Focused Build for SAP Solution Manager 7.2 (SP6)
Requirement Management, Work Package and Work Item

Customer Experience & Solutions, SAP SE
June 2020
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.
Change Paces in Requirements-to-Deploy

Requirements Management
- Initial Requirement Backlog Planning
- Requirements Upload
- Create Requirements
- Requirements Workflow
- Requirements Document Management
- Requirements Reporting
- Requirements News with SP06

Work Items
- Sprint Backlog Planning
- Create Work Items
- Work Items Workflow
- Work Items Document Management
- Work Items Reporting
- Work Packages and Work Items News with SP06

Work Packages
- Product Backlog Planning
- Create Work Packages
- Work Packages Workflow
- Work Packages Document Management
- Work Packages Reporting

Additional Topics
- Manage Changes in SFT and AT with Defect Corrections
- Build the Release
- Go-Live
Change Paces in Requirements-to-Deploy
### Three Different Change Paces in Requirements-to-Deploy

#### Fix

<table>
<thead>
<tr>
<th>Trigger</th>
<th>Business disruption or standard change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scope</td>
<td>Small</td>
</tr>
<tr>
<td>Process Impact</td>
<td>None</td>
</tr>
<tr>
<td>Approval</td>
<td>Individual</td>
</tr>
<tr>
<td>Deployment</td>
<td>Unbundled on request or bundled with release</td>
</tr>
<tr>
<td>Time to delivery</td>
<td>1 day – 1 week</td>
</tr>
</tbody>
</table>

#### Enhance

<table>
<thead>
<tr>
<th>Trigger</th>
<th>Improvement request</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scope</td>
<td>Medium</td>
</tr>
<tr>
<td>Process Impact</td>
<td>Minimal</td>
</tr>
<tr>
<td>Approval</td>
<td>Individual</td>
</tr>
<tr>
<td>Deployment</td>
<td>Bundled with release</td>
</tr>
<tr>
<td>Time to delivery</td>
<td>1 month</td>
</tr>
</tbody>
</table>

#### Innovate

<table>
<thead>
<tr>
<th>Trigger</th>
<th>Transformation projects, new solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scope</td>
<td>Large</td>
</tr>
<tr>
<td>Process Impact</td>
<td>Significant</td>
</tr>
<tr>
<td>Approval</td>
<td>Pre-Approved</td>
</tr>
<tr>
<td>Deployment</td>
<td>Bundled with release</td>
</tr>
<tr>
<td>Time to delivery</td>
<td>3-12 months</td>
</tr>
</tbody>
</table>

---

S/4HANA Conversion Project Support provided with Focused Build SP06 as additional starting point for Requirements-to-Deploy
## SAP Solution Manager Integration Model

### Process Flow

The Requirements-to-deploy value chain supports the three different change paces at an optimum.

<table>
<thead>
<tr>
<th>Fix</th>
<th>Enhance</th>
<th>Innovate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Program fix required to resolve disruption</td>
<td>Enhancement required for daily business operations</td>
<td>Strategic initiative for new business model</td>
</tr>
<tr>
<td>Fix immediately, deliver break-fixes and standard changes</td>
<td>Assess enhancement request, negotiate delivery and cost</td>
<td>Model to-be processes, collect Requirements</td>
</tr>
<tr>
<td>As fast as needed.</td>
<td>Deliver enhancement</td>
<td>Plan solution delivery</td>
</tr>
<tr>
<td></td>
<td>Bundled in minor release</td>
<td>Deliver solution with continuous business feedback</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Bundled in major release</td>
</tr>
</tbody>
</table>

No overhead.

---

Monitor Solution Readiness
SAP Solution Manager Integration Model
Focus Build Document Flow

- **Demand**
- **Design**
- **Development**
- **Test**
- **Deploy**

<table>
<thead>
<tr>
<th>Fix</th>
<th>Enhance</th>
<th>Innovate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Incident</td>
<td>Business Requirement</td>
<td>Requirement</td>
</tr>
<tr>
<td>FB Request for Change</td>
<td>IT Requirement</td>
<td>Work Package, Scope Change</td>
</tr>
<tr>
<td>FB Urgent Change FB Standard Change</td>
<td>Change Document</td>
<td>Work Item</td>
</tr>
<tr>
<td>Immediately after approval</td>
<td>Minor releases</td>
<td>Major releases</td>
</tr>
</tbody>
</table>

Solution Readiness Dashboard

© 2020 SAP SE or an SAP affiliate company. All rights reserved. | PUBLIC
SAP Solution Manager Integration Model

Transaction Types

<table>
<thead>
<tr>
<th>Demand</th>
<th>Design</th>
<th>Development</th>
<th>Test</th>
<th>Deploy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fix</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Incident</td>
<td>S1CR</td>
<td>S1HF / S1SG</td>
<td></td>
<td>Immediately after approval</td>
</tr>
<tr>
<td>Enhance</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SMBR</td>
<td>SMIR</td>
<td>SMHF / SMMJ / SMGC / SMAD</td>
<td></td>
<td>Minor releases</td>
</tr>
<tr>
<td>Innovate</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S1BR</td>
<td>S1IT</td>
<td>S1MJ / S1CG</td>
<td></td>
<td>Major releases</td>
</tr>
</tbody>
</table>

Solution Readiness Dashboard

© 2020 SAP SE or an SAP affiliate company. All rights reserved. | PUBLIC
S/4 HANA Conversion Project Support
Integration Readiness Check in Focused Build

SAP Readiness Check for SAP S/4 HANA

Excel Export/Import

Simplification Item Management UI5 App

Evaluation and Categorization of Activities

Focused Build Project

Focused Build Requirement

Focused Build Work Package
Simplification Item Management UI5 Application

- Graphical display of Simplification Items and related Conversion Activities by Condition, Phase, Type, Status and Follow-Up Type
- Table display with all Simplification Items and related Conversion Activities and their most-important information
- Powerful and fast multi-filter options
- Direct Creation of Follow-Up Requirements or Work Packages and Assignment of Focused Build Projects
- Postponement or Rejection of non-required activities directly possible from the UI5 application

For more information see L2 presentation for S/4HANA Conversion Project Support
Requirements Management
Initial Requirement Backlog Planning
Digital Business
Delivery Platform: The Big Picture

Discover & Prepare
- One Content Cloud
- Roadmaps
- SAP Best Practices
- Cloud Appliance Library

Explore
- WBS & Q-Gates
  - Readiness Check

Realize & Deploy
- Business Process
  - Requirements
  - Work Packages
  - Solution Readiness Dashboard
  - Solution Landscape
    - Build Factories

Operate
  - Monitor & Support

Value Discovery
- SAP S/4HANA Value Assurance Service Packages
  - planning and safeguarding
  - technical implementation
  - function implementation
  - innovation

Innovation Control Center
SAP Mission Control Center
Operations Control Center
Initial Backlog

Definition

A **Release** can be assigned to a:
- Project -&gt; more Waterfall approach
- Wave -&gt; more Agile approach

There must be an aligned and **conscious decision how agile** the project should be implemented and organized. This is less a question of implementation tools, but how much a team and management(!) is able to change it’s way of work

An **Initial Backlog** in Focused Build means to have:
- A list of approved and prioritized Requirements
- Building the content of the to be implemented Release
**Build Design: Example New Implementation**

**Fit Gap Analysis and Delta Design – Methodology Overview**

**Step A**
Fit Gap Analysis / Solution Validation

1. **Finalize System Setup**
   - Prepare additional sample data
   - Enhance system setup with additional scope

2. **Fit Gap Workshops / Validation of SAP Solution**
   - Show and tell SAP standard key design elements / Gap Identification

3. **Gap Documentation**
   - Document and specify identified Gaps in initial Backlog

4. **Delta Scope Prioritization**
   - Prioritization according to agreed criteria (e.g. Business Value, Criticality)

5. **Product Backlog**

   - Prioritize
   - Create Work Packages to be part of the first Wave with Requirement Management app
   - Approve Requirements to be part of release with Mass Change app
   - Verify Priority, Effort and Value with Requirement Management app
   - Rework Requirements via Requirement Management or My Requirements app
   - Central review of identified gaps: Completeness, Prioritization

6. **Verify & Accept**
   - Verify solution design
   - Acceptance Procedure

**Step B**
Delta Design

5.1 **Create Product Backlog**
   - Prioritization according to technical criteria (e.g. Development sequence)

5.2 **Design/Specify**
   - For real Gaps and WRICEF, create Functional Specification in My Work Packages app

6. **Verify & Accept**
   - Verify solution design
   - Acceptance Procedure

Process Management Solution Readiness Dashboard

Capture and classify identified gaps in backlog:
- Document Requirements
- Classification (process gap, functional gap, etc)
- Initial solution proposal
- Initial effort estimates

Extend Processes which are not part of Best Practices, e.g. QM

Central review of identified gaps:
- Completeness
- Prioritization

Create solution design:
- Slice the Requirements into Work Packages which need to fit in one Wave
- Document relevant configuration
- Document solution for identified gaps, e.g. One Delta Design for Sourcing and Procurement

Initial Backlog

<table>
<thead>
<tr>
<th>Priority / Effort / Value</th>
<th>Would</th>
<th>Should</th>
<th>Could</th>
<th>Would</th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>4</td>
</tr>
<tr>
<td>Medium</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>5</td>
</tr>
<tr>
<td>Low</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>1</td>
</tr>
<tr>
<td>Very Low</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>8</td>
</tr>
<tr>
<td>Very Low</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>2</td>
</tr>
<tr>
<td>Very Low</td>
<td>11</td>
<td>11</td>
<td>11</td>
<td>7</td>
</tr>
<tr>
<td>Very Low</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>Very Low</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>4</td>
</tr>
<tr>
<td>Very Low</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>Very Low</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>2</td>
</tr>
<tr>
<td>Very Low</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>Very Low</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>Very Low</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Very Low</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>Very Low</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>Very Low</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Product Backlog

<table>
<thead>
<tr>
<th>Wave</th>
<th>Initial</th>
<th>Should</th>
<th>Could</th>
<th>Would</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wave 1</td>
<td>16 4</td>
<td>15 5</td>
<td>14 1</td>
<td>13 8</td>
</tr>
<tr>
<td>Wave 2</td>
<td>12 2</td>
<td>11 7</td>
<td>10 3</td>
<td>09 4</td>
</tr>
<tr>
<td>Wave 3</td>
<td>08 2</td>
<td>07 2</td>
<td>06 4</td>
<td>05 3</td>
</tr>
<tr>
<td>Wave 4</td>
<td>04 3</td>
<td>03 6</td>
<td>02 0</td>
<td>01 5</td>
</tr>
</tbody>
</table>

To sign-off release Functional Specification in My Work Packages app and set according Work Package Status

For real Gaps and WRICEF, create Functional Specification in My Work Packages app

Repeat this Wave Planning activity for each Wave -> agile Plan all Waves at once -> waterfall

Result: Product Backlog
Fit Gap Analysis and Delta Design – Step 1
Prepare a Fit/Gap Workshop

Step A
Fit Gap Analysis / Solution Validation

1. Finalize System Setup
   - Prepare additional sample data
   - Enhance system setup with add. Scope

   Extend Processes which are not part of Best Practices, e.g. QM

Step 1: How to prepare a Fit/Gap Workshop with Focused Build?

FB Role: Business Analyst
Functionality: Process Management

Procedure:
- Choose a Sandbox or On-Premise SAP Solution Manager System
- Check predefined content offerings from SAP (SAP Best Practice or Model Company content)
- Choose and further prepare the business process(es) to be part of the Fit/Gap analysis
- In case it is in the knowledge of the consultant or there are already more documents or process models available, extend and adjust the predefined process diagrams

Advantage:
- Accelerated set-up of Sandbox system
Create Work Package
Activate Best Practices with Focused Build

Step 1
Baseline build, as basis for Requirements creation in the Fit/Gap Workshops, shall be done on the Sandbox system:
- By default, it is not foreseen to maintain Requirements and Work Packages in Discover and Prepare project phase
- So, just activate the Best Practices via Solution Builder without explicit Focused Build specific reporting

Step 2
For the configuration of the Development system, it’s recommended to create one Requirement per Business Process. There are several options dependent of the configuration complexity and duration:

- Simple and quick configuration done in one Wave: 1 Requirement, 1 WP, 1 WI, 1 Transport
- Simple but long lasting config. done in several Waves: 1 Requirement, X WPs, 1 WI, 1 Transport
- Complex but quick config. done in one Wave: 1 Requirement, 1 WP, X WIs, X Transports
- Complex and long lasting config. done in several Waves: 1 Requirement, X WPs, X WIs, X Transports

The Work Packages which specify the Solution Builder activities are classified as FIT.
Release Planning: Roles and Responsibilities

1. Define Product Backlog

Business Responsibility
Business Analyst

2. Prioritize Product Backlog

Create Requirement based on business process
Use Value points, effort and priority to rank

5. What would you like in the release?

Approve, postpone Requirements

3. Analysis of Dependencies

IT Responsibility
Architect

4. Estimate Product Backlog

Create Work Packages

6. Scope Release: Build logical and technical packages

High-Level Release Plan Based on Master Project

Detailed Release Plan based on Build Project
Fit Gap Analysis and Delta Design – Step 2
Execute a Fit/Gap Workshop

Step A
Fit Gap Analysis / Solution Validation

1. Finalize System Setup
   - Prepare additional sample data
   - Enhance system setup with add. Scope

2. Fit Gap Workshops / Validation of SAP Solution
   - Show and tell SAP standard key design elements / Gap Identification

Best Practices / Model Companies uploaded to SAP Solution Manager

Fit Gap Workshop per Solution Capability, e.g.
   - operational procurement:
   - Process / Steps
   - Solution / Function
   - Roles
   - Frontend (Fiori vs. GUI)

Step 2: How to execute a Fit / Gap Workshop?

FB Role: Business Analyst
Functionality: Process Management
Naming: Gap in Activate is called Requirement in Focused Build

Procedure:
  ▪ Logon to SAP Solution Manager and review the business process (either in Process Structure or Diagram)
  ▪ Maintain Requirements (= Gaps in Activate)
    - Document Requirements in SAP Solution Manager as draft in Process View or Diagram. Requirements in FB are classified as Gap, WRICEF, Fit, Non-functional).
    - Alternatively it is possible to document Requirements in xls or another system e.g. from Model Company, and upload them to On-premise SAP Solution Manager system later.
  ▪ Release Process to Development Branch (with the help of a Work Package and Work Item).

Reporting:
  ▪ Solution Readiness Dashboard (Unassigned Requirements) /Requirements Management

Advantage:
  ▪ Efficient Fit-Gap Workshops: Requirements are linked/displayed to Business Process enabling an efficient discussion and reducing the duration of the workshops
Fit Gap Analysis and Delta Design – Step 3
Detailed Requirement Documentation

Step A
Fit Gap Analysis / Solution Validation

1. Finalize System Setup
   - Prepare additional sample data
   - Enhance system setup with add.

2. Fit Gap Workshops / Validation of SAP Solution
   - Show and tell SAP standard key design elements / Gap Identification

3. Gap Documentation
   - Document and specify identified Gaps in initial Backlog

Step 3: How to do detailed Requirement documentation?

FB Role: Business Analyst
Functionality: Requirement Management, My Requirements

Procedure:
- After the Fit/Gap workshops, rework the Requirements in status ‘Draft’. When ready, set the Requirement status ‘To be Approved’
- In case the Requirements have been documented in xls, or another system, upload to On premise SAP Solution Manager. Assign the Solution Documentation. Check and rework/precise the Requirements. When ready, set the Requirement status ‘To be Approved’

Advantage:
- Efficient communication between functional and technical experts: Detailed Requirement description and business context helps the Architect and Development to better understand the Requirements
Step 4: How to define the Initial Backlog?

FB Role: Release/Project Manager, Architect

Functionality: Mass Change Operation

Procedure:
- Select Requirements in Status ‘To be Approved’
- Sort Requirements according Priority, Value or Effort Points
- Check and adapt the values
- Sign-off: Approve those Requirements to be part of the Release
- Postpone those Requirements to be part of the next Release

Reporting:
- Solution Readiness Dashboard (Unassigned Requirements, Requirements Management)

Advantages:
- Efficient prioritization via filter and sort functionality
- Efficient approval process via mass change functionality

Note:
- In fix price/waterfall projects, the Initial Backlog is a fixed list of Requirements defining the complete content of a Release. So you need only one Release for the whole project and you could plan the complete project already at this point in time
- In agile projects, the Initial Backlog might change. So the Architects check the Initial Backlog each wave again. They plan their product backlog according the current prioritization. So typically you have a Release for each wave.
How to come to an Initial Backlog - Summary

Steps

Step 1: Requirements are gathered with the help of Fit-Gap Workshops
- The creation of an Initial Backlog is a process which typically lasts several weeks or month
- The Requirements definition process can vary very much, dependent of the SAP Partner and Implementation methodology

Step 2: Working with processes and process structures and diagrams
- Best Practice is, to start with predefined SAP Content (Best Practice or Model Company) and directly maintain the Requirements in SAP Solution Manager. The advantage is, that process structure elements are automatically assigned and all documents are automatically stored at the correct place
- This data is then inherited by each follow-up document, e.g. the Work Packages and Items
- Working with diagrams coming with the SAP Content and prepared by the SAP consultant beforehand the workshop accelerate the discussion
- Alternatively it’s as well possible draw processes on brown paper and gather the Requirements in xls. When this is done you can bring the Requirements with one upload in SAP Solution Manager. Then the process structures in Solution Documentation need to be maintained, optionally the process diagrams redrawn, and after the Requirement upload, the process structure manually assigned to the Requirements.

Step 3: There is a slim approval workflow for the Requirements
- During the Fit-Gap Workshops the Requirements are in Status ‘Draft’
- When the Gap definition for a scenario or process is ready, the responsible Business Analyst sets the Requirement on status ‘To be Approved’

Step 4: The approval procedure for Requirements for each release is done by an Approval Board
- This is commonly done via the Mass Change Analysis, where the ‘To Be Approved’ Requirements are collected and checked
- Requirements which are not so urgent are put on status ‘Postponed’ and are re-checked at the next Approval Workshop for the next Release
- The activity shall be executed by an Architect (having change authorization for the Mass Change) but under the participation of the program and project managers
When to do plan the Initial Backlog

Release

Initial Backlog

Epics

Content

Fit / Gap analysis WS

Gap

Prioritized/ Ranked, rough effort estimation

Fit

Gap

WRICEF, Fit (only configuration)

Non functional Requirements

Requirements

Functional analysis & release Planning (aggregation or splitting) duplicates eliminated

Product Backlog

Work Packages

Gap, WRICEF, Fit (config) Non functional Requirements

Sprint Backlog

Work Item

Sprint planning (Which dev team, which order)

Work Item

Work Item

Waves

Sprints

Sprints
Agile Project Execution with Focused Build
How to come to a Product Backlog and create Work Packages - Summary

Steps

The Creation of Work Packages is not an activity an Architect executes in one attempt, but in a phased approach:

Step 1: When the Requirements are approved for the first release, the Architects create the Work Packages

- Create WPs for each Build Project, e.g. Purchase to Pay, Logistics, Controlling, Master Data, …
- Doing that, they build the Product Backlog for the first or all Waves. So a WP should be developed in one Wave.
- Planning for the first Wave only means a more agile attempt as you are fully flexible for the next wave. Planning for all waves would be typical for a classical waterfall approach

Step 2: When the Product Backlog Planning for the first Wave is done, the Functional Specifications or Configuration Guides are created. Ideally, Test Cases or at least the Templates are assigned as well.

Step 3: When the planned Development is clearly described in the Functional Specifications, the needed development activity can be described and distributed. This is done by scoping Work Items for the Developers

Step 4: Once the Planning activities are finalized, the Architect approves the Scope -> as a result the Work Package is set to status Scope finalized.

- This activity can be done by an Architect in his single Work Packages
- Or in form of an Approval Board Workshop with the help of the Mass Change Application

Step 5: When the Development starts all Work Packages are handed over to development

- Like above this can be done by each Architect or commonly via the Mass Change Application
Requirements Upload
Transparent Requirements-to-Deploy
Agile Delivery Model with distributed teams

Project Team Onsite

Process and Application Landscape

Requirements

Solution Readiness Dashboard

Work Packages / Work Items

Fit

Onsite Delivery

System Integrators

WRICEF/Gap

Gap

Build Factories

Development Factories

© 2020 SAP SE or an SAP affiliate company. All rights reserved. | PUBLIC
How to start - Methods for Requirement Maintenance or Upload

Overview

Process Management Diagram
➔ Directly maintain Requirement at Business Processes or Business Process Steps

Process Management Column Browser
➔ Directly maintain Requirement at Library or process structure elements

Excel upload
➔ Initial upload of Requirements into Requirements Management (manual assignment to Process Structure Elements as follow-up activity needed)

Excel down-/upload
➔ Initial down- and upload of process structures including Requirements, e.g. from cloud or Model Company systems

Download from JIRA
➔ Initial download of Requirements from JIRA or other external Requirements Mgmt. tool via API. incl. status update when processing Requirements in Focused Build
Down and Upload Requirements (since SP03)

**Use Case 1 - Other SAP Solution Manager system as data source**

Down and upload Requirements with Solution Documentation structure information, e.g. process, process step, library

**Use Case 2 - External data source**

Upload without Solution Documentation structure information, e.g. process, process step, library.
Use Case 1: Down and Upload Requirements from other SolMan

Applications
Requirements Mgmt., Excel

Feature
• Down and Upload requirements from other systems like Model Company or CALM

Use Case
The project preparation phase started on another SolMan system, like a CALM or Model company.

There process structure and Requirements already have been defined.

Now you can do an initial upload of the Requirements to the design branch of the on-premise SolMan with automated assignment to the process structure.
Use Case 2: Upload Requirements from External Data Source

Applications
Excel, Requirements Mgmt.

Feature
• Upload requirements from other data sources

Use Case
The project preparation phase without another SolMan system, e.g. xls-based or gathered in documents

You can do an initial upload of the Requirements to Requirements Management app. of your SAP Solution Manager

Create the process structure/diagrams and assign them to the Requirements.
Create Requirements
How to start - Methods for Requirement Maintenance or Upload

Overview

Process Management Diagram

➔ Directly maintain Requirement at Business Processes or Business Process Steps

Process Management Column Browser

➔ Directly maintain Requirement at Library or process structure elements

Excel upload

➔ Initial upload of Requirements into Requirements Management (manual assignment to Process Structure Elements as follow-up activity needed)

Excel down-upload

➔ Initial down- and upload of process structures including Requirements, e.g. from cloud or Model Company systems

Download from JIRA

➔ Initial download of Requirements from JIRA or other external Requirements Mgmt. tool via API. incl. status update when processing Requirements in Focused Build
Digital Business
Use SAP Activate for SAP S/4HANA for delta scoping

SAP Activate provides SAP S/4HANA best-practice process content that consists of process diagrams, documentation, and configuration.

- You can download this content into your SAP Solution Manager
- From the diagram, you can jump into SAP FIORI apps in a pre-activated SAP S/4HANA trial system
- Execute show and tell of the SAP S/4HANA innovations hands-on
- Document Requirements as a result of this fit/gap analysis
Best Practice is to add Requirements to process steps and processes, but they can also be linked to other elements if required.

Requirements can be maintained at the following SolDoc Elements:

- Structure Elements
  - Process
  - Process Step (Reference)
- Library Elements
  - Configuration Item
  - Process Step (Original)
  - Executable
  - Development
  - Interface
What’s in Focused Build on top of SAP Solution Manager?

Explore Phase

Requirements Management

- Create Requirements in Validation workshops with customers for delta identification (Fit/Gap analysis)
- Requirements are integrated into process context allowing better handover to build team
- Multi-language support for Requirements
- Tight integration of Requirements Mgmt. into solution landscape, process models, Work Packages, Work Items, Solution Documentation, and the Solution Readiness Dashboard
Requirements can be assigned within the diagram editor or in the browser by selecting the process step or other structure elements.
Requirements for Process Steps

Relationship from Requirements to processes in Solution documentation

Feature details

• Indicate in the process if Requirements are available
  • 4 different decorator icon shows the different status of Requirements
  • Preview of Requirement in pop up window
• Direct access to all existing Requirements for the process step to create further Requirements

Benefits

Monitor in the process structure how many Requirements exist and which status
Avoid creating redundant Requirements for the same process
The outcomes of the EXPLORE workshops may initially be referred to as ‘Gaps’ or ‘Deltas’ from which we create “Requirements” when using Focused Build with SAP Solution Manager.

Within Focused Build the Requirements are classified; the follow-on Work Packages (WP) and Work Items (WI) are also classified.

The classification options are fixed (\textit{\& not customizable})

The Classification of the Requirement is for Information only and to help define Work Packages (\textit{and although defaulted to a WP can be changed when a requirement is assigned to a Work Package})

<table>
<thead>
<tr>
<th>Requirement Classification</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gap</td>
<td>is a completely new development which needs to be specified in detail with significant ‘as-is /to-be’ evaluation, often with no technical information or idea how to realize it in the beginning. Technical design is fully done by developer in WI with the help of a technical design document. (There will be few requirements with this classification)</td>
</tr>
<tr>
<td>WRICEF</td>
<td>is a typical and from SAP expected extension, where no business background needs to be described. The consultant often already knows how to implement and configure. So the Specification is often already a mixture between functional and technical design</td>
</tr>
<tr>
<td>Fit \textit{(configuration)}</td>
<td>there is no coding adjustment, but only customizing. So specification is often an existing standard configuration guide and only a customizing documentation is needed to be maintained on Work Item level</td>
</tr>
<tr>
<td>Non-Functional</td>
<td>is used for documentation upload or a parameter settings without the need to document a Functional Specification (no document KPI maintained in customizing for Work Package and Work Item).</td>
</tr>
</tbody>
</table>
What is the right Requirements Granularity?
- From a well defined Requirement you can directly create a Functional Specification or assign a Configuration Guide

Define and discuss within project team how to use priority, category and other attributes across project teams (!)

Decide on categories before starting requirements gathering in order to simplify searching and filtering (relevant for Mass Change, SRD); reuse them in WP and WI as well

Add meaningful names (naming convention starting with Process abbreviations) and description to requirements – especially when working on bigger teams!

If required also document Fit requirements for holistic documentation, e.g. in config library
- but define how this shall be handled otherwise the amount of requirements is growing too fast

Define the Sign-off process (Requirements and Process Structure) together with the customer to avoid a bottleneck at the beginning of the configuration/development activities
Requirements Workflow
Requirements Management Process Flow as of SP05
Detailed Workflow

- **Draft**
  - Send for Approval
  - Revise
  - Recover
  - Withdraw

- **Rejected**
  - Set by WP
  - To Be Approved
  - Approve
  - Reject
  - Postpone
  - Recover
  - Withdraw

- **Canceled (FINI)**
  - Set by WP

- **Postponed**
  - Set by WP

- **To Be Approved**
  - Create WP

- **Approved**
  - Set by WP

- **In Realization**
  - Set by WP

- **Realized**
  - Set by WP

- **Completed (FINI)**
  - Set by WP

- Manual status setting
- Automated status setting

© 2020 SAP SE or an SAP affiliate company. All rights reserved. ǀ PUBLIC
Requirements – Document Management
**Basic functionality and usage overview**

**dropDoc Attachments in Work Package and Work Item Applications**

**dropDoc Attachments integration to Requirement Application**

**Note:** Attachments are not stored in Solution Documentation, but are copied into the related Work Packages.

In Solution Administration and by customizing, you can define Attachment templates to ensure a standardized document use.
Basic functionality and usage overview

**dropDoc Attachments in Work Package and Work Item Applications**
- UI5 technology
- Mass upload of documents
- Access to document templates
- Drag and Drop functionality

**dropDoc Attachments integration to Requirement Application**
- Integration into Requirement Application
- Mass upload of documents into selected Requirement
- Direct access to document templates
- Drag and Drop functionality
Requirements Reporting
Requirements Management

Applications

Requirements Management Application

Feature

• Main purpose is to create Requirements and Work Packages
• It also allows to get an overview of Requirements in a specific context, e.g. Scope, Business Process, Owner etc.

Use Cases

• Filter & sort Requirements to do Release planning based on Value and Effort Points.
• Find Requirements without Solution/Process Structure assignment.
• Check Status of Requirements of a selected Scope/Business Process
• Etc.
Mass Change

Applications
Mass Change Application

Feature
Mass Change of selected Requirements

Use Case
Select Requirements to do collective changes, e.g. approve them while a Release planning Q-Gate
Solution Readiness Dashboard

Applications
Solution Readiness Dashboard

Feature
Predefined reporting based on Requirement ID and Project

Use Case
Track unassigned Requirements and Requirements assigned to your Build Project
Solution Readiness Dashboard - Requirements Tile in Detail

Overview of all Requirements, either unassigned or assigned via Work Package to Project. Each bar of the tile is clickable and offers a filtered list of Requirements

Total Backlog means unassigned Requirements:
- don't have Work Packages assigned to them or
- are associated with Work Packages that aren't assigned to any project
- have the detail status Draft, To be Approved or Approved

<table>
<thead>
<tr>
<th>Requirement Id</th>
<th>Title</th>
<th>Description</th>
<th>Status</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>8000006629</td>
<td>HZ NR 3000</td>
<td>HZ NR 3000 - attachment w/o doctype</td>
<td>Unassigned</td>
<td>1: Very High</td>
</tr>
<tr>
<td>8000006630</td>
<td>HZ RM 3001</td>
<td></td>
<td>Unassigned</td>
<td>1: Very High</td>
</tr>
<tr>
<td>8000006699</td>
<td>HZ 1610 03New</td>
<td>HZ 1610 03New</td>
<td>Unassigned</td>
<td>1: Very High</td>
</tr>
<tr>
<td>8000007843</td>
<td>HZ 2505 01</td>
<td></td>
<td>Unassigned</td>
<td>1: Very High</td>
</tr>
<tr>
<td>8000008124</td>
<td>HZ 3005 No01</td>
<td></td>
<td>Unassigned</td>
<td>1: Very High</td>
</tr>
</tbody>
</table>
Solution Readiness Dashboard - Requirements Tile in Detail

Requirements ‘To be scoped’

- are associated with Work Packages from current Project which still need to be scoped.
- have the detail status ‘Approved’
Solution Readiness Dashboard

Example: Unassigned Requirements

- Status overview with aggregated numbers
- Detailed list view with filter options
- Navigation to each single Requirement for all details
Traceability Matrix

Applications
Test Suite Dashboard

Feature
- Predefined reporting based on Project and Test Plan Selection
- Starts with Project
- Includes Requirements, Work Packages, Work Items, Defect, Defect Corrections and assigned detailed information

Use Case
Traceability from Requirement to Defect & Transport
Traceability Report

Applications
/SALM/Traceability_Report

Feature
- Report starting with Requirement ID
- Offers all data like Traceability Matrix
- Includes flexible selection of additional data

Use Case
Enables customer specific reporting via xls export
Key Takeaways Requirements Management

Workflow

Requirements

- Are created on the Design branch and not released to Development branch
- Should be related to any process structure in SAP Solution Manager process management, process or non-process related
- Should be specific and granular
- Have an 1:m or n:1 relationship to Work Packages
- Are ideally consolidated (not x Requirements for the same need)
- Need to be approved
- Get an automated status update from the Work Package
- Maintain categories reflecting your build projects
Key Takeaways Requirements Management

How-to Document Requirements

Do’s and Don’ts

- Ideally define Requirements in a 1:1 Relationship to a Work Package (in an Agile Development that means it can be realized in one Wave)
- Description should be in a clear and agreed user story format (especially when working in big teams)
- Define and follow clear meaningful naming conventions for Requirements
- Define and discuss within project team how to use priority, category and other attributes across project teams
- Decide on Requirements categories before starting Requirements gathering (This will help later tracking of the build and also simplify searching and filtering)
- Requirements are ideally consolidated (not n Requirements for the same need)
- If required also document ‘Fit-Fit Requirements’ for holistic documentation & all requirements that the business want to later test
- Description captured by Business Analysts; Classification needs consulting input; Approval of Requirements by the business and programme management.

The ‘Requirements Management’ transaction can be used to create/import requirements, or filter/display lists of requirements. There is also a link to ‘Mass change’ function where filtered requirements can have the status & ownership maintained.

Requirements can be maintained at the following SolDoc Elements:

- Structure Elements
  - Process
  - Process Step (Reference)
- Library Elements
  - Configuration Item
  - Process Step (Original)
  - Executable
  - Development
  - Interface

“As a (role), I need (feature), so that (benefit).”

Capture Requirements as User Stories (against the Process/Step)
What’s New with Focused Build SP6?

Requirement Management
Requirements Management – new with Focused Build SP06
Performance Improvements

Applications
Requirements Management

Features
Reduced time to display the Requirements result list

Use Case
In some cases it is helpful to display several hundreds or thousands of Requirements, e.g. to:
- use the app for an ad hoc reporting purposes
- do an xls export to create a customer specific reporting
- do an xls export to upload to another SolMan system
**Requirements Management – new with Focused Build SP06**

**New value 'No Solution'**

**Applications**
Requirements Management

**Features**
- New value 'No Solution' is available in the dropdown menu ‘Solution’
- It lists all Requirements without Solution assignment in Requirements Management.

**Use Case**
This feature allows to efficiently clean-up the Solution, e.g. identify Requirements, which have been created once, but not further processed.
Requirements Management – new with Focused Build SP06

‘Status’ Filter with Multi Selection

Applications
Requirements Management

Features
• Multi selection checkboxes for the ‘Status’ filter drop-down
• Allows an extended search pattern in the Requirements Management app

Use Case
The potential combination of Requirement status improves the reporting capabilities of the Requirements Management app, e.g. it allows to display the all Requirements:
• in the final status ‘Completed’ or ‘Canceled’
• which are approved, ‘In Realization’ or ‘Realized’ but not completed
Requirements Management – new with Focused Build SP06
Navigation Improvement on Requirement Detailed View

Applications
Requirements Management

Features
• Improved navigation for Requirements Management details pop-up
• Additional button ‘Save and New’

Use Case
In order to easily set the status ‘To be Approved’ directly after creating a new Requirement the remains open, after saving.

In case you want to create a new Requirement without setting the status ‘To be Approved’, there’s a new button ‘Save and New’

If you work with a Requirement in status ‘To Be Approved’ and save, the pop-up remains open, because the user might just want to save the current state of work.

When choosing the Action ‘Send for Approval’, the pop-up closes, because the Approval is done by another responsible
Requirements Management – new with Focused Build SP06
Direct Navigation to Requirements Management Detailed View

Applications
Requirements Management

Features
Direct Navigation to Requirements Management Detailed View via link

Use Case
In case you copy the Requirements Management Detailed View link (e.g. into a document), you are now directly directed to the detailed screen of the Requirement you copied it from.
Before SP06 only the link to the Requirements Overview is possible.
Mass Change – new with Focused Build SP06
New filter for Sub-Elements and Sub-Categories

Applications
Mass Change

Features
New filter possibility ‘Sub-Elements’ introduced for:

• Requirements

New filter possibility ‘Sub-Categories’ introduced for:

• Requirements
• Work Packages
• Work Items
• Defects
• Defect Corrections
• Change Requests
• Changes
• Risk

Use Case
Allows more granular filtering like in Requirements Management application.
Work Packages
Product Backlog Planning
Build Design: Example New Implementation

Fit Gap Analysis and Delta Design – Methodology Overview

### Step A
**Fit Gap Analysis / Solution Validation**

1. **Finalize System Setup**
   - Prepare additional sample data
   - Enhance system setup with add. scope

2. **Fit Gap Workshops / Validation of SAP Solution**
   - Show and tell SAP standard key design elements / Gap Identification

3. **Gap Documentation**
   - Document and specify identified Gaps in initial Backlog

4. **Delta Scope Prioritization**
   - Prioritization
   - Verification of identified gaps in backlog

5. **5.1 Create Product Backlog**
   - Prioritization according to technical criteria (e.g., Development sequence)
   - Document relevant configuration
   - Document solution for identified gaps e.g. One Delta Design for Sourcing and Procurement

6. **Verify & Accept**
   - Verify solution design
   - Acceptance Procedure

### Step B
**Delta Design**

5. **5.2 Design/Specify**
   - Specify the work to be done, e.g.
     - Create Delta Design Documents
     - Upload Configuration Documents
     - Prepare distribution of work by definition of Work Items

6. **Verify & Accept**
   - Verify solution design
   - Acceptance Procedure

- Workshops per Line of Business, e.g. Sourcing and Procurement:
  - Present solution based on delta design
  - Capture open issues if required
  - Obtain sign off

- For real Gaps and WRICEF, create Functional Specification in My Work Packages app and set according Work Package Status

- To sign-off release Functional Specification

- Approve Requirements to be part of release, with Mass Change app

- Plan all Waves at once > agile

- Result: Product Backlog

---

© 2020 SAP SE or an SAP affiliate company. All rights reserved. | PUBLIC
Release Planning: Roles and Responsibilities

1. Define Product Backlog

2. Prioritize Product Backlog

3. Analysis of Dependencies

4. Estimate Product Backlog

5. What would you like in the release?

6. Scope Release: Build logical and technical packages

High-Level Release Plan Based on Master Project

Detailed Release Plan based on Build Project
Step 5.1: How to define the Product Backlog?

FB Role: Project Manager, Architect
Functionality: Requirement Management

Procedure for Product Backlog preparation:
- Select Requirements in Status ‘Approved’ and in your responsibility, e.g. filter according process element or owner
- Create Work Packages and assign them to your Project
- Maintain Priority, Value or Effort Points

Procedure for Product Backlog Creation:
- In Mass Change Operation filter for Work Packages part of your project
- Assign Wave

Reporting:
- Solution Readiness Dashboard (‘To be Scoped’ Requirements, Work Packages), Mass Change Operation

Advantages:
- Good overview of Requirements in scope
- Efficient prioritization via filter and sort functionality

Fit Gap Analysis and Delta Design – Step 5.1
Define Product Backlog

Step B
Delta Design

5.1 Create Product Backlog
- Prioritization according to technical criteria (e.g. Development sequence)

Create Work Packages to be part of the first Wave
with Requirement Management app
Repeat this Wave planning activity for each Wave → agile
Plan all Waves at once → waterfall
Result: Product Backlog
Fit Gap Analysis and Delta Design – Step 5.2
Draft Delta Design

Step 5.2: How to do Delta Design?

FB Role: Architect
Functionality: My Work Packages

Procedure Delta Design:
- In Work Package upload Configuration Document or create Functional Specification

Reporting:
- Solution Readiness Dashboard (Functional Specifications)

Advantages:
- Automated Check, if KPI-date has been maintained by Project Manager. Solution Readiness Dashboard > tile Functional Specification > bar ‘Milestone Missing’
- Project Manager benefits from automated reporting on Functional Specifications, e.g. Overdue or Completed

For real Gaps and WRICEF, create Functional Specification in My Work Packages app
Fit Gap Analysis and Delta Design – Step 6
Verify & Accept

Step 6: How to do verification and sign-off of Delta Design?

FB Role: Architect, Customer
Functionality: My Work Packages
Procedure:
▪ Upload Functional Specification (drag and drop)
▪ Set the Document Status ‘In Review’
▪ Approver/Customer gets notified via e-mail
▪ Set’s document in status ‘Released’

Reporting:
▪ Solution Readiness Dashboard (Functional Specifications, Work Packages, Current Wave Progress)

Advantages:
▪ Project Manager benefits from automated reporting on Functional Specifications
▪ Progress tracking of all Work Packages
When to do: Planning the Product Backlog

- **Release**
  - Fit / Gap analysis WS
  - Prioritized/Ranked, rough effort estimation
  - Gap
  - Fit

- **Product Backlog**
  - Requirements
    - Gap, WRICEF, Fit (only configuration)
    - Non functional Requirements
  - Functional analysis & release planning (aggregation or splitting)
    - duplicates eliminated

- **Sprint Backlog**
  - Work Packages
    - Gap, WRICEF, Fit (config), Non functional Requirements
  - Sprint planning (Which dev team, which order)

- **Epics**
  - User Stories
  - User Stories

- **Content**
  - Waves
  - Sprints
  - Work Item
  - Work Item
  - Work Item
Waves

Wave details

- A wave comprises a well-defined functional scope of work packages to define what needs to be done
- Starts with scope definition and a preparation time
  - Provides at least the functional specification required to start the first sprint of the wave
- Actual build execution is done in Sprints
- Execute Single Functional Testing
- Ends with the q-gate “Wave exit-criteria fulfilment” for passing the q-gate
- Optional: Release can be assigned to a Wave to allow Go-Live after the Wave
Define Product Backlog

Example: Use Mass Change Application to define the Scope of the first Wave
Create Work Packages
Create Work Package
Guidelines

Distribute Work Packages (WPs)

- Follow the organizational structure of the team, e.g. P2P, Logistics, Master Data, …
- Each team works on its Product Backlog which is the summary of the assigned Work Packages

Work Package tailoring

- You need to find a balance between detailed planning/description and efficiency/flexibility.
  - Firstly enable functioning of standard business processes with high level process description and configuration guide as functional spec.
  - Then work on gaps and fits for this process during next wave.
    Here you do specific extensions to standard transactions or own developments
  - Note: don’t misunderstand the (n:m) Relationship of Requirement to Work Package
    n:1 is mainly used to combine similar Requirements

- A Work Package needs to be testable by an end- or key user (single functional and acceptance test)
- It needs to be realized in one wave
  → Try to slice small Work Packages at the beginning. We’ve seen a lot of Work Packages couldn’t be finished in one wave (then reassignment to next planned release needs to be done)
Create Work Package

Guidelines

A Work Package needs to be implementable in one wave (~4-12 weeks)

-> the smaller the Wave duration, the better the tracking and status based reporting

Example Waterfall approach with fix planned waves - rough Wave planning according to the WP type

- Wave 1: Fit (Customizing) -> baseline configuration
- Wave 2: Gap (Development) -> bigger developments, with integrative aspects
- Wave 3: WRICEF -> smaller developments and adjustments
- Wave 4: Left overs and Integration Testing

Example Scaled Agile approach with fix planned waves

- Wave 1: Plan WPs with the priority according to the ranking in the Product Backlog. Create appropriate Specification
- Wave 2: Plan WPs with the priority according to the ranking in the Product Backlog. Extend appropriate Specification
- Wave 3: like above

In case a Work Package is too big to finalize it in one Wave, slice it in several and smaller parts, e.g.

- one WP for developing the basic functionality
- one WP to extend the functionality.
- one WP for finish the functionality
Create Work Package
Guidelines

Maintain Work Packages

- Link to a process structure or library element
  - To automatically inherit the Process context, create Work Packages based on Requirements
  - Exception can be “basis module” customizing: then there is no direct link to a Requirement, but a direct link to a process structure element, e.g. Configuration Item in the Configuration Library

- To ease filtering, establish a nomenclature for all Work Package Titles, e.g. inherited by Requirement

- Assign at least one Functional Specification or Configuration Guide

- Subdivide into several Work Items in case the work is distributed between project members or dev. Teams
  → Don’t include too many developers in one Work Item, only in case they share the same timeline and similar effort
WP and WI Classification

When a Work Package (WP) is created it’s classification defaults to that of the requirement, but can be changed prior to the creation of Work Items (WI) and reaching WP status “to be Developed”.

Similarly, the WI classification defaults to that of the WP classification and can be changed when in status “Created”.

The assignment of the WP classification defines valid WI Classification options to be used (see below)

<table>
<thead>
<tr>
<th>WP Classification</th>
<th>WI Classification</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Gap</td>
</tr>
<tr>
<td>Gap</td>
<td>✓</td>
</tr>
<tr>
<td>WRICEF</td>
<td>✓</td>
</tr>
<tr>
<td>Fit (configuration)</td>
<td></td>
</tr>
<tr>
<td>Non-Functional</td>
<td></td>
</tr>
</tbody>
</table>
Create Work Package for Baseline Build

Step 1
Baseline build, as basis for Requirements creation in the Fit/Gap Workshops, shall be done on the Sandbox system:
- By default, it is not foreseen to maintain Requirements and Work Packages in Discover and Prepare project phase
- So, just activate the Best Practices via Solution Builder without explicit Focused Build specific reporting

Step 2
For the configuration of the Development system, it’s recommended to create one Requirement per Business Process. There are several options dependent of the configuration complexity and duration:

- Simple and quick configuration done in one Wave: 1 Requirement, 1 WP, 1 WI, 1 Transport
- Simple but long lasting config. done in several Waves: 1 Requirement, X WPs, 1 WI, 1 Transport
- Complex but quick config. done in one Wave: 1 Requirement, 1 WP, X WIs, X Transports
- Complex and long lasting config. done in several Waves: 1 Requirement, X WPs, X WIs, X Transports

The Work Packages which specify the Solution Builder activities are classified as Fit.
Work Package Workflow
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

- Fit RT
  - Handed over to Release

**Production Branch**

- Realized
- Completed

- Production Branch
  - Handed over to Release
  - Productive/Completed

- Scoping
  - Scoping
  - In Development

- To Be Tested
  - Successfully Tested
  - Handled over to Release

- In Realization
  - Scoping
  - In Development

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

© 2020 SAP SE or an SAP affiliate company. All rights reserved. | PUBLIC
Focused Build methodology – Release Phase
Standard Workflow

1. **Planned**
   - Approved
   - In Realization
   - Create Work Package
     - Created
     - Scoping
     - Scope Finalized
     - Create Work Item

2. **Prepare**
   - To Be Developed
     - In Development
   - Single Functional Test & AT
     - In Repair
     - To Be Tested
     - Successfully Tested
   - Unit Test
     - In Development
     - To Be Tested
     - Successfully Tested
   - Create Transports
     - Created
     - Deploy
     - In Repair
     - Create Transports
     - Created
     - Scoping
     - Scope Finalized
     - Create Work Item

3. **Build/Test**
   - FIT & RT
     - Handed over to Release
     - Hand over to Release
     - To Be Tested
     - Successfully Tested

4. **Deploy**
   - Realized
   - Completed
   - Productive/Completed

Requirements

Work Package

Work Item
SP06 Work Package Process Flow
Detailed Workflow Part I

The Functional Gap functionality is inactive by default
The Functional Gap functionality is inactive by default.

- **Postpone**
  - **Scope Change**
    - Set by WI
  - **Functional Gap**
    - Create Functional Gap
    - Set by WI
  - **In Development**
  - **Postponed**
  - **Extend Scope**
    - Reverse Scope Extension
  - **In Repair**
    - Set by WI
  - **To Be Tested**
    - Set by WI
    - Preliminary Import Requested
    - Confirm Successful Test
    - Successfully Tested
    - Handover to Release
  - **Productive**
    - Set by WI
  - **Completed (FINI)**

- **Postpone**
  - Functional Gap
  - Create Functional Gap
  - Set by WI
  - Preliminary Import Requested
  - Confirm Successful Test
  - Successfully Tested
  - Handover to Release
  - Productive
  - Completed (FINI)

- **Manual status setting**
- **Automated status setting**
Work Packages – Document Management
Document Maintenance with dropDoc

**dropDoc**

**Definition**
dropDoc helps as a part of Solution Documentation to manage numerous file types, which simplifies the default usability of file management inside Solution Documentation.
dropDoc can be integrated as a part of Work Package (WP), Work Item (WI) and Requirement applications.
In addition, dropDoc can be implemented directly, as a standalone option in the Solution Documentation scenario.

**An extract of dropDoc features**
- insert files using drag and drop for processes and steps
- mass maintenance of documents and documents type
- change the document status
- delete one or more documents at the same time
- optimized for different screen resolutions
- Etc.
Focused Build methodology – Document Upload

Standard Workflow

Upload Func. Spec (Spec. Status = Released)
Create Work Item type NC for Transports incl. Func. Spec. + Tech Design
Create Work Item type GC for Test Cases
Result
Work Packages are created and documentation is assigned

- In Focus Build we limit the documentation to the really required documentation
- So there needs to be a document describing what needs to be customized and tested
- The availability of the functional specs can be tracked via the Solution Readiness Dashboard
Functionality of dropDoc

dropDoc integration to Work Package and Work Item Applications

CREATE AND ASSIGN DOCUMENTS

There are several possibilities available to create a documents in dropDoc

The documents is always automatically assigned to structure and consequently to WP/WI
Functionality of dropDoc

dropDoc integration to Work Package and Work Item Applications

CREATE AND ASSIGN DOCUMENTS

1. Create documents by drag-and-drop
   - several documents can be selected from the local storage and dropped over the drop area

![Document Area with Drag-and-Drop Area]
Functionality of dropDoc

dropDoc integration to Work Package and Work Item Applications

DEFINE DOCUMENT

The decision where the dropped document will be stored is done in the background by dropDoc and based on the standard document type configuration e.g.:

- **Functional specification** shall be stored at `<Step origin>`
- **Single Functional Test** at `<Step origin>`
- **Technical Design** at development or executable elements
- **Use Case** at `<Step reference>`
Direct Creation of Work Items in WP Status To be Tested

In My Work Packages you can create Work Items in Work Package status ‘To be Tested’

**Benefit**

Early Single Functional Test: To upload Test Cases in Status ‘To be Tested’ you can now directly create Work Items. So there’s no need to go in back in status Scope Extension to create the Work Item there. This is only required if the Work Package has several Work Items assigned, otherwise you could upload the test case directly in the existing Work Item.
Work Packages Reporting
Work Package Tile

- Number of Work Packages assigned to selected Project
- Aggregated view on Work Packages per Category and Wave
- Comparison of Planned and Actual Effort
- Drill down to details
Functional Specification Tile

Percentage overview on Work Packages according to KPI definition for Functional Specification and related due dates.

**Completed:** Required document available and released

**To be done:** Required document does not meet KPI definition but Due Date has not been reached yet.

**Milestone Missing:** Date of related milestone or milestone is missing in project for assigned Wave

**Overdue:** Required document does not meet KPI definition and Due Date is in the past.

Click on bar allows drill down to list of relevant Work Packages

<table>
<thead>
<tr>
<th>Work Package</th>
<th>Title</th>
<th>Project Name</th>
<th>Status</th>
<th>Priority</th>
<th>WP Class</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>80000000455</td>
<td>WP01</td>
<td>Focused Build - Build Pr...</td>
<td>In Repair</td>
<td></td>
<td>GAP</td>
<td>Application</td>
</tr>
<tr>
<td>80000000456</td>
<td>Work Package 02</td>
<td>Focused Build - Build Pr...</td>
<td>Scoping</td>
<td></td>
<td>WRICEF</td>
<td>Unassigned</td>
</tr>
<tr>
<td>80000000457</td>
<td>Test Project</td>
<td>Focused Build - Build Pr...</td>
<td>Scope Ex...</td>
<td></td>
<td>Fit</td>
<td>Unassigned</td>
</tr>
<tr>
<td>80000000462</td>
<td>Test BP issue</td>
<td>Focused Build - Build Pr...</td>
<td>Scope Ex...</td>
<td></td>
<td>WRICEF</td>
<td>IT Infrastructure</td>
</tr>
<tr>
<td>80000000448</td>
<td>Sales Order improvement</td>
<td>Focused Build - Build Pr...</td>
<td>To Be De...</td>
<td></td>
<td>GAP</td>
<td>Project</td>
</tr>
<tr>
<td>80000000453</td>
<td>Sales Order improvement</td>
<td>Focused Build - Build Pr...</td>
<td>In Develop...</td>
<td></td>
<td>WRICEF</td>
<td>IT Infrastructure</td>
</tr>
</tbody>
</table>
New Current Wave Progress Tile

The actual completion rate in percentage across all work packages that are in scope of the current wave.

The Schedule tab displays the progress of the work packages of a selected wave. If you want to view a more detailed view of the current wave progress, you can also display the progress for work items per sprint by choosing Current Wave Progress for Work Items.

The KPIs tab shows the status of the documents for the work packages of the selected wave.
Work Items
Sprint Backlog Planning
Agile Project Execution with Focused Build
When to do: Sprint Planning

- **Release**
  - Epics
    - User Stories
  - Initial Backlog
    - Fit / Gap analysis WS
      - Gap
      - Prioritized/Ranked, rough effort estimation
      - Fit
  - Product Backlog
    - Requirements
      - Gap, WRICEF, Fit (only configuration)
      - Non functional Requirements
      - Functional analysis & release planning
      - (aggregation or splitting)
      - duplicates eliminated
    - Work Packages
      - Gap, WRICEF, Fit (config) Non functional Requirements
  - Sprint Backlog
    - Work Item
      - Sprint planning (Which dev team, which order)
    - Work Item
    - Work Item

- **Time**
  - Release
    - Waves
  - Sprints
How to come to a Sprint Backlog and process Work Items - Summary

Steps

The assignment and processing of Work Items is performed before and during each sprint.

Step 1: When the Architects perform the Handover to Development of the Work Packages the Work Items are generated.
   ▪ During the scoping of the Work Items the responsible Development Team should be already assigned.
   ▪ All Work Items of the same wave build the Sprint Backlog. In the Sprint Planning Meeting it is decided which Work Items will be processed in the next Sprint. A Work Item should be completed in one Sprint.

Step 2: When the Sprint Planning for the first Sprint is done, the development activities will be performed. In addition the Technical Design Document will be created and uploaded to the requested Solution Documentation Element.

Step 3: After the development has been finished the Unit Test will be performed.

Step 4: When the Unit Test was successful the developer confirms this by setting the appropriate status and continues with the next Work Item assigned to his team.
   ▪ The choice of the next Work Item can be aligned with the rest of the Development Team in the Daily Stand-up Meeting.

Step 5: After the Sprint is over Show & Tell Sessions will be performed to align with the Product Owner that the performed developments were done as requested or whether adjustments need to be done in the upcoming Sprint.
Sprints

Sprint details

- A sprint comprises a well-defined scope of work items to define how to do it
- Starts with sprint backlog planning and prioritize the work items
- Provides technical design documents and software developments for review in show and tell sessions / sprint reviews
- Unit test to confirm work item is completed
Define Sprint Backlog

Example: Use Mass Change Application to define the Scope of the first Sprint
Create Work Items
Create Work Items

Guidelines

A Work Item needs to be implementable in one sprint (~1-4 weeks)

You can have following relations between Work Package and Work Item:

- **1:1.** In case just one developer or consultant takes the action to fulfill the Work Package

- **1:n.** In case you want to:
  - Address different organizations and distribute required tasks (outsourced partners and internal organization)
    - Configuration goes to SAP consultant who is a member of the project
    - Frontend development goes to outsourced partner
    - Backend development goes to internal organization
  
  - Address different types of activities:
    - Configuration goes to consultant who is a member of the project
    - Frontend development goes to developer1 who is a member of the project
    - Backend development goes to developer2 who is a member of the project
To be able to start development and process a Work Item...

... the current Sprint needs to be released

Note: In case you release a Wave all underlying Sprints are automatically released. So if you want to work Agile, keep the Wave status ‘Created’ and release the Sprints only when the developers shall start to process their Work Items.
Tasks for Work Items

It’s possible to create tasks for Work Items

• There is a new “Tasks” tab in the “My Work Item” application to create/edit tasks

• For flexible use, the simple Task status schema is decoupled from Work Item status

Benefit

Break down of work to be done for a Work Item into smaller chunks to help developers organizing their work.
Work Items – Workflow
Work Items – Document Management
Relation between Solution Documentation, Requirements, Work Packages/Items

Overview

Processes

- Process Step Library
- Executable Library
- Development Library

Interface Library

Configuration Library

Requirements

- Functional Specification
- Test Case
- Interface Specification
- Technical Specification
- Configuration Guide

Work Package

- Requirements
- Description

Work Item

- Project & Wave

- Sprint

Processes

- Created
- Scoping
- Productive
- Handled over to release

Requirement

- Rejected
- Draft
- Approved
- Completed

Requirement Status

- Postponed
- SCOPE change

Work Package Status

- To Be developed
- Successfully tested

Project & Wave

- To be tested
- Successfully tested

Work Item Status

- Withdrawn
- Created
- In Development
- To be tested
Work Items Reporting
Work Item Tile

- Number of Work Items assigned to selected Project
- Aggregated view on Work Items per Category and Classification with drill down to details
Technical Design Tile

Percentage overview on Work Items according to KPI definition for Technical Design and related due dates.

**Completed:** Required document available and released

**To be done:** Required document does not meet KPI definition but Due Date has not been reached yet.

**Milestone Missing:** Date of related milestone or milestone is missing in project for assigned Sprint

**Overdue:** Required document does not meet KPI definition and related Due Date is in the past.

Click on bar allows drill down to list of relevant Work Items

<table>
<thead>
<tr>
<th>Work Item ID</th>
<th>Type</th>
<th>Title</th>
<th>Status</th>
<th>WI Classification</th>
<th>Category</th>
<th>Work Package</th>
<th>Wave</th>
<th>Dev...</th>
<th>Tester</th>
</tr>
</thead>
<tbody>
<tr>
<td>800001</td>
<td>NC</td>
<td>WORK ITEM 1</td>
<td>Creat...</td>
<td>Report</td>
<td>Appl...</td>
<td>Test ST-OST 200 SP04</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>800001</td>
<td>GC</td>
<td>WI2 from WP WO STRUCT</td>
<td>Creat...</td>
<td>Report</td>
<td>Una...</td>
<td>MSK_WO_STRUCTURE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>800001</td>
<td>GC</td>
<td>WI_for_myWP_01</td>
<td>Creat...</td>
<td>Workflow</td>
<td>Una...</td>
<td>OrderProcess WP_01</td>
<td></td>
<td></td>
<td>FB...</td>
</tr>
<tr>
<td>800001</td>
<td>GC</td>
<td>WI_AssignStruct_08_04</td>
<td>Creat...</td>
<td>N/A</td>
<td>Una...</td>
<td>WP_AssignStruct12</td>
<td>Wav...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>800001</td>
<td>GC</td>
<td>WI_AssignStruct_08_03</td>
<td>Creat...</td>
<td>N/A</td>
<td>Una...</td>
<td>WP_AssignStruct12</td>
<td>Wav...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>800001</td>
<td>GC</td>
<td>WI_AssignStruct_08_01</td>
<td>Suc...</td>
<td>Workflow</td>
<td>Una...</td>
<td>WP_AssignStruct13</td>
<td>Kafa...</td>
<td>Kafa...</td>
<td></td>
</tr>
<tr>
<td>800001</td>
<td>NC</td>
<td>WI test_02</td>
<td>Suc...</td>
<td>N/A</td>
<td>Una...</td>
<td>WP for WI test</td>
<td>Wav...</td>
<td>OST...</td>
<td></td>
</tr>
</tbody>
</table>
Development Tile

Percentage of Work Items according to status and due date.

**Completed:** The Work Items have the status To Be Tested or later and the related Due Date has not been reached yet.

**To be done:** Work Items don't have the status To Be Tested yet and the related Due Date has not been reached yet.

**Milestone Missing:** Date of related milestone or milestone is missing in project for assigned Sprint

**Overdue:** The Work Items don't have the status To Be Tested or later and the related Due Date is in the past.

<table>
<thead>
<tr>
<th>Wor...</th>
<th>Type</th>
<th>Title</th>
<th>Status</th>
<th>W...</th>
<th>Cat...</th>
<th>Work Package</th>
<th>Wave</th>
<th>Dev...</th>
<th>Tester</th>
<th>Dev...</th>
<th>Pri...</th>
</tr>
</thead>
<tbody>
<tr>
<td>800001</td>
<td>GC</td>
<td>Wi_AssignStruct08_01</td>
<td>Succ...</td>
<td>W...</td>
<td>N/A</td>
<td>WP_AssignStruc</td>
<td>Wav...</td>
<td>Kata...</td>
<td>Kata...</td>
<td>4: Low</td>
<td></td>
</tr>
<tr>
<td>800001</td>
<td>NC</td>
<td>Wi_test_02</td>
<td>Succ...</td>
<td>N/A</td>
<td>N/A</td>
<td>WP for Wi_test</td>
<td>Wav...</td>
<td>OST...</td>
<td>OST...</td>
<td>2: High</td>
<td></td>
</tr>
<tr>
<td>800001</td>
<td>GC</td>
<td>Wi_test_01</td>
<td>Succ...</td>
<td>N/A</td>
<td>N/A</td>
<td>WP for Wi_test</td>
<td>Wav...</td>
<td>OST...</td>
<td>OST...</td>
<td>2: High</td>
<td></td>
</tr>
<tr>
<td>800001</td>
<td>GC</td>
<td>Wi for testing SP04 21</td>
<td>To B...</td>
<td>N/A</td>
<td>N/A</td>
<td>ST-OST 200 SP 04 - Proj</td>
<td>Wav...</td>
<td>OST...</td>
<td>FB...</td>
<td>2: High</td>
<td></td>
</tr>
</tbody>
</table>
Unit Test Tile

Percentage of Work Items according to status and due date.

**Completed:** Work Items have the status ‘Successfully Tested’ or later

**To be done:** Status is not yet ‘Successfully Tested’ and the related Due Date has not been reached yet.

**Milestone Missing:** Date of related milestone or milestone is missing in project for assigned Sprint

**Overdue:** Status is not yet ‘Successfully Tested’ and the related Due Date is in the past.
What’s New with Focused Build SP6?

Work Package / Work Item
My Work Packages, My Work Items - New with SP06
Modified Preliminary Import Workflow

Applications
My Work Packages, My Work Items

Features
Work Items can only be preliminary imported to Production in case there is a 1:1-relationship between Work Package and Work Item. As soon as the Preliminary Import is requested in the Work Item the Work Package now reflects this status as well.

Use Case
A Work Package should be treated as a complete unit where all related Work Items are tested and deployed together. With the possibility of the preliminary import workflow it was possible to bypass this idea which led to inconsistent system landscapes, as parts of a Work Package were already in Production System while other parts were still in Development. With the limitation of the preliminary import option to 1:1-relationships between Work Packages and Work Items we ensure consistent system landscapes as a Work Package is always transported and tested as a complete package.

Preliminary Import not possible. More than one Work Item exist.
My Work Packages, My Work Items - New with SP06
Display Long Text Information of Error Messages in UI5

Applications
My Requirements, My Work Packages, My Work Items, My Defects, My Defect Corrections, My Risks, My Requests for Change, My Change Documents

Features
When an error or warning message is raised in UI5-application a link is visible in the popup to get the helpful long text information displayed.

Use Case
So far the valuable long text information of a warning or error message were only available in the CRM UI. End users need to jump from the UI5-application to CRM UI to get this information displayed. Now there is an additional link in the short text window of the message available that opens the long text information directly in the UI5-application.
My Work Packages, My Work Items - New with SP06
Automated setting of completion rate to 100%

Applications
My Work Packages, My Work Items

Features
When a Work Item or a Work Package is set to ‘Successfully tested’ the completion rate is set automatically to 100%.

Use Case
The completion rate is not always used by customers or the project team. As a consequence the field remains 0 although the development of the Work Item has been finished or the test of the Work Package was successful which lead to incorrect reporting in the Solution Readiness Dashboard. Also the tile ‘Current Wave Progress’ shows a more accurate result if the completion rate is automatically filled.
My Work Packages, My Work Items - New with SP06
New validation for completed checklist

Applications
My Work Packages, My Work Items

Features
When a Work Item or a Work Package is set to ‘Successfully tested’ it is checked whether all checklist steps have been completed. If not an error message is raised and the status is not set.

Use Case
A Work Package or Work Item can only be completed if all related items are closed. Also checklist steps are checked if they are finalized, if this is not the case the completed status is not set. This caused problems in the past as the Completed-status is usually set via the Release Batch Import automatically after the successful import in the Production System. An open checklist step prevented this and led to manual extra work after the Go-Live. With the new check we inform Developers and Solution Architects at the right time about open items.
Mass Change – new with Focused Build SP06
Improved Display of Column ‘Release’

Applications
Mass Change

Features
Improved display of Release information in columns:
• Actual Release
• Requested Release

Harmonization of Release display, e.g. in the value help, the Release is now displayed as a three-digit number to the right of the description

Use Case
In Mass Change - Actual and Requested Release:

In the result lists, the Release is displayed as a three-digit number.

The full Release description is displayed via tooltip.

This reduces the table widths and allows to display more results, especially on smaller screens
Manage Changes in SFT and AT with Defect Corrections
(for more details see L2 presentation Defect & Defect Correction)
Wave End: Perform Single Functional Test and Acceptance Test

- After the successful test on Work Item Level the related Work Packages will be ready for SFT and AT
- AT should be executed in a formal way with Test Suite Integration to leverage all benefits, e.g. automatic status update of affected Work Package (set to ‘in Repair’)
- After all Test Cases related to a Work Package are successfully completed the status of the Work Package is set to ‘Successfully tested’

Test Suite Integration with SP06:
Integrated Defect Correction - Details

Best Practice is to use the Integrated Defect Correction, which means:

- Automatic assignment of Defect Corrections to Work Package with visibility in ‘Scope’ Tab
- New Work Package Status ‘In Repair’
- In WP Status ‘Handed over to Release’ correlated Defect Corrections are switched to this Status

Benefit

While Single Functional Test: Existing Defect Corrections are automatically assigned to a Work Package in case of Assignment Analysis Usage and a 1:1 relationship of WP and Test Package.

Automated switch of Work Package Status in case of new and confirmed Defect Correction
Build the Release
Decide on Work Packages to be part of the Release

- After switching the Release Phase to 'Build' use the Release Dashboard as starting point for the decision which Work Packages shall be part of the next release
- Release Dashboard bridges the gap between WP status and readiness of underlying transport to be imported to Preproduction system
- Release Manager has more decision criterias which WP shall be part of the release and which shall be left behind (e.g. due to missing documents)
- Direct access from Release Dashboard to Mass Change app to take over Release Selection and perform the status switch to ‘Handed over to Release’ (Point of no Return!)

<table>
<thead>
<tr>
<th>Work Packages (125)</th>
<th>Standard</th>
<th>Include Work Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rating Work Pack...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>△ 8000002631</td>
<td></td>
<td>△</td>
</tr>
<tr>
<td>□ 8000004766</td>
<td></td>
<td>□</td>
</tr>
<tr>
<td>□ 80000004773</td>
<td></td>
<td>□</td>
</tr>
<tr>
<td>□ 8000004774</td>
<td></td>
<td>□</td>
</tr>
<tr>
<td>Description Work...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISD-03 execute O...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create_Email_not...</td>
<td></td>
<td>□</td>
</tr>
<tr>
<td>Create_new_quot...</td>
<td></td>
<td>□</td>
</tr>
<tr>
<td>New-check_befor...</td>
<td></td>
<td>□</td>
</tr>
<tr>
<td>Rating Document...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>△</td>
<td></td>
<td>□</td>
</tr>
<tr>
<td>□</td>
<td></td>
<td>□</td>
</tr>
<tr>
<td>□</td>
<td></td>
<td>□</td>
</tr>
<tr>
<td>□</td>
<td></td>
<td>□</td>
</tr>
</tbody>
</table>
Go-Live
(for more details see L2 presentation Deployment and Release)
What happens with Work package, Work Item and Requirement at Go-Live
Combined Productive and Completed status

- When switching to release phase ‘Hypercare’ the Work Packages and Work Items must be in status ‘Completed’ to express they were finished in the current release. The status ‘Productive’ is not sufficient.
- It is not possible to set the status ‘Hypercare’ in the Release if there are non-completed Work Packages and Work Items with status ‘Handed over to Release’ or later (e.g. ‘Productive’, ‘Preliminary Import Requested’)
- Default since ST-OST SP2: Batch Import automatically sets (via asterisk setting, triggering standard after import status setting) Status ‘Completed’ for Normal Changes (automated change with transport)
- To have an additional status ‘Completed’ with the FINI status instead of the status ‘Productive’ doesn’t seem offer additional value. But the benefit of this behavior is: Enabling the 4 eyes principle for General Changes (manual change without transport) in productive environment
Focus Build methodology
Standard Workflow at Go-Live

Parallel documentation activation to Production Branch and transport to Production System for Work Items (NC)

Work Items (GC) are set via Mass Change to ‘Productive’ which triggers the activation of Solution Documentation Elements to Production Branch

Handed over to Release cannot be set for Work Packages if assigned Documents and Test Steps are not released

Warning appears at “Successfully tested” in Work Items to inform developer about missed activities

© 2020 SAP SE or an SAP affiliate company. All rights reserved. | PUBLIC
Thank you.