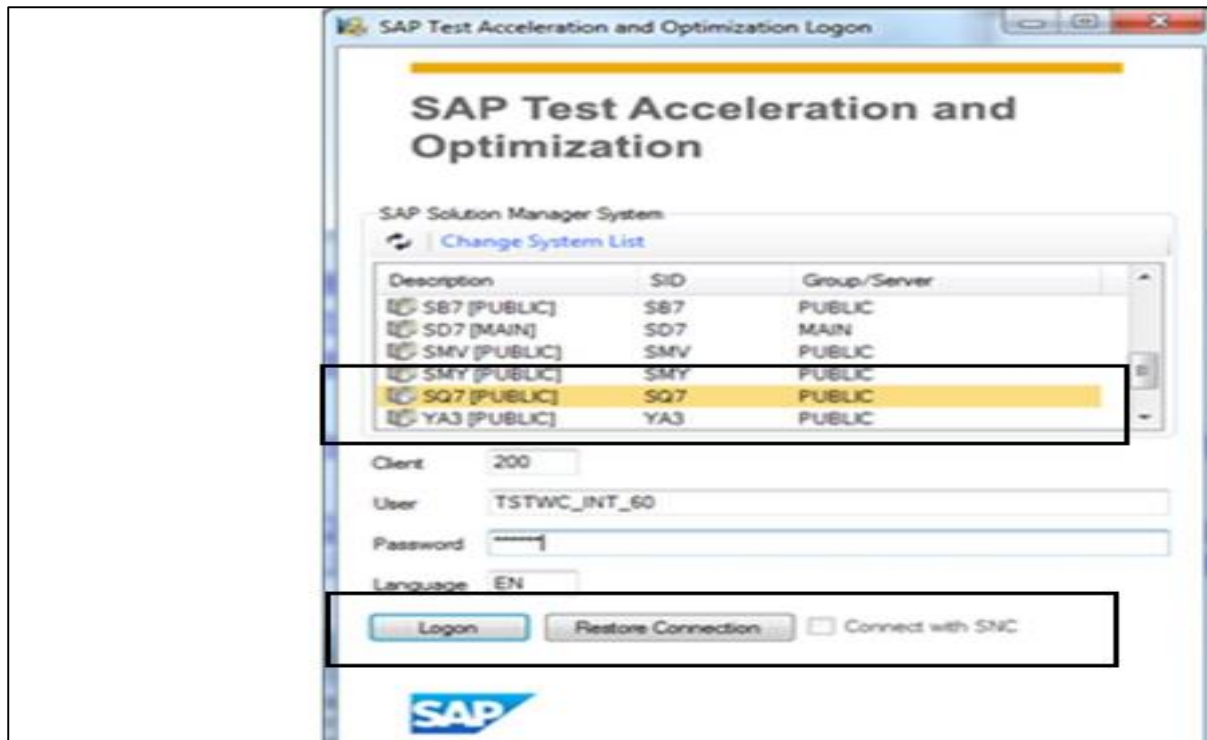
Note: SAP TAO 3.0 is a component for Solution Manager 7.1 and does not support Solution Manager 7.0. Open SAP Solution Manager Configuration work center and run Transaction code: SOLMAN_SETUP.

Open SAP Logon and add the system to be tested under logon.



How to Connect SAP SOLMAN to TAO?

Go to SAP TAO and login and select SAP SOLMAN in the list. This list of system in SAP TAO is fetched from the configuration file of SAP logon. To add a system in SAP TAO, you need to add a new system in SAP Logon and refresh the list in TAO.



Enter the login credentials and click on logon. SAP TAO will be connected to SAP Solution Manager and TAO configuration wizard will open up.

21. SAP Testing – HP Business Process Testing

Business Process Testing (BPT) is a part of QTP automation framework and is used with the Quality Center by HP. BPT is used to create automation test scenarios and run those scenarios without any prior knowledge of automation.

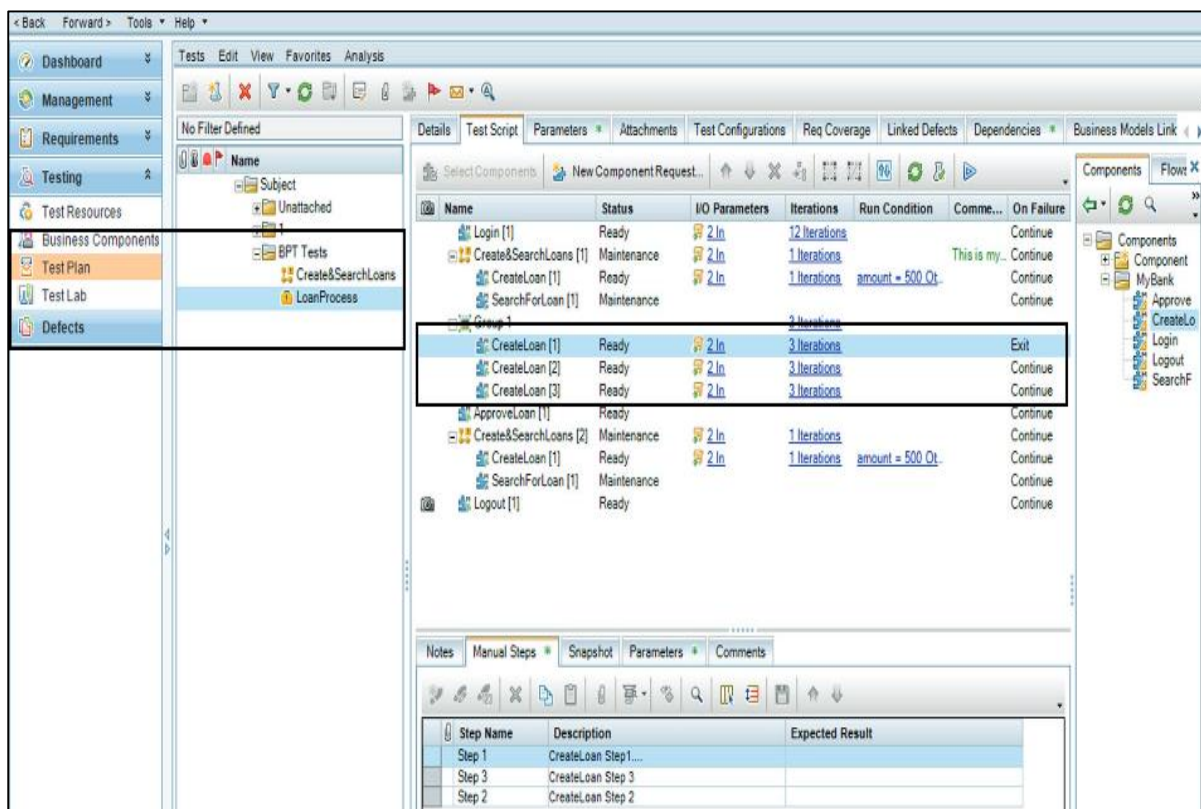
HP BPT removes the complexity of test-case creation and maintenance and combines all the documentation and test automation in one effort.

Business Process Testing aligns the testing process with business goals and reduces the testing lifecycle time considerably.

How to Use HP Business Process Testing Tool?

Business Process Testing tool uses reusable components for creating test-cases and hence reduces the testing maintenance time and increases the efficiency of testing process.

To fasten the process of test automation, it uses the method of keyword driven. You can add common best practices to the testing process. It allows you to use a test solution which is not based on test scripts. Once a manual test is created in BPT, you can easily automate the test-case.



You can also maintain different versions and baseline for different test components, process flows without any chance of overwriting the old cases.

Test cases that are created using Business Process Testing tool can be executed using HP Quality Center QC.

Benefits of Using HP Business Process Testing

Using BPT, a non-technical SME can easily create, maintain, and run the test-cases and can document them in a Web-based system.

- It allows you to design and create the reusable components in test-cases and use them as per the business requirements.
- You can also run testing scripts using HP Sprinter.
- With the availability of framework to use reusable components, it decreases the effort for maintaining the test-cases.

HP Quality Center (ALM)

HP Quality Center, a test management tool, is now popularly known as **Application Life Cycle Management** (ALM) tool, as it is no longer just a test management tool but it supports various phases of the software development life cycle.

HP-ALM helps us to manage project milestones, deliverables, resources and keeping track of project health, standards that allow Product owners to gauge the current status of the product. It is important to understand history, architecture, and Quality Center Workflow.

The Requirements module enables users to define, manage and track requirements at all stages of the software lifecycle. The following are the key functionalities in requirements module.

Functionality	Description
Creating Requirements	Create requirements, assign to releases/cycles.
Uploading Requirement	Uploading requirements using ALM-MS Excel Add-ins
Requirement Traceability	Enables how to define traceability links between requirements and dependencies that exist between the requirements.
Traceability Matrix	Enables user to view the traceability matrix that lists source requirements and their associated requirements and tests.

Test Plan

The crucial step in testing any application is to develop a clear and a precise test plan. A good test plan enables the team to assess the quality of the application under test at any point in the software development life cycle.

Following functionalities are very important in order to understand the test plan module better.

Functionality	Description
Creating Tests	This module describes how to create folders of test subjects in the test plan tree and also to add tests.
Uploading Tests	Uploading Tests using ALM-MS Excel Addins
Requirement and Test Coverage	Enables how to define the relationship between the requirements and tests.
Test Configuration	Specifies the subset of data or a run-time environment that the test should use.

More details on how HP QC, please go to: <http://www.tutorialspoint.com/qc/index.htm>

22. SAP Testing – QTP

HP Quick Test Professional (QTP) is an automated functional testing tool that helps testers to perform automated regression testing in order to identify any gaps, errors/defects in contrary to the actual/desired results of the application under test.

Object Repository

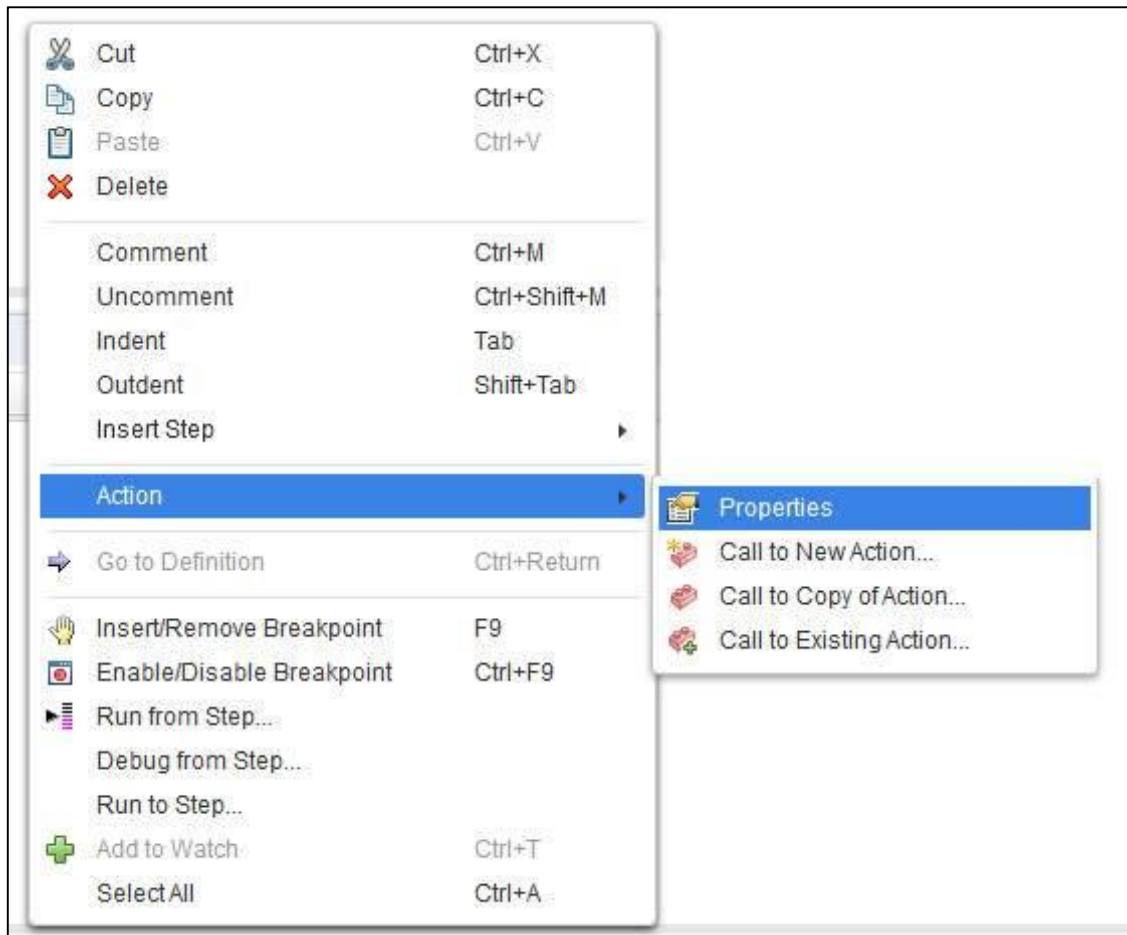
Object Repository is a collection of objects and properties with which QTP will be able to recognize the objects and act on it. When a user records a test, the objects and its properties are captured by default. Without understanding the objects and its properties, QTP will NOT be able to play back the scripts.

Actions

Actions help testers to divide scripts into groups of QTP statements called **actions**. Actions are similar to functions in VBScript, however there are a few differences. By default, QTP creates a test with 1 action.

Actions	Functions
Actions are inbuilt feature of QTP.	VBScript Functions are supported by both VBScript and QTP.
Actions parameters are passed byvalue only.	Function parameters are passed either by byvalue or byref.
Actions have extension .mts	Functions are saved as .vbs or .qfl
Actions may or maynot be reusable.	Functions are always reusable.

The properties of the action can be accessed by right-clicking the Script Editor Window and selecting "Properties".



Action properties contain following information-

- Action Name
- Location
- Reusable Flag
- Input Parameters
- Output Parameters

Types of Actions

There are three types of actions:

- **Non-reusable action** – An action that can be called only in that specific test in which it has been designed and can be called only once.
- **Reusable action** – An action that can be called multiple times any test in which it resides and can also be used by any other tests
- **External Reusable action** – It is a reusable action stored in another test. External actions are read-only in the calling test, but it can be used locally with the editable copy of the Data Table information for the external action.

For more information, please use the link: <http://www.tutorialspoint.com/qtp/index.htm>

23. SAP Testing – Case Study

Most of the companies that implement SAP need to perform testing. As the scope of testing is too large, an automated approach can be followed to maintain the changes in SAP system.

Various companies have designed their internal solutions to meet client requirements to perform SAP Testing. Clients can be from banking, finance, manufacturing or healthcare domain.

Example

Given below is an example of performing SAP testing for a manufacturing company.

Client Requirement: The client is a UK based manufacturing company. Project requirement was to perform SAP testing using QTP and to perform automation and functional testing key operations in field of Human Resource, Supply Chain, Logistics, Material Management and Plant maintenance and to use automated test cases for SAP upgrade and to perform integration and Regression testing.

Tasks Performed: It started with understanding of key business processes and SAP system tasks to be automated. Testing team referenced an old pilot project to finalize test strategy, time and effort required to run test execution in HP QTP tool.

As Part of project implementation 100 business processes were successfully automated. Implemented solution resulted in faster execution, more accuracy, increased scope and quality of service.

Tools Used: The following tools were used: SAP R/3, HP QTP, Test scripts written in VB, and Data in XML and XLS format.

Key Benefits Achieved: The following benefits were achieved:

- System Validation
- Quality and Revenue
- Cost and Predictability
- Compliance Management
- New Implementation and Configuration Changes

24. SAP Testing – Interview Questions

1. What do you understand by SAP Testing?

SAP Testing is about testing the functionality of various SAP modules to ensure that they are performing as per the configuration. SAP system undergoes various changes like patch management and fixes, new module implementations and various other configuration changes. All these raise a need for Regression testing to be performed in SAP environments. SAP testing automation tools like SAP TAO can be used for this purpose and is recommended by SAP for testing.

2. Explain the key benefits of performing SAP Testing.

The benefits of performing SAP Testing are manifold. They are as follows:

- **System Validation:** SAP Testing involves complete end to end testing and validation of all SAP modules in SAP ERP environment.
- **Quality and Revenue:** SAP Testing is an output based testing and not like conventional testing methods which are input based and it ensures the quality of SAP system and also focuses on revenue and cost of the organization.
- **Cost and Predictability:** SAP Testing involves reducing the SAP development costs and improve predictability.
- **Compliance Requirement:** SAP Testing ensures that SAP implementation is meeting the new compliance requirements in a specific organization and all modules are working as per expected configuration.
- **New Implementation and Configuration Changes:** There are different type of changes implemented in SAP system, like patches and fixes, new implementation, configurational changes. SAP testing ensures that all the modules are performing as per requirement in this dynamic system environment.
- **Integration:** SAP testing is performed to test various reports, data flows and work flows, GUI forms, etc. It is used to check system integration between different modules. For example, if an order posting is done that requires an action in Sales and Distribution, MM and FICO so SAP testing check the integration between these systems.
- **Performance:** It is also used to ensure if system will be able to meet the Service Level agreements, time taken by system to perform specific actions, performance of the system, etc.

3. Name a few modules on which SAP testing can be performed.

SAP testing can be performed on the following modules:

- SAP Material Management
- SAP Financial Accounting and Controlling
- SAP Sales and Distribution
- SAP Human Resource
- SAP Supply Chain Management
- SAP Plant Management

4. Name a few Automation testing tools.

Some of the automating testing tools are: HP Quick Test Professional (QTP), Selenium, SAP TAO, ECATT, IBM Rational Functional Tester, WinRunner, and LoadRunner.

5. What are the different stages involved in Software Testing life cycle?

The different stages that come under Software Testing Life Cycle are: Requirements phase, Test Planning, Test Analysis, Test Design Phase, Test Implementation, Test Execution Phase, and Test Closure Phase.

6. What is the use of Unit testing in SAP? Explain with an example.

Unit testing is used to test the functionality of various components in a SAP system. It is performed by domain and configuration experts who know the functionality of each unit in a system.

Example: To create a sales order and to save it. To perform Unit testing for this task, tester should know that the sales order can be saved using the SAP organization elements like customer master data, partner functions, material master data, company code, credit control area, sales organization, etc.

7. Explain unit testing with an example.

Unit testing is used to test functionality of pieces in SAP system. It is performed by domain and configuration expert who knows functionality of each unit in a system.

Example: To create a sales order and to save it. To perform Unit testing for this task, tester should know that the sales order can be saved using the SAP organization elements like customer master data, partner functions, material master data, company code, credit control area, sales organization, etc.

8. Explain System Testing with an example.

System Testing involves integration of elements of SAP system to ensure that related SAP functionality are linked together in the development environment.

Example: If you say a Cash flow for a quotation in an organization would show that a quote can be used to create a sales order, a delivery can be created and processed from the order, the delivery can be billed, the billing released to accounting, and a customer

payment applied against the accounting invoice. Each unit is tested like this and then the test results are combined.

9. Explain Scenario-based Testing with an example.

Scenario-based testing, as the name suggests, is performed as per specific business cases.

Example: Suppose there are a few tasks that are specific to a customer segment or a given product line or a set of services. For these specific line of target, you have different scenarios you need to test.

This testing is also performed in the development environment, an argument can be made to say this is a test case you would cover in system testing.

10. Explain Integration Testing with an example.

In this testing, testing data is coming from a real data extraction source, conversion is done and load exercise and data is known to a business end user.

Example: Integration testing is used to present that the business process as designed and configured in SAP, runs using real world data. In addition the testing shows that interface triggers, reports, workflows are working.

11. Explain SAP Interface Testing.

Interface testing ensures that a business process on a SAP system runs automatically. Ideally interface testing involves larger testing activities as a project progresses. Interface testing shows that triggering works, the data selection is accurate and complete, data transfer is successful, and the receiver is able to consume the sent data.

12. Where is SAP User Acceptance Testing useful?

SAP UAT is used to ensure that end-users are able to perform assigned job functions with the new system. The important aspect of this testing is to understand the business requirement and to ensure that the expected features, functions, and capabilities are available.

13. Why is Performance Testing necessary?

Performance testing identifies bottlenecks and coding inefficiencies in a SAP system. It is carried out to check:

- Whether the system response time is acceptable as per the business requirement
- Whether periodic processes are running within permissible time
- Whether the expected concurrent user load can be supported

14. What is the importance of SAP Security and Authorizations Testing?

Security and Authorizations Testing is used to ensure that the users are only able to execute transactions and access appropriate data that is relevant to their project.

As with the implementation of Security standards, this is really important to test if security and authorization is placed in a system. Test IDs for job roles are created and used to both confirm what a user can do and what a user cannot do.

15. Explain the use of SAP Cutover Testing.

This testing is usually performed once in a project lifecycle. The term “cutover” means a full scale execution of the all tasks involved to extract data from legacy systems and then to perform any kind of data conversion, load the results into the SAP system and fully validate the results, including a user sign-off.

16. What is SAP Regression Testing?

SAP Regression Testing is used to find new functionalities and to test the old functionalities in a system when it is upgraded or a new system is set up. The key role of regression testing is to test the existing functionality and newly updated configuration and code base.

When you upgrade your SAP system or apply a patch, it shouldn't affect the functionality that is expected to be performed by users and to check new features that are supposed to be introduced in new release.

17. What is the purpose of using Navigation in SAP Testing?

SAP testing navigation ensures that you cover each module of your SAP system and at least one test to be performed for each functionality.

It also reduces the manual testing effort and covers most of the testing paths in a SAP system. OPA tests can be performed to check SAP Testing- Navigation.

18. What is Screen flow logic in SAP Testing?

Screen flow logic in SAP Testing is like an ABAP code and it is used to contain the processing blocks. It contains procedural part of screen and is created in screen painter and this screen painter is similar to an ABAP editor.

19. What are the different modules in a SAP system and their use?

Financial Modules: Finance Accounting and Controlling (FICO), Treasure (TR), and Enterprise Control (EC).

Logistics Modules: Material Management (MM), Plant Maintenance (PM), Sales and Distribution (SD), Quality Management (QM), etc.

Human Resource Management: Accounting Payroll, Time Management, Training and Event Management.

20. What is a test-case? What would be the scope of test-cases in a SAP system?

SAP Test-cases are required to perform a check on the installation and configuration of SAP system, any new implementation, Multilanguage and device testing, intranet testing, real-time testing, etc.

Since an ERP system is a common centralized system and is used by multiple users simultaneously in real time, it creates a need to write the test cases with lot of effort and dedication.

ERP systems also involve various FI transactions, so each test case should cover the scope of all the configuration and implementation part. Test data should be passed carefully and each test should have a column with name output data.

21. What is SAP Test Acceleration and Optimization?

SAP Test Acceleration and Optimization TAO 2.0 is a tool that is used to create automatic test-cases during Regression testing of a system. It helps SAP customers to create different test components from the screens of a transaction and parameterizes them.

These test components are created normally for one transaction code and later can be combined to test scenarios. It can be easily integrated to the Business Process Change Analyzer in SAP Solution Manager.

22. What is the minimum hardware requirement to perform SAP TAO installation on a local system?

- RAM: 4 GB
- Free disk space: 500 MB
- LAN connection to SAP Quality Center QC server and SOLMAN
- Administration Rights
- SAPGUI with the latest patch level
- Microsoft Excel 97 or higher.
- Microsoft Internet Explorer for CRM UI support

23. What is the use of Process Flow Analyzer in SAP TAO?

Process Flow Analyzer is used to automatically find out the user interfaces used in transaction codes executed in a SAP system. It automatically creates the test components and upload them to Quality Center. It is also used to identify the sequence of test components as per user actions and creation of spreadsheet values.

24. What is the purpose of consolidation tab in SAP TAO?

Consolidate is known as a process to combine SAP TAO components with inbuilt components to create test scenarios as single transactional business components. It allows you to collect multiple test components into one test.

25. How to connect to SAP TAO to SAP solution Manager?

Go to SAP TAO and login and select SAP SOLMAN in the list. This list of system is SAP TAO is fetched from configuration file of SAP logon. So to add a system in SAP TAO, you need to add a new system in SAP Logon and refresh the list in TAO.

Enter the login credentials and click on logon.

SAP TAO will be connected to SAP Solution Manager and the TAO configuration wizard will open up.