Focused Build for SAP Solution Manager 7.2 (SP5)
Test Management

January 2019
Disclaimer

This presentation outlines our general product direction and should not be relied on in making a purchase decision. This presentation is not subject to your license agreement or any other agreement with SAP. SAP has no obligation to pursue any course of business outlined in this presentation or to develop or release any functionality mentioned in this presentation. This presentation and SAP's strategy and possible future developments are subject to change and may be changed by SAP at any time for any reason without notice. This document is provided without a warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement. SAP assumes no responsibility for errors or omissions in this document, except if such damages were caused by SAP intentionally or grossly negligent.
Agenda

Introduction:
- Capabilities, Test Strategy, Test Types, Branches & Test Systems in Focused Build

How-to:
- Unit Test
- Single Functional Test & Acceptance Test
- Functional Integration Test
- Regression Test

Summary

What’s New with Focused Build SP5

Excursus:
- Test Steps, Technical Preparation
Introduction:
Test Strategy, Test Types, Branches & Test Systems in Focused Build
On Top Functionality of Focused Build for SAP Solution Manager

- Requirements / Work Package based Test Plan Generation
- Test Steps Designer
- Test Execution: My Test Executions
- Test Suite Dashboard

SAP Solution Manager – Test Suite

- Solution Documentation
- Change Impact Analysis
- Test Planning
- Test System Setup
- Test Execution and Analytics

- Manual Test Cases
- Automated Test Cases
- Test Automation Framework with 3rd party test tool integration
- Test Composition Environment to manage SAP and partner tools
- Test Data Management
- Change Impact Analysis
- Test Plan and Test Package Management
- Test Execution of manual and automated tests
- Gap and Completeness Analytics
- Test Execution Analytics
Focused Build for Application / Software

**PMO**
- Create project plan
  - Track project readiness, issues and risks
  - Manage scope change

**Validation Team (Business)**
- Validation WS for requirements

**Build Team (Plan)**
- Create work package
- Assign work items & build teams
- Manage show & tell

**Build Team (Development)**
- Develop, configure, unit test
- Document work item & progress
- Release work item
- Defect correction
  - Plan & execute SFT, FIT, UAT, RT (per wave)
  - Manage defects

**Test Management**
- Manage defects

**Release Management**
- Build, test and validate release
- Manage hyper-care

**PMO**
- Manage q-gate deliverables, sign-off
- Hand-over release to customer

**Validation Team (Business)**
- Create work package

**Build Team (Plan)**
- Assign work items & build teams
- Develop, configure, unit test
- Document work item & progress
- Release work item
- Defect correction

**Test Management**
- Manage defects

**Release Management**
- Build, test and validate release
- Manage hyper-care

---

- **PMO**
  - Create project plan
    - Track project readiness, issues and risks
    - Manage scope change

- **Validation Team (Business)**
  - Validation WS for requirements

- **Build Team (Plan)**
  - Create work package
  - Assign work items & build teams
  - Manage show & tell

- **Build Team (Development)**
  - Develop, configure, unit test
  - Document work item & progress
  - Release work item
  - Defect correction
    - Plan & execute SFT, FIT, UAT, RT (per wave)
    - Manage defects

- **Test Management**
  - Manage defects

- **Release Management**
  - Build, test and validate release
  - Manage hyper-care

**PMO**
- Manage q-gate deliverables, sign-off
- Hand-over release to customer

**Validation Team (Business)**
- Create work package

**Build Team (Plan)**
- Assign work items & build teams
- Develop, configure, unit test
- Document work item & progress
- Release work item
- Defect correction

**Test Management**
- Manage defects

**Release Management**
- Build, test and validate release
- Manage hyper-care

---

- **PMO**
  - Create project plan
    - Track project readiness, issues and risks
    - Manage scope change

- **Validation Team (Business)**
  - Validation WS for requirements

- **Build Team (Plan)**
  - Create work package
  - Assign work items & build teams
  - Manage show & tell

- **Build Team (Development)**
  - Develop, configure, unit test
  - Document work item & progress
  - Release work item
  - Defect correction
    - Plan & execute SFT, FIT, UAT, RT (per wave)
    - Manage defects

- **Test Management**
  - Manage defects

- **Release Management**
  - Build, test and validate release
  - Manage hyper-care

**PMO**
- Manage q-gate deliverables, sign-off
- Hand-over release to customer

**Validation Team (Business)**
- Create work package

**Build Team (Plan)**
- Assign work items & build teams
- Develop, configure, unit test
- Document work item & progress
- Release work item
- Defect correction

**Test Management**
- Manage defects

**Release Management**
- Build, test and validate release
- Manage hyper-care

---

- **PMO**
  - Create project plan
    - Track project readiness, issues and risks
    - Manage scope change

- **Validation Team (Business)**
  - Validation WS for requirements

- **Build Team (Plan)**
  - Create work package
  - Assign work items & build teams
  - Manage show & tell

- **Build Team (Development)**
  - Develop, configure, unit test
  - Document work item & progress
  - Release work item
  - Defect correction
    - Plan & execute SFT, FIT, UAT, RT (per wave)
    - Manage defects

- **Test Management**
  - Manage defects

- **Release Management**
  - Build, test and validate release
  - Manage hyper-care

**PMO**
- Manage q-gate deliverables, sign-off
- Hand-over release to customer

**Validation Team (Business)**
- Create work package

**Build Team (Plan)**
- Assign work items & build teams
- Develop, configure, unit test
- Document work item & progress
- Release work item
- Defect correction

**Test Management**
- Manage defects

**Release Management**
- Build, test and validate release
- Manage hyper-care

---

- **PMO**
  - Create project plan
    - Track project readiness, issues and risks
    - Manage scope change

- **Validation Team (Business)**
  - Validation WS for requirements

- **Build Team (Plan)**
  - Create work package
  - Assign work items & build teams
  - Manage show & tell

- **Build Team (Development)**
  - Develop, configure, unit test
  - Document work item & progress
  - Release work item
  - Defect correction
    - Plan & execute SFT, FIT, UAT, RT (per wave)
    - Manage defects

- **Test Management**
  - Manage defects

- **Release Management**
  - Build, test and validate release
  - Manage hyper-care

**PMO**
- Manage q-gate deliverables, sign-off
- Hand-over release to customer

**Validation Team (Business)**
- Create work package

**Build Team (Plan)**
- Assign work items & build teams
- Develop, configure, unit test
- Document work item & progress
- Release work item
- Defect correction

**Test Management**
- Manage defects

**Release Management**
- Build, test and validate release
- Manage hyper-care
Example of Project Structure
Build Projects for Waterfall Approach

Project structure sample with 1 wave and 1 sprint

Wave 1
- Scope defined
- Functional Specification available
- Technical Design available
- Work Package Build completed
- Unit Test completed
- Work Item Build completed

Sprint 1
- Single Functional Test completed

Acceptance Test
- Functional Integration Test
- Functional Integration Test (final) & Regression Test

Handover to Release

Discover & Prepare
Explore
Realize
Deploy
Run
# Testing Types

## Definition

<table>
<thead>
<tr>
<th>Test Types</th>
<th>Test Level</th>
<th>Test Requirement</th>
<th>Definition</th>
<th>Test System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit Test</td>
<td>Sprint</td>
<td>Work Item (Tech. Design)</td>
<td>Technical feature test to ensure that the feature was implemented correctly.</td>
<td>QAS</td>
</tr>
<tr>
<td>Single Functional Test</td>
<td>Wave</td>
<td>Work Package (Func. Spec)</td>
<td>Business function test to ensure that the feature was implemented correctly. Depending on the project approach the Single Functional Test might be informal (without Test Plan creation).</td>
<td>QAS</td>
</tr>
<tr>
<td>Acceptance Test</td>
<td>Wave</td>
<td>Single Work Packages (Requirement)</td>
<td>Validation of requested functionality by requestor. After validation, corresponding work packages and defect corrections would be ready for handover to release. This is the most important test, as after setting the Work Package status Hand over to release, all assigned transports will end in PRD system.</td>
<td>QAS</td>
</tr>
<tr>
<td>Functional Integration Test</td>
<td>Project (Optional)</td>
<td>Related Work Packages Process with requirements</td>
<td>Early validation of modular processes.</td>
<td>QAS</td>
</tr>
<tr>
<td>Functional Integration Test</td>
<td>Release (Final)</td>
<td>E2E Processes</td>
<td>E2E tests of the new or extended functionality validating the end to end business process. Final hands-on validation by business users.</td>
<td>PRE</td>
</tr>
<tr>
<td>Regression Test</td>
<td>Release</td>
<td>Productive Processes</td>
<td>Test of productive processes or functions.</td>
<td>PRE</td>
</tr>
</tbody>
</table>
Test Strategy: Test Types, Test Systems
Scaled Agile Example: One Release per project, two Waves, and two Sprints per Wave

- Prepare
- Build
- Test
- Deploy
- Hypercare

Project Duration

Project Phase/SAP Activate Phase

Release Phase

Prepare

Build

Test

Deploy

Run

Wave 1
- Requirements
- Work Packages
- Work Items
- Defect Corrections

Wave 2
- Requirements
- Work Packages
- Work Items
- Defect Corrections

- Single Functional Test
- Functional Integration Test (Optional)
- Acceptance Test
- Regression Test

Sprint 1
- Unit Test

Sprint 2
- Unit Test

QAS

PRE

© 2019 SAP SE or an SAP affiliate company. All rights reserved. | PUBLIC
Requirements based Test Approach
All Test Types

- **Definition and Specification**
  - Requirements
  - Work Packages (Functional Specification)

- **Implementation and Configuration**
  - Work Items (Technical Design)
  - Software Implementation and related Configuration

- **Functional Tests**
  - Single Functional Tests
  - Unit Tests

- **Process Tests**
  - Functional Integration Tests (optional)
  - Functional Integration Tests (final)

- **PRD use**
  - Acceptance Test
  - Regression Tests (optional)
  - Regression Tests (final)

© 2019 SAP SE or an SAP affiliate company. All rights reserved. I PUBLIC
Focused Build methodology
Standard Workflow

Business Analyst/Process Expert
Approved → In Realization

Solution Architect
Created → Scoping → Scope Finalized → To Be Developed → In Development

Test Coordinator
Single Functional Test & AT
In Repair → To Be Tested → Successfully Tested

Technical Tester (Developer)
Unit Test
Created → In Development → To Be Tested → Successfully Tested

Developer
Created → In Development → To Be Tested → Successfully Tested

Realized → Completed
FIT/RT
Handed over to Release

In Realization → Productive/Completed

In Development → Realized

To Be Tested → Successfully Tested

In Development → Completed

To Be Developed → Successfully Tested

To Be Developed → Handed over to Release

Successful"
Focused Build methodology – Branches
Standard Workflow

Design Branch
- Approved
- In Realization

Development Branch
- Created
- Scoping
- Scope Finalized
- To Be Developed
- In Development
- Single Functional Test & AT
  - In Repair
  - To Be Tested
  - Successfully Tested
- To Be Tested
- Successfully Tested
- Created
- In Development
- Unit Test
  - To Be Tested
  - Successfully Tested

Production Branch
- Realized
- Completed
- FIT/RT
- Handed over to Release
- Productive/Completed

- = Automated action
- = Manual action
- = WI with Normal Change
- = WI with General Change

Work Package
- Requirements
- Work Item
Focused Build methodology – Systems
Standard Workflow
Focused Build methodology – Release Phase
Standard Workflow

Planned
- Approved
- In Realization
- Create Work Package

Prepare
- Created
- Scoping
- Scope Finalized
- To Be Developed
- In Development
- Single Functional Test & AT
  - In Repair
  - To Be Tested
  - Successfully Tested

Build/Test
- Realized
- FIT & RT
  - Handed over to Release
- Completed

Deploy
- Completed
- Productive/Completed

Requirements
- Approved
- In Realization
- Create Work Package

Work Package
- Created
- Scoping
- Scope Finalized
- To Be Developed
- In Development
- Single Functional Test & AT
  - In Repair
  - To Be Tested
  - Successfully Tested

Work Item
- Created
- In Development
- Unit Test
  - To Be Tested
  - Successfully Tested

Create Work Package
Create Work Item
Create Transports
Create Transports
Hand over to Release
Create Transports
Hand over to Release
Create Transports
# Unit Test

<table>
<thead>
<tr>
<th>Test Types</th>
<th>Test Level</th>
<th>Test Requirement</th>
<th>Definition</th>
<th>Test System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit Test</td>
<td>Sprint</td>
<td>Work Item (Tech. Design)</td>
<td>Technical feature test to ensure that the feature was implemented correctly</td>
<td>QAS</td>
</tr>
<tr>
<td>Single Functional Test</td>
<td>Wave</td>
<td>Work Package (Func. Spec)</td>
<td>Business function test to ensure that the feature was implemented correctly</td>
<td>QAS</td>
</tr>
<tr>
<td>Acceptance Test</td>
<td>Wave</td>
<td>Single Work Packages (Requirement)</td>
<td>Validation of requested functionality by requestor. After validation, corresponding work packages and defect corrections would be ready for handover to release. This is the most important test, as after setting the Work Package status Hand over to release, all assigned transports will end in PRD system</td>
<td>QAS</td>
</tr>
<tr>
<td>Functional Integration Test (Optional)</td>
<td>Project</td>
<td>Related Work Packages Process with requirements</td>
<td>Early validation of or modular processes.</td>
<td>QAS</td>
</tr>
<tr>
<td>Functional Integration Test (Final)</td>
<td>Release</td>
<td>E2E Processes</td>
<td>E2E tests of the new or extended functionality validating the end to end business process. Final hands-on validation by business users.</td>
<td>PRE</td>
</tr>
<tr>
<td>Regression Test</td>
<td>Release</td>
<td>Productive Processes</td>
<td>Test of productive processes or functions</td>
<td>PRE</td>
</tr>
</tbody>
</table>
Perform Unit Test
Start Unit Test by importing Transport of Copies into QAS

Steps
1. Developer selects Work Item.
2. Developer performs action ‘Pass To Test’:
   - Transport of Copies (ToC) will be imported to QAS system automatically
   - Unit test can be performed in QAS

Effects
- In Solution Readiness Dashboard, development will be shown as completed for this Work Item.

Recommendations
- Please make sure the unit tester is not the same user as the developer.
Perform Unit Test
Perform & confirm successful Unit Test

Steps
1. Tester executes Unit Test in QAS.
2. Tester selects Work Item.
3. (Optional): Tester documents test results.
4. Tester performs action ‘Confirm Successful Test’.

Effects
- In Solution Readiness Dashboard the Unit Test will be calculated as completed for this Work Item.
- The Work Package will be changed to ‘To be Tested’ automatically when all its Work Items are successfully tested.
- Transport(s) (TR(s)) will be released automatically and could be imported to QAS via scheduled job defined by Release Manager.

Recommendations
- Test results can be maintained as plain text in the Text tab using text type ‘Test Report’.
Steps

1. All Requirements should be completed.
2. All Functional Spec should be completed.
3. All Technical Design should be completed.
4. All Development should be completed.
5. All Unit Tests should be completed.

Check Q-Gate for Realize Phase Completed
Check Unit-Test milestone in Solution Readiness Dashboard
# Single Functional Test & Acceptance Test

<table>
<thead>
<tr>
<th>Test Types</th>
<th>Test Level</th>
<th>Test Requirement</th>
<th>Definition</th>
<th>Test System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit Test</td>
<td>Sprint</td>
<td>Work Item (Tech. Design)</td>
<td>Technical feature test to ensure that the feature was implemented correctly</td>
<td>QAS</td>
</tr>
<tr>
<td>Single Functional Test</td>
<td>Wave</td>
<td>Work Package (Func. Spec)</td>
<td>Business function test to ensure that the feature was implemented correctly</td>
<td>QAS</td>
</tr>
<tr>
<td>Acceptance Test</td>
<td>Wave</td>
<td>Single Work Packages (Requirement)</td>
<td>Validation of requested functionality by requestor. After validation, corresponding work packages and defect corrections would be ready for handover to release. This is the most important test, as after setting the Work Package status Hand over to release, all assigned transports will end in PRD system.</td>
<td>QAS</td>
</tr>
<tr>
<td>Functional Integration Test</td>
<td>Project</td>
<td>Related Work Packages Process with requirements</td>
<td>Early validation of or modular processes.</td>
<td>QAS</td>
</tr>
<tr>
<td>Functional Integration Test</td>
<td>Release</td>
<td>E2E Processes</td>
<td>E2E tests of the new or extended functionality validating the end to end business process. Final hands-on validation by business users.</td>
<td>PRE</td>
</tr>
<tr>
<td>Regression Test</td>
<td>Release</td>
<td>Productive Processes</td>
<td>Test of productive processes or functions</td>
<td>PRE</td>
</tr>
</tbody>
</table>
### Single Functional Test (SFT) and Acceptance Test (AT)

#### Three Variants supported by Focused Build - Overview

<table>
<thead>
<tr>
<th>Variant A1:</th>
<th>Without formal documentation, without test plan</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>w/o documentation of test results</strong></td>
<td><strong>not recommended</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variant A2:</th>
<th>With formal documentation, without test plan, possibility to create defect correction during Single Functional Test starting directly from the Work Package</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>w/ documentation of test results</strong></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variant A3:</th>
<th>With formal documentation, with test plan(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>w/ documentation of test results</strong></td>
<td></td>
</tr>
</tbody>
</table>

---

You can choose the variant fitting best to your individual test requirements.
Single Functional Test (SFT) and Acceptance Test (AT)
Variants A1 & A2 supported by Focused Build

**Variant A1 - Without formal documentation, without test plan**
- Use 'Show and Tell' sessions for testing
- In case of errors, create defect correction and retest
- Change status of each WP from 'To be Tested' to 'Successfully Tested', or use mass change functionality to change status of several WPs
- In very informal approaches not even test cases are required

**Variant A2 - With formal documentation, without test plan**
- Use 'Show and Tell' sessions for testing
- Make sure that 'Show and Tell' are guided by test cases added to the WPs
- After the sessions, add 'Test Notes' documenting results of sessions to each WP
- In case of errors, create defect correction directly from the work package in tab Scope
- Work Package will switch into status “to be tested” for retesting when Defect Correction is confirmed
- Change status of WP from 'To be Tested' to 'Successfully Tested' manually WP by WP
Single Functional Test (SFT) and Acceptance Test (AT)

Variant 3 supported by Focused Build

Variant A3 - With formal documentation, with test plan(s)

- Use test plan as Agenda for 'Show and Tell' sessions
- Add test cases to the WPs (Even in phase scoping a test case or an empty template can be uploaded)
- Create at least SFT Test Plan and/or an AT test plan via Assignment Analysis, which makes sure that the test plan is linked to the project. It is recommended to use the option '1 Test Package per WP' during creation of the test plan, as this option makes it easy to identify the WP that can be set to 'Successfully Tested' when the TP is finished. This is also required for automatic status change of WP, depending on Defect Correction status (WP 'in repair') and enables the linking of Defect Corrections automatically to WP, which makes sure that DC must be released / imported together with the WP
- **Remark:** In case the DCs are not finished (user status Confirmed) please refer to L2 presentation Defect and Defect Correction for SP04 for further details.

  *Only one Test Plan:*
  a) Use Test Plan only for AT. For SFT, follow A1 or A2.

  *Separate Test Plans for SFT and AT:*
  b) AT test plan contains a subset of test packages / test cases of SFT test plan but is executed by different testers (key user, person responsible for requirement)
  c) AT test plan is a 1:1 copy of SFT test plan but is executed by different testers (key user, person responsible for requirement)
  d) AT test plan contains different test cases than SFT test plan

- Ensure that status is set and 'Test Results' are documented per test case
- In case of errors create test defects, defect correction and retest
- Optional: Create final Test Report for Test Plan from Test Plan Analytics and add it to respective WPs
- Change status of WP from 'To be Tested' to 'Successfully Tested' only if exit criteria for test phase(s) are met. Mass change functionality for WPs could be used for this if no final Test Report is used
Steps

1. Use 'Show and Tell' sessions for testing
2. In case of errors, create test defects directly from the Work Package using the scope tab and retest
3. Change status of WP from 'To be Tested' to 'Confirm Successfully Test'

Remark

• In very informal approaches test cases are not required at all
Variant A2 - Single Functional Test and Acceptance Test
Update & finalize Test Cases, Execute Tests

Recommendation

- A Business Analyst should review the test cases available in Solution Documentation (SolDoc). If an update of existing test cases or the upload of additional test cases is required:
  - Test cases can be added directly in SolDoc
  - Test cases can be assigned to the respective Work Item (WI)
- Note: If Work Package (WP) is in Status ‘To be Tested’, a WI could be created directly in order to upload test cases. There’s no need to go back in the status ‘Scope extension’ to create the WI there. Even in phase scoping a test case or an empty template can be uploaded to the WP.

Steps to upload Test Case to WI:

1. Select Work Item.
2. Navigate to Documentation tab.
3. Drag and drop the test cases to the Documents area from user’s local computer.
4. Select New Version when using the same file name (Only for update).
5. Change test case status to ‘Released’ when finalized.
6. Set Work Package status to ‘Confirm Successful Test’ when single test execution is finalized.
All variants - Single Functional Test and Acceptance Test

Test Preparation

**Tab 'Test Preparation':**

- In this tile you can view the amount of Work Packages for a given Project and Wave.
- It will help you to easily identify the work packages to which no test cases (of any types) are assigned.
- It will also highlight the Work Packages which do not have any Test Case assigned according to their statuses.
- In this example, you can see that a Work Package in the status 'To Be Tested' which does not have a test case assignment is ranked as 'Red'.
- In the table you will find the Work Package details (ID, Name, Priority, Test Case Assignment, …).
- You can jump directly to the Work package Application to update any Work Package.
- You can also navigate to the 'Assignment Analysis' application to add any missing Test Cases.
Variant A3 - Single Functional Test (SFT) and Acceptance Test (AT)
Schematic view on test plan creation

In Focused Build projects, the creation of test plans is done with the Assignment Analysis and Test Plan Generation application. With this application, you can check the test plan coverage of the work packages of a project and create a test plan based on the test cases assigned to the work packages.
Variant A3 - Single Functional Test (SFT) and Acceptance Test (AT)
Sample Workflow

Solution Documentation
- Setup of Solution Documentation
- Setup of Project(s) and Release
- Create Requirements and Work Packages

Create / Assign Test Cases to Process Steps
- Assign Test Cases to Work Packages

Test Planning
- Check Assignment of Test Case(s) to Work Package (WP)
- Create Test Plan based on WP status
- Create / Adjust Test Packages
- Assign Tester(s)

Test Execution and Analytics
- Access My Test Executions (MTE)
- Execute Test Case(s) (manual / automatic)
- Document test results/ set status/ create defects
- Defect Correction (Developer provides fix)
- Retest and set test case status

Test Suite Dashboard
Test Suite Analytics
**Variant A3 - Single Functional Test (SFT) and Acceptance Test (AT)**

**Update & finalize Test Cases**

**Recommendation**
- A Business Analyst should review the test cases available in Solution Documentation (SolDoc). If an update of existing test cases or the upload of additional test cases is required:
  - Test cases can be added directly in SolDoc
  - Functionality ‘Test Steps’ can be used – see Excursus ‘Test Steps’
  - Test cases can be assigned to the respective Work Item (WI)
- Note: If Work Package (WP) is in Status ‘To be Tested’ a WI can be created directly in order to upload test cases. There’s no need to go back in the status ‘Scope extension’ to create the WI there.

**Steps**
1. Select Work Item (GC)
2. Navigate to Documentation tab
3. Drag and drop the test cases to the Documents area from user’s local computer
4. Select New Version when using the same file name (Only for update)
5. Change test case status to ‘Released’ when finalized
6. Go to Assignment Analysis for crating one or more Test Plans
Variant A3 - Single Functional Test (SFT) and Acceptance Test (AT)  
Create Test Plan(s)

Steps

1. Select Project and Wave.
2. Click ‘Assignment Analysis and Test Plan Generation’.
3. Select Work Package(s) which are required for SFT or AT.
4. Click Test Plan → ‘Save as new’.
5. Possibility to load and enhance an existing Test Plan (Save)

Remarks:

Further prerequisites are the Test Case Type ‘Additive’ in Solution Documentation and a document type ‘Test Document’ in transaction ‘SOLADM’ (see Excursus ‘Technical Prerequisites’ for more details).
Variant A3 - Single Functional Test (SFT) and Acceptance Test (AT)

Filter Function

- Work Packages part of the Project and Wave but not covered by any Test Plan are listed here.
- The Work Package ID can be used to navigate forward and find more details.
- There should be no WP in status ‘To Be Tested’, after test plan creation activities are finished.

Remarks:
Within the assignment analysis, filters e.g. on test types (SFT, FIT, UAT) are available, please make sure you maintain the respective Test Case Attributes.
Variant A3 - Single Functional Test (SFT) and Acceptance Test (AT)

Create Test Plan(s)

Steps

1. Input mandatory information for Test Plan.

2. Select ‘One Test Package per Work Package for Test Package Creation’.

Effects

- Test Plan automatically created
- Test Packages automatically created and adjustable
- Easy Tester Assignment via embedded Assign Testers functionality
Variant A3 - Single Functional Test (SFT) and Acceptance Test (AT)  
Assign Tester

Steps

1. Click on Test Package Details
2. Click Assign Testers link to assign Tester for each Test Package.

Remark:

Tester will be automatically assigned to the Test Package if maintained in the Work Package.
The application supports selecting ‘n’ testers and assigning to ‘m’ Test Packages.

Variant A3 - Single Functional Test (SFT) and Acceptance Test (AT)
Tipps & Tricks: Multiple Tester Assignment
Variant A3 - Single Functional Test (SFT) and Acceptance Test (AT)

Tipps & Tricks: Replace Testers

- The application supports un-assignment or replacement of testers in case they become unavailable.
- If the notify flag is checked in the first screen and Business Partner details have email address maintained, Testers can be notified if their assignments are changed.
Steps

1. Test Plan > Maintain Attributes
2. Change Release Status to “Released for Test”

Remark:

This step is optional according to the Release Status Schema that you are using for the test plan.
Variant A3 - Single Functional Test (SFT) and Acceptance Test (AT)
Overview – Activities during Test Execution

- **Manual Test Execution**
- **Test Defect**
- **Correction / Adjustment**
- **Transport**
- **Retest**
- **Test Status Reporting**
- **Test sign-off**

The test cases were executed and Test Status values are set. If required, a Defect is created for the related test case.

In case of a defect in the application, a software correction or configuration adjustment is needed. This changes are transported to the test system, were a retest is required.

The Test Manager monitors the Test Status; at the end of the test phase, the Project Manager is informed about the status and the Test is finally signed off.
**Variant A3 - Single Functional Test (SFT) and Acceptance Test (AT)**

Execute Test Case and Document results

**Steps**

1. Tester opens the test case document and run the test according to test case.
2. Tester uploads the result doc via ‘drag and drop’.
3. Tester creates or assigns the Defects.
4. Tester does retest the Test Case, when Defect Correction is solved.
5. Tester updates the test status.
Variant A3 - Single Functional Test (SFT) and Acceptance Test (AT)

Tipps & Tricks: Structure tester’s work in My Tests Execution

Tester is provided with filter criteria, helping to structure the workload.

Steps

1. Select overdue items (items where planned execution date is already past current date).
2. Select items Ready for Test.
3. Select Tests which are now available for retests.

Remarks

- Text based search can be used to find a specific Test Plan or Test Package.
All variants - Single Functional Test (SFT) and Acceptance Test (AT)

In case of an error: Report Defect

Steps

1. Input all the related information and assign the processor.
2. Assign Category to specify the team.
3. Reproduce the steps and attach screenshots to the Defect. Thus the Processor can get a quick analysis of the Defect.

Recommendations

- Defect Type should be Defect. It is possible to prefill this field via the app Personalization.
- System should be the test system, then later, if the Defect is fixed via a Defect Correction with transport, the transport will be imported to the test system via scheduled import variant of the Release Batch Import.
All variants - Single Functional Test (SFT) and Acceptance Test (AT)

In case of an error: Process Test Defect

After a Test Defect might have been dispatched to the corresponding processor, it will be processed by executing the following steps:

Steps

1. Select Defect
2. Click Actions → Set to ‘In Process’
3. Perform appropriate actions:
   - **Propose Solution:** The Defect does not need a code fix, but the tester is provided with further/more detailed instructions.
   - **Request Error Correction:** The Defect requires a code fix. Defect Correction will be created automatically, and the Work Package status will be changed to ‘In Repair’ automatically.
   - **Set to ‘Tester Action’:** The Defect is missing information and needs the tester to provide more details.

Remarks

- Tester will get email notification when the Defect is in ‘Propose Solution’ or ‘Tester Action’*

* customer specific customizing needed
All variants - Single Functional Test (SFT) and Acceptance Test (AT)

In case of an error: Process Defect Correction, Create TR & Return to Retest

Steps

1. Select Defect Correction and set to ‘To Be Corrected’.

2. Then navigate to tab ‘Transport for TR creation’.

3. The related Work Package status will change from ‘To Be Tested’ to ‘In Repair’.

4. Click icon + to create TR in the Dev system accordingly.

5. Go to Dev system for bug fixing, and save the changes to the TR.

6. Release the task in Dev system after bug fixing.

7. Execute actions → Set to ‘Retest with Transport’
   • The TR will be released automatically.
   • The Defect status will be changed to ‘Solution Proposal’ automatically.

8. Optional: Tester will get email notification for retest (manually customizing needed).

* customer specific customizing needed
All variants - Single Functional Test (SFT) and Acceptance Test (AT) Monitoring

**Steps to monitor Test Defect**

1. Navigate to Test Suite Dashboard
2. Select test plan
3. Navigate to Defect Status tab in order to monitor all the related defects

**Recommendation**

- Tile Change & Release Management can be used to access additional details of Test Defects

**Steps to monitor or distribute Defects and Defect Corrections**

1. Navigate to Mass Change app
2. Select Defect Correction / Defects per Project and Wave
3. Search the Defect Correction within scope for monitoring
4. Distribute the unassigned Defects to responsible architects
All variants - Single Functional Test (SFT) and Acceptance Test (AT) Retest

Steps

1. Update the Defect status after retest
   • ‘Confirm’ Defect is fixed
   • Set to ‘In Process’ if Defect still exists

Effects

1. When the Defect is confirmed, the Defect Correction will be confirmed automatically. Thus the related TR will be ready for import to the Pre-production System.
2. However the import to Pre-production will take place after the related Work Package with assigned Work Item(s) and Defect Correction(s) will have been set to Status ‘Hand over to Release’.
3. The related Work Package status will change from ‘In Repair’ to ‘To be Tested’.
All variants - Single Functional Test (SFT) and Acceptance Test (AT)  
Update Work Package Status when all related Test Cases are passed

Steps

1. 'Confirm Successful Test' for the related Work Package.

2. The Work Package status will change to 'Successfully Tested', which indicates that this Work Package has passed SFT successfully.
Variant A3 - Single Functional Test (SFT) and Acceptance Test (AT)  
Track Test Status and create Test Report

Steps

1. Via Test Suite Dashboard, Test Manager is able to:

- Get overview of all test type
- Track all test plans, test package execution status
- Track all Defect status via different dimensions
- Check Traceability Matrix from Requirement to Defect
**Variant A3 - Single Functional Test (SFT) and Acceptance Test (AT)**

Track Test Status and create Test Report

**Steps**

- Status and progress is monitored with the Test Suite Dashboard and operational reports of type Test Execution Analytics.
- Defects are analyzed by test manager in CRM and dispatched to the respective architects.
- In case a correction is needed a Defect Correction is created (with or without transport).
- Defect Correction is confirmed but not handed over to release
  
  **Note:** It is important to first set the WP to Handed Over to Release by the release manager. By this the status of the defect correction is automatically switched to handed over to release (in case the defect correction is in scope of the work package). Otherwise you produce inconsistencies in subsequent systems.
- All corrections are done and retested
- The Test Cases are set to successfully tested
- In the Test Suite Dashboard navigate to view Traceability Matrix and select for the Project and Wave
- Select the WP status To be Tested and check test case status for the WPs.
  - Check that for all test cases in status red a Defect exists
  - Check that for all test cases in status green no Defects are still open
  - Check for all test cases in status green that all Defect Corrections are in status successfully tested
- In case all SFT test for a WP are green navigate to the WP and set its status to successfully tested.
## Functional Integration Test

<table>
<thead>
<tr>
<th>Test Types</th>
<th>Test Level</th>
<th>Test Requirement</th>
<th>Definition</th>
<th>Test System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit Test</td>
<td>Sprint</td>
<td>Work Item (Tech. Design)</td>
<td>Technical feature test to ensure that the feature was implemented correctly</td>
<td>QAS</td>
</tr>
<tr>
<td>Single Functional Test</td>
<td>Wave</td>
<td>Work Package (Func. Spec)</td>
<td>Business function test to ensure that the feature was implemented correctly</td>
<td>QAS</td>
</tr>
<tr>
<td>Acceptance Test</td>
<td>Wave</td>
<td>Single Work Packages (Requirement)</td>
<td>Validation of requested functionality by requestor. After validation, corresponding work packages and defect corrections would be ready for handover to release. This is the most important test, as after setting the Work Package status Hand over to release, all assigned transports will end in PRD system.</td>
<td>QAS</td>
</tr>
<tr>
<td>Functional Integration Test</td>
<td>Project (Optional)</td>
<td>Related Work Packages Process with requirements</td>
<td>Early validation of or modular processes.</td>
<td>QAS</td>
</tr>
<tr>
<td>Functional Integration Test</td>
<td>Release (Final)</td>
<td>E2E Processes</td>
<td>E2E tests of the new or extended functionality validating the end to end business process. Final hands-on validation by business users.</td>
<td>PRE</td>
</tr>
<tr>
<td>Regression Test</td>
<td>Release</td>
<td>Productive Processes</td>
<td>Test of productive processes or functions</td>
<td>PRE</td>
</tr>
</tbody>
</table>
To create a Work Package that is linked to all Requirements of a Process, use the ‘Element’ filter of the ‘Requirement app’:

**Steps**

1. Select the E2E Process in ‘Element’.
2. Activate the ‘Search in sub-elements’ checkbox.
3. Click on ‘Go’ to display the Requirements.
4. Use the 'Select All' checkbox to select all filtered Requirements.
5. Use ‘Work Package’ → ‘Create New Work Package’ to create a Work Package that is linked to all those Requirements.
**Functional Integration Test (FIT)**

**Option A: Create Work Package with Requirement to have full traceability**

**Advantage**

The Traceability Matrix is based on Projects & Requirements.

Creating a WP for the FIT Test Cases you can control, that all Requirements of a Process are integration tested and completed.
Steps below are only to be performed if Test Cases for FIT are missing or to be updated.

**Prerequisite**
- Release is in status Prepare
- Quality Gate Hand-over to Release has not yet happened in order to create a new Work Package

**Steps**
1. Click Create New Work Package.
2. Select WP Category ‘Non Functional’.
3. Select Project and Wave.
4. After the test case upload finished, switch the Work Package status to ‘Hand over to Release’.

**Remark**
As Development Branch is Change Control enabled a WI is needed to be created in order to upload FIT Test Cases.
**Functional Integration Test (FIT)**

Optional: Link process element and its content to Work Package

As all the content can only be released to the parent branch via a change document, all content related to the process need to be assigned at least to one Work Item.

**Steps**

1. Assign the process structure in Work Package.
2. Link other content like Process Description, User Guide, etc.
3. Create General Change (S1CG) in Scope tab. (Assign the Solution Architect or Test Manager as the Developer).
4. Then Solution Architect or Test Manager can use this Work Item type General Change to upload test cases

Note: For the General Change the availability of a Technical Design document is checked and reported in Solution Readiness Dashboard.

In case the General Change is not used to document changes on non-SAP functionalities you can switch this KPI check off.

Alternatively you can upload a dummy Technical Design document.
Functional Integration Test (FIT)
Optional: Upload additional or updated FIT Test cases

Steps

1. Go to Processes in scope for FIT/UAT (Make sure the current Branch is Development Branch)
2. Select the Work Item (Change Document) which used for Test case upload
3. Right click the Elements of <Process> area.
4. Click Show drop area
5. Use drag and drop to upload the FIT test case from local computer
6. Use Test Case Classification FIT for later filtering with Assignment Analysis
Functional Integration Test (FIT)
FIT Test Case Readiness Check

Steps

1. Select Solution and Development branch, click ‘Test Case Assignment’.
2. Use filter for test case classification
3. Check whether there are WP without test cases of type FIT
**Functional Integration Test (FIT)**

Create Test plan for FIT/UAT

**Steps**

1. Select Project and Wave.
2. Click ‘Assignment Analysis and Test Plan Generation’.
3. Select FIT WPs under scoped process.
4. Click Test Plan → Save as new.

**Remarks:**

- Within the assignment analysis, filters e.g. on test types (SFT, FIT, UAT) are available, if the respective Test Cases Attributes are maintained. Further prerequisites are the Test Case Type ‘Additive’ in Solution Documentation and a document type ‘Test Document’ in transaction ‘SOLADM’ (see Appendix for more details)

- A Master Project can be used to build Cross Wave and Cross Project Test Plans (for the required prerequisites refer to the Focused Build Configuration Guide).
Functional Integration Test (FIT)
Create Test Plan

Steps

1. Input mandatory information for Test Plan:
   - System Role ID: PRE (Please select Pre-Production System as this FIT is to be performed in PRE system)
   - Test Plan ID
   - Description
   - Test Classification: Functional Integration Test/User Acceptance Test
   - Document Type: Used for Test Notes document
   - Planned Date

2. Select ‘One Test Package per Work Package’ for Test Package Creation, then later create test package per process in test plan

   - In the Test Suite Dashboard, the test package test result will be mapped to each FIT/UAT test case upload related Work Package.

Note: After test plan creation, the following activities are similar to SFT, please refer to the corresponding How-to!
# Regression Test

<table>
<thead>
<tr>
<th>Test Types</th>
<th>Test Level</th>
<th>Test Requirement</th>
<th>Definition</th>
<th>Test System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit Test</td>
<td>Sprint</td>
<td>Work Item (Tech. Design)</td>
<td>Technical feature test to ensure that the feature was implemented correctly</td>
<td>QAS</td>
</tr>
<tr>
<td>Single Functional Test</td>
<td>Wave</td>
<td>Work Package (Func. Spec)</td>
<td>Business function test to ensure that the feature was implemented correctly</td>
<td>QAS</td>
</tr>
<tr>
<td>Acceptance Test</td>
<td>Wave</td>
<td>Single Work Packages (Requirement)</td>
<td>Validation of requested functionality by requestor. After validation, corresponding work packages and defect corrections would be ready for handover to release. This is the most important test, as after setting the Work Package status Hand over to release, all assigned transports will end in PRD system.</td>
<td>QAS</td>
</tr>
<tr>
<td>Functional Integration Test</td>
<td>Project (Optional)</td>
<td>Related Work Packages Process with requirements</td>
<td>Early validation of modular processes.</td>
<td>QAS</td>
</tr>
<tr>
<td>Functional Integration Test</td>
<td>Release (Final)</td>
<td>E2E Processes</td>
<td>E2E tests of the new or extended functionality validating the end to end business process. Final hands-on validation by business users.</td>
<td>PRE</td>
</tr>
<tr>
<td>Regression Test</td>
<td>Release</td>
<td>Productive Processes</td>
<td>Test of productive processes or functions</td>
<td>PRE</td>
</tr>
</tbody>
</table>
Sample usage scenarios for Regression Tests

Scenarios

**Scenario 1 – Regression Test before Deploy**

- Re-execute selected Test Cases across the Test Types executed before to ensure no side-effects on test results of earlier Test Types occurred
- Scope: All units, functionalities and processes of the entire release
- Recommendation: Focus on high priority Test Cases

**Scenario 2 – Regression Tests after Waves or Sprints**

- Re-execute selected Test Cases across the Test Cases in former Waves and Sprints in order to make sure respective test results are still 'OK' after new developments or bugfixes have been introduced
- Scope: Selected units, functionalities and processes of former Waves and Sprints
- Recommendation: Focus on high priority Test Cases
Regression Test Recommendations

- Regression Test activities such as Creation of Test Plans, Test Execution can be handled very similar to the procedure described for FIT.
- To further facilitate those activities SAP Solution Manager offers:
  - Business Process Change Analyzer (BPCA) to define an optimized test scope
  - Test Automation Framework (TAF) including Component Based Test Automation (CBTA) to reduce manual testing efforts
Regression Test Support
Business Process Change Analyzer

- Change impact analysis for business processes resulting from software change events
- Use cases:
  - Impact analysis for customizing and code changes
- Benefits:
  - Precise impact analysis and significant test scope reduction

Identify your test scope and significantly reduce your test effort for regression tests with BPCA!
# Regression Test Support

## BPCA Prerequisites

### Solution Documentation

- **Minimum:** Executable Library

### Usage data setup

- **SAP Solution Manager Configuration:** Scenario Usage Logging (root flag!)

### Usage data collection

- **ABAP Call Monitor (SCMON) or UPL data will be collected and extracted from managed system**

### BPCA Self Check

- **Checks BPCA prerequisites in SAP Solution Manager and managed systems**

### Semi-dynamic TBOM

- **Where:** Execution Library or scope in Solution Documentation
- **How:** Background Job to generate semi dyn. TBOMs using UPL or SCMON data

### Dynamic TBOM

- **Where:** high-priority process steps
- **How:** Manual test execution, Manual process step execution, automated test

### Completeness Reports

- **Check:** Process Steps without (dynamic) TBOMs

---

© 2019 SAP SE or an SAP affiliate company. All rights reserved. | PUBLIC

60
Regression Test Support
Test Automation Framework

Test Automation Framework

Test Design
- Seamless integration between SAP Solution Manager and test tools from SAP and Partners
- Test Data Container (TDC) provide test data for test script parameters
- Central management of System under Tests

Test Execution
- Start of automated tests from Workcenter Test Management
- Ad-hoc start or scheduler for unattended execution
- Integrated test execution logs from partner tools

Test Result Analysis
- Test Manager can view test results from manual and automated tests in one environment
- Status and progress reports
- Integrated partner reports

Accelerated Repair
- Workflow between Tester and Test Engineer to trigger test case repair
- Rich environment for Test Engineer to analyze problems and perform repair activities
Regression Test Support
Flow to create new automated Test Configuration

Solution Documentation
Select a Business Process, Process Step or an Executable

Test Composition Environment (TCE)
Create new Test Configuration

Select SAP or non-SAP tools
Create CBTA Test Configuration

CBTA – Test Creation Wizard

Quotation
Sales Order
Delivery

Test Composition Environment (TCE)
Finalize Test Configuration

Test Data Assignment Wizard
Assign test data

Test Data Container

Create Test Configuration

Attributes
Test Script
Parameters
Test Data
Search Test

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Test Script</th>
<th>Parameters</th>
<th>Test Data</th>
<th>Search Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>ZCBTA_CREATE_QUOTATION</td>
<td>Description</td>
<td>Default Value</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ZCBTA_CREATE_SALES_ORDER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ZCBTA_CREATE_DELIVERY</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ZCBTA_CREATE_TRANSFER_ORDER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ZCBTA_POST_GOODS_ISSUE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ZCBTA_CREATE_SUPPLY_DOCUMENT</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

© 2019 SAP SE or an SAP affiliate company. All rights reserved. | PUBLIC
Regression Test Support
Flow to execute automated Test Configurations

1. User starts test execution
2. Test data selection
3. Test execution: handover of Test Script + Test Data + SUT info
4. Logon and automated business process execution
5. Test results and logs

Test Script
CBTA

System under Test

Test Suite
Tester Worklist

Test Data Container (TDC)

Test Configuration
Test Script | Test Data | System Data

Test Script Steps

Test results and logs

Logon and automated business process execution
Summary
# SAP Solution Manager - Test Suite

## Capabilities

<table>
<thead>
<tr>
<th>Solution Documentation</th>
<th>Change Impact Analysis</th>
<th>Test Planning</th>
<th>Test Execution and Reporting</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Solution Documentation</strong></td>
<td><strong>Change Impact Analysis</strong></td>
<td><strong>Test Planning</strong></td>
<td><strong>Test Execution and Reporting</strong></td>
</tr>
<tr>
<td>- Executable Library</td>
<td>- Test Plan Management</td>
<td>- Test Plan</td>
<td>- Test Execution</td>
</tr>
<tr>
<td>- Process Step Library</td>
<td>- Test Plan</td>
<td>- Test Packages</td>
<td>- Manual Test Execution</td>
</tr>
<tr>
<td>- Business Processes</td>
<td>- Test Plan</td>
<td>- Test Sequences</td>
<td>- Automated Test Execution</td>
</tr>
<tr>
<td>- Test Cases</td>
<td>- Test Plan</td>
<td>- Tester assignment</td>
<td>- Defect Management</td>
</tr>
</tbody>
</table>

**Test Automation Framework**

- CBTA (Component Based Test Automation)
- eCATT

- Micro Focus UFT
- Worksoft Certify
- other 3rd party test automation tools

- Supported UI technologies by SAP test tools: SAPGUI, WebDynpro, CRM Web-Client, SAPUI5, Fiori, …
- Partner test tools: all other UI technologies

**Test scope estimation before upgrade**

**Test scope optimization during upgrade**

© 2019 SAP SE or an SAP affiliate company. All rights reserved. I CUSTOMER
SAP Solution Manager - Test Suite
Capabilities of SAP Solution Manager Test Suite and Focused Build

**On Top Functionality of Focused Build for SAP Solution Manager**

- Requirements / Work Package based Test Plan Generation
- Manual Testing: Test Steps
- Test Execution: My Test Executions
- Test Suite Dashboard

**SAP Solution Manager – Test Suite**

- Solution Documentation
- Change Impact Analysis
- Test Planning
- Test System Setup
- Test Execution and Analytics

- Manual Test Cases
- Automated Test Cases
- Test Automation Framework with 3rd party test tool integration
- Test Composition Environment to manage SAP and partner tools
- Change Impact Analysis
- Test Plan and Test Package Management
- Test Execution of manual and automated tests
- Gap and Completeness Analytics
- Test Execution Analytics

© 2019 SAP SE or an SAP affiliate company. All rights reserved. I PUBLIC
What’s New with Focused Build SP5
Test Suite
Test Suite - New with SP05
Improved Executable F4 help

Applications
Test Steps Designer

Features
• Enter search criteria directly without waiting time
• Initial search is still running in background

Use Case
By entering the **Select: Executable** dialog the first time, an initial search of all Executable on the Test Case Location is triggered automatically. You can interrupt this search by entering values in the search fields at any time.

Nevertheless, if you enter the dialog the next time, it will show directly the results from the initial search without an additional run.
Avoid unnecessary branch selection

Applications
Test Steps Designer

Features
- Editable branch is selected automatically

Use Case
In general there is just one branch available in Test Steps test case which is editable without creating a new version. If this branch is not change controlled, it will be selected automatically by editing the test case.
Test Suite - New with SP05
Inform user if changes were done since last load

Applications
Test Steps Designer

Features
- Information about changes made on the test case since the last load

Use Case
If more than one person is working on the same test case, it is now possible to see, if there were changes made in the meantime and who was doing the changes.
**Test Suite - New with SP05**

**Inactive flag for Result Attributes**

**Applications**
Test Steps Designer

**Features**
- Result Attributes which are no longer used can be defined as *Inactive*

**Use Case**
Inactive Result Attributes are still visible in test cases where they are used for test execution (shown as *Inactive*) but can not be defined for new Test Steps test cases.
Test Suite - New with SP05
Public Result Attributes for Test Sequences

Applications
Test Steps Designer, My Test Executions

Features
• Result Attributes can be defined as Public to make them visible in Test Sequences

Use Case
By working with Test Sequences it is essential to hand over data between test cases. For Test Steps test cases Result Attributes can be defined as Public which are then shown in My Test Executions as “Previous Results from Test Sequence” and can be used directly for the execution.
Test Suite - New with SP05

Email Notification with Link to My Test Executions

Applications
My Test Executions

Features
• Standard Email Notification contain additional Link to My Test Executions

Use Case
In general testers can decide which Test Execution Tool in Solution Manager they want to use. If the Workflow for an test plan is activated, the Standard Email Notification contains now the links to both applications: My Test Executions and the Tester Worklist.
Test Suite - New with SP05
Improved “Assign Defect” dialog

Applications
My Test Executions

Features
The "Assign Defect" dialog supports now
• Search for Defect Title
• Search for Defect Priority

Use Case
The Assign Defect dialog contains now two new fields to provide a faster and more efficient search for already created Defects.
Test Suite - New with SP05
Inform user if changes were done since last load

Applications
My Test Executions

Features
• Information about changes made on the test case since the last load

Use Case
If more than one tester is working on the same test case by using Shared Results, the tester is informed after clicking on the Save button, if there were changes made since the last load of it.
Test Suite – Focused Build SP05 features
Further improvements

Application
My Test Executions

Feature details
- Change Log visibility depends on authorization object
- Usage of On-behalf mode depends on authorization object
- Improved exception handling
- Display of version title and test mode in header tab
- Support of mandatory Test Note / Test Result
- Filter for empty test package priority
- Avoid status "untested" for changed test cases

Application
Test Steps Designer

Feature details
- Change Log visibility depends on authorization object
- Authorization based on folders
- Limit status values in aggregation customizing
Test Suite – Focused Build SP05 features
Further improvements

Application
Assignment Analysis

Feature details
- Support Site “Global”
- Adjusted filter setting “Overwrite”

Application
Test Suite Dashboard

Feature details
- Inplace navigation and filtering
- Using standard libraries
Excursus: Create Test Steps
Test Steps
Test Step Designer

Design time application to create and maintain manual test cases of type Test Steps

Entry points
• Standalone – start from SAP Solution Manager Launchpad
• Solution Documentation (Process Steps and Executables can automatically transferred into your Test Case)

Test Steps
• Steps + sub-steps
• Descriptions, Executables
• Partner
• Attachments
• Evidence
• Result Attributes
• File upload using (csv)
Excursus: Technical Prerequisites
With the check report you can check your project for test management requirements.

<table>
<thead>
<tr>
<th>Message Text</th>
<th>Start Date</th>
<th>End Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project: BUILD 2 AT SP04 (BUILD 2 AT SP04)</td>
<td>01.05.2019</td>
<td>29.11.2019</td>
</tr>
<tr>
<td>Release: AT_FB_SP04_SOL_SUB_1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Solution Documentation: AT_FB_SP04_SOL (AT FB SP04 Solution)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Project is released</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Project Relationship</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Focused Build - Master Project: MASTER AT SP04 (MASTER AT SP04)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Focused Build - Build Project: BUILD 1 AT SP04 (BUILD 1 AT SP04)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Focused Build - Build Project: BUILD 2 AT SP04 (BUILD 2 AT SP04)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wave Relationship</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MASTER AT SP04 - Wave 1</td>
<td>24.05.2019</td>
<td>20.07.2019</td>
</tr>
<tr>
<td>- BUILD 1 AT SP04 - Wave 1</td>
<td>14.05.2019</td>
<td>31.07.2019</td>
</tr>
<tr>
<td>- BUILD 2 AT SP04 - Wave 1</td>
<td>14.05.2019</td>
<td>31.07.2019</td>
</tr>
<tr>
<td>MASTER AT SP04 - Wave 2</td>
<td>21.07.2019</td>
<td>05.10.2019</td>
</tr>
<tr>
<td>MASTER AT SP04 - Wave 3</td>
<td>06.10.2019</td>
<td>28.11.2019</td>
</tr>
</tbody>
</table>
Test Case Attribute Additive
Document Type – Test Document
Appendix
SP03 Requirements Management Process Flow
Detailed Workflow

- Draft: Send for Approval, Revise, Reject, Recover, Withdraw
- Rejected: Postpone, Withdraw, Recover
- Canceled (FINI): Withdraw
- Postponed: Postpone, Withdraw, Recover
- To Be Approved: Approve
- Approved: Create WP, Set by WP
- In Realization: Create WP, Set by WP
- Realized: Create WP, Set by WP
- Completed (FINI): Set by WP

= manual status setting
= automated status setting
SP03 Work Package Process Flow
Detailed Workflow Part II

In Development → In Repair
- Postpone
- Scope Extension
- Reverse Scope Extension

To Be Tested
- Confirm Successful Test
- Handover to Release
- Successfully Tested

Successfully Tested → Handed Over to Release
- Successfully Tested
- Handover to Release
- Productive
- Complete

Functional Gap
- Create Functional Gap

Postponed
- Set by WI

Scope Change
- Create Scope Change

Set by WI

= manual status setting
= automated status setting
Focused Build SP03 Defect Process Flow
Detailed Workflow for Defect

- **New**
  - Set to "In Process"
  - Forwarded
  - Withdrawn (FINI)

- **In Process**
  - Forward
  - Wait for Error Correction
  - Information Requested
  - Information Provided
  - Error Correction in Process
  - Solution Proposal
  - Confirmed (FINI)

- **Tester Action**
  - Set to "In Process"
  - Set to "Tester Action"
  - Forwarded

- **Withdrawn (FINI)**
  - Set by DC

- **Propose Solution**
  - Information Delivered
  - Information Required

- **Error Correction in Process**
  - Set by DC

- **Wait for Error Correction**
  - Set by DC

- **Withdraw**
  - Withdrawn (FINI)

- **Automated Status Setting**
- **Manual Status Setting**

© 2019 SAP SE or an SAP affiliate company. All rights reserved. PUBLIC
Focused Build SP03 Defect Correction Process Flow
Detailed Workflow for Defect Correction (Defect as Predecessor Document)

Created
- Set by Defect

Withdrawn (FINI)
- Withdraw Defect Correction

Being Corrected
- Set by Defect
- Information Required
- Developer
  - Set to 'To Be Corrected'
- Solution Architect
  - Set to 'To Be Corrected'

Transport to Retesting
- Set to 'Retest with Transport'
- Change to Task Plan
- Developer
  - Set to 'In Correction'

To Be Retested w/o Transport
- Set to 'Retest Without Transport'
- Confirm Defect Correction
- Developer
  - Reset to 'In Correction'

Confirmed
- Confirm Error Correction with Transport
- Handover to Release
- Close Error Correction (FINI)

Handed over to Release
- Release Manager

Closed (FINI)
- Solution Architect

Set to 'To Be Corrected'
- Reset to 'In Correction'
- Set to 'Retest with Transport'
- Set to 'Retest without Transport'

- = automated status setting
- = manual status setting
Focused Build SP03 Defect Correction Process Flow
Detailed Workflow for Defect Correction (1:1 Relation for Test Package and WP)

- **Created**
  - Set by Defect
  - Solution Architect: Set to 'To Be Corrected'
  - Developer: Request Information

- **Being Corrected**
  - Information Required
  - Developer: Reset to 'In Correction'

- **To Be Retested w/o Transport**
  - Developer: Set to 'Retest Without Transport'
  - Change to Task Plan

- **Transport to Retesting**
  - Confirm Error Correction with Transport

- **Confirmed**
  - Developer: Confirm Defect Correction
  - Set to 'Retest with Transport'

- **Handed over to Release**
  - Close Error Correction
  - Release Manager

- **Closed (FINI)**
  - Solution Architect

- **Withdrawn (FINI)**
  - Set by Defect
  - Withdraw Defect Correction

- **Withdrawn**
  - Set to WP
  - Solution Architect

**Note:**
- **= automated status setting**
- **= manual status setting**
Focused Build SP03 Defect Correction Process Flow
Detailed Workflow for Defect Correction (Work Package as Predecessor Doc.)

- Created
- Being Corrected
- Information Required
- To Be Retested w/o Transport
- Transport to Retesting
- Confirmed
- Handed over to Release
- Closed (FINI)
- Withdrawn (FINI)

Set by Defect

= automated status setting
= manual status setting