



**PUBLIC (ÖFFENTLICH)**

Dokumentversion: Draft – 2020-05-11

# Test-Suite

# Inhalt

<b>1</b>	<b>Test-Suite</b> .....	<b>5</b>
<b>2</b>	<b>Launchpad-Gruppe „Test-Suite“</b> .....	<b>7</b>
<b>3</b>	<b>SAP Solution Manager – Test-Suite</b> .....	<b>10</b>
3.1	Test-Suite in SAP Solution Manager .....	11
3.2	Testautomatisierung mit dem Testautomatisierungs-Framework .....	14
3.3	Partner-Testmanagement .....	15
	Konfiguration des Partner-Testmanagements .....	16
3.4	Infrastruktur und Testwerkzeuge .....	16
	SUT-Verwaltung .....	17
	Testkonfiguration im Testautomatisierungs-Framework .....	17
	Testautomatisierung mit dem Testautomatisierungs-Framework .....	25
	Komponentenbasierte Testautomatisierung .....	26
<b>4</b>	<b>Konfiguration der Test-Suite</b> .....	<b>121</b>
<b>5</b>	<b>Verwaltung - Test-Suite</b> .....	<b>123</b>
<b>6</b>	<b>Meine Aufgaben: TBOM-Arbeitsvorrat</b> .....	<b>126</b>
6.1	Überwachen des TBOM-Aufgabenstatus als Qualitätsexperte .....	127
6.2	Bearbeiten von TBOM-Aufgaben als Geschäftsprozess-Experte .....	128
<b>7</b>	<b>Business Process Change Analyzer (BPCA)</b> .....	<b>130</b>
7.1	Verwaltung - Änderungseinfluss-Analyse .....	131
7.2	Vorbereitung: Anlegen von technischen Stücklisten (TBOMs) .....	132
	Was ist eine Technical Bill of Materials (TBOM)? .....	133
	TBOM-Arten auswählen .....	136
	Site-abhängige TBOMs .....	139
	Anlegen von statischen oder semidynamischen TBOMs .....	140
	Voraussetzungen für Sonderfälle der TBOM-Aufzeichnung .....	143
	Delegieren von TBOM-Aufzeichnungen an Geschäftsexperten .....	148
	Anlegen dynamischer TBOMs .....	150
	Bearbeiten der TBOM-Gewichtung .....	156
	Prüfen von TBOM-Abdeckung und -Status .....	158
	Identifizieren und Handhaben veralteter TBOMs .....	163
7.3	Analysieren der von Änderungen betroffenen Geschäftsprozesse .....	170
7.4	Anzeigen von Details zu Änderungsvorgängen .....	174
7.5	Analysieren der von Änderungen betroffenen Geschäftsprozessvarianten .....	175

7.6	Analyseergebnisse auswerten und Testpläne erstellen. . . . .	176
	TBOM-Filter bearbeiten. . . . .	178
	Prüfen des TBOM-Contents. . . . .	179
7.7	Optimieren des Testumfangs. . . . .	180
	Gruppieren von ausführbaren Einheiten für die Testumfangsoptimierung. . . . .	183
	Bearbeiten von Standardzeiten zum Anlegen und Ausführen von Testfällen. . . . .	184
	Optimierungsoptionen. . . . .	186
	Festlegen von Testumfang oder Zeitaufwand. . . . .	195
	Anpassen des Testumfangs in einer Grafik. . . . .	196
	Testumfang manuell übersteuern. . . . .	198
	Prüfen von Objekten, die nicht im Testumfang enthalten sind. . . . .	199
	Ausschließen von Objekten von der Optimierung. . . . .	200
	Mit Optimierungsansätzen arbeiten. . . . .	201
	Anlegen oder Erweitern von Testplänen. . . . .	203
7.8	Verwenden von Testplänen und Testfällen aus Partnerwerkzeugen. . . . .	203
	Partner-Testwerkzeug registrieren. . . . .	204
	Testfalltypen und -attribute für Partnerwerkzeuge. . . . .	205
	Zu einem Testplan oder Testfall im Partnerwerkzeug navigieren. . . . .	206
	Durchführen der Änderungseinfluss-Analyse mit dem Partnerwerkzeug. . . . .	207
<b>8</b>	<b>Umfangs- und Aufwandsanalyse. . . . .</b>	<b>208</b>
8.1	Anlegen einer Analyse. . . . .	210
8.2	Analyseergebnisse. . . . .	211
	Weitere Empfehlungen für kundeneigene Entwicklungen. . . . .	213
<b>9</b>	<b>Test-Repository: Testskripte und Testkonfigurationen. . . . .</b>	<b>216</b>
9.1	Anlegen von Testkonfigurationen und Testskripten. . . . .	217
	Definieren von Testkonfigurationsattributen. . . . .	220
	Anlegen eines Testskripts. . . . .	222
	Bearbeiten von Testskripten. . . . .	223
	Definieren von Eingabe- und Ausgabeparametern. . . . .	226
	Weitere Einstellungen für Testkonfigurationen und Testskripte. . . . .	227
	Verwalten von Komponenten automatisierter Testfälle. . . . .	229
9.2	Testkonfigurationen und Testskripte ausführen. . . . .	230
<b>10</b>	<b>Testfälle. . . . .</b>	<b>232</b>
<b>11</b>	<b>Testplanverwaltung. . . . .</b>	<b>235</b>
11.1	Workflow und Benachrichtigungen. . . . .	237
11.2	Workflow für Testdokument-Signaturen. . . . .	238
11.3	Anlegen und Bearbeiten von Testplänen. . . . .	239
11.4	Testpläne kopieren. . . . .	246
11.5	Freigabestatusschema. . . . .	248

11.6	Planen von Testzeitplänen und -aufwänden. . . . .	252
11.7	Testpakete anlegen und bearbeiten. . . . .	253
11.8	Kopieren von Testpaketen und Testsequenzen. . . . .	255
11.9	Zuordnen von Testern. . . . .	256
	Zuordnen von Tester-Teams. . . . .	258
	Vertreter zuordnen. . . . .	258
<b>12</b>	<b>Meine Aufgaben - Tester-Arbeitsvorrat. . . . .</b>	<b>260</b>
12.1	Manuelle Testfälle ausführen. . . . .	262
12.2	Ausführen automatisierter Testfälle. . . . .	265
12.3	Bearbeiten von Testnotizen. . . . .	267
12.4	Anlagen von Testpaketen anzeigen. . . . .	268
12.5	Einplanen automatischer Tests. . . . .	268
12.6	Zuordnen von Vorgangsarten für Test-Suite. . . . .	269
<b>13</b>	<b>Tester-Arbeitsvorrat überprüfen. . . . .</b>	<b>271</b>
<b>14</b>	<b>Personalisierung. . . . .</b>	<b>272</b>
<b>15</b>	<b>Beschädigte Testfälle. . . . .</b>	<b>273</b>
<b>16</b>	<b>Testausführungsreports. . . . .</b>	<b>274</b>
<b>17</b>	<b>Test-Suite - Analysen. . . . .</b>	<b>275</b>
17.1	Vollständigkeits- und Lücken-Reports. . . . .	275
17.2	Testausführungsanalyse. . . . .	276
17.3	Status- und Fortschrittsanalyse (BW-basiert). . . . .	279

# 1 Test-Suite

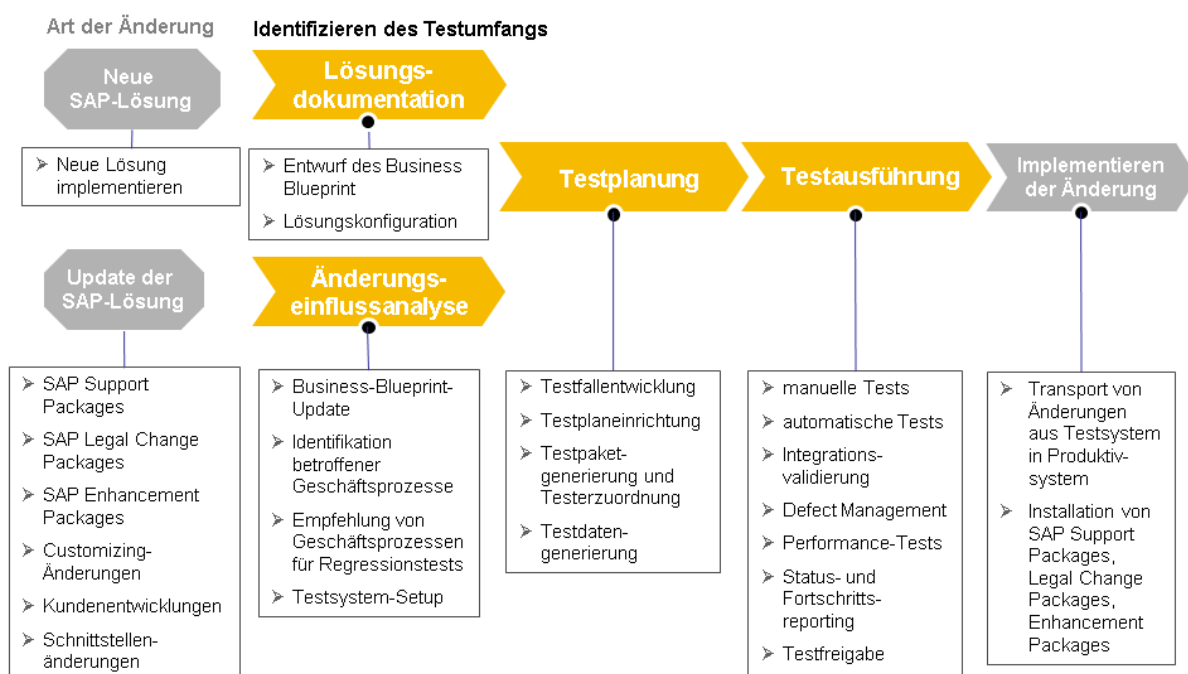
## Verwendung

Sie können den erforderlichen Testumfang für systemübergreifende Geschäftsprozesse ermitteln, die Tests zentral verwalten und die Tests durchführen.

Das Testmanagement umfasst die folgenden Phasen:

1. Identifizieren des Testumfangs  
Bei Einführung neuer SAP-Lösungen ergibt sich der Testumfang aus der [Lösungsdokumentation](#). Bei Änderungen des Systems (z.B. bei Aktivierung von Support Packages) ermitteln Sie die kritischen Geschäftsprozesse mithilfe einer [Änderungseinfluss-Analyse \[Seite 130\]](#).
2. Testplanung  
Sie richten ein Testsystem ein, generieren Testdaten, entwickeln manuelle oder automatisierte Testfälle, legen Testpläne an und ordnen Testpakete den richtigen Testern zu.
3. Testen  
Sie veranlassen die Ausführung manueller und automatischer Tests und überprüfen den Fortschritt und das Ergebnis der Tests.
4. Übernahme der Änderungen in die Produktion

Die folgende Grafik veranschaulicht die Phasen des Testmanagements:



Phasen des Testmanagements

## Integration

Bevor Sie mit dem Testen beginnen können, müssen Sie in der Lösungsdokumentation eine Projektstruktur mit Ihren Geschäftsprozessen anlegen und Ihre Geschäftsprozesse konfigurieren. Weitere Informationen finden Sie unter [Lösungsdokumentation](#).

Sie können sowohl die Testwerkzeuge von SAP als auch Lösungen von Partnern einsetzen und diese wie folgt kombinieren:

- [Test-Suite in SAP Solution Manager \[Seite 10\]](#) und Test Data Migration Server (TDMS, siehe <https://support.sap.com/solution-manager/integrated-tools/tdms.html>)
- SAP Solution Manager, SAP TDMS, [SAP Quality Center by HP \[Seite 15\]](#) und SAP Test Acceleration and Optimization (siehe <http://help.sap.com/saptao30>)
- SAP Solution Manager, SAP TDMS, SAP Quality Center und sonstige Werkzeuge für das Partner-Testmanagement

## 2 Launchpad-Gruppe „Test-Suite“

### Verwendung

Diese [Launchpad](#)-Gruppe ist der zentrale Einstiegspunkt für den Zugriff auf alle notwendigen Informationen und Funktionen, die Sie zur Ermittlung des für die systemübergreifenden Prozesse erforderlichen Testumfangs, zur zentralen Verwaltung der Tests und zur Ausführung der Tests benötigen.

### Funktionsumfang

Die Gruppe besteht aus den folgenden standardmäßig angezeigten Kacheln:

Kachel	Beschreibung
<a href="#">Test-Suite – Übersicht</a>	<p>Sie können die Statusübersichten zu folgenden Themen prüfen, die auf Registerkarten bereitgestellt werden:</p> <ul style="list-style-type: none"><li>• Registerkarte <a href="#">Testplanung</a>: Testpläne und die zugeordneten Testfälle und Testpakete im gesamten System, mit Informationen wie Datumsangaben, verantwortlichen Personen, Anzahl der Tests, Testfallstatus und Fehler.</li><li>• Registerkarte <a href="#">Testausführung</a>: Die individuellen Testpläne des Benutzers und die zugeordneten Testfälle und Testpakete, mit denselben Informationen wie unter Testmanagement.</li><li>• Registerkarte <a href="#">Business Process Change Analyzer</a>: Anzahl der für ein System gefundenen Änderungen, die durch Support Packages, Enhancement Packages, Business Functions oder Transporte verursacht wurden.</li></ul>
<a href="#">Test-Suite – Testvorbereitung</a>	<p>Sie können Ihre Lösungshierarchie prüfen. Navigieren Sie zu den enthaltenen ausführbaren Einheiten, Testfällen, zugehörigen Dokumenten (z.B. Geschäftsanforderungen und Meldungen) und Testinformationen. Von hier aus können Sie Vollständigkeits- und Lücken-Reports für die Lösung ausführen.</p>
<a href="#">Partner-Testmanagement</a>	<p>Sie können die Assoziationen zwischen Ihrer Lösung im SAP Solution Manager und den Testprojekten von Partnerwerkzeugen verwalten.</p>
<a href="#">Meine Aufgaben – TBOM-Arbeitsvorrat</a>	<p>Sie können TBOM-Erfassungsaufgaben als Geschäftsexperte oder Tester anzeigen und bearbeiten. TBOMs sind für das Ausführen von Änderungsanalysen mit dem Business Process Change Analyzer erforderlich. Weitere Informationen finden Sie unter <a href="#">Meine Aufgaben: TBOM-Arbeitsvorrat [Seite 126]</a>.</p>

Kachel	Beschreibung
<a href="#">Business Process Change Analyser</a>	Sie können Änderungen am System vornehmen, z.B. Support Packages, Kundenentwicklungen oder Add-on-Installationen, die sich auf verschiedene Geschäftsprozesse auswirken können. Um sicherzustellen, dass die Geschäftsprozesse auch nach einer solchen Änderung noch einwandfrei funktionieren, identifizieren und testen Sie die betroffenen Geschäftsprozesse oder Teile davon. Weitere Informationen finden Sie unter <a href="#">Business Process Change Analyser (BPCA) [Seite 130]</a> .
<a href="#">Umfangs- und Aufwandsanalyse - Upgrade-Planung</a>	Sie können den Entwicklungs- und Testaufwand für geplante Wartungsprojekte zum Implementieren von Enhancement Packages oder Support Packages schätzen. Die Anwendung liefert Ihnen eine Übersicht über die Analysen und stellt mehrere Optionen zur Bearbeitung der Analysen bereit. Weitere Informationen finden Sie unter <a href="#">Umfangs- und Aufwandsanalyse [Seite 208]</a> .
<a href="#">Testplanverwaltung</a>	Sie können als Test Organizer diese Sicht verwenden, um Testpläne und -pakete zu verwalten (z.B. Testern Testpakete hinzufügen und Anlagen festlegen). Weitere Informationen finden Sie unter <a href="#">Testplanverwaltung [Seite 235]</a> .
<a href="#">Tester-Arbeitsvorrat überprüfen</a>	Sie können alle einem bestimmten Tester zugeordneten Testfälle aufrufen. Weitere Informationen finden Sie unter <a href="#">Tester-Arbeitsvorrat überprüfen [Seite 271]</a> .
<a href="#">Meine Aufgaben – Tester-Arbeitsvorrat</a>	Sie können alle Testfälle aufrufen, die Ihnen als Tester zugeordnet wurden. Weitere Informationen finden Sie unter <a href="#">Meine Aufgaben - Tester-Arbeitsvorrat [Seite 260]</a> .
<a href="#">Meine Aufgaben – beschädigte Testfälle</a>	Sie können alle beschädigten Testfälle aufrufen. Weitere Informationen finden Sie unter <a href="#">Beschädigte Testfälle [Seite 273]</a> .
<a href="#">Testausführung – Protokoll</a>	Sie können die Ausführungsprotokolle automatisierter Testfälle anzeigen. Weitere Informationen finden Sie unter <a href="#">Testausführungsreports [Seite 274]</a> .
<a href="#">Testausführung – Jobs</a>	Sie können Informationen zu den Hintergrundjobs anzeigen, mit denen Sie die Ausführung automatischer Testfälle eingeplant haben: <ul style="list-style-type: none"> <li>• <b>Jobstatus:</b> Hat die Ausführung stattgefunden?</li> <li>• <b>Teststatus:</b> Welches Ergebnis hatte die Ausführung?</li> <li>• <b>Protokoll-ID:</b> ID des Ausführungsprotokolls für den Job, dem ein Testpaket zugeordnet ist.</li> <li>• <b>Jobname:</b> Link zu Ausführungsdetails des Jobs</li> </ul> Weitere Informationen finden Sie unter <a href="#">Testausführungsreports [Seite 274]</a> .
<a href="#">Test-Repository – Testkonfigurationen</a>	Sie können über das Test-Repository auf die Umgebung für die Testzusammenstellung zugreifen, in der Sie automatische Testfälle anlegen und bearbeiten können. Weitere Informationen finden Sie unter <a href="#">Test-Repository: Testskripte und Testkonfigurationen [Seite 216]</a> .
<a href="#">Test-Repository – Testskripte</a>	Sie können über das Test-Repository auf einzelne Testskripte zugreifen, die Bestandteil der Testkonfigurationen sind. Weitere Informationen finden Sie unter <a href="#">Test-Repository: Testskripte und Testkonfigurationen [Seite 216]</a> .



<b>Kachel</b>	<b>Beschreibung</b>
<i>Test-Suite – Analysen</i>	Sie können unterschiedliche dem Testmanagement zugehörige Reports einsehen. Weitere Informationen erhalten Sie unter <a href="#">Test-Suite - Analysen [Seite 275]</a>
<i>Personalisierung – Test-Suite</i>	Sie können die Personalisierungsanwendung aufrufen, um Ihre bevorzugten Einstellungen zu definieren. Weitere Informationen finden Sie unter <a href="#">Personalisierung [Seite 272]</a> .

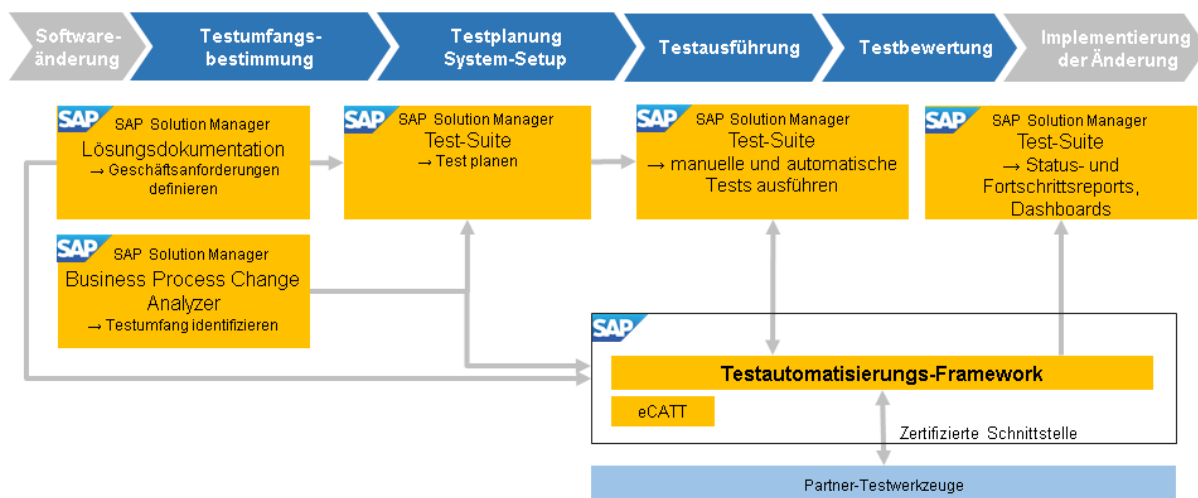
Die folgenden Kacheln werden nicht standardmäßig angezeigt:

<b>Kachel</b>	<b>Beschreibung</b>
<i>Konfiguration – Test-Suite</i>	Sie können die Solution-Manager-Konfiguration für Test-Suite-Szenarios aufrufen. Weitere Informationen finden Sie unter <a href="#">Konfiguration der Test-Suite [Seite 121]</a> .
<i>Verwaltung – Test-Suite</i>	Sie können die Grundeinstellungen für den Business Process Change Analyzer (BPCA) aufrufen. Weitere Informationen finden Sie unter <a href="#">Verwaltung - Test-Suite [Seite 123]</a> .
<i>Verwaltung – Änderungseinfluss-Analyse</i>	Sie können die Grundeinstellungen für die Test-Suite aufrufen. Weitere Informationen finden Sie unter <a href="#">Verwaltung - Test-Suite [Seite 123]</a> .

# 3 SAP Solution Manager – Test-Suite

## Verwendung

Die Testfunktionen und Werkzeuge werden fast ausschließlich von SAP Solution Manager und vom SAP Test Data Migration Server bereitgestellt. Die nachfolgende Abbildung verdeutlicht den Prozess.



SAP Solution Manager u. Testautomat.-Framework

## Prozess

1. Sie identifizieren den Testumfang.  
Je nach Art der Änderung führen Sie folgende Prozessschritte aus:
  - **Neue SAP-Lösung:** Mit den SAP-Solution-Manager-Funktionen für die **Lösungsdokumentation** entwerfen, dokumentieren und katalogisieren Sie Geschäftsprozesse in hierarchischer Form. Die Fachabteilungen formulieren geschäftliche Anforderungen in der Lösungsdokumentation. Sie ordnen den Geschäftsprozessen in der Lösungsdokumentation manuelle und automatische Testfälle zu. Mit dem **Testautomatisierungs-Framework [Seite 17]** legen Sie direkt im Kontext der Geschäftsprozesse automatisierte Testfälle auf Basis der **Testautomatisierung mit eCATT [Seite 14]**, der komponentenbasierten Testautomatisierung (**Basisinformationen [Seite 26]**) oder zertifizierter externer Werkzeuge an. Ihre Testfälle können Sie im **Test-Repository [Seite 216]** verwalten.
  - **Update einer bestehenden SAP-Lösung:** Mit dem **Business Process Change Analyzer [Seite 130]** von SAP Solution Manager prüfen Sie, ob wichtige Anwendungen oder kritische Geschäftsprozesse von dem Update betroffen sind.
2. Sie planen die Tests und setzen Testsysteme auf.
  - Der SAP Test Data Migration Server (TDMS) unterstützt Sie beim automatischen Aufbau von Testsystemen mit geringen Datenvolumen. (Weitere Informationen finden Sie unter <https://support.sap.com/solution-manager/integrated-tools/tdms.html>).

- Legen Sie eine der folgenden Arten von Testplänen an:
  - In der Anwendung [Testplanverwaltung \[Seite 235\]](#) können Sie Testpläne anzeigen, ändern, kopieren oder löschen.  
In der Anwendung [Business Process Change Analyzer](#) können Sie Geschäftsprozesse analysieren, die von einer Änderung betroffen sind und erneut getestet werden müssen. Basierend auf den Analyseergebnissen erstellen Sie maßgeschneiderte Testpläne.
  - Fassen Sie ausgewählte Testfälle in Testpaketen zusammen, und ordnen Sie den Testern die Testpakete zu.
  - Planen Sie automatische Tests ein, z.B. jede Nacht.
  - Sie können zur besseren Kontrolle der Testausführung einen Workflow einrichten, der auf einem [Freigabestatusschema \[Seite 248\]](#) basiert. Tester erhalten eine E-Mail, in der ihnen mitgeteilt wird, wann sie mit der Ausführung ihrer zugewiesenen Testpakete anfangen können oder wann die Ausführung angehalten werden muss.
- 3. Sie führen die Tests aus.  
Sie nutzen die Gruppe [Test-Suite](#) im SAP Solution Manager Launchpad als zentralen Einstieg, um Tests auszuführen und zu analysieren.
  - In der Anwendung [Meine Aufgaben - Tester-Arbeitsvorrat \[Seite 260\]](#) sieht der Tester eine Liste aller ihm zugewiesenen Testpakete.
  - Der Tester kann seinen Arbeitsvorrat auch direkt über den Link in der E-Mail-Benachrichtigung aufrufen und abarbeiten.
  - Über [Test-Suite - Analysen \[Seite 275\]](#) können Sie den Status Ihrer Testfälle und Testausführungen überwachen. Mit den Funktionen des BI-Reportings haben Sie die Möglichkeit, grafisch aufbereitete Übersichten über den Status der Testpläne oder den Verlauf der Bearbeitung anzuzeigen.

## 3.1 Test-Suite in SAP Solution Manager

### Verwendung

Die Gruppe [Test-Suite](#) im SAP Solution Manager Launchpad dient als zentraler Einstiegspunkt in sämtliche Funktionen zum Vorbereiten, Anlegen, Verwalten und Auswerten von Testplänen, Testskripten, Testkonfigurationen usw.

Darüber hinaus können Sie mit dem [Business Process Change Analyzer \(BPCA\) \[Seite 130\]](#) Änderungseinfluss-Analysen durchführen und aus den Ergebnissen maßgeschneiderte Testpläne generieren. Mit der Umfangs- und Aufwandsanalyse können Sie den Aufwand für das Testen und Installieren neuer Software berechnen.

### Funktionsumfang

Über die Gruppe [Test-Suite](#) im SAP Solution Manager Launchpad können Sie die folgenden Sichten starten:

#### Test-Suite - Übersicht

Diese Anwendung stellt auf mehreren Registerkarten Statusübersichten zu folgenden Bereichen bereit:

- Registerkarte *Testplanung*: Testpläne und die zugeordneten Testfälle und Testpakete im gesamten System, mit Informationen wie Datumsangaben, verantwortlichen Personen, Anzahl der Tests, Testfallstatus und Fehler.
- Registerkarte *Testausführung*: Die individuellen Testpläne des Benutzers und die zugeordneten Testfälle und Testpakete, mit denselben Informationen wie unter *Testmanagement*.
- Registerkarte *Business Process Change Analyzer*: Anzahl der für ein System gefundenen Änderungen, die durch Support Packages, Enhancement Packages, Business Functions oder Transporte verursacht wurden.

### **Test-Suite - Testvorbereitung**

Übersicht über Ihre Lösungshierarchie. Sie können zu den enthaltenen ausführbaren Einheiten, Testfällen, zugehörigen Dokumenten (z.B. Geschäftsanforderungen und Meldungen) und Testinformationen navigieren. Von hier aus können Sie Vollständigkeits- und Lücken-Reports für die Lösung ausführen.

### **Partner-Testmanagement**

In dieser Anwendung können Sie die Assoziationen zwischen Ihrer Lösung im SAP Solution Manager und den Testprojekten von Partnerwerkzeugen verwalten.

### **Meine Aufgaben - TBOM-Arbeitsvorrat**

Zeigen Sie TBOM-Erfassungsaufgaben als Geschäftsexperte oder Tester an, und bearbeiten Sie sie. TBOMs sind für das Ausführen von Änderungsanalysen mit dem Business Process Change Analyzer erforderlich. Weitere Informationen finden Sie unter [TBOM-Arbeitsvorrat \[Seite 126\]](#).

### **Business Process Change Analyzer**

Änderungen am System, z.B. Support Packages, Kundenentwicklungen oder Add-on-Installationen, können sich auf verschiedene Geschäftsprozesse auswirken. Um sicherzustellen, dass die Geschäftsprozesse auch nach einer solchen Änderung noch einwandfrei funktionieren, identifizieren und testen Sie die betroffenen Geschäftsprozesse oder Teile davon. Weitere Informationen finden Sie unter [Business Process Change Analyzer \[Seite 130\]](#).

### **Umfangs- und Aufwandsanalyse**

Ermittelt den geschätzten Entwicklungs- und Testaufwand für geplante Wartungsprojekte zum Implementieren von Enhancement Packages oder Support Packages. Weitere Informationen dazu finden Sie unter [Umfangs- und Aufwandsanalyse \[Seite 208\]](#).

### **Testplanverwaltung**

Verwenden Sie als Test Organizer diese Sicht, um Testpläne und -pakete zu verwalten (z.B. Testern Testpakete hinzufügen und Anlagen festlegen). Weitere Informationen finden Sie unter [Testplanverwaltung \[Seite 235\]](#).

### **Tester-Arbeitsvorrat überprüfen**

Rufen Sie alle einem bestimmten Tester zugeordneten Testfälle auf. Weitere Informationen finden Sie unter [Tester-Arbeitsvorrat überprüfen \[Seite 271\]](#).

### **Meine Aufgaben - Tester-Arbeitsvorrat**

Greifen Sie auf alle Testfälle zu, die Ihnen als Tester zugeordnet wurden. Weitere Informationen finden Sie unter [Tester-Arbeitsvorrat \[Seite 260\]](#).

### **Meine Aufgaben - beschädigte Testfälle**

Rufen Sie alle beschädigten Testfälle auf. Weitere Informationen finden Sie unter [Meine Aufgaben - Tester-Arbeitsvorrat \[Seite 260\]](#).

### **Testausführung - Protokoll**

Zeigt die Ausführungsprotokolle automatisierter Testfälle an.

Weitere Informationen finden Sie unter [Ausführungen \[Seite 274\]](#).

### **Testausführung - Jobs**

Hier zeigen Sie Informationen zu den Hintergrundjobs an, mit denen Sie die Ausführung automatischer Testfälle eingeplant haben:

- *Jobstatus*: Hat die Ausführung stattgefunden?
- *Test-Status*: Welches Ergebnis hatte die Ausführung?
- *Log-ID*: ID des Ausführungsprotokolls für den Job, dem ein Testpaket zugeordnet ist.
- *Jobname*: Link zu Ausführungsdetails des Jobs

Weitere Informationen finden Sie unter [Ausführungen \[Seite 274\]](#).

### **Test-Repository - Testkonfigurationen**

Das Test-Repository ermöglicht Ihnen den Zugriff auf die Umgebung für die Testzusammenstellung, in der Sie automatisierte Testfälle anlegen und bearbeiten können. Weitere Informationen finden Sie unter [Test-Repository \[Seite 216\]](#).

### **Test-Repository - Testskripte**

Das Test-Repository ermöglicht Ihnen den Zugriff auf einzelne Testskripte, die Bestandteil der Testkonfigurationen sind. Weitere Informationen finden Sie unter [Test-Repository \[Seite 216\]](#).

### **Test-Suite - Analysen**

Verschiedene Reports zum Testmanagement. Weitere Informationen finden Sie unter [Analysen \[Seite 275\]](#).

### **Personalisierung – Test-Suite**

Rufen Sie die Personalisierungsanwendung auf, um Ihre bevorzugten Einstellungen zu definieren. Weitere Informationen finden Sie unter [Personalisierung \[Seite 272\]](#).

### **Konfiguration – Test-Suite**

Rufen Sie das Setup von Solution Manager für Test-Suite-Szenarios auf. Weitere Informationen finden Sie unter [Konfiguration der Test-Suite \[Seite 121\]](#).

### **Verwaltung - Änderungseinfluss-Analyse**

Grundeinstellungen für den Business Process Change Analyzer (BPCA). Weitere Informationen finden Sie unter [Administration \[Seite 123\]](#).

### **Verwaltung - Test-Suite**

Grundeinstellungen für die Test-Suite. Weitere Informationen finden Sie unter [Administration \[Seite 123\]](#).

## 3.2 Testautomatisierung mit dem Testautomatisierungs-Framework

### Verwendung

Das Testautomatisierungs-Framework ist in die SAP-Solution-Manager-Systemlandschaft integriert und stellt u.a. Funktionen bereit, mit denen Sie Testkonfigurationen und Testskripte für automatisierte Tests, Testdatencontainer und Systemdaten-Container anlegen können.

### Funktionsumfang

- Testvorbereitung
  - Richten Sie in der Anwendung *Test-Suite - Testvorbereitung* automatisierte Tests ein. Weitere Informationen finden Sie unter [Lösungsdokumentation: Anlegen von Testfällen \[Seite 232\]](#).
  - Wenn Sie die Landschaft Ihrer Lösung definieren, wird der entsprechende Systemdatencontainer automatisch angelegt oder mit der relevanten logischen Komponentengruppe aktualisiert.
  - Auf der Benutzungsoberfläche der SUT-Verwaltung (SUT = System unter Test) können Sie die Informationen zum Systemdaten-Container (SDC) erweitern, die automatisch aus der Lösungsdokumentation abgerufen werden. Beim Ausführen von Testfällen werden die RFC-Destinationen abhängig von logischer Komponente und Systemrolle automatisch abgeleitet.
  - Beim Anlegen eines Testdatencontainers legen Sie Parameter an. Das Testautomatisierungs-Framework unterstützt Plausibilitätsprüfungen für Parameter auch dann, wenn die entsprechenden Datenelemente nicht in SAP Solution Manager, sondern nur im Zielsystem verwendet werden.
  - Bei der Bearbeitung des Contents eines eCATT-Testskripts wird die Schnittstelle der Transaktionen oder Funktionsbausteine automatisch aus der SAP-Solution-Manager-Systemlandschaft importiert und angezeigt, sofern dem Testfall ein Systemdaten-Container zugeordnet ist.
- Testplanverwaltung:  
Ordnen Sie dem Testplan die Systemrolle zu. Die RFC-Destinationen werden bei der Ausführung der Testfälle automatisch abgeleitet.

### Weitere Informationen

Weitere Informationen finden Sie in der Hilfe zur Anwendung *eCATT: Extended Computer Aided Test Tool* im SAP Help Portal unter <http://help.sap.com>.

## 3.3 Partner-Testmanagement

### Verwendung

Das Partner-Testmanagement integriert den SAP Solution Manager und die Testmanagement-Anwendung des Partners. Dies gestattet die Übertragung von Business Blueprints, zugehörigen Dokumenten, Spezifikationen, geschäftlichen Anforderungen, Links und Testobjekten, wie z.B. Transaktionen und kundenspezifischen Programmen sowie die Planung einer effizienten Testphase.

Diese Funktion steht Ihnen nicht standardmäßig zur Verfügung. Um diese Funktion nutzen zu können, müssen Sie das Partner-Testmanagement für SAP Solution Manager erwerben und die Business Function (ST\_QCA) explizit aktivieren. Weitere Informationen finden Sie unter [2433483](#).

Sie können die Anwendung *Partner-Testmanagement* auf dem SAP Solution Manager Launchpad öffnen.

### Funktionsumfang

Im Abschnitt *Branch-Assoziation* haben Sie folgende Möglichkeiten:

- Assoziationen anlegen, ändern und löschen: Assoziationen beziehen sich auf eine Eins-zu-eins-Zuordnung eines Partnerprojekts zu einer Kombination aus Branch und mindestens einem Umfang. Beachten Sie, dass die Zuordnung eines einzelnen Partnerprojekts zu mehreren SAP-Lösungen innerhalb eines SAP-Solution-Manager-Systems nicht möglich ist. Beim Anlegen einer Assoziation können Sie den Umfang der Assoziation bearbeiten. Sie können eine vorhandene Assoziation löschen. Diese ist dann frei für die Verknüpfung mit einem anderen Branch.
- *Lösungsdokumentation senden*: Sie können die Lösungsdokumentation aus dem SAP-Solution-Manager-System an das Partner-Testmanagementsystem senden. Es werden nur Elemente vom Typ „Ausführbare Einheiten“, „Dokumente“ und „Prozessvarianten“ übertragen. Anschließend kann der Qualitätsmanager testbezogene Aktivitäten ausführen und beispielsweise Testfälle anlegen und den Testfällen Fehler zuordnen.
- *Testergebnisse anzeigen*: Nach dem Senden der Lösungsdokumentation an das Partnersystem können Sie die Testergebnisse betrachten, indem Sie *Testergebnisse anzeigen* wählen. Diese umfassen u.a. folgende Details:
  - Den Status der Testfälle. Bei Vorliegen eines Fehlers können Sie z.B. die Anzahl der mit dem Testfall verknüpften nicht behobenen Fehler (im SAP-System) anzeigen.
  - Den Zeitpunkt, zu dem das Lösungsdokument zuletzt synchronisiert wurde, und den Benutzer, der die Synchronisierung vorgenommen hat.
  - Wenn Sie das Ankreuzfeld *Knoten mit Testergebnissen anzeigen* auswählen, können Sie nur Knoten anzeigen, die Testergebnisse enthalten.
  - Sie können den Geschäftsprozess, für den Sie die Testergebnisse anzeigen möchten, über die Filteroption in *Geschäftsprozess* suchen.
- *Lösungsdokumentation aktualisieren*: Wenn Sie das Ankreuzfeld markieren, werden alle an der Lösungsdokumentation vorgenommenen Änderungen automatisch im Partner-Testmanagementsystem aktualisiert. Ein täglich ausgeführter Hintergrundjob stellt sicher, dass die aktualisierten Informationen im Partnersystem verfügbar sind.

- **Testergebnisse aktualisieren:** Wenn Sie das Ankreuzfeld markieren, werden die aktualisierten Testergebnisse aus dem Partner-Testmanagementsystem an den SAP Solution Manager übertragen. Ein stündlich ausgeführter Hintergrundjob stellt sicher, dass die aktualisierten Testergebnisse im SAP-Solution-Manager-System verfügbar sind.
- **Sendestatus:** Zeigt den Status der Lösungsdokumentation an. Der Status wird angezeigt, wenn die Lösungsdokumentation aus dem SAP Solution Manager an das Partnersystem gesendet wird und wenn sie durch das Partnersystem aus dem SAP Solution Manager abgerufen wird.

In den Abschnitten unter *Details* haben Sie folgende Möglichkeiten:

- Sie können die aus dem SAP Solution Manager an das Partnersystem übertragenen Dokumentarten und Typen von ausführbaren Einheiten filtern.
- **Stichwortmapping:** Sie können kundeneigene Attribute im SAP Solution Manager unter Verwendung der benutzerdefinierten Felder des Partnersystems zuordnen.
- **Protokolle:** Sie können Details wie den Zeitpunkt der Übertragung der Lösungsdokumentation aus dem SAP Solution Manager an das Partnersystem und Details zu dem Benutzer, der diese Aktion ausgeführt hat, anzeigen.

## 3.3.1 Konfiguration des Partner-Testmanagements

### Verwendung

Um die Test-Suite in Verbindung mit Partnerwerkzeugen nutzen zu können, müssen Sie dies im SAP Solution Manager entsprechend einrichten.

### Prozess

Wählen Sie in der *SAP-Solution-Manager-Konfiguration* (Transaktion `SOLMAN_SETUP`) den Pfad ► *Test-Suite* ► *SAP Quality Center/SAP TAO* ► *Partner-Testmanagement* ▾.

Folgen Sie der Guided Procedure. Weitere Informationen finden Sie in den Hilfetexten zu den einzelnen Schritten und Aktivitäten.

## 3.4 Infrastruktur und Testwerkzeuge

In den folgenden Abschnitten werden die verschiedenen Werkzeuge und Benutzerschnittstellen beschrieben, die für die Testverwaltung mit SAP Solution Manager 7.2 verwendet werden:

- [Test-Suite in SAP Solution Manager \[Seite 11\]](#)
- [SUT-Verwaltung \[Seite 17\]](#)
- [Testkonfigurationen im Testautomatisierungs-Framework anlegen \[Seite 17\]](#)
- [Testautomatisierung mit eCATT \[Seite 14\]](#)



- Komponentenbasierte Testautomatisierung ([Basisinformationen \[Seite 26\]](#))

## 3.4.1 SUT-Verwaltung

### Verwendung

Mithilfe der Verwaltung von Systemen unter Test (SUT-Verwaltung) können Sie Systemdatencontainer (SDC) für die Systeme verwalten, die für Tests zur Verfügung stehen. Für eCATT- und Drittanbieterwerkzeuge können Sie die RFC-Destination und für die komponentenbasierte Testautomatisierung zusätzlich die Geschäftsbenutzer angeben.

**Die SUT-Verwaltung ist ein obligatorischer und kritischer Schritt für die Testautomatisierung**, da die Systeme unter Test aus den hier vorgenommenen Einstellungen abgeleitet werden.

### Aktivitäten

Sie können die SUT-Verwaltung wie folgt abrufen:

- Wählen Sie im SAP Solution Manager Launchpad den Pfad ► [Verwaltung - Test-Suite](#) ► [Testautomatisierungs-Framework](#) ► [Verwaltung des Systems unter Test](#) ►.
- Alternativ wählen Sie in der Umgebung für die Testzusammenstellung ► [Springen](#) ► [SUT-Verwaltung](#) ► aus der Dropdown-Liste.

Weitere Informationen zur SUT-Verwaltung in der komponentenbasierten Testautomatisierung finden Sie unter [Konfigurieren eines SUT mit Testprofil \[Seite 47\]](#)

Für eCATT- und Drittanbietertestwerkzeuge:

1. Importieren Sie den Systemdaten-Container.
2. Wählen Sie den Systemdatencontainer, die logische Komponentengruppe, die logische Komponente und die Systemrolle.
3. Wählen Sie die Registerkarte [eCATT- und Drittanbieter-Tools](#) auf der Registerkarte [SAP-ABAP-Backend](#).
4. Legen Sie unter [Testprofil](#) die RFC-Destination an, die von Tests verwendet werden soll.

## 3.4.2 Testkonfiguration im Testautomatisierungs-Framework

**Wir empfehlen Ihnen, die Testkonfigurationen direkt in Ihrer Lösungsdokumentation anzulegen, damit der SAP-Solution-Manager-Kontext ordnungsgemäß für den Abruf der relevanten Systeme unter Test definiert ist.**

Alternativ können Sie Ihre Testkonfigurationen auch in der Anwendung [Test-Repository - Testkonfigurationen](#) anlegen. Von dort aus können Sie bei Bedarf zum Testautomatisierungs-Framework für ein ausgewähltes Objekt navigieren, um beispielsweise den Content für ein eCATT-Skript zu definieren. Mit dem

Testautomatisierungs-Framework können Sie die folgenden Komponenten automatisierter Testfälle (Testkonfigurationen) verwalten:

- Testskripte
- Systemdatencontainer
- Testdatencontainer

Weitere Informationen finden Sie unter [Komponenten automatisierter Testfälle \[Seite 18\]](#).

Mit dem Testautomatisierungs-Framework können Sie automatisierte Testfälle erstellen, indem Sie eine Testkonfiguration und das zugehörige Testskript in einem einzigen Schritt erstellen.

Sie können entweder Testfälle mit der komponentenbasierten Testautomatisierung (CBTA), mit zusammengesetzten Tests oder eCATT-Testfälle anlegen. Alternativ können Sie zum Anlegen von Testfällen auch externe Werkzeuge verwenden.

Sie können Testfälle bearbeiten, kopieren, umbenennen oder löschen und sie ausführen und Ausführungsprotokolle anzeigen.

Das Testautomatisierungs-Framework ist in die Systemlandschaft von SAP Solution Manager integriert. Das heißt, dass Sie keine RFC-Destinationen manuell anlegen müssen:

- Systemdaten-Container werden automatisch auf Basis der Lösungslandschaft angelegt, und RFC-Destinationen werden automatisch je nach logischer Komponentengruppe des Testfalls abgerufen.
- Beim Anlegen von Systemdatencontainern ordnen Sie logische Komponenten zu, während die RFC-Destinationen automatisch abgerufen werden.
- Wenn die Systemrolle eines Testplans für alle im Testplan enthaltenen Testfälle geändert wird, werden die RFC-Destinationen bei Ausführung der Testfälle automatisch abgeleitet.
- Bei der Bearbeitung eines eCATT-Testskripts wird die Schnittstelle der Transaktionen oder Funktionsbausteine automatisch aus der SAP-Solution-Manager-Systemlandschaft importiert und angezeigt, sofern dem Testfall ein Systemdaten-Container zugeordnet ist.

### 3.4.2.1 Komponenten automatisierter Testfälle

Die getesteten Systeme ändern sich im Allgemeinen im Laufe der Zeit. Beispielsweise beginnen Sie damit, ein bestimmtes Release eines getesteten Systems zu testen. Wenn Sie Ihre Testfälle entwickelt haben, möchten Sie diese ggf. erneut zum Testen späterer Releases des getesteten Systems verwenden, oder Sie möchten einen Testfall in dem Testsystem, das Sie im Entwicklungssystem erstellt haben, erneut verwenden.

Durch Nutzung der Informationen, die SAP Solution Manager über die Systemlandschaft bereitstellt, können Sie die getesteten Systeme aktualisieren oder sogar austauschen, ohne Ihre Testobjekte zu ändern.

Deshalb handelt es sich bei einem **Testfall** (entweder eCATT oder mit einem Drittanbieterwerkzeug angelegt) tatsächlich um eine **Testkonfiguration** wiederverwendbarer Objekte. Eine Testkonfiguration umfasst folgende Objekttypen:

- Das **Testskript** stellt die Schritte eines Geschäftsprozesses dar, der durch den automatisierten Testfall ausgeführt wird, z. B.:
  1. Im CRM-System wird eine Bestellung eingegeben.
  2. Sie wird an das ERP-System übergeben, wo die Produktion eingeplant wird.
  3. Das ERP-System übergibt Informationen an das SCM-System, in dem eine Tabelle aktualisiert wird.

- Die **logische Komponentengruppe** definiert die Systeme, in denen diese Schritte durchgeführt werden. Ein System wird folgendermaßen definiert:
  - Operative Funktion (z. B. CRM-, ERP-System)
  - Technische Rolle (Systemrollen wie z. B. Entwicklungs-, Test-, Evaluierungs-, Produktivsystem)
 Die Systeme werden durch logische Komponenten dargestellt. Relevante Parameter der jeweiligen Systemlandschaft (Zielsysteme und RFC-Destinationen) werden automatisch aus der Lösung, die der Testkonfiguration zugeordnet ist, abgerufen. Wenn Sie einen Testfall ausführen, können Sie den Branch und die Systemrolle wechseln (z. B. vom Entwicklungs- zum Testsystem), ohne Systemparameter anpassen zu müssen.
- Der **Testdatencontainer** ist ein wiederverwendbares Datenobjekt mit einem Satz benutzerdefinierter Parameter, die unabhängig vom Testskript genutzt werden können. Weitere Informationen finden Sie in der Anwendungshilfe *eCATT: Extended Computer Aided Test Tool* im SAP Help Portal unter <http://help.sap.com> unter *Testdaten organisieren*.

## 3.4.2.2 Systemlandschaft

### Kontext

Ein Testfall wird einer Lösung der [Lösungsdokumentation](#) zugeordnet. Die Lösung beschreibt die Systemlandschaft, in der der Test aufgezeichnet und ausgeführt wird. Die logische Komponentengruppe der Lösung besteht aus logischen Komponenten. Sie enthält die technischen Systeme und teilt die logischen Komponenten in Entwicklungs-, Produktiv- oder Testsysteme ein.

In einer Lösung wird die logische Komponente einer logischen Komponentengruppe vom Branch der Lösung abgeleitet. Der Branch gibt die logische Komponente an, und die Systemrolle bestimmt, welches System aus der logischen Komponente ausgewählt wird.

## 3.4.2.3 Anlegen automatisierter Testfälle

### Kontext

Ein automatisierter Testfall ist eine Testkonfiguration, die aus einem Testskript, einem Systemdaten-Container und einem Testdatencontainer besteht. Um einen automatisierten Testfall anzulegen, legen Sie eine Testkonfiguration an. Damit wird gleichzeitig ein zugehöriges Testskript angelegt, Sie müssen dieses nicht separat anlegen. Im nächsten Schritt kann das Testskript bearbeitet werden.

Der Systemdaten-Container wird automatisch aus der Lösung abgerufen, für die die Testkonfiguration angelegt wurde.

Definieren Sie einen vollständigen SAP-Solution-Manager-Kontext, und ordnen Sie diesen Ihrer Testkonfiguration oder Ihrem Testskript zu.

Beim Bearbeiten eines Testskripts können Sie sehen, welche logische Komponentengruppe und welcher SAP-Solution-Manager-Kontext (Lösung, Branch und Systemrolle) dem Testfall zugeordnet sind.

Wenn für eine Lösung und einen Branch eine Site aktiviert wurde, ist dies für den Kontext ebenfalls relevant. Die Site wird nur in automatisierten Tests verwendet, wenn der Test beispielsweise aus der Lösungsdokumentation heraus ausgeführt wird, jedoch nicht in die Testplanverwaltung integriert ist. Weitere Informationen finden Sie unter [Sites](#).

Weitere Informationen finden Sie unter [Komponenten automatisierter Testfälle \[Seite 18\]](#).

## Vorgehensweise

Zum Anlegen eines automatischen Testfalls haben Sie folgende Möglichkeiten:

- [Anlegen von Testkonfigurationen und Testskripten \[Seite 217\]](#)
- [Anlegen von Testkonfigurationen \[Seite 20\]](#)

### 3.4.2.4 Anlegen von Testkonfigurationen

## Vorgehensweise

1. Gehen Sie wie folgt vor, um im SAP Solution Manager Launchpad eine Testkonfiguration anzulegen:
2. Wechseln Sie zur *Lösungsdokumentation*, wählen Sie eine Lösung, und navigieren Sie zum Geschäftsprozess. Klicken Sie mit der rechten Maustaste, und wählen Sie ► *Neu* ► *Testkonfiguration* ► *Anlegen* ►. Dies ist die einfachste Möglichkeit, um einen Test direkt im richtigen Kontext anzulegen.

Alternativ können Sie auch ► *Test-Repository - Testkonfigurationen* ► wählen und einen neuen Test anlegen.

#### **i** Hinweis

Legen Sie keine Testkonfigurationen über die Transaktion `secatt` an.

3. Geben Sie die erforderlichen Daten ein:
  - Wenn Sie die Testkonfiguration in der Anwendung *Test-Repository - Testkonfiguration* anlegen, wählen Sie eine Lösung aus.
  - Geben Sie einen Namen für die Testkonfiguration ein.
  - Wählen Sie ein Testwerkzeug aus.
  - Geben Sie einen Namen für das Testskript sowie die Version, den Titel und das Paket ein.
4. Wählen Sie zur Bestätigung *OK*.

Wechseln Sie zum Editor für Testkonfigurationen.

5. Geben Sie die erforderlichen Daten ein:
  - Ordnen Sie eine ausführbare Einheit zu. Sie können nach verfügbaren ausführbaren Einheiten suchen. Die Liste zeigt die dem Knoten in der Lösungsdokumentation zugeordneten ausführbaren Einheiten an, wenn die Testkonfiguration aus der Lösungsdokumentation heraus angelegt wird.
  - Die Anwendungskomponente, sofern verfügbar, wird aus der ausgewählten ausführbaren Einheit abgeleitet. Wählen Sie andernfalls die Anwendungskomponente aus, die der Art des zu testenden Geschäftsprozesses entspricht.
  - Wählen Sie einen Verantwortlichen aus.
  - Die logische Komponentengruppe wird aus der ausgewählten ausführbaren Einheit abgeleitet.
  - Die technischen Daten (Systemdaten-Container, Zielkomponente und Zielsystem) werden automatisch auf Basis des SAP-Solution-Manager-Kontexts und der logischen Komponentengruppe ausgewertet.
6. Geben Sie ein Testprofil ein, das in der Verwaltung von Systemen unter Test (SUT-Verwaltung) definiert wurde.
7. Um eine eCATT-Testkonfiguration zu bearbeiten, wählen Sie [eCATT starten](#).
8. Um das Drittanbieterwerkzeug zu bearbeiten, wählen Sie [Bearbeiten](#).
9. Optional: Wählen Sie auf der Registerkarte [Testdaten](#) einen Testdatencontainer aus. Um Varianten zu pflegen, wählen Sie die Registerkarte [Varianten](#).

Sie können Varianten definieren, bei denen es sich um Sätze unterschiedlicher Parameterwerte handelt. ECATTDEFAULT ist die Standardvariante. Die Werte dieser Variante werden auf der Registerkarte [Parameter](#) festgelegt. Wenn in einer Variante ein Parameter keinen Wert aufweist, wird der entsprechende Wert aus ECATTDEFAULT verwendet.

Weitere Informationen finden Sie in der Hilfe zur Anwendung [eCATT: extended Computer Aided Test Tool](#) unter <http://help.sap.com> unter [▶ Testdaten organisieren](#) [▶ Varianten in Testdatencontainern anlegen](#) [▶](#).

10. Sichern Sie Ihre Eingaben.

Der Testfall wird angelegt. Die Testkonfiguration und das Testskript werden gesichert.

## Ergebnisse

Der automatisierte Testfall wird angelegt und kann bei der Testplanung und Testausführung verwendet werden.

### 3.4.2.5 Verwalten von Komponenten automatisierter Testfälle

## Kontext

Sie können folgende Komponenten von Testkonfigurationen bearbeiten, kopieren, umbenennen oder löschen:

- Testskripte (siehe [Bearbeiten von Testskripten \[Seite 223\]](#))
- Testdatencontainer
- Systemdatencontainer

In SAP Solution Manager wird zum Bearbeiten von Testskripten und Testkonfigurationen die Verwendung der [Test-Repository](#)-Sichten empfohlen. Nur wenn dort keine Änderungen möglich sind, sollten Sie die Schnittstelle für die erweiterte Testautomatisierung in der Transaktion `eCATT` verwenden.

Systemdaten-Container (SDCs) werden automatisch angelegt, wenn Sie die Lösungslandschaft ändern. SDCs werden nur verwendet, um die Kommunikation mit der `eCATT`-Schicht zu ermöglichen. Sie legen die Verbindungseinstellungen über die [SUT-Verwaltung \[Seite 17\]](#) fest. Das Aktualisieren von SDCs in der erweiterten Testautomatisierung ist nicht mehr relevant.

## Vorgehensweise

1. Wählen Sie in der Transaktion `SECATT` die Testkonfiguration, das Testskript, den Testdatencontainer oder den Systemdatencontainer aus, die/das/den Sie pflegen möchten.
2. Sie haben folgende Möglichkeiten:
  - Wählen Sie [Anzeigen](#) oder [Bearbeiten](#), um den Editor zu öffnen.  
Der entsprechende Editor wird geöffnet.
  - Wählen Sie [Objekt löschen](#).  
Sie gelangen auf ein Dialogfenster.
    - Wenn Sie eine Testkonfiguration ausgewählt haben, können Sie wählen, ob Sie nur die Testkonfiguration selbst, die Testkonfiguration mit der aktuellen Version oder die Testkonfiguration mit allen Versionen des Testskripts auswählen möchten. Wenn Sie nur die Testkonfiguration selbst löschen, können Sie das entsprechende Testskript beim Anlegen einer neuen Testkonfiguration wiederverwenden.
    - Wenn Sie einen Test- oder Systemdatencontainer ausgewählt haben, der einer Testkonfiguration zugeordnet ist, müssen Sie zuerst die Zuordnung entfernen.  
Wählen Sie dazu den [Verwendungsnachweis](#), um die entsprechenden Objekte anzuzeigen und zu bearbeiten.  
Wählen Sie [Ja](#).
  - Um ein Objekt zu kopieren oder umzubenennen, gehen Sie folgendermaßen vor:
    1. Wählen Sie [Objekt kopieren](#) oder [Objekt umbenennen](#).  
Sie gelangen auf ein Dialogfenster.
    2. Geben Sie den Namen des Objekts ein, das kopiert oder umbenannt werden soll.
    3. Um die Testkonfiguration zu kopieren oder umzubenennen, ohne das entsprechende Testskript zu kopieren oder umzubenennen, entmarkieren Sie das Ankreuzfeld [Einschl. Testskript](#).  
Wenn Sie einen Testfall kopieren, wird das Testskript des Quelltestfalls dem Zieltestfall zugeordnet. Wenn Sie einen Testfall umbenennen, wird das Testskript des Quelltestfalls dem Zieltestfall zugeordnet, ohne umbenannt zu werden.
    4. Wählen Sie [Kopieren](#) oder [Umbenennen](#).

3. Sichern Sie Ihre Eingaben.
4. Beenden Sie die Funktion.

## 3.4.2.6 Anlegen von Testdatencontainern

### Verwendung

Beim Anlegen von Testdatencontainern können Sie das Zielsystem angeben und auf Datenelemente verweisen, die im Zielsystem vorhanden sind, aber nicht unbedingt in Ihrem lokalen System. Dies ermöglicht automatisierte Plausibilitätsprüfungen, mit denen das Risiko fehlerhafter Testdatencontainer reduziert wird.

### Vorgehensweise

Sie legen einen Testdatencontainer für eCATT und Drittanbieterwerkzeuge direkt aus dem Testautomatisierungs-Framework an.

1. Starten Sie die Transaktion `SECATT`.
  2. Geben Sie im Feld *Testdatencontainer* einen Namen ein. Verwenden Sie beim Eingeben des Namens keine Leerzeichen.
  3. Wählen Sie *Anlegen*.  
Geben Sie auf dem Bild *Testdatencontainer anlegen* die erforderlichen Daten ein.
  4. Definieren Sie den SAP-Solution-Manager-Kontext:
    1. Wählen Sie **Hilfsmittel** **Einstellungen**.
    2. Wählen Sie eine Lösung, einen Branch und (sofern in Ihrem Kontext relevant) eine Site aus.
    3. Wählen Sie eine Systemrolle aus.
- i Hinweis**

Die Systemrolle ist benutzerspezifisch. Ein anderer Benutzer kann denselben Systemdatencontainer mit einer anderen Systemrolle verwenden, ohne die RFC-Destination manuell ändern zu müssen.
4. Wählen Sie *Sichern und beenden*.  
Beachten Sie, dass Sie die Standard-Workbench-Einstellungen über das Dialogfenster *SAP-Solution-Manager-Kontext* aufrufen können.
  5. Wählen Sie auf der Registerkarte "Allgemeine Daten" einen Systemdaten-Container und ein Zielsystem aus.  
Das ausgewählte Zielsystem wird im oberen Bildbereich angezeigt.
  6. Legen Sie auf der Registerkarte *Parameter* einen Parameter an.
  7. Wählen Sie die *Eingabehilfe* für die Spalte *Parameterreferenz*.
  8. Wählen Sie *Suche nach Datenelementen*.  
Die Datenelemente, die im Zielsystem vorhanden sind, werden angezeigt.
  9. Wählen Sie ein Datenelement aus.
  10. Geben Sie einen Wert ein.

11. Sichern Sie Ihre Eingaben.

Um sicherzustellen, dass die Werte mit den Angaben für das Datenelement im Zielsystem übereinstimmen, wird eine Plausibilitätsprüfung durchgeführt.

## 3.4.2.7 Systemrolle eines Testplans ändern

### Voraussetzungen

Sie haben die Berechtigung eines Testorganisors.

Der Freigabestatuswert des Testplans lässt Änderungen zu.

### Kontext

In der Test-Suite des SAP Solution Manager können Sie die Systemrolle eines Testplans ändern, ohne den Systemdaten-Container bearbeiten und die RFC-Destinationen manuell ändern zu müssen.

Die Systemrolle wird verwendet, um je nach SAP-Solution-Manager-Kontext (Lösung, Branch und logische Komponentengruppe) das richtige System unter Test für die Ausführung der Testfälle anzusprechen. Die Systemrolle ist das einzige Attribut des Kontexts, das nach dem Anlegen des Testplans geändert werden kann. Eine Änderung nach dem Start der Testausführung hat keine Auswirkungen auf bereits vorhandene Ergebnisse. Bei der erneuten Ausführung automatischer Tests wird die Änderung nur zur Auswertung der RFC-Destination berücksichtigt.

### Vorgehensweise

1. Wählen Sie im SAP Solution Manager Launchpad die Option *Testplanverwaltung*.
2. Wählen Sie einen Testplan aus.
3. Wählen Sie unter **► Allgemeine Daten ► Systemrollen-ID** eine Systemrolle aus.
4. Sichern Sie Ihre Angaben, und schließen Sie das Fenster.

### Ergebnisse

Die Systemrolle wurde dem Testplan zugeordnet. Die RFC-Destination wird bei der Ausführung der Testfälle automatisch abgeleitet.



Änderungen an der Systemrolle nach Beginn der Testausführung werden erst bei der nächsten Testausführung wirksam.

### 3.4.3 Testautomatisierung mit dem Testautomatisierungs-Framework

#### Verwendung

Das Testautomatisierungs-Framework ist in die SAP-Solution-Manager-Systemlandschaft integriert und stellt u.a. Funktionen bereit, mit denen Sie Testkonfigurationen und Testskripte für automatisierte Tests, Testdatencontainer und Systemdaten-Container anlegen können.

#### Funktionsumfang

- Testvorbereitung
  - Richten Sie in der Anwendung *Test-Suite - Testvorbereitung* automatisierte Tests ein. Weitere Informationen finden Sie unter [Lösungsdokumentation: Anlegen von Testfällen \[Seite 232\]](#).
  - Wenn Sie die Landschaft Ihrer Lösung definieren, wird der entsprechende Systemdatencontainer automatisch angelegt oder mit der relevanten logischen Komponentengruppe aktualisiert.
  - Auf der Benutzungsoberfläche der SUT-Verwaltung (SUT = System unter Test) können Sie die Informationen zum Systemdaten-Container (SDC) erweitern, die automatisch aus der Lösungsdokumentation abgerufen werden. Beim Ausführen von Testfällen werden die RFC-Destinationen abhängig von logischer Komponente und Systemrolle automatisch abgeleitet.
  - Beim Anlegen eines Testdatencontainers legen Sie Parameter an. Das Testautomatisierungs-Framework unterstützt Plausibilitätsprüfungen für Parameter auch dann, wenn die entsprechenden Datenelemente nicht in SAP Solution Manager, sondern nur im Zielsystem verwendet werden.
  - Bei der Bearbeitung des Contents eines eCATT-Testskripts wird die Schnittstelle der Transaktionen oder Funktionsbausteine automatisch aus der SAP-Solution-Manager-Systemlandschaft importiert und angezeigt, sofern dem Testfall ein Systemdaten-Container zugeordnet ist.
- Testplanverwaltung:

Ordnen Sie dem Testplan die Systemrolle zu. Die RFC-Destinationen werden bei der Ausführung der Testfälle automatisch abgeleitet.

#### Weitere Informationen

Weitere Informationen finden Sie in der Hilfe zur Anwendung *eCATT: Extended Computer Aided Test Tool* im SAP Help Portal unter <http://help.sap.com>.

## 3.4.4 Komponentenbasierte Testautomatisierung

### 3.4.4.1 Basisinformationen

Sie können in SAP Solution Manager zum Anlegen automatisierter Tests die komponentenbasierte Testautomatisierung verwenden. Ein automatischer Test setzt sich aus Testskripten zusammen, die Komponenten enthalten. Es gibt zwei Arten von Komponenten:

- **Standardkomponenten** sind für grundlegende Aktionen wie die Auswahl einer Drucktaste oder einer Registerkarte zuständig. Sie werden im SAP-Solution-Manager-System bereitgestellt.
- **Bildkomponenten** werden entweder für SAP-GUI-Transaktionsbilder oder SAP-CRM-Web-UI-Sichten generiert. Web-Dynpro-ABAP-Anwendungssichten werden unterstützt, jedoch nur im nativen Modus und nicht, wenn sie im Portal eingebettet sind. Sie werden mit einem Parameter für jedes Feld auf dem Bild angelegt. Der Wert des Parameters wird während der Ausführung an das entsprechende Feld übergeben. Zum Ende einer Testaufzeichnung werden automatisch Bildkomponenten generiert. Pro SAP-GUI-Transaktionsbild, SAP-CRM-Web-UI-Anwendungssicht und SAP-Web-Dynpro-Anwendung liegt immer nur eine Bildkomponente vor. Und für jede logische Komponente ist nur eine Bildkomponente vorhanden, um sicherzustellen, dass die getestete Anwendung eine dedizierte Version aufweist. Eine Bildkomponente kann also gemeinsam von mehreren Tests verwendet werden.

Komponentenbasierte Testautomatisierung ist eng in SAP Solution Manager und seine Lösungsverwaltungs- und Lösungsdokumentationskonzepte eingebunden. SAP Solution Manager stellt verschiedene Werkzeuge und Methoden zur Implementierung und Pflege von SAP- und Kundenlösungen bereit. Das Lösungsdokumentations-Modell basiert auf einer hierarchischen Struktur, die eine Lösung als einzigen, zentralen Zugriffspunkt für den gesamten Inhalt der Lösungsdokumentation verwendet.

- Sie verwalten Ihre Lösungen im Tool *Lösungsverwaltung* (Transaktionscode `SOLADM`) in Ihrem SAP-Solution-Manager-System.
- Sie verwalten Ihre Lösungsdokumentation entweder im Werkzeug *Lösungsdokumentation* (Transaktion `SOLDOC`) oder in der App *Lösungsdokumentation* im SAP Solution Manager Launchpad.

Verwenden Sie zum Anlegen und Pflegen von Testkonfigurationen die App *Lösungsdokumentation*. Die mit der komponentenbasierten Testautomatisierung angelegten Testskripte sind mit dem eCATT-Framework kompatibel.

## Integration

- SAP Solution Manager: Speichern von Daten zum Erstellen, Optimieren und Bearbeiten von Tests und Komponenten
- Umgebung für die Testzusammenstellung: Anlegen von Testkonfigurationen und Testskripten, die mit der komponentenbasierten Testautomatisierung aufgezeichnet wurden. Weitere Informationen finden Sie unter [Test-Repository \[Seite 216\]](#).
- Testplanverwaltung: Einplanen der Ausführung von Tests.

## Funktionsumfang

- Testskript-Aufzeichnung (verbesserter SAP-S/4HANA-Support)
- Technologieübergreifende Aufzeichnung in einer Sitzung
- Aufzeichnung mehrerer SAP-GUI-Fenster
- Aufzeichnung der eingebetteten HTML in SAP-GUI-Transaktionen
- Wiederverwendung von CBTA-Skripten in einem anderen CBTA-Skript
- Einfache Pflege
- Verwalten von Testkonfigurationen mit gemeinsam verwendeten Komponenten
- Verwalten von Bildkomponenten und deren Parameter
- Verwenden des Testdatencontainers (TDC) zur Definition mehrerer Datensätze

### i Hinweis

Informationen zu Einschränkungen der komponentenbasierten Testautomatisierung erhalten Sie im SAP-Hinweis [1835958](#).

Tests für die komponentenbasierte Testautomatisierung sind eCATT-Tests und können somit die eCATT-Funktionen verwenden.

Die Objektstruktur wird folgendermaßen definiert:

- Die Lösungsdokumentation referenziert Testkonfigurationen zum Testen der ausgewählten ausführbaren Einheiten.
- In den Apps *Test-Repository - Testskripte* und *Test-Repository - Testkonfigurationen* sind alle vorhandenen Testkonfigurationen und Testskripte aufgelistet.
- Eine Testkonfiguration umfasst folgende Assets:
  - Einen Systemdatencontainer (SDC) zum Festlegen des Systems, auf dem die Testaufzeichnung und -ausführung durchgeführt werden. Einer Lösung ist nur ein SDC angehängt. In diesem Szenario wird der SDC automatisch mit demselben Namen wie die Lösung generiert und das Zeichen z vorangestellt.
  - Einen Testdatencontainer zur Definition verschiedener Datensätze, um die Eingabeparameter für die Tests bereitzustellen.
  - Ein Testskript, das Folgendes umfasst:
    - Das ausführbare Objekt, das Transaktion, Programm oder URL für den Test definiert. Dieses Objekt ist mit einer logischen Komponentengruppe verknüpft, wie in der zugehörigen Lösungslandschaft definiert.
    - Ein Testprofil, das festlegt, welcher Anwendungsbenutzer für Aufzeichnung und Ausführung verwendet wird. Das Testprofil wird im Tool *SUT-Verwaltung* definiert. Weitere Informationen finden Sie unter [SUT-Verwaltung \[Seite 17\]](#).  
Die Schritte der automatisierten Tests, beispielsweise Klicken auf eine Schaltfläche, Auswählen einer Registerkarte oder Eingeben von Daten in ein Eingabefeld.

### 3.4.4.1.1 Wichtige SAP-Hinweise

Bevor Sie die komponentenbasierte Testautomatisierung (CBTA) verwenden, sollten Sie die folgenden SAP-Hinweise lesen und implementieren:

SAP-Hinweis	Titel
<a href="#">2436142</a> 	Komponentenbasierte Testautomatisierung – Supportfähigkeit und Kompatibilitätsmatrix
<a href="#">1763697</a> 	Komponentenbasierte Testautomatisierung – Installationsinformationen
<a href="#">1778899</a> 	CBTA 3.0 – Releasedokumentation
<a href="#">2640485</a> 	CBTA 3.0: Release-Dokumentation SP10
<a href="#">1835958</a> 	Bekannte Probleme
<a href="#">2029868</a> 	CBTA: Korrekturen und Verbesserungen an der Laufzeitbibliothek
<a href="#">1666201</a> 	WebCUIF – Sammelhinweis zum Aktivieren des Testmodus für CRM-Web-UIs
<a href="#">2177107</a> 	CBTA - Testautomatisierung von Web-Dynpro-Szenarios: Stabile-ID-Modus nicht an untergeordnete Fenster propagiert

### 3.4.4.1.2 Unterstützte UI-Technologien

Die folgenden UI-Technologien werden in der komponentenbasierten Testautomatisierung (CBTA) unterstützt:

UI-Technologie	Beschreibung
SAP GUI	Von SAP-Transaktionen verwendete UI-Technologie In MS Internet Explorer angezeigter SAP-GUI-Content
Web-/Plain-HTML	Von Nicht-SAP-Anwendungen verwendete UI-Technologie
SAP-CRM-WebClient-UI-Framework (WEBCUIF)	Vorwiegend von SAP-CRM-Web-UIs verwendetes UI-Framework
SAP UI5 und SAP Fiori	Von SAP-UI5- und SAP-Fiori-Apps verwendetes UI-Framework

UI-Technologie	Beschreibung
Unified Rendering Light Speed (LS)	UI-Framework, das gemein für SAP-Anwendungen ist, wie z.B.: <ul style="list-style-type: none"> <li>• Web-Dynpro-Anwendung (ABAP)</li> <li>• Web-Dynpro-Anwendungen (Java)</li> <li>• SAP GUI für HTML</li> </ul>
Business Server Page (BSP)	Auf Business Server Pages basierende Anwendungen
Java Web Dynpro	Schicht, die zuvor von Java-Web-Dynpro-Anwendungen verwendet wurde




## Nicht unterstützte UI-Technologien

In der komponentenbasierten Testautomatisierung werden die folgenden Technologien nicht unterstützt:

- Adobe Flash
- Java Applets

### 3.4.4.1.3 Voraussetzungen für die Testautomatisierung

Beim automatisierten Testen muss die Testautomatisierungslogik die zu testende Anwendung u.U. in einem spezifischen Modus starten. Abhängig von der von der Anwendung verwendeten UI-Technologie sind unterschiedliche Einstellungen erforderlich. Die folgenden SAP-Hinweise beschreiben verschiedene technische Voraussetzungen, die Sie erfüllen müssen, um die Kompatibilität der komponentenbasierten Testautomatisierungsfunktionen mit den zu testenden Anwendungen sicherzustellen.

SAP-Hinweis	Titel
<a href="#">2436142</a> 	Zusatzinformationen: Supportfähigkeit bei der komponentenbasierten Testautomatisierung
<a href="#">1666201</a> 	WebCUIF – Sammelhinweis zum Aktivieren des Testmodus für CRM-Web-UIs
<a href="#">2177107</a> 	CBTA - Testautomatisierung von Web-Dynpro-Szenarios: Stabile-ID-Modus nicht an untergeordnete Fenster propagiert

## 3.4.4.1.4 Glossar

Es ist wichtig, sich mit den folgenden Kernkonzepten und der Terminologie vertraut zu machen.

Begriff	Definition
Testskript	<p>Ein Testskript ist eine Entität, die dauerhaft im Repository des SAP-Solution-Manager-Systems gespeichert wurde. CBTA-Testskripte sind Objekte, die eine Liste mit Schritten enthalten, um Benutzerinteraktionen mit der Anwendung zu simulieren. Jeder Schritt enthält die folgenden Informationen:</p> <ul style="list-style-type: none"><li>• Eine Referenz auf eine Komponente oder ein anderes CBTA-Testskript</li><li>• Import- und Exportparameter, d.h. die Werte, die zu oder von jedem Schritt übergeben werden</li><li>• Eine Zuordnung zu einer ausführbaren Einheit</li></ul> <p>Wenn Sie den Test durchführen, werden diese Informationen mit den SAP-Solution-Manager-Kontext kombiniert, um das Zielsystem zu ermitteln und wie die Anwendung gestartet wird.</p>
Umgebung für die Testzusammenstellung	<p>Die Umgebung für die Testzusammenstellung (TCE) ist einer der Haupteinstiegspunkte für Aktivitäten zur Testautomatisierung, unter anderem:</p> <ul style="list-style-type: none"><li>• Neue Testskripte anlegen</li><li>• CBTA-Testrecorder starten</li><li>• Vorhandene Testskripte ausführen</li><li>• Ausführungs-Reports prüfen</li><li>• Testskripte pflegen</li></ul>
System unter Test	<p>Das System unter Test (SUT) ist das System in Ihrer Landschaft, in dem Sie Tests aufzeichnen und ausführen. Einige Systeme sind geschäftskritisch und sollten aus Sicherheitsgründen nicht zum Ausführen von automatisierten Testskripten verwendet werden.</p>

Begriff	Definition
Testprofil	<p>Ein Testprofil ist eine benannte, wiederverwendbare Kombination aus Benutzername und Kennwort, mit dem Sie ein Testskript verknüpfen können. Das Testprofil wird verwendet, um den Benutzer automatisch bei der Ausführung des Testskripts zu authentifizieren. Sie definieren und pflegen Testprofile in der SUT-Verwaltung.</p> <p>Das SUT wird gemäß dem SAP-Solution-Manager-Kontext ermittelt. Daher ist es wichtig, Testprofile korrekt zu definieren und die entsprechenden Benutzeranmeldedaten für jede Kombination aus logischen Komponentengruppen, Branches und Systemrollen zu pflegen.</p>
Ausführbare Einheit	<p>Ein ausführbares Objekt stellt den Einstiegspunkt für eine Anwendung dar, z.B. eine Transaktion oder eine SAP-Fiori-App. Die von ihm bereitgestellten Informationen sind wichtig und werden verwendet, um zu ermitteln, wie die zu testende Anwendung gestartet wird.</p>
Komponente	<p>Eine Komponente ist die Entität, die zum Simulieren von Benutzeraktionen verwendet wird. Eine Standardkomponente bedeutet eine Komponente, die von SAP ausgeliefert wird.</p> <p>Zusätzliche Komponenten wie die Bildkomponenten und Sichtkomponenten werden dynamisch beim Aufzeichnen der Geschäftsszenarios generiert, die getestet werden.</p>
Testrecorder	<p>Sie legen CBTA-Testskripte an, indem Sie einen Geschäftsprozess aufzeichnen, während er ausgeführt wird. CBTA schließt das Testrecorder-Werkzeug ein, das die von der zu testenden Anwendung ausgelösten Ereignisse sammelt. Diese Informationen werden dann verwendet, um automatisch Testskripte zu generieren.</p> <p>Sie können von der Umgebung für die Testzusammenstellung aus auf den Testrecorder zugreifen.</p>

### 3.4.4.2 Technische Konzepte

Sie sollten sich mit den folgenden technischen Konzepte im Zusammenhang mit der komponentenbasierten Testautomatisierung vertraut machen.

### 3.4.4.2.1 SAP-Solution-Manager-Kontext

Der SAP-Solution-Manager-Kontext ist eine Abstraktion Ihrer Landschaft. Dieser Kontext besteht aus den folgenden Elementen:

- Einer Lösung und dem zugehörigen Systemdatencontainer (SDC)
- Einer Filiale
- Einer Site
- Einer Systemrolle

Das Zielsystem wird automatisch ermittelt, indem die Informationen aus dem Kontext und dem ausführbaren Objekt, das dem Testskript zugeordnet ist, kombiniert werden.

#### **i** Hinweis

Pflegen Sie die technische Infrastruktur, die für den Zugriff auf die Systeme in der SUT-Verwaltung erforderlich ist, in der Sie RFC-Destinationen angeben, Testprofile definieren usw..

### 3.4.4.2.2 Testskriptschritte

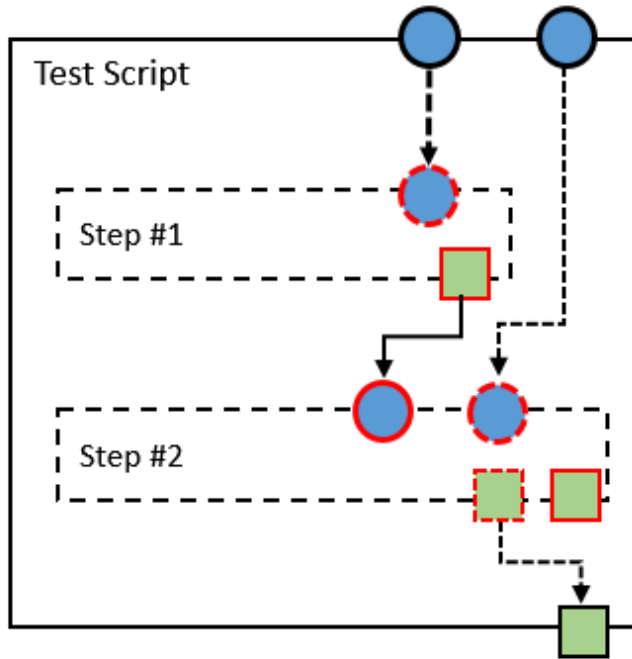
Die in der modellbasierten Testentwicklungsumgebung angezeigten Testskripte umfassen ungeachtet ihres Typs eine Reihe von Schritten, die jeweils Eingabe- und Ausgabeparameter beinhalten können. Eingabeparameter können Ausgabeparametern eines vorhergehenden Schritts zugeordnet werden, sodass sie z.B. Daten abrufen können, die in einem vorherigen Schritt generiert wurden. Schrittparameter können exponiert werden, damit Sie als Testskriptparameter gesehen werden. Testskriptparameter enthalten einen Standardwert, der durch Daten aus dem Testdatencontainer und Testdatenvarianten ersetzt werden kann.

Ein Schritt kann sich auf eine einzelne komponentenbasierte Testautomatisierungskomponente oder ein anderes komponentenbasiertes Testautomatisierungs-Testskript beziehen, das als Bestandteil eines zusammengesetzten Testskripts wiederverwendet wird.

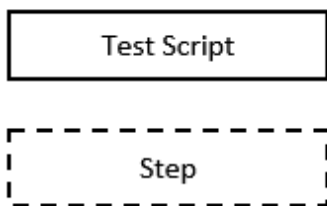


## Beispiel

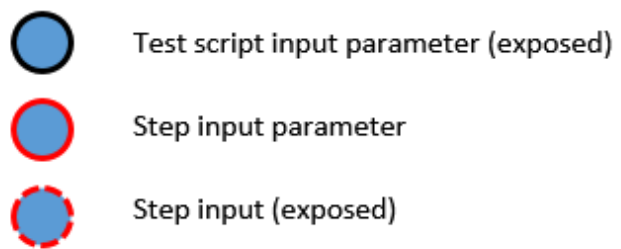
### Test Script



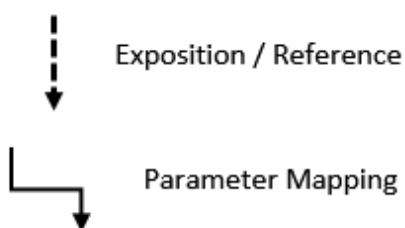
#### Legend



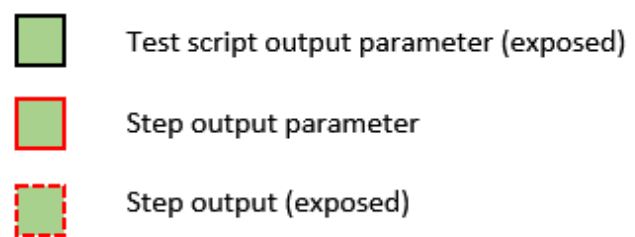
#### Inputs



#### Links



#### Outputs



In diesem Beispiel besteht das Testskript aus zwei Schritten. Jeder Schritt exponiert einen Eingabeparameter. Daher sehen Sie zwei Eingabeparameter auf Testskriptebene:

- In Schritt 1 wird ein Ausgabeparameter einer der Eingaben von Schritt 2 zugeordnet.
- Schritt 2 exponiert einen seiner Ausgabeparameter und macht diesen als Testskript -Ausgabeparameter sichtbar.

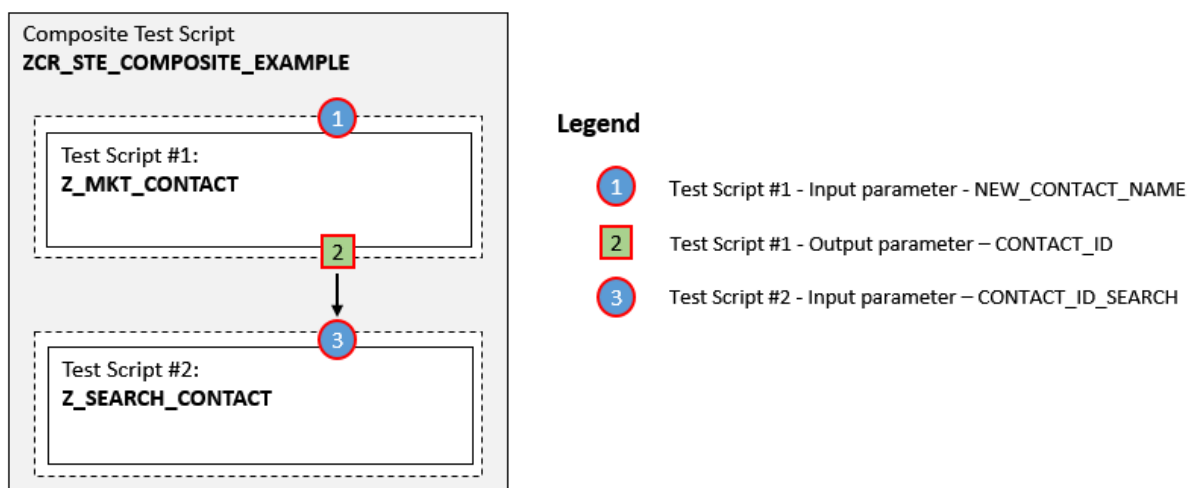
### 3.4.4.2.3 Zusammengesetzte Testskripte

Testskripte der komponentenbasierten Testautomatisierung können als untergeordnete Testskripte von anderen CBTA-Testskripten wiederverwendet werden. Jedes Testskript wird dann als Schritt des übergeordneten Testskripts behandelt. D.h., Sie können dieselbe Logik zum Exponieren und Zuordnen von Eingabe- und Ausgabeparametern zum Übergeben von Informationen von einem untergeordneten Skript an ein anderes verwenden, genauso, wie Sie dies von einem Schritt zu einem anderen tun würden.

#### i Hinweis

Nachdem Sie ein Testskript anhand einer Aufzeichnung generiert haben, können Sie die Parameter und ihre Zuordnung in der Umgebung für die Testzusammenstellung auf der Registerkarte *Testskript* bearbeiten.

Wenn Sie z.B. ein Testskript zum Anlegen eines neuen Vertrags mit einem zufälligen Namen und ein zweites Testskript, das den Vertrag über seine ID sucht, haben. Die offensichtliche Schwierigkeit besteht darin, dass Sie die ID des neuen Vertrags beim Anlegen des Testskripts nicht kennen, da sie beim Anlegen des Vertrags automatisch generiert wird. Daher müssen Sie in einem Szenario wie diesem diese Informationen von einem Testskript an ein anderes Testskript übergeben. Anders ausgedrückt, Sie müssen die Ausgabe des ersten Testskripts dem zweiten Testskript zuordnen, wie in der folgenden Abbildung gezeigt:



In diesem Fall ist das zweite Testskript ein untergeordnetes Testskript des ersten Testskripts. Dies hat den Vorteil, dass die Komplexität des untergeordneten Testskripts ausgeblendet ist. Sie verwenden das Testskript einfach unverändert wieder, und ihre Eingabe- und Ausgabeparameter werden wie reguläre Schrittparameter behandelt. Es ist nicht erforderlich, sie auf der Ebene der zusammengesetzten Testskripts zu exponieren.

Der erste Schritt beinhaltet einen Eingabeparameter, der den Namen des neu anzulegenden Ansprechpartners bereitstellt. Um sicherzustellen, dass Sie nie mehr als einmal einen Ansprechpartner mit demselben Namen anlegen, können Sie mit den von der komponentenbasierten Testautomatisierung angebotenen Token-Funktionen einen Zufallsnamen generieren, indem Sie den Wert `NEW_CONTACT_NAME` auf `Contract_%TODAY%_%RANDOM%` setzen.

Der erste Schritt enthält auch einen Ausgabeparameter mit der Bezeichnung `CONTACT_ID`. Um diesen Wert dem nachfolgenden Schritt zu übergeben, müssen Sie die Zuordnung in der Spalte *RefParameter* definieren. In diesem Beispiel ist der Name des referenzierten Parameters `CONTACT_ID_SEARCH`.

Test Script Steps						
View: * Claude						
<span>Up</span> <span>Down</span> <span>Move</span>   <span>Copy</span> <span>Paste</span> <span>Add</span> <span>Remove</span>   <span>Where-used</span> <span>CBTA Inspection</span> <span>Set State</span>						
Index	Type	Name	Version	Comment		
1	CBTA Test Script	Z_MKT_CONTACT	1	Creation of a new contact with a random name		
2	CBTA Test Script	Z_SEARCH_CONTACT	1	Searching for the newly created contact		
0						
0						

Parameters						
View: * CR						
<span>Set Usage</span> <span>Get Technical UI Information</span>						
Direction	Usage	Parameter Name	Value	Ref. Parameter	Reference	
Import	Fixed	NEW_CONTACT_NAME	Contact_%today%_%random%			
Export	Local	CONTACT_ID		CONTACT_ID_SEARCH	Z_SEARCH_CONTACT	

Der Wert der ID muss durch Ausführung unseres ersten Testskripts abgerufen werden. In der Regel können Sie dies tun, indem Sie Komponenten verwenden, die Informationen von den UI-Controls abrufen können. Das

Abrufen dieser Informationen ist sehr einfach. Sie müssen lediglich eine „getter“-Komponente wie *CBTA\_CRM\_GetAttribute* verwenden, wie in der folgenden Abbildung gezeigt:

**CBTA Test: Z\_MKT\_CONTACT**

Title Creation of Contact Version 1 Test Tool CBTA Executable NONE Target System █████

Save Edit Launch CBTA Execute Display Log Logon Where-Used Goto Refresh Close

Attributes Test Script Parameters Search Terms Administrative Data

**Test Script Steps**

View: Claude Up Down Move Copy Paste Add Remove Where-used CBTA Inspection Set State

Index	Type	Name	Version	Comment
1	CBTA Default Component	CBTA_CRM_NAVB_CLICKNAVLINK	1	Click on 'Master Data'
2	CBTA Default Component	CBTA_CRM_L_CLICKLINK	1	Click on 'Contact'
3	CBTA Default Component	CBTA_CRM_IF_SETVALUE	1	Set input field 'Last Name' with value 'Jul'
4	CBTA Default Component	CBTA_CRM_PRESSKEY	1	Press Key ENTER on 'Last Name'
5	CBTA Default Component	CBTA_CRM_GETATTRIBUTE	1	Get Property 'value'
6	CBTA Default Component	CBTA_CRM_BTN_CLICKBUTTON	1	Click on 'Save'
7	CBTA Default Component	CBTA_CRM_A_LOGOFF	1	Log off

Parameters Attributes Description Image

View: \* CR Set Usage Get Technical UI Information

Directi...	Usage	Parameter Name	Value	Ref. Parameter	Reference
Import	Fixed	URI	label=ID;; tag=INPUT; crm.i...		
Import	Fixed	ATTRIBUTENAME	value		
Import		EXPECTEDVALUE			
Import		TARGETFIELD			
Export	Exposed	OUTPUT		CONTACT_ID	Z_MKT_CONTACT

Die Informationen werden im Ausführungsreport angezeigt.

### 3.4.4.2.4 Parameter

Bei CBTA-Testskripts stimmen die Eingabe- und Ausgabeparameter des Schritts mit denen der CBTA-Komponenten überein. Weitere Informationen zu Komponenten finden Sie im PDF-Handbuch für Runtime Library and Default Components im Test-Suite-Wiki unter <https://wiki.scn.sap.com/wiki/x/OhdQGg>

#### i Hinweis

Nachdem Sie ein Testskript anhand einer Aufzeichnung generiert haben, können Sie die Parameter und ihre Zuordnung in der Umgebung für die Testzusammenstellung auf der Registerkarte *Testskript* bearbeiten.

Wenn Sie einen der Schritte im Testskript auswählen können, werden die entsprechenden Eingabe- und Ausgabeparameter unten im Bild angezeigt.

Im folgenden Beispiel verwendet der ausgewählte Schritt die Komponente `CBTA_WEB_CheckProperty`, um beispielsweise zu prüfen, ob der angezeigte Wert größer als 4 ist.

Details of **CBTA Default Component** CBTA\_WEB\_CHECKPROPERTY

Test Tool CBTA Version 1 Priority Medium Status Active Application Component SV-SMG-TWB Re

**Parameters** Attributes Description Image

View: \* [Standard View] Print Version Set Usage Get Technical UI Information URI Mass update

Usage	Direction	Parameter Name	Value
Fixed	Import	URI	ui5.type=sap.m.NumericContent; ..../ui5.data.title=Active S
Fixed	Import	PROPERTYNAME	ui5.data.value
Fixed	Import	OPERATOR	>
Exposed	Import	EXPECTEDVALUE	4
	Import	OPTIONS	
	Export	OUTPUT	

Diese Art von Prüfung wird in der Regel durch das Hinzufügen von Checkpoints während der Aufzeichnung des Szenarios generiert. Weitere Informationen finden Sie im PDF-Handbuch für Test Recorder im Test-Suite-Wiki unter: <https://wiki.scn.sap.com/wiki/x/OhdQGg>

### 3.4.4.2.4.1 Parameterverwendung

Wenn Sie in der Umgebung für die Testzusammenstellung auf der Registerkarte *Testskript* einen Schritt auswählen, werden die entsprechenden Eingabe- und Ausgabeparameter unten auf dem Bild angezeigt. Sie können verschiedene Einstellungen für die Parameter ändern, einschließlich der Angabe der *Verwendung*. Folgende Werte stehen zur Verfügung:

- **Kein Wert:** Kein Parameterwert wird an den Parameter des Testskripts übergeben. Der Testskriptparameter verfügt über keinen Standardwert.
- **Fest:** Ein Konstanten-Festwert wird an den Parameter übergeben. Der Konstantenwert ist nicht von der Ausführung des Testskripts abhängig.
- **Lokal:** Der an den Parameter übergebene Wert wird dem Parameter eines anderen Schritts zugeordnet. Der während der Ausführung des Testskripts tatsächlich verwendete Wert hängt von der Ausführung des übergeordneten Skripts/vorigen Schritts und der in den vorigen Schritten angelegten Werten ab.
- **Exponiert:** Der Parameter wird dem Parameter des übergeordneten Skripts zugeordnet. Der Parameterwert wird vom Parameter des übergeordneten Skripts übernommen (Importparameter) bzw. an den Parameter des übergeordneten Skripts (Exportparameter) übergeben. Importparameter können die Werte vom Testdatencontainer empfangen. Mehrere exponierte Importparameter können ihre Werte vom selben Parameter eines übergeordneten Skripts empfangen. Ein einzelner exponierter Exportparameter kann seinen Wert an mehrere Exportparameter des übergeordneten Skripts übergeben.

Technische Parameter sind in der Regel als fest gekennzeichnet. Beim Entwickeln oder Anpassen eines Testskripts müssen Sie den Wert eines festen Parameters ggf. ändern. Sobald das Testskript jedoch stabil ist, sollten Sie den Wert der festgelegten Parameter nicht mehr ändern.

Nichttechnische Parameter, d.h. Parameter, die von den Testern ggf. geändert werden müssen, sind standardmäßig exponiert.

## Weitere Informationen

[Bearbeitung von Parametern \[Seite 60\]](#)

### 3.4.4.2.4.2 Zuordnung von Eingabe- und Ausgabeparametern

In dem in der folgenden Abbildung gezeigten Beispiel prüft die Komponente `CBTA_WEB_CheckProperty` nicht nur den Wert (Wert > 4), sondern sammelt diese Informationen auch und macht sie über ihren Ausgabeparameter wieder verfügbar:

Details of **CBTA Default Component** CBTA\_WEB\_CHECKPROPERTY  
 Test Tool CBTA Version 1 Priority Medium Status Active Application Component SV-SMG-TWB Re

**Parameters** Attributes Description Image

View: \* [Standard View]

Usage	Direction	Parameter Name	Value
Fixed	Import	URI	ui5.type=sap.m.NumericContent; ../ui5.data.title=Active €
Fixed	Import	PROPERTYNAME	ui5.data.value
Fixed	Import	OPERATOR	>
Exposed	Import	EXPECTEDVALUE	4
	Import	OPTIONS	
	Export	OUTPUT	

Sie können einen neuen Schritt hinzufügen, und die Ausgabe einem der Eingabeparameter der neuen Komponente wie folgt zuordnen:

### Test Script Steps

Type	Name	Comment
CBTA Default Comp...	CBTA_WEB_CLICK	
CBTA Default Comp...	CBTA_WEB_CLICK	
CBTA Default Comp...	CBTA_WEB_CHECKPROPERTY	OUTPUT
CBTA Default Comp...	CBTA_WEB_A_CAPTURESCREEN	
CBTA Default Comp...	CBTA_GUI_A_REPORTMESSAGE	MESSAGE

### Parameters

Usage	Dire...	Paramete...	Value	Ref. Parameter	Reference
Fixed	Import	SEVERITY	INFO		
Fixed	Import	TOPIC	Demo		
Local	Import	MESSAGE		OUTPUT	CBTA_WEB_CHECKPROPERI
	Import	OPTIONS			

Die Komponente `CBTA_GUI_A_ReportMessage` kann zum Schreiben zusätzlicher Informationen in den Ausführungsreport verwendet werden. Dazu müssen Sie den Eingabeparameter `MESSAGE` dem Parameter `OUTPUT` des vorherigen Schritts zuordnen. Dies können Sie durch Ändern der Verwendung in *Lokal* und Angabe des Namens der Ausgabe in der Spalte *RefParameter* tun.

### 3.4.4.2.4.3 Meldungsparameter

Um Daten aus Meldungen für die Testautomatisierung von Web- und Web-Dynpro-Szenarien abzurufen, können Sie Meldungsparameter verwenden.

Sie können mit der Standardkomponente `CBTA_WEB_A_GETMESSAGEPARAMS` Informationen aus einer Meldung abzurufen, die von der Anwendung im Meldungsbereich angezeigt wird. Diese Komponente wird beim Aufzeichnen des getesteten Geschäftsszenarios nicht standardmäßig verwendet. Sie müssen diese manuell zu dem generierten Testskript hinzufügen. Dies ermöglicht die Suche nach einer Meldung im Meldungsbereich. Der Meldungssatz muss dem Muster entsprechen, das als Eingabeparameter eingegeben wird. Der Satz wird geparkt, und dessen Fragmente werden als Ausgabeparameter bereitgestellt.

#### Standardkomponente `CBTA_WEB_A_GETMESSAGEPARAMS`

- Eingabeparameter

- `URI` (optional): der URI, der den Meldungsbereich identifiziert  
Falls leer, sucht die Komponente im Hauptdokument des Hauptfensters von MS Internet Explorer nach den Meldungen.
- `MessagePattern`: das Muster, dem die Meldung entsprechen muss
- `Options` (optional): `/u` (Großschreibung), um den Fall zu ignorieren, wenn das Muster verwendet wird
- **Ausgabeparameter**
  - `MessageText`: stellt den gesamten Meldungstext bereit
  - `MessageParameter1`: stellt den Platzhalter `{1}` bereit, der dem Fragment entspricht
  - `MessageParameter2`: stellt den Platzhalter `{2}` bereit, der dem Fragment entspricht
  - `MessageParameter3`: stellt den Platzhalter `{3}` bereit, der dem Fragment entspricht
  - `MessageParameter4`: stellt den Platzhalter `{4}` bereit, der dem Fragment entspricht
- **Platzhalter**  
Die Syntax des Meldungsmusters kann Platzhalter enthalten, um einige Wörter aus dem Text zu extrahieren, der in der Anwendung angezeigt wird:
  - `{?}`: Übereinstimmung eines oder mehrerer Wörter. Verwenden Sie diesen Platzhalter, um einen Teil des Texts zu ignorieren.
  - `{1}`: Übereinstimmung eines oder mehrerer Wörter und Bereitstellung über `MessageParameter1`
  - `{2}`: Übereinstimmung eines oder mehrerer Wörter und Bereitstellung über `MessageParameter2`
  - `{3}`: Übereinstimmung eines oder mehrerer Wörter und Bereitstellung über `MessageParameter3`
  - `{4}`: Übereinstimmung eines oder mehrerer Wörter und Bereitstellung über `MessageParameter4`

### 3.4.4.2.4.3.1 Beispiel: Arbeiten mit Meldungsparametern

Das Szenario **Warenkorb anlegen** endet normalerweise mit einer Meldung, die den Benutzer über den Erstellungsstatus informiert. Diese Meldung enthält die Nummer des neuen Warenkorbs. Sie rufen sie ab und übergeben sie an die Folgeschritte des Testskripts.

Um eine Warenkorbnummer abzurufen, können Sie beispielsweise folgendes Meldungsmuster verwenden:

**Warenkorb „{?}“ mit Nummer {1} erfolgreich gesichert**

Mit diesem Muster werden die Wörter ignoriert, die Auskunft über den Benutzer und das Datum geben, und die Warenkorbnummer wird vom Ausgabeparameter `MessageParameter1` bereitgestellt.

Sie können auch die Informationen mit der folgenden Syntax abrufen:

**Warenkorb „{1} {2} {3}“ mit Nummer {4} erfolgreich gesichert**

In diesem Fall erhalten wir die folgenden Ausgabeparameterwerte:

- `MessageParameter1`: TESTER\_01
- `MessageParameter2`: 20.09.2013
- `MessageParameter3`: 22:02
- `MessageParameter4`: 1000133807



## Konfigurieren des Testskripts

1. Wählen Sie in der App *Test-Repository - Testskripte* ein Testskript aus.
2. Wählen Sie *Bearbeiten*.

Im Szenario *Warenkorb anlegen* ist die Anwendung in eine Portalseite eingebettet. Das aufgezeichnete Szenario endet mit der Meldung, die die Warenkorbnr. enthält. Standardmäßig ist kein Schritt zum Abrufen der Warenkorbnr. vorhanden, was erforderlich wäre, um zu prüfen, ob die Testausführung fehlgeschlagen ist. Wenn Sie während der Aufzeichnung einen Prüfpunkt (mit dem *Check-Selektor*) definiert haben, schlägt er fehl, da der Meldungstext Informationen enthält, die stets Änderungen unterliegen, beispielsweise Datum und Uhrzeit.

3. Verwenden Sie den Object Spy, um den URI des Meldungsbereichs abzurufen.

### **i** Hinweis

CBTA verwendet den URI, um UI-Elemente im HTML-Content zu identifizieren. Für die meisten Komponenten wird der Wert des URI-Parameters automatisch vom Testrecorder ausgefüllt und muss nicht angepasst werden. Der Meldungsbereich stellt eine Ausnahme dar. Die hier angezeigten Meldungen haben keine stabilen IDs und können mit dem üblichen Ansatz nicht gefunden werden.

1. Um den Object Spy zu starten, wählen Sie in der Umgebung für die Testzusammenstellung *Technische Informationen abrufen*.
2. Geben Sie den Meldungsbereich an.  
HTML-Attribute und -Eigenschaften werden in einer Baumstruktur angezeigt. In unserem Beispiel ist die Anwendung *Warenkorb anlegen* eine WebDynpro-(ABAP)-Anwendung auf Basis des Unified Rendering (Light Speed) Framework. Der URI des Meldungsbereichs wird im Kontextknoten *Light Speed* der Baumstruktur angezeigt.
4. Passen Sie das Testskript an:
  1. Wenn Sie während der Aufzeichnung einen Prüfpunkt definiert haben, deaktivieren Sie die Komponente `CBTA_WEB_CHECKPROPERTY`.
  2. Fügen Sie einen neuen Schritt hinzu, um die Komponente `CBTA_WEB_A_GETMESSAGEPARAMS` aufzurufen.
  3. Geben Sie den URI des Meldungsbereichs an, den Sie mit dem Object Spy ermittelt haben.
  4. Verwenden Sie Platzhalter, um die erforderlichen Fragmente aus dem Text der Meldung zu extrahieren.
5. Führen Sie das Testskript aus, um zu prüfen, ob die Meldungsdaten korrekt abgerufen werden.  
Die Standardkomponente `CBTA_WEB_A_GETMESSAGEPARAMS` ruft die Informationen aus der Meldung ab und stellt sie über die Ausgabeparameter für Folgeschritte zur Verfügung. Die Informationen werden im Ausführungsreport angezeigt.

### 3.4.4.2.5 Token

In der Umgebung für die Testzusammenstellung können Sie Parameter zuordnen, sodass die Ausgabeparameterwerte als Eingabe für die Folgeschritte wiederverwendet werden können. Diese Funktion ist sehr nützlich, jedoch nicht sehr flexibel. Deshalb können Sie Token verwenden, um die gemeinsame Nutzung von Informationen zwischen den Komponenten zu erleichtern. Alle Eingabeparameter können Token verwenden. Jedes Mal, wenn eine Komponente startet, sucht die Laufzeitbibliothek nach Token und ersetzt sie

durch den Wert der entsprechenden Variable. Der Wert wird direkt vom Ausführungskontext der komponentenbasierten Testautomatisierung abgerufen.

Es ist wichtig zu verstehen, dass Token während der Ausführung des Testskripts mandantenseitig aufgelöst werden. Die Tokensyntax ist ziemlich flexibel und ermöglicht Ihnen die Auswertung von VBScript-Ausdrücken oder den Zugriff auf Betriebssystem-Umgebungsvariablen.

Weitere Informationen finden Sie im Runtime Library and Default Components PDF Guide im Test-Suite-Wiki unter <https://wiki.scn.sap.com/wiki/x/OhdQGg>

### 3.4.4.3 Konfiguration

Bevor Sie die komponentenbasierte Testautomatisierung (CBTA) verwenden können, müssen Sie verschiedene Konfigurationsschritte ausführen. Sie führen die Konfiguration an verschiedenen Stellen aus.

#### 3.4.4.3.1 Installieren des CBTA-Clients

##### Kontext

Laden Sie den Client für die komponentenbasierte Testautomatisierung herunter, und installieren Sie ihn auf Ihrem lokalen Rechner. Mit der komponentenbasierten Testautomatisierung (CBTA) können Sie folgende Optionen konfigurieren:

- Auf Anfrage des SAP-Supports können Sie die Protokolleinstellungen anpassen und Fehlerprotokolle herunterladen.
- Zur Verbesserung der Lesbarkeit können Sie das Thema der CBTA-Benutzeroberfläche anpassen.

CBTA 3.0 SPO5 wird in einem SAP-Setup-Installationsprogramm wie SAP GUI ausgeliefert. Dieses Installationsprogramm bietet mehr Funktionen als der Microsoft System Installer (MSI), der für frühere Versionen der komponentenbasierten Testautomatisierung (CBTA) verwendet wurde. So können Sie mit SAP Setup:

- den Großteil der Windows™-basierten SAP-Frontend-Komponenten auf demselben Installationsserver warten
- SAP-Komponenten auf Workstations ohne Administratorrechte auf allen Rechnern implementieren
- Workstations automatisch aktualisieren und ggf. mithilfe des Services für automatische Workstation-Updates neu starten, wenn der Installationsserver gepatcht wird oder installierte Pakete aktualisiert werden

Das neue CBTA-Installationsprogramm ist eine automatisch extrahierbare Datei, die bei Ihrer Ausführung den SAP-Frontend-Installer ausführt

##### SAPSetup-Leitfaden

Der SAP Setup Guide ist im Installationsprogramm für die komponentenbasierte Testautomatisierung enthalten: Verwenden Sie WinRAR zur Extraktion der PDF-Datei mit dem SAP Setup Guide. Dieses Dokument

beschreibt alle Funktionen von SAPSetup und erläutert das Setup eines Installationsservers. Wir empfehlen außerdem, folgende SAP-Hinweise zu lesen:

Neueste Informationen zu SAPSetup

SAP-Hinweis	Beschreibung
<a href="#">1587566</a>	Detaillierte Release-Informationen zu und bekannte Probleme von SAPSetup Version 9.0 Beschreibung des Abrufvorgangs der neuesten Version von SAPSetup
<a href="#">1583967</a>	Allgemeiner Release-Informationshinweis zu SAPSetup Version 9.0
<a href="#">1162270</a>	Beschreibung der Installation des DS-Installationsservices (Analysehinweise) Weitere Informationen über das Erstellen und Warten eines Installationsservers zur Implementierung der SAP-Frontend-Software für Windows auf Client-Computern mit Local Security Handling
<a href="#">1177282</a>	Einstellung des Supports für SAP-Releases, die Microsoft-Produkte verwenden

### Erstinstallation

Wenn eine frühere Version der komponentenbasierten Testautomatisierung installiert ist, die den Microsoft Installer (MSI) verwendet, wird empfohlen, sie manuell zu deinstallieren und eine neue Version der komponentenbasierten Testautomatisierung zu installieren, die den SAPSetup-Installer verwendet.

#### i Hinweis

Außerdem empfehlen wir, alle SAP-GUI-Fenster zu schließen und sicherzustellen, dass der SAP-Anmeldeprozess vor dem Installieren oder Deinstallieren der komponentenbasierten Testautomatisierung nicht ausgeführt wird.

Zum Herunterladen des CBTA-Client-Installationspakets gehen Sie wie folgt vor:

### Vorgehensweise

1. Gehen Sie zu <https://support.sap.com/en/my-support/software-downloads.html>
2. Wählen Sie *Support Packages und Patches*.
3. Öffnen Sie den Knoten *By Alphabetical Index (A - Z)* und navigieren Sie zu **S** > *SAP Solution Manager* > *SAP Solution Manager 7.2* > *SUPPORT USER FRONTEND GUI* > *COMP.BASED TEST AUTOM. 3.0*.
4. Wählen Sie die entsprechende ZIP-Datei für die neueste Version.

Weitere Informationen finden Sie im SAP-Hinweis [1763697](#).

## 3.4.4.3.2 SAP-Solution-Manager-Konfiguration für die komponentenbasierte Testautomatisierung

Vor der Verwendung der komponentenbasierten Testautomatisierung müssen Sie die folgenden Guided Procedures in der SAP-Solution-Manager-Konfiguration (Transaktion `SOLMAN_SETUP`) ausführen:

- ► [Test-Suite](#) ► [Testautomatisierungs-Vorbereitung](#) ►
- ► [Test-Suite](#) ► [Komponentenbasierte Testautomatisierung](#) ►

### 3.4.4.3.3 CBTA-Konfigurationswerkzeug

Die Testautomatisierungslogik wird hauptsächlich auf Mandantenseite ausgeführt, während Sie ein Testskript aufzeichnen oder ausführen oder wenn Sie Probleme bei der Testausführung beheben. Sie müssen daher einige Einstellungen auf jedem Computer ausführen, den Sie zur komponentenbasierten Testautomatisierung verwenden.

SAP stellt Ihnen zur Unterstützung bei diesem Prozess das CBTA-Konfigurationswerkzeug zur Verfügung, auf das Sie über das Windows-Startmenü zugreifen können.

#### 3.4.4.3.3.1 Protokollierungs- und Tracing-Optionen

Sie nehmen die Einstellungen für Protokollierung und Tracing im CBTA-Konfigurationswerkzeug auf der Registerkarte [Protokoll](#) vor. Sie können z.B. die Protokoll- und Trace-Ebenen erhöhen, damit mehr Informationen über die Gründe gespeichert werden, aus denen Fehler aufgetreten sind oder aus denen sich die Anwendung nicht wie erwartet verhält. Sie können auch verschiedene Protokollierungsinformationen anzeigen.

#### 3.4.4.3.3.2 Konfigurieren zusätzlicher Browser für Aufzeichnungen

##### Kontext

Standardmäßig können Sie Aufzeichnungen nur mit Internet Explorer 11 erstellen. Allerdings können Sie die komponentenbasierte Testautomatisierung so konfigurieren, dass Aufzeichnungen in Google Chrome, Mozilla Firefox oder Microsoft Edge möglich sind.

##### Vorgehensweise

1. Öffnen Sie im Konfigurationswerkzeug der komponentenbasierten Testautomatisierung die Registerkarte für das [Browser-Setup](#).

2. Wählen Sie unter *Browser-Typ* den Browser aus, den Sie für Aufzeichnungen aktivieren möchten, beispielsweise Google Chrome.
3. Wählen Sie unter der *Aufzeichnungseinrichtung* die Drucktaste *Prüfen*.
4. Wenn der Prüfjob abgeschlossen wurde, wird ein Popup geöffnet, in dem Sie zur Konfiguration der Browserverbindung mit Ihrem Rechner aufgefordert werden.
5. Wenn Sie die Verbindung konfiguriert haben, legen Sie auf der Registerkarte *Aufzeichnung* den neu konfigurierten Browser als Standardbrowser für die Aufzeichnung von Testfällen fest.

### 3.4.4.3.3 Aufzeichnungsoptionen

Sie haben verschiedene Möglichkeiten zur Aufzeichnung, die Sie beim Starten des Testrecorders je nach Fall angeben können. Außerdem können Sie auf der Registerkarte *Aufzeichnung* im CBTA-Konfigurationswerkzeug auch die Standardeinstellungen für den Testrecorder ändern.

### 3.4.4.3.4 Ausführungsoptionen

Im CBTA-Konfigurationswerkzeug auf der Registerkarte *Ausführung* können Sie folgende Einstellungen für die Ausführung von Testfällen vornehmen.

- Ein Ausgabeformat für die Ausführungsreports auswählen  
CBTA-Ausführungsreports, die von einem Testskript generiert wurden, werden im Browser als HTML-Inhalt angezeigt. Sie können Reports jedoch auch in einem anderen Ausgabeformat anzeigen, z.B. als PDF- oder Word-Dokument.
- Auswählen, ob Browser-Sitzungen automatisch geschlossen werden sollen.  
Die komponentenbasierte Testautomatisierung (CBTA) schließt automatisch alle vorherigen Browsersitzungen, die durch eine Testausführung geöffnet wurden, jedoch nicht die Fenster, die Sie manuell geöffnet haben. Sie können jedoch festlegen, dass alle Fenster automatisch geschlossen werden, wenn Sie einen Testlauf starten.
- Geben Sie den für die Testausführung zu verwendenden Browser an.  
Dabei gelten folgende Einschränkungen:
  - Google Chrome: Nur Version 40 oder höher wird unterstützt.
  - Mozilla Firefox: Nur Version 40 oder höher wird unterstützt.
  - Microsoft Edge: Nur Versionen 10586, 14393, 15063, 16299 und 17134 werden unterstützt. Außerdem müssen Sie für die Kommunikation der komponentenbasierten Testautomatisierung mit dem Browser MS Edge Web Driver installieren.

#### **i** Hinweis

Obwohl MS Edge 14393 in der komponentenbasierten Testautomatisierung im allgemeinen unterstützt wird, funktioniert das Object-Spy-Werkzeug mit dieser Version nicht ordnungsgemäß.

- Das Ausführen von Testfällen für Webanwendungen, die IFrames mit anderen Domänen enthalten (ursprungsübergreifende gemeinsame Nutzung von Ressourcen) wird momentan nicht in Chrome, Firefox oder MS Edge unterstützt. Für diese Testfälle müssen Sie Internet Explorer 11 verwenden.

### 3.4.4.3.3.5 Zusätzliche Optionen

Im CBTA-Konfigurationswerkzeug auf der Registerkarte *Sonstige* können Sie verschiedene Optionen für verschiedene Einstellungen auswählen. So können Sie das Benutzungsoberflächen-Thema ändern, den Barrierefreiheitsmodus aktivieren und die Standardsprache der zur komponentenbasierten Testautomatisierung gehörenden Anwendungen ändern. Außerdem können Sie den Speicherort für die von der komponentenbasierten Testautomatisierung gespeicherten temporären Dateien festlegen.

### 3.4.4.3.4 SUT-Verwaltung

Das Werkzeug „SUT-Verwaltung“ ist ein wichtiger Bestandteil des Testautomatisierungs-Konfigurationsprozesses. Sie können die SUT-Verwaltung zum Erweitern des Systemdatencontainers (SDC) zur Bereitstellung zusätzlicher Informationen, die im Testautomatisierungsprozess benötigt werden, verwenden. Sie können z.B. folgende Informationen definieren:

- RFC-Destination zum SUT
- Testprofile für Anwendungsbenutzer
- URL zum Starten von Webanwendungen

Wählen Sie zum Öffnen der SUT-Verwaltung ein Testskript und wählen ► *Springen* ► *SUT-Verwaltung* ►.

Im Systemdatencontainer (SDC) definieren Sie die Zugriffsinformationen für die Systeme unter Test. Dies sind die Systeme, in denen Tests aufgezeichnet und ausgeführt werden. Hierzu sind Angaben zu RFC, URL, Benutzernamen und Kennwörtern erforderlich.

In der Umgebung für die Testzusammenstellung verknüpft das Testprofil einen Benutzer und ein Kennwort mit einem Testskript. So können Sie Kombinationen aus Destinationen und Benutzern für verschiedene Systemrollen wiederverwenden. Wenn Sie die Systemrolle ändern, können die Anmeldedaten, je nach Art der Definition, rollenabhängig oder -unabhängig sein. Ein Testprofil ist im Wesentlichen eine Kombination aus Benutzername und Kennwort, die dem Testskript zugeordnet ist.

In der SUT-Verwaltung können Sie Testprofile definieren. Sie können durch die logische Komponentengruppe, die Site und die Systemrolle definiert werden. Im Fall der komponentenbasierten Testautomatisierung (CBTA) werden die Testprofile zusammen mit einer technischen RFC-Destination definiert, die technische Informationen aus dem SUT oder der einfachen URL abrufen soll.

### 3.4.4.3.4.1 Importieren des Systemdaten-Containers

#### Kontext

Um ein Testprofil für ein System unter Test (SUT) anlegen zu können, müssen Sie zuerst den Systemdaten-Container (SDC) in die SUT-Verwaltung importieren. Nach dem Import wird der SDC in der erweiterten SDC-Hierarchie angezeigt.

## i Hinweis

Der SDC-Name setzt sich aus dem Buchstaben **z** und dem technischen Namen der Lösung zusammen.

Der SDC-Knoten enthält die logische Komponentengruppe (LKG), die in der mit dem SDC verknüpften Lösung definiert wurde. Unter einem LKG-Knoten befinden sich die logischen Komponenten (LKs), die zu den Lösungs-Branches und -Sites passen. Unter dem LK-Knoten befinden sich die Systemrollen, die auf das Zielsystem zeigen. Die Hierarchie hat in der Regel folgende Struktur:

## ❁ Beispiel

### Hierarchiestruktur

SDC (Lösung)

- LKG1 (CRM)
- LK1 (Branch-Produktion)
  - Systemrolle1 (Produktion)
  - Systemrolle2 (Qualität)
  - Systemrolle3 (Entwicklung)
- LK2 (Branch-Pflege – Site Frankreich)
- LK2 (Branch-Pflege – Site England)
- LKG2 (ERP)

## Vorgehensweise

1. Öffnen Sie die SUT-Verwaltungsanwendung.
2. Wählen Sie *Importieren*.
3. Suchen Sie nach dem SDC, den Sie importieren möchten, und bestätigen Sie Ihre Eingabe. Die Hierarchiestruktur wird befüllt.

## i Hinweis

Wenn Sie Änderungen an der SDC-Struktur vornehmen, indem Sie logische Komponentengruppen, logische Komponenten oder Systemrollen aus der Lösung hinzufügen oder daraus entfernen, synchronisieren Sie die SUT-Verwaltung, um sicherzustellen, dass Sie mit der aktuellsten SDC-Struktur arbeiten. Wählen Sie dazu *Synchronisieren* in der SUT-Verwaltung.

## 3.4.4.3.4.2 Konfigurieren eines SUT mit Testprofil

Der Name des Testprofils wird als Attribut eines Testskripts verwendet, um den Anwendungsbenutzer in verschiedenen Systemen zu repräsentieren. Das bedeutet, dass der Name des Testprofils derselbe Name in verschiedenen Systemrollen einer logischen Komponente sein kann, daher müssen keine doppelten Benutzer und Kennwörter für jedes System in der Landschaft verwendet werden.

Über die Drucktaste [Propagieren](#) wird das Testprofil zusammen mit dem zugehörigen Benutzer und Kennwort in der gesamten logischen Komponentengruppe zugeordnet.

Die SUT-Verwaltung unterscheidet zwei Arten von SUT: SAP ABAP-basierte Systeme und URI-basierte Systeme.

### 3.4.4.3.4.2.1 Testprofile für SAP-ABAP-basierte SUTs

Für SAP-ABAP-basierte Systeme ist die Konfiguration unterschiedlich, je nachdem, welches Werkzeug Sie für den Test verwenden. Für die komponentenbasierte Testautomatisierung müssen Sie eine technische Destination zuordnen, bevor Sie Testprofile hinzufügen können. Alle anderen Werkzeuge basieren auf RFC-Destinationen mit Anmeldebenutzern.

#### Zuordnen der technischen Destination für die komponentenbasierte Testautomatisierung

Die komponentenbasierte Testautomatisierung benötigt eine technische Destination, die Sie in der SAP-Solution-Manager-Konfiguration im Szenario [▶ Test-Suite](#) [▶ CBTA-Konfiguration](#) [▶](#) angelegt haben sollten.

In der SUT-Verwaltung können Sie [Destination vorschlagen](#) wählen, um die technische Destination automatisch auszuwählen. Die von der LMDB angelegten Destinationen für das ausgewählte System werden zum Abrufen der Systeminformationen und zum Berechnen des Namens der technischen Destination verwendet, falls vorhanden. Sie können [Prüfen](#) wählen, um die Gültigkeit der Destination zu prüfen. Der Status der Destination für das ausgewählte System wird entsprechend aktualisiert.

#### CBTA-Testprofil für SAP-ABAP-Backend

Für die komponentenbasierte Testautomatisierung (CBTA) besteht ein Testprofil aus einem Testprofilnamen, einem Anwendungsbenutzer und seinem Kennwort auf dem SUT. Wählen Sie in der SUT-Verwaltung die Drucktaste [Prüfen](#), um sicherzustellen, dass dieser Benutzer im System vorhanden ist. Der Status wird entsprechend aktualisiert.

### 3.4.4.3.4.2.2 Testprofile für URI-basierte SUTs

URI-basierte Systeme können nicht über RFC von SAP Solution Manager erreicht werden. Daher bestehen Testprofile für URI-basierte SUTs aus einem Testprofilnamen, einem Verbindungstyp, einer Verbindungs-URI, einem Anwendungsbenutzer und einem Kennwort. Die folgenden Verbindungstypen stehen zur Verfügung:

- **Basis-URL**  
Der Wert des ausführbaren Objekts vom Typ URL ist mit dem Verbindungs-URI verknüpft. Die Verbindung sollte zu einer gültigen URL führen, die mindestens ein Protokoll, einen Host und einen Port enthält.
- **HTTP-Destination**  
Der Wert für den Verbindungs-URI ist der Name der HTTP-Destination. Das Protokoll, der Host und der Port werden von der HTTP-Destination bestimmt. Dann wird das ausführbare Objekt vom Typ URL damit verknüpft.
- **Externer Service**  
Der Wert für den Verbindungs-URI wird automatisch von dem System vom Typ „Externer Service“ abgerufen. Der Verbindungs-URI und das ausführbare Objekt (alle unterstützten Typen außer Transaktion) werden berechnet, um eine gültige URL zu generieren.



## 3.4.4.3.5 Ausführen von Selbstchecks


### Kontext

Wenn Sie Probleme bei der Konfiguration Ihrer Landschaft für die Testautomatisierung haben, können Sie einen Selbstcheck ausführen, um die Einstellungen in Ihrem System, die die Voraussetzung nicht erfüllen, automatisch anzuzeigen.

Mithilfe der Selbstchecks für die komponentenbasierte Testautomatisierung (CBTA) kann der Administrator des SAP-Solution-Manager-Systems die Installation und Konfiguration des CBTA-Mandanten und den Zugriff auf das System unter Test (SUT) verifizieren. Darüber hinaus helfen Selbstchecks auch dem SAP-Support, Probleme bei der Installation und Konfiguration zu ermitteln. Unter Umständen werden Sie vom SAP-Support gebeten, Selbstchecks auszuführen und die Ergebnisse an den SAP-Support zu übermitteln.

Um sicherzustellen, dass Ihr System ordnungsgemäß für die UI-Technologie der zu testenden Anwendungen konfiguriert ist, können Sie den Typ des Frontend-Recorders anzeigen und damit die UI-Technologie (z.B. SAP GUI, CRM-Web-Client oder Web Dynpro ABAP) des Selbstchecks ermitteln.

### Voraussetzungen

- Sie haben die CBTA-Backend-Komponenten installiert.
- Sie haben den CBTA-Client lokal installiert. Weitere Informationen finden Sie im SAP-Hinweis [1763697](#) .


### Vorgehensweise

1. Wählen Sie in der Anwendung *Verwaltung - Test-Suite* den Pfad  *CBTA*  *Selbstchecks durchführen* .

Je nach Konfiguration des SAP Logon werden Sie vom System möglicherweise aufgefordert, das Senden von Anforderungen vom SAP-Solution-Manager-System an Ihren CBTA-Client zuzulassen. Bestätigen Sie die Meldung.

Das Bild *Testmanagement-Selbstchecks* wird geöffnet. Das System stellt eine Verbindung mit dem CBTA-Client her, um die verfügbaren Selbstchecks zu ermitteln. Dieser dynamische Abrufvorgang unterstützt zukünftige Erweiterungen sowie das Hinzufügen neuer Selbstchecks über Service Packs und Patches.

2. Wählen Sie die Registerkarte *Selbstchecks Frontend-Komponente*.

Liegen keine Konfigurationsprobleme vor, ist der erste Selbstcheck (CBTA COM Object Registration) grün und die Client-Version (COM-Komponente) wird in unter *Hilfe* angezeigt.
3. Wenn kein Selbstcheck verfügbar ist, wurde entweder der CBTA-Client nicht installiert, oder es liegt ein Konfigurationsproblem vor. In diesem Schritt führen Sie Folgendes aus:
  - Installieren Sie den CBTA-Client. Weitere Informationen finden Sie im SAP-Hinweis [1763697](#) .
  - Führen Sie die CBTA-Konfiguration erneut durch.
4. Führen Sie die Selbstchecks aus.
  - a. Um die SUT-Selbstchecks abzurufen, wählen Sie auf der Registerkarte *Systemdatencontainer-Prüfungen* die Zielkomponente, die logische Komponente und die Systemrolle.

- b. Wählen Sie [Selbstchecks abrufen](#).
  - c. Wählen Sie die verfügbaren Selbstchecks aus.
  - d. Wählen Sie [Ausführung starten](#).
  - e. Zum Beheben von Problemen folgen Sie den Anweisungen unter [Hilfe](#).
  - f. Wiederholen Sie diese Schritte, bis alle Selbstchecks ausgeführt wurden.
  - g. Um die Ergebnisse der Selbstchecks zu exportieren, z.B. um sie an den SAP Support zu senden, wählen Sie [Alle exportieren](#).
5. Die UI-Technologie eines Selbstchecks können Sie in der Spalte [Provider](#) auf den Registerkarten [Selbstchecks Frontend-Komponente](#) und [Systemdatencontainer-Prüfungen](#) nachschlagen.

### 3.4.4.3.6 CBTA Protokollkonfigurations-Tray

Zusätzlich zu diesen Einstellungen im CBTA-Hauptkonfigurationswerkzeug können Sie für schnellen Zugriff auf einige Grundeinstellungen auch das Symbol für den CBTA-Konfigurationswerkzeug-Tray in Ihrem Benachrichtigungsbereich installieren. So können Sie beispielsweise durch Auswahl des Tray-Symbols ein kleines Menü öffnen, in dem Sie den zur Aufzeichnung oder Ändern einiger grundlegender Ausführungsoptionen zu verwendenden Browser schnell ändern können, ohne das vollständige CBTA-Konfigurationswerkzeug öffnen zu müssen.

### 3.4.4.4 CBTA-Prozessübersicht

Der folgende Abschnitt bietet einen Überblick über den allgemeinen Prozess zum Testen mit der komponentenbasierten Testautomatisierung (CBTA). Einzelheiten zu den einzelnen behandelten Themen finden Sie entweder in der allgemeinen Anwendungshilfe für die SAP-Solution-Manager-Test-Suite oder in den entsprechenden Handbüchern auf dem [Test-Suite-Wiki](#), die im gesamten Abschnitt verlinkt sind.

## Voraussetzungen

- Um ein Drittanbieter-Testwerkzeug zu verwenden, haben Sie das Testwerkzeug auf dem Client-Rechner installiert, auf dem das SAP GUI für das SAP-Solution-Manager-System und für die Systeme unter Test installiert ist, und Sie haben das Testwerkzeug im SAP Solution Manager registriert.
- Sie haben Ihre Lösungsdokumentation definiert und dort die relevanten ausführbaren Einheiten angelegt. Die Auswahl einer ausführbaren Einheit beim Anlegen von automatisierten Tests ist ein obligatorischer Schritt, da so das System unter Test abgeleitet wird.
- Optional: Sie haben einen Testdatencontainer angelegt. Weitere Informationen finden Sie unter [Testdatencontainer im Testautomatisierungs-Framework anlegen \[Seite 23\]](#).

## 3.4.4.4.1 Anlegen von Testkonfigurationen von der Lösungsdokumentation

### Kontext

Eine Testkonfiguration enthält einen Systemdatencontainer, einen Testdatencontainer und ein Testskript. Die Daten aus diesen Elementen können in mehreren Testkonfigurationen wiederverwendet werden. Sie ordnen einer Lösung, die mit einem Systemdatencontainer verknüpft ist, eine Testkonfiguration und ein Testskript zu.

Wenn Sie eine Testkonfiguration anlegen, legen Sie damit automatisch einen Testfall an. Da ohne ein Testskript keine Testkonfiguration möglich ist, müssen Sie parallel das entsprechende Testskript anlegen. Zu einem späteren Zeitpunkt können Sie ein zugeordnetes Testskript bearbeiten oder durch ein anderes Testskript ersetzen.

#### i Hinweis

Sie können Ihre Testkonfigurationen in der App *Test-Repository - Testkonfigurationen* anlegen. Von dort aus können Sie zur Umgebung für die Testzusammenstellung navigieren und das entsprechende Testskript pflegen. Jedoch wird dringend empfohlen, Testkonfigurationen direkt von Ihrer Lösungsdokumentation aus anzulegen. Dadurch wird sichergestellt, dass der SAP-Solution-Manager-Kontext für die Ableitung der relevanten Systeme unter Test richtig definiert wird.

### Vorgehensweise

1. Öffnen Sie eine Lösung oder einen Branch über die App *Test-Suite - Testvorbereitung*.
2. Führen Sie unter *Browser* einen Drilldown auf den Geschäftsprozessschritt aus, für den Sie einen Test anlegen möchten. In der Tabelle der Elemente sehen Sie die ausführbaren Einheiten und Testfälle, die bereits zugeordnet wurden.
3. (Optional) Um dem Geschäftsprozessschritt eine neue ausführbare Einheit zuzuordnen, klicken Sie mit der rechten Maustaste in diese Tabelle, und wählen **Neu > Ausführbare Einheiten**. Diese ausführbare Einheit kann eine Transaktion, ein Programm, eine SAP-GUI-Transaktion, eine CRM-Anwendung (unter Verwendung der Rolle und der logischen Verknüpfungparameter), eine Web-Dynpro-ABAP-Anwendung oder eine URL sein. Alternativ können Sie eine vorhandene ausführbare Einheit verwenden.
4. Klicken Sie mit der rechten Maustaste auf die ausführbare Einheit, die Sie testen möchten, und wählen Sie **Neu > Testfälle > Testkonfiguration (Anlegen)**. Der Editor für die Testkonfiguration wird geöffnet.
5. Geben Sie bei Aufforderung grundlegende Informationen ein, wie z.B. Name für die Testkonfiguration, und füllen Sie die anderen obligatorischen Felder aus. Wählen Sie *CBTA* als Testwerkzeug aus, und bestätigen Sie Ihre Eingaben.
6. Nehmen Sie auf der Registerkarte *Attribute* die erforderlichen Einstellungen vor. Die von Ihnen ausgewählte ausführbare Einheit wird automatisch zugeordnet und zum Ableiten des Systems, in dem der Test ausgeführt werden soll, verwendet. Weitere Informationen finden Sie unter [Pflegen von Attributen \[Seite 220\]](#).

7. Wählen Sie unter *Zusätzliche Testoptionen* ein Testprofil. Dieses sollte bereits definiert sein. Sie können jedoch die Konsistenz der Konfiguration prüfen und ggf. Änderungen vornehmen:
  - a. Wählen Sie **Springen** **SUT-Verwaltung**. Die SUT-Verwaltung wird in einem neuen Fenster gestartet und wählt automatisch die relevanten Informationen durch Aufklappen der Knoten der Systemdatencontainer-Hierarchie basierend auf dem Solution-Manager-Kontext und Auswählen der aktuellen Systemrolle aus.
  - b. Wählen Sie *Bearbeiten*, um die RFC-Destination (falls relevant) und die Testprofile zu definieren.
  - c. Wählen Sie *Check*, um die Verbindung zum SUT zu überprüfen und stellen Sie sicher, dass ein Testprofil mit Benutzeranmeldedaten für jede Systemrolle vorhanden ist.
  - d. Wenn Sie zufrieden sind, sichern Sie die Daten und kehren Sie zum Testkonfigurationserstellungs-Bild zurück.
8. Wählen Sie unter *Zusätzliche Testoptionen* ein Anmeldeschema aus.

In den meisten Fällen müssen die zu testenden Anwendungen den Benutzer authentifizieren. Je nach verwendeter UI-Technologie ist die entsprechende Anmeldephase unterschiedlich.

#### **i** Hinweis

Wenn die ausführbare Einheit auf ein SAP-System abzielt, ist das Anmeldeschema möglicherweise implizit. Dies ist jedoch bei Verwendung einer ausführbaren Einheit vom Typ URL nicht der Fall, z.B. weil das Tool die Anmeldeseite nicht ermitteln kann. Daher müssen Sie manuell ein Login-Schema angeben.

- Für (auf einem ABAP-basierten SAP-System gehostete) SAP-Anwendungen wird automatisch das Login-Schema `ABAP_TAF_LOGON` verwendet.
  - Verwenden Sie für (auf einem Java Application Server gehostete) NetWeaver-Anwendungen das Login-Schema `NW_STANDARD_LOGON`.
  - Für in der SAP Cloud Platform gehostete Anwendungen basiert die Single-Sign-On-Benutzerauthentifizierung auf Zertifikaten. Daher müssen Sie das Login-Schema `SAP_CLOUD_IDENTITY` verwenden.  
Weitere Informationen finden Sie im PDF-Handbuch „Login Schema for Web Applications“ auf dem Test-Suite-Wiki unter: <https://wiki.scn.sap.com/wiki/x/OhdQGg>.
9. Wenn Sie fertig sind, sichern Sie die Testkonfiguration.

### **3.4.4.4.1.1 Login-Schema anlegen**

#### **Voraussetzungen**

Sie bearbeiten in der Umgebung für die Testzusammenstellung eine CBTA-basierte Testkonfiguration.

#### **i** Hinweis

Verwenden Sie nur nicht-produktive Benutzer und Systeme.

Verwenden Sie keine Login-Schemata für Testskripte, deren Testprofile sich auf ein SUT beziehen, für das Sie einen anonymen, URI-basierten Zugriff definiert haben. Weitere Informationen finden Sie unter [Konfigurieren eines SUT mit Testprofil \[Seite 47\]](#).

## Kontext

Um webbasierte Szenarios automatisieren zu können, müssen der Recorder und die Ausführungs-Engine in der Lage sein, sich unter Verwendung der Anmeldeseite automatisch am System unter Test (SUT) anzumelden. Da die SUT verschiedene Anmeldeseiten haben, muss der Benutzer ein Login-Schema definieren, indem er Uniform Resource Identifiers (URIs) der UI-Elemente angibt. Ein Login-Schema kann beispielsweise URIs von Eingabefeldern, Drucktasten, Ankreuzfeldern und Links definieren.

Um die Definition zu erleichtern, können Sie den URI der Anmeldeseitenelemente festlegen.

Mit der komponentenbasierten Testautomatisierung (CBTA) wird das Login-Schema `NW_STANDARD_LOGON` bereitgestellt, mit dem Sie Testskripte in SAP-NetWeaver-Java-Systemen mit Version 7.01 oder höher aufzeichnen und ausführen können.

Alle Login-Schemata werden im MIME Repository (`SE80`) des SAP-Solution-Manager-Systems gespeichert.

Ein Login-Schema besteht aus drei Teilen: Die Schritte vor der Anmeldung, Schritte zur Anmeldung und Schritte nach der Anmeldung. Jeder Teil umfasst mehrere Aktionen. Eine Aktion wird durch einen URI und einen optionalen Wert definiert. Dieser Wert wird verwendet, wenn für die Aktion ein Parameter erforderlich ist, z.B., um einen Wert in einem Eingabefeld festzulegen oder ein Ankreuzfeld auszuwählen (in diesem Fall wäre der Wert `geprüft`). Sie können keine Werte für das Benutzerfeld, das Kennwortfeld und die Anmeldedrucktaste definieren.

- Schritte vor der Anmeldung  
Führen Sie obligatorische Aktionen aus, bevor Sie den Benutzernamen und das Kennwort eingeben. Sie können so viele Aktionen wie nötig definieren.
- Schritte während der Anmeldung  
Hauptteil eines Login-Schemas. Definieren Sie drei obligatorische URIs:
  - Feld zur Erfassung des Benutzernamens
  - Feld zur Kennworteingabe
  - Drucktaste oder Link zur Aktivierung der Überprüfung der AnmeldeinformationenWenn das Anmeldebild andere Felder enthält (z.B. zum Festlegen der Sprache), können Sie zusätzliche URIs definieren. Die entsprechenden Aktionen werden ausgeführt, bevor die Anmeldeinformationen des Benutzers bestätigt werden.
- Schritte nach der Anmeldung  
Wenn Schritte nach dem Betätigen der Anmeldetaste auszuführen sind, können Sie entsprechende Aktionen definieren. Sie können so viele Aktionen wie nötig definieren.

## Vorgehensweise

1. Öffnen Sie die Anmeldeseite der zu testenden Anwendung.
2. Öffnen Sie das Testskript in der Umgebung für die Testzusammenstellung.

3. Öffnen Sie auf der Registerkarte *Attribute* unter *Zusätzliche Testoptionen* die Eingabehilfe (F4) des Login-Schemas.

Es wird ein Dialogfenster mit den im MIME-Repository vorhandenen Login-Schemas angezeigt.

Die komponentenbasierte Testautomatisierung stellt standardmäßig das Login-Schema `NW_STANDARD_LOGON` bereit, das Sie zum Aufzeichnen und Ausführen von Testskripten in Java-Systemen (Version 7.01 oder höher) verwenden können.

4. Wählen Sie *Anlegen*.

Der Assistent zum Erstellen von Login-Schemas wird geöffnet.

5. Geben Sie die erforderlichen Daten ein.

Wenn Sie den Namen eines vorhandenen Login-Schemas eingeben, wird standardmäßig ein Zählersuffix angehängt. Um ein vorhandenes Login-Schema zu überschreiben, wählen Sie das entsprechende Feld.

6. Wählen Sie im Login-Schema-Assistenten *Aktualisieren*.
7. Wählen Sie in der Liste der verfügbaren Sitzungen diejenige aus, die der Anmeldeseite der getesteten Anwendung entspricht.
8. Geben Sie die URIs auf der Anmeldeseite an:
  - a. Wählen Sie die Drucktaste *Spy* des zu identifizierenden Felds, z.B. des Felds „Benutzername“.
  - b. Wechseln Sie zur Anmeldeseite der getesteten Anwendung.
  - c. Setzen Sie auf der Anmeldeseite den Mauszeiger auf ein Objekt, z.B. das Feld „Benutzername“. Die jeweiligen UI-Elemente werden hervorgehoben. Wenn UI-Elemente nicht ordnungsgemäß hervorgehoben werden, überprüfen Sie den Zoom-Faktor im Browser (muss 100 % sein).
  - d. Wählen Sie das zu identifizierende Objekt aus.
  - e. Gehen Sie zurück zum Assistenten für die Login-Schema-Erstellung. Der URI des gewählten UI-Elements wird im zugehörigen Feld angezeigt.
  - f. Wiederholen Sie diese Schritte für alle festzulegenden UI-Elemente. Die folgenden Felder sind obligatorisch:
    - Feld zur Erfassung des Benutzernamens
    - Feld zur Kennworteingabe
    - Drucktaste „Bestätigung“

Um weitere Felder festzulegen (z. B. ein Feld für die Spracheneinstellung, definieren Sie zusätzliche Anmeldeschritte. Um den Wert festzulegen, wählen Sie das Feld *Mit dem Wert*.

### **i** Hinweis

Für die Felder „Benutzername“ und „Kennwort“ sind die Felder *Mit dem Wert* deaktiviert, da der Benutzername und das Kennwort automatisch aus der SUT-Verwaltung abgerufen werden.

9. Definieren Sie, falls erforderlich, die Schritte vor und nach der Anmeldung.

Wenn einige Aktionen ausgeführt werden müssen, **bevor die Anmeldedrucktaste gewählt wird**, definieren Sie zusätzliche Anmeldeschritte.

Wenn **vor der Eingabe des Benutzernamens** Aktionen ausgeführt werden müssen, wiederholen Sie diesen Vorgang mit einem Schritt vor der Anmeldung.

Wenn einige zusätzliche Aktionen ausgeführt werden müssen, **nachdem die Anmeldedrucktaste** ausgewählt wurde, wiederholen Sie den Vorgang mit einem Schritt nach der Anmeldung.

10. Geben Sie einen Namen und eine optionale Beschreibung für das neue Login-Schema ein. Es besteht die Möglichkeit, ein bestehendes Login-Schema mit demselben Namen zu überschreiben. Wenn diese Option nicht ausgewählt wurde, wird an den Namen dieses neuen Login-Schemas ein Zäblersuffix angehängt.

11. Wählen Sie *Weiter*.

Das Login-Schema wird in das MIME-Repository des SAP-Solution-Manager-Systems geladen.

Das Dialogfenster *Objektkatalogeintrag anlegen* wird angezeigt.

Wenn das Dialogfeld in der Windows-Taskleiste nicht angezeigt wird oder der Ladevorgang des Assistenten nicht funktioniert, minimieren Sie alle anderen Anwendungen.

12. Wählen Sie ein Transportpaket aus, und sichern Sie Ihre Eingabe.

13. Gehen Sie zurück zum Assistenten für die Login-Schema-Erstellung.

Der Status sollte Grün sein.

14. Wählen Sie *Fertig stellen*.

## Ergebnisse

Das Login-Schema kann beim Definieren der Testskriptattribute in der Umgebung für die Testzusammenstellung im entsprechenden Feld ausgewählt werden:

Bei der Aufzeichnung und Ausführung wird die Login-Schema-Datei in die Frontend-Umgebung heruntergeladen und dort bei der Anmeldung verwendet.

Login-Schema-Definitionsdateien (mit der Erweiterung `.lscd`) werden zentral im SAP Solution Manager hinterlegt und auf allen Frontend-Geräten, die CBTA verwenden, automatisch synchronisiert. Sie werden im MIME Repository bei den `CBASE`-Dateien abgelegt. Wenn Ihr Benutzer über die Rolle `S_DEVELOP` verfügt, können Sie diese Dateien über die Transaktion `SE80` unter `SAP/PUBLIC/CBTA/LOGIN_SCHEMA` anzeigen.

### 3.4.4.4.2 Erfassung von Geschäftsszenarios

#### Kontext

Nachdem Sie eine Testkonfiguration angelegt haben, können Sie mit der Aufzeichnung von Geschäftsszenarios beginnen. Verwenden Sie den Testrecorder zum Erfassen Ihrer Interaktionen mit der von Ihnen getesteten Anwendung und legen Sie automatisch ein Testskript auf Basis dieser Interaktionen an. Wenn Sie beispielsweise auf eine bestimmte Drucktaste klicken, erzeugt der Recorder einen Schritt im Testskript für das Klicken auf dieselbe Drucktaste usw.

Diese Tests beziehen sich auf eine einzelne Transaktion und können zu einem Szenariotest zusammengefasst werden. Sie pflegen die Komponenten in der Umgebung für die Testzusammenstellung.

Sie können folgende Anwendungsarten testen:

- SAP-ABAP-Server:
  - SAP GUI
  - SAP CRM Web Client
  - Web Dynpro ABAP
  - Business Server Pages (BSP)
  - SAP GUI für HTML
- SAP-Java-Server:
  - SAP Portal
  - Java Web Dynpro
- Web-Server
  - URL (HTML-Seiten)
  - SAP UI5/FIORI

### **i** Hinweis

Im Umfang von CBTA gibt es Einschränkungen. Weitere Informationen finden Sie im SAP-Hinweis [1835958](#)



## **Voraussetzungen**

- Sie haben den CBTA-Mandanten auf Ihrem lokalen Rechner installiert
- Sie haben ein Testprofil angelegt
- Sie haben eine Testkonfiguration oder ein Testskript angelegt und bearbeiten diese nun in der Umgebung für die Testzusammenstellung.
- Sie haben die Attribute der Testkonfiguration oder des Testskripts definiert.

## **Vorgehensweise**

1. Öffnen Sie die Testkonfiguration in der Umgebung für die Testzusammenstellung.
2. Wählen Sie **CBTA starten**. Der Testerstellungsassistent wird geöffnet.
3. Wenn Sie möchten, können Sie Änderungen an den Optionen wie dem Standardnamen vornehmen. Wenn Sie damit fertig sind, wählen Sie **Weiter**. Die von Ihnen getestete Anwendung wird im Aufzeichnungsmodus geöffnet..
4. Führen Sie das Geschäftsszenario, das Sie gerade testen, aus. Jedes Mal, wenn Sie mit der Benutzungsoberfläche interagieren, wird ein neuer Schritt zur Liste im Testrecorder hinzugefügt.
5. Sie können zusätzliche Aktivitäten wie Hinzufügen von Checkpoints, Entfernen von Schritten und Einfügen von Screenshots ausführen. Weitere Informationen finden Sie in der entsprechenden Dokumentation.
6. Wenn Sie fertig sind, wählen Sie die Drucktaste im Testrecorder, um die Aufzeichnung anzuhalten.
7. Vergewissern Sie sich, dass der Prozess korrekt aufgezeichnet wurde.
8. Um den Test anzulegen, wählen Sie **Weiter**. Das Testskript und die erforderlichen Testkomponenten werden generiert und in SAP Solution Manager hochgeladen.
9. Wählen Sie **Fertig stellen**.



10. Wählen Sie in der Umgebung für die Testzusammenstellung *Aktualisieren*. Die Umgebung für die Testzusammenstellung wird mit den Ergebnissen der Aufzeichnung aktualisiert. Die Testskriptparameter (sofern definiert) werden auf der Registerkarte *Parameter* angezeigt.

## Ergebnisse

Ihre Interaktionen mit der Anwendung werden zum Anlegen von Testskriptschritten verwendet. Diese werden nun in der Umgebung für die Testzusammenstellung auf der Registerkarte *Testskript* aufgelistet. Sie können die folgenden Aufgaben ausführen:

- Checkpoints zum Abrufen von Informationen von der Anwendungsoberfläche definieren und deren Konsistenz prüfen
- Einen Schritt zur Aufnahme eines Screenshots hinzufügen
- Unnötige Schritte entfernen
- Die Aufzeichnungsaktivität unterbrechen und neu starten

Weitere Informationen finden Sie im PDF-Handbuch „Test Recorder“ im Test-Suite-Wiki unter: <https://wiki.scn.sap.com/wiki/x/OhdQGg>

### 3.4.4.4.2.1 Hinzufügen von Checkpoints

#### Kontext

Sie können während der Aufzeichnung des Geschäftsszenarios Checkpoints definieren. Jeder Checkpoint wird im generierten Testskript in einen Schritt umgewandelt. Die zur Ausführung des Checks verwendete Komponente kann in Abhängigkeit von der verwendeten UI-Technologie variieren, die von der zu testenden Anwendung verwendet wird.

Sie können Checkpoints zum Prüfen der Konsistenz der Anwendung oder einfach zur Bereitstellung von Informationen aus der UI definieren.

#### Vorgehensweise

1. Wählen Sie während der Aufzeichnung eines Geschäftsszenarios im Assistenten zum Erstellen von Tests die Option zum *Hinzufügen von Checkpoints*.
2. Wählen Sie in der zu testenden Anwendung das UI-Element, das Sie prüfen möchten. Im Assistenten zum Erstellen von Tests wird der Liste an Schritten eine neue Zeile hinzugefügt. Die Liste der Eigenschaften, die geprüft werden können, wird entsprechend der UI-Elementart automatisch abgerufen. Wenn das UI-Element in der getesteten Anwendung noch sichtbar ist, wird der Wert der im Assistenten ausgewählten Eigenschaft automatisch eingefügt und aktualisiert.

3. Wählen Sie den entsprechenden Checkpoint-Typ aus:

- **Daten prüfen:** Wenn Sie einen Check ausführen möchten, wählen Sie *Daten prüfen*. Wählen Sie unter *Parameter* den Operator des auszuführenden Checks. Sie können einen Wert für den Check festlegen oder den Standardwert beibehalten, der nur abgerufen wird, wenn das UI-Control, auf das der Check abzielt, in der getesteten Anwendung noch verfügbar ist. Wenn die getestete Anwendung asynchron aktualisiert wird, wählen Sie unter *Optionen* die Option *Daten asynchron aktualisiert*, um den Check zu wiederholen, bis sie erfolgreich ist oder eine Zeitüberschreitung erreicht wird. Verwenden Sie diese Option, wenn die Testausführung für unbestimmte Zeit angehalten werden muss, z.B., bis ein Hintergrundjob abgeschlossen wurde. Wenn diese Option ausgewählt ist, können Sie unter *Optionen* einige zusätzliche Parameter festlegen:

- *Initiale Wartezeit:* die Zeit, die vor der ersten Ausführung des Checks abgewartet werden muss
- *Intervall:* die Zeit, die zwischen zwei Ausführungen des Checks abgewartet werden muss
- *Timeout:* die Gesamtzeit, in der Checks ausgeführt werden

Unter *Parameter* wird das Feld *Optionen* gemäß dem Wert dieser Parameter automatisch aktualisiert.

Dieser Wert wird für den `OPTIONS`-Parameter der `CBTA_WEB_CHECKPROPERTY`-

Standardkomponenteninstanz festgelegt, die in den generierten Test eingefügt wird. Sie können diesen Wert auch später in der Umgebung für die Testzusammenstellung definieren.

- **Daten beschaffen:** Wenn Sie diesen Checkpoint nur benötigen, um den Wert der UI-Elementeigenschaft für einen anderen Schritt verfügbar zu machen, legen Sie den Checkpoint-Typ auf *Daten beschaffen* fest. Es wird kein Check ausgeführt. Sie müssen keinen Operator und keinen Wert definieren, deshalb sind diese Felder unter *Parameter* deaktiviert. Eine `CBTA_WEB_GETPROPERTY`-Standardkomponente (oder `CBTA_WEB_GETATTRIBUTE`-Standardkomponente) wird in den generierten Test eingefügt. Ein Ausgabeparameter stellt den Wert der ausgewählten Eigenschaft für das gewünschte UI-Element bereit. Dieser Wert wird auch im Ausführungskontext gesichert.

## 3.4.4.4.2 Weitere Aktivitäten

In der Aufzeichnungsphase können Sie zusätzlich zum Hinzufügen von Checkpoints folgende Aktivitäten durchführen:

- **Screenshots einfügen**

Um den Testausführungs-Report übersichtlicher zu machen, können Sie einige Instanzen der Standardkomponente `CBTA_WEB_CAPTURESCREEN` während der Aufzeichnung in den Test einfügen. Um der Aufzeichnung den Screenshot-Schritt anzuhängen, wählen Sie *Bildschirmabgriff hinzufügen*. Sie können auch unter *Schritte* mit der rechten Maustaste auf einen Schritt klicken und *Screenshot einfügen* wählen. Unter *Schritte* wird ein neuer Schritt für einen Screenshot eingefügt.

- **Schritte entfernen**

Wählen Sie einen Schritt aus, und wählen Sie *Entfernen*, um einen Schritt aus der aktuellen Aufzeichnung zu löschen. Dies kann zu einem nicht ausführbaren Test führen, wenn Sie einen obligatorischen Schritt entfernen.

- **Anhalten der Aufzeichnung**

Um eine Aufzeichnung anzuhalten, wählen Sie *Pause*. Wählen Sie *Wiederaufnahme*, um fortzufahren.

### 3.4.4.4.3 Bearbeiten von Testskripten

#### Kontext

Nachdem Sie ein Geschäftsszenario aufgezeichnet haben, können Sie manuelle Änderungen an den Testskriptschritten vornehmen, die auf Basis der Aufzeichnung automatisch angelegt wurden.

Ein CBTA-Testskript besteht aus drei Arten von Komponenten:

- **Standardkomponenten** werden von SAP zusammen mit der CBTA-Software ausgeliefert. Sie sind im SAP-Solution-Manager-Repository für eCATT-Testskripte hinterlegt. Standardkomponenten werden zum Ausführen von Standardaktionen verwendet, z.B. zum Eingeben eines Werts in ein Feld, zum Betätigen einer Drucktaste oder zum Auswählen einer Registerkarte.
- **Bildkomponenten** werden, abhängig von den im aufgezeichneten Prozessschritt verwendeten Bildern, nach der Aufzeichnung eines Tests generiert. Mit dem Prüfprozess werden statische Informationen auf dem Bild aus dem System unter Test abgerufen. Für jedes Bild, das ein logischer Bestandteil einer Seite ist, wird eine Bildkomponente generiert. Eine Bildkomponente enthält einen Parameter für jedes Feld des Bilds. Wenn einem solchen Parameter ein Wert zugeordnet wird, gibt das System diesen Wert zur Ausführungszeit in das entsprechende Feld ein.  
Eine Bildkomponente bildet jeweils eine einzelne Version eines Bildes ab. Verschiedene Versionen eines Bilds werden für verschiedene Tests in unterschiedlichen logischen Komponenten für unterschiedliche logische Komponentengruppen gespeichert. Dadurch wird sichergestellt, dass die Test-Suite-Anwendungen zwischen unterschiedlichen Versionen desselben Bilds unterscheiden können. Die Version eines Bilds ist daher in allen Systemen mit derselben logischen Komponente identisch. Anderenfalls werden Bildkomponenten angelegt.
- CBTA-Testskripte können – wie Komponenten – auf den Inhalt eines anderen CBTA-Testskripts verweisen und untergeordnete Testskripte als Teil des CBTA-Haupttestskripts ausführen. So können Sie Gruppen von Standardkomponenten anlegen und in verschiedenen CBTA-Skripten wiederverwenden.

Sie können Standard- und Bildkomponenten in mehreren Testskripten wiederverwenden. Beim Aktualisieren eines Bilds können dessen Bildkomponenten repariert werden, sodass die Reparatur eines Testskripts vereinfacht wird. Dieser Prozess verwendet die TBOM (Technical Bill of Material) und den BPCA (Business Process Change Analyzer), um den Testaufwand nach einem System-Update oder -Upgrade zu verringern. Weitere Informationen finden Sie unter [TBOM-Arbeitsvorrat \[Seite 126\]](#) und [Business Process Change Analyzer \[Seite 130\]](#).

#### Vorgehensweise

1. Wählen Sie in der App [Test-Repository - Testkonfigurationen](#) die Testkonfiguration aus, die das Testskript enthält, das Sie bearbeiten möchten. Die Umgebung für die Testzusammenstellung wird geöffnet.
2. Wählen Sie auf der Registerkarte [Testskript](#) die Option [Aktualisieren](#). Die automatisch generierten Testskriptschritte werden aufgelistet. Die Schritte in einem CBTA-Testskript sind entweder Standard- oder Bildkomponenten.

Sie haben folgende Möglichkeiten:

- In der Spalte *Kommentar* können Sie eine Beschreibung für den Testskriptschritt eingeben oder eine vorhandene Beschreibung bearbeiten.
  - In der Spalte *Status* können Sie einen Testskriptschritt vorübergehend deaktivieren. Während der Testausführung wird der Schritt übersprungen, so können Sie z.B. verschiedene Varianten eines Testskripts probeweise ausführen, ohne dazu Schritte entfernen zu müssen.
3. Unter *Testskriptschritte* können Sie neue Schritte hinzufügen:
- a. Wählen Sie in der Spalte *Typ* eine Komponentenart aus:
    - *Standardkomponenten* verwenden Sie zum Ausführen grundlegender Aktionen, z.B. zum Eingeben eines Werts in ein Feld, zum Auswählen einer Registerkarte, zum Betätigen einer Drucktaste oder zum Klicken auf einen Link.
    - *Bildkomponenten* verwenden Sie zum Festlegen von Werten für eine Reihe von Feldern desselben Bilds (SAP GUI) oder derselben Sichten (CRM Web Client oder Web Dynpro ABAP). Wählen Sie zum Identifizieren des Bilds oder der Sichten eines Felds in der komponentenbasierten Testautomatisierung (CBTA) die Drucktaste *Zusatzinformationen abrufen*. Wählen Sie dann die Sitzung aus, in der das Feld angezeigt wird, und positionieren Sie den Mauszeiger auf dem Feld. Die technischen Informationen werden angezeigt: Diese umfassen den URI für das Feld sowie das Bild (Programm und Dynpronummer für SAP GUI) bzw. die Anwendung und die Sicht (CRM Web Client und Web Dynpro ABAP).  
Um anhand dieser Informationen eine Bildkomponente anzulegen, wählen Sie die Drucktaste *CBTA-Prüfung*.
  - b. Wählen Sie in der Spalte *Name* das Testskript aus.
  - c. Sichern Sie Ihre Eingaben. Leere Zeilen werden automatisch entfernt.
4. Um die Reihenfolge der Testskriptschritte zu definieren, verwenden Sie die Drucktasten *Nach oben* und *Nach unten*.
5. Um mehrere Schritte mit einem Testskript zu duplizieren, können Sie die Funktion zum Kopieren und Einfügen verwenden.
6. Um einen überflüssigen Schritt zu entfernen, z.B. weil Sie ihn versehentlich zweimal ausgeführt haben, wählen Sie *Entfernen*.
7. Sichern Sie Ihre Änderungen.

### 3.4.4.4.3.1 Bearbeitung von Parametern

## Kontext

Gehen Sie wie folgt vor, um die an die Testskriptschritte übergebenen Werte (Import- und Exportparameter) zu bearbeiten:

## Vorgehensweise

1. Öffnen Sie das Testskript in der Umgebung für die Testzusammenstellung.
2. Gehen Sie zu ► *Testskript* ► *Parameter* ▾.
3. Legen Sie im Feld *Verwendung* die Verwendungsart fest:
  - **Kein Wert:** Kein Parameterwert wird an den Parameter des Testskripts übergeben. Der Testskriptparameter verfügt über keinen Standardwert.
  - **Fest:** Ein Konstantenwert wird an den Parameter übergeben. Der Konstantenwert ist nicht von der Ausführung des Testskripts abhängig.
  - **Lokal:** Der an den Parameter übergebene Wert wird dem Parameter eines anderen Schritts zugeordnet. Der während der Ausführung des Testskripts tatsächlich verwendete Wert hängt von der Ausführung des übergeordneten Skripts/vorigen Schritts und der in den vorigen Schritten angelegten Werten ab.
  - **Exponiert:** Der Parameter wird dem Parameter des übergeordneten Skripts zugeordnet. Der Parameterwert wird vom Parameter des übergeordneten Skripts übernommen (Importparameter) bzw. an den Parameter des übergeordneten Skripts (Exportparameter) übergeben. Importparameter können die Werte vom Testdatencontainer empfangen. Mehrere exponierte Importparameter können ihre Werte vom selben Parameter eines übergeordneten Skripts empfangen. Ein einzelner exponierter Exportparameter kann seinen Wert an mehrere Exportparameter des übergeordneten Skripts übergeben.
4. Sichern Sie Ihre Eingaben.

## Beispiel

### Fester Umfang für Parameter

Wenn Sie immer einen Kundenauftrag über 5 Pumpen anlegen möchten, geben Sie **5** für den Parameter *Menge* ein. Die Werte verändern sich bei keiner Testskript-Ausführung.

Legen Sie in der Spalte *Wert* als Verwendung **Festwert** fest und geben Sie den Festwert (z.B. 5) ein. Zum Definieren einer leeren Zeichenfolge verwenden Sie den Wert **%BLANK%**.

### Lokaler Umfang für Parameter

Sie möchten ein Haupttestskript, das 3 Schritte (Art: BCTA-Testskripte) umfasst, mit den folgenden Parametern anlegen:

- Angebot anlegen
  - Importparameter: Kunde, Material, Menge
  - Exportparameter: Angebotsnummer

- Kundenauftrag aus Angebot anlegen
  - Importparameter: Angebotsnummer
  - Exportparameter: Kundenauftragsnummer
- Lieferung aus Kundenauftrag anlegen
  - Importparameter: Kundenauftragsnummer
  - Exportparameter: Lieferungsnummer

Legen Sie 3 Schritte in der Sequenz an. Um die lokale Verwendung für Parameter festzulegen, können Sie Exportparameter aus dem Schritt *Angebot anlegen – Angebotsnummer* mit dem Schritt *Kundenauftrag aus Angebot anlegen – Angebotsnummer* verknüpfen. Während der Testausführung wird nach dem Abschluss des ersten Testskripts die abgerufene Angebotsnummer an das zweite Testskript in den Angebotsparameter übergeben.

### 3.4.4.4.3.2 Anlegen von Bildkomponenten für ABAP-Web-Dynpro-Anwendungen

#### Voraussetzungen

- Sie haben ein Testskript mit einem ausführbaren ABAP-Web-Dynpro-Objekt angelegt.
- Das ausführbare ABAP-Web-Dynpro-Objekt enthält den Namen der Anwendung.

#### Kontext

Es kann vorkommen, dass Sie zur manuellen Bearbeitung eines Tests Bildkomponenten manuell anlegen müssen.

#### Achtung

Das manuelle Anlegen von Bildkomponenten ist eine erweiterte Funktion. Wir empfehlen, Testskripte automatisch mit dem Aufzeichnungsassistenten der komponentenbasierten Testautomatisierung aufzuzeichnen und nur dann Bildkomponenten manuell anzulegen, wenn dies unbedingt erforderlich ist.

Eine Bildkomponente kann im Rahmen eines Testskriptschritts eingefügt werden. Jeder Parameter einer Bildkomponente entspricht einem Eingabefeld auf der automatisierten Benutzungsoberfläche, z.B. Dropdown-Menü, Ankreuzfeld oder Auswahlknopf. Zur Ausführungszeit wird jeder Parameter, der einen Wert hat, auf der entsprechenden Benutzungsoberfläche mit diesem Wert befüllt.

Sie können die Bildkomponenten in mehreren Tests wiederverwenden. Dadurch wird das Reparieren von Tests einfacher, wenn Systeme durch neue Service Packs oder kundeneigene Entwicklungen aktualisiert werden.

#### Hinweis

Sie können Bildkomponenten in Tests nur dann wiederverwenden, die dieselbe logische Komponente verwenden. Dadurch wird sichergestellt, dass die getesteten Produkte dieselbe Version haben. Wenn keine

logische Komponente in der Zielkomponente des Systemdatencontainers des Tests verwendet werden, dann werden die Produktversionen des Systems unter Test zur Zusammensetzung kompatibler Bildkomponenten verwendet.

Für ABAP-Web-Dynpro-Anwendungen werden die Bildkomponenten basierend auf statisch definierten Web-Dynpro-Views aus derselben oder einer verwendeten Komponente generiert.

Wenn eine Web-Dynpro-View dynamisch zur Laufzeit generiert wird, können Sie keine Bildkomponente anlegen. Verwenden Sie stattdessen eine Standardkomponente. Weitere Informationen finden Sie im separaten Dokument zu Standardkomponenten unter <https://wiki.scn.sap.com/wiki/x/OhdQGg>.

## Vorgehensweise

1. Um den Namen der relevanten View zu identifizieren, wählen Sie auf der Benutzungsoberfläche der Web-Dynpro-Anwendung *Technische Hilfe* im Kontextmenü eines Felds, und prüfen Sie Folgendes:
  - Web-Dynpro-Komponente, die unter **Web-Dynpro-Anwendung** **Start Komponente** angezeigt wird
  - View-Name, der unter *Views und View Elemente* angezeigt wird
2. Öffnen Sie in den *Test-Repository*-Anwendungen das Testskript oder die Testkonfiguration zur Bearbeitung.  
Um die entsprechende View zu prüfen, wählen Sie auf der Registerkarte *Testskript* die Option *CBTA-Prüfung*. Der Prüfassistent wird gestartet.
3. Wählen Sie die View und dann *Weiter*.  
Die Bildkomponente wird generiert.
4. Wählen Sie *Fertig stellen*.
5. Fügen Sie die Bildkomponente und die entsprechenden Parameter dem Skript hinzu.
6. Identifizieren Sie den Parameterwert des übergeordneten Kontext-URIs (`PARENTCONTEXTURI`).

### ⚠ Achtung

- Der Parameter `PARENTCONTEXTURI` ist für ABAP Web Dynpros vorgesehen. Er ist für die Bedienbarkeit der Bildkomponente erforderlich.
- Verwenden Sie die Funktion *Object Spy*, um die ID (URI) der Benutzungsschnittstellen-Steuerung abzurufen. Verwenden Sie sie nur in Anwendungen mit stabilen IDs (Parameter `sap-wd-stableids = X` muss in der URL gesetzt sein).

- a. Öffnen Sie die Anwendung, die Sie testen möchten.
  - b. Kehren Sie zum Editor für Testskripte zurück. Wählen Sie auf der Registerkarte *Testskript* unter *Parameter* den Eintrag *Technische UI-Infos abrufen*.
  - c. Wechseln Sie zu Object Spy.
  - d. Wählen Sie auf der Registerkarte *Web-Applikation* die Sitzung aus, die Sie zuvor geöffnet hatten.
  - e. Wählen Sie *Spy*.
  - f. Platzen Sie den Cursor auf ein Feld in der Web-Dynpro-View. Das Feld wird rot hervorgehoben.
  - g. Um den Wert des Parameters `PARENTCONTEXTURI` anzuzeigen, gehen Sie in der Baumstruktur *Eigenschaftsname* zu **Light Speed UI** **Light Speed Context** **Parent Context Uri**.
7. Geben Sie den Wert des Parameters `PARENTCONTEXTURI` in den Parametern der Bildkomponente ein.

## 3.4.4.4.4 Ausführen von Testskripten

### Kontext

Sie können Testskripte direkt aus der Umgebung für die Testzusammenstellung ausführen.

### Vorgehensweise

1. Wählen Sie in der Umgebung für die Testzusammenstellung die Option *Ausführen*. Das Bild *Startoptionen* wird geöffnet.
2. Standardmäßig dient der SAP-Solution-Manager-Kontext zur Ermittlung des Zielsystems. Auf der Registerkarte *Allgemein* können Sie das SUT jedoch über die verschiedenen Optionen manuell ändern.
3. Passen Sie die verschiedenen Startoptionen an Ihre Anforderungen an, und starten Sie zum gewünschten Zeitpunkt das Testskript über *Ausführen*. Das Testskript wird auf Mandantenseite ausgeführt und das Testergebnis als XML-Datei generiert. Diese wird an das SAP-Solution-Manager-System zurückgesendet und dort zentral im Test-Repository gesichert.

## 3.4.4.4.5 Ausführungs-Reports anzeigen

### Kontext

Testausführungs-Reports stellen detaillierte Informationen zur Testausführung sowie zu ausgeführten Schritten, Parametern und aufgetretenen Fehlern bereit. Sie ermöglichen Ihnen eine Übersicht über den Inhalt eines durchgeführten Tests und über die Gründe beim Fehlschlagen des Tests.

Standardmäßig werden Ausführungs-Reports im HTML-Format in Ihrem Browser angezeigt. Sie können jedoch mit dem Konfigurationswerkzeug der komponentenbasierten Testautomatisierung ein anderes Ausgabeformat wählen, beispielsweise ein Microsoft-Word-Dokument. Um ein neues Ausgabeformat auszuwählen, gehen Sie zu **CBTA-Konfiguration > Ausführung > Report** und wählen Sie aus der Dropdown-Liste *Reportart* das gewünschte Ausgabeformat aus.

#### **i** Hinweis

Diese Änderung wird erst dann wirksam, wenn Sie sich das nächste Mal am SAP-System anmelden. Um die Änderung sofort zu übernehmen, melden Sie sich ab und wieder an.



## Vorgehensweise

1. Führen Sie das Testskript aus.
2. Wählen Sie in der Anwendung *Test-Repository - Testskripte* ein Testskript aus. Die Umgebung für die Testzusammenstellung wird geöffnet.
3. Wechseln Sie in den Bearbeitungsmodus.
4. Wählen Sie ► *Protokoll anzeigen* ► *Testwerkzeugprotokoll* ▾. Der Ausführungs-Report wird im gewünschten Format geöffnet.

### 3.4.4.4.6 Debuggen von Testskripten

#### Kontext

Wenn ein CBTA-Testskript fehlschlägt, können Sie die einzelnen Komponenten (Testschritte) nacheinander ausführen. CBTA stellt eine grafische Übersicht der Testschritte bereit und zeigt die entsprechenden UI-Komponentenattribute an.

Mit dem CBTA-Debugger können Sie einen Test auf Komponentenebene debuggen, ohne dass der VBS-Code (Visual Basic Script) angezeigt wird. Sie sehen den aktuellen Status der getesteten Anwendungsoberfläche sowie die Werte der Eingabe- und Ausgabeparameter.

Weitere Informationen finden Sie im PDF-Handbuch „Test Script Debugger“ im Test-Suite-Wiki unter: <https://wiki.scn.sap.com/wiki/x/OhdQGg>

#### Vorgehensweise

1. Öffnen Sie in der Launchpad-Gruppe *Test-Suite* die Anwendung *Test-Repository - Testskripte*.
2. Wählen Sie ein CBTA-Testskript aus, und wählen Sie *Ausführen*. Das Bild *Startoptionen* wird geöffnet.
3. Wählen Sie auf der Registerkarte *UI-Ansteuerung* im Feld *Modus für externes Tool* die Option *Debug-Modus*, und wählen Sie dann *Ausführen*. Der CBTA-Debugger wird gestartet. Es wird eine grafische Übersicht über die Testschritte angezeigt.
4. Sie haben folgende Möglichkeiten:
  - Wenn Sie einen Haltepunkt einfügen möchten, wählen Sie eine Komponente aus, und wählen Sie die Drucktaste *Toggle Breakpoint*. Ein Haltepunkt wird durch einen roten Punkt neben dem Komponentennamen dargestellt.
  - Um Parameterwerte dynamisch zu ändern, doppelklicken Sie auf den entsprechenden Tabelleneintrag, bearbeiten Sie den neuen Wert, und wählen Sie *Enter*.
5. Um die Testskriptschritte auszuführen, während die entsprechenden UI-Komponenten angezeigt werden, wählen Sie *Run* oder *Step Over*

- Wenn Sie *Run* wählen, werden alle Komponenten ausgeführt, bis ein Haltepunkt erreicht ist, ein Fehler auftritt oder der letzte Schritt ausgeführt wurde.
  - Wenn Sie *Step Over* wählen, wird nur eine Komponente ausgeführt. Der Debugger hält die Ausführung bei der nächsten Komponente an. Während der Ausführung der Testskriptschritte werden die entsprechenden UI-Komponenten angezeigt.
6. Sie haben folgende Möglichkeiten:
- Die **Eingabe- und Ausgabeparameter** werden rechts auf den entsprechenden Registerkarten angezeigt. Sie können Eingabeparameter der aktuellen Komponente ändern, indem Sie auf einen Wert doppelklicken, ihn bearbeiten und „Enter“ wählen. Die Änderungen werden bei der aktuellen Komponentenausführung berücksichtigt. Auf der Registerkarte für interne Parameter werden alle Parameter und Variablen angezeigt, die derzeit in der CBTA-Laufzeit definiert sind.
  - Zur Identifikation von **Fehlern im CBTA Protokoll** (z.B. nach Aufforderung durch den SAP Support), wählen Sie *Protokolle*. Sie können stattdessen auch die *CBTA-Konfiguration* öffnen und auf der Registerkarte *Protokoll* die Option *Protokoll anzeigen* wählen. Bei der Ausführung auftretende Fehler werden unten im CBTA-Debugger-Fenster angezeigt. Wenn ein Fehler auftritt, wählen Sie *Ausführen*, um die Ausführung zu beenden, und gehen Sie zurück zur SAP-GUI-Sitzung mit dem Bild *Startoptionen*, um den Ausführungsreport anzuzeigen.
  - Öffnen Sie die Registerkarte *Report* um eine dynamische Anzeige des HTML-Reports aufzurufen, der nach jeder Komponentenausführung generiert wird.

### 3.4.4.4.6.1 Object Spy

Object Spy ist ein Werkzeug zur Unterstützung bei der Behebung technischer Fehler in Testskripten, das mit der komponentenbasierten Testautomatisierung ausgeliefert wird. Es ist besonders nützlich bei der Identifizierung von Fehlern z.B. im Zusammenhang mit dem URI von Steuerelementen.

Sie können zum Öffnen von Object Spy den im Ausführungsreport neben jedem fehlgeschlagenen Schritt angezeigten Link auswählen. Alternativ wählen Sie in der Umgebung für die Testzusammenstellung den fehlgeschlagenen Schritt, und wählen Sie *Technische Benutzungsoberflächeninformationen abrufen*.

Weitere Informationen finden Sie im Object-Spy-PDF-Handbuch im Test-Suite-Wiki unter: <https://wiki.scn.sap.com/wiki/x/OhdQGg>

### 3.4.4.4.7 Aktualisieren von Testskripten

#### Kontext

Eine der größten Schwierigkeiten bei der Testautomatisierung ist, dass Testskripte kontinuierlich aktualisiert und bearbeitet werden müssen. Dies geschieht in der Regel dann, wenn eine neue Version der zu testenden Anwendung implementiert wird. In solchen Situationen könnten Sie theoretisch das Object-Spy-Werkzeug zum Beheben von durch Aktualisierung der zu testenden Anwendung entstandenen Problemen bei der

Testausführung verwenden. Diese Vorgehensweise kann sich jedoch als unmöglich erweisen, wenn ein Testskript potenziell aus Hunderten von Schritten bestehen kann.

Stattdessen können Sie versuchen, ein Testskript zu aktualisieren, indem Sie die betroffenen Teile des Geschäftsszenarios erneut aufzeichnen. Sie können die neu angelegten Schritte am Ende eines bereits vorhandenen Testskripts hinzufügen, um das Szenario zu erweitern, oder Sie können sie nach einem bestimmten Schritt in der Mitte des Testskripts einfügen.

Weitere Informationen finden Sie im PDF-Handbuch „Test Script Debugger“ im Test-Suite-Wiki unter: <https://wiki.scn.sap.com/wiki/x/OhdQGg>

## Vorgehensweise

1. Öffnen Sie in der Launchpad-Gruppe *Test-Suite* die App *Test-Suite - Testskripte*.
2. Wählen Sie das zu reparierende Testskript. Die Umgebung für die Testzusammenstellung wird geöffnet.
3. Wählen Sie auf der Registerkarte *Testskript* den Schritt, nach dem Sie das Testskript aktualisieren möchten.
4. Wählen Sie **► Ausführen ► Ausführen bis Schritt ▾**. Das Testskript wird bis zu diesem Schritt ausgeführt und wird dann angehalten.
5. Wählen Sie *CBTA starten*. Der Assistent zum *Anlegen von Tests* wird geöffnet.
6. Wählen Sie auf der Registerkarte *Einfach* nach Bedarf eine der folgenden Aufzeichnungsmöglichkeiten:
  - *Zusätzliche Schritte aufzeichnen und nach dem ausgewählten Schritt einfügen*
  - *Zusätzliche Schritte aufzeichnen und alles nach dem ausgewählten Schritt ersetzen*
7. Wählen Sie *Weiter* und fahren Sie wie gewohnt fort, um die neue Untergruppe von Schritten, die Sie einfügen möchten, erneut aufzuzeichnen.

## 3.4.4.5 Anpassung

### 3.4.4.5.1 CBTA um kundeneigene Entwicklungen erweitern

Es gibt Aktionen, die mit der standardmäßigen komponentenbasierten Testautomatisierung (CBTA) nicht ausgeführt werden können (z.B. das Aufzeichnen oder Abspielen von Tests mit der Adobe-Flash-Technologie, mit Java-Applets oder mit Web-Controls mit komplexem JavaScript-Handling). Für solche Fälle haben Sie die Möglichkeit, CBTA um kundeneigene Funktionen zu erweitern. Die CBTA-Engine basiert auf einer VBS-Bibliothek (Visual Basic Scripting), die auch als „Laufzeitbibliothek“ (Runtime Library, RTL) bezeichnet wird. Mithilfe des CBTA-Runtime-Library-Managers kann diese Bibliothek um kundeneigene Funktionen erweitert werden.

## Terminologie

**Laufzeitbibliothek:** Bei der Ausführung eines Testskripts wird das VB-Skript-Coding für den Test an den Client Computer gesendet und mit dem VB-Skript-Interpreter ausgeführt. Die Laufzeitbibliothek besteht aus einer

Reihe von VB-Skripten, die Helper-Klassen, Funktionen und Vorgänge bereitstellen, um Aktionen zu simulieren, die normalerweise von einem regulären Benutzer ausgeführt werden.

**Standardkomponenten:** Standardkomponenten sind Komponenten, die automatische Operationen für UI-Elemente ausführen. Die Laufzeitbibliothek umfasst Standardkomponenten für alle UI-Technologien, die von CBTA unterstützt werden.

**GUI-Scripting-API:** Wenn Sie als Testingenieur kundeneigene Funktionen für SAP-GUI-Anwendungen schreiben, müssen Sie möglicherweise das SAP-GUI-Scripting-API verwenden, das Schnittstellen zum Anwendungsinhalt bereitstellt.

**Query-API:** Wenn Sie als Testingenieur kundeneigene Funktionen für Webanwendungen schreiben, benötigen Sie möglicherweise Zugriff auf den HTML-Content, den die Anwendung im Microsoft Internet Explorer anzeigt.

Um eine neue kundeneigene Funktion zur Behebung von Testautomatisierungs-Issues anzulegen, laden Sie die Laufzeitbibliothek aus dem SAP-Solution-Manager-System herunter, und stellen Sie sie im lokalen Dateisystem Ihres Rechners zur Verfügung. Hinweis: Wenn der RTL-Manager erstmals gestartet wird, ist keine Anpassung aktiv, daher basierte die Ausführung des CBTA-Testskripts zuvor auf der offiziellen, von SAP ausgelieferten Laufzeitbibliothek.

## Voraussetzungen

- Sie haben den RTL-Manager-Webservice aktiviert.
- Sie verfügen über die Berechtigung für den RTL-Manager.
- Sie haben SAP-Hinweis [1912801](#) gelesen.

## Weitere Informationen

[Query-API für Tests des dynamischen Szenarios \[Seite 91\]](#)

### 3.4.4.5.1.1 Starten von Runtime Library Manager

#### Vorgehensweise

1. Wählen Sie im Launchpad *Test-Suite* die Anwendung *Verwaltung - Test-Suite* aus.
2. Wählen Sie **CBTA** **RTL anpassen**. Geben Sie im Anmeldebild des RTL-Managers Ihre Anmeldedaten ein. So stellen Sie eine Verbindung zum SAP-Solution-Manager-System mit der CBTA-Laufzeitbibliothek her.
3. Wählen Sie *Zum Bearbeiten öffnen*.

4. Geben Sie an, wo die Laufzeitbibliothek (RTL; runtime library) im lokalen Dateisystem gesichert werden soll.
5. Wählen Sie *OK*, um fortzufahren.

Im RTL-Manager wird ein zusätzlicher Bereich angezeigt. Dieser gibt den Status der Anpassung (*Ausstehend*) und den gewählten Speicherort an.

6. Wählen Sie *Durchsuchen*, um zu dem Ordner zu navigieren, in dem die kundeneigene Entwicklung gespeichert werden soll.
  - Dort befinden sich folgende Daten:
    - Der Stammordner *CBTA\_Customization* enthält eine ZIP-Datei mit der offiziellen Laufzeitbibliothek (*CBASE.zip*).
    - Der *CBASE*-Ordner enthält die CBASE-Konfiguration und die VB-Skriptdateien (\*.vbs). Der Inhalt der Datei *CBASE.zip* wird automatisch extrahiert.

#### **i Hinweis**

Modifizieren Sie die CBASE-Konfiguration und die VB-Skriptdateien nicht. Nur die im Ordner *Kundeneigen* gespeicherten VB-Skriptdateien können modifiziert werden.

- Der Unterordner *Kundeneigen* enthält die kundeneigene Entwicklung, die aus dem SAP-Solution-Manager-System heruntergeladen wurde. Beim ersten Mal kann dieser Ordner leer sein (oder sogar fehlen). Er wird automatisch angelegt, wenn Sie *Durchsuchen* wählen.

## **3.4.4.5.1.2 Manuelles Anlegen kundeneigener Funktionen**

### **Vorgehensweise**

1. Legen Sie die kundeneigene VB-Skriptdatei an, oder bearbeiten Sie eine vorhandene Datei. Zum Schreiben des Codes können Sie einen beliebigen Texteditor verwenden.

Stellen Sie Folgendes sicher:

- Die VB-Skriptdateien weisen die Erweiterung *.vbs* auf.
  - Die Bibliothek hat einen aussagekräftigen Namen, um Konflikte mit anderen Bibliotheken zu vermeiden.
  - Der Code deklariert keine globalen Variablen.
  - Funktions- und Variablennamen beginnen mit einem kundenspezifischen Präfix.
2. Speichern Sie Ihre kundeneigenen VB-Skriptdateien im Ordner „Kundeneigen“ (beispielsweise unter **C:\XBTA-Anpassung\CBASE\Kundeneigen**). Auf der Registerkarte „Funktionen“ werden die kundeneigenen VB-Skriptdateien angezeigt. Wenn der Ordner *Kundeneigen* leer ist, wird auf der Registerkarte *Funktionen* eine leere Liste angezeigt.
  3. Um die Syntax zu prüfen, starten Sie die VB-Skriptdatei manuell, indem Sie einen Doppelklick auf den Dateinamen ausführen.
  4. Wählen Sie *Aktualisieren*.

### 3.4.4.5.1.3 Verwendung von Mustern kundeneigener Entwicklungen in der CBASE-Anpassung

In der komponentenbasierten Testautomatisierung wird Ihnen eine Reihe von Mustern zur Verfügung gestellt, die der Testingenieur wiederverwenden und anpassen kann, um kundeneigene Funktionen anzulegen. Im Runtime Library Manager können Sie zusätzliche kundeneigene Funktionen schreiben, die in Testskripten zur Automatisierung eines Tests von Geschäftsszenarios verwendet werden sollen, für die der übliche Ansatz (basierend auf Standardkomponenten) nicht ausreicht.

Weitere Informationen finden Sie unter [CBTA um kundeneigene Entwicklungen erweitern \[Seite 67\]](#).

- SAP GUI: Von SAP-Anwendungen verwendetes UI-Framework
- SAP UI5/FIORI
- WebCUIF: Schicht, die den von SAP-CRM-Anwendungen angezeigten Content erzeugt.
- Unified Rendering Light Speed (LS): Schicht, die den von SAP-Anwendungen angezeigten Content erzeugt, wie z.B.:
  - Web-Dynpro-Anwendung (ABAP)
  - Web-Dynpro-Anwendungen (Java)
  - SAP GUI HTML – In MS Internet Explorer angezeigter SAP-GUI-Content
- BSP: Auf Business Server Pages basierende Anwendungen
- Java Web Dynpro: Schicht, die zuvor von Java-Web-Dynpro-Anwendungen verwendet wurde
- Plain-HTML-Seiten: Über reguläre HTML-Tags erzeugter Content (mit keinen oder wenigen Scripting-Funktionen)

Für die Testautomatisierung bringt jede UI-Technologie spezifische Probleme mit sich, von denen die meisten vom Test Recorder und Test Player bewältigt werden. Bei einem Geschäftsszenario mit hoher Dynamik ist das Standardverhalten jedoch nicht immer ausreichend, sodass der Testingenieur kundeneigene Funktionen schreiben muss.

## Voraussetzungen

- Sie haben die komponentenbasierte Testautomatisierung konfiguriert und das Testwerkzeug für die komponentenbasierte Testautomatisierung im Mandanten installiert.
- Sie haben den SAP-Hinweis [1778899](#) (CBTA-Sammelhinweis) gelesen.

## Prozess

Folgende Muster werden standardmäßig mit der komponentenbasierten Testautomatisierung ausgeliefert. SAP kann zusätzliche Muster über SAP-Hinweise ausliefern, je nach den Anforderungen des Kunden.

### 3.4.4.5.1.3.1 Muster für Web Dynpro ABAP: CheckForMessages

Diese Funktion sucht in einem Meldungscontainer nach Anwendungsmeldungen. Sie prüft, ob die Anwendungen Feedback melden, und ermittelt z.B. Fehler und Warnmeldungen.

#### i Hinweis

Die `CBTA_WEB_A_GETMESSAGEPARAM`-Standardkomponente ruft Meldungsparameter ab.. Sie ist generisch und unterstützt alle Web-UI-Technologien, kann jedoch keine Web-Dynpro-spezifischen Informationen wie z.B. den Meldungstyp (Fehlermeldung, Warnmeldung, Information) abrufen.

CBTA kann nicht ermitteln, ob diese Meldungen im Geschäftskontext normal sind, daher werden sie während der Laufzeit vom Testskript ignoriert und das Testskript wird fortgesetzt. Diese Meldungen kommen u.U. unerwartet, und es gibt Situationen, in denen Sie prüfen sollten, ob das Testskript fortfahren kann. Als Testingenieur können Sie die kundeneigene Funktion anlegen, um den Inhalt des Meldungsbereichs im Muster im Runtime Library Manager zu prüfen.

Signatur des `CheckForMessages`-Musters

- Eingabeparameter
  - `MessageContainerUri`  
URI des HTML-Containers, der Anwendungsmeldungen anzeigt. Falls leer, sucht die Funktion nach Meldungen im Meldungsbereich, die vom Hauptdokument des Hauptfensters angezeigt werden.  
Standardwert: `ls.rid=WDR_MESSAGE_AREA.MESSAGE_AREA.ROOTUIELEMENTCONTAINER`
  - `MessageType`  
Der Meldungstypparameter kann folgende Werte annehmen: **E** = error (Fehler), **W** = warning (Warnung) und **I**= information
  - `MessageRule`  
Der MessageRule-Parameter kann folgende Werte annehmen: **F** = forbidden (verboten), **R** = required (erforderlich) und **A**= allowed (erlaubt)
  - `MessagePattern`  
Nicht verwendet – reserviert zur späteren Verwendung
  - `Options`  
Der Options-Parameter kann folgende Werte annehmen: `/c` = Screenshot aufnehmen) und `/e` = Screenshot nur vom Meldungsbereich aufnehmen.
- Ausgabeparameter
  - `Output` = Anzahl der Meldungen, die mit den Kriterien übereinstimmen

Verwenden Sie das `CheckForMessages`-Muster, um zu prüfen, ob Anwendungsfehler angezeigt werden und um Fehler bei der Ausführung des Tests zu melden. Die Eingabeparameter könnten z.B. wie folgt aussehen:

- `MessageContainerURI =`  
`ls.rid=WDR_MESSAGE_AREA.MESSAGE_AREA.ROOTUIELEMENTCONTAINER`
- `MessageType = E`
- `MessageRule = F`
- `MessagePattern = [leer]`
- `Options = /c /e`

Anlegen und Verwenden einer benutzerdefinierten Funktion

Angenommen, die benutzerdefinierte Funktion wurde vom Runtime Library Manager angelegt und die entsprechende Bibliothek `Custom\CheckForMessages.vbs` wurde generiert. Der Runtime Library Manager zeigt Folgendes an:

...

Ein Testskript für die komponentenbasierte Testautomatisierung kann die benutzerdefinierte Funktion verwenden, sie ist jedoch nicht standardmäßig sichtbar. Sie können sie aufrufen, indem Sie eine der folgenden Komponenten verwenden: `CBTA_WEB_A_INVOKE_FUNCTION` und `CBTA_WEB_A_EXECUTE_STATEMENT`.

Im folgenden Beispiel ruft `CBTA_WEB_A_INVOKE_FUNCTION` die kundeneigene Entwicklung auf und prüft diese auf Anwendungsfehler:

...

Um die `CBTA_WEB_A_INVOKE_FUNCTION`-Komponente zu verwenden, muss der Testingenieur den Namen der benutzerdefinierten Bibliothek, den Namen der benutzerdefinierten Funktion und die Werte der Eingabeparameter angeben.

...

In diesem Beispiel werden folgende Parameter verwendet:

- `LIBRARY = Custom\CheckForMessages.vbs`
- `FunctionName = CheckForMessages`
- `PARAMETER1` = leer, da der `MessageContainerUri`-Parameter optional ist
- `PARAMETER2` = Meldungstyp **E** für Fehler
- `PARAMETER3` = Meldungsregel **F** für Forbidden (verboten)
- `PARAMETER4` = N/A
- `OPTIONS = /c /e` zur Aufnahme eines Screenshots des Meldungsbereichs

Wenn Sie `MessageType` auf **E** (Fehler) und `MessageRule` auf **F** (forbidden) setzen, schlägt der Test fehl, sobald der Meldungsccontainer eine oder mehrere Anwendungsfehler enthält.

Die Implementierung hat zwei Funktionen (beide Namen werden durch den jeweiligen Funktionsnamen ersetzt, den der Testingenieur angibt, wenn er die kundeneigene Funktion anlegt):

- Die Funktion `PATTERN_NAME` ist eine Fassade der Implementierung; sie umfasst die Ausnahmebehandlung, die normal lesbares Feedback im Fall von Skripting-Fehlern ausgibt.
- Die Funktion `PATTERN_NAME_Impl` ist die Implementierung.

## Syntax

### Code Syntax

```
Function PATTERN_NAME ( MessageContainerUri, MessageType, MessageRule,
MessagePattern, Options)
On Error Resume Next
EventComponentBegin()
If ConditionsManager().CheckConditions() Then
PATTERN_NAME = PATTERN_NAME_Impl ( MessageContainerUri, MessageType,
MessageRule, MessagePattern, Options )
End If
EventComponentEnd()
End Function
Function PATTERN_NAME_Impl ( MessageContainerUri, MessageType, MessageRule,
MessagePattern, Options )
...
End Function
```



Alle von SAP ausgelieferten Standardkomponenten werden mit diesem Ansatz entwickelt. Diese Struktur wird für sämtliche kundeneigene Entwicklungen empfohlen. Sie umfasst die Ausnahmebehandlung und ruft zwei Funktionen auf, die erforderlich sind, um die kundeneigene Entwicklung mit CBTA-Anforderungen kompatibel zu machen.

## Initialisierungsphase

Die ersten Anweisungen (**ImportLibrary...**) der Implementierung dienen lediglich dem Import der Bibliothek, von der die Implementierung abhängt.

Die benutzerdefinierte Funktion (**PATTERN\_NAME\_Impl = ""**) kann einen Wert zurückgeben, der den nachfolgenden Schritten des Testskripts über die Ausgabeparameter der Komponente zur Verfügung gestellt wird. Der Code initialisiert den Rückgabewert zu einer leeren Zeichenfolge:

Die nächsten Anweisungen (**IF ... END IF**) validieren die Eingabeparameter und setzen sie auf ihre Standardwerte zurück, wenn sie nicht vom Testskript definiert werden.

Die Funktion schreibt mithilfe der Methode ReportDebugLog Informationen in die (mit dem Ausführungsreport verknüpften) technischen Traces. Dies kann Ihnen bei der Behebung von Ausführungsproblemen helfen.

### CodeSyntax

```
ImportLibrary "WEB_WebController.vbs"
ImportLibrary "WEB_WebControls.vbs"
CBTA_WEB_A_INVOKE_FUNCTION
PATTERN_NAME_Impl = ""
If IsNull(MessageContainerUri) Then

MessageContainerUri="ls.rid=WDR_MESSAGE_AREA.MESSAGE_AREA.ROOTUIELEMENTCONTAIN
ER"
```

### CodeSyntax

```
End If
If IsNull(MessageType) Then MessageType = ""
End If
If IsNull(MessageRule) Then
MessageRule = "F"
End If
```

## Implementierung

Der unten angezeigte Code verwendet das Query-API für den Aufruf des MS-Internet-Explorer-Fensters und die Suche nach UI-Elementen im HTML-Content.

Informationen zur Query-API finden Sie unter

## Code Syntax

```
' Preparation of the Query
Dim query, filter Set query = LsController().CreateQuery()
query.ParentControlUri = MessageContainerUri ' Specifying
the criteria (using filters and conditions) Set filter =
query.SetFilter() filter.AddCondition "tag", "=", "IMG"
filter.AddCondition "ls.type", "=", "LN" filter.AddCondition
"ls.application", "=", "WDR_MESSAGE_AREA"
filter.AddCondition "ls.view", "=", "MESSAGE_AREA"
filter.AddCondition "ls.field", "=", "MSG_LIST_ICON"
```

Die Query sucht nach Bildern (<IMG>-HTML-Elemente), die untergeordnete Elemente des Meldungscontainers sind. Die Kriterien definieren auch Bedingungen, um nur Bilder auszuwählen, die die Meldungstypen darstellen. Aus diesem Grund werden hier Light-Speed-Attribute verwendet:

- **ls.type**: Art der Light-Speed-Steuerung; LN für Link
- **ls.application**: Name der Web-Dynpro-Anwendung
- **ls.view**: Name der Sicht, in der die Anwendungsmeldungen angezeigt werden
- **ls.field**: Name des Felds in der Sicht

Zusätzliche Bedingungen werden abhängig vom gesuchten Meldungstyp festgelegt. Die nachfolgend aufgeführten Bedingungen prüfen den Namen einer GIF-Datei, die für den Typ der Meldung steht.

Verwenden Sie diese Informationen zur Ermittlung des Meldungstyps, da dies die einzigen Informationen sind, die das Light-Speed-Framework zur Verfügung stellt. Daher wurde dieser Code nicht als Standardkomponente ausgeliefert.

## Code Syntax

```
If MessageType="E" Then
  filter.AddCondition "src", "{endsWith}", "ErrorMessage.gif" ElseIf
  MessageType="W" Then
  filter.AddCondition "src", "{endsWith}", "WarningMessage.gif" ElseIf
  MessageType="I" Then
  filter.AddCondition "src", "{endsWith}", "SuccessMessage.gif" Else
  ' No filter - This will show all messages (and count them)
End If
```

Die Query ist nun definiert. Die nächste Anweisung löst die Query auf und ruft eine Reihe von Light-Speed-Steuerelementen auf, die den Kriterien entsprechen:

## Code Syntax

```
Dim controlCollection
Set controlCollection = query.Select()
If controlCollection Is Nothing Then ReportLog "FAILED", "PATTERN_NAME",
"Operation Failed - Message container not found" Else ...
End If
```

Wenn die Sammlung nicht leer ist, führt die Implementierung die Prüfungen gemäß der angegebenen Meldungsregel aus. Die ReportLog-Methode schreibt normal lesbares Feedback in den Ausführungsreport.

## Code Syntax

```
If MessageRule="R" Then 'Required
```

```

    If controlCollection.Count = 0 Then      ReportLog "FAILED", "PATTERN_NAME",
"Operation Failed - No message of type: " &
MessageType
    Else      ReportLog "PASSED", "PATTERN_NAME", "Message Type: " & MessageType
&
_ controlCollection.Co      End If
ElseIf MessageRule="F" Then 'Forbidden
    If controlCollection.Count = 0 Then
        ReportLog "PASSED", "PATTERN_NAME", "No messages of type: " & MessageType
    Else
        ReportLog "FAILED", "PATTERN_NAME", "Operation Failed - Message Type: " &
MessageType &
_ vbCrLf & "Number of
messages: " & _ controlCollection.Cou      End If
ElseIf MessageRule="A" Then 'Allowed
    ReportLog "INFO", "PATTERN_NAME", "Number of messages: " &
controlCollection.Count
End If

```

Der folgende Abschnitt behandelt die Reihe von Steuerelementen zum Abrufen des Texts der mit dem IMG-HTML-Element verknüpften Meldung. Die `GetProperty`-Methode ruft die HTML-Eigenschaft `innerText` auf, die den angezeigten Text enthält.

#### CodeSyntax

```

Dim childControl, associatedControl
For i=0 To controlCollection.Count-1
    Set childControl = controlCollection.ControlAt(i)
    If Not childControl Is Nothing Then      Dim imgId,
spanId      imgId =
childControl.GetRelevantControl().GetLsId()

```

#### CodeSyntax

```

spanId = Replace( imgId, "MSG_LIST_ICON", "MSG_LIST_TEXTVIEW" )
Set associatedControl = childControl.FindElementById( spanId )
    If Not associatedControl Is Nothing Then      ReportLog "INFO",
"PATTERN_NAME", "Message #" & (i+1) & " - " &
_ associatedControl.GetProperty( "innerText" )      End If
    End If
Next

```

Der letzte Teil überprüft den Options-Parameter und nimmt einen Screenshot auf, indem die `WEB_CaptureScreen`-Methode (die mit der `WEB_CAPTURESCREEN`-Standardkomponente verknüpft ist) aufgerufen wird.

Zum Schluss legt die Funktion `PATTERN_NAME_Impl` fest, um die Anzahl der mit dem `MessageType` übereinstimmenden Meldungen zurückzugeben.

#### CodeSyntax

```

If InStr(Options, "/c") Then
    If InStr(Options, "/e") Then      WEB_CaptureScreen MessageContainerUri, "/e"
' Captures the Message Container only      Else
        WEB_CaptureScreen MessageContainerUri, "" ' Capture the whole browser
window
    End If
End If

PATTERN_NAME_Impl = "" & controlCollection.Count

```

### 3.4.4.5.1.3.2 Muster für CRM-Webanwendungen: SelectMenuItemByText

Einige CRM-Anwendungen verknüpfen ein Kontextmenü, mit dem der Benutzer zusätzliche Aktionen durchführen kann, mit bestimmten Steuerelementen. Der Testrecorder kann solche Aktionen normalerweise aufzeichnen und sie zur Laufzeit ausführen, indem er das Kontextmenü öffnet und einen Mausklick auf das untergeordnete Element simuliert.

Leider funktioniert das Standardverhalten bei einigen Szenarios nicht, da die mit dem untergeordneten Element verknüpfte ID nicht stabil ist: Die ID ist bei jeder Ausführung der CRM-Anwendung anders.

Daher suchen wir nach dem untergeordneten Element mithilfe des **SelectMenuItemByText**-Musters nach Text.

Verwenden Sie die CRM-Anwendung, z. B. um eine Meldung anzulegen: Benutzerrolle **SOLMANPRO** und logische Verknüpfung **SM-IM-CR**.

Der Object Spy oder das Microsoft Developer Tool (**F12** in MS Internet Explorer) zeigen an, dass die ID eine servergenerierte GUID enthält, daher können Sie das Element nicht anhand seiner ID suchen.

Signatur des **SelectMenuItemByText**-Musters

Sucht nach Anwendungsmeldungen in einem Meldungscontainer. Diese Funktion prüft, ob die Anwendungen Feedback melden, und ermittelt z.B. Fehler und Warnmeldungen.

- Eingabeparameter
  - **Uri**: URI des Steuerelements, mit dem das Kontextmenü verknüpft ist
  - **Text**: Text des auszuwählenden untergeordneten Elements. Bei diesem Text wird Groß-/Kleinschreibung beachtet, jedoch nicht, wenn die Option **/u** angegeben ist.
  - **Parameter3**: nicht verwendet
  - **Parameter4**: nicht verwendet
  - **Optionen**: mehrere Optionen, die das Verhalten beeinflussen, sind verfügbar.
  - **/u** = Großschreibung, bei der Elementauswahl wird dann keine Groß-/Kleinschreibung beachtet
  - **/click** = Wenn diese Option festgelegt ist, öffnet die Implementierung das Kontextmenü durch Simulieren eines Mausklicks auf das Hauptsteuerelement. Dies erfolgt nicht standardmäßig, da dies beim vorherigen Testschritt bereits ausgeführt wurde.
- Ausgabeparameter
  - **Output**: nach Ausführung der Aktion auf DONE setzen. Unterstützt die folgenden Werte:

**SelectMenuItemByText** verfügt über 5 Parameter, um der Standardkomponente zum Aufrufen von benutzerdefinierten Funktionen zu entsprechen.

**CBTA\_CRM\_A\_INVOKE\_FUNCTION**. Lassen Sie Parameter3 und Parameter4 beim Aufruf dieser kundeneigenen Entwicklung leer.

Die Implementierung besteht wie alle anderen Muster aus zwei Funktionen. Die erste ist eine Fassade für die Implementierung und enthält Anweisungen zur Behandlung von Ausnahmefehlern. Die zweite Methode führt den Job aus.

Das Fragment **PATTERN\_NAME** (siehe unten) wird bei Verwendung der Funktion Code-Assistent des Runtime Library Managers automatisch durch den Namen der finalen benutzerdefinierten Funktion ersetzt.

## ↳ Codesyntax

```
Function PATTERN_NAME ( Uri, Text, Parameter3, Parameter4, Options )
On Error Resume Next ' Important - Exception Handling - Do not change it!
EventComponentBegin()
If ConditionsManager().CheckConditions() Then      PATTERN_NAME =
PATTERN_NAME_Impl ( Uri, Text, Parameter3, Parameter4, Options )
End If
EventComponentEnd()
End Function
Function PATTERN_NAME_Impl( Uri, Text, Parameter3, Parameter4, Options )
'internal
ImportLibrary "CRM_WebController.vbs"
ImportLibrary "CRM_WebControls.vbs"
PATTERN_NAME_Impl = "NOT DONE"
...
End Function
```

Die ersten Anweisungen der Implementierung initialisieren lokale Variablen, validieren die Eingabeparameter und schreiben Informationen zur Fehlerbehebung in die Traces:

## ↳ Codesyntax

```
Dim bodyControl, bodyElement, collection, childElement, anchors, anchor,
currentText
If IsNull(Uri) Then
ReportLog "FAILED", "PATTERN_NAME", "Operation Failed - Button Uri not
specified"
End If
If IsNull(Options) Then
Options = ""
End If
```

Der nächste Abschnitt prüft die Options-Parameter, um zu ermitteln, ob es erforderlich ist, das Kontextmenü zu öffnen:

## ↳ Codesyntax

```
If InStr(Options, "/click" ) Then

CRM_WebControl_Click Uri
End If
```

## i Hinweis

Die folgende Implementierung ruft die `CRM_WebControl_Click`-Methode auf, die der Standardkomponente `CBTA_CRM_CLICK` entspricht.

Nach dem Öffnen des Menüelements kann die Implementierung mit der Suche nach untergeordneten Elementen fortfahren. Der folgende Code bewertet zuerst den Hauptteil des Arbeitsbereichs, indem er die von der CRM-Anwendungssteuerung bereitgestellte Methode `GetCrmControlByUri` verwendet. Dieser Schritt wartet, bis Dokumente bereit sind:

## ↳ Codesyntax

```
' Gets the body of the current workarea
```

```
Set bodyControl = CrmController().GetCrmControlByUri("crm.area=WorkArea;tag=body")
```

Wenn der Dokumenthauptteil bereit ist, sollte der Inhalt des Menüs sichtbar sein und die Implementierung kann fortgesetzt werden. Mit den folgenden Anweisungen wird auf das HTML-Element im Hauptteil über die Methode **GetHTMLElement** zugegriffen und über das Microsoft-API **MSHTML** nach **SPAN**-HTML-Elementen gesucht:

#### CodeSyntax

```
' Gets the body of the current workarea
Set bodyControl = CrmController().GetCrmControlByUri("crm.area=WorkArea;tag=body")
If bodyControl is Nothing Then
  ReportLog "FAILED", "PATTERN_NAME", "Operation Failed - Workarea not found"
Else
  Set bodyElement = bodyControl.GetWebControl().GetHTMLElement()
  If bodyElement is Nothing Then
    ReportLog "FAILED", "PATTERN_NAME", "Operation Failed - Workarea not found"
  Else
    Set collection = bodyElement.all.tags("SPAN")
    If collection is Nothing Then
      ReportLog "FAILED", "PATTERN_NAME", "Operation Failed - No SPAN HTML Elements"
    Else
      ReportLog "DONE", "PATTERN_NAME", "Count : " & collection.length
    ...
  End If
End If
End If
```

Folgende Anweisungen stützen sich nicht auf CBTA-Schnittstellen. Das **bodyElement** implementiert einige Skripting-Schnittstellen (vom **MSHTML**-API) und die Eigenschaften **all** und **tags** geben einen Satz Objekte zurück, die auch Scripting-Schnittstellen implementieren.

#### CodeSyntax

```
Set collection = bodyElement.all.tags("SPAN")
```

Die Sammelvariable enthält eine Liste der **SPAN**-HTML-Elemente im Arbeitsbereich der CRM-Anwendung. Die Implementierung filtert die **SPAN**-HTML-Elemente, die keine Menüeinträge sind, indem sie verifiziert, ob das **SPAN**-Element eine ID hat, einschließlich des **T\_MENU\_ITEM**-Fragments, und dass sie mindestens ein untergeordnetes Element vom Typ **ANCHOR** (**<A>**-HTML-Tag) aufweist.

Der sichtbare Text des Eintrags ist mit dem Element **ANCHOR** verknüpft. Sein Wert wird durch die **innerText**-Eigenschaft abgerufen und gegen den erwarteten Wert geprüft.

#### CodeSyntax

```
For i=0 To collection.length-1
  Set childElement = collection.item(i)
  If Not IsNull(childElement.id) Then
    If InStr( childElement.id, ":T_MENU_ITEM:") Then
      Set anchors = childElement.all.tags("A")
      If anchors Is Nothing Then
        ReportLog "FAILED", "PATTERN_NAME", "Operation Failed - No Anchor HTML Elements"
      Else
        ReportLog "DONE", "PATTERN_NAME", "Child Anchors -
```

```

Count : " & anchors.length          Set anchor = anchors.item(0)
      CBTA.Report CBTA.INFO, "PATTERN_NAME", "Anchor Element - id=" &
      anchor.id,
      ""
      currentText = anchor.innerText
      If InStr(Options, "/u" ) Then
      currentText = UCase(currentText)
      Text = UCase(Text)
      End If          If InStr(
      currentText, Text ) Then
      anchor.FireEvent("onmousedown")
      anchor.setActive()
      anchor.focus()
      anchor.FireEvent("onmouseup")
      anchor.click()
      PATTERN_NAME_Impl = "DONE"
      Exit For
      End If
      End If
      End If
      End If
Next

```

Wenn der Text mit dem gesuchten Text übereinstimmt, simuliert die Implementierung einen Mausklick auf den Eintrag und ruft so mehrere Methoden durch das API **MSHTML** auf:

#### CodeSyntax

```

anchor.FireEvent("onmousedown")
anchor.setActive()
anchor.focus()
anchor.FireEvent("onmouseup")
anchor.click()

```

Weitere Informationen finden Sie im Abschnitt *Schnittstellen für die Skripterstellung für Objekte von Microsoft* unter *Weitere Informationen*.

Anstatt die vom **MSHTML**-API bereitgestellten Methoden aufzurufen, könnte die Implementierung die **CRM\_WebControl\_Click**-Methode zum Auswählen von Einträgen verwenden, z.B. folgendermaßen:

#### CodeSyntax

```

Dim anchorUri anchorUri = "label=" & anchor.innerText & ";
crm.area=WorkArea; tag=A; id=" & anchor.id
CRM_WebControl_Click anchorUri

```

## Einschränkungen

Das **SelectMenuItemByText**-Muster sucht im Hauptteil eines HTML-Dokuments nach einem HTML-Element. Die aktuelle Implementierung hat folgende Einschränkungen:

- Sie unterstützt nicht die Selektion von Untereinträgen (untergeordnete Einträge eines anderen untergeordneten Eintrags).
- Sie sucht nur nach Einträgen im Hauptarbeitsbereich der CRM-Anwendung. Der Code muss angepasst werden, wenn der Eintrag im HTML-Dokument eines modalen Popup-Fensters angezeigt wird.

- Das mit dem Hauptsteuerelement verknüpfte Kontextmenü wird per Mausklick geöffnet. Für einige Steuerelemente ist u.U. ein Rechtsklick erforderlich.
- Der angezeigte Code setzt voraus, dass die Anwendung nur ein einzelnes Kontextmenü anzeigen kann, daher wird nicht geprüft, ob der Eintrag mit dem Hauptsteuerelement verknüpft ist.

### 3.4.4.5.1.3.3 Muster für CRM-Webanwendungen: SelectTransactionType

Das Muster SelectTransactionType sucht in dem Popup-Fenster, das beim Start mancher CRM-Anwendungen angezeigt wird, nach einer Vorgangsart und selektiert sie.

Manche SAP-CRM-Anwendungen werden gestartet, indem der Endbenutzer zur Eingabe einer Vorgangsart aufgefordert wird. Diese Funktion führt diese Aktion aus, indem die Vorgangsart anhand ihres Texts in der Liste sucht.

#### i Hinweis

Dieses Muster soll lediglich illustrieren, wie dieses Problem gelöst werden kann und der Query-API zu erklären, dass

CBTA solche Probleme lösen kann. Eine gleichwertige Komponente ist

**CBTA\_CRM\_SelectTransactionType**, die Sie ohne Anpassung der Laufzeitbibliothek verwenden können.

## Beispiel

Die Erstellung eines Kundenauftrags ist ein typisches Beispiel des Geschäftsszenarios, das mit der Auswahl der Vorgangsart beginnt. Der Typ ist nicht immer an derselben Position (abhängig von der Anzahl der Vorgangsarten in der Datenbank). Passen Sie das Testskript an, damit es nicht nach der Zeilennummer, sondern anhand des angezeigten Texts nach der Vorgangsart sucht.

Hinweis: Die Zeile wird standardmäßig nach Zeilennummer ausgewählt. Diese Information wird während der Aufzeichnung des Szenarios gesammelt und in dem generierten Testskript verwendet.

Die folgende Grafik illustriert die Automatisierung dieses Schritts. In der ersten Spalte wird die Vorgangsart, in der zweiten Spalte eine Kurzbeschreibung angezeigt. Beide Spalten können die auszuführende Vorgangsart identifizieren.

Unten auf dem Bildschirm befindet sich ein „Seiten“-Steuerelement zur Navigation zu den nächsten Seiten, wenn die Vorgangsart sich nicht auf der aktuellen Seite befindet.

## Signatur des SelectTransactionType-Musters

- Eingabeparameter



- **TransactionType**: Text in der Spalte *Vorgangsart*
- **Description**: (optional) Text in der Spalte *Beschreibung*
- **Operator**: (optional) der zu verwendende Operator beim Definieren des Filters (Standard ist =) Er vergleicht die Parameterwerte **TransactionType** und **Description** mit den tatsächlichen Werten der entsprechenden Spalten. Unterstützte Operatoren sind die im Query-API dokumentierten Operatoren.
- (optional) die Anzahl der Seiten, in denen nach der Vorgangsart gesucht werden soll Standardwert = **10**. Die Implementierung navigiert zur nächsten Seite, wenn die Vorgangsart nicht in der aktuellen Seite enthalten ist.
- **Options**: mehrere Optionen, die das Verhalten beeinflussen, sind verfügbar
  - **/t** = für abschneiden (trimming). Ignoriert voran- und nachgestellte Leerzeichen beim Vergleich der Parameterwerte **TransactionType** und **Description** mit den tatsächlichen Werten der entsprechenden Spalten.
  - **/u** = upper-case (Großbuchstaben). Für einen Vergleich ohne Berücksichtigung von Groß-/Kleinschreibung
- Ausgabeparameter
  - **Output**: Berichtstatus, **OK** oder **FAILED**.  
**SelectTransactionType** verfügt über 5 Parameter, um der Standardkomponente **CBTA\_CRM\_A\_INVOKE\_FUNCTION** zum Aufrufen von benutzerdefinierten Funktionen zu entsprechen.

### 3.4.4.5.1.3.4 Muster für SAP GUI: ME21N\_ExpandAllSections

Stellt den initialen Status der Transaktion ME21N/ME51N Hauptbild wieder her.

Das Bild besteht aus Bereichen. Jeder Bereich kann auf- oder zugeklappt werden, je nachdem, wie die Transaktion während der vorherigen Sitzung beendet wurde.

Die Implementierung besteht wie die meisten Muster aus zwei Funktionen. Die erste ist eine Fassade für die Implementierung und enthält Anweisungen zur Behandlung von Ausnahmefehlern. Die zweite Methode führt den Job aus.

### 3.4.4.5.1.3.5 Muster für SAP GUI: WaitForCondition

Prüft auf eine Bedingung und stoppt so vorübergehend den Ausführungsablauf.

Einige SAP-GUI-Anwendungen starten einen Hintergrundjob, um Geschäftsdaten zu berechnen. Das Testskript muss ggf. unterbrechen und warten, bis der Hintergrundjob abgeschlossen ist, bevor es fortgesetzt wird.

Das Fragment **PATTERN\_NAME** wird bei Verwendung der Funktion Code-Assistent des Runtime Library Managers automatisch durch den Namen der finalen benutzerdefinierten Funktion ersetzt.

#### ☰ Codesyntax

```
Function PATTERN_NAME ( TransactionType, Description, Operator, MaxPage,
Options )
On Error Resume Next ' Important - Exception Handling - Do not change it!
```

```

EventComponentBegin()
  If ConditionsManager().CheckConditions() Then      PATTERN_NAME =
PATTERN_NAME_Impl ( TransactionType, Description, Operator,
MaxPage, Options)
  End If
  EventComponentEnd()
End Function
Function PATTERN_NAME_Impl ( TransactionType, Description, Operator, MaxPage,
Options ) 'internal
  ImportLibrary "CRM_WebController.vbs"
  ImportLibrary "CRM_WebControls.vbs"
  PATTERN_NAME_Impl = "NOT DONE"
  ...
End Function

```

## Initialisierungsphase

Die ersten Anweisungen der Implementierung initialisieren lokale Variablen und validieren die Eingabeparameter:

### Code syntax

```

Dim query, filter, subFilter, selectorInPopup, maxPageAsInt, i, uriNextPage,
nextPageControl
maxPageAsInt = ToInt(MaxPage) If
IsNull(maxPageAsInt) Or maxPageAsInt = 0 Then
maxPageAsInt = 10 End If
If IsNull(Operator) Then
Operator = "="
End If
If IsNull(Options) Then
Options = ""
End If

```

**ReportDebugging** schreibt zur Fehlerbehebung Informationen in die Traces:

### Code syntax

```

ReportDebugLog "CUSTOM Function - PATTERN_NAME" &
" - Parameters: TransactionType " & TransactionType & " -
Description: " & Description
ReportDebugLog "CUSTOM Function - PATTERN_NAME" &
" - Operator: " & Operator & " - MaxPage: " & maxPageAsInt &
Options: " & Options

```

## Implementierung

Die Implementierung besteht aus einer Schleife (die die Anweisungen **FOR/NEXT**/ verwendet), um auf jeder Seite nach der Vorgangsart zu suchen.

Innerhalb dieser Schleife definiert das Query-API die Zeilenabgleichskriterien. Das Query-Objekt wird vom CRM-Controller angelegt. Das Query-API geht mit Situationen um, die spezifisch für die zugrundeliegende UI-

Technologie sind. CRM-Anwendungen stützen sich auf das **WebCUIF**-Framework. Die im folgenden Beispiel angelegte Abfrage nutzt die Meta-Informationen, die dieses Framework in den HTML-Content einfügt, sodass der Testingenieur gängige Anwendungsfälle wie diesen bearbeiten kann. Weitere Informationen finden Sie unter Query-API für Tests des dynamischen Szenarios [Seite 41].

### Codesyntax

```
For i=1 To maxPageAsInt
    ' Preparation of the CRM Query - Identifying the parent Table container
    (in the
    Popup) Set query = CrmController().CreateQuery()
    query.ParentControlUri = "popupId=1;
    crm.id=A_btfollowup:V_ProcType:T_cellerator:C_proctype:I "
    ' Specifying the element type to search for - ROW_SELECTOR
    Set filter = query.SetFilter() filter.AddCondition
    "crm.type", "=", "ROW_SELECTOR" filter.AddCondition
    "crm.context", "=", "proctype" If Not
    IsNull(TransactionType) Then
        ' filtering on the Transaction Type
        filter.AddCondition "crm.column:proctype.process_type", Operator,
    TransactionType, Options
    End If
    If Not IsNull(Description) Then ' filtering
    on the Transaction Type Description
        filter.AddCondition "crm.column:proctype.proc_type_descr_20", Operator,
    Description, Options
    End If
    ...
Next
```

Die Eigenschaft **ParentControlUri** beschränkt den Umfang auf die Abfrage. In diesem Beispiel wird der URI der Tabelle (im Popup-Fenster angezeigt) angegeben.

Ein Filter wird definiert, um anzugeben, was gesucht werden soll: Die erste Bedingung prüft die **crm.type**-Attribute. Sie gewährleistet, dass nur **ROW\_SELECTOR** mit den Kriterien übereinstimmt. Die zweite Bedingung prüft auf den Kontext **C\_fragment** der ID, die vom **WebCUIF**-Framework generiert wird.

Es werden zwei weitere Bedingungen definiert, abhängig von den Eingabeparametern: Wenn der Parameter **TransactionType** festgelegt wird, wird eine zusätzliche Bedingung angelegt, um nach den Inhalt der Spalte mit dem technischen Namen **proctype.process\_type** zu überprüfen. Wenn der Parameter **Description** festgelegt wird, wird eine weitere Bedingung angelegt, um den Inhalt der Spalte mit dem technischen Namen **proctype.proc\_type\_descr\_20** zu überprüfen.

### i Hinweis

Die Abfrage wird innerhalb der Schleife angelegt, da das Query-Objekt eine eingeschränkte Lebensdauer hat, die von dem angezeigten HTML-Dokument abhängt. Bei der Navigation zur nächsten Seite wird das aktuelle HTML-Dokument verworfen und ein neues Dokument wird angezeigt, daher ist eine neue Abfrage erforderlich.

## Abfrageauflösung

Nachdem die Abfrage definiert wurde, kann sie aufgelöst werden, um die Controls aufzurufen, die mit den Kriterien übereinstimmen. In diesem Anwendungsfall erwarten wir nur ein einzelnes Control (eine einzelne

Zeile), daher verwenden wir hier die Methode **SelectSingle**. Die Variable **selectorInPopup** empfängt das Abfrageergebnis:

#### CodeSyntax

```
' Selecting the first element matching the criteria
ReportDebugLog " CUSTOM Function - PATTERN_NAME - now resolving query -
Current
Page: #" & i
Set selectorInPopup = query.SelectSingle()
```

Wenn die Vorgangsart gefunden wird, ist die Variable nicht **NULL**, und wir überprüfen sie, um zu ermitteln, ob wir zur nächsten Seite navigieren müssen.

#### CodeSyntax

```
If selectorInPopup Is Nothing Then uriNextPage = "label=Navigation to Page
#" &
(i+1) & "; " & _ "popupId=1; tag=SPAN; crm.tag=TD; " & _
"crm.id=A_btfollowup:V_ProcType:T_PAGER:C_proctype:I_:K_pag_pg-" &
(i+1)
Set nextPageControl = CrmController().GetCrmControlByUri(uriNextPage)
If nextPageControl Is Nothing Then
ReportLog "FAILED", "PATTERN_NAME", "Operation Failed - Last page has been
reached" Exit For
Else
' Control not found - Let's navigate to the next page (if any)
CRM_WebControl_Click uriNextPage
If ReportStatus <>"OK" Then
Exit For
End If
End If
Else
...
End If
```

Der Code erstellt den URI der nächsten Seite und simuliert einen Mausklick, um dorthin zu navigieren.

Beim Erreichen der letzten Seite wird die Schleife unterbrochen, und das Testskript kann fehlschlagen, da keine Vorgangsart ausgewählt wurde.

## Zeilenselektion

Wenn das Zeilenselektor-UI-Element gefunden wird, wird die Variable **selectorInPopup** gesetzt. Die Implementierung verwendet die Methode **GetControlUri** zur Ermittlung des URI des Zeilenselektors und ruft die Funktion **CRM\_Table\_SelectRow** zum Ausführen der Selektion auf.

#### CodeSyntax

```
' Control found - let's click on the ROW_SELECTOR
CRM_Table_SelectRow selectorInPopup.GetControlUri()
If ReportStatus = "FAILED" Then
ReportDebugLog " CUSTOM Function - PATTERN_NAME - Operation Failed"
Else
ReportDebugLog " CUSTOM Function - PATTERN_NAME - Operation Succeeded"
End If
```

```
PATTERN_NAME_Impl = ReportStatus  
Exit For
```

Die Variable **PATTERN\_NAME\_Impl** gibt Feedback zum Status der Operation. Es ist der vom Ausgabeparameter der Komponente **CBTA\_CRM\_A\_INVOKE\_FUNCTION** empfangene Wert. Ergebnis: Die Vorgangsart ist ausgewählt. Die **Exit For**-Anweisung unterbricht die Schleife, um zu verhindern, dass die nächsten Seiten durchsucht werden.

### 3.4.4.5.1.3.6 Muster für SAP GUI: ME21N\_ExpandAllSections

Das Muster **ME21N\_ExpandAllSections** prüft den Status der Kopfabschnitte und klappt sie automatisch auf. Es stellt einen erkannten initialen Status wieder her, wenn eine Transaktion in einem anderen Status gestartet wird, abhängig z.B. von den Benutzereinstellungen, damit das Testskript in einem erwarteten Kontext fortgesetzt werden kann.

#### Beispiel

Dieses Problem tritt normalerweise bei Transaktion **ME21N** auf, bei der das Hauptbild aus mehreren Bereichen besteht. Der initiale Status der Bereiche ist beim Start unbekannt. Jeder Bereich kann auf- oder zugeklappt werden, je nachdem, wie die Transaktion während der vorherigen Sitzung beendet wurde. Das Testskript muss diesen Bereich aufklappen, bevor es fortfährt. Die folgende Grafik zeigt den Übersichtsbereich, einmal in zugeklapptem und einmal in aufgeklapptem Zustand.

Die kundeneigene Entwicklung dieses Musters prüft den Zustand jedes Bereichs und klappt ihn ggf. auf.

#### i Hinweis

Diese Situation kann in der Regel mithilfe von Standardkomponenten wie **CBTA\_A\_SETCONDITION** gelöst werden, jedoch nicht mit Transaktion **ME21N**, die eine spezifische Handhabung dieser Bereiche pflegt. Sie verwendet mehrere Teilbilder, um den Inhalt jedes Bereichs anzuzeigen, daher ist eine benutzerdefinierte Funktion erforderlich.

#### Signatur des **SelectTransactionType**-Musters

- Eingabeparameter: keine
- Ausgabeparameter: keine

Es wird angenommen, dass die benutzerdefinierte Funktion zuvor mithilfe der Code-Assistent-Funktion von Runtime Library Manager angelegt wurde.

Definition des Testskripts: Da die benutzerdefinierte Funktion keine Eingabeparameter hat, kann sie mit der Standardkomponente **CBTA\_GUI\_A\_EXECUTESTATEMENT** aufgerufen werden:

Eingabeparameter: Die Komponente zum Aufrufen der benutzerdefinierten Funktion erwartet mindestens den **Library**-Parameter (Name der VB-Skriptdatei) und den **Statement**-Parameter (auszuführende Anweisung). Falls die benutzerdefinierte Funktion keine Eingabeparameter erwartet, ist die Anweisung einfach.

Ausführungsreport: Die von der benutzerdefinierten Funktion ausgeführten Aktionen werden in dem Ausführungsreport angezeigt.

Die Implementierung besteht aus Teilprozeduren. Die erste Teilprozedur ist eine Fassade der Implementierung. Die zweite Methode führt den Job aus.

Das Fragment **PATTERN\_NAME** wird bei Verwendung der Code-Assistent-Funktion von Runtime Library Manager automatisch durch den Namen der finalen kundeneigenen Entwicklung ersetzt.

### CodeSyntax

```
Sub PATTERN_NAME () On Error Resume Next ' Important - Exception Handling
for
SAPGUI - Do not change it! EventComponentBegin()
  If ConditionsManager().CheckConditions() Then
  If ConditionsManager().CheckConditions() Then
  End If
  EventComponentEnd()
End Sub
Sub PATTERN_NAME_Impl () 'internal
ReportDebugLog "CUSTOM Function - PATTERN_NAME"
...
End Sub
```

Die Teilprozedur ruft die Funktion **Expand\_Impl** für jeden zuklappbaren Abschnitt in der Transaktion **ME21N** auf.

### CodeSyntax

```
Uri = "label=Expand Header; type=GuiButton;
id=wnd[0]/usr/subSUB0:SAPLMEGUI:0016/subSUB1:SAPLMEVIEWS:1100/
subSUB1:SAPLMEVIEWS:4000/btnDYN_4000-BUTTON" Expand_Impl Uri
Uri = "label=Expand Item Overview; type=GuiButton;
id=wnd[0]/usr/subSUB0:SAPLMEGUI:0016/subSUB2:SAPLMEVIEWS:1100/
subSUB1:SAPLMEVIEWS:4001/btnDYN_4000-BUTTON"
```

### CodeSyntax

```
Expand_Impl Uri Uri = "label=Expand Item Detail; type=GuiButton;
id=wnd[0]/usr/subSUB0:SAPLMEGUI:0016/subSUB3:SAPLMEVIEWS:1100/
subSUB1:SAPLMEVIEWS:4002/btnDYN_4000- BUTTON" Expand_Impl Uri
```

Die **Expand\_Impl**-Prozedur wird ebenfalls mit dem Mustercode ausgeliefert. Sie sucht über den URI nach der Drucktaste zum Zuklappen, indem sie die Methode **FindGuiControlByUri** aufruft. Der Aufruf gibt einen Verweis auf ein GuiScripting-Objekt zurück (GuiButton in diesem Fall). Der Code ermittelt anschließend anhand der Eigenschaft **IconName**, ob der Bereich auf- oder zugeklappt ist.

## i Hinweis

Die Logik ist spezifisch für die Transaktion **ME21N**. Normalerweise wird über „IconName“ nicht ermittelt, ob ein Bereich zugeklappt ist. Es ist jedoch die einzige Information, die in der Transaktion **ME21N** verfügbar ist.

## Code Syntax

```
Sub Expand_Impl ( Uri ) 'internal
    Set GS_guiControl = Nothing On Error Resume Next
    ' Switching ON a local error handling
    Set GS_guiControl = GuiScripting().FindGuiControlByUri(Uri)
    On Error Goto 0 ' Switching OFF the local error
    handling
    If GS_guiControl Is Nothing Then ' Control not found ReportLog
    "INFO", "Expand", "Operation skipped - the button does not exists"
    Else
    If GS_guiControl.IconName = "DAARSO" Then ReportLog "INFO",
    "Expand", "Operation skipped - the section is already expanded" Else
    GS_guiControl.Press ' Control has been found - let's click on
    it
    ReportLog "INFO", "Expand", "Operation has been performed" _
    & vbCrLf & "Target: " & GetMeaningfulName(GS_guiControl)
    End If
    End If
End Sub
```

### 3.4.4.5.1.3.7 Muster für SAP GUI: ME51N\_ExpandAllSections

**ME51N\_ExpandAllSections** ist analog zu **ME21N\_ExpandAllSections**. Der einzige Unterschied besteht darin, dass es über **ME51N** komprimierbare Bereiche aufklappt. Weitere Informationen hierzu finden Sie im vorherigen Abschnitt [Muster für SAP GUI: ME21N\\_ExpandAllSections \[Seite 85\]](#).

### 3.4.4.5.1.3.8 Muster für SAP GUI: WaitForCondition

Das Muster **WaitForCondition** sorgt dafür, dass ein Testskript auf den Abschluss des Hintergrundprozesses wartet, bevor es fortgesetzt wird. Mit SAP GUI können einige Transaktionen asynchrone Hintergrundprozesse auf dem Server starten, sodass das Operationsergebnis nicht automatisch auf der Benutzungsoberfläche angezeigt werden kann. Der Endbenutzer muss Aktionen (z.B. Drücken der **ENTER**-Taste) ausführen, um die Benutzungsoberfläche mit dem Serverstatus zu aktualisieren.

Das Muster führt eine Aktion mehrmals aus, bis ein Text in der SAP-GUI-Benutzungsoberfläche angezeigt wird.

Signatur des Musters **WaitForCondition**

- Eingabeparameter
  - **Uri**: URI zur Identifikation des SAP-GUI-Steuerelements, zeigt das Vorgangsergebnis an  
Es wird der Text angezeigt, auf den die Wait-Bedingung prüft. Wenn es leer ist, wird standardmäßig der URI der Statusleiste verwendet. Standardwert: **label=Main Window; type=GuiMainWindow; id=/app/con[0]/ses[0]/ wnd[0]**

- **TextToCheck**: Text oder Teil des Texts, der von dem Steuerelement angezeigt wird, wenn der Hintergrundprozess noch ausgeführt wird  
Standardwert: **Scheduled**
- **WaitTime**: Wartezeit in Millisekunden zwischen Versuchen  
Standardwert: **1000** (eine Sekunde)
- **MaxAttempts**: maximale Anzahl an Versuchen  
Wenn die maximale Anzahl erreicht ist, stoppt die Implementierung die Prüfung und wird mit **NOT\_DONE** als Ausgabeparameter beendet. Standard: **30**.
- : nicht verwendet
- Ausgabeparameter:
  - **Output**: Status **NOT\_DONE** oder **DONE**

Das Muster hat 5 Parameter, es kann also von der Standardkomponente **CBTA\_GUI\_A\_INVOKE\_FUNCTION** aufgerufen werden.

Die Implementierung besteht wie die meisten Muster aus zwei Funktionen. Die erste ist eine Fassade der Implementierung und enthält Anweisungen zur Behandlung von Ausnahmefehlern. Die zweite Methode führt den Job aus.

## Initialisierungsphase

Die ersten Anweisungen validieren die Eingabeparameter und legen ggf. die Standardwerte fest.

### Code syntax

```
If IsNull(Uri) Then
    Uri = "label=Status Bar; type=GuiStatusbar; id=/app/con[0]/ses[0]/wnd[0]/sbar"
End If
If IsNull(TextToCheck) Then
    TextToCheck = "Scheduled"
End If
If IsNull(WaitTime) Then
    WaitTime = "1000" '1 second
End If
If IsNull(MaxAttempts) Then    MaxAttempts = "30" '30 * 1000 ==> 30 seconds
    max!
End If
```

Der Code schreibt Informationen zur Fehlerbehebung in die Traces und initialisiert das Ergebnis in **NOT\_DONE**.

### Code syntax

```
ReportDebugLog "CUSTOM Function - PATTERN_NAME" & _
vbCrLf & " - Uri: " & Uri & _                vbCrLf & " -
TextToCheck: " & TextToCheck & _            vbCrLf & "
- WaitTime " & WaitTime & _                vbCrLf & " -
MaxAttempts: " & MaxAttempts
PATTERN_NAME_Impl = "NOT_DONE"
```



## Implementierung

Eine Schleife führt unter Verwendung der Schlüsselwörter **For/Next** mehrmals eine Prüfung des angezeigten Texts durch, bis der Grenzwert erreicht ist, der über den Eingabeparameter **maxAttempts** definiert wird.

Die Methode **GetPropertyValue** ruft den Text ab, der von dem Steuerelement angezeigt wird, auf das der URI verweist. Dann prüft sie, ob dieser Text den Wert im **TextToCheck**-Eingabeparameter enthält: Ist dies der Fall, wird für die definierte Zeitspanne gewartet und **ENTER** ausgeführt, um das Bild zu aktualisieren. Ist dies nicht der Fall, wird davon ausgegangen, dass der Hintergrundprozess abgeschlossen ist, und die Iteration wird beendet.

### Code syntax

```
Dim i, statusBarUri, mainWindowUri, statusBarText, max mainWindowUri =
"label=Main Window; type=GuiMainWindow; id=/app/con[0]/ses[0]/ wnd[0]"
max = ToInt(MaxAttempts) For i=1 To max ReportDebugLog
"PATTERN_NAME - Now getting the text of the Control" statusBarText
= GetPropertyValue ( Uri, "Text", Null ) ReportDebugLog
"PATTERN_NAME - Control Text: " & statusBarText
If InStr(statusBarText, TextToCheck) Then
' Let's wait ReportLog "INFO", "PATTERN_NAME", "Waiting for
background job to complete - attempt #" & i GS_Wait WaitTime, Null
'Let's Press Enter to refresh the UI
GS_PressKey mainWindowUri, "Enter", Null
Else
' Text not visible anymore
ReportLog "DONE", "PATTERN_NAME", "Background job has completed"
PATTERN_NAME_Impl = "DONE"
'Let's stop the loop
Exit For
End If
Next
```

### 3.4.4.5.1.3.9 Speicherort der Muster

Die von SAP vordefinierten Muster sind im *MIME-Repository* (SE80) des SAP-Solution-Manager-Systems unter **SAP > PUBLIC > CBTA > PATTERNS** abgelegt. Muster werden je nach Art in verschiedenen Unterordnern gesichert.

### 3.4.4.5.1.3.10 Runtime Library Manager

Mit dem Runtime Library Manager wird die Laufzeitbibliothek angepasst. Er bietet folgende Funktionen:

- Öffnen der Laufzeitbibliothek zum Bearbeiten
- Manuelles Schreiben von kundeneigenen Entwicklungen
- Schreiben kundeneigener Entwicklungen unter Verwendung der Muster mit dem Code-Assistenten
- Testen der kundeneigenen Entwicklung vor dem Übermitteln

- Übermitteln von Änderungen, um sie für alle Tester verfügbar zu machen

## Code-Assistent

Wenn Sie den Code-Assistenten verwenden, können Sie mithilfe einer Guided Procedure ein Muster auswählen und die kundeneigene Entwicklung generieren.

### Auswählen eines Musters

Wählen Sie ein Muster aus, bevor Sie eine kundeneigene Funktion anlegen. Wenn Sie anstelle der manuellen Auswahl den Assistenten verwenden, steht die generierte kundeneigene Entwicklung zur Verwendung bereit. Wählen Sie entsprechend Ihren Anforderungen oder dem Problem, das Sie beheben möchten, ein Muster aus. Hauptkriterium für die Musterauswahl ist die UI-Technologie der getesteten Anwendung.

### Generieren einer kundeneigenen Funktion

Der Code-Assistent beendet die Generierung der kundeneigenen Entwicklung und verwendet den vom Testingenieur angegebenen Funktionsnamen.

Das VB-Skript mit der kundeneigenen Entwicklung wird lokal im Dateisystem an dem vom Testingenieur angegebenen Speicherort abgelegt. Passen Sie das VB-Skript an Ihre Anforderungen an und testen Sie es, bevor Sie die Änderungen übermitteln.

Weitere Informationen finden Sie unter [CBTA um kundeneigene Entwicklungen erweitern \[Seite 67\]](#).

## 3.4.4.5.1.4 Weitere Informationen

- SAP-Hinweis [1778899](#) (Sammelhinweis der komponentenbasierten Testautomatisierung, einschließlich Release-Dokumentation zu den SPs)
- SAP-Hinweis [1912801](#) (CBTA - RTL Manager: Customizing der Laufzeitbibliothek)
- Schnittstellen für die Skripterstellung für Objekte von Microsoft  
Das **MSHTML**-API besteht aus einer Reihe von Microsoft-COM-Schnittstellen, die Zugriff auf den von Microsoft Internet Explorer angezeigten HTML-Content bieten. Die offizielle Dokumentation ist online auf MSDN unter [https://msdn.microsoft.com/en-us/library/aa741322\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/aa741322(v=vs.85).aspx) verfügbar.

Schnittstellenname	Methodenname	Verknüpfung
<b>IHTML</b> Element	<b>click</b>	<a href="https://msdn.microsoft.com/en-us/library/%20aa752277(v=vs.85).aspx">https://msdn.microsoft.com/en-us/library/%20aa752277(v=vs.85).aspx</a>
<b>IHTML</b> Element	<b>all</b>	<a href="https://msdn.microsoft.com/en-us/library/%20aa752281(v=vs.85).aspx">https://msdn.microsoft.com/en-us/library/%20aa752281(v=vs.85).aspx</a>

Schnittstellenname	Methodenname	Verknüpfung
<b>IHTMLCollection</b>	<b>tags</b>	<a href="https://msdn.microsoft.com/en-us/library/%20aa703932(v=vs.85).aspx">https://msdn.microsoft.com/en-us/library/%20aa703932(v=vs.85).aspx</a> ➔
<b>IHTMLFormElement</b>	<b>fireEvent</b>	<a href="https://msdn.microsoft.com/en-us/library/%20aa703947(v=vs.85).aspx">https://msdn.microsoft.com/en-us/library/%20aa703947(v=vs.85).aspx</a> ➔
<b>IHTMLFormElement</b>	<b>setActive</b>	<a href="https://msdn.microsoft.com/en-us/library/%20aa703969(v=vs.85).aspx">https://msdn.microsoft.com/en-us/library/%20aa703969(v=vs.85).aspx</a> ➔

### 3.4.4.5.2 Query-API für Tests des dynamischen Szenarios

Das Testen einiger Szenarios kann nicht gemäß dem üblichen Ansatz, der auf Komponenten basiert, automatisiert werden.

Wie bei den meisten Webanwendungen, die SAP ausliefert, wird der Content im Microsoft Internet Explorer auf HTML-Seiten angezeigt. Der HTML-Content kann sehr komplex und dynamisch sein. Er wird auf dem Server von der Präsentationsschicht generiert, die je nach UI-Technologie, die zur Entwicklung der Anwendung verwendet wurde, variieren kann.

UI-Haupttechnologien bei SAP

- SAPUI5 / FIORI
- WebCUIF  
Schicht, die den HTML-Content für SAP-CRM-Web-UI-Anwendungen generiert.
- Unified Rendering Light Speed (LS)  
Schicht, die den HTML-Content für SAP-Anwendungen wie den folgenden generiert:
  - Web-Dynpro-Anwendung (ABAP)
  - Web-Dynpro-Anwendungen (Java – neueste Versionen)
  - SAP GUI für HTML– Im Browser angezeigter SAP-GUI-Content
- BSP
- Auf Business Server Pages basierende Anwendungen.
- Java Web Dynpro  
Schicht, die zuvor von Java-Web-Dynpro-Anwendungen verwendet wurde.

Nicht-SAP-UI-Technologien

- Plain-HTML-Seiten (unterstützt in der komponentenbasierten Testautomatisierung)  
Content, der über reguläre HTML-Tags generiert wurde (ohne oder mit wenigen Scripting-Funktionen).
- Adobe Flash (in der komponentenbasierten Testautomatisierung nicht unterstützt)  
Schicht, die von Adobe-Flash-Anwendungen unterstützt und mit dem Adobe Flex Software Development Kit generiert wurde.
- Java-Applets (in der komponentenbasierten Testautomatisierung nicht unterstützt)

Anwendung, die in Java geschrieben wurde. Java-Applets können auf einer Webseite oder in einem neuen Anwendungsfenster angezeigt werden.

Jede UI-Technologie hat eine eigene Logik und generiert HTML-Content, der sich von anderem HTML-Content komplett unterscheiden kann. Dies ist für die Testautomatisierung eine Herausforderung und kann eigene Funktionen erfordern. In eigenen Funktionen kann der Testingenieur den von der Anwendung für ein UI-Element angezeigten HTML-Content suchen, indem er die native COM-Schnittstellen (von Microsoft) verwendet, die durch das MS-Internet-Explorer-Objekt bereitgestellt werden. Dieser Ansatz ist jedoch komplex und fehleranfällig. Anstelle der nativen COM-Schnittstellen können Querys über ein API verwendet werden.

## Weitere Informationen

[Struktur des Query-API \[Seite 92\]](#)

[Beispiele zur Verwendung des Query-API \[Seite 96\]](#)

[Technische Informationen für das Query-API \[Seite 109\]](#)

### 3.4.4.5.2.1 Struktur des Query-API

Das Query-API blendet die HTML-Komplexität aus und sucht nach UI-Elementen mit Filtern und Bedingungen.

## Abfragearten

Unabhängig von der zugrunde liegenden Technologie ist der Abfrageansatz ähnlich, einige zusätzliche Funktionen können jedoch für bestimmte UI-Technologien zur Verfügung stehen. Die Abfragen sind anwendungsspezifisch. In der folgenden Tabelle sind die Abfragearten und ihre UI-Technologien dargestellt.

Abfrageart	UI-Technologie	Anwendungen
CRM-Abfragen	WebCUIF	SAP-CRM-Anwendungen
LS-Abfragen	Unified Rendering Light Speed (LS)	Web-Dynpro-Anwendung (ABAP) Web-Dynpro-Anwendungen (Java) Web GUI – SAP-GUI-Content angezeigt in MS Internet Explorer
Webabfragen	Unified Rendering Light Speed (LS)	Webanwendungen
JWDP-Abfragen	Java Web Dynpro (alte Releases)	Java-Web-Dynpro-Anwendungen

Beispielabfrage: „Suchen Sie nach allen UI-Elementen in einer HTML-Tabelle, in denen die *Partnerfunktion* dem Wert *Vertriebsbeauftragter* und die *Partner-ID* dem Wert *244* entspricht.“

## Anwendungssteuerungen

Anwendungs-Controller sind der Haupteinstiegspunkt in die Anwendung, die getestet wird, und bieten Zugriff auf den HTML-Content, der in der aktuellen Sitzung angezeigt wird.

Mehrere Anwendungs-Controller werden bereitgestellt, jeder für einen spezifischen Anwendungsfall, abhängig von der zugrunde liegenden UI-Technologie:

- Light Speed Controller: Controller für Anwendungen, die auf dem Unified Rendering Light Speed Framework basieren (wie etwa Web-Dynpro-ABAP-Anwendungen).
- CRM-Controller: Controller für SAP-CRM-Anwendungen, die auf der WebCUIF-UI-Technologie basieren.
- Web-Controller: Controller für Webanwendungen, deren zugrunde liegende UI-Technologie unbekannt ist, oder Webanwendungen, die keinen spezifischen Controller haben (wie BSP-Anwendungen).
- Java-Web-Dynpro-Controller: Controller für Anwendungen, die auf vorherigen Releases von Java Web Dynpro basieren.

Die Controller-Schnittstellen sind die Einstiegspunkte in die Abfrageschnittstellen. Jeder Controller hat mindestens eine Methode zum Anlegen von Abfrageobjekten.

Anlegen eines Webabfrageobjekts: Das folgende Skript legt ein Abfrageobjekt zur Suche nach Web-Controls an:

### CodeSyntax

```
Dim controller, query
Set controller = WebController()
Set query = controller.CreateQuery()
```

Anlegen eines WebCUIF-Abfrageobjekts: Das folgende Skript legt ein Abfrageobjekt zur Suche nach WebCUIF-Controls (für SAP-CRM-Anwendungen) unter Verwendung des **theCrmController** an.

### CodeSyntax

```
Dim controller, query
Set controller = CrmController()
Set query = controller.CreateQuery()
```

Anlegen eines Light-Speed-Abfrageobjekts: Das folgende Skript legt ein Abfrageobjekt zur Suche nach Light-Speed-Controls an. Der Code ist ähnlich; der einzige Unterschied besteht darin, dass anstelle des Web-Controllers der Light-Speed-Controller verwendet wird.

### CodeSyntax

```
Dim controller, query
Set controller = LsController()
Set query = controller.CreateQuery()
```

Anlegen eines Webabfrageobjekts aus einem beliebigen Controller: Im folgenden Beispiel wird explizit eine Webabfrage anstelle einer Light-Speed-Abfrage angefordert, indem die Methode `CreateWebQuery` aufgerufen wird.

### CodeSyntax

```
Dim controller, query
```

```
Set controller = LsController()
Set query = controller.CreateWebQuery()
```

Abfrageerstellung mit Optionen: Die Abfrageobjekte können mit Optionen angelegt werden, die spezifisch für die zugrunde liegende UI-Technologie sind. Dieser zusätzliche Parameter ist optional. Im folgenden Beispiel wird die Erstellung einer Light-Speed-Abfrage mit der Option **/ls** angezeigt, die die Optimierung aktiviert. Weitere Informationen dazu finden Sie in den folgenden Abschnitten.

#### Code syntax

```
Dim query
Set query = LsController().CreateQuery("/ls")
```

## Web-Controls

### Light-Speed-Web-Controls

Anwendungen, die auf dem Light Speed Framework basieren, generieren Controls, in denen mindestens eines der HTML-Elemente Metainformationen durch zusätzliche HTML-Attribute bereitstellt:

- **ct**: Control-Typ
- **lsdata**: Light-Speed-Daten; eine Sammlung von Schlüssel-/Wert-Paaren, die das Control und dessen aktuellen Status beschreibt.

Beispiel für Light-Speed-Attribute für ein Control vom Typ Ankreuzfeld:

- **Checked**: Status des Ankreuzfelds
- **Enabled**: falsch, wenn das Ankreuzfeld deaktiviert ist (der Status ist nicht relevant)
- **Readonly**: falsch, wenn der Status nicht geändert werden kann
- **Text**: der neben dem Ankreuzfeld angezeigte Text
- **Tooltip**: der Text in der Quick-Info

Die Light-Speed-Abfrageschnittstellen sind speziell zur Verwendung dieser Metainformationen geeignet, Sie können also mit den Light-Speed-Daten nach einem Control suchen.

### CRM-Web-Controls

CRM-Anwendungen generieren ihren Content mit der WebCUIF-UI-Technologie, bei der jedes Control ein ausgeblendetes **SPAN**-HTML-Element hat, das über sein ID-Attribut Metainformationen bereitstellt. Eine solche ID kann beispielsweise folgendermaßen aussehen:

**A\_btpartner:V\_Partner:T\_ROW\_SELECTOR:C\_btpartner:I\_partner\_fct:R\_1**

Die ID besteht aus den folgenden Elementen:

- **A\_**: Anwendungsname
- **V\_**: Sichtname
- **T\_**: Control-Typ
- **C\_**: Kontext
- **I\_**: Schnittstelle, die ein Feldname oder ein Attributname im **C\_**-Kontext ist

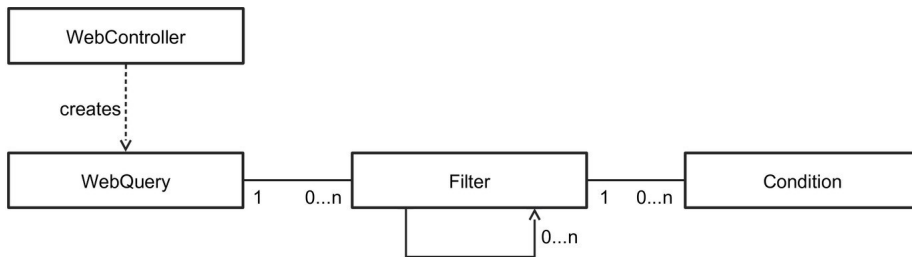
- **R\_**: Zeilennummer; beispielsweise, wenn das Control in einen Tabellencontainer eingebettet ist
- **K\_**: Schlüssel; beispielsweise kann ein Element in einer Dropdown-Liste per Schlüssel ausgewählt werden

Die CRM-Abfrageschnittstellen sind speziell zur Verwendung dieser Metainformationen geeignet, Sie können also mit diesen Fragmenten nach einem Control suchen.

## Abfragen, Filter und Bedingungen

Das Query-API sucht in HTML-Seiten nach Controls. Die mit diesem API angelegten Objekte haben folgende Merkmale:

- Abfragen werden mit einem Anwendungs-Controller angelegt. Wählen Sie den entsprechenden für die zu testende UI-Technologie aus.
- Die Abfrage kann einen oder mehrere Filter haben.
- Jeder Filter kann eine oder mehrere Bedingungen aufweisen.
- Jeder Filter kann Unterfilter enthalten.
- Jede Bedingung kann die Werte der UI-Elementattribute und -eigenschaften prüfen.



Der Lebenszyklus von Abfragen ist beschränkt. Sie können nur in demselben Kontext verwendet werden, in dem sie angelegt werden. Eine Abfrage kann mehrmals verwendet werden, jedoch nur für den aktuellen Satz an HTML-Dokumenten; deshalb müssen Sie eine neue Abfrage anlegen, wenn Sie zu einer anderen HTML-Seite navigieren, d.h. nach jedem HTTP-Server-Roundtrip.

## Schnittstellenübersicht

### Controller-Schnittstelle

Der Controller ist der Einstiegspunkt in das Query-API. Sie können damit Abfragen anlegen.

Mit der **CreateQuery**-Methode können Sie Objekte für die Abfrage anlegen.

Für jede UI-Technologie ist ein Controller vorhanden. Wählen Sie den Controller für das Element aus, nach dem Sie suchen.

Weitere Informationen finden Sie unter [Technische Informationen für das Query-API \[Seite 109\]](#).

### Abfrageschnittstelle

Die wichtigsten Methoden der Abfrageschnittstelle sind:

- **SetFilter**: entfernt alle Filter und definiert den ersten

- **AddFilter**: fügt einen zusätzlichen Filter hinzu
- **Select**: führt die Abfrage aus und gibt eine Reihe von Controls zurück
- **SelectSingle**: führt die Abfrage aus und gibt das erste Control zurück, das den Kriterien entspricht

Die Abfrageeigenschaft **ParentControlUri** definiert den Umfang, in dem Sie nach Controls suchen.

Sie können die Abfrageschnittstellen angeben, um die Metainformationen für die zugrunde liegende UI-Technologie zu nutzen.

Weitere Informationen finden Sie unter [Technische Informationen für das Query-API \[Seite 109\]](#).

## Filter- und Bedingungsschnittstelle

Der Filter enthält die Kriterien, die die Controls erfüllen müssen. Methoden:

- **SetCondition**: entfernt alle Bedingungen und definiert die erste
- **AddCondition**: fügt eine zusätzliche Bedingung hinzu
- **SetFilter**: entfernt alle Unterfilter und definiert den ersten
- **AddFilter**: fügt einen zusätzlichen Unterfilter hinzu

Eine Bedingung enthält Informationen zu den Kriterien, die zuvor mit der Methode **SetCondition** definiert wurden.

Eigenschaften:

- **AttributeName**: Name des Attributs (oder der Eigenschaft), das bzw. die von dieser Bedingung verwendet wird
- **BooleanOperator**: Operator, der zur Bewertung dieser Bedingung verwendet wird
- **Value**: erwarteter Wert des Attributs oder der Eigenschaft
- **Option**: Bewertungsoptionen für Bedingung

Weitere Informationen finden Sie unter [Technische Informationen für das Query-API \[Seite 109\]](#).

## Web-Control-Schnittstellen

Eine Abfrage durchsucht den HTML-Content nach Controls und gibt eine oder mehrere zurück. Diese Controls implementieren Schnittstellen, die Zugriff auf die HTML-Attribute und -Eigenschaften der zugrunde liegenden HTML-Elemente bieten.

Ein einzelnes Control kann eine Kombination mehrerer HTML-Elemente sein, und das API ist speziell dazu geeignet, über zusätzliche Schnittstellen wie etwa folgende Zugriff auf die Metainformationen zu geben:

Weitere Informationen finden Sie unter [Technische Informationen für das Query-API \[Seite 109\]](#).

### 3.4.4.5.2 Beispiele zur Verwendung des Query-API



## Beispiel: Anlegen von Ansprechpartnern

Zum Eingeben der Kontaktdaten werden drei Sichten verwendet:

- `bp_cont/ContactDetails`
- `bp_cont/SalesEmployee`
- `bp_addr/StandardAddress`

Das Query-API kann alle Eingabefelder in der Sicht `SalesEmployee` abrufen.

### Nach Web-Controls suchen

Im folgenden Skript wird gezeigt, wie eine Abfrage angelegt und ausgeführt wird:

```
Set query = CrmController().CreateQuery() ← query creation
Set filter = query.SetFilter() ← filter creation
filter.AddCondition "crm.view", "=", "SalesEmployee"
filter.AddCondition "crm.type", "=", "inputField" ← adding conditions

Set collection = query.Select() ← execution of query
If collection Is Nothing Then
  MsgBox "Nothing has been found"
Else
  MsgBox "Element Found: " & collection.Count ← showing some results
End If
```

Das VB-Skript greift auf das HTML-Dokument zu, indem die Methode `CrmController` aufgerufen wird, und legt unter Verwendung der Methode `CreateQuery` eine Abfrage an. Die Abfrage ist leer. Geben Sie unter Verwendung von Filtern und Bedingungen die Kriterien an, um die betreffenden Controls zu identifizieren. In unserem Beispiel wird ein einzelner Filter mit zwei Bedingungen angegeben. Der Filter wird mit der Methode `SetFilter` angelegt, und die zwei Bedingungen werden mit der Methode `AddCondition` hinzugefügt.

Nachdem die Kriterien festgelegt wurden, kann die Abfrage mit den folgenden Methoden ausgeführt werden:

- `Select`: gibt eine Sammlung von CRM-Web-Controls zurück, die den Kriterien entsprechen.
- `SelectSingle`: gibt das erste CRM-Web-Control zurück, das den Kriterien entspricht, und keine Sammlung

In den nächsten Beispielabschnitten wird beschrieben, wie die Ergebnisse der Methoden `Select` und `SelectSingle` verwendet werden.

### Den Eingabefeldwert festlegen

Im ersten Beispiel wird erklärt, wie eine Abfrage angelegt und ausgeführt wird. Im zweiten Beispiel wird gezeigt, wie das Ergebnis verwendet und das abgerufene CRM-Web-Control-Objekt geändert wird.

#### Code syntax

```
Dim controller, query, filter, crmControl, controlUri
Set controller = CrmController()
Set query = controller.CreateQuery()
Set filter = query.SetFilter()
filter.AddCondition "crm.view", "=", "SalesEmployee"
filter.AddCondition "crm.type", "=", "inputField"
Set crmControl = query.SelectSingle()
If Not crmControl Is Nothing Then
  controlUri = crmControl.GetControlUri()
  controller.SetElementValue controlUri, "1234"
End If
```

Dieses Skript ruft die Methode `SelectSingle` auf, um ein einzelnes CRM-Web-Control abzurufen (deshalb muss keine Iteration durch eine Sammlung ausgeführt werden). Es prüft, ob die Variable `crmControl`

festgelegt ist, und fragt nach dem URI zur Identifikation des CRM-Web-Controls, indem die Methode `GetControlUri` aufgerufen wird. Dieser Aufruf gibt einen URI zurück, der denen entspricht, die von der Laufzeitbibliothek erwartet werden, wenn Sie Standardkomponenten verwenden.

In diesem Beispiel wird das CRM-Web-Control mit diesem URI identifiziert:

```
crm.area=WorkArea; tag=INPUT;

crm.id=A_bp_cont:V_SalesEmployee:T_inputField:C_salesemployee:I_struct.salesemployee
```

Mit diesem URI kann das Skript eine der vom CRM Web Controller API bereitgestellten Methoden aufrufen, um Aktionen auszuführen, die normalerweise mit Standardkomponenten ausgeführt werden. In dem Beispiel legt das Skript den Wert des Eingabefelds auf **1234** fest, indem die Methode `SetElementValue` aufgerufen wird.

## Beispiel: Anlegen von Terminen

Die bisher erläuterten Szenarios sind nicht dynamisch, deshalb ist das Query-API nicht erforderlich, da die Standardkomponenten normalerweise zur Automatisierung ausreichen. Im nächsten Beispiel wird erklärt, wie CRM-Web-Controls ausgewählt werden, indem die Werte im `INPUT`-HTML-Element, das im Browser angezeigt wird, gefiltert werden.

### Nach Web-Controls in einer Tabelle suchen

Ein dynamisches Szenario muss automatisiert werden, in dem das Skript die erste Zeile auswählt, in der die Partnerfunktion `Vertriebsbeauftragter` und die Partner-ID **244** entspricht.

Die Schwierigkeit besteht darin, dass die Bedingungen in Bezug auf `CELL`-Inhalte in derselben Zeile, aber einer anderen Spalte geprüft werden sollen und nicht in Bezug auf das gesuchte CRM-Web-Control. Diese Art von Abfrage kann sehr komplex sein, wenn die interne Hierarchie von HTML-Elementen durchsucht wird. Um dies zu vereinfachen, wurden die Filter und Bedingungen angegeben, um die Komplexität auszublenden und allgemeine Anwendungsfälle wie diesen zu behandeln.

Das VB-Skript-Coding für die Zeilenauswahl sieht wie folgt aus:

#### Code Syntax

```
Dim controller, query, filter, selector, controlUri
Set controller = CrmController()
Set query = controller.CreateQuery() Set filter = query.SetFilter()
filter.AddCondition "crm.application", "=", "btpartner"
filter.AddCondition "crm.view", "=", "Partner" filter.AddCondition
"crm.context", "=", "btpartner" filter.AddCondition "crm.type", "=",
"ROW_SELECTOR" filter.AddCondition "crm.column:btpartner.partner_no",
"=", "244" filter.AddCondition "crm.column~Partner Function", "=",
"Sales employee" Set selector = query.SelectSingle()
If Not selector Is Nothing Then
    If Not selector Is Nothing Then
        If Not selector Is Nothing Then
            End If
        End If
    End If
```

In diesem Skript werden die Bedingungsarten angezeigt, die von der Abfrage unterstützt werden. Hier prüfen die Bedingungen die Attribute der CRM-Web-Controls. Mit dem API können Sie die regulären HTML-Attribute und -Eigenschaften und einige CRM-spezifische Attribute prüfen, um die Fragmente der CRM-IDs zu überprüfen.

Der Object Spy (oder das Entwicklerwerkzeug des MS Internet Explorer) zeigt die CRM-ID des ersten Zeilenselektors an:

**A\_btpartner:V\_Partner:T\_ROW\_SELECTOR:C\_btpartner:I\_:R\_1**

Die Fragmente stellen Metainformationen zum CRM-Web-Control bereit:

- **A\_**: Anwendung
- **V\_**: Sichtname
- **T\_**: Art
- **C\_**: Kontextknoten
- **I\_**: Schnittstelle; für Eingabefelder der technische Name
- **R\_**: Zeilennummer

Die Bedingung kann den Wert dieser Fragmente anhand von CRM-Attributen wie den Folgenden prüfen:

- **crm.application**: prüft die Anwendung
- **crm.view**: prüft den Sichtnamen
- **crm.context**: prüft die UI-Elementart

Durch folgende Filterung wird sichergestellt, dass die Abfrage nur die Zeilenselektoren zurückgibt:

#### Code Syntax

```
filter.AddCondition "crm.application", "=",  
"btpartner" filter.AddCondition "crm.view", "=",  
"Partner" filter.AddCondition "crm.context", "=",  
"btpartner" filter.AddCondition "crm.context", "=",  
"btpartner"
```

Nun muss nur der zweite (die zweite sichtbare Zeile) ausgewählt werden. Wenn das Szenario nicht dynamisch wäre, könnte die zweite Zeile anhand einer Standardkomponente (beispielsweise SelectRow) und durch Angabe des URI direkt ausgewählt werden:

**A\_btpartner:V\_Partner:T\_ROW\_SELECTOR:C\_btpartner:I\_:R\_2**

Da das Szenario dynamisch ist, befindet sich die auszuwählende Zeile möglicherweise nicht immer an zweiter Stelle. Deshalb muss sie durch Prüfung des Werts der anderen Spalten gesucht werden. Wenn Sie nach Spaltenwerten filtern, sind, abhängig davon, ob Sie den Spaltentitel oder den technischen Namen der Spalte übertragen, zwei Syntaxen möglich.

Die Syntax kann **crm.column:<context>.<interface>** or **crm.column~<Column Title>** lauten, daher sind die folgenden beiden Arten der Filterung äquivalent:

#### Code Syntax

```
filter.AddCondition "crm.column:btpartner.partner_no", "=", "244"  
filter.AddCondition "crm.column:btpartner.partner_fct ", "=", "Sales  
employee"
```

...

#### Code Syntax

```
filter.AddCondition "crm.column~Partner ID", "=", "244"  
filter.AddCondition "crm.column~Partner Function", "=", "Sales  
employee"
```

```
employee"
```

In dem Beispiel werden beide Syntaxen verwendet:

#### Code syntax

```
filter.AddCondition "crm.column:btpartner.partner_no", "=", "244"  
filter.AddCondition "crm.column~Partner Function", "=", "Sales  
employee"
```

## Tipps und Tricks

### Durch eine Reihe von Web-Controls iterieren

Im folgenden Code wird gezeigt, wie durch die Reihe von CRM-Web-Controls iteriert wird, die von der Abfrage abgerufen werden. Die Sammlung implementiert die in [Technische Informationen für das Query-API \[Seite 109\]](#) beschriebene Web-Control-Sammlungsschnittstelle.

#### Code syntax

```
Dim collection  
Set collection = query.Select()  
If collection Is Nothing Then  
MsgBox "Nothing has been found"  
Else  
MsgBox "Element Found: " & collection.Count  
Dim childControl  
for i=0 To collection.Count-1 Set  
childControl = collection.ControlAt(i)  
MsgBox "Uri: " & childControl.GetControlUri()  
Next  
End If
```

#### i Hinweis

Die **For Each**-Anweisung des VB-Skripts wird nicht unterstützt.

### Informationen für Folgekomponenten bereitstellen

Üblicherweise ist nur ein Teil des getesteten Szenarios dynamisch und macht kundeneigene Codierung und Abfragen erforderlich. Dann kann es nützlich sein, den URI der abgerufenen CRM-Web-Controls für Folgekomponenten verfügbar zu machen und das Szenario mit Standardkomponenten fortzusetzen. Dies erfolgt normalerweise mit der Methode SetToken, wie im folgenden Beispiel dargestellt:

#### Code syntax

```
Dim crmControl, crmControlUri  
Set crmControl = query.SelectSingle()  
If crmControl Is Nothing Then CBTA.Log "Unexpected  
Situation - Control not found" Else  
crmControlUri = crmControl.GetControlUri() CBTA.Log  
"Uri: " & crmControlUri ExecutionContext.SetToken  
"rowSelector", crmControlUri  
End If
```

Folgekomponenten können den im Ausführungskontext gespeicherten Wert anhand der Syntax `%token%` wiederverwenden. In diesem Beispiel können die Folgekomponenten `%rowSelector%` verwenden, um Vorgänge für den `ROW_SELECTOR` auszuführen, der von der Abfrage ausgewählt wurde.

### Informationen in den Ausführungsreport schreiben

Das CBTA-Objekt stellt folgende Methoden bereit:

- **CBTA.Log**: führt die Fehlerbehebung in komplexen Szenarios durch  
**Öffentliches Unterprotokoll (Meldung)**
- **CBTA.Report**: gibt dem Tester Feedback aus dem Ausführungsreport  
**Öffentlicher Subreport (Schweregrad, Thema, Meldung, Optionen)**

### Den Umfang einer Abfrage reduzieren

Standardmäßig suchen die CRM-Abfragen im Arbeitsbereich **FRAME** des Browser-Hauptfensters automatisch nach CRM-Web-Controls. Die Abfrage analysiert den Content des HTML-Dokument-Bodys und prüft die Filter und Bedingungen für alle CRM-Web-Controls, die gefunden wurden.

Aus Performancegründen können Sie den Umfang der Abfrage reduzieren und angeben, wo die Suche begonnen werden soll. Wenn Sie beispielsweise nach CRM-Web-Controls vom Typ `ROW_SELECTOR` suchen, geben Sie unter Verwendung der Eigenschaft `ParentControlUri` der Abfrage zuerst das übergeordnete Container-HTML-Tabellenelement an. Beispiel:

#### Code syntax

```
'-----  
' Query Example with Restricted Scope  
'-----  
Dim query, filter, subFilter Set query =  
CrmController().CreateQuery() query.ParentControlUri =  
"tag=DIV; crm.id=A_btpartner:V_Partner:T_cellerator:C_btpartner:I_"
```

#### i Hinweis

Der URI sollte vom Object Spy ermittelt werden, doch die HTML-Elementtabelle kann, je nach Stil, nicht immer ausgewählt werden.

Die Alternative zum Object Spy besteht darin, mit dem Entwicklerwerkzeug des MS Internet Explorer nach der CRM-ID des UI-Elements `cellerator` zu suchen.

### Nach Web-Controls in einem Popup-Fenster suchen

Standardmäßig suchen die Abfragen im Arbeitsbereich **FRAME** des Browser-Hauptfensters nach CRM-Web-Controls. Dieses Standardverhalten können Sie mit der Eigenschaft `ParentControlUri` ändern.

Beispiel-VB-Skript zur Suche nach einem Control in einem modalen Popup-Fenster:

#### Code syntax

```
Dim controller, query, filter  
Set controller = CrmController()  
Set query =  
controller.CreateQuery()  
query.ParentControlUri = "popupId=1; crm.area=WorkArea;  
tag=body" Set filter = query.SetFilter() filter.AddCondition
```

```

"crm.type", "=", "ROW_SELECTOR" filter.AddCondition
"crm.context", "=", "proctype" filter.AddCondition
"crm.context", "=", "proctype"
filter.AddCondition "crm.column:proctype.proc_type_descr_20", "=",
"Meeting" Dim selectorInPopup, selectorUri Set selectorInPopup =
query.SelectSingle() selectorInPopup.HighLight "red" selectorUri =
selectorInPopup.GetControlUri() controller.SearchResult_SelectRow
selectorUri

```

Dieses Beispiel umfasst keine neuen Inhalte in Bezug auf die Auswahl von ROW\_SELECTOR. Das Skript verwendet einige Bedingungen, um den Vorgang mit Typ **1001** und Typbeschreibung **Besprechung** auszuwählen. Der Unterschied besteht darin, dass nun die Suche im Arbeitsbereich **FRAME** des Popups angegeben wird, indem die Eigenschaft **ParentControlUri** festgelegt wird:

```
query.ParentControlUri = "popupId=1; crm.area=WorkArea; tag=body"
```

In diesem Beispiel wäre es möglich gewesen, den Umfang der Abfrage weiter einzuschränken, indem der URI des **TABLE**-HTML-Elements wie folgt angegeben wird:

```

query.ParentControlUri = "popupId=1;
crm.id=A_btfollowup:V_ProcType:T_cellerator:C_proctype:I_"

```

### Filter für komplexe Abfragen kombinieren

Bisher wird in allen Beispielen nur ein Filter mit mehreren Bedingungen verwendet. Die Bedingungen werden nacheinander ausgewertet und die CRM-Web-UI-Elemente werden ausgeschlossen, sobald eine Bedingung nicht erfüllt wird. Die Bedingungen sind durch einen logischen **AND**-Operator verbunden.

Im vorherigen Beispiel wurde die erste Zeile mit den folgenden Bedingungen ausgewählt: Partnerfunktion entspricht „Vertriebsbeauftragter“ UND Partner-ID entspricht „244“.

Anstatt nach dem **ROW\_SELECTOR** zu suchen, könnten wir Controls abrufen, die entweder Ankreuzfelder oder Eingabefelder sind, die mit einer **F4**-Hilfe verknüpft sind, indem, wie im folgenden Beispiel, mehrere Filter kombiniert werden:

### Code syntax

```

Dim query, filter, subFilter, collection, childControl
Set query = CrmController().CreateQuery()
query.ParentControlUri = "tag=DIV;
crm.id=A_btpartner:V_Partner:T_cellerator:C_btpartner:I_" Set filter =
query.SetFilter() filter.AddCondition "crm.application", "=",
"btpartner" filter.AddCondition "crm.view", "=", "Partner"
filter.AddCondition "crm.context", "=", "btpartner" filter.AddCondition
"crm.column:btpartner.partner_no", "=", "244" filter.AddCondition
"crm.column~Partner Function", "=", "Sales employee" Set subFilter =
filter.SetFilter()
subfilter.AddCondition "crm.tagType", "=", "valueHelp"
Set subFilter = filter.AddFilter()
subfilter.AddCondition "crm.tagType", "=", "checkbox"
Set collection = query.Select()
If collection Is Nothing Then
    CBTA.Report CBTA.FAILED, "Demo Query", "No controls have been found", ""
Else
    CBTA.Report CBTA.INFO, "Demo Query", "Nb. Controls: " &
collection.Count, "" for i=0 To collection.Count-1 Set childControl =
collection.ControlAt(i)
    CBTA.Report CBTA.INFO, "Demo Query", "Uri: " &
childControl.GetControlUri(), ""
Next
End If

```

In diesem Beispiel prüft der Hauptfilter nicht mehr nach dem **ROW\_SELECTOR**-Typ, daher erfüllen alle Zellen in der Zeile seine Kriterien. Der Unterschied besteht darin, dass wir nun zwei Unterfilter definiert haben, die den Typ der gesuchten UI-Elemente angeben.

Der erste Unterfilter prüft auf Elemente, die mit einer F4-Hilfe verknüpft sind, die die **Eingabehilfe** verwenden.

#### Code Syntax

```
Set subFilter = filter.SetFilter()  
subfilter.AddCondition "crm.tagType", "=", "valueHelp"
```

Der zweite Unterfilter prüft auf Ankreuzfeld-UI-Elemente:

#### Code Syntax

```
Set subFilter = filter.AddFilter()  
subfilter.AddCondition "crm.tagType", "=", "checkbox"
```

Hier gibt es zwei Unterfilter. Beide haben eine einzelne Bedingung, die auf das Attribut **crm.tagType** prüft.

- Der erste Unterfilter wird durch Aufruf der Methode **SetFilter** im Hauptfilterobjekt angelegt.
- Der zweite Unterfilter wird von der Methode **AddFilter** hinzugefügt.

Durch die Definition mehrerer Filter geben wir an, dass wir nach Zellen suchen, die beiden CRM-Tag-Typen entsprechen. Dies ist möglich, da die Filter, im Gegensatz zu Bedingungen, durch den logischen Operator **OR** verbunden sind.

### Standardmäßige Komponentenimplementierungen wiederverwenden

Die Controller-Schnittstellen stellen außerdem Methoden zur Interaktion mit der Anwendungsbenutzeroberfläche bereit. Pro Standardkomponente stellt CBTA mindestens eine Methode zur Verfügung.

Die beiden folgenden Beispiele sind äquivalent:

#### Code Syntax

```
CRM_Table_SelectRow crmControl.GetControlUri()  
If ReportStatus = "FAILED" Then  
  ' Nothing to do  
End If
```

...

#### Code Syntax

```
Set operationResult = CrmController().Table_SelectRow  
crmControl.GetControlUri()  
If operationResult.Status = "FAILED" Then  
  CBTA.Report CBTA.FAILED, "Demo Query", "Row selection failed", ""  
Else  
  CBTA.Report CBTA.INFO, "Demo Query", "Row has been selected", ""  
End If
```

Das erste Beispiel ruft die Komponentenimplementierung auf, während das zweite die entsprechende Methode über die Controller-Schnittstelle aufruft. Der Vorteil des ersten Ansatzes ist der Nutzen, der aus der standardmäßigen Ausnahmebehandlung resultiert (Schreiben von Informationen in den Ausführungsreport, wenn ein Fehler auftritt). Im zweiten Beispiel ist das Skript zuständig für die Prüfung, ob der Vorgang erfolgreich ist.

### **i** Hinweis

Generell geben die öffentlichen Methoden der Controller-Schnittstellen ein *Operationsergebnisobjekt* zurück, das die *Schnittstelle für Operationsergebnisse* implementiert (weitere Informationen finden Sie unter [Technische Informationen für das Query-API \[Seite 109\]](#)). Die dort verfügbaren Informationen können ein lesbares Feedback im Ausführungsreport bereitstellen.

Weitere Informationen finden Sie im Abschnitt *Schreiben von Informationen in den Ausführungsreport* weiter vorne in diesem Dokument.

### Zuordnung zwischen dem Ausführungsreport für öffentliche Methoden und Komponentenimplementierungen

Die folgende Liste enthält Beispiele von Methoden, die von den Controller-Schnittstellen bereitgestellt werden, und ihren Komponentenimplementierungen:

Controller	Methodenname	Komponentenimplementierung (VB-Skriptfunktion)
<code>CrmController()</code>	<code>CaptureScreen</code>	<code>SetElementValue</code>
	<code>WebControl_Click</code>	<code>CRM_WebControl_Click</code>
	<code>SetElementValue</code>	<code>N/A</code>

Controller	Methodenname	Komponentenimplementierung (VB-Skriptfunktion)
<code>WebController()</code>	<code>CaptureScreen</code>	<code>WEB_CaptureScreen</code>
	<code>WebControl_SetValue</code>	<code>WEB_WebControl_SetValue</code>
	<code>WebControl_CheckProperty</code>	<code>WEB_WebControl_CheckProperty</code>

### Beispiel: Eine Zeile nach Content suchen

CBTA enthält Standardkomponenten, die Tabellen anhand des Contents einer der Zellen nach einer Zeile durchsuchen. Folgende Komponenten stehen zur Verfügung:

- `CBTA_IS_T_FindRow`: sucht eine Light-Speed-Tabelle für eine Zeile, indem der Wert einer einzelnen Zelle geprüft wird.



Eingabeparameter:

- **URI:** gibt den URI des Tabellencontainers an
- **ColumnName:** Titel der Spalte (der sichtbare Text)
- **Operator:** Boolescher Operator für den Vergleich des tatsächlichen Werts mit dem erwarteten Wert
- **Content:** erwarteter Wert der Zelle
- **Options:** Einige Optionen können verwendet werden, um die Werte vor dem Vergleich zu konvertieren.  
Ausgabeparameter: Die Nummer der Zeile, die den Kriterien entspricht.

- **CBTA\_CRM\_T\_FindRow:** ähnlich wie **CBTA\_LS\_T\_FindRow**, aber für SAP-CRM-Webanwendungen  
Beide Komponenten wurden mit dem Query-API implementiert, sodass ein Testingenieur den Quellcode der Komponentenimplementierung kopieren und einfügen und an seine Anforderungen anpassen kann.

## i Hinweis

Der Code von **CBTA\_LS\_T\_FindRow** befindet sich in der Laufzeitbibliothek, er ist in der Datei **LS\_Functions.vbs** sichtbar.

Verwenden Sie den Runtime-Library-Manager, um die Laufzeitbibliothek anzupassen und benutzerdefinierte Funktionen zu schreiben. Wenn Sie die Laufzeitbibliothek zum Bearbeiten öffnen, wird die Datei **LS\_Functions.vbs** lokal im Dateisystem an dem Speicherort zur Verfügung gestellt, der vom Testingenieur angegeben wurde.

## VB-Skriptfunktion – **LsTable\_FindRowByContent**

### Code syntax

```
Public Function LsTable_FindRowByContent ( Uri, ColumnTitle, Operator,
Content,
Options )
    On Error Resume Next
    EventWebComponentBegin()
    LsTable_FindRowByContent = LsTable_FindRowByContent_Impl( Uri, ColumnTitle,
Operator, Content, Options )
    EventWebComponentEnd()
End Function
Public Function LsTable_FindRowByContent_Impl ( Uri, ColumnTitle, Operator,
Content, Options )
LsTable_FindRowByContent_Impl = ""
    If IsNull(Uri) Or IsNull(ColumnTitle) Or IsNull(Content) Then
        Exit Function
    End If
    If IsNull(Operator) Then
        Operator = "="
    End If
    If IsNull(Options) Then
        Options = ""
    End If
    ReportDebugLog "LsTable_FindRowByContent" & _
vbCrLf & "Uri: " & Uri & _                vbCrLf & _
"ColumnName: " & ColumnTitle & _        vbCrLf & _
& "Operator: " & Operator & _          vbCrLf & _
"Content: " & Content & _              vbCrLf & _
"Options: " & Option
    Dim controller, query, filter
    Set controller = LsController()
    Set query = controller.CreateLsQuery()
    query.ParentControlUri = uri
    Set filter = query.SetFilter()
    filter.AddCondition "tag", "=", "TD"
    filter.AddCondition "ls.subtype", "=", "SC"
    filter.AddCondition "ls.column~" & ColumnTitle, Operator, Content, Options
    Set control = query.SelectSingle()
```

```

    If control Is Nothing Then      ReportLog "FAILED",
    "LsTable_FindRowByContent",
    "Operation Failed - Row could not be found" Else
    LsTable_FindRowByContent_Impl = control.GetRelevantControl().GetLsRow()
    If InStr(Options, "/Select" ) Then
    ReportLog "INFO", "LsTable_FindRowByContent", _
    "Operation succeeded - Row has been selected" & vbCrLf & _
    "Row is: " & LsTable_FindRowByContent      ReportDebugLog
    "LsTable_FindRowByContent - Row Selector Uri: " &
    control.GetControlUri()      WEB_WebControl_Table_SelectRow
    control.GetControlUri()
    Else
    ReportLog "INFO", "LsTable_FindRowByContent", _
    "Operation succeeded - Row is: " & LsTable_FindRowByContent
    End If
    End If
End Function

```

## Komponente LsTable\_FindRowByContent

In den folgenden Abschnitten wird ausführlich erklärt, wie diese Komponente implementiert wurde.

Ausnahmebehandlung: Die Implementierung besteht aus zwei Funktionen. Die erste Funktion wird nur benötigt, weil die VB-Skriptsyntax nicht für die Ausnahmebehandlung geeignet ist und das hier verwendete On Error Resume Next-Schlüsselwort die einzige verfügbare Option darstellt.

Alle Standardkomponenten sind nach demselben Konzept aufgebaut; das erste umfasst den Aufruf der tatsächlichen Implementierung zum Abfangen von Fehlern.

### Code syntax

```

Public Function LsTable_FindRowByContent ( Uri, ColumnTitle, Operator,
Content,
Options )
    On Error Resume Next
    EventWebComponentBegin()
    LsTable_FindRowByContent = LsTable_FindRowByContent_Impl( Uri, ColumnTitle,
Operator, Content, Options )
    EventWebComponentEnd()
End Function

```

Eingabeparameter validieren: Die ersten Anweisungen der Implementierung validieren die Eingabeparameter, um häufig auftretende Issues zu vermeiden.

### Code syntax

```

Public Function LsTable_FindRowByContent_Impl ( Uri, ColumnTitle, Operator,
Content, Options )
    LsTable_FindRowByContent_Impl = "
    If IsNull(Uri) Or IsNull(ColumnTitle) Or IsNull(Content) Then
    Exit Function
    End If
    If IsNull(Operator) Then
    Operator = "="
    End If
    If IsNull(Options) Then
    Options = ""
    End If

```

```
...  
End If
```

In diesem Beispiel sind der URI und einige Parameter obligatorisch, einige haben einen Standardwert. Das Skript prüft sie alle und reagiert entsprechend.

Informationen in den Ausführungsreport schreiben: Der folgende Code stellt dem Testingenieur Feedback zur Verfügung. Der Wert jedes Eingabeparameters wird in den Ausführungsreport geschrieben.

#### Code syntax

```
ReportDebugLog "LsTable_FindRowByContent" & _  
vbCrLf & "Uri: " & Uri & _           vbCrLf &  
"ColumnName: " & ColumnTitle & _      vbCrLf &  
& "Operator: " & Operator & _         vbCrLf &  
"Content: " & Content &                vbCrLf &  
"Options: " & Option
```

### Anlegen der auf einen Tabellencontainer beschränkten Abfrage

Wenn die Eingabeparameter konsistent sind, wird mit dem Light Speed Controller eine Abfrage angelegt, deren Umfang auf die Tabelle beschränkt wird, indem die Eigenschaft **ParentControlUri** festgelegt wird:

#### Code syntax

```
Dim controller, query, filter  
Set controller = LsController() Set  
query = controller.CreateLsQuery()  
query.ParentControlUri = uri
```

### Definition der Kriterien

Um eine Zeile durch Prüfung des Contents einer der Zellen zu finden, wird ein einzelner Filter mit drei Bedingungen verwendet:

#### Code syntax

```
Set filter = query.SetFilter() filter.AddCondition "tag", "=", "TD"  
filter.AddCondition "ls.subtype", "=", "SC" filter.AddCondition  
"ls.column~" & ColumnTitle, Operator, Content, Options
```

Die erste Bedingung gibt das Tag an, das vom Light Speed Framework zur Generierung einer Zelle verwendet wird. Es wird das reguläre **<TD>**-HTML-Tag verwendet.

### i Hinweis

Diese Bedingung ist nicht obligatorisch, das Herausfiltern der anderen Tags wirkt sich jedoch erheblich auf die Performance aus, da die Anzahl an UI-Elementen, die dieser Bedingung entsprechen, möglicherweise gering ist.

Die zweite Bedingung basiert auf Light-Speed-Daten zur Ermittlung des Subtyps des Controls. Das Light Speed Framework markiert alle Zellen mit dem **sc**-Subtyp, unabhängig vom tatsächlichen Typ (wie etwa Eingabefeld oder Ankreuzfeld). Weitere Informationen zu für das Light Speed Framework spezifischen Daten finden Sie unter [Technische Informationen für das Query-API \[Seite 109\]](#).

Die dritte Bedingung ist die wichtigste; sie verwendet den **ColumnTitle**-Parameter, der zum Suchen nach der Zelle angegeben wurde, die in der gesuchten Spalte angezeigt wird. Diese Bedingung verwendet außerdem die **Operator**-, **Content**- und **Options**-Parameter, um den Wert anzugeben, der in den Zellen erwartet wird. Weitere Informationen zu Booleschen Operatoren und Optionen für Bedingungen finden Sie unter [Technische Informationen für das Query-API \[Seite 109\]](#).

### Die Abfrage auflösen

Die Kriterien wurden definiert; die Abfrage ist jetzt zur Ausführung bereit. Die Abfrageauflösung wird durch den folgenden Code ausgelöst:

#### Code syntax

```
Set control = query.SelectSingle()
If control Is Nothing Then ReportLog "FAILED", "LsTable_FindRowByContent",
"Operation Failed - Row could not be found" Else
    LsTable_FindRowByContent_Impl = control.GetRelevantControl().GetLsRow()
    ...
End If
```

Durch Aufruf der Methode **SelectSingle** sucht die Komponente **CBTA\_LS\_T\_FindRow** nur nach einer einzelnen Zeile, d.h., nach der ersten Zeile, die den Kriterien entspricht. Es besteht auch die Möglichkeit, eine Reihe von Zellen abzurufen, die den Kriterien entsprechen, indem Sie stattdessen die Methode **Select** aufrufen.

### Die Zeilennummer zurückgeben

Der folgende Code ruft die Zeilennummer anhand von Light-Speed-Daten ab, die von Schnittstellen zur Verfügung gestellt werden, die spezifisch für die zugrunde liegende UI-Technologie sind.

Ausführliche Informationen zu Light-Speed-relevanten Control-Schnittstellen finden Sie unter [Technische Informationen für das Query-API \[Seite 109\]](#).

#### Code syntax

```
LsTable_FindRowByContent_Impl = control.GetRelevantControl().GetLsRow()
...
```

#### i Hinweis

Die Komponente **CBTA\_LS\_T\_FindRow** gibt die Zeilennummer in ihrem Ausgabeparameter zurück. Die zum Festlegen des Ausgabeparameters verwendete interne Methode ist nicht im Umfang dieses Dokuments enthalten.

### Die Zeile auswählen

Mit der Komponente **CBTA\_LS\_T\_FindRow** kann die Zeile ausgewählt und gesucht werden. Das Skript muss nach dem URI der Zelle fragen, die den Kriterien entspricht, und dann die Zeilenauswahl auslösen.

Der URI wird abgerufen, indem die folgende Methode **GetControlUri** aufgerufen wird, und die Auswahl erfolgt durch Aufruf der Funktion **WEB\_WebControl\_Table\_SelectRow**, die der **CBTA\_WEB\_SelectRow**-Standardkomponente entspricht. Im folgenden Beispiel erfolgt die Auswahl nur, wenn der Options-Parameter Folgendes enthält: **/Select**

## Code Syntax

```
If InStr(Options, "/Select" ) Then
    ReportLog "INFO", "LsTable_FindRowByContent",
        "Operation succeeded - Row has been selected" & vbCrLf & _
        "Row is: " & LsTable_FindRowByContent    ReportDebugLog
    "LsTable_FindRowByContent - Row Selector Uri: " &
    control.GetControlUri()    WEB_WebControl_Table_SelectRow
    control.GetControlUri()
Else
    ReportLog "INFO", "LsTable_FindRowByContent",
        "Operation succeeded - Row is: " & LsTable_FindRowByContent
End If
```

### 3.4.4.5.2.3 Technische Informationen für das Query-API

In diesem Kapitel werden zusätzlich zu den vorherigen Dokumenten zum Query-API die technischen Informationen aufgeführt.

## Attribut- und Eigenschaftsnamen

### HTML-Attribute und -Eigenschaften

Mit der Bedingung können Sie die regulären HTML-Attribute und -Eigenschaften prüfen. Folgende Attributnamen werden unterstützt:

- **id**: ID des HTML-Elements
- **tag**: Tag des HTML-Elements
- **innerText** (\*): Wert der **innerText**-Eigenschaft des HTML-Elements
- **value** (\*): Wertattribut des HTML-Elements
- **readOnly** (\*): Wert des **readOnly**-Attributs als Boolescher Ausdruck
- **disabled** (\*): Boolescher Wert des **disabled**-Attributs als Boolescher Ausdruck
- **class**: Liste der auf das HTML-Element angewandten Stile
- **className**: identisch mit **class**
- **href**: **href**-Attribut einer Verknüpfung, d.h. des Anker-HTML-Elements

### ⚠ Achtung

(\*) Einige HTML-Attribute spiegeln, abhängig von der UI-Technologie, möglicherweise nicht den visuellen Aspekt der Web-Controls wider. Beispielsweise beruhen die UI-Technologien nicht zwangsläufig auf einem deaktivierten HTML-Attribut zur Aktivierung oder Deaktivierung eines Controls, deshalb ist das deaktivierte Attribut bedeutungslos. In einer solchen Situation kann ein anderes Attribut, das spezifisch für die zugrunde liegende UI-Technologie ist, die Informationen bereitstellen.

### WebCUIF-Attribute

Die WebCUIF-Attribute, die zum Testen von CRM-Anwendungen verwendet werden können, beginnen mit dem Präfix **crm.** Die folgenden Attribute werden unterstützt:

- **crm.value**: Wert des CRM-Web-Controls  
Das zugrunde liegende HTML-Attribut, mit dem die Informationen abgerufen werden, kann, je nach UI-Elementart variieren. Beispielsweise wird das HTML-Wertattribut für HTML-Eingabelemente zurückgegeben, das **innerText**-HTML-Attribut wird aber stattdessen für Texte und Beschriftungen verwendet.
- **crm.id**: CRM-ID des CRM-Web-Controls  
Diese ID ist im HTML-Content eindeutig. Sie hat eine CRM-spezifische Syntax und stellt folgende Informationen bereit:
  - **A\_**-Fragment: gibt die Anwendung an
  - **V\_**-Fragment: gibt den Sichtnamen an
  - **T\_**-Fragment: gibt die Art an
  - **C\_**-Fragment: gibt den Kontextknoten an
  - **I\_**-Fragment: entspricht der Schnittstelle (optional); für Eingabefelder der technische Name des Felds
  - **R\_**-Fragment: stellt die Zeilennummer bereit (optional)
  - **K\_**-Fragment: stellt den Schlüssel bereit (optional)
- **crm.type**: Typ des CRM-Web-Controls  
Dieser Typ entspricht dem **T\_**-Fragment von **crm.od**. Er ist unabhängig von dem internen HTML-Tag, das zum Rendern des UI-Elements verwendet wird. Beispielsweise wird der Typ **inputField** für Ankreuzfelder verwendet.
- **crm.tagType**: Tag-Typ oder -Subtyp, abhängig von der UI-Elementart.  
CRM-Web-Controls vom Typ **inputField** können folgende **tagType**-Einstellungen aufweisen:
  - **checkbox** für Ankreuzfeldsteuerungen
  - **dropdownListBox** für Kombinationsfeldsteuerungen
  - **valueHelp** für Eingabefelder mit **F4**-Hilfe
- **crm.application**: **A\_**-Fragment von **crm.id**
- **crm.view**: **V\_**-Fragment von **crm.id**
- **crm.context**: **C\_**-Fragment von **crm.id**
- **crm.interface**: **I\_**-Fragment von **crm.id**
- **crm.row**: **R\_**-Fragment von **crm.id**
- **crm.key**: **K\_**-Fragment von **crm.id**
- **crm.column.title**: Titel der Spalte, die das CRM-Web-Control enthält, wenn ein **R\_**-Fragment vorhanden ist
- **crm.wrongName**: gibt [**wrongName**] zurück, um einen unerwarteten Attributnamen anzugeben, der als Eingabeparameter übergeben wird
- **crm.column.<field>**: Wert der Zelle, die das Feld enthält, das von dem **<field>**-Fragment angegeben wird  
Das **<field>**-Fragment stellt den technischen Namen eines Felds bereit, das von der Zelle angezeigt wird. Es muss das Format **<Kontext>.<Schnittstelle>** aufweisen.
- **crm.column~<title>**: Wert der Zelle, die von der Spalte mit **<title>** im Spaltenkopf angezeigt wird

### Light-Speed-Attribute

Auf dem URI Light Speed Framework basierende Anwendungen generieren Controls, die einige Metainformationen umfassen, die von Abfragen zur Suche nach Controls verwendet werden können:

- **ls.type**: Typ des Controls
- **ls.subtype**: Subtyp des Controls (falls vorhanden)

- ls.id:** ID des HTML-Elements für das beste aktuelle Control
- ls.value:** Wert des HTML-Elements für das beste aktuelle Control
- ls.value:** Tag des HTML-Elements für das beste Control

### Web-Dynpro-ABAP-Attribute

Die folgenden Attribute sind nur verfügbar, wenn Sie Anwendungen testen, die in Web Dynpro ABAP entwickelt wurden. Sie stellen weitere Informationen zu den Feldern bereit:

- **ls.component:** Name der Komponente, zu der das Control gehört
- **ls.application:** (Alias) identisch mit **ls.component**
- **ls.view:** technischer Name der Sicht, zu der das Control gehört
- **ls.field:** technischer Name des Felds in der entsprechenden Sicht
- **ls.row:** Zeilennummer (wenn das Control in eine Tabelle eingebettet ist)
- **ls.column.title:** Titel der Spalte, die das Control enthält, wenn das Control in eine Tabelle eingebettet ist
- **ls.column.<field>:** Wert der Zelle, die das Feld enthält, das von dem **<field>**-Fragment angegeben wird  
Das **<field>**-Fragment stellt den technischen Namen eines Felds bereit, das von der Zelle angezeigt wird.
- **ls.column~<title>:** Wert der Zelle, die von der Spalte mit **<title>** als Spaltenkopf angezeigt wird

### Light-Speed-Daten

Light Speed Controls haben ein HTML-Attribut, das zusätzliche Metainformationen bereitstellt: das **ls.data-**Attribut. Diese Informationen werden geparkt und dem Test-Framework als Light-Speed-Attribute zur Verfügung gestellt, die zur Definition von Bedingungen verwendet werden können. Hier werden nicht alle Light-Speed-Attribute angegeben, da jedes Control, abhängig von der Art, einen anderen Datensatz aufweisen kann. Mit dem Object Spy können Sie nach einem Control suchen und die verfügbaren Daten sehen.

Folgende Light-Speed-Attribute sind gebräuchlich:

- **ls.data.value:** Wert des Controls
- **ls.data.state:** Status des Controls, beispielsweise **checked** für ein Ankreuzfeld
- **ls.data.enabled:** definiert, ob das Control aktiviert ist (**true** oder **false**)
- **ls.data.readonly:** definiert, ob der Control-Wert geändert werden kann (**true** oder **false**)
- **ls.data.text:** Text des Controls  
Dieser ist üblicherweise ähnlich wie das **innerText**-HTML-Attribut, doch wenn das Control komplex ist, gibt das **innerText**-Attribut möglicherweise einen mehrzeiligen Wert (einschließlich Zeilenumschaltungszeichen) zurück.  
**ls.data.text:** mit dem Control verknüpfte Quick-Info

## Boolesche Operatoren

Das Query-API unterstützt die folgenden Operatoren, wenn Sie Bedingungen definieren.

- **=:** gleich
- **==:** entspricht =
- **<>:** ungleich
- **!:** entspricht <>

- <: kleiner
- <=: kleiner gleich
- >: größer
- >=: größer gleich
- **{contains}**: enthält
- **{startsWith}**: beginnt mit • **{endsWith}**: endet mit
- **{matches}**: Wert entspricht einem regulären Ausdruck; reguläre Ausdrücke verwenden die C#-Syntax.

## Optionen für Bedingungen

Für die meisten Attribute vergleichen die Bedingungen Operandenwerte als Zeichenfolgen; beim Vergleich müssen Klein- und Großschreibung beachtet werden, wie im folgenden Beispiel:

```
filter.AddCondition "crm.column~Partner Function", "=", "Sales employee"
```

Für das Skript ist u.U. ein Vergleich unter Berücksichtigung von Klein- und Großschreibung erforderlich. In diesem Fall gibt ein zusätzlicher Parameter Bedingungsoptionen an. Beispiel: `filter.AddCondition "crm.column~Partner Function", "=", "SaLeS EmPIOYee", "/u"`

Mit der Option `/u` werden beide Werte vor dem Vergleich in Großbuchstaben umgewandelt.

Folgende Optionen stehen zur Verfügung:

- **/u**: Großbuchstaben. Vergleich unabhängig von Groß-/Kleinschreibung. Beide Operandenwerte werden vor dem Vergleich in Großbuchstaben umgewandelt.
- **/t**: abgeschnitten. Beide Operandenwerte werden vor dem Vergleich abgeschnitten, um vor- und nachgestellte Leerzeichen zu entfernen.
- **/b**: Boolescher Wert. Beide Operandenwerte werden vor dem Vergleich in einen Booleschen Wert (**true** oder **false**) umgewandelt. Diese Option ist implizit für Boolesche Attribute wie **readOnly** und **disabled**
- **/i**: Ganzzahl. Beide Operandenwerte werden vor dem Vergleich in eine Ganzzahl umgewandelt. Der Wert wird nicht abgeschnitten; beispielsweise wird der Wert **10,50** in **11** umgewandelt.
- **/f**: Float. Beide Operandenwerte werden vor dem Vergleich in Float umgewandelt.

## Webabfrage-API

### Web-Controller-Schnittstelle

Die Web-Controller-Schnittstelle ist der Haupteinstiegspunkt zum Testen von Webanwendungen. Sie stellt folgende Methoden bereit:

- **CreateQuery**: legt ein Abfrageobjekt an, um den HTML-Content nach Web-Controls zu durchsuchen
- **CreateWebQuery**: identisch mit der **CreateQuery**-Methode des Web-Controllers
- **GetWebControlByUri**: löst den URI-Parameter auf und gibt das entsprechende Web-Control zurück (falls vorhanden)
- **GetElementByUri**: löst den URI-Parameter auf und gibt das entsprechende HTML-Element zurück (falls vorhanden)



## Webabfrageschnittstelle

Die Webabfrageschnittstelle ist die Hauptschnittstelle aller Abfrageobjekte. Sie stellt folgende Methoden bereit:

- **SetFilter**: entfernt alle Filter und definiert den ersten Eingabeparameter: Keiner. Verwenden Sie die Methoden **SetCondition** und **AddCondition** zur Definition der Kriterien.
- **AddFilter**: fügt einen zusätzlichen Filter hinzu  
Eingabeparameter: Keiner. Verwenden Sie die Methoden **SetCondition** und **AddCondition** zur Definition der Kriterien.
- **SelectSingle**: löst die Abfrage auf und gibt das erste Web-Control zurück, das den Kriterien entspricht
- **Select**: löst die Abfrage auf und gibt eine Reihe von Web-Controls zurück  
Eingabeparameter: **maxHits** (optional), gibt die maximale Anzahl an zurückzugebenden Controls an.
- **GetMainControl**: löst den **ParentControlUri** auf (falls angegeben) und gibt das entsprechende Web-Control zurück  
Außerdem werden folgende Eigenschaften bereitgestellt:
  - **ParentControlUri**: definiert den Umfang der Suche nach Controls  
Der Wert entspricht dem URI des übergeordneten Containers, der gesucht werden soll. Wenn er leer ist, durchsucht die Abfrage das Hauptdokument der Hauptseite. Weitere Informationen finden Sie unter *Reduzieren des Umfangs einer Abfrage* und *Suchen nach Web-Controls in einem Popup-Fenster*.

## Webfilterschnittstelle

Ein Filterobjekt, das eine Liste von Bedingungen mit den Kriterien enthält, denen die Controls entsprechen müssen, und das Unterfilter haben kann, in denen zusätzliche Kriterien angegeben werden.

- **SetCondition**: entfernt alle Bedingungen (falls vorhanden) und definiert die erste.  
Eingabeparameter:
  - **AttributeName**: Attributname (oder Eigenschaftsname)
  - **BooleanOperator**: Operator, der zur Auswertung der Bedingung verwendet werden soll
  - **Value**: erwarteter Wert
  - **Optionen**: Konvertierungsoptionen  
Weitere Informationen finden Sie außerdem unter *Attribut- und Eigenschaftsnamen*, *Boolesche Operatoren* und *Optionen für Bedingungen*.
- **AddCondition**: fügt eine Bedingung hinzu  
Eingabeparameter: identisch mit dem für **SetCondition**.
- **RemoveAllConditions**: entfernt alle Bedingungen (falls vorhanden)
- **SetFilter**: entfernt alle Unterfilter und definiert den ersten  
Eingabeparameter: Keiner. Verwenden Sie die Methoden **SetCondition** und **AddCondition** zur Definition der Kriterien.  
Weitere Informationen finden Sie unter *Kombinieren von Filtern für komplexe Abfragen*.
- **AddFilter**: fügt einen zusätzlichen Unterfilter hinzu  
Eingabeparameter: Keiner. Verwenden Sie die Methoden **SetCondition** und **AddCondition** zur Definition der Kriterien.
- **RemoveAllFilters**: entfernt alle Filter (falls vorhanden)

## Webbedingungsschnittstelle

Bedingungen werden mit den Methoden **SetCondition** und **AddCondition** der Webfilterschnittstelle definiert. Das Bedingungsobjekt, das angelegt wird, stellt folgende Eigenschaften bereit:

- **AttributeName**: Name des Attributs oder der Eigenschaft, das bzw. die von dieser Bedingung verwendet wird
- **BooleanOperator**: Operator, der zur Auswertung dieser Bedingung verwendet wird
- **Value**: erwarteter Wert des Attributs oder der Eigenschaft
- **Option**: Optionen, die zur Auswertung dieser Bedingung verwendet werden

### Web-Control-Schnittstelle

Die Web-Control-Schnittstelle bietet Zugriff auf die HTML-Attribute und -Eigenschaften des zugrunde liegenden HTML-Elements. Sie stellt folgende Methoden bereit:

- **GetHtmlElement**: gibt das zugrunde liegende HTML-Element zurück  
Das hier zurückgegebene Objekt implementiert, je nach Art, Microsoft-Schnittstellen.
- **GetControlUri**: gibt den URI des Controls im aktuellen HTML-Content zurück  
Mit diesem URI kann eine standardmäßige Komponentenimplementierung aufgerufen werden, um Benutzeraktionen wie etwa einen Mausklick zu simulieren.
- **GetAttribute**: gibt den Wert eines HTML-Attributs des zugrunde liegenden HTML-Elements zurück  
Eingabeparameter:
  - **AttributeName**: Name (Wert) des Attributs.
- **GetProperty**: gibt den Wert einer HTML-Eigenschaft des zugrunde liegenden HTML-Elements zurück  
Eingabeparameter: **PropertyName**. Name (Wert) der Eigenschaft (zum Beispiel **innerText**).
- **Highlight**: hebt das HTML-Element zur Laufzeit hervor, zur Fehlerbehebung  
Eingabeparameter: **ColorName**. Die Farbe, die zum Hervorheben der Control-Eigenschaft verwendet wird (beispielsweise **innerText**), zum Beispiel rot, grün oder blau.
- **FindFirstChildElement**: sucht nach einem untergeordneten Control Eingabeparameter:
  - **TagName**: zu suchendes HTML-Tag, z.B. **SPAN**
  - **Recurse** (optional): wahr zur rekursiven Suche in der HTML-**DOM**-Hierarchie von HTML-Elementen
- **FindLastChildElement**: sucht nach dem neuesten untergeordneten Control Eingabeparameter:
  - **TagName**: zu suchendes HTML-Tag, z.B. **SPAN**
  - **Recurse** (optional): wahr zur rekursiven Suche in der HTML-**DOM**-Hierarchie von HTML-Elementen
- **FindChildElementAtPosition**: sucht nach einem untergeordneten Control an einer Position (Index)  
Eingabeparameter:
  - **TagName**: zu suchendes HTML-Tag, z.B. **SPAN**
  - **Index**: Positionsnummer in dem Satz an untergeordneten Controls
  - **Recurse** (optional): wahr zur rekursiven Suche in der HTML-**DOM**-Hierarchie von HTML-Elementen

### Web-Control-Sammlungsschnittstelle

Der Satz an Web-Controls ist das Ergebnis der Abfrage, die von der Methode **Select** aufgerufen wird. Das Testskript kann mit den folgenden Methoden und Eigenschaften durch den Satz iterieren:

- Methode **ControlAt**: gibt das Web-Control an der angegebenen Position zurück Eingabeparameter: **Index**. Die Positionsnummer in dem Satz an Web-Controls.  
Siehe auch *Durch eine Reihe von Web-Controls iterieren*.
- **Count**-Eigenschaft (schreibgeschützt): Anzahl an Controls in dem Satz

### Vorgangsergebnisschnittstelle

Schnittstelle, die Ergebnisse bereitstellt, wenn Sie eine standardmäßige Komponentenimplementierung aufrufen. Sie hat folgende Eigenschaften:

- **Status** (schreibgeschützt): **DONE**, **PASSED**, **WARNING** oder **FAILED**
- **Feedback** (schreibgeschützt): Meldung, die Feedback zu dem durchgeführten Vorgang gibt
- **Value** (schreibgeschützt): Ausgabewert des Vorgangs
- **Comment** (schreibgeschützt): zusätzliche Informationen (optional)
- **ImageTitle** (schreibgeschützt): Titel des Screenshots (falls vorhanden)
- **ImagePath** (schreibgeschützt): Pfad zu dem Screenshot (falls vorhanden)
- **OutputParameters** (schreibgeschützt): Satz an Ausgabeparametern

## Schnittstellen speziell für WebCUIF

### CRM-Controller-Schnittstelle

Die Controller-Schnittstelle für CRM stellt folgende Methoden bereit:

- **CreateCrmQuery**: legt ein Abfrageobjekt an, um den HTML-Content nach CRM-Controls zu durchsuchen
- **CreateQuery**: identisch mit der **CreateCrmQuery**-Methode bei Verwendung dieses CRM-Controllers
- **CreateWebQuery**: identisch mit der **CreateQuery**-Methode des Web-Controllers
- **GetWebControlByUri**: löst den URI-Parameter auf und gibt das entsprechende CRM-Control zurück (falls vorhanden)

### Abfrageschnittstelle

Die CRM-Abfrageschnittstelle stellt dieselben Methoden wie die Webabfrageschnittstelle bereit, die sie erweitert, doch statt regulärer Web-Controls gibt sie WebCUIF-Controls zurück.

### CRM-Control-Schnittstelle

Die CRM-Control-Schnittstelle erweitert die Web-Control-Schnittstelle. Sie übernimmt Funktionen aus der Web-Control-Schnittstelle. Sie stellt folgende zusätzliche Methoden bereit:

- **GetTestModeControl**: gibt das Control zurück, das Metainformationen zu dem CRM-Control bereitstellt. Dieses Control ist normalerweise ein ausgeblendetes **SPAN**-Element, das von dem WebCUIF Framework in den HTML-Content eingefügt wird, wenn CRM-Anwendungen im Testmodus ausgeführt werden.
- **GetTestModeElement**: gibt das SPAN-Element zurück, das Metainformationen zu dem CRM-Control bereitstellt

### CRM-Testmodus-Control-Schnittstelle

Diese Schnittstelle gibt Zugriff auf die Metainformationen, die das WebCUIF Framework in den HTML-Content einfügt, wenn CRM-Anwendungen im Testmodus ausgeführt werden.

Methodenname	Beschreibung	Äquivalent beim Filtern
GetCrmId	gibt die ID des Controls zurück	crm.id
GetCrmType	gibt den Typ des Controls zurück (T_- Fragment der ID)	crm.type

Methodenname	Beschreibung	Äquivalent beim Filtern
GetCrmTagType	gibt den Subtyp des Controls zurück, wie vom <b>tagType</b> -HTML-Attribut definiert, falls vorhanden  Attribut, falls vorhanden	crm.tagType
GetApplication	gibt die Anwendung zurück, zu der das Control gehört ( <b>A_</b> -Fragment der ID)	crm.application
GetViewName	gibt die Sicht zurück, zu der das Control gehört ( <b>V_</b> -Fragment der ID)	crm.application
GetCrmContext	gibt den Kontext zurück, zu dem das Control gehört ( <b>C_</b> -Fragment der ID)	crm.context
GetCrmInterface	gibt den technischen Namen des Control zurück ( <b>I_</b> -Fragment der ID)	crm.context
GetCrmRow	gibt die Zeilennummer zurück, wenn in eine Zelle in einer Tabelle eingebettet	crm.context
GetCrmKey	gibt den Schlüssel des Controls zurück, wenn untergeordnetes Element einer Dropdown-Liste	crm.key

## Schnittstellen speziell für Light Speed

### Light-Speed-Controller-Schnittstelle

Die Controller-Schnittstelle für Light Speed stellt folgende Methoden bereit:

- **CreateIsQuery**: legt ein Abfrageobjekt an, um den HTML-Content nach Light-Speed-Controls zu durchsuchen  
Eingabeparameter: **Options**. Aktiviert die Standardfilterung für gängige Attribute, um die Anzahl der von der Abfrage gefundenen Elemente einzuschränken.
  - **/all**: gibt alle HTML-Elemente zurück, die ein untergeordnetes Element eines Light-Speed-Controls sind. Dies ist der Standardwert.
  - **/ls**: gibt alle Light-Speed-Elemente zurück, die ein **ct**- oder **subct**-Attribut haben.
  - **/ct**: gibt alle Light-Speed-Elemente mit einem **ct**-Attribut zurück
  - **/subct**: gibt alle Light-Speed-Elemente mit einem **subct**-Attribut zurück
  - **/web**: gibt alle HTML-Elemente zurück, auch die, die kein untergeordnetes Element eines Light-Speed-Controls sind Hinweis: Die hier gewählte Option wirkt sich auf die Performance einer Abfrage aus.
- **CreateQuery**: identisch mit der **CreateIsQuery**-Methode bei Verwendung dieses Light-Speed-Controllers
- **CreateWebQuery**: identisch mit der **CreateQuery**-Methode des Web-Controllers

- **GetWebControlByUri**: löst den URI-Parameter auf und gibt das entsprechende WebCUIF-Control zurück, falls vorhanden

### Light-Speed-Abfrageschnittstelle

Die Light-Speed-Abfrageschnittstelle stellt dieselben Methoden wie die Webabfrageschnittstelle bereit, die sie erweitert, doch die Abfrage gibt Objekte zurück, die statt Web-Controls die Light-Speed-Controls implementieren.

### Light-Speed-Control-Schnittstelle

Die Light-Speed-Control-Schnittstelle erweitert die Web-Control-Schnittstelle. Sie übernimmt Funktionen aus der Web-Control-Schnittstelle. Sie stellt folgende zusätzliche Methoden bereit:

- **GetRelevantControl**: gibt das entsprechende Control zurück, dasjenige mit HTML-Attributen (wie **ct** oder **subct**), die den Control-Typ definieren
- **GetRelevantElement**: gibt das relevante Element zurück, das zugrunde liegende HTML-Element des relevanten Controls, dasjenige mit HTML-Attributen (wie etwa **ct** oder **subct**), die den Control-Typ definieren

### Light-Speed-relevante Control-Schnittstelle

Die **relevante** Control-Schnittstelle erleichtert lediglich den Zugriff auf einige Metainformationen, die das Light Speed Framework bereitstellt.

Methodenname	Beschreibung	Äquivalent beim Filtern
GetLsId	gibt die ID des Controls zurück	ls.id
GetLsTag	gibt das Tag des zugrunde liegenden HTML-Elements zurück	ls.tag
GetLsType	gibt den Typ des Controls zurück	ls.type
GetLsSubtype	gibt den Subtyp des Controls zurück	ls.subtype
GetLsComponent	gibt die Komponente des Controls zurück (nur Web Dynpro ABAP)	ls.component
GetLsViewName	gibt die Sicht des Controls zurück (nur Web Dynpro ABAP)	ls.view
GetLsFieldName	gibt den Namen des Controls zurück (nur Web Dynpro ABAP)	ls.field
GetLsRow	gibt die Zeilennummer zurück (wenn in eine Zelle in einer Tabelle eingebettet)	ls.row

### 3.4.4.5.3 Transportverwaltung für Tests und Komponenten

#### Verwendung

Der Transport der Objekte in das Extended Computer Aided Test Tool (eCATT) und der Metadaten in die Umgebung für die Testzusammenstellung, die die Tests und Bildkomponenten der komponentenbasierten Testautomatisierung definiert, ist integriert. Auf diese Weise können Tests für die komponentenbasierte Testautomatisierung transportiert werden.

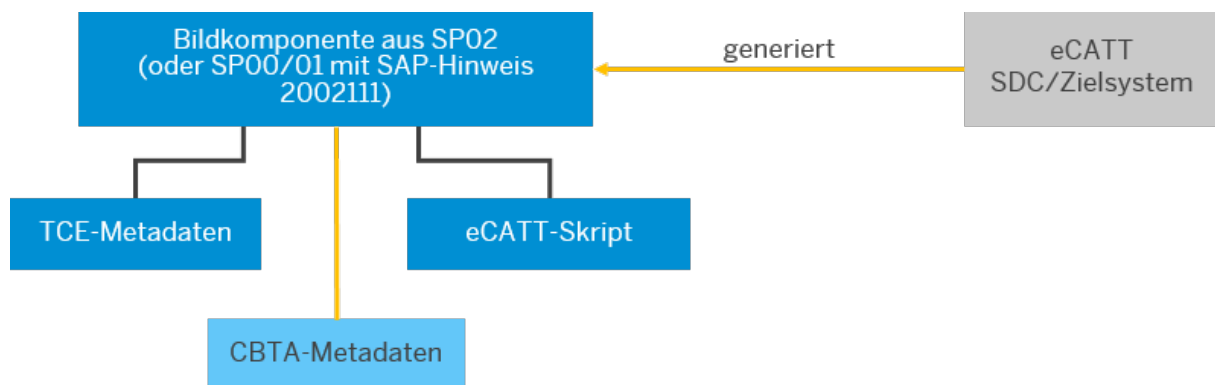
Die komponentenbasierte Testautomatisierung verwendet ein Repository zur Verwaltung ihrer Bildkomponenten. Dort werden Metadaten gespeichert, die vom Komponentengenerierungsprozess verwendet werden, mit dem die komponentenbasierte Testautomatisierung die Komponenten nach Katalog sortieren kann. Ein Katalog der Bildkomponenten ist für einen eCATT-Systemdatencontainer und ein Zielsystem gültig, um eine konsistente Systemlandschaft darzustellen.

#### i Hinweis

Wenn Sie die Umgebung für die Testzusammenstellung verwenden, wenden Sie SAP-Hinweis [1871671](#) (*Fehlende Transportauftragsnummer bei BAdI-Aufruf über CL\_APL\_ECATT\_OBJECT*) an. Anderenfalls sind die Metadaten für die Umgebung für die Testzusammenstellung und die komponentenbasierte Testautomatisierung nicht in Ihrem Transportauftrag.

Im Gegensatz zur Transportverwaltung wird der Download und Upload von Testskripten, die nicht erweitert wurden, für Testskripte der Umgebung für die Testzusammenstellung und der komponentenbasierten Testautomatisierung nicht unterstützt.

Die komponentenbasierte Testautomatisierung generiert zur Aufzeichnungszeit einen Bildkomponentennamen pro Systemdaten-Container (SDC) und Zielsystem (Aktivierung der Bildkomponentenoption während der Aufzeichnung erforderlich). Der Komponentennamen wird durch die Metadaten des Repositories der komponentenbasierten Testautomatisierung bestimmt.



Eine Bildkomponente wird von einem SDC und Zielsystem generiert, die deren Gültigkeit definieren. Eine Bildkomponente kann weiterhin für andere SDC/Zielsysteme ausgeführt werden, da die Testausführungseingabe deren Gültigkeit nicht prüft und stets versucht, sie auszuführen, da sie noch kompatibel sein könnte.

## Funktionsumfang

### Automatische Transportselektion

Der Auftrag des Stammtests skripts wird verwendet, um neue oder aktualisierte Bildkomponenten zu speichern. (Diesen wählen Sie aus der Umgebung für die Testzusammenstellung aus, wenn Sie Ihren Stammtest sichern.)

#### **i** Hinweis

Möglicherweise wird eine CTS-bezogene Meldung angezeigt und bleibt im Hintergrund. Sie blockiert den Prozessablauf im Assistenten für die komponentenbasierte Testautomatisierung. Schließen Sie sie mit **OK**.

### Organisation von Transportaufträgen und Aufgaben

Wenn eine Bildkomponente, die von mehreren Geschäftsprozessen und den zugehörigen Tests gemeinsam verwendet wird, bereits Bestandteil eines anderen Transportauftrags ist, bleibt sie ohne weitere Anmerkung im ursprünglichen Auftrag erhalten. Dies entspricht standardmäßiger CTS-Objektbehandlung. Wenn Sie den Auftrag freigeben, wird er keiner Konsistenzprüfung unterzogen. Sie können einen Test für die komponentenbasierte Testautomatisierung mit nur einer Teilmenge der am Szenario beteiligten Bildkomponenten der komponentenbasierten Testautomatisierung transportieren. Wenn Sie einen Test in Ihrem Transportzielsystem öffnen, werden von der Test Workbench in diesem Fall Fehler zu den fehlenden Komponenten ausgegeben, bis der Test und dessen Verwendung von Komponenten konsistent sind. Nachdem die fehlenden Bildkomponenten transportiert wurden, können Sie die komponentenbasierte Testautomatisierung ausführen, wenn Sie Systemdaten-Container, Zielkomponente und Testerprofil separat definiert haben.

Um dieses Problem zu vermeiden, organisieren Sie Transportaufträge und Aufgaben auf Basis des Umfangs Ihres Projekts. Da Bildkomponenten einem Systemdatencontainer und einer Zielkomponente zugeordnet sind, können Sie einen Transportauftrag anlegen, der von allen Testingenieuren verwendet wird, die an der zugehörigen Testlandschaft beteiligt sind. Wenn der Testumfang zu groß und die Freigabe einer synchronisierten Aufgabe zu schwierig ist, können Sie einen Transportauftrag für jeden Funktionsbereich anlegen, solange geschäftsübergreifende Komponenten ordnungsgemäß verwaltet werden.

### FAQ

#### **Kann ich den Transport einer Bildkomponente erzwingen?**

Eine Bildkomponente der komponentenbasierten Testautomatisierung ist lediglich ein eCATT-Skript, das mit Metadaten aus der Umgebung für die Testzusammenstellung und der komponentenbasierten Testautomatisierung angereichert wird. Deshalb können Sie alle vorhandenen Standardeinstiegspunkte für eCATT-Objekttransporte verwenden, um eine Bildkomponente manuell zu transportieren.

#### **Werden der Systemdatencontainer und die Zielkomponente (dem Stammtest der komponentenbasierten Testautomatisierung zugeordnet) ebenfalls transportiert?**

Wenn Sie einen Test der komponentenbasierten Testautomatisierung sichern, werden der Systemdaten-Container und die Zielkomponente des Tests nicht automatisch dem Transportauftrag hinzugefügt. Sie können dies in der Transaktion **SECATT** definieren. Da erweiterte SDC-Daten nicht transportiert werden können, müssen Sie im Zielsystem mit dem SUT-Verwaltungswerkzeug neu definiert werden. Dies ist nicht wirklich relevant, da die zugrunde liegende RFC-Destination und Benutzeranmeldedaten sowieso nicht transportiert werden können.

Das SUT-Verwaltungswerkzeug kann unter *Verwaltung - Test-Suite* ▶ *Testautomatisierungs-Framework* ▶ *Verwaltung des Systems unter Test* ▶ aufgerufen werden.



# 4 Konfiguration der Test-Suite

## Verwendung


Die grundlegende Einrichtung erfolgt in der *SAP-Solution-Manager-Konfiguration*. Sie wird von mehreren anderen Aktivitäten abgeschlossen, die bei Bedarf ausgeführt werden.

## Aktivitäten

1. Starten Sie die *SAP-Solution-Manager-Konfiguration* (Transaktion **SOLMAN\_SETUP**), und folgen Sie den Guided Procedures unter *Test-Suite*.  
Dies ist für die Test-Suite-Vorbereitung, die Testautomatisierungsvorbereitung, die komponentenbasierte Testautomatisierung und den Business Process Change Analyzer verfügbar.  
Weitere Informationen finden Sie in den Hilfetexten für die Schritte in der Guided Procedure.
2. In der Anwendung *Verwaltung - Test-Suite* können Sie weitere Grundeinstellungen vornehmen. Weitere Informationen finden Sie unter [Administration \[Seite 123\]](#).
3. Die vollständige Liste der Einstellungen für die Test-Suite ist im Customizing von SAP Solution Manager (Transaktion **SPRO**) unter ► *Capabilities* ► *Test-Suite* ► *Test-Suite für SAP Solution Manager* ► verfügbar.  
Weitere Informationen finden Sie in der Dokumentation zu den einzelnen Customizing-Aktivitäten.

## Einrichten des Testwerkzeugs


Bevor Sie Testkonfigurationen für automatisierte Testfälle anlegen können, müssen Sie die erforderlichen Testwerkzeuge einrichten.

1. Wählen Sie in der Anwendung *Verwaltung - Test-Suite* den Pfad ► *Testautomatisierungs-Framework* ► *Definition der Testwerkzeugattribute* ►.
2. Sie können die Attribute, die ein Testwerkzeug dem SAP-System zum Erstellen eines Testskripts bereitstellen kann, prüfen und bearbeiten. Informationen zu den Partner-Testwerkzeugen finden Sie in der Dokumentation des jeweiligen Anbieters.
3. Sichern Sie Ihre Angaben, und schließen Sie das Fenster.
  - Hilfe zur Anwendung *Zentrale Einstellungen der Test Workbench* im SAP Help Portal unter <http://help.sap.com> .
  - [Zuordnen von Fehlerarten \[Seite 269\]](#)

## Registrieren eines Client-seitigen automatischen Testwerkzeugs oder eines Drittanbieter-Testwerkzeugs

Sie können ein bestimmtes Client-seitiges automatisches Testwerkzeug (CBTA oder Drittanbieter-Testwerkzeug) registrieren, mit dem Sie automatische Testfälle anlegen möchten.

## Voraussetzungen

- Sie haben SAP-Hinweis [519858](#)  gelesen.
- Sie haben ein zertifiziertes externes Werkzeug auf dem lokalen Rechner des Benutzers installiert, oder das Werkzeug ist über den Citrix-Client verfügbar. Sie registrieren das Werkzeug im Customizing (Transaktion

S<sub>PRO</sub>) unter ► [SAP Solution Manager](#) ► [Capabilities](#) ► [Test-Suite](#) ► [Test-Suite für SAP Solution Manager](#) ► [Testautomatisierungs-Framework](#) ► [Setup](#) ► [Testwerkzeug registrieren](#) ►:





- Sie haben im Customizing (Transaktion S<sub>PRO</sub>) unter ► [SAP Solution Manager](#) ► [Capabilities](#) ► [Test-Suite](#) ► [Test-Suite für SAP Solution Manager](#) ► [Testautomatisierungs-Framework](#) ► [Setup](#) ► [Benutzer generieren](#) ► einen Standardbenutzer generiert.
- Das Drittanbieterwerkzeug muss für die Kommunikation mit SAP Solution Manager den vordefinierten Benutzer ECATT\_ET\_USR verwenden. Diesem Benutzer ist die Rolle SAP\_ECET zugeordnet.
- Auf dem Frontend-Computer ist SAP GUI 7.40 installiert.
- Das GUI-Scripting ist auf dem Frontend-Computer aktiviert.

### Vorgehensweise

1. Wählen Sie im Customizing von SAP Solution Manager (Transaktion S<sub>PRO</sub>) den Pfad ► [Capabilities](#) ► [Test-Suite](#) ► [Test-Suite für SAP Solution Manager](#) ► [Testautomatisierungs-Framework](#) ► [Setup](#) ► [Testwerkzeug registrieren](#) ►.  
Sie gelangen zum Bild *Customizing-Tabelle für externe Testwerkzeuge*.
2. Wählen Sie [Ändern](#).
3. Wählen Sie [Neue Einträge](#).
4. Wählen Sie unter [Werkzeugname](#) ein Testwerkzeug aus, und geben Sie die erforderlichen Informationen ein.  
Weitere Informationen zum Eingeben der erforderlichen Informationen finden Sie in der Benutzerdokumentation des jeweiligen Drittanbieterwerkzeugs.
5. Wählen Sie [Weiter](#).
6. Sichern Sie Ihre Angaben, und schließen Sie das Fenster.

## 5 Verwaltung - Test-Suite

In der Anwendung *Verwaltung - Test-Suite* können Sie zusätzlich zu den Setup-Aktivitäten in der *SAP-Solution-Manager-Konfiguration* und im Customizing (Transaktion `SPRO`) weitere Customizing-Einstellungen für die Test-Suite festlegen.

- Registerkarte **Vorbereitung**:
  - *Testfallstatus*: Passen Sie die Status an, die ein Tester während der Testausführung einem Testfall zuordnen kann.  
Weitere Informationen finden Sie in den Anweisungen im Customizing (Transaktion `SPRO`) für SAP Solution Manager unter [▶ Capabilities ▶ Test-Suite ▶ Test-Suite für SAP Solution Manager ▶ Vorbereitung ▶ Verwaltung ▶ Teststatuswerte definieren](#) .
  - *Hinweis- und Testergebnisvorlage* für die Testdokumentation (Notizen und Testergebnisse): Laden Sie eine benutzerdefinierte Microsoft-Word-Vorlage hoch, damit Tester beim Ausführen eines manuellen Testfalls anhand der Vorlage eine Notiz oder ein Testergebnis anlegen können.  
Weitere Informationen finden Sie in den Anweisungen im Customizing für SAP Solution Manager unter [▶ Capabilities ▶ Test-Suite ▶ Test-Suite mit SAP Solution Manager ▶ Vorbereitung ▶ Verwaltung ▶▶ Vorlage für Testdokumentation \(Notizen\) definieren](#)  und [▶ Vorlage für Testdokumentation definieren \(Notizen\)](#).
  - *Einstellungen für E-Mail-Benachrichtigung*: Ordnen Sie vorhandene E-Mail-Smart-Forms zu Benachrichtigungsarten zu. Diese E-Mails werden an Tester gesendet, um sie über die Testausführung und über Testfehler zu informieren.  
Weitere Informationen finden Sie in den Anweisungen im Customizing für SAP Solution Manager unter [▶ Capabilities ▶ Test-Suite ▶ Test-Suite für SAP Solution Manager ▶ Vorbereitung ▶ Setup ▶ Standardkonfiguration ▶ E-Mail-Benachrichtigungen ▶ E-Mail-Formulare für Benachrichtigungen zuordnen](#) .
  - Im selben Customizing-Zweig finden Sie auch folgende Optionen:
    - *Benutzer für den Empfang von E-Mail-Benachrichtigungen registrieren*
    - *E-Mail-Formulare für die Test-Suite definieren*
  - *Ausschlussliste für Testplanfilter pflegen*: Um die Aktivität für die Benutzer zu vereinfachen, definieren Sie, welche Attribute bei der Testplanerstellung in den Listen ausgeblendet werden sollen.  
Weitere Informationen finden Sie in den Anweisungen im Customizing für SAP Solution Manager unter [▶ Capabilities ▶ Test-Suite ▶ Test-Suite für SAP Solution Manager ▶ Vorbereitung ▶ Verwaltung ▶ Ausschlussliste für Testplanfilter pflegen](#) .
  - *Globale Einstellungen pflegen*: Definieren Sie die globalen Einstellungen für die Testplanverwaltung, Testausführung, Testautomatisierung und andere Einstellungen.  
Unter *Sonstige Einstellungen* können Sie festlegen, dass ein Tester einen Testnotiz- oder Testergebniseintrag anlegen muss, wenn der Test einen definitiven Status hat (also rot oder grün ist). Der Tester kann keine Sicherung ausführen, ohne den entsprechenden Eintrag angelegt zu haben. Entsprechend können Sie unter *Sonstige Einstellungen* festlegen, dass die manuelle Pflege des Teststatus von automatisierten Testfällen unterbunden wird.

- *Automatische Aufzeichnung des Testaufwands*: Legen Sie die automatische Aufzeichnung des Zeitaufwands für manuelle Testfälle fest:  
Weitere Informationen finden Sie in den Anweisungen im Customizing für SAP Solution Manager unter [▶ Capabilities ▶ Test-Suite ▶ Test-Suite für SAP Solution Manager ▶ Vorbereitung ▶ Verwaltung ▶ Autom. Aufzeichnung des Testaufwands für manuelle Testfälle aktivieren ▶](#).
- *Standardmäßig erforderliche Ausführungszeit von Testfalltypen*: Legen Sie die für die unterschiedlichen Testfalltypen erwarteten Standardausführungszeiten fest.  
Weitere Informationen finden Sie in den Anweisungen im Customizing für SAP Solution Manager unter [▶ Capabilities ▶ Test-Suite ▶ Test-Suite für SAP Solution Manager ▶ Vorbereitung ▶ Setup ▶ Standardkonfiguration ▶ Zentrale Test-Suite-Einstellungen ▶ Standardausführungszeiten für Testfälle je nach Testfalltyp festlegen ▶](#).
- *Definition der Testklassifizierung*: Definieren Sie Testklassifizierungen, z.B. Unit-Test, Funktionstest, Integrationstest, User Acceptance Test oder Regressionstest. Weitere Informationen finden Sie in den Anweisungen im Customizing für SAP Solution Manager unter [▶ Capabilities ▶ Test-Suite ▶ Test-Suite für SAP Solution Manager ▶ Vorbereitung ▶ Verwaltung ▶ Testklassifikation definieren ▶](#).
- *Definition der Vorlage für Testergebnisreport*: SAP liefert eine Vorlage zum Konfigurieren der Testreports aus. Sie können eine eigene Vorlage definieren, wenn Sie z.B. die Titel- und Endseite, die Stichwörter des Dokuments und andere Eigenschaften ändern möchten.
- Registerkarte **Testautomatisierungs-Framework**:
  - *Definition der Testwerkzeugattribute*: Registrieren Sie testwerkzeugspezifische Attribute, die während der Testskriptaufzeichnung in das Werkzeug übertragen werden sollen.  
Weitere Informationen finden Sie in den Anweisungen im Customizing für SAP Solution Manager unter [▶ Capabilities ▶ Test-Suite ▶ Test-Suite für SAP Solution Manager ▶ Testautomatisierungs-Framework ▶ Verwaltung ▶ Testwerkzeugattribute registrieren ▶](#).  
Weitere Informationen finden Sie unter [Pflegen von Attributen \[Seite 220\]](#).
  - *Systeme unter Test pflegen*: Legen Sie die Verbindung und die Anmeldeinformationen für Systeme unter Test in einem Testprofil fest, damit Sie nicht für jeden Geschäftsbutzer Ihres Tests eine separate RFC-Destination definieren müssen.  
Weitere Informationen finden Sie im Customizing für SAP Solution Manager unter [▶ Capabilities ▶ Test-Suite ▶ Test-Suite für SAP Solution Manager ▶ Komponentenbasierte Testautomatisierung ▶ Verwaltung ▶ System unter Test konfigurieren ▶](#).  
Weitere Informationen finden Sie unter [SUT-Verwaltung \[Seite 17\]](#).
- Registerkarte **CBTA**:
  - *Mandanteneinstellungen pflegen*: Konfigurieren Sie den Mandanten für die komponentenbasierte Testautomatisierung, den Sie lokal installiert haben.  
Weitere Informationen finden Sie in den Anweisungen im Customizing für SAP Solution Manager unter [▶ Capabilities ▶ Test-Suite ▶ Test-Suite für SAP Solution Manager ▶ Komponentenbasierte Testautomatisierung ▶ Verwaltung ▶ CBTA-Mandant konfigurieren ▶](#).
  - *RTL anpassen*: Passen Sie die Laufzeitbibliothek für die komponentenbasierte Testautomatisierung an. Sie können benutzerdefinierte Funktionen hinzufügen, z.B. zum Aufzeichnen oder Abspielen von Tests mit Adobe Flash oder Java-Applets.  
Weitere Informationen finden Sie in den Anweisungen im Customizing für SAP Solution Manager unter [▶ Capabilities ▶ Test-Suite ▶ Test-Suite für SAP Solution Manager ▶ Komponentenbasierte Testautomatisierung ▶ Verwaltung ▶ Laufzeitbibliothek anpassen ▶](#).

- Führen Sie *Selbstchecks* für das Werkzeug zur komponentenbasierten Testautomatisierung und für Systeme unter Test durch. Prüfen Sie, ob der Mandant für die die komponentenbasierte Testautomatisierung korrekt installiert und konfiguriert wurde. Weitere Informationen finden Sie in den Anweisungen im Customizing für SAP Solution Manager unter [▶ Capabilities ▶ Test-Suite ▶ Test-Suite für SAP Solution Manager ▶ Komponentenbasierte Testautomatisierung ▶ Verwaltung ▶ Self-Checks durchführen ▶](#).

# 6 Meine Aufgaben: TBOM-Arbeitsvorrat

## Verwendung

Zur Verwendung des Business Process Change Analyzer (BPCA) sind technische Stücklisten (TBOMs) erforderlich. Eine TBOM enthält eine Liste aller Objekte, die an einer bestimmten ausführbaren Einheit beteiligt sind. Anhand von TBOMs kann der BPCA feststellen, ob eine ausführbare Einheit von einer Änderung betroffen ist oder nicht.

Ein Qualitätsexperte kann die Aufzeichnung von TBOMs an einen Geschäftsprozess-Experten oder Tester delegieren, indem er TBOM-Aufgaben (TBOM-Aufzeichnungsaufgaben) anlegt. Im TBOM-Arbeitsvorrat

- überwachen Qualitätsexperten den Fortschritt der von ihnen angelegten Aufzeichnungsaufgaben
- erhalten Geschäftsprozess-Experten einen Überblick über die ihnen zugewiesenen Aufgaben

Hintergrund: Oftmals sind die Benutzer von SAP Solution Manager zwar Qualitätsexperten mit Kenntnissen über SAP Solution Manager, haben aber keine tiefer gehenden Kenntnisse der Geschäftsprozesse in der Lösungsdokumentation. Sie wissen, welche TBOMs angelegt werden sollten, es fehlt ihnen jedoch an den erforderlichen Kenntnissen der Geschäftsprozesse, um sie anzulegen. Auf der anderen Seite haben Geschäftsprozess-Experten Erfahrung mit Geschäftsprozessen, jedoch mangelt es ihnen an Kenntnissen über SAP Solution Manager, daher schließen TBOM-Aufgaben die Wissenslücke zwischen Qualitätsexperten und Geschäftsprozess-Experten.

Bei jeder entscheidenden Statusänderung der Aufzeichnungsaufgabe versendet das System automatisch E-Mails an die beteiligten Benutzer. Die E-Mails enthalten jeweils einen Link zu einer einzelnen TBOM-Aufgabe oder zur Aufgabenübersicht.

## Prozess

1. Um einen Geschäftsprozess-Experten zur Aufzeichnung einer TBOM aufzufordern, legt der Qualitätsexperte eine Aufzeichnungsaufgabe an. Weitere Informationen finden Sie unter [Delegieren von TBOM-Aufzeichnungen an Geschäftsexperten \[Seite 148\]](#).
2. Das System benachrichtigt sowohl den Qualitätsexperten als auch den Geschäftsprozess-Experten per E-Mail über die neue Aufgabe.
3. Der Geschäftsprozess-Experte zeichnet die gewünschte TBOM auf.
4. Das System benachrichtigt den Qualitätsexperten per E-Mail über die Erledigung der Aufzeichnungsaufgabe.
5. Der Qualitätsexperte verifiziert, dass die TBOM aufgezeichnet wurde und bestätigt die Aufgabe.

## 6.1 Überwachen des TBOM-Aufgabenstatus als Qualitätsexperte

### Verwendung

Überwachen Sie als Qualitätsexperte den Status von TBOM-Aufgaben, die Sie angelegt haben. Bestätigen Sie abgeschlossene Aufgaben, oder prüfen Sie Aufgaben, die vom Geschäftsprozess-Experten zurückgegeben wurden.

### Voraussetzungen

Sie haben TBOM-Aufgaben Geschäftsprozess-Experten zugeordnet. Weitere Informationen finden Sie unter [Delegieren von TBOM-Aufzeichnungen an Geschäftsexperten \[Seite 148\]](#).

### Vorgehensweise

#### Prüfen des Status von TBOM-Aufgaben

1. Öffnen Sie Ihren TBOM-Arbeitsvorrat. Sie haben folgende Möglichkeiten:
  - Öffnen Sie die Aufgabe in der Sicht *Meine Aufgaben - TBOM-Arbeitsvorrat*.
  - Wählen Sie den Link in der Benachrichtigungs-E-Mail, wenn auf die Aufgabe Folgendes zutrifft:
    - Sie wurde ausgeführt.
    - Sie wurde vom Geschäftsexperten zurückgegeben.
2. Wählen Sie *Bearbeiten*.  
Sie gelangen zum Dialogfenster *Aufgabe ändern*.
3. Wählen Sie je nach Status eine der folgenden Vorgehensweisen:
  - Status *Abgeschlossen*:
    1. Wählen Sie die Registerkarte *Administrative Daten*.
    2. Prüfen Sie in den Feldern *Angelegt um* oder *Aktualisiert um/von*, ob die TBOM angelegt oder aktualisiert wurde.
    3. Ändern Sie auf der Registerkarte *Allgemeine Daten* den Status in *Bestätigt*.
    4. Wählen Sie *Sichern*.  
Bestätigte Aufgaben werden aus der Liste entfernt.
  - Status *Anfrage zurück zum QA Experten*.
    1. Prüfen Sie auf der Registerkarte *Allgemeine Daten* den Kommentar des Geschäftsexperten, und fügen Sie Ihre Antwort auf die Frage in einen neuen Kommentar ein.
    2. Geben Sie im Feld *Bemerkungen* weitere Erläuterungen ein.
    3. Wählen Sie der Aufzeichnungsaufgabe entsprechend einen Status, z.B. *Neuaufzeichnen von TBOMs*.
    4. Wählen Sie *Sichern*.  
Der Geschäftsexperte erhält eine Benachrichtigung zur Aufgabe.

## Zusätzliche Optionen

Um die Verantwortung für die Aufzeichnungsaufgabe an einen anderen Qualitätsexperten abzugeben, geben Sie unter *Partner* den Benutzertyp und den Namen des neuen Qualitätsexperten ein.

Die Aufgabe wird nun nicht mehr in Ihrem TBOM-Arbeitsvorrat angezeigt, sondern im TBOM-Arbeitsvorrat des neu eingetragenen Qualitätsexperten. Bei Verwendung der Standardkonfiguration informiert das System den neuen Qualitätsexperten per E-Mail über seine neue Aufgabe.

## 6.2 Bearbeiten von TBOM-Aufgaben als Geschäftsprozess-Experte

### Kontext

Zeichnen Sie auf Anfrage eines Qualitätsexperten eine TBOM auf. Sie werden per E-Mail gebeten, eine TBOM aufzuzeichnen. Sie gelangen zur Sicht *TBOM-Arbeitsvorrat*, wenn Sie den Link in der E-Mail wählen.

### Vorgehensweise

1. Markieren Sie die Aufgabe, und wählen Sie *Bearbeiten*.
2. Um den Testfall anzuzeigen, in dem die Bearbeitung der Aufgabe beschrieben wird, wählen Sie *Dokumentation anzeigen*.
3. Falls erforderlich: Leiten Sie die Aufzeichnungsaufgabe an einen anderen Geschäftsprozess-Experten weiter
  1. Wählen Sie *Weiterleiten*.
  2. Geben Sie den Benutzertyp und Benutzernamen des neuen Geschäftsprozess-Experten an.
  3. Optional: Geben Sie einen Kommentar ein, um zu erklären, weshalb Sie die Aufgabe weiterleiten.
  4. Wählen Sie *Weiterleiten*.

Die Aufgabe verschwindet aus dem Arbeitsvorrat des bisherigen Geschäftsprozess-Experten und wird dafür im Arbeitsvorrat des neuen Geschäftsprozess-Experten angezeigt. Bei Verwendung der Standardkonfiguration informiert das System den neuen Geschäftsprozess-Experten per E-Mail über seine neue Aufgabe.
4. Wählen Sie ► *Ausführen* ► *Ausführen* ▾.

Alternativ können Sie auch den Link in der Spalte *Beschreibung* oder *Ausführbare Einheit* wählen.

Wenn Sie die Aufzeichnung beginnen, wechselt der Ausführungsstatus der Aufzeichnungsaufgabe von *Offen* auf *In Ausführung*.
5. Führen Sie die erforderlichen Schritte im verwalteten System aus.
6. Beenden Sie ausführbare Einheiten im SAP GUI mit dem Symbol *Beenden*. Schließen Sie bei Webanwendungen das Browserfenster der Anwendung.



7. Halten Sie die TBOM-Aufzeichnung an.

Wenn Sie die TBOM-Aufzeichnung abgeschlossen haben, wechselt der Ausführungsstatus der TBOM-Aufgabe auf *Ausgeführt*. Wenn die Aufzeichnung fehlerhaft war, bleibt der Ausführungsstatus auf *Offen*.

8. Falls erforderlich, führen Sie einen der folgenden Schritte aus:

- Um eine TBOM-Aufzeichnung zu wiederholen, wählen Sie **▶ Ausführen ▶ Letzte Ausführung wiederholen ▶**.

Beispiel: Sie haben in einem Vorgang versehentlich einen falschen Pfad ausgewählt.

Die letzte aufgezeichnete TBOM-Erweiterung wird gelöscht und mit der neuen Aufzeichnung ersetzt. Wenn die TBOM zuvor schon Erweiterungen enthalten hat, bleiben diese unverändert bestehen.

- Geben Sie die TBOM-Aufzeichnungsaufgaben an den Qualitätsexperten zurück.

Beispiel: In der Dokumentation wird das Vorgehen nicht detailliert genug beschrieben.

1. Wählen Sie *Zurück an QA-Experten*.

2. Geben Sie in das *Kommentar*-Feld ein, welche Informationen Sie benötigen, um die TBOM aufzeichnen zu können.

3. Wählen Sie *Zurückschicken*.

Die Aufzeichnungsaufgabe verschwindet aus Ihrem Arbeitsvorrat. Bei Verwendung der Standardkonfiguration informiert das System den Qualitätsexperten per E-Mail über Ihre Rückfrage.

4. Setzen Sie die Bearbeitung der Aufgabe fort, wenn der Qualitätsexperte auf Ihre Frage geantwortet hat und die Aufgabe in Ihrem Arbeitsvorrat ist.

9. Um zu bestätigen, dass Sie die Aufzeichnung beendet haben, wählen Sie *Abgeschlossen*.

Die Aufzeichnungsaufgabe verschwindet aus Ihrem Arbeitsvorrat. Bei Verwendung der Standardkonfiguration informiert das System den Qualitätsexperten per E-Mail über die Ausführung der Aufgabe.

Das System setzt den Ausführungsstatus der Aufgabe auf *Erledigt*. Der Qualitätsexperte prüft die TBOM-Aufzeichnung und bestätigt die Aufgabe.

# 7 Business Process Change Analyzer (BPCA)

## Verwendung

Änderungen an Support Packages, Kundenentwicklungen, Add-On-Installationen und andere Änderungen des Systems können sich auch auf die Geschäftsprozesse auswirken. Die BPCA vergleicht die Objekte in einem Transport mit den Objekten in den technischen Stücklisten (TBOMs) des Zielsystems. Zusätzlich zu in Transporten enthaltenen Objekten kann der BPCA auch eine Objektliste zu Business Functions abrufen und diese Objektliste mit den Objekten in TBOMs vergleichen. Da jede TBOM einer ausführbaren Einheit in einem Szenario, Geschäftsprozess oder Prozessschritt eindeutig zugeordnet ist, können Sie feststellen, welche Teile einer Lösungsdokumentation von der Änderung betroffen sind.

Verwenden Sie die Testfälle in der Lösungsdokumentation, um maßgeschneiderte Testpläne für Ihre Änderungen zu erstellen.

## Integration

Prüfen Sie die Auswirkungen von Änderungen auf Geschäftsprozesse, nachdem Sie eine Lösungsdokumentation angelegt haben.

## Voraussetzungen

Sie haben die Geschäftsprozesse konfiguriert und zur jeder relevanten ausführbaren Einheit eine TBOM erstellt.

## Funktionsumfang

Der BPCA bietet folgende Möglichkeiten:

- Analysieren Sie, welche Teile einer Lösungsdokumentation von der Änderung betroffen sind.
- Analysieren Sie, welche Teile einer Lösungsdokumentation von der geplanten Aktivierung einer Business Function betroffen sind.
- Sichern Sie Analyseeinstellungen als Variante.
- Legen Sie einen Startzeitpunkt für die Analyse fest.
- Entfernen Sie eingeplante Analysen.
- Werten Sie Analyseergebnisse aus.
- Optimieren Sie den Testumfang eines Analyseergebnisses.

- Legen Sie anhand der Analyseergebnisse Testpläne an.

## Aktivitäten

- Um eine Änderungseinfluss-Analyse auszuführen, wählen Sie im SAP Solution Manager Launchpad die Option *Business Process Change Analyzer*.
- Um zu prüfen, ob alle Änderungen von den Analysen abgedeckt werden, wählen Sie in der Anwendung *Test-Suite - Übersicht*, und wechseln Sie zur Registerkarte *Business Process Change Analyzer*. Die BPCA-Analyse wird für jedes System angezeigt. In einer grafischen Übersicht werden analysierte und nicht analysierte Änderungen nach Änderungsart angezeigt. Um Details anzuzeigen, wählen Sie eine Spalte in der Grafik.
- Wählen Sie zum Analysieren von Änderungen ► *Test-Suite - Übersicht* ► *Business Process Change Analyzer* ►. Wählen Sie ein System und prüfen Sie, ob alle Änderungen abgedeckt sind. Wählen Sie einen Typ, der noch nicht analysiert wurde, und dann *BPCA-Analyse erstellen* wählen.

### i Hinweis

Verbesserungen in SAP Solution Manager 7.2 SPS06 reduzieren den durch TBOMs beanspruchten Datenbankspeicherplatz und verbessern die Leistung der BPCA-Analyse.

Falls Sie BPCA vor SAP Solution Manager SPS06 verwendet haben, empfehlen wir die Migration Ihrer TBOMs, damit Sie von den Optimierungen in SPS06 profitieren können. Zum Migrieren Ihrer TBOMs gehen Sie wie folgt vor:

1. Öffnen Sie die Solution-Manager-Konfiguration (Transaktion **SOLMAN\_SETUP**).
2. Wählen Sie das Konfigurationsszenario ► *Test-Suite* ► *Business Process Change Analyzer* ► aus.
3. Gehen Sie zum Schritt *BPCA-Vorbereitung*, Unterschritt *Migration von TBOMs*.
4. Befolgen Sie die Anweisungen zur Migration Ihrer TBOMs.

## 7.1 Verwaltung - Änderungseinfluss-Analyse

Zusätzlich zu den Setup-Aktivitäten in der *SAP-Solution-Manager-Konfiguration* können Sie in der Anwendung *Verwaltung - Änderungseinfluss-Analyse* weitere Customizing-Einstellungen für den BPCA vornehmen.

- Registerkarte **BPCA-Vorbereitung**:
  - *BPCA-Selbstchecks ausführen*  
Stellen Sie sicher, dass die Daten des Usage and Procedure Logging (UPL) im System erzeugt werden und zur Generierung von semidynamischen TBOMs abgerufen werden können. Weitere Informationen erhalten Sie unter [Voraussetzungen für die Erzeugung semidynamischer TBOMs prüfen \[Seite 142\]](#).
  - *TBOMs generieren*  
Legen Sie semidynamische oder statische TBOMs für die ausführbaren Einheiten in Ihrer Bibliothek der ausführbaren Einheiten an. Weitere Informationen finden Sie unter [Anlegen von statischen oder semidynamischen TBOMs \[Seite 140\]](#).
  - *Aktualitätsprüfung ausführen*

Identifizieren Sie potenziell veraltete TBOMs. Weitere Informationen finden Sie unter [Einplanen von Aktualitätsprüfungen \[Seite 164\]](#).

- *TBOMs löschen*  
Löschen Sie veraltete TBOMs aus mindestens einer Lösung. Weitere Informationen finden Sie unter [Löschen von TBOMs \[Seite 169\]](#).
- *TBOM-Filtertabelle pflegen*: Definieren Sie Filter zur Optimierung von Geschäftsprozessänderungsanalysen, indem nicht relevante Objekte in TBOMs ignoriert werden. Weitere Informationen finden Sie unter [TBOM-Filter bearbeiten \[Seite 178\]](#).
- *Tabelle der TBOM-Gewichtungen bearbeiten*: Passen Sie den Testumfang und die Optimierungsreihenfolge entsprechend der Relevanz der einzelnen TBOMs an. Weitere Informationen erhalten Sie unter [Bearbeiten der TBOM-Gewichtung \[Seite 156\]](#).
- Registerkarte **BPCA-Testumfangsoptimierung**:
  - *Erforderliche Standardzeit für Testfalltypen*: Definieren Sie die Standardzeiten zum Anlegen von Testfällen je nach Testfalltyp. Weitere Informationen finden Sie unter [Festlegen von Standard-Erstellungszeiten für Testfälle \[Seite 185\]](#).
  - *Testumfangsoptimierung: Befreiungen*: Schließen Sie Objekte von der Optimierung des Testumfangs aus, um sie in ihrem jeweiligen Knotenkontext zu testen. Weitere Informationen finden Sie unter [Ausschließen von Objekten von der Optimierung \[Seite 200\]](#).

## 7.2 Vorbereitung: Anlegen von technischen Stücklisten (TBOMs)

### Verwendung

Das Werkzeug BPCA analysiert die Geschäftsprozesse, die von Änderungen hinsichtlich TBOMs betroffen sind. Um den BPCA nutzen zu können, sind Vorbereitungen in Ihrem System erforderlich: Erfassen Sie die TBOMs, und nehmen Sie Filter- und Gewichtungseinstellungen vor.

Bitte beachten Sie die folgenden erforderlichen Vorbereitungen:

- Ordnen Sie der Lösungsdokumentation ausführbare Einheiten zu.
- Legen Sie für jede ausführbare Einheit eine Liste der beteiligten TBOM-Objekte an.
- Definieren Sie Testfälle für die Lösungsdokumentation.

Dieser Prozess führt Sie durch die erforderlichen Schritte.

Weitere Informationen finden Sie unter [Business Process Change Analyzer \(BPCA\) \[Seite 130\]](#).

### Voraussetzungen

- Sie haben eine Lösungsdokumentation angelegt, die Geschäftsszenarios, Geschäftsprozesse und Prozessschritte enthält.
- Die Systemlandschaft der Lösung wurde konfiguriert.
- Sie haben in dem verwalteten System, in dem Ihre Prozesse laufen, mindestens `ST-PI 2008_1 SP14` oder `ST-PI 740 SP04` installiert.

- Sie haben die erforderlichen Berechtigungen, um TBOMs zu ändern. Weitere Informationen erhalten Sie im Security Guide zu SAP Solution Manager unter <http://help.sap.com/solutionmanager72>.

## Prozess

1. Ordnen Sie in Ihrer Lösungsdokumentation ausführbare Einheiten und Testfälle den entsprechenden Knoten Ihrer Geschäftsprozesse zu.
2. Legen Sie TBOMs für alle ausführbaren Einheiten an.  
Stattdessen können Sie auch die TBOM-Erfassung an den zuständigen Geschäftsprozess-Experten übergeben, der die TBOMs anlegt.
3. Optional: Konfigurieren Sie die TBOM-Gewichtung, um Änderungsanalysen auf die wichtigsten Objekte zu beschränken und so den Testaufwand zu begrenzen.
4. Identifizieren Sie gemäß Ihren Entwicklungszyklen ausführbare Einheiten, deren TBOMs fehlen oder veraltet sind. Legen Sie für diese TBOMs an, oder aktualisieren Sie deren TBOMs.

## Ergebnis

Ihre Lösungsdokumentation enthält Testfälle und ausführbare Einheiten mit aktuellen TBOMs.

Sie können nun den Business Process Change Analyzer ausführen und anhand der Ergebnisse die notwendigen Testpläne erzeugen.

## 7.2.1 Was ist eine Technical Bill of Materials (TBOM)?

### Definition

Eine TBOM enthält eine Liste aller Objekte, die an einer bestimmten ausführbaren Einheit beteiligt sind. Diese Objekte umfassen Elemente der Benutzungsoberfläche, Modul-Pools, Funktionsbausteine und Tabellen.

Diese Objekte sind jeweils einer sogenannten **TBOM-Erweiterung** zugeordnet. Bereits beim Erstellen einer TBOM legt das System eine erste TBOM-Erweiterung an, die alle Objekte dieser initialen Aufzeichnung enthält. Sie können einer TBOM eine beliebige Anzahl von Erweiterungen hinzufügen und somit die Liste der relevanten Objekte immer weiter erweitern.

TBOM-Erweiterungen können **dynamisch**, **statistisch** oder **semidynamisch** generiert werden. Weitere Informationen über die Erstellungsarten finden Sie unter [Auswählen der TBOM-Arten \[Seite 136\]](#).

### Verwendung

Mit einer TBOM kann der BPCA feststellen, ob eine ausführbare Einheit von einer Änderung im System (z.B. von geänderten Objekten in einem Transportauftrag) betroffen ist oder nicht.

# Struktur

## Informationen zur TBOM-Erweiterung

- Bezeichnung der Erweiterung
- Art der Erweiterung (dynamisch, statisch oder automatischer Testfall)
- Logische Komponente
- Name des verwalteten Systems
- Mandant des verwalteten Systems
- Status der TBOM-Erweiterung
- Anzahl der in der Erweiterung enthaltenen Objekte
- Produktversion des verwalteten Systems
- Systemrolle des verwalteten Systems
- Ersteller und Zeitpunkt der Erstellung
- Name des automatischen Testfalls, über den die Erweiterung erstellt wurde

Zusätzlich speichert das System zu jeder TBOM-Erweiterung folgende Datumsangaben:

- Erstellungsdatum
- Letztes Änderungsdatum
- Datum der letzten Aktualitätsprüfung

Diese Datumsangaben werden bei der Aktualitätsprüfung verwendet. Anhand dieser Datumsangaben legt das System den Umfang der Transportaufträge fest, deren Objekte überprüft werden müssen.

## Informationen zu den enthaltenen Objekten

Die TBOM enthält außerdem folgende Informationen (sichtbar in der TBOM-Content-Anzeige) über die Objekte in der ausführbaren Einheit: die folgenden Informationen (auch verfügbar im Dialogfenster „TBOM“ auf der Registerkarte „Erweiterungen“) werden mit jeder TBOM-Erweiterung gespeichert:

- **Programm-ID:** ID des Objekts. Beispiel: `R3TR`. Die folgenden Informationen (auch verfügbar im Dialogfenster „TBOM“ auf der Registerkarte „Erweiterungen“) für komplexe Objekte wie Programme und alle zugehörigen Elemente, `LIMU` für Teilobjekte
- **Objekttyp:** Zuordnung des Objekts zu einem Typ. Z.B. `REPS` für Report Source, `FUNC` für Funktionen, `TABL` für Tabellen.  
Es gibt mehrere hundert verschiedene Objekttypen.
- **Objektname:** Technischer Name des Objekts

Programm-ID, Objekttyp und Objektname ergeben zusammen eine eindeutige Kennzeichnung des Objekts.

- **Herkunft:** Herkunft des Objekts
  - D für dynamisch ermittelte Objekte
  - S für statisch ermittelte Objekte
- **Klassifikationstyp:** Unterschiedliche Objekttypen werden zu einem Klassifikationstyp zusammengefasst. Das System speichert die standardmäßige Zuordnung von Klassifikationstypen und Objekttypen. Folgende Klassifikationstypen sind vordefiniert:
  - `TABC` = Tabellenklasse
  - `UI` = User Interface
  - `TRAN` = Business Transaction
  - `FUNC` = Business Function

- DOKU = Technical Documentation Object
- DDIC = Data Dictionary
- AUTH = Authorization Object
- CUSX = Customer Extension Points (EnhSpots)
- MISC = sonstige Klassifikationstypen
- **Klassifikationswert:** Objekte vom Klassifikationstyp TABC(Tabellen) werden anhand der Klassifikationswerte folgendermaßen weiter unterteilt:
  - A = Anwendungstabelle (Stamm- und Bewegungsdaten)
  - C = Customizing-Tabelle (Eingaben und Änderungen nur durch Kunden, kein SAP-Import)
  - L = Tabelle für temporäre Daten (wird leer ausgeliefert)
  - G = Customizing-Tabelle (neue Einträge nur durch SAP, bestehende Einträge im Kundensystem sind geschützt)
  - E = Steuerungstabelle (SAP und Kunde haben eigene Key-Bereiche)
  - S = Systemtabelle (Eingaben und Änderungen nur durch SAP, Änderung = Modifikation)
  - W = Systemtabelle (Inhalt über eigene TR-Objekte transportierbar)
- **Paket:** Entwicklungsklasse
- **Softwarekomponente:** Name der Anwendung, aus der das Objekt stammt, z.B. SAP\_BASIS.
- **Objekttyp:** Bezeichnung aus der Versions-Datenbank. Da dort Referenzobjekte verwendet werden, kann die Bezeichnung gelegentlich vom oben genannten Objekttyp abweichen.
- **Objektname:** Bezeichnung aus der Versions-Datenbank. Da dort Referenzobjekte verwendet werden, kann die Bezeichnung vom obigen Objektname abweichen.
- **Tabellenschlüssel:** Bei Zugriffen auf Datenbanktabellen werden zusätzlich zu den Datenbanktabellen auch die Werte der Schlüsselfelder erfasst, die für den Tabellenzugriff verwendet werden. Dadurch kann das System die Anzahl der Treffer bei der Änderungseinfluss-Analyse reduzieren.

### i Hinweis

Es werden nur Zugriffe auf folgende Tabellenarten unterstützt:

- C = Customizing-Tabelle
- G = Customizing-Tabelle
- E = Steuerungstabelle
- S = Systemtabelle
- W = Systemtabelle

Ob das System für einen Tabellenzugriff einen Schlüsselwert aufzeichnen kann, hängt außerdem auch von der Art des Zugriffs ab. Folgende Zugriffsarten werden unterstützt:

- Alle Zugriffe, die über den SAP Tabellenpuffer abgedeckt werden
- Zugriffe, bei denen mindestens ein Schlüsselfeld spezifiziert ist

Folgende Zugriffsarten werden aktuell nicht unterstützt:

- Zugriffe mit der SQL-Anweisung `SELECT FOR ALL ENTRIES`
- Zugriffe mit Ranges-Tabellen
- Zugriffe mit Join-Ausdrücken

Wenn das System bei einem Tabellenzugriff keinen Schlüsselwert aufzeichnen kann, enthält die TBOM nur den Tabellennamen. In der Änderungseinfluss-Analyse führt jeder geänderte Eintrag in solchen Tabellen zu Treffern in allen ausführbaren Einheiten, die auf diese Tabellen zugreifen.

Weitere Informationen zu Einschränkungen finden Sie im SAP-Hinweis [2621178](#) (Restrictions for Table Key Recording in Dynamic TBOMs).

## Integration

TBOMs werden in der Sicht *Lösungsdokumentation* den ausführbaren Einheiten zugeordnet.

**Dynamische TBOMs:** In Systemen mit Kernel 620/640 können dynamische TBOMS nur nach dem Status `ST-PI 2008_01 SP03` angelegt werden. In allen anderen Systemen (Kernel 46C oder höher als 640) können dynamische TBOMS angelegt werden.

**Systemübergreifende TBOMs:** Wenn in einer ausführbaren Einheit andere Systeme angesprochen werden (z.B. RFC), dann werden auch die Objekte der anderen Systeme mit aufgezeichnet. Für diese Objekte wird pro System jeweils eine eigene TBOM-Erweiterung angelegt. Diese Funktion ist verfügbar, wenn alle beteiligten verwalteten Systeme einen Kernelstand von 700 oder höher haben.

**Webanwendungen:** Für die Aufzeichnung von TBOMs für Webanwendungen müssen die verwalteten Systeme einen Kernelstand von 700 oder höher haben.

**Statische TBOMs:** Statische TBOMs können für verwaltete Systeme ab `ST-PI 2008_01` generiert werden.

## 7.2.2 TBOM-Arten auswählen

### Voraussetzungen

- Sie haben eine Lösungsdokumentation angelegt. Sie haben den Knoten Testfälle und ausführbare Einheiten zugeordnet.
- Um TBOMs semidynamisch mit SCMON anzulegen, nutzen Sie Folgendes:
  - SAP Solution Manager 7.2 SPS 1 (`ST 720 SP03`)
  - Verwaltetes System:
    - `SAP_BASIS 750` oder `SAP_BASIS 740 SP14` und `SAP_KERNEL 742 Patch Level 310`
    - `ST-PI 2008_1 SP14` oder `ST-PI 740 SP04`

Mit SAP Solution Manager 7.2 SP05 ändern sich die Voraussetzungen für das verwaltete System wie folgt.

- `SAP_KERNEL 7.22 Patch-Level 123` oder `SAP_KERNEL 742 Patch-Level 310` oder `SAP_KERNEL 7.45+`
- `ST-PI 2008_1 SP14` oder `ST-PI 740 SP04`

Weitere Informationen zu den Anforderungen für UPL finden Sie in SAP-Hinweis [1828848](#).

Um zu prüfen, ob das UPL richtig konfiguriert ist, führen Sie einen BPCA-Selbstcheck aus. Weitere Informationen finden Sie unter [Prüfen der Voraussetzungen für die Generierung von semidynamischen TBOMs \[Seite 142\]](#).

- Um TBOMs semidynamisch mit UPL anzulegen, nutzen Sie Folgendes:
  - SAP Solution Manager 7.2 SPS 1 (`ST 720 SP01`) oder höher



- Verwaltetes System:
  - SAP\_BASIS 701 SP10 und Kernel 720 Patch 94, SAP\_BASIS 702 SP09 und Kernel 720 Patch 94, SAP\_BASIS 703 SP03
  - ST-PI 2008\_1\_xx\_SP12

Weitere Informationen zu den Anforderungen für UPL finden Sie in SAP-Hinweis [1828848](#).

Um zu prüfen, ob das UPL richtig konfiguriert ist, führen Sie einen BPCA-Selbstcheck aus. Weitere Informationen finden Sie unter [Prüfen der Voraussetzungen für die Generierung von semidynamischen TBOMs \[Seite 142\]](#).

- Sie haben folgende Sonderfälle ermittelt und deren Anforderungen angelegt. Weitere Informationen finden Sie unter [Erfüllen der Voraussetzungen für Sonderfälle der TBOM-Aufzeichnung \[Seite 143\]](#).

## Kontext

Eine **TBOM (technische Stückliste)** [\[Seite 133\]](#) enthält alle Objekte in einer ausführbaren Einheit. Mithilfe von TBOMs kann der Business Process Change Analyzer (BPCA) feststellen, ob eine ausführbare Einheit von einer Änderung betroffen ist oder nicht. Um den BPCA verwenden zu können, legen Sie eine TBOM für jede ausführbare Einheit in Ihrer Lösungsdokumentation an.

Zum Anlegen einer TBOM haben Sie folgende Möglichkeiten (absteigende Sortierung nach Genauigkeit):

- **Dynamisch:**  
Dynamische TBOMs sind am genauesten, können allerdings nicht automatisch generiert werden. Sie können sie z.B. für geschäftskritische Transaktionen verwenden.  
Zeichnen Sie beispielsweise eine Transaktion auf, während diese läuft. Der relevante RFC-Benutzer im verwalteten System ruft die zugehörige ausführbare Einheit auf. Sie benötigen im verwalteten System die Berechtigung zum Ausführen der ausführbaren Einheit. Geben Sie im verwalteten System die für die ausführbare Einheit erforderlichen Daten ein.  
Alle Objekte in der ausführbaren Einheit werden im Hintergrund erfasst und nach Ausführung der ausführbaren Einheit in die TBOM übernommen. Die dynamische Erstellung umfasst auch RFC-Verbindungen und verzweigte GUI-Abläufe des Benutzers innerhalb desselben (primären) Systems. Die dynamische Erstellung umfasst RFC-Verbindungen zu anderen (sekundären) Systemen. Bei der dynamischen Erstellung werden auch Erweiterungen angelegt. Weitere RFC-Verbindungen in Drittsysteme werden ebenfalls aufgezeichnet, nur verzweigte GUI-Abläufe innerhalb der Sekundär- oder Drittsysteme sind in der Aufzeichnung nicht mehr enthalten.  
Erweiterungen werden auch bei Einplanung von Hintergrundjobs durch die ausführbare Einheit angelegt. Das System legt eine separate TBOM-Erweiterung für jeden Hintergrundjob mit den beteiligten Objekten an.

### i Hinweis

Durch das Ausführen der ausführbaren Einheiten werden neue Datensätze im verwalteten System angelegt. Beachten Sie dies beim Anlegen von TBOMs in einem Produktivsystem.

### i Hinweis

Bei der Aufzeichnung einer dynamischen TBOM werden alle Aktionen im verwalteten System aufgezeichnet, die von demselben Benutzer ausgeführt werden. Daher sollte dieser Benutzer während der TBOM-Aufzeichnung alle Aktionen unterlassen, die nicht aufgezeichnet werden sollen.

- **Semi-dynamisch**

Semidynamische TBOMs werden per Massenvorgang generiert, und sie sind präziser als statische TBOMs. Wie beim Generieren einer statischen TBOM analysiert das System die Umgebung, um zu ermitteln, welche Objekte sich in der ausführbaren Einheit befinden. Nur tatsächlich verwendete Objekte sind in der TBOM vorhanden.

Ab Solution Manager 7.2 SPS 3 können semidynamische TBOMs auf dem ABAP-Aufrufmonitor (Transaktion SCMON) basieren. SCMON erfasst ebenso wie Usage and Procedure Logging (UPL) den Einstiegspunkt, von dem ein Objekt verwendet wird. SCMON verbessert die Verbindung der verwendeten Objekte zu den ausführbaren Einheiten. Darüber hinaus können auf SCMON basierende semidynamische TBOMs auch dynamisch aufgerufene Objekte erfassen.

Semidynamische TBOMs können auch UPL (Usage and Procedure Logging) verwenden. Auf UPL basierende semidynamische TBOMs sind weniger präzise als TBOMs aus SCMON. Die UPL-Funktion Ihres Systems stellt die Objektverwendungsdaten zum Filtern von Objekten bereit.

Semidynamische TBOMs können auch UPL verwenden. Auf UPL basierende semidynamische TBOMs sind weniger präzise als TBOMs aus SCMON. Die UPL-Funktion Ihres Systems stellt die Objektverwendungsdaten zum Filtern von Objekten bereit.

- **Statisch:**

Bei der statischen (wie bei der semidynamischen) TBOM-Massengenerierung wird die ausführbare Einheit nicht aufgerufen. Das System prüft mit einer Umfeldermittlung, welche Objekte an der ausführbaren Einheit beteiligt sind. Sie benötigen also keine Berechtigungen im verwalteten System. Dynamische Aufrufe oder generierte Objekte werden nicht festgestellt und finden somit auch keinen Eingang in die TBOM. Das Ergebnis ist weniger präzise als bei der dynamischen oder semi-dynamischen TBOM-Massengenerierung.

### **i** Hinweis

Statische und semidynamische TBOMs unterstützen keine SAP-Web-Link-Anwendungen (`SAP_LONG_URL`) (z.B. die URL eines Portals, über das eine SAP-Anwendung in einem ABAP-Application-Server gestartet wird). Legen Sie in diesem Fall dynamische TBOMs an.

## Vorgehensweise

1. Um die komplette Bibliothek der ausführbaren Einheiten abzudecken, legen Sie statische oder semidynamische TBOMs per Massengenerierung an. Weitere Informationen finden Sie unter [Anlegen von statischen oder semidynamischen TBOMs \[Seite 140\]](#).
2. Identifizieren Sie ausführbare Einheiten, die genauere dynamische TBOMs benötigen:
  - ausführbare Einheiten in geschäftskritischen Geschäftsprozessen
  - ausführbare Einheiten, die SAP-Web-Links verwenden, die nicht von semidynamischen oder statischen TBOMs unterstützt werden.

Weitere Informationen finden Sie unter [Prüfen von TBOM-Abdeckung und -Status \[Seite 158\]](#).

3. Legen Sie je nach Aufwand oder bei fehlender Unterstützung von statischen oder semidynamischen TBOMs dynamische TBOMs an. Weitere Informationen finden Sie unter Referenz:
  - [Anlegen dynamischer TBOMs \[Seite 150\]](#)
  - [Aufzeichnen von TBOMs für Webanwendungen \[Seite 146\]](#)

## 7.2.3 Site-abhängige TBOMs

In der Lösungsdokumentation beziehen sich Site-abhängige TBOMs auf Situationen, in denen dasselbe SAP-Produkt für unterschiedliche Produktivsysteme verwendet wird, wie z.B. für Niederlassungen verschiedener Länder oder Regionen.

Jeder Standort („Site“) stellt ein anderes Produktivsystem dar, das von eigenen Entwicklungs- und Testsystemen mit Daten versorgt werden kann. Es ist auch möglich, dass einige Produktivsysteme dieselben Entwicklungssysteme gemeinsam nutzen. In Ausnahmefällen hat jeder Standort eine exklusive Gruppe von Systemen. Die Zuordnung der Systeme zu Sites nehmen Sie in der Systemlandschaft eines Branches vor.

Innerhalb der für Sites aktivierten Branches und logischen Komponentengruppe kann jede Site ihre eigene logische Komponente mit individuellen Systemen besitzen. Wenn Sites dieselben Systeme nutzen, können sich diese Sites eine logische Komponente teilen.

Jedem Element in der Lösungsdokumentation können Sie eine Site-Attribute zuordnen, um dieses Element für ausgewählte Sites gültig zu machen. Fehlt die Definition des Attributs, ist das Element für alle Produktivsysteme in einer logischen Komponentengruppe gültig.

Weitere Informationen finden Sie unter [Sites](#).

Ausführbare Elemente, die für mehr als eine Site gültig sind, können in den Systemen der verschiedenen Sites auf unterschiedliche Weise ausgeführt werden. So können ausführbare Elemente über verschiedene Konfigurationseinstellungen oder individuellen benutzerdefinierten Code unterschiedlich ausgeführt werden. Das bedeutet, dass sich die TBOMs, die für dieselben ausführbaren Einheiten an verschiedenen Standorten aufgezeichnet wurden, in ihrem Inhalt voneinander unterscheiden können. Ausführbare Einheiten an verschiedenen Standorten können jedoch auch identisch sein, und somit identische TBOMs besitzen.

Für TBOMs haben Sie folgende Möglichkeiten:

- Ein TBOM-Element kann genau einer Site zugeordnet werden
- Eine TBOM kann durch mehrere TBOM-Elemente, die über dieselbe TBOM GUID auf dieselbe TBOM verweisen, für mehrere Sites gültig sein.
- Fehlt das Site-Attribut, ist die TBOM universell für alle Sites gültig.

## Site-abhängige TBOMs anlegen

Beim Erstellen einer TBOM für eine ausführbare Einheit in einem Site-abhängigen Branch und einer Site-abhängigen logischen Komponentengruppe können Sie die folgenden Attribute definieren:

- **Site:** Zusammen mit der Systemrolle bestimmt die Aufzeichnungs-Site, in welchem System die TBOM aufgezeichnet wird.
- **Gültig für Sites:** Legt fest, für welche Sites die TBOM gültig ist. Als allgemeine Richtlinie sollten zur Abdeckung aller ausführbaren Einheiten und Sites so wenig TBOMs wie möglich erzeugt werden. Sie haben folgende Möglichkeiten:
  - **Sites mit derselben logischen Komponente wie die Aufzeichnungs-Site:** Diese Option ist nur verfügbar, wenn Sites vorhanden sind, die dieselbe logische Komponente wie die Aufzeichnungs-Site verwenden. Wählen Sie diese Option, wenn es mehrere Sites gibt, die dieselbe logische Komponente und damit dieselben Systeme nutzen. Alle diese Sites werden von einer TBOM abgedeckt.
  - **Nur Aufzeichnungs-Site:** Wählen Sie diese Option, wenn Sie sicher sind, dass die ausführbare Einheit eine spezielle Verhalten hinsichtlich der ausgewählten Aufzeichnungs-Site besitzt.

- *Site-unabhängig*: Eine Site-unabhängige TBOM wird für alle Sites verwendet, in denen keine Site-spezifische TBOM existiert.

Dieselben Optionen stehen in den folgenden Szenarien zur Verfügung:

- [Anlegen von statischen oder semidynamischen TBOMs \[Seite 140\]](#)
- [Aufzeichnen von TBOMs über automatisierte Testfälle \[Seite 152\]](#)
- [Aufzeichnen von TBOMs während manueller Tests \[Seite 154\]](#)

## 7.2.4 Anlegen von statischen oder semidynamischen TBOMs

### Verwendung

Legen Sie statische oder semidynamische TBOMS für die ausführbaren Einheiten in Ihrer Bibliothek der ausführbaren Einheiten an.

Das System prüft mit einer Umfeldermittlung der statischen TBOMs, welche Objekte an der ausführbaren Einheit beteiligt sind.

Wenn Ihr System *Usage and Procedure Logging* (UPL) unterstützt, können Sie TBOMs semidynamisch anlegen. Zusätzlich zu den Umfeldinformationen werden die Objekte anhand der Informationen zur Objektverwendung gefiltert, die die UPL-Funktion Ihres Systems liefert. Nur tatsächlich verwendete Objekte werden in der semidynamischen TBOM verwendet.

Wenn Ihr System den *ABAP-Aufrufmonitor* (SCMON) unterstützt und SCMON-Daten in den SAP Solution Manager extrahiert werden, erkennt der BPCA automatisch, ob SCMON-Daten für eine ausführbare Einheit verfügbar sind. Falls ja, erzeugt das System basierend auf SCMON semidynamische TBOMs. Andernfalls erzeugt das System basierend auf UPL semidynamische TBOMs.

### Voraussetzungen

Um semidynamische TBOMs generieren zu können, stellen Sie sicher, dass die Daten aus SCMON oder UPL im System erzeugt werden und abrufbar sind. Führen Sie dazu den BPCA-Selbstcheck wie in [BPCA-Selbstcheck für semidynamische TBOMs \[Seite 142\]](#) beschrieben durch

### Vorgehensweise

1. Starten Sie die Anwendung *Verwaltung - Änderungseinfluss-Analyse*, die über den Kachelkatalog im SAP Solution Manager Launchpad hinzugefügt werden kann.
2. Wählen Sie **► BPCA-Vorbereitung ► TBOM-Werkzeuge ► TBOMs generieren ►**.
3. Sie navigieren zum Bild *Statische und semidynamische TBOMs anlegen*. Geben Sie die Lösung, den Branch und etwaige Umfänge an, für die Sie TBOMs anlegen möchten. Wenn der ausgewählte Branch site-aktiviert ist, wählen Sie die Site und die Site-Gültigkeit der TBOM. Um TBOMs in mehreren Sites zu generieren, starten Sie einen TBOM-Aufzeichnungsjob für jede Site.

Wählen Sie die Systemrolle aus. Wählen Sie *Systeme anzeigen, aus denen TBOMs erstellt werden*, um die Systeme anzuzeigen, die der Systemrolle zugeordnet sind.

#### 4. *Aktualisierungsoptionen*

Auf Anfrage des SAP-Supports oder wenn bei der TBOM-Generierung Fehler auftreten, können Sie die Aktualisierungsoptionen verwenden:

- *Veraltete TBOMs aktualisieren.*
- *Alle statischen o. semidynamischen TBOMs neu anlegen.* Mit dieser Option erzeugt das System nicht nur fehlende statische oder semidynamische TBOMs, sondern aktualisiert auch alle bereits vorhandenen TBOMs. Wenn zu einer TBOM sowohl dynamische als auch statische Erweiterungen vorliegen, werden nur bereits existierende statische Erweiterungen dieser TBOM aktualisiert, es werden jedoch keine neuen statischen Erweiterungen angelegt.
- *Nur veraltete TBOMs aktualisieren.*
- *Fehlende TBOMs anlegen* Anlegen von TBOMs für alle ausführbaren Einheiten, zu denen es noch keine TBOM gibt.
- *Semidynamische TBOMs anlegen* Um genauere Ergebnisse zu erzielen, ist diese Option standardmäßig markiert.
  - Standardwert: Wenn Sie SCMON verwenden, werden TBOMs nur für ausführbare Einheiten in SCMON angelegt. Dies ist die effizienteste und zuverlässigste Art, da jetzt nur relevante Elemente berücksichtigt werden. Für ausführbare Einheiten, die nicht in SCMON erfasst sind, werden keine TBOMs generiert.
  - Falls SCMON nicht aktiv ist, werden die UPL-Verwendungsdaten zum Anlegen von TBOMs verwendet.  
Es wird jedoch empfohlen, die Sammlung von SCMON-Daten mit Stamminformationen zu aktivieren, um die maximale Genauigkeit semidynamischer TBOMs sicherzustellen.
  - Falls die SCMON-Daten aus irgendeinem Grund nicht ausreichend sind, können Sie die Option auswählen, um UPL-TBOMs anzulegen.
  - Wenn keine semidynamische TBOM angelegt werden kann, können Sie festlegen, dass stattdessen eine statische TBOM angelegt wird.
  - Wenn Sie die Laufzeit verringern möchten, können Sie den Verzweigungsgrad einschränken. Legen Sie einen Verzweigungsgrad von mindestens 3 fest. Wenn das System auf den ersten beiden Verzweigungsgraden keine UPL-Daten findet, wird eine statische TBOM erstellt.

#### 5. *Semidynamische Optionen:*

- Um das System festzulegen, dessen SCMON- oder UPL-Daten verwendet werden, wählen Sie die Systemrolle. Für realistische Ergebnisse verwenden Sie das Produktivsystem.
- Definieren Sie den Zeitraum (Monat und Jahr) für die Datensammlung. Stellen Sie sicher, dass die Ausführungen der Transaktionen, die Sie aufzeichnen möchten, enthalten sind.  
Um beispielsweise Periodenabschlussaktivitäten zu erfassen, wählen Sie eine Zeitspanne von mindestens zwei Monaten.
- Wählen Sie *Systeme einblenden, aus denen UPL/SCMON-Daten abgerufen werden*, um die Verfügbarkeit und Qualität der UPL/SCMON-Daten für die beteiligten Systeme zu prüfen. Sind die Lücken zu groß, könnten Objekte in den UPL-Daten und somit auch in den TBOMs fehlen. Sind keine UPL-Daten verfügbar, führen Sie einen BPCA-Selbstcheck durch.
- Wenn Sie Systemrolle oder -datum geändert haben, wählen Sie *Liste und Verfügbarkeit der UPL-/SCMON-Systeme aktualisieren*.

#### 6. Um das Programm auszuführen, wählen Sie *Ausführen*.

Empfehlung: Für Lösungen oder Branches mit vielen ausführbaren Einheiten sollte das Programm als Hintergrundjob eingeplant werden. Definieren Sie das Intervall abhängig davon, wie häufig Änderungen implementiert werden (z. B. monatlich oder vierteljährlich).

UPL-Daten werden standardmäßig automatisch aktualisiert. Bei Bedarf können Sie z.B. ein bestimmtes Intervall zur UPL-Datensammlung festlegen, um die Verwendung in bestimmten Zeiträumen abzudecken.

## 7.2.4.1 BPCA-Selbstcheck für semidynamische TBOMs

### Verwendung

Um semidynamische TBOMs generieren zu können, müssen Sie sicherstellen, dass die Daten aus UPL oder SCMON im System erzeugt werden und abrufbar sind. Mit einem Selbstcheck-Report werden die folgenden Voraussetzungen überprüft:

- Sie haben das UPL oder SCMON in der SAP-Solution-Manager-Konfiguration unter [Verwaltung kundeneigener Entwicklungen](#) [Infrastruktur konfigurieren](#) [Extraktoren aktivieren](#) im Zielsystem aktiviert. Um aussagekräftige Daten zu erhalten, ziehen Sie die Verwendung eines Produktivsystems in Betracht.
- Es bestehen RFC-Verbindungen zwischen den beteiligten Systemen.
- Zwischen dem SAP-Solution-Manager-System und dem Zielsystem ist eine BW-Extraktion eingerichtet.
- SCMON ist aktiv.
- In der SAP-Solution-Manager-Konfiguration (Transaktion SOLMAN\_SETUP) wurde unter [Verwaltung kundeneigener Entwicklungen](#) [Einstellungen für periodische Aufgaben](#) das Wurzel-Flag gesetzt.
- Sie haben die SCMON-Sammlung (Job SM: SCMON\_CONTROL) eingeplant und den Job SM: SCMON\_UPLOAD\_STATUS\_HK in SOLMAN\_SETUP unter [Obligatorische Konfiguration](#) [Basiskonfiguration](#) [Jobs einplanen](#) eingerichtet.

### Vorgehensweise

1. Starten Sie die Sicht [Verwaltung - Änderungseinfluss-Analyse](#).
2. Wählen Sie [BPCA-Vorbereitung](#) [TBOM-Werkzeuge](#) [BPCA-Selbstchecks ausführen](#).
3. Geben Sie folgende Daten ein:
  - **Lösung:** Geben Sie die Lösung ein, für die Sie semidynamische TBOMs anlegen möchten.
  - **Verwaltetes System für die TBOM-Generierung:** Geben Sie das verwaltete System ein, in welchem Sie semidynamische TBOMs aufzeichnen möchten.  
Achtung: Die TBOM-Generierung lastet das System stark aus, **verwenden Sie deshalb nicht Ihr Produktivsystem zur TBOM-Generierung.**
  - **System mit UPL/SCMON-Daten:** Geben Sie das SAP-Solution-Manager-System ein, für das Sie semidynamische TBOMs anlegen möchten. Üblicherweise verwenden Sie das Produktivsystem.
4. Wählen Sie [Selbstcheck ausführen](#). Es wird eine Auswertung erzeugt.
5. Analysieren und lösen Sie die Issues.

## 7.2.5 Voraussetzungen für Sonderfälle der TBOM-Aufzeichnung

### Kontext

Bestimmte dynamische TBOMs haben Erweiterungen, Erweiterungen haben beispielsweise Transaktionen, die über mehrere Systeme hinweg ausgeführt werden, oder Transaktionen, die Hintergrundjobs starten. Sie können diese Sonderfälle identifizieren und die Voraussetzungen für die TBOM-Erfassung erfüllen. Informieren Sie sich, inwiefern dieser Prozess vom Standardprozess abweicht.

*Vorgehensweise*

### Vorgehensweise

1. Identifizieren Sie in Ihrer Lösung ausführbare Einheiten, die
  - Hintergrundjobs auslösen
  - über mehrere Systeme hinweg ausgeführt werden
2. Stellen Sie abhängig vom Anwendungsfall sicher, dass die Voraussetzungen erfüllt sind und Sie mit den Unterschieden zwischen diesem Prozess und dem Standardprozess vertraut sind:
  - [Aufzeichnen von Batch-TBOMs \[Seite 143\]](#)
  - [Aufzeichnen systemübergreifender TBOMs \[Seite 145\]](#)

*Nächste Schritte*

Für bestimmte Vorgänge sind Erweiterungen erforderlich, Sie können sie jedoch aufzeichnen, ohne spezielle Voraussetzungen anzulegen. Weitere Informationen erhalten Sie unter [Aufzeichnen einer dynamischen TBOM in mehreren Läufen \[Seite 152\]](#).

### 7.2.5.1 Aufzeichnen von Batch-TBOMs

### Kontext

Um TBOMs für ausführbare Einheiten anzulegen, die einen oder mehrere Hintergrundjobs aufrufen, stellen Sie sicher, dass die erforderlichen Vor- und Nachbereitungen getroffen wurden.

*Vorgehensweise*

## Vorgehensweise

1. Stellen Sie Folgendes sicher:
  - Die RFC-Verbindung zum verwalteten System erfordert keine manuelle Anmeldung. Um eine Batch-TBOM ohne RFC-Verbindung anzulegen, lösen Sie die Batch-TBOM-Sammlung manuell aus. Weitere Informationen finden Sie unter [Status einer Batch-TBOM prüfen \[Seite 160\]](#).
  - Sie haben in SAP-Hinweis [1684704](#) geprüft, ob es weitere Anforderungen gibt.
2. Erstellen Sie eine dynamische TBOM. Folgen Sie den allgemeinen Anleitungen zum Aufzeichnen einer TBOM. Weitere Informationen finden Sie unter [Anlegen dynamischer TBOMs \[Seite 150\]](#).

Ergebnis: Das System legt eine TBOM-Erweiterung mit den Objekten an, die von der ausführbaren Einheit verwendet werden.

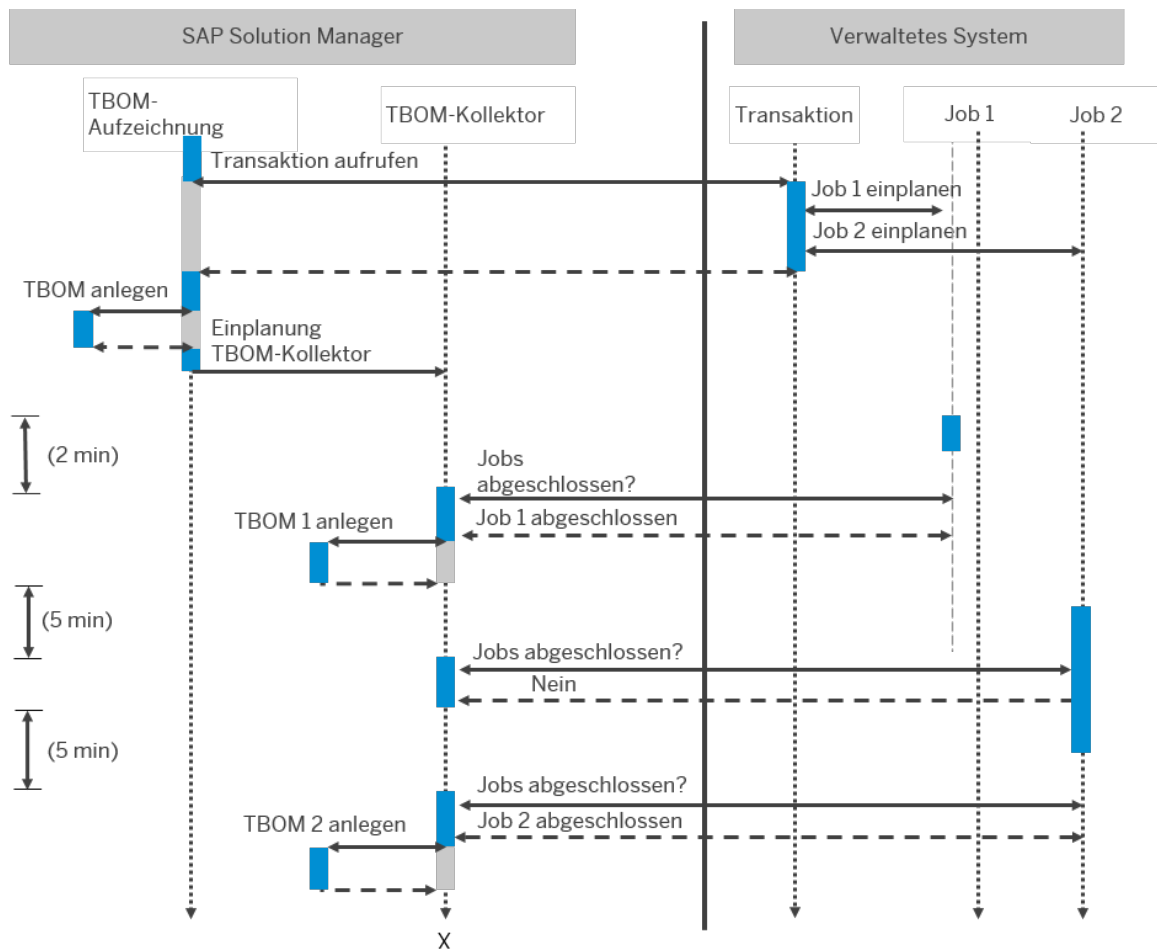
Das System plant einen Hintergrundjob für den TBOM-Sammler im SAP-Solution-Manager-System ein. Als Zeitpunkt der ersten Ausführung wählt das System zwei Minuten nach dem Beginn des ersten Hintergrundjobs im verwalteten System.

Der TBOM-Sammler prüft, ob die eingeplanten Hintergrundjobs im verwalteten System bereits ausgeführt wurden. Wenn ein Hintergrundjob abgeschlossen ist, erstellt der TBOM-Sammler für die vom Job betroffenen Objekte eine separate TBOM-Erweiterung. Solange noch nicht alle Jobs abgeschlossen sind, plant das System den TBOM-Sammler alle fünf Minuten ein.

Wenn alle Jobs im verwalteten System abgeschlossen sind und bei der Ausführung des TBOM-Sammlers Fehler auftreten, führt das System ihn zwei weitere Male aus. Nach insgesamt drei Fehlversuchen plant das System den TBOM-Sammler aus. Fehlerversuche kommen beispielsweise zustande, wenn für die RFC-Verbindung eine manuelle Anmeldung erforderlich ist.

Das folgende Flussdiagramm zeigt eine schematische Darstellung dieses Prozesses.





Anlegen einer Batch-TBOM

3. Prüfen Sie den Status der Batch-TBOM. Weitere Informationen finden Sie unter [Prüfen des Status einer Batch-TBOM \[Seite 160\]](#).

## 7.2.5.2 Aufzeichnen systemübergreifender TBOMs

### Kontext

Wenn ausführbare Einheiten systemübergreifend ausgeführt werden, führen Sie die Systemlandschaftsvorbereitung durch.

## Vorgehensweise

1. Identifizieren Sie Vorgänge, die systemübergreifend durchgeführt werden, sowie die zugehörigen Systeme.
2. Sie müssen Folgendes sicherstellen:
  - Alle beteiligten verwalteten Systeme haben einen Kernelstand von 700 oder höher.
  - Alle beteiligten verwalteten Systeme haben ST-PI 2008\_01 SPO3 oder höher.
  - Alle beteiligten verwalteten Systeme haben dieselbe Systemrolle wie das Hauptsystem der ausführbaren Einheit.
  - Die logischen Komponenten der verwalteten Systeme sind in die Systemlandschaft eingebunden.
  - Der Zugriff auf weitere Systeme erfolgt über RFC oder aus dem SAP Web Application Server ABAP über HTTP.
3. Folgen Sie den Anweisungen zum Aufzeichnen einer TBOM. Weitere Informationen finden Sie unter [Anlegen dynamischer TBOMs \[Seite 150\]](#).

## Ergebnisse

Vor Beginn der TBOM-Aufzeichnung meldet sich SAP Solution Manager bei allen Systemen in der Systemlandschaft an, die dieselbe Systemrolle wie die Hauptkomponente der ausführbaren Einheit haben.

## 7.2.5.3 Aufzeichnen von TBOMs für Webanwendungen

### Voraussetzungen

- Sie haben die Lösungsdokumentationsknoten oder Typen von ausführbaren Einheiten identifiziert, die Webanwendungen verwenden.  
Weitere Informationen finden Sie unter [Prüfen von TBOM-Abdeckung und -Status \[Seite 158\]](#).
- Sie kennen den Webanwendungsbenutzer, der eine Verbindung zum SAP-ABAP-System herstellt.

### Kontext

Um TBOMs für ausführbare Einheiten anzulegen, die nicht im klassischen SAP GUI ausgeführt werden, stellen Sie sicher, dass die Voraussetzungen erfüllt sind, und machen Sie sich mit den Unterschieden zum Standardprozess vertraut.

Folgende Arten von Webanwendungen werden von **semidynamischen** oder **statischen** TBOMs unterstützt:

- Web Dynpro
- BSP

- CRM WebClient
- CRM People-Centric
- SAP-Fiori-Anwendungen (nur dynamische TBOMs)

Bitte beachten Sie: *SAP Web-Link* (`SAP_LONG_URL`) wird von statischen oder semidynamischen TBOMs **nicht** unterstützt. Legen Sie in diesem Fall eine **dynamische** TBOM an. Sie können eine Portal-URL angeben, über die eine SAP-Anwendung auf einem ABAP-Application-Server startet.

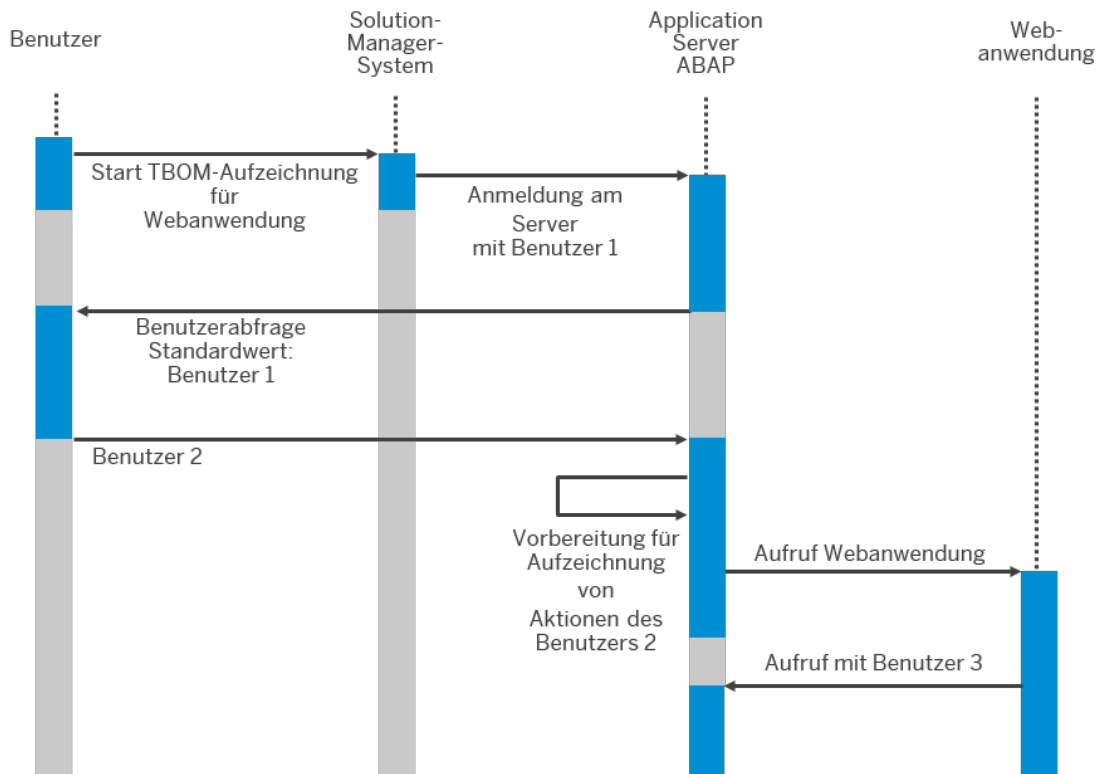
## Vorgehensweise

1. Stellen Sie Folgendes sicher:
  - In der Anwendung *SAP Web-Link* startet die URL eine SAP-Anwendung auf einem ABAP-Anwendungsserver.
  - Die verwalteten Systeme haben einen Kernelstand von 700 oder höher.
  - Die verwalteten Systeme haben ST-PI 2008\_01 SPO3 oder höher.
2. Folgen Sie den Anweisungen zum Aufzeichnen einer TBOM. Weitere Informationen finden Sie unter [Anlegen dynamischer TBOMs \[Seite 150\]](#).
3. Wählen Sie *Aufzeichnung starten*.
4. Geben Sie den Benutzer an.

Bevor die ausführbare Einheit gestartet wird, fordert Sie das System zur Eingabe eines Benutzers auf. Als Vorschlagswert wird der Benutzer angegeben, mit dem sich das SAP-Solution-Manager-System beim ABAP-Anwendungsserver angemeldet hat („Benutzer 1“ in nachfolgender Grafik).

Geben Sie hier den Benutzer an, mit dem die Webanwendung auf den ABAP-Anwendungsserver zugreift („Benutzer 3“ in nachfolgender Grafik).

Der Benutzer 3 wird von der Webanwendung festgelegt. Der von Ihnen angegebene zweite Benutzer muss mit Benutzer 3 identisch sein, um die Aufzeichnung zu gewährleisten.



Behandlung der Benutzer bei der TBOM-Aufzeichnung für Webanwendungen

5. Führen Sie die ausführbare Einheit aus.
6. Schließen Sie das Browserfenster, in dem Sie die ausführbare Einheit ausgeführt haben.

#### **i** Hinweis

Sie können nur dann sicher sein, dass alle Objekte in der ausführbaren Einheit aufgezeichnet werden, wenn Sie das Browserfenster schließen, bevor Sie die Aufzeichnung beenden.

7. Wählen Sie im Dialog *Aufzeichnung für <Typ der ausführbaren Einheit>* die Drucktaste *Aufzeichnung beenden*.

## 7.2.6 Delegieren von TBOM-Aufzeichnungen an Geschäftsexperten

### Verwendung

Ein Qualitätsexperte kann die Aufzeichnung von dynamischen TBOMs an einen Geschäftsprozess-Experten oder Tester delegieren, indem er eine TBOM-Aufgabe anlegt.

## Voraussetzungen

- Sie haben die Voraussetzungen zum Aufzeichnen von Sonderfällen von TBOMs erfüllt. Weitere Informationen finden Sie unter [Erfüllen der Voraussetzungen für Sonderfälle der TBOM-Aufzeichnung \[Seite 143\]](#).
- Sie haben die folgenden Aktionen ausgeführt:
  - BPCA-Arbeitsvorrats-Workflow eingerichtet
  - Benutzer für Geschäftsprozess-Experten angelegtWeitere Informationen zu den Aktivitäten finden Sie im Customizing für SAP Solution Manager unter [▶ Capabilities ▶ Test-Suite ▶ Business Process Change Analyzer ▶ Setup ▶ TBOM-Aufgaben ▶](#).

## Vorgehensweise

### Automatisches Anlegen mehrerer Aufgaben

1. Wählen Sie in der Sicht *Lösungsdokumentation* die Sicht *Liste*.
2. Führen Sie einen Drilldown zu dem Knoten durch, für den TBOMs angelegt, neu aufgezeichnet oder erweitert werden sollen (z.B. *Bibliotheken*).
3. Geben Sie Filterattribute an (z.B. *Ausführbarer TBOM-Typ* = **semidynamisch** und *Ausführbarer TBOM-Status* = **fehlt**).
4. Wählen Sie mindestens ein Objekt aus.
5. Wählen Sie *TBOM-Aufgaben anlegen* aus dem Kontextmenü.
6. Ordnen Sie den Geschäftsexperten wie unten beschrieben zu.

### Anlegen einer Wiederaufzeichnungs-Aufgabe aus TBOM

Legen Sie eine einzelne Aufgabe an, wenn eine bestimmte TBOM veraltet ist oder erweitert werden muss (z.B. wenn ein Vorgang systemübergreifend oder in mehreren Läufen ausgeführt wird).

1. Führen Sie in der Sicht *Test-Suite - Testvorbereitung* einen Drilldown zur TBOM aus, und wählen Sie deren Namen aus, um sie zu öffnen.  
Die Sicht *Technische Stückliste* wird geöffnet.
2. Wählen Sie unter *Aufgabe* den Eintrag *TBOM-Aufzeichnungsaufgabe anlegen*.  
Sie gelangen zur Sicht *TBOM-Aufzeichnungsaufgaben anlegen*.
3. Ordnen Sie den Geschäftsexperten wie unten beschrieben zu.

### Zuordnen des Geschäftsexperten

1. Ordne Sie in der Spalte *GP-Experte* für alle ausgewählten Zeilen den Geschäftsprozess-Experten zu, der die TBOMs aufzeichnen soll. Sie haben folgende Möglichkeiten:
  - Geben Sie die Daten direkt in die entsprechenden Zeile in der Tabelle ein.
  - Wählen Sie *Massenänderung*, um alle markierten Zeilen auf einmal zu verarbeiten. Geben Sie als nächstes die Daten im Dialogfenster *Benutzer & Systemrolle zuweisen - Massenänderung* ein.
2. Wenn das Change-Request-Management für einen Branch der Lösung aktiviert ist, ordnen Sie der Aufgabe einen Änderungsbeleg zu. Hierzu verwenden Sie die Spalte *Änderungsbeleg*. Zur Aktivierung öffnen Sie eine Lösung und einen Branch in der Anwendung *Lösungsverwaltung*. Bearbeiten Sie auf der Registerkarte *Eigenschaften* die Branch-Einstellungen.

Wenn es kein Änderungsdocument für den Geschäftsprozess-Experten gibt, vergewissern Sie sich, dass der Change-Manager dem Geschäftspartner des Geschäftsprozess-Experten ein Änderungsdocument zuordnet.

3. Wenn für den Branch und die logische Komponentengruppe der ausführbaren Einheit Sites aktiviert sind, ordnen Sie in der Spalte *Site* die Site zu, deren Systeme für die TBOM-Aufzeichnung verwendet werden sollen. Das logische System, in dem die TBOM aufgezeichnet wird, ändert sich je nach ausgewählter Site und Systemrolle.  
Wählen Sie die Site-Gültigkeit in der Spalte *Gültig für Sites*. Weitere Informationen finden Sie unter [Site-abhängige TBOMs \[Seite 139\]](#).
4. Wenn bereits eine TBOM existiert, wählen Sie in der Spalte *Aktion* abhängig vom Anwendungsfall *Wieder aufzeichnen* oder *Erweitern*.
5. Optional: Fügen Sie einen Kommentar hinzu (z.B. Ausführungsanweisungen).
6. Wählen Sie *Prüfen*.  
Das System prüft u.a., ob Sie für jeden ausgewählten Eintrag einen Geschäftsprozess-Experten angegeben haben, ob der Geschäftsprozess-Experte eine E-Mail-Adresse hat, und ob für den Geschäftsprozess-Experten sowohl ein Systembenutzer als auch ein Geschäftspartner vorliegen. Das System überprüft außerdem die E-Mail-Adresse, den Systembenutzer und den Geschäftspartner zum Qualitätsexperten. Um weitere Benutzer für Geschäftsprozess-Experten anzulegen, gehen Sie wie folgt vor:
  1. Wählen Sie in der SAP-Solution-Manager-Konfiguration (Transaktion **SOLMAN\_SETUP**) den Pfad **► Test-Suite ► Business Process Change Analyzer ►**.
  2. Führen Sie im Schritt **► BPCA-Vorbereitung ► TBOM-Aufgaben ►** die Aktivität *Benutzer Geschäftsprozessexperte anlegen* aus.
7. Wählen Sie *Workitems anlegen*.  
Das System erstellt für jeden ausgewählten Eintrag eine TBOM-Aufzeichnungsaufgabe.

## Ergebnis

Sowohl der Geschäftsprozess-Experte als auch der Qualitätsexperte werden per E-Mail benachrichtigt und können in der Sicht *Meine Aufgaben - TBOM-Arbeitsvorrat* auf die Aufzeichnungsaufgabe zugreifen. Sie können ihren TBOM-Arbeitsvorrat über die Links in den E-Mails aufrufen.

## Weitere Informationen

[TBOM-Arbeitsvorrat \[Seite 126\]](#)

## 7.2.7 Anlegen dynamischer TBOMs

## Voraussetzungen

Sie haben sichergestellt, dass die Voraussetzungen zum Aufzeichnen von Sonderfällen von TBOMs erfüllt sind. Weitere Informationen finden Sie unter [Erfüllen der Voraussetzungen für Sonderfälle der TBOM-Aufzeichnung \[Seite 143\]](#).

## Kontext

Zeichnen Sie eine Transaktion auf, während diese ausgeführt wird.

## Vorgehensweise

1. Starten Sie die Sicht [Test-Suite - Testvorbereitung](#).
2. Wählen Sie eine Lösung und einen Branch aus, und öffnen Sie diese.
3. Führen Sie einen Drilldown auf die ausführbare Einheit aus, die Sie bearbeiten möchten.  
In der Sicht [Browser](#) wird die ausführbare Einheit auf der Registerkarte [Elemente von <Knoten>](#) angezeigt.
4. Führen Sie im Kontextmenü eine der folgenden Aktionen aus:
  - Wenn zur Ausführung des Testfalls geschäftliche Kompetenz erforderlich ist, delegieren Sie die TBOM-Aufzeichnung an einen Geschäftsexperten oder Tester. Wählen Sie [TBOM-Aufgaben anlegen](#). Weitere Informationen finden Sie unter [Delegieren von TBOM-Aufzeichnungen an Geschäftsexperten \[Seite 148\]](#).
  - Um die TBOM direkt anzulegen, wählen Sie auf der Registerkarte [Elemente von <Name der ausführbaren Einheit>](#) im Kontextmenü einer leeren Zeile **Neu** **TBOMs** **TBOM (Anlegen)**.
5. Wenn die ausführbare Einheit einem Site-abhängigen Branch und einer Site-abhängigen logischen Komponentengruppe zugeordnet ist, wählen Sie die Aufzeichnungs-Site und Site-Gültigkeit der TBOM. Weitere Informationen finden Sie unter [Site-abhängige TBOMs \[Seite 139\]](#).
6. Geben Sie eine TBOM-Beschreibung ein.  
Standardmäßig umfasst die Beschreibung folgende Informationen:  
`[Branch-ID]-[Knotentext]-[Komponente]-[Typ]-[Name der ausführbaren Einheit]`  
Falls erforderlich, können Sie den Namen ändern.
7. Um Ihre Eingaben zu bestätigen, wählen Sie [OK](#).
8. Wählen Sie [Aufzeichnung starten](#).  
Die ausführbare Einheit wird in einem neuen Modus bzw. Browserfenster geöffnet.  
Wenn Sie eine Webanwendung aufzeichnen, verwenden Sie die richtigen Benutzer. Weitere Informationen finden Sie unter [Aufzeichnen von TBOMs für Webanwendungen \[Seite 146\]](#).
9. Führen Sie die erforderlichen Aktivitäten im verwalteten System aus.
10. Nachdem Sie die Aktivitäten beendet haben, verlassen Sie die Funktion bzw. schließen Sie das Browserfenster, um sicherzustellen, dass alle Objekte erfasst wurden.

11. Wählen Sie *Aufzeichnung beenden*.

SAP Solution Manager sammelt nun alle verwendeten Objekte aus dem System/den Systemen und legt daraus eine TBOM an. Nach erfolgreicher Erstellung der TBOM wird eine Bestätigung angezeigt.

12. Wenn die TBOM-Aufzeichnung fehlschlägt, prüfen Sie das verwaltete System. Weitere Informationen hierzu finden Sie unter [Fehlerbehebung: \[Seite 162\]](#). [Prüfen des verwalteten Systems \[Seite 162\]](#).

## 7.2.7.1 Aufzeichnen einer dynamischen TBOM in mehreren Läufen

### Voraussetzungen

Sie haben eine dynamische TBOM für die ausführbare Einheit angelegt.

### Kontext

Wenn eine Transaktion über verschiedene Pfade nacheinander ausgeführt wird, erweitern Sie die TBOM entsprechend.

### Vorgehensweise

1. Wählen Sie in der Anwendung *Test-Suite - Testvorbereitung* eine Lösung und einen Branch aus.
2. Wählen Sie *TBOMs und Testfälle*.
3. Klicken Sie auf den TBOM-Namen, um den Dialog für TBOMs zu öffnen.
4. Auf der Registerkarte *Erweiterung* werden alle Erweiterungen der ausgewählten TBOM angezeigt. Die erste Erweiterung enthält die Objekte der initialen TBOM-Aufzeichnung.
5. Wählen Sie *Erweiterung anlegen*.
6. Führen Sie den nächsten Pfad in der Transaktion aus.
7. Sichern Sie Ihre Einträge mit *OK*.
8. Wiederholen Sie den Prozess bei allen Pfaden.

## 7.2.7.2 Aufzeichnen von TBOMs über automatisierte Testfälle



## Voraussetzungen

Automatisieren Sie das Anlegen und Aktualisieren von TBOMs durch Aufzeichnen von TBOMs über automatisierte Testfälle. Wenn die automatisierten Testfälle ausgeführt werden, erzeugt oder aktualisiert das System automatisch auch die betroffenen TBOMs.

- Sie haben eine Lösungsdokumentation angelegt.
- Sie haben automatisierte eCATT- und CBTA-Testfälle angelegt.
- Sie haben den relevanten Knoten ausführbare Einheiten zugeordnet.
- Sie haben diesen Knoten automatisierte Testfälle zugeordnet.
- Sie haben sichergestellt, dass die RFC-Verbindung vom SAP-Solution-Manager-System zum verwalteten System existiert. Sie haben außerdem sichergestellt, dass die RFC-Rückverbindung vom verwalteten System zum SAP-Solution-Manager-System existiert. Die RFC-Rückverbindung folgt der Namenskonvention **SM\_<SAP-Solution-Manager-System>CLNT<SAP-Solution-Manager-Mandant>\_BACK**. Sie haben die erforderlichen Anmeldedaten eingegeben. Sie bestätigen, dass das System die Rückverbindung automatisch hergestellt hat, wenn Sie eine RFC-Verbindung anlegen, in der SAP-Solution-Manager-Konfiguration (Transaktion **SOLMAN\_SETUP**) unter **► Konfiguration der verwalteten Systeme** **▾**.
- Sie haben einen Benutzer für die Aufzeichnung automatisierter Testfälle angelegt. Diese Aktivität gehört zum Schritt **BPCA-Vorlagenbenutzer anlegen** in der Konfiguration des Business Process Change Analyzers.

## Vorgehensweise

1. In **Test-Suite - Testvorbereitung**:
  1. Führen Sie einen Drilldown zu einer ausführbaren Einheit aus, und vergewissern Sie sich, dass ihr mindestens ein automatisierter Testfall zugeordnet wurde.
  2. Wählen Sie im Kontextmenü der ausführbaren Einheit **Für TBOM-Generierung zuordnen**.
  3. Wählen und bestätigen Sie die Testfälle, die für die automatische TBOM-Generierung verwendet werden sollen.

Weitere Informationen zum Identifizieren von ausführbaren Einheiten zum Anlegen von TBOMs finden Sie unter [Prüfen von TBOM-Abdeckung und -Status \[Seite 158\]](#).

2. In der Anwendung **Testplanverwaltung**:
  1. Legen Sie einen dedizierten Testplan für die TBOM-Generierung an, der die zuvor zugeordneten automatisierten Testfälle enthält.

### i Hinweis

Der Testausführungsstatus wird rot angezeigt, wenn eine automatische TBOM-Generierung fehlschlägt, selbst wenn der Testfall erfolgreich ausgeführt wird. Bitte beachten Sie, dass sich hierdurch die Testplanergebnisse verfälschen können.

Hintergrund: Wenn die automatische Generierung einer TBOM fehlschlägt, ist der Testausführungsstatus rot, auch wenn der Testfall erfolgreich ausgeführt wurde. Die Ergebnisse eines Testplans können daher verfälscht sein.

Weitere Informationen finden Sie unter [Testpläne anlegen und bearbeiten \[Seite 239\]](#).

2. Legen Sie im Testplan mindestens ein Testpaket an, das die zuvor zugeordneten Testfälle enthält, und ordnen Sie es dem Tester zu, der die TBOMs generiert.  
Wenn der ausgewählte Branch für Sites aktiviert wurde, wählen Sie die Site-Gültigkeit der TBOMs, die aus dem Testpaket erstellt wurden. Weitere Informationen finden Sie unter [Site-abhängige TBOMs \[Seite 139\]](#).
3. Planen Sie die Ausführung der Testpakete ein, welche die TBOM-relevanten Testfälle enthalten:
  1. Wählen Sie in der Sicht *Tester-Arbeitsvorrat* das Testpaket aus.
  2. Wählen Sie ► *Automatischer Test* ► *Ausführung einplanen* ►.
  3. Markieren Sie im Dialog *Ausführung einplanen* die Ankreuzfelder *Status nach TWB kopieren* und *TBOM-Aufzeichnung aktivieren*. Nur wenn beide Ankreuzfelder markiert sind, kann das System bei der Ausführung des Testfalls eine TBOM aufzeichnen.
  4. Prüfen Sie im Dialog *Laufzeit* die Angaben zur Laufzeit. Planen Sie beispielsweise Hintergrundjobs für Testfälle ohne SAP GUI ein. Planen Sie Dialogjobs (*Vordergrund*) für Testfälle ein, die die SAP-GUI-Interaktion simulieren. Da diese Option standardmäßig aktiviert ist, stellen Sie sicher, dass die Transaktion *STPFE* aktiv ist.  
Suchen Sie nach *Tests für die automatische Ausführung im Vordergrund einplanen* im SAP Help Portal unter <http://help.sap.com>, um weitere Informationen zu erhalten.

#### **i** Hinweis

Sie können ein Testpaket auch sofort ausführen, anstatt es einzuplanen. In diesem Fall müssen Sie im Dialog *Startoptionen* u.a. Systemdaten und Zielsystem für die Ausführung angeben.

Markieren Sie im Dialog *Startoptionen* folgende Ankreuzfelder:

- *RFC-Verbindung schließen*
- *Status in Test Workbench kopieren*
- *TBOM-Aufzeichnung aktivieren*

5. Um die Einplanung zu sichern, wählen Sie *OK*.

## Ergebnisse

Das System generiert für alle betroffenen ausführbaren Einheiten TBOM-Erweiterungen. Die Übersicht über die TBOM-Erweiterungen zeigt die generierten Erweiterungen.

Wenn eine TBOM-Erweiterung bereits zuvor über einen automatisierten Testfall generiert wurde, aktualisiert das System diese TBOM-Erweiterung.

Das eCATT-Protokoll zeigt die automatisierten Testfälle, die die TBOMs generiert haben.

### 7.2.7.3 Aufzeichnen von TBOMs während manueller Tests

## Voraussetzungen

- Sie haben eine Lösungsdokumentation angelegt.
- Sie haben dem relevanten Knoten eine ausführbare Einheit zugeordnet.
- Sie haben einen manuellen Testfall angelegt.
- Sie haben in der Lösungsdokumentation folgende Aktionen durchgeführt:
  - den manuellen Testfall demselben Lösungsdokumentations-Branch auf der Registerkarte *Elemente von <Name der ausführbaren Einheit>* zugeordnet
  - die ausführbare Einheit als Testobjekt angegeben:
    1. Wählen Sie im Kontextmenü des Testfalls *Ausführbare Elemente zuordnen*.  
Es werden alle ausführbaren Einheiten angezeigt, auf die der Testfall angewendet werden kann. Für einen Testfall auf der Ebene des Geschäftsprozesses können Sie alle ausführbaren Einheiten in Bezug auf den Prozess und seine untergeordneten Knoten, wie z.B. die Geschäftsprozessschritte, zuordnen.
    2. Geben Sie in der Spalte *Getestet* die zu testenden ausführbaren Einheiten an. Als Standardauswahl wählen Sie alle ausführbaren Einheiten.
- Sie haben einen Testplan oder ein Testpaket mit manuellen Testfällen für die Lösung angelegt. Der manuelle Testfall ist in diesem Testplan bzw. Testpaket enthalten.
- Sie sind dem Testpaket als Tester zugeordnet.

## Kontext

Zeichnen Sie dynamische TBOMs durch Ausführung manueller Testfälle auf.

### → Empfehlung

Testfälle werden oft nach dem Einspielen von Änderungen ausgeführt. Da die vorhandenen TBOMs nach Änderungen möglicherweise nicht mehr aktuell sind, ist es sinnvoll, TBOMs bei Ausführung der Testfälle anzulegen.

Geben Sie den Testern in diesem Fall die nachfolgenden Anweisungen.

## Vorgehensweise

1. Starten Sie die Sicht *Meine Aufgaben - Tester-Arbeitsvorrat*.
2. Wählen Sie das Testpaket, in dem der manuelle Testfall enthalten ist.
3. Markieren Sie den Testfall, und wählen Sie *Ausführen*.

Navigieren Sie zum Bild *Manuelle Testfallausführung*.

4. Wählen Sie im Feld *TBOM-Trace* die Option *Anlegen*.

Wenn das Feld *TBOM-Trace* nicht sichtbar ist, legen Sie ein Standardpersonalisierungsprofil an. Legen Sie im Abschnitt *Testausführung* das Feld *TBOM-Optionen* auf *TBOM-Trace - anlegen* oder *TBOM-Trace - nicht anlegen* fest.

5. Geben Sie in der Liste [Ausführbare Einheiten](#) die ausführbaren Einheiten an, für die TBOMs aufgezeichnet werden sollen.
6. Zeigen Sie den Testfall an.
7. Wählen Sie [Testausführung starten](#).  
Das Dialogfenster [Test ausführen](#) wird angezeigt.
8. Prüfen Sie den Testfall,
  - um die Pfade anzugeben, für die die TBOMs aufgezeichnet werden sollen, wenn mehrere ausführbare Referenzpfade angezeigt werden
  - um anzugeben, ob die TBOMs erneut aufgezeichnet werden sollen (Standard) oder ob eine Transaktion über verschiedene Pfade ausgeführt wird (TBOM-Ausführungsoption [Erweitern](#))  
Weitere Informationen erhalten Sie unter [Aufzeichnen einer dynamischen TBOM in mehreren Läufen \[Seite 152\]](#).
9. Wählen Sie zur Bestätigung [OK](#).  
Das Dialogfenster [Test ausführen](#) wird angezeigt.
10. Wählen Sie zur Bestätigung [OK](#).  
Das System startet nun die Anwendung und zeichnet gleichzeitig die TBOM auf.
11. Wählen Sie [Aufzeichnung starten](#).
12. Führen Sie den Testfall aus.
13. Halten Sie die TBOM-Aufzeichnung an.

## Ergebnisse

Im Bild [Manuelle Testfallausführung \[GJ1\]](#) wird die Meldung `TBOM erfolgreich angelegt` angezeigt. Sie haben den manuellen Testfall ausgeführt und gleichzeitig eine TBOM aufgezeichnet.

## 7.2.8 Bearbeiten der TBOM-Gewichtung

### Verwendung



Ordnen Sie Gewichtungswerte den Objekten in der TBOM zu. Beispielsweise können Sie Tabellen höher gewichten als Elemente der Benutzungsoberfläche. Sie können für jede Objektklassifikationsart eine andere Gewichtung festlegen.

Der Business Process Change Analyzer (BPCA) kann je nach Einstellung die Liste der von einer Änderung betroffenen Objekte auf Objekte mit einer bestimmten Gewichtung beschränken. Diese Beschränkung wirkt dann wie ein Filter auf TBOM-Objekte, die eine entsprechende Gewichtung haben.

Sie können die TBOM-Gewichtung für einen Benutzer oder für alle definieren.

Weitere Informationen finden Sie unter [Optimieren des Testumfangs mit Gewichtungsoptionen \[Seite 191\]](#).

## Vorgehensweise

1. Wählen Sie unter *Verwaltung - Änderungseinfluss-Analyse* die Option  *Tabelle der TBOM-Gewichtungen bearbeiten* .
2. Sie gelangen zum Bild *Gewichtungen pflegen*. Wechseln Sie in den Bearbeitungsmodus.
3. Legen Sie fest, auf welche TBOM-Objekte die einzelnen Gewichtungen anwendbar sind.
4. Prüfen Sie die Gewichtungsangaben.
  1. Um eine einzelne Zeile zu prüfen, doppelklicken Sie in die betreffende Zeile.  
Um alle Zeilen zu prüfen, wählen Sie *Prüfen*.  
Das Prüfergebnis zeigt alle fragwürdigen oder ungültigen Gewichtungen an. Diese entstehen z.B. durch falsche TBOM-IDs, falsche Benutzernamen oder doppelte Einträge.  
Gewichtungsangaben werden von oben nach unten verarbeitet, deshalb können auch Zeilen zu einer Warnung führen, wenn sie weiter unten in der Tabelle und allgemeiner formuliert sind als weiter oben aufgeführte Zeilen.
  2. Wählen Sie *Warnungen löschen und fortfahren*, um die ungültigen Einträge zu löschen.  
Wenn Sie die TBOM-Gewichtungen sichern, wird die Prüfung ebenfalls aufgerufen, Sie können jedoch auch ungültige Gewichtungswerte sichern. Die Änderungseinfluss-Analyse ignoriert ungültige Gewichtungen.
5. Sichern Sie Ihre Eingaben.

## Automatisches Anlegen von Gewichtungen

Sie können auch eine Objektliste aus der TBOM verwenden, um TBOM-Gewichtungen zu erzeugen. In diesem Fall sind viele Feldwerte vordefiniert. Gehen Sie dazu folgendermaßen vor:

1. Öffnen Sie eine der folgenden Anzeigen:
  - TBOM-Content-Anzeige
  - Schnittmengenanzeige einer Änderungseinfluss-Analyse
  - Anzeige der ungetesteten Objekte bei der Optimierung des Testumfangs
  - Ergebnisanzeige der Aktualitätsprüfung
2. Markieren Sie in der Tabellenübersicht eine oder mehrere Zeilen auf der Ebene der logischen Komponenten oder Pakete. Alternativ können Sie auch im Detailreport eine oder mehrere Zeilen markieren.
3. Wählen Sie *Der Gewichtung hinzufügen*.  
Das System legt aus den ausgewählten Zeilen automatisch eine Gewichtung an. Ein Dialogfenster zeigt standardmäßig die neuen Gewichtungsangaben an.  
Wenn sie logische Komponenten oder Pakete ausgewählt haben, sind nur diese Felder vorbelegt. Wenn Sie einzelne Objekte ausgewählt haben, sind alle Felder bis auf Benutzername und TBOM-Gültigkeit vorbelegt.
4. Um Felder für die Gewichtung anzupassen, bearbeiten Sie die Werte direkt im aktuellen Dialogfenster. Sie können z.B. einen Festwert durch ein Maskierungszeichen ersetzen, um eine Gewichtungsangabe zu verallgemeinern. Wir empfehlen, die Gewichtungsangabe so allgemein wie möglich zu definieren.
5. Um mehrere Gewichtungsangaben gleichzeitig zu ändern, gehen Sie folgendermaßen vor:
  1. Wählen Sie *Massenänderung*.  
Wenn Sie eine einzelne Zeile markiert haben, sind die Felder mit den Werten dieser Gewichtungsangabe vorbelegt. Wenn Sie mehrere Zeilen markiert haben, sind die Felder mit Maskierungszeichen vorbelegt.
  2. Ändern Sie die Gewichtung.
  3. Wählen Sie die Markierungen rechts vom Dialogfenster für die Felder, deren Werte Sie übernehmen möchten.

4. Um die Werte in die zuvor markierten Zeilen zu übernehmen, wählen Sie [Übernehmen](#).  
Um die Werte in alle Zeilen zu übernehmen, wählen Sie [Für alle übernehmen](#).
6. Markieren Sie alle Gewichtungsangaben, die Sie sichern wollen.
7. Wählen Sie [Prüfen](#).  
Das System prüft alle markierten Gewichtungsangaben. Die Spalte [Status](#) zeigt an, ob eine Gewichtungsangabe gültig ist.  
Das System weist z.B. darauf hin, wenn im System bereits eine identische TBOM-Gewichtung vorliegt. Wenn es eine identische TBOM-Gewichtung gibt, können Sie die redundante Zeile expandieren, um die bestehende Gewichtung unterhalb der redundanten Zeile schreibgeschützt einzublenden. Wenn die Gewichtungsangabe fehlerhaft ist, erzeugt das System einen Warnhinweis.
8. Sichern Sie Ihre Eingaben.  
Das System sichert alle markierten Gewichtungsangaben. Gewichtungsangaben, die mit einer Warnung versehen sind, werden ebenfalls gesichert, allerdings ignoriert das System diese Gewichtungsangaben bei der Änderungseinfluss-Analyse.

## 7.2.9 Prüfen von TBOM-Abdeckung und -Status

### Verwendung

Prüfen Sie regelmäßig, ob die Voraussetzungen zum Einsatz des Business Process Change Analyzer (BPCA) erfüllt sind. Wählen Sie Teile der Lösung aus, und stellen Sie Folgendes sicher:

- dass TBOMs und Testfälle allen Geschäftsprozessen zugeordnet sind
- dass existierende TBOMs aktuell sind

### Voraussetzungen

- Sie haben eine Lösungsdokumentation angelegt. Sie haben den relevanten Knoten ausführbare Einheiten und Testfälle zugeordnet und TBOMs angelegt.
- Um veraltete TBOMs zu identifizieren, haben Sie eine Aktualitätsprüfung eingeplant.  
Weitere Informationen finden Sie unter [Einplanen von Aktualitätsprüfungen \[Seite 164\]](#).

### Vorgehensweise

1. Starten Sie die Anwendung [Test-Suite - Testvorbereitung](#).
2. Wählen Sie die Lösung und den Branch aus.
3. Wählen Sie [TBOMs und Testfälle](#).
4. Ändern Sie ggf. die Reportdefinition.
5. Wählen Sie [Ausführen](#), um zur Report-Ergebnisliste zu gelangen.
6. Wählen Sie ein von SAP ausgeliefertes oder benutzerdefiniertes Aspektfilterlayout, um Ihre Lösung zu analysieren.

Verwenden Sie Aspektfilter, um

- Objekte, beispielsweise geschäftskritische Vorgänge, anzuzeigen und deren Status zu prüfen
  - Geschäftsprozesse anzuzeigen, denen noch keine Testfälle oder TBOMs zugeordnet sind (*Testfalltyp*-Attributwert *Nicht existent*, *Ausführbarer TBOM-Typ = fehlt*).
  - zu prüfen, ob automatisierte Testfälle zum Erstellen von TBOMs zugeordnet sind (Attribut *TBOM erzeugt von Auto-Test*).
  - Typen von ausführbaren Einheiten zu identifizieren (z.B. ausführbare Einheiten, die Webanwendungen verwenden), um die Voraussetzungen zum Anlegen von TBOMs zu erfüllen
  - Knoten zu identifizieren, denen gegenwärtig nur statische oder semidynamische TBOMs zugeordnet sind. Sie können dann die Erstellung von präziseren dynamischen TBOMs für geschäftskritische Knoten in Betracht ziehen.
  - veraltete TBOMs zu identifizieren (*TBOM-Status = Veraltet*).
7. Fügen Sie entsprechend dem Report den Objekttyp TBOM für folgende Auswertungen hinzu:
- Verwenden Sie das Attribut *Batch-Job-Status* für Transaktionen, die Hintergrundjobs starten, um zu kennzeichnen, ob Hintergrundjobs während der TBOM-Aufzeichnung eingeplant wurden.
  - Identifizieren Sie TBOMs, die von bestimmten Personen oder zu bestimmten Zeiten (*Angelegt um/von*, *Aktualisiert um/von*) angelegt wurden
  - Stellen Sie sicher, dass TBOMs in bestimmten Systemen (*System/Mandant*) angelegt werden
  - Identifizieren Sie TBOMs, deren Aufzeichnung an Geschäftsprozess-Experten delegiert wurde (*Aufgabenexistenz*)
8. Wählen Sie in der Spalte *Pfad* die Verknüpfung, um das Strukturelement zu öffnen.
9. Auf der Registerkarte *Elemente von <Knoten>* im Kontextmenü stehen Ihnen folgende Optionen zur Verfügung:
- Wenn einem Strukturelement, das getestet werden soll, noch kein Testfall zugeordnet ist. Ordnen Sie fehlende Testfälle zu, oder legen Sie sie an. Wählen Sie beispielsweise ► *Neu* ► *Testfälle* ► *Testkonfiguration (Anlegen)* ► abhängig vom Testfalltyp oder davon, ob Sie einen Testfall zuordnen oder anlegen möchten.
  - Wenn für eine ausführbare Einheit noch keine TBOM angelegt wurde, haben Sie folgende Möglichkeiten:
    - Legen Sie per Massengenerierung automatisch semidynamische oder statische TBOMs an.
    - Legen Sie manuell eine dynamische TBOM an.
    - Ordnen Sie die entsprechende ausführbare Einheit für die automatische TBOM-Generierung mit automatisierten Testfällen zu.
- Weitere Informationen finden Sie unter [Auswählen der TBOM-Arten \[Seite 136\]](#).
- Wenn eine veraltete TBOM einem Strukturelement zugeordnet ist, zeichnen Sie die veralteten TBOMs erneut auf. Wenn Sie veraltete TBOMs verwenden, ist die Änderungsanalyse möglicherweise ungenau.
  - Wenn Sie prüfen möchten, ob eine TBOM-Erweiterung bereits für Objekte angelegt wurde, die in Transaktionen verwendet werden, mit denen Batch-Jobs gestartet werden, prüfen Sie das Attribut *Batch-Job-Status*. Weitere Informationen finden Sie unter [Status einer Batch-TBOM prüfen \[Seite 160\]](#).

## i Hinweis

Möglicherweise möchten Sie das Anlegen oder Aktualisieren einer TBOM einem Geschäftsexperten überlassen. Legen Sie TBOM-Aufzeichnungsaufgaben für diese Experten an, um die TBOM-Aufzeichnung zu delegieren. Weitere Informationen finden Sie unter [Delegieren von TBOM-Aufzeichnungen an Geschäftsexperten \[Seite 148\]](#).

## Ergebnis

Sie können Ihre Geschäftsprozesse analysieren und aus den Analyseergebnissen Testpläne erstellen. Weitere Informationen finden Sie unter [Analysieren der von Änderungen betroffenen Geschäftsprozesse \[Seite 170\]](#).

### 7.2.9.1 Exportieren und Importieren von TBOMs

Sie können vollständige Branches in die Anwendung *Lösungsverwaltung* von SAP Solution Manager importieren.

#### Kontext

In der Anwendung *Lösungsverwaltung* können Sie entscheiden, ob Sie TBOMs in den Export einschließen möchten. Anschließend können exportierte TBOMs in andere Lösungen und Branches importiert werden.

### 7.2.9.2 Prüfen des Status von Batch-TBOMs

#### Voraussetzungen

Die Voraussetzungen zum Aufzeichnen von Batch-TBOMs sind erfüllt, und Sie haben eine Batch-TBOM aufgezeichnet. Weitere Informationen finden Sie unter [Aufzeichnen von Batch-TBOMs \[Seite 143\]](#).

#### Kontext

Wenn Sie eine TBOM für eine ausführbare Einheit aufzeichnen, mit der Hintergrundjobs eingeplant werden, prüfen Sie, ob bei der Ausführung des TBOM-Sammlers Fehler aufgetreten sind. Starten Sie ihn ggf. manuell.

*Vorgehensweise*

#### Vorgehensweise

1. Wählen Sie in der Anwendung *Test-Suite - Testvorbereitung* eine Lösung und einen Branch aus.
2. Wählen Sie *TBOMs und Testfälle*.



3. Ändern Sie ggf. die Reportdefinition.
4. Wählen Sie *Ausführen*, um zur Report-Ergebnisliste zu gelangen.
5. Um die Status von Batch-TBOMs anzuzeigen, legen Sie in der Sicht *Auswertungen* auf Basis des Attributs `Batch Job Status` einen Aspektfilter an.

Im Statusfeld für die Batch-TBOM sehen Sie, ob das System bereits alle TBOM-Erweiterungen für alle Hintergrundjobs erzeugt hat. Folgende Status werden unterstützt:

- *Ausstehend*: Die Hintergrundjobs sind eingeplant und wurden möglicherweise auch bereits ausgeführt, es wurden aber noch keine TBOMs erzeugt. Dies geschieht beim nächsten Lauf des TBOM-Sammlers. Bitte warten Sie, bis der Status auf *Abgeschlossen* wechselt.
- *Abgeschlossen*: Die Hintergrundjobs wurden ausgeführt, und die TBOM-Erweiterung wurde angelegt.
- *Fehlerhaft*: Während der Erzeugung der TBOM-Erweiterung für Hintergrundjobs sind Fehler aufgetreten (z.B. konnte sich das System nicht automatisch mit dem verwalteten System verbinden). Wenn die RFC-Verbindung nicht funktionsfähig war, lösen Sie die Batch-TBOM-Sammlung manuell aus:
  1. Führen Sie in der Sicht *Lösungsdokumentation* einen Drilldown in die TBOM aus.
  2. Wählen Sie auf der Registerkarte *Elemente von <Name der ausführbaren Einheit>* den TBOM-Namen aus, um die TBOM zu öffnen.  
Auf der Registerkarte *Kopfinformationen* wird der Batch-TBOM-Status *Fehlerhaft* angezeigt.
  3. Identifizieren Sie unter *Umfeld* das verwaltete System, und melden Sie sich an, um eine der folgenden Aktionen auszuführen:
    - Wenn keine RFC-Verbindung besteht, beheben Sie das Problem.
    - Wenn die RFC-Verbindung funktionsfähig ist, fahren Sie mit dem nächsten Schritt fort.
  4. Klicken Sie auf der Registerkarte *Kopfinformationen* auf den Statuswert *Fehlerhaft*. Gehen Sie zum Bild *Hintergrund-TBOM: Statusdetails*. In der Spalte *TBOM-Status* haben TBOM-Sammler, die nicht ausgeführt werden konnten, den Status *Fehler*.
  5. Stellen Sie sicher, dass die Hintergrundjobs im verwalteten System abgeschlossen sind. Nur wenn die Hintergrundjobs abgeschlossen sind, kann das System eine vollständige TBOM-Erweiterung erzeugen.
    1. Wählen Sie *Aktualisieren*.
    2. Stellen Sie sicher, dass der Status im Feld *Jobstatus* dem Wert *Abgeschlossen* entspricht.
  6. Um die TBOM-Sammler für alle Batch-Jobs zu starten, wählen Sie in der Spalte *Start* die Option *Starten*.  
Um die TBOM-Sammlung nur für fehlgeschlagene Batch-Jobs zu starten, markieren Sie die fehlgeschlagenen Jobs, und wählen Sie das Symbol *TBOM sammeln*.  
Der TBOM-Sammler sammelt alle verwendeten Objekte aus dem verwalteten System und erstellt eine TBOM-Erweiterung. Nach der Erstellung wird eine Bestätigungsmeldung angezeigt.  
Legen Sie in anderen Fehlerfällen die Batch-TBOM erneut an.
- *Keine Hintergrundjobs*: Die ausführbare Einheit löst keinen Batch-Job aus.

Wenn während der Aufzeichnung einer TBOM-Erweiterung die ausführbare Einheit mehrere Hintergrundjobs eingeplant hat, legt das System eine eigene TBOM-Erweiterung für jeden Hintergrundjob an.

Unter *TBOM-Erweiterungen* können Sie den Status für jede einzelne Erweiterung prüfen. Wenn Sie im Bild *TBOM-Erweiterungen* das Statussymbol wählen, sehen Sie nur Details zu den Hintergrundjobs, die während der Aufzeichnung der Erweiterung eingeplant wurden.

6. Um die Batch-TBOM-Statusdetails zu prüfen, klicken Sie auf Statuswert oder -symbol.

Folgende Informationen werden angezeigt:

- **TBOM-Sammler-Status, Sammler:** Die Ampel und der Status im Feld *Sammler* zeigen den Status des TBOM-Sammlers an. Wählen Sie das Feld *Sammler*, um weitere Informationen zum TBOM-Sammler anzuzeigen.
- **Starten:** Die Option *Starten* wird nur angezeigt, wenn der TBOM-Sammler aktuell nicht eingeplant ist und die TBOM-Erweiterungen noch nicht vollständig angelegt wurden. Sie ermöglicht die Einplanung des TBOM-Sammlers mit dem Startzeitpunkt *Sofort*. Im weiteren Verlauf wird der TBOM-Sammler alle fünf Minuten ausgeführt.
- **Anhalten:** Die Drucktaste *Anhalten* wird nur angezeigt, solange der TBOM-Sammler eingeplant ist und noch nicht alle TBOM-Erweiterungen angelegt hat. Mit dieser Drucktaste kann der TBOM-Sammler ausgeplant werden, sodass er nicht mehr alle fünf Minuten ausgeführt wird.
- **TBOM-Status:** Zeigt den Status der TBOM-Erstellung an.
- **Job:** Name des Hintergrundjobs im verwalteten System. Klicken Sie in das Feld *Job*, um Informationen zu dem Hintergrundjob anzuzeigen, der auf dem verwalteten System eingeplant wurde.
- **Periodischer Job:** Gibt an, ob ein Hintergrundjob periodisch ausgeführt wird.
- **Job angelegt von:** Name des Benutzers, der den Hintergrundjob auf dem verwalteten System ausgeführt hat.
- **Jobstatus:** Zeigt an, ob der Hintergrundjob bereits im verwalteten System ausgeführt wurde.
- **Start um, Ende um:** Uhrzeit, zu der der Hintergrundjob im verwalteten System gestartet und beendet wurde.

## 7.2.9.3 Fehlersuche: Prüfen des verwalteten Systems

### Kontext

Wenn die manuelle Aufzeichnung von TBOMs fehlschlägt, prüfen Sie die Systemparameter, oder aktualisieren Sie die TBOM-Attribute.

### Vorgehensweise

1. Führen Sie in der Lösungsdokumentationssicht einen Drilldown in die TBOM aus, die nicht aufgezeichnet werden konnte.
2. Wählen Sie die Registerkarte *Elemente von <ausführbare Einheit>*.
3. Öffnen Sie die TBOM. Wenn dies nicht möglich ist, erstellen Sie eine neue Aufzeichnung, und prüfen Sie etwaige Fehlermeldungen.
4. Prüfen Sie die Verbindung zum verwalteten System.
  1. Wählen Sie unter *Umgebung* den Eintrag *System prüfen*.  
Das System identifiziert das verwaltete System anhand der logischen Komponente und der Systemrolle aus der Lösung und zeigt es an.

2. Wenn die Verbindung zum verwalteten System nicht funktioniert, wenden Sie sich an die Systemverwaltung.

## Ergebnisse

Sie können TBOMs manuell aufzeichnen.

## 7.2.10 Identifizieren und Handhaben veralteter TBOMs

### Kontext

Wenn viele TBOMs seit längerer Zeit veraltet sind, werden Analysen ungenauer. Identifizieren Sie veraltete TBOMs, und legen Sie, abhängig vom Aufwand, den Sie erbringen möchten, und von der benötigten TBOM-Genauigkeit, neue an.

#### → Empfehlung

Wenn Sie wichtige Tests planen, bitten Sie die Tester, veraltete dynamische TBOMs erneut aufzuzeichnen, wenn sie manuelle oder automatisierte Testfälle ausführen. Weitere Informationen finden Sie unter

- [Aufzeichnen von TBOMs über automatisierte Testfälle \[Seite 152\]](#)
- [Aufzeichnen von TBOMs während manueller Tests \[Seite 154\]](#)

### Vorgehensweise

1. Planen Sie einen Hintergrundjob ein, um veraltete TBOMs zu identifizieren, und kennzeichnen Sie sie in der Datenbank als veraltet.
2. Überprüfen Sie die als veraltet gekennzeichneten TBOMs regelmäßig.
3. Zeichnen Sie veraltete dynamische TBOMs erneut auf, oder generieren Sie per Massenvorgang neue statische oder dynamische TBOMs.

## 7.2.10.1 Einplanen von Aktualitätsprüfungen

### Verwendung

Planen Sie einen Hintergrundjob ein, um zu prüfen, ob TBOMs aktuell ist, und markieren Sie TBOMs als potenziell veraltet, um eine Übersicht über potenziell veraltete TBOMs zu erhalten.

Eine TBOM kann veralten, wenn sich beispielsweise durch den Import eines neuen Support Packages eine Transaktion ändert. Führen Sie diese Aktivität manuell aus, nachdem Sie Änderungen vorgenommen haben.

Die Aktualitätsprüfung beruht auf einem Vergleich von geänderten Objekten in Transportaufträgen und Versionen der Objekte in TBOMs. Das System kann nicht feststellen, ob ein geändertes Objekt zu einer Änderung im Programmfluss geführt hat und die TBOM daher veraltet ist.

Aus diesem Grund ist der Status *Veraltet* nur ein Hinweis darauf, dass die TBOM potenziell veraltet ist und Sie sie beim nächsten eingeplanten TBOM-Aktualisierungszyklus neu aufzeichnen sollten. Sie können die TBOM weiterhin für Analysen benutzen. Weitere Informationen zum Ergebnis einer Analyse mit veralteten TBOMs finden Sie im Abschnitt **Voraussetzungen** unter [Von Änderungen betroffene Geschäftsprozesse analysieren \[Seite 170\]](#).

Die Aktualitätsprüfung vergleicht die Objekte in der TBOM mit den Objekten im betroffenen System. Dies ist das System, in dem Sie die ausführbare Einheit ausgeführt haben, um die TBOM aufzuzeichnen. Es werden all jene Objekttypen berücksichtigt, die den Programmfluss einer ausführbaren Einheit ändern können:

- CINC: Klasseninclude (ABAP Objects)
- CLAS: Klasse (ABAP Objects)
- CLSD: Klassendefinition (ABAP Objects)
- CPRI: Private Header (ABAP Objects)
- CPRO: Protected Header (ABAP Objects)
- CPUB: Public Header (ABAP Objects)
- DIAL: Dialogbaustein
- ENHO: Erweiterungsimplementierung
- ENHS: Enhancement Spot
- FUGR: Funktionsgruppe
- FUNC: Funktionsbaustein
- METH: Methode (ABAP Objects)
- PROG: Programm
- REPS: Reportsource
- SHLP: Suchhilfe
- TABU: Tabelleninhalt  
Geänderte Tabelleninhalte führen nur zu einer veralteten TBOM, wenn in der TBOM ein Schlüssel für den Tabellenzugriff aufgezeichnet wurde.
- WAPA: BSP (Business Server Pages)-Applikation
- WAPP: Seite/Controller einer BSP-Applikation
- WDYC: Controller (Web Dynpro)
- WDYN: Web-Dynpro-Komponenten
- WDYV: View (Web Dynpro)
- SQLT: Pool/Cluster-Tabelle

- TABL: Tabelle
- TTYP: Tabellentyp
- VIEW: View

## Vorgehensweise

1. Starten Sie *Verwaltung - Änderungseinfluss-Analyse*, und wählen Sie ► *TBOM-Werkzeuge* ► *Aktualitätsprüfung ausführen* ►.
2. Geben Sie die Lösung, den Branch und etwaige Umfänge an, für die Sie TBOMs prüfen möchten. Wenn der ausgewählte Branch für Sites aktiviert wurde, wählen Sie die Sites für die Prüfung aus.
3. Optional: Geben Sie im Bereich *Spezielle Prüfoptionen* ein Datum an.  
Das vorgeschlagene Datum ergibt sich aus dem TBOM-Erstellungsdatum, dem letzten Änderungsdatum der TBOM und der letzten Aktualitätsprüfung dieser TBOM. Wenn Sie nur Änderungen des betroffenen Systems nach einem bestimmten Datum berücksichtigen möchten, können Sie ein späteres Datum angeben.  
Hinweis: Wenn Sie ein späteres Datum angeben, wird die TBOM möglicherweise als *Aktualisiert* gekennzeichnet, obwohl zu einem früheren Zeitpunkt entsprechende Änderungen im betroffenen System erfolgten.
4. Wählen Sie die Systemrolle aus.
5. Um zu prüfen, ob die Systeme abgedeckt sind, in denen die Änderungen implementiert wurden, wählen Sie *Systeme anzeigen, in denen TBOMs geprüft werden*.
6. Geben Sie die Quelle der Änderungen an:
  - Standardwert: Wählen Sie *Änderungen von Transportaufträgen verwenden*.  
Die Aktualitätsprüfung berücksichtigt alle Transportaufträge in dem betroffenen System, also sowohl Transportaufträge, die in das System importiert wurden als auch Transportaufträge, die von dort exportiert wurden. Der Zeitraum für die überprüften Transportaufträge ergibt sich standardmäßig aus dem Datum der letzten Aktualitätsprüfung. Sie können auch einen Zeitraum im Bereich *Spezielle Prüfoptionen* angeben.
  - Wählen Sie *Versionsänderungen verwenden*, damit nur neue Versionen von Entwicklungsobjekten im betroffenen System berücksichtigt werden.  
Neue Versionen von Entwicklungsobjekten werden durch den Import oder Export von Transportaufträgen erzeugt. Die auf diese Weise ermittelten Objekte sind also in jedem Fall eine Teilmenge der anhand der Transportaufträge ermittelten Objekte. Sie können diese Option z.B. verwenden, wenn Sie kundeneigene Objekte entwickeln und die Prüfung auf Objekte mit Versionen beschränken wollen.
7. Wenn ein Fehler auftritt, können Sie Prüfungen auswählen, beispielsweise auf Anfrage des SAP-Supports:
  - Wenn der Fehler weiterhin besteht, wählen Sie *Veraltete TBOMs erneut prüfen* und *Alle Änderungen prüfen seit letzter TBOM-Aktualisierung*.
  - Um alle TBOMs als aktuell zu markieren und nur zukünftige Änderungen zu berücksichtigen, wählen Sie *Veraltete TBOMs neu kalibrieren*.
8. Optional: Um den Prozess im Testmodus auszuführen, heben Sie die Markierung des Ankreuzfelds *In Datenbank schreiben* auf.
9. Wählen Sie *Ausführen*.
10. Empfohlen: Planen Sie einen täglichen Hintergrundjob ein.  
Um nach abgeschlossenen Jobs zu suchen, erweitern Sie den Bereich *Jobauswahl*.

## Ergebnis

Das System hat die in der TBOM enthaltenen Objekte überprüft. Das System hat der TBOM unter folgenden Bedingungen den Status *Veraltet* zugewiesen:

- Wenn Sie die Option *Transportaufträge* gewählt haben: Die Aktualitätsprüfung hat in den überprüften Transportaufträgen mindestens ein Objekt gefunden, das auch in der TBOM enthalten ist.
- Wenn Sie die Option *Versionen* gewählt haben: Die Aktualitätsprüfung hat im betroffenen System eine Objektversion gefunden, die im überprüften Zeitraum erzeugt wurde.

## 7.2.10.2 Identifizieren veralteter TBOMs in der Lösungsdokumentation

### Voraussetzungen

Sie haben die Aktualitätsprüfung durchgeführt. Weitere Informationen finden Sie unter [Einplanen von Aktualitätsprüfungen](#) [Seite 164].

### Kontext

Identifizieren Sie Objekte, denen in der letzten Aktualitätsprüfung der TBOM-Status *Veraltet* zugewiesen wurde. Prüfen Sie den TBOM-Status regelmäßig, um festzustellen, ob und wann veraltete TBOMs aktualisiert werden sollen.

### Vorgehensweise

1. Wählen Sie in der Anwendung *Test-Suite - Testvorbereitung* eine Lösung und einen Branch aus.
2. Wählen Sie *TBOMs und Testfälle*.
3. Wählen Sie in der Sicht *Liste* ein Filterlayout, das das Attribut `Executable TBOM Status` enthält.

Die Anzahl der angelegten, fehlenden oder veralteten TBOMs wird angezeigt.

4. Wählen Sie den Attributwert *Veraltet*.

Die geänderten Objekte, die zum Status *Veraltet* geführt haben, werden angezeigt.

#### **i** Hinweis

Bei der Aktualitätsprüfung setzt das System den Status *Veraltet*, wenn es in einem Transportauftrag ein Objekt findet, das auch in der TBOM enthalten ist. Das System bricht dann die Suche ab, da es die

TBOM bereits als veraltet erkannt und markiert hat. Weitere Transportaufträge werden aus Performancegründen nicht mehr überprüft. Die hier angezeigten Objekte stammen daher immer nur aus einem einzigen Transportauftrag.

5. Legen Sie fest, ob und wie veraltete TBOMs aktualisiert werden sollen. Dabei haben Sie folgende Möglichkeiten:
  - Eine dynamische TBOM erneut aufzeichnen. Bitten Sie die Tester, die TBOMs während des nächsten Tests bei der Ausführung der entsprechenden Testfälle erneut aufzuzeichnen. Weitere Informationen finden Sie unter [Neuaufzeichnen veralteter dynamischer TBOMs \[Seite 167\]](#).
  - Neue statische und semidynamische TBOMs per Massenvorgang anlegen. Weitere Informationen finden Sie unter [Anlegen von statischen oder semidynamischen TBOMs \[Seite 140\]](#).

Filtern Sie die Objekte entsprechend.

Kennzeichnen Sie eine veraltete TBOM als aktualisiert. Weitere Informationen finden Sie unter [Kennzeichnen einer veralteten TBOM als aktualisiert \[Seite 168\]](#).

## 7.2.10.3 Erneutes Aufzeichnen von obsoleten dynamischen TBOMs

### Voraussetzungen

Sie haben eine veraltete dynamische TBOM identifiziert.

Weitere Informationen finden Sie unter [Identifizieren veralteter TBOMs in der Lösungsdokumentation \[Seite 166\]](#).

### Kontext

Obsoletere dynamische TBOMs erneut aufzeichnen.

### Vorgehensweise

1. Wählen Sie in der Anwendung *Test-Suite - Testvorbereitung* eine Lösung und einen Branch aus.
2. Wählen Sie *TBOMs und Testfälle*.
3. Wählen Sie *Ausführen*, um zur Report-Ergebnisliste zu gelangen.
4. Klicken Sie auf die Spalte *Pfad*, um zu dem ausführbaren Objekt zu gelangen.
5. Gehen Sie zur Registerkarte *Elemente von ....*

6. Klicken Sie auf den TBOM-Namen, um das TBOM-Dialogfenster zu öffnen.
7. Wählen Sie *Wiederaufnahme*.

## Ergebnisse

Die TBOM hat den Status *Aktualisiert*.

## 7.2.10.4 Kennzeichnen einer veralteten TBOM als aktualisiert

### Voraussetzungen

Sie haben eine veraltete dynamische TBOM identifiziert.

Weitere Informationen finden Sie unter [Identifizieren veralteter TBOMs in der Lösungsdokumentation \[Seite 166\]](#).

### Kontext

Kalibrieren Sie neu, wenn die vorliegende TBOM von den Änderungen am verwalteten System nicht betroffen ist und Sie den Gesamtstatus der unveränderten TBOM auf *Aktualisiert* setzen möchten.

Wenn Sie eine TBOM neu kalibrieren, wird das Datum der letzten Aktualitätsprüfung auf das aktuelle Datum gesetzt; daher kann die TBOM nur veralten, weil nach diesem Datum Änderungen vorgenommen werden.

#### ❁ Beispiel

Sie haben eine TBOM für ein kundeneigenes Programm erzeugt. Aktualisieren Sie Objekte dieser kundenspezifischen Entwicklung, ohne den Programmablauf zu ändern. Sie können dies tun, indem Sie beispielsweise einen Kommentar hinzufügen.

Während einer Aktualitätsprüfung wird der TBOM der Status *Veraltet* zugeordnet, doch Sie wissen, dass sie noch aktuell ist. Kalibrieren Sie die TBOM neu, um sie als aktuell zu kennzeichnen.

#### i Hinweis

Sie können mit dieser Funktion auch veraltete TBOMs als *Aktualisiert* kennzeichnen. Prüfen Sie daher vor einer Neukalibrierung, ob die Änderungen am verwalteten System für die TBOM irrelevant sind.



## Vorgehensweise

1. Wählen Sie in der Anwendung *Test-Suite - Testvorbereitung* eine Lösung und einen Branch aus.
2. Wählen Sie *TBOMs und Testfälle*.
3. Wählen Sie *Ausführen*, um zur Report-Ergebnisliste zu gelangen.
4. Wählen Sie die Spalte *Pfad*, um zu dem ausführbaren Objekt zu gelangen.
5. Gehen Sie zur Registerkarte *Elemente von ....*
6. Klicken Sie auf den TBOM-Namen, um das Dialogfenster für TBOMs zu öffnen.
7. Wählen Sie im Bildbereich *Kopfdaten* unter *Gesamtstatus* das Symbol *TBOM-Rekalibrierung*.

## Ergebnisse

Der TBOM-Gesamtstatus lautet nun wieder *Aktualisiert*. Die TBOM-Content bleibt unverändert.

## 7.2.10.5 Löschen von TBOMs

### Verwendung

Löschen Sie veraltete TBOMs, um Datenbankspeicherplatz zu sparen.

Wenn Sie für eine bestimmte TBOM Filter oder Gewichtungen angelegt haben, bleiben diese auch nach dem Löschen erhalten. Sobald Sie eine neue TBOM für die ausführbare Einheit erstellt haben, werden die Filter und Gewichtungen bei der Änderungseinfluss-Analyse wieder berücksichtigt.

## Vorgehensweise

### Löschen einzelner TBOMs

1. Wählen Sie in der Anwendung *Test-Suite - Testvorbereitung* eine Lösung und einen Branch aus.
2. Wählen Sie *TBOMs und Testfälle*.
3. Wählen Sie *Ausführen*, um zur Report-Ergebnisliste zu gelangen.
4. Klicken Sie auf die Spalte *Pfad*, um zu dem ausführbaren Objekt zu gelangen.
5. Gehen Sie zur Registerkarte *Elemente von ....*
6. Klicken Sie auf den TBOM-Namen, um das Dialogfenster für TBOMs zu öffnen.
7. Führen Sie einen der folgenden Schritte aus:
  - Um die komplette TBOM mit allen Erweiterungen zu löschen, wählen Sie *Löschen* im Kontextmenü.
  - Um nur eine einzelne TBOM-Erweiterung zu löschen:
    1. Wählen Sie die Registerkarte *TBOM-Erweiterungen*.
    2. Kreuzen Sie die Erweiterungen an, die Sie löschen wollen.

3. Wählen Sie [Erweiterungen löschen](#).

### Massenlöschung von TBOMs

Sie können die TBOM-Massenlöschung z.B. in folgenden Fällen durchführen:

- Sie haben TBOMs zu Testzwecken erstellt.
- Sie haben weniger präzise statische TBOMs durch dynamische TBOMs ersetzt und wollen die statischen TBOMs löschen.

Die Massenlöschung ist zum Optimieren der Datenbank-Performance nicht mehr erforderlich.

1. Wählen Sie in der Anwendung [Verwaltung - Änderungseinfluss-Analyse](#) den Pfad [BPCA-Vorbereitung](#) [TBOM-Werkzeuge](#) [TBOMs löschen](#).
2. Legen Sie die Lösung, den Branch und etwaige Umfänge fest, aus denen Sie TBOMs löschen möchten. Wenn der ausgewählte Branch für Sites aktiviert wurde, wählen Sie die Sites, aus denen Sie TBOMs löschen möchten.
3. Geben Sie die TBOM-Art an, beispielsweise [Dynamisch](#) oder [Statisch](#).

4. **⚠ Achtung**

Sie werden vom System nicht zur Bestätigung aufgefordert und die TBOMs werden unwiderruflich gelöscht.

Wählen Sie [Ausführen](#).

Wenn viele TBOMs gelöscht werden sollen, planen Sie einen Hintergrundjob ein.

## 7.3 Analysieren der von Änderungen betroffenen Geschäftsprozesse

### Voraussetzungen

- Sie haben für alle ausführbaren Entitäten in Ihrer Lösungsdokumentation TBOMs angelegt.
- Die TBOMs sind auf dem neuesten Stand. Die Analyseergebnisse stimmen nur, wenn die TBOMs aktuell sind.

Weitere Informationen zum Identifizieren und Handhaben veralteter TBOMs finden Sie unter [Identifizieren und Handhaben veralteter TBOMs \[Seite 163\]](#).

#### **i Hinweis**

##### **Analyseergebnisse mit veralteten TBOMs**

Sie können Analysen auch mit TBOMs ausführen, die den Status [Veraltet](#) aufweisen. Dieser Status identifiziert TBOMs, die potenziell veraltet sind. Eine veraltete TBOM kann in den folgenden Fällen zu ungenauen Analyseergebnissen führen:

- Ein Einflussfaktor ist unbekannt, obwohl die analysierten Änderungen Objekte eines Programmablaufs betreffen.

Dieser Fall kann eintreten, wenn eine TBOM zu wenige Objekte enthält. Seit der letzten Aktualisierung der TBOM sind neue Objekte zum Programmfluss hinzugekommen. Die geplante Änderung betrifft ausschließlich diese neuen Objekte.

- Ein Einflussfaktor ist unbekannt, obwohl die analysierten Änderungen keinen Einfluss auf den Programmfluss haben.

Dieser Fall kann eintreten, wenn eine TBOM zu viele Objekte enthält. Seit der letzten Aktualisierung der TBOM sind Objekte aus dem Programmfluss entfernt worden. Die geplante Änderung betrifft ausschließlich diese nicht mehr relevanten Objekte.

Beide Fälle treten selten auf. Eine Änderung wirkt sich allgemein auf ältere Objekte aus, die bereits in der veralteten TBOM enthalten sind, und nicht nur die neuen oder gelöschten Objekte.

- Um die Analyse auf kritische Objekte einzuschränken, haben Sie den Objekten in den TBOMs Gewichtungsangaben zugeordnet. Weitere Informationen finden Sie unter [Bearbeiten der TBOM-Gewichtung \[Seite 156\]](#).

## Kontext

Nutzen Sie diese Analyse, um zu prüfen, ob sich Änderungen auf eine Lösungsdokumentation auswirken. Sie können sowohl die relevanten Teile der Prozessstruktur als auch die relevanten Änderungen festlegen. Als Analyseergebnis erhalten Sie eine Liste der betroffenen Knoten in der Lösungsdokumentation. Anhand des Ergebnisses können Sie Testpläne anlegen, um die betroffenen Teile der Lösungsdokumentation nach der Änderung erneut zu testen.

## Vorgehensweise

1. Klappen Sie den Infobereich [Änderungseinfluss-Analyse](#) auf.
2. Wählen Sie unter Schritt **1** die Analyseart. Folgende Umfangsarten stehen Ihnen zur Verfügung:
  - **SAP-S/4HANA-Release-Wechsel**: Verwenden Sie diese Option, wenn Sie Ihr SAP S/4HANA On-Premise-System auf ein höheres Release upgraden möchten. Diese Option umfasst Analyseeinflüsse von relevanten S4/HANA-Vereinfachungselementen. Eine vollständige Liste der Vereinfachungselemente für SAP S/4HANA finden Sie im SAP-Hinweis [2313884](#). Diese Option bietet Ihnen eine vereinfachte Auswahl von Installationspaketen an. Das Upgrade muss mit dem Maintenance Planner im verwalteten System durchgeführt werden. Weitere Informationen zum Maintenance Planner finden Sie auf dem SAP Help Portal unter <http://help.sap.com/maintenanceplanner>.
  - **Support Packages/Support Package Stacks**: Diese Option bietet Ihnen eine vereinfachte Auswahl von Support Packages an. Voraussetzung ist, dass die Support Packages mit dem Maintenance Planner im verwalteten System installiert sind. Weitere Informationen finden Sie im SAP Help Portal unter <http://help.sap.com/maintenanceplanner>.
  - **Erweiterungspakete**: Diese Option bietet Ihnen eine vereinfachte Auswahl von Support Packages und Erweiterungspaketen an. Voraussetzung ist, dass das Update mit dem Maintenance Planner geplant ist. Weitere Informationen finden Sie im SAP Help Portal unter <http://help.sap.com/maintenanceplanner>.

- **Geplante Aktivierung einer Business Function:** Vor der Aktivierung können Sie Business Functions auswählen und prüfen, welche Teile von Geschäftsprozessen betroffen sein werden. Wählen Sie die Business Functions direkt aus dem verwalteten System aus. Das verwaltete System muss mindestens einen der folgenden Softwarestände haben:
  - SAP ERP 600 Enhancement Package 4: Support Package Stack 06
  - SAP ERP 600 Enhancement Package 5: Initiale Auslieferung
  - SAP CRM 7,0 Enhancement Package 1: Initiale Auslieferung

### **i** Hinweis

Sie können auch für andere Systeme den Umfang der Änderungsereignisse über Business Functions auswählen, wenn diese mindestens einen der folgenden Softwarestände haben:

- SAP NetWeaver Enhancement Package 1: Support Package Stack 06
- SAP NetWeaver Enhancement Package 2: Support Package Stack 02

Für diese Systeme ermittelt der Business Process Change Analyzer, Objekte, die in einer Erweiterungsimplementierung geändert wurden. Bei der Änderungseinfluss-Analyse werden möglicherweise nicht alle geänderten Objekte erfasst.

- **Transportaufträge:** Analysieren Sie den Einfluss bestimmter Transportaufträge. Sie sind nicht auf Ihre eigenen Transportaufträge beschränkt. Sie können auch Auslieferungstransport von SAP auswählen.

### **❁** Beispiel

Die Suche nach Support Packages oder Enhancement Packages liefert keine Treffer. Dies geschieht, wenn Sie ein Support Package installiert haben, ohne das Update mit dem SAP Maintenance Planner einzuplanen. Versuchen Sie, nach Transportaufträgen zu suchen und filtern Sie nach den relevanten Stücklisten (*Stückliste für Support-Paket* (D) oder *Stückliste für Upgrade* (P)).

### **i** Hinweis

Sie können eine Liste von Transportaufträgen im CSV-Format hochladen.

- **Objektliste:** Führen Sie die Änderungseinfluss-Analysen für Objekte aus, die Sie ändern möchten. Die Objektnamen können Sie direkt eingeben oder über eine CSV-Datei importieren.
- **Änderungsvorgang:** Hier können Sie den Einfluss von Änderungsvorgängen aus dem Change-Request-Management analysieren.

Voraussetzungen:

- Sie verwenden das Change-Request-Management.
- In Ihrem SAP-Solution-Manager-System existiert mindestens ein Änderungsdokument. Die Änderungen wurden bereits implementiert. Änderungen werden in Transportaufträgen in einem verwalteten System erfasst.

3. Geben Sie unter **2** das System und den Mandanten an, in denen Sie Änderung durchgeführt haben oder Änderungen planen.

Wenn Sie Änderungsvorgänge analysieren, wählen Sie die Art des Änderungsvorgangs.

- **Änderungszyklus:** Analysieren Sie alle Änderungsdokumente für ein oder mehrere Änderungszyklen.
- **IT-Anforderung/Änderungsantrag:** Analysieren Sie alle Änderungsdokumente für mindestens eine IT-Anforderung oder mindestens einen Änderungsantrag.
- **Änderungsdokument:** Analysieren Sie nur angegebene Änderungsdokumente.

4. Geben Sie unter Schritt 3 Details zu der Änderung an, die Sie analysieren wollen (z.B. die Nummer des Transportauftrags, die Nummer des Änderungsantrags oder die Namen der Objekte).

Sie können eine Liste der Änderungen im CSV-Format hochladen. Um die gewünschte Datenstruktur zu prüfen, gehen Sie folgendermaßen vor:

1. Wählen Sie *Hochladen*.  
Ein Dialogfenster wird angezeigt.

#### **i** Hinweis

Sie können eine Liste der Objekte als XLS-Datei herunterladen. Wählen Sie hierzu *Herunterladen*.

Diese XLS-Datei können Sie mit MS Excel als CSV-Datei sichern. Die CSV-Datei können Sie zu einem späteren Zeitpunkt für weitere Analysen über *Hochladen* hochladen und so die Objektliste importieren.

2. Wählen Sie *Soforthilfe*.

#### **i** Hinweis

Um lange Antwortzeiten für die Eingabehilfe zu vermeiden, geben Sie Suchkriterien an, die so spezifisch wie möglich sind. Füllen Sie zumindest das Feld *Objekttyp* aus, und grenzen Sie den Objektnamen oder das übergeordnete Objekt mit Platzhaltern so weit wie möglich ein. Wenn dies nicht möglich ist, geben Sie das Paket oder den Objektverantwortlichen an.

5. Geben Sie unter Schritt 4 an, welchen Branch oder welchen Umfang Sie analysieren möchten.

Wenn kein Umfang ausgewählt wird, findet keine Umfangsbeschränkung statt.

Wenn ein site-aktivierter Branch ausgewählt wird, wählen Sie eine zu analysierende Site. Die Analyse berücksichtigt nur Inhalte der Lösungsdokumentation und TBOMS, die für die ausgewählte Site gültig sind. Dies gilt, wenn der Inhalt der Lösungsdokumentation zu der ausgewählten Site zugeordnet ist oder sein Site-Attribut leer ist.

Wenn Sie *Verwendete ausführbare Einheiten und Prozessschrittoriginals ausschließen* markieren, schließt das System die folgenden Objekte aus:

- Ausführbare Originale, die in Szenarios, Prozessen, Prozessschritt-Referenzen oder Prozessschritt-Originalen referenziert werden.
- Prozessschritt-Originale, die im Prozessschritt referenziert werden, in dem System, in dem Sie Ihre Testfälle verwalten.

Die BPCA-Analyse berücksichtigt ausschließlich nicht referenzierte ausführbare Originale und Prozessschritt-Originale. Dadurch wird sichergestellt, dass verwendete ausführbare Einheiten und Prozessschritte nicht mehrfach (als Original und als Referenzen) analysiert werden. Prozessschritt-Originale, die in Prozessschritt-Referenzen referenziert werden.

Wenn Sie diese Option nicht markieren, werden Originale in die Änderungseinfluss-Analyse einbezogen, unabhängig davon, ob sie verwendet werden oder nicht.

6. Geben Sie unter Schritt 5 eine Beschreibung für die Analyse an.
7. Optional: Um weitere optionale Parameter für die Analyse anzugeben, klicken Sie unter Schritt 6 auf *Details*. Sie können die folgenden optionalen Parameter verwenden:
  - *Testfall-Speicherort*

Wenn Sie Ihre Testfälle nicht zentral verwalten, können Sie hier die Systemrolle des Systems wählen, in welchem Sie die Testfälle verwalten. Wenn Sie Ihre Testfälle zentral verwalten, hat diese Auswahl keine Wirkung.

- **Filteroptionen für Geschäftsprozess-Knoten**
  - **Logische Komponentengruppe:** Sie können die Analyse auf ausführbaren Einheiten beschränken, die einer angegebenen logischen Komponentengruppe zugeordnet sind.
  - **Erweiterte Parameter:** Wenn Sie bestimmten Knoten in der Lösungsdokumentation Attribute zugeordnet haben, können Sie diese Attribute verwenden, um die Knoten für die Änderungsanalyse einzuschränken.
  - **Filteroptionen für TBOMs:** Geben Sie an, ob Sie die Analyse auf Objekte mit einer bestimmten Gewichtung beschränken möchten.  
Weitere Informationen finden Sie unter [Bearbeiten der TBOM-Gewichtung \[Seite 156\]](#).
  - **Filteroptionen für ausführbare Einheiten:** Geben Sie hier weitere Parameter zur Einschränkung der ausführbaren Einheiten an, die für die Analyse relevant sind.  
Sie können zusätzlich für jeden Typ von ausführbarer Einheit weitere Filterkriterien angeben.
- 8. Um die Analyse sofort zu starten, wählen Sie [Ausführen](#).
- 9. Wenn Sie die Analyse später starten möchten, gehen Sie wie folgt vor:
  1. Wählen Sie [Einplanen](#).
  2. Wählen Sie den gewünschten Startzeitpunkt, und bestätigen Sie Ihre Eingaben.

Im oberen Bildschirmbereich wird die Ergebnis-ID angezeigt, unter der Sie die Analyseergebnisse finden können.

10. Aktualisieren Sie die Ergebnisliste, um Ihre Analyseergebnisse auswählen zu können.

#### **i Hinweis**

Wenn der Aktualisierungsvorgang mit einer Zeitüberschreitung abbricht (etwa bei der Analyse großer SAP-Solution-Manager-Projekte oder -Lösungen), planen Sie die Aktualisierung als Hintergrundjob ein.

Eingeplante, aber noch nicht ausgeführte Analysen sind ebenfalls in der Ergebnisliste enthalten. Für diese Analysen wird das geplante Startdatum und die geplante Uhrzeit angezeigt. Nach der Ausführungszeit können Sie die Analyseergebnisse genau wie bei den anderen Analysen aufrufen.

## **Ergebnisse**

Sie haben die Analyse durchgeführt. Sie können nun die Analyseergebnisse auswerten und aus den Analyseergebnissen Testpläne erstellen.

## **7.4 Anzeigen von Details zu Änderungsvorgängen**

Für Änderungseinfluss-Analysen von [Änderungsvorgängen](#) können Sie sich weitere Details anzeigen lassen.

## Voraussetzungen

Sie verwenden das Change-Request-Management

## Vorgehensweise

1. Führen Sie eine Änderungseinfluss Analyse für *Änderungsvorgänge* aus.
2. Wählen Sie in Schritt 3 einen oder mehrere Änderungsvorgänge.
3. Wählen Sie *Details*.

## Ergebnisse

Alternativ können Sie die folgenden Aktionen für eine abgeschlossene Analyse der Art „Änderungsvorgang“ ausführen:

1. Wählen Sie *Parameter*.
2. Markieren Sie in Schritt 3 des Dialogs einen oder mehrere Änderungsvorgänge und wählen Sie *Details*.

Als Ergebnis erscheint ein Dialogfenster und zeigt Details zu den ausgewählten Änderungsvorgängen an:

- Änderungszyklus-ID, Art, Bezeichnung und Status
- ID der IT-Anforderung bzw. des Änderungsantrags, Art, Beschreibung und Status
- PPM-Projekt und -Aufgabe (falls zutreffend)
- Änderungsbeleg-ID, Beschreibung und Art
- Transportauftrag und Beschreibung des Transportauftrags
- System und Mandant, aus dem die Informationen zum Transportauftrag abgerufen werden

## 7.5 Analysieren der von Änderungen betroffenen Geschäftsprozessvarianten

Sie können eine Prüfung auf der Ebene der Geschäftsprozessvarianten ausführen. Prüfen Sie, ob die Änderungen sich auf Ihre Geschäftsprozesse auswirken. Prüfen Sie, ob neue Tests erforderlich sind.

## Voraussetzungen

- Sie haben Geschäftsprozessvarianten in der Lösungsdokumentation gepflegt.
- Sie haben TBOMs für die ausführbaren Einheiten von Geschäftsprozessvarianten angelegt.
- Sie haben Testfälle entweder direkt Geschäftsprozessvarianten oder Geschäftsprozessschritten zugeordnet, die den Geschäftsprozessvarianten zugeordnet sind.

## Vorgehensweise

1. Führen Sie in der Anwendung *Business Process Change Analysis* eine Änderungseinfluss-Analyse für einen Branch aus, der Geschäftsprozessvarianten enthält.
2. Prüfen Sie, ob Geschäftsprozessvarianten von der Änderung betroffen sind.

In der folgenden Tabelle wird beschrieben, unter welchen Bedingungen Geschäftsprozessvarianten und die einzelnen, den Varianten zugeordneten Prozessschritte im Ergebnis der Änderungseinfluss-Analyse angezeigt werden:

			Kennzeichen „Geschäftsprozessstest“ in Lösungsdokumentation ist festgelegt auf:	Geschäftsprozessvariante gehört zu Analyseergebnis in Testumfangsoptimierung (TUO)	Einzelne Geschäftsprozessschritte gehören zum Analyseergebnis und zur TUO
Ohne GP-Schritte	Ausführbare Einheit mit TBOM in GP-Variante	Durchgängige Testfälle nicht relevant	Nicht relevant	Ja	Entfällt
	Keine ausführbare Einheiten in GP-Variante	Durchgängige Testfälle nicht relevant	Nicht relevant	Nein	Entfällt
Mit GP-Schritten in GP (mit TBOMs)	Ausführbare Einheiten mit TBOM in GP-Variante	Durchgängiger Testfall nicht relevant	Kein durchgängiger Test erforderlich	Ja*	Ja
			Durchgängiger Test erforderlich	Ja**	Nein
	Keine ausführbare Einheiten in GP-Variante	Mit durchgängigen Testfällen in GP-Variante	Kein durchgängiger Test erforderlich	Ja*	Ja
			Durchgängiger Test erforderlich	Ja**	Nein
Keine ausführbare Einheiten in GP-Variante	Ohne durchgängige Testfälle in GP-Variante	Kein durchgängiger Test erforderlich	Nein	Ja	
		Durchgängiger Test erforderlich	Ja**	Nein	

\* = Nur betroffene ausführbare Einheiten, die in durchgängigen Tests getestet werden, werden berücksichtigt.

\*\* = Alle ausführbaren Einheiten von zugeordneten Prozessschritten gelten als betroffen, wenn mindestens eine Einheit betroffen ist.

## Ergebnisse

Sie haben die Analyse durchgeführt. Sie können nun die Analyseergebnisse auswerten und aus den Analyseergebnissen Testpläne erstellen.

## 7.6 Analyseergebnisse auswerten und Testpläne erstellen

### Kontext

Passen Sie den Testumfang an, und generieren Sie einen Testplan.



## Vorgehensweise

1. Wählen Sie aus der Ergebnisliste einen Ergebnisdatensatz.
  - Mit [Parameter](#) können Sie die Einstellungen prüfen, die zum ausgewählten Ergebnisdatensatz geführt haben.
  - Mit [Extern anzeigen](#) können Sie den ausgewählten Ergebnisdatensatz in einem neuen Fenster anzeigen.
2. Wählen Sie in der oberen Detailtabelle ([Ergebnis-ID ...](#)) die Sicht, die Sie auswerten möchten.
  - Wählen Sie [Testplan](#) [Testplan anlegen](#), um einen Testplan aus allen Testfällen für die betroffenen ausführbaren Einheiten zu generieren.
  - Verwenden Sie [Testplan](#) [Testplan erweitern](#), um einen bestehenden Testplan um Testfälle der betroffenen ausführbaren Einheiten zu erweitern. Das System ermöglicht das Erweitern aller Testpläne für die Lösung und den Branch, die Sie ausgewählt haben.
  - Objekte sind häufig mehreren Knoten in der Lösungsdokumentation zugeordnet und kommen in mehreren Testfällen vor. Über [Testumfang optimieren](#) können Sie den Umfang der anzulegenden Testpläne so optimieren, dass Redundanzen vermieden und möglichst viele Objekte mit möglichst wenig Testaufwand getestet werden. Weitere Informationen finden Sie unter [Optimieren des Testumfangs \[Seite 180\]](#).
3. Wählen Sie [Details anzeigen](#), um die betroffenen ausführbaren Einheiten in der unteren Detailtabelle anzuzeigen.
  - Mit [Gesamte Schnittmenge anzeigen](#) können Sie die betroffenen Objekte aller ausführbarer Einheiten anzeigen.
  - Mit [Schnittmenge anzeigen](#) können Sie die betroffenen Objekte der ausgewählten ausführbaren Einheit anzeigen. Weitere Informationen zu dieser Anzeige finden Sie unter [Prüfen des TBOM-Contents \[Seite 179\]](#).  
Die Funktion [Schnittmenge anzeigen](#) befindet sich unter der mehrfach belegten Drucktaste, die standardmäßig zunächst die Funktion [Gesamte Schnittmenge anzeigen](#) anbietet.
  - [Details aller Vereinfachungselemente](#) zeigt die Details der SAP-S/4HANA-Vereinfachungseinflüsse auf alle betroffenen Entitäten.
  - [Vereinfachungselementdetails](#) zeigt die Details der SAP-S/4HANA-Vereinfachungseinflüsse auf die ausgewählten Entitäten.

### **i** Hinweis

[Vereinfachungselementdetails](#) ist nur verfügbar, wenn ein SAP-S4/HANA-Release-Wechsel analysiert wurde.

- Mit [Zusätzliche Spalten](#) [Statistiken anzeigen](#) können Sie statistische Informationen zu den Objekten der ausführbaren Einheiten anzeigen (Gesamtanzahl der Objekte und Anzahl der Objekte in den einzelnen Klassifikationskategorien).

4. Wenn zu viele nicht relevante Änderungen angezeigt werden, die zu einem unverhältnismäßigen Testaufwand führen würden, definieren Sie einen TBOM-Filter, und wiederholen Sie die Analyse.

Weitere Informationen finden Sie unter [TBOM-Filter bearbeiten \[Seite 178\]](#).

## 7.6.1 TBOM-Filter bearbeiten

### Verwendung


Verwenden Sie TBOM-Filter, wenn eine Geschäftsprozess-Änderungsanalyse zu viele nicht relevante Änderungen enthält, die zu einem unverhältnismäßigen Testaufwand führen würden.

Legen Sie TBOM-Filter fest, um bestimmte, in der TBOM enthaltene Objekte zu ignorieren. Dies hat zur Folge, dass diese Objekte nicht in der Geschäftsprozess-Änderungsanalyse enthalten sind. Die Objekte werden nicht im Ergebnis angezeigt, auch wenn diese Objekte von einer Änderung betroffen sind.

Weitere Informationen finden Sie unter [Analysieren der von Änderungen betroffenen Geschäftsprozesse \[Seite 170\]](#).

Sie können den TBOM-Filter für einen Benutzernamen definieren oder allgemeingültig für alle Benutzer.

### Vorgehensweise

1. Wählen Sie in der Anwendung *Verwaltung - Änderungseinfluss-Analyse* den Pfad **► BPCA-Vorbereitung ► TBOM-Filter bearbeiten** .
2. Wählen Sie *TBOM-Filtertabelle bearbeiten*.
3. Wählen Sie *Anzeigen <--> Ändern*, um in den Bearbeitungsmodus zu wechseln.
4. Geben Sie die gewünschten Filterbedingungen für die TBOM-Eigenschaften an.
5. Prüfen Sie die Filterbedingungen.
6. Sichern Sie Ihre Eingaben.

### Filterbedingungen prüfen

1. Um eine einzelne Zeile zu prüfen, doppelklicken Sie in die betreffende Zeile.  
Um alle Zeilen zu prüfen, wählen Sie *Prüfen*.
2. Das Prüfergebnis zeigt alle fragwürdigen oder ungültigen Filterbedingungswerte an. Diese entstehen z.B. durch falsche TBOM-IDs, falsche Benutzernamen oder doppelte Einträge.
3. Wählen Sie *Warnungen löschen und fortfahren*, um ungültige Einträge zu löschen.
4. Wenn Sie den TBOM-Filter sichern, wird die Prüfung ebenfalls aufgerufen, Sie können jedoch auch ungültige Filterbedingungen sichern. Die Änderungseinfluss-Analyse ignoriert ungültige Filterbedingungen.

### Filterbedingungen automatisch anlegen

1. Sie können Filterbedingungen auch aus einer Objektliste erzeugen. In diesem Fall sind viele Feldwerte vordefiniert.  
Öffnen Sie eine der folgenden Ansichten:
  - TBOM-Content-Anzeige
  - Schnittmengenanzeige einer Änderungseinfluss-Analyse
  - Anzeige der ungetesteten Objekte bei der Optimierung des Testumfangs
  - Ergebnisanzeige der Aktualitätsprüfung
2. Markieren Sie in der Übersichtstabelle eine oder mehrere Zeilen auf der Ebene der logischen Komponenten oder Pakete. Alternativ können Sie auch im Detailreport eine oder mehrere Zeilen markieren.

3. Wählen Sie *Zu Filter hinzufügen*.  
Das System erstellt aus den ausgewählten Zeilen automatisch eine passende Filterbedingung. Ein Dialogfenster zeigt standardmäßig die neue Filterbedingung an.  
Wenn Sie logische Komponenten oder Pakete ausgewählt haben, sind nur diese Felder vorbelegt. Wenn Sie einzelne Objekte ausgewählt haben, sind alle Felder bis auf Benutzername und TBOM-Gültigkeit vorbelegt.
4. Um die einzelnen Felder für die Filterbedingungen anzupassen, bearbeiten Sie die Werte direkt im Dialogfenster.  
Sie können z.B. einen Festwert durch ein Maskierungszeichen ersetzen, um eine Filterbedingung zu verallgemeinern. Wir empfehlen, die Filterbedingungen so allgemein wie möglich zu definieren.
5. Um mehrere Filterbedingungen gleichzeitig zu ändern, gehen Sie folgendermaßen vor:
  1. Wählen Sie *Massenänderung*.  
Wenn Sie nur eine einzelne Zeile markiert haben, sind die Felder mit den Werten dieser Filterbedingung vorbelegt. Wenn Sie mehrere Zeilen markiert haben, sind die Felder mit Maskierungszeichen vorbelegt.
  2. Ändern Sie die Filterbedingungen.
  3. Stellen Sie sicher, dass im rechten Teil des Dialogfensters die Ankreuzfelder für jene Felder markiert sind, deren Werte Sie übernehmen wollen.
  4. Um die Werte in die markierten Zeilen zu übernehmen, wählen Sie *Übernehmen*.  
Um die Werte in alle Zeilen zu übernehmen, wählen Sie *Für alle übernehmen*.
6. Markieren Sie alle Filterbedingungen, die Sie sichern wollen.
7. Wählen Sie *Prüfen*.  
Das System prüft alle markierten Filterbedingungen. Die Spalte *Status* zeigt an, ob eine Filterbedingung gültig ist.  
Das System erzeugt Warnungen, wenn der Filter bereits eine identische oder eine allgemeiner formulierte Filterbedingungen enthält. Sie können die betreffende Zeile expandieren, um die bestehende identische oder allgemeinere Bedingung schreibgeschützt einzublenden.
8. Sichern Sie Ihre Eingaben.  
Es werden alle ausgewählten Filterbedingungen, einschließlich derjenigen mit Warnungen, im System gesichert, diese werden jedoch bei der Änderungseinfluss-Analyse ignoriert.

## 7.6.2 Prüfen des TBOM-Contents

### Kontext

Wenn Fehler während der Analyse auftreten, prüfen Sie den TBOM-Content.

#### ❁ Beispiel

Sie haben ein benutzerdefiniertes Programm geändert, aber die Änderung kommt in den Ergebnissen Ihrer Änderungsanalyse nicht vor. Prüfen Sie, ob das geänderte Objekt in der TBOM vorhanden ist.

## Vorgehensweise

1. Führen Sie in der Sicht *Lösungsdokumentation* einen Drilldown zur TBOM aus, und wählen Sie den Namen aus, um sie zu öffnen.
2. Wählen Sie *Content anzeigen*.

Wenn es zu Ihrer TBOM Erweiterungen gibt, wählen Sie außerdem *Inhalt anzeigen* im Dialog *Erweiterungen*.

Die Anzeige enthält sowohl statistische Informationen als auch eine detaillierte Liste aller Objekte.

3. Wählen Sie den relevanten **Content-Typ (Inhaltstyp)**.

Wenn die TBOM sowohl statische als auch dynamische Erweiterungen enthält, können Sie über die Auswahlknöpfe im Kopfbereich angeben, welche Art von Erweiterung Sie anzeigen möchten.

Die *Grafische Übersicht* zeigt die statistische Verteilung der Objekte auf die TBOM-Klassifikationsarten und Softwarekomponenten.

Die *Tabellenübersicht* zeigt die absolute und prozentuale Verteilung aller Objekte der TBOM auf die Klassifikationsarten.

4. Expandieren Sie die Daten in der tabellarischen Übersicht.

Sie erhalten die absolute und prozentuale Verteilung der Objekte für folgende Detailebenen:

- Logische Komponentengruppe (bei systemübergreifenden TBOMs werden mehrere logische Komponentengruppen angezeigt)
- Softwarekomponente
- Paket (Entwicklungs-klasse)

Die prozentualen Angaben beziehen sich jeweils auf den Anteil am Gesamt-Content. Die Anzahl der Objekte vom Klassifikationstyp *Tabelleninhalt* ergibt sich aus der Anzahl der Tabellenschlüssel, die in der TBOM aufgezeichnet wurden. Wenn für eine Tabelle keine Tabellenschlüssel aufgezeichnet wurden, wird die Tabelle als ein einziges Objekt gezählt.

5. Wählen Sie in der *Tabellenübersicht* ein Paket aus.

Die Liste der betroffenen Objekte wird in einem *Detailreport* angezeigt.

Wenn es sich bei den angezeigten Objekten um Tabellen handelt, können Sie in der Spalte *Schlüssel existiert* sehen, ob für den Tabelleneintrag Tabellenschlüssel aufgezeichnet wurden.

6. Um einzelne Tabellenschlüssel anzuzeigen, markieren Sie die Zeile oder die Zeilen für Objekte mit Tabellenschlüsseln, und wählen Sie *Tabellenschlüssel einblenden*.

Ein Report unterhalb des Detailreports listet die **Tabellenschlüssel** zu den ausgewählten Objekten auf.

7. Wählen Sie *Details anzeigen*, um die einzelnen Werte eines Schlüssels in einem separaten Dialogfenster anzuzeigen.

## 7.7 Optimieren des Testumfangs

## Kontext

Je größer eine Änderung an Ihrem System ist, desto stärker wirkt sich diese Änderung auf Lösungsdokumentationsknoten aus. Wenn Sie beispielsweise ein Support Package einspielen, ist oftmals die gesamte Lösungsdokumentation davon betroffen. Dies bedeutet, dass die Änderungseinfluss-Analyse den Testumfang zunächst nicht verringert.

Die von der Änderung betroffenen Objekte sind mehreren Lösungsdokumentationsknoten zugeordnet und werden somit auch mehrfach getestet. Mit der Funktion kann eine Testumfangsoptimierung (TUO) durchgeführt werden, sodass jedes Objekt mindestens einmal, und nicht mehrmals, getestet wird. Dadurch können Sie den Testaufwand deutlich reduzieren.

### i Hinweis

Das System optimiert den Testumfang unter der Annahme, dass ein Objekt vollständig getestet ist, sobald es mindestens einmal getestet wurde. Auch wenn ein bestimmtes Objekt in mehreren Geschäftsprozessen verwendet wird, genügt für die Optimierung also der Test eines einzigen Lösungsdokumentationsknotens (Optimierungsstrategie „Jedes Objekt nur einmal testen“). Wenn Sie bestimmte Objekte im Kontext aller Knoten testen wollen, müssen Sie diese Objekte von der Optimierungsstrategie ausschließen.

Um eine Optimierung zu ermöglichen, bringt das System alle Lösungsdokumentationsknoten in eine bestimmte Optimierungsreihenfolge. Dazu sortiert das System die Knoten anhand ihrer Testeffektivität. Die Testeffektivität ist eine Berechnung aus dem Verhältnis Anzahl der Objekte zu benötigtem Zeitaufwand. Abhängig von den Optimierungsoptionen verwendet das System dabei entweder den Gesamtzeitaufwand oder nur den Zeitaufwand für manuelle Tests.

Sie können die Optimierungsreihenfolge auf mehrere Arten übersteuern. Entsprechend den Übersteuerungsarten werden die Knoten in folgende Bereiche eingeteilt:

1	Manuell ausgewählte Knoten
2	Zwangseinschlussbereich: Knoten, die in den Testumfang eingeschlossen werden müssen
3	Vorzugsbereich: Knoten, die in den Lösungsdokumentationsoptionen als „Bevorzugt berücksichtigen“ gekennzeichnet sind
4	Nicht übersteuerte Knoten: alle Knoten, die in keinem anderen Bereich enthalten sind
5	Zwangsausschlussbereich: Knoten, die aus dem Testumfang ausgeschlossen sind

Die Bereiche 3-4 werden auch **Optimierungsbereiche** genannt.

Innerhalb der Bereiche sind die Knoten nach ihrer Testeffektivität sortiert. Wenn Sie einen Testumfang oder die verfügbare Zeit festlegen, übernimmt die Optimierungsreihenfolge Knoten in das Testergebnis oder entfernt sie daraus. Der Testumfang enthält immer alle Knoten in den Bereichen 1 und 2 und die meisten Knoten in den Bereichen 1 bis 4.

Falls nicht allen Knoten Testfälle zugeordnet sind, können Sie den Aufwand zum Anlegen und Ausführen manueller oder automatischer Testfälle berechnen.

## Vorgehensweise

1. Wählen Sie in der Ergebnisliste den gewünschten Analyseergebnis-Datensatz.

Dadurch wird die Beschreibung des Analyseergebnisses angezeigt.

2. Wählen Sie *Testumfang optimieren*.

Die Abdeckung der getesteten Objekte und der Zeitaufwand werden in einem separaten Fenster dargestellt. Die Registerkarten enthalten folgende Informationen:

- *Ergebnisübersicht*: Der durch die Optimierung des Testumfangs zu erwartende Effekt, verglichen mit der Gesamtzahl der auszuführenden Testfälle und der Anzahl der nach der Geschäftsprozess-Auswirkungsanalyse verbleibenden Testfälle.
  - *Testumfangsoptimierung: Rangfolge*: Übersicht der durch Änderungen betroffenen Knoten. Die Knoten sind nach Testeffektivität sortiert.  
Um eine Übersicht aller Knoten anzuzeigen, aggregiert das System diese automatisch. Um einzelne Knoten anzuzeigen, wählen Sie *Grafikeinstellungen*, geben Sie die Knoten an (anhand der Zahlen von 1 bis 20 oder anhand ihrer Testabdeckung von 80 % bis 100 %), und setzen Sie die Granularität auf **1**.
  - *Lösungsdokumentation*: Der zu erwartende Testaufwand pro Lösungsdokumentationsknoten. Objekte, die nicht im Testumfang enthalten sind.
  - *Testfallempfehlungen*: Die Testfalloptimierung berechnet den Aufwand für das Anlegen manueller oder automatisierter Testfälle. Diese Berechnung hilft, den Prozentsatz der durch Testfälle abgedeckten Objekte zu steigern und den langfristigen Aufwand für die Durchführung von Testfällen zu reduzieren. Weitere Informationen finden Sie unter [Empfehlungen für Testfälle generieren \[Seite 193\]](#).
  - *Vereinfachungselemente*: Auf dieser Registerkarte werden Einzelheiten zu den Auswirkungen der SAP-S/4HANA-Vereinfachung für alle betroffenen Knoten der Lösungsdokumentation angezeigt.
3. Bestimmen Sie den Testumfang entsprechend Ihren Erfordernissen. Folgende Optionen sind verfügbar:
    - [Optimierungsoptionen \[Seite 186\]](#)
    - [Festlegen von Testumfang oder Zeitaufwand \[Seite 195\]](#)
    - [Anpassen des Testumfangs in einer Grafik \[Seite 196\]](#)
    - [Testumfang manuell übersteuern \[Seite 198\]](#)
    - [Objekte prüfen, die nicht im Testumfang enthalten sind \[Seite 199\]](#)
    - [Ausschließen von Objekten von der Optimierung \[Seite 200\]](#)

### **i** Hinweis

Das System berücksichtigt Ihre Vorgaben in folgender Reihenfolge:

1. Manuell übersteuerte Knoten  
Knoten, die Sie durch manuelles Übersteuern hinzugefügt haben, sind im Testumfang enthalten. Knoten, die Sie manuell aus dem Testumfang entfernt haben, sind nicht im Testumfang enthalten (Zwangsausschlussbereich).
2. Lösungsdokumentationsoptionen, Gewichtungsoptionen, Vereinfachungselement-Optionen:  
Diese Optionen können die Lösungsdokumentationsknoten in Bereiche der Testumfangsoptimierung stellen. Es ist möglich, dass die unterschiedlichen Einstellungen im Konflikt zueinander stehen.  
Folgende Bereiche stehen zur Verfügung: Die Liste ist nach Präferenz sortiert:
  - Zwangseinschlussbereich

- Vorzugsbereich
- Zwangsausschlussbereich
- Optimierungsbereich (Knoten ohne spezielle Behandlung)

Zur Lösung dieser Konflikte wird folgende Regel angewendet:

**Ein Knoten wird entsprechend der oben angegebenen Reihenfolge dem obersten Bereich hinzugefügt.**

Beispiel:

Angenommen ein Knoten muss unter Verwendung diverser Optionen den folgenden Bereichen hinzugefügt werden. Die letzte Spalte zeigt, welcher Bereich gemäß der oben angegebenen Regel von der Testumfangsoptimierung gewählt wird.

Lösungsdokumentationsoptionen	Gewichtungsoptionen	Optionen für kundeneigene Entwicklungen und Änderungen	Vereinfachungselement-Optionen	Oberster Bereich = Bereich nach TUO
Zwangseinschlussbereich	Zwangsausschlussbereich	Optimierungsbereich	Optimierungsbereich	Zwangseinschlussbereich
Vorzugsbereich	Zwangsausschlussbereich	Zwangseinschlussbereich	Vorzugsbereich	Zwangseinschlussbereich
Optimierungsbereich	Zwangsausschlussbereich	Optimierungsbereich	Optimierungsbereich	Zwangsausschlussbereich
Optimierungsbereich	Optimierungsbereich	Zwangsausschlussbereich	Vorzugsbereich	Vorzugsbereich

### 3. Testumfang

Nach Berücksichtigung der manuellen Übersteuerung und der Optimierungsoptionen bestimmen Ihre Vorgaben bezüglich Testabdeckung und Zeitaufwand den Testumfang.

4. Sie können über **► Testplan > Testplan anlegen** einen Testplan für alle Testfälle für die betroffenen ausführbaren Einheiten anlegen.
5. Über **► Testplan > Testplan erweitern** können Sie einen Testplan um Testfälle der betroffenen ausführbaren Einheiten erweitern. Das System bietet Ihnen alle Testpläne zum Erweitern an, die für das ausgewählte Projekt bzw. die ausgewählte Lösung vorliegen.

## Gruppieren von ausführbaren Einheiten für die Testumfangsoptimierung

## Kontext

Hintergrund: In früheren Releases entsprach die feinste Granularität für die TUO einem kompletten Knoten in der Lösungsdokumentation. Alle Testfälle der betroffenen ausführbaren Einheiten eines Knotens mussten im Testumfang enthalten sein. Dadurch wurden höhere Optimierungsergebnisse bei Knoten mit zahlreichen ausführbaren Einheiten gefährdet.

Ab SAP Solution Manager 7.2 SPS 5 werden Knoten mit mehreren betroffenen ausführbaren Einheiten bei der TUO in kleinere Gruppen aufgeteilt:

- Ausführbare Einheiten, die in einem Testfall zusammengefasst sind, verbleiben in einer Gruppe.
- Ausführbare Einheiten, die als einzige Einheiten in einem Testfall enthalten sind, haben eine eigene Gruppe.
- Ausführbare Einheiten ohne Testfall haben ebenfalls eine eigene Gruppe. Die ausführbare Einheit testet sich selbst.

Wenn jede ausführbare Einheit durch einen eigenen Testfall und nicht gemeinsam mit anderen ausführbaren Einheiten getestet wird, maximiert dies die Anzahl der Gruppen.

Mehr Gruppen bedeuten ein höheres Optimierungspotenzial.

**Hinweis:** Standardmäßig wird angenommen, dass jeder Testfall alle ausführbaren Einheiten testet. Dadurch gibt es nur eine Gruppe für die Optimierung. Um eine Aufteilung in Gruppen zu ermöglichen, empfehlen wir die ausschließliche Zuordnung jeder ausführbaren Einheit zu mindestens einem eigenen Testfall in der Lösungsdokumentation. Als Ergebnis verfügt jede ausführbare Einheit über eine eigene Gruppe.

In der Anwendung *Business Process Change Analyzer* werden Ihnen unter *Testumfang optimieren* die Aufteilung in die einzelnen Gruppen über Knotentexte auf der Registerkarte *Testumfangsoptimierung: Rangfolge* (vergrößern Sie die Grafik) oder *Lösungsdokumentation* angezeigt.

Die Gruppen werden jedes Mal neu berechnet, wenn TUO-Testfalldaten aktualisiert oder die Testfalloptionen geändert werden.

## 7.7.1 Bearbeiten von Standardzeiten zum Anlegen und Ausführen von Testfällen

### Verwendung

Geben Sie Standardwerte an, damit das System die benötigte Zeit zum Anlegen und Ausführen von Testfällen berechnen kann.

### Vorgehensweise

Weitere Informationen:

- [Festlegen von Standard-Erstellungszeiten für Testfälle \[Seite 185\]](#)
- [Festlegen von Standardausführungszeiten für Testfälle \[Seite 185\]](#)



## 7.7.1.1 Festlegen von Standard-Erstellungszeiten für Testfälle

### Verwendung

Damit BPCA Testfallempfehlungen generieren kann, passen Sie die Standardwerte für den durchschnittlichen Aufwand zum Anlegen eines automatisierten oder manuellen Testfalls an. Die angepassten Werte dienen als Standardzeiten, wenn Sie das Attribut in den Testfallattributen nicht festlegen.

### Vorgehensweise

1. Wählen Sie in der Sicht *Verwaltung - Änderungseinfluss-Analyse* den Pfad ► *BPCA-Testumfangsoptimierung* ► *Standard-Erstellungszeit für Testfalltypen* ►.
2. Tragen Sie pro Testfalltyp die entsprechenden Zeiten ein.
3. Sichern Sie Ihre Eingaben.

## 7.7.1.2 Festlegen von Standardausführungszeiten für Testfälle

### Verwendung

Definieren Sie den Zeitaufwand für jeden Testfall je nach Testfalltyp. Diese Standardzeit wird verwendet, wenn Sie dieses Attribut in den Testfallattributen nicht festlegen. Verwenden Sie diese Einstellung auch für den Business Process Change Analyzer bei der Definition des Testumfangs. Hier kann der zeitliche Aufwand der Testfälle nach manuellem und automatisiertem Aufwand zusammengerechnet und gegenübergestellt werden.

### Vorgehensweise

1. Wählen Sie in der Sicht *Verwaltung - Testmanagement* den Pfad ► *BPCA-Testumfangsoptimierung* ► *Erforderliche Standardzeit für Testfalltypen* ►.
2. Tragen Sie pro Testfalltyp die entsprechenden Zeiten ein.
3. Sichern Sie Ihre Eingaben.

**Beispiel:** Sie können diese Einstellung auch für den Business Process Change Analyzer (BPCA) zur Definition des Testumfangs verwenden. Hier kann der zeitliche Aufwand der Testfälle nach manuellem und automatisiertem Aufwand zusammengerechnet und gegenübergestellt werden.

## 7.7.2 Optimierungsoptionen

Wenn Sie in der Sicht *Business Process Change Analyzer* eine Analyse ausgewählt und die Option *Testumfang optimieren* gewählt haben, können Sie im oberen Bildbereich die *Optimierungsoptionen* expandieren. Hier können Sie den Rang und die Gewichtung der Objekte in der Testumfangsoptimierung definieren.

Sie haben folgende Möglichkeiten:

- [Bevorzugen von Knoten mit dynamischen TBOMs \[Seite 186\]](#)
- [Bevorzugen von Knoten mit Testfällen \[Seite 186\]](#)
- [Bevorzugen von Knoten mit automatisierten Testfällen \[Seite 187\]](#)
- [Lösungsdokumentationsoptionen \[Seite 188\]](#)
- [Testfalloptionen \[Seite 189\]](#)
- [Gewichtungsoptionen \[Seite 191\]](#)
- [Testfallempfehlungen \[Seite 193\]](#)
- [Vereinfachungselement-Optionen \[Seite 192\]](#)

### 7.7.2.1 Bevorzugen von Knoten mit dynamischen TBOMs

Im Bereich *Optimierungsoptionen* können Sie wählen, ob Knoten mit dynamischen TBOMs bevorzugt behandelt werden sollen:

- Wenn Sie das Ankreuzfeld *Knoten mit dynamischen TBOMs in Optimierungsrangfolge bevorzugen* markieren, erhöht das System die Gewichtung der Objekte aus dynamischen TBOMs für die Berechnung der Testabdeckung. Dadurch wird die Testeffizienz von Knoten mit dynamischen TBOMs deutlich erhöht. Da das System die Knoten für die Optimierung nach ihrer Effizienz sortiert, werden Knoten mit dynamischen TBOMs bevorzugt behandelt. Diese Sortierung wirkt sich auch auf die Reihenfolge der im Diagramm angezeigten Knoten aus.
- Wenn Sie das Ankreuzfeld entmarkieren, erhalten Objekte aus dynamischen TBOMs keine Sonderbehandlung.

### 7.7.2.2 Bevorzugen von Knoten mit Testfällen

Im Bereich *Optimierungsoptionen* können Sie wählen, ob Knoten mit Testfällen gegenüber Knoten ohne Testfälle bevorzugt behandelt werden sollen.

- Wenn Sie das Ankreuzfeld *Knoten mit Testfällen in der Optimierungsrangordnung bevorzugen* markieren, wird die Testeffizienz von Knoten mit fehlenden Testfällen deutlich verringert. Da das System die Knoten für die Optimierung nach ihrer Effizienz sortiert, werden Knoten mit Testfällen bevorzugt behandelt. Diese Sortierung wirkt sich auch auf die Reihenfolge der im Diagramm angezeigten Knoten aus.
- Wenn Sie das Ankreuzfeld entmarkieren, wird der Testaufwand für Knoten mit fehlenden Testfällen nicht über die definierten Standardwerte hinaus erhöht.

## 7.7.2.3 Bevorzugen von Knoten mit automatisierten Testfällen

Im Bereich *Optimierungsoptionen* können Sie wählen, ob Knoten mit automatisierten Testfällen bevorzugt behandelt werden sollen:

- Wenn Sie das Ankreuzfeld *Knoten mit automatisierten Testfällen in der Optimierungsrangordnung bevorzugen* markieren, ignoriert das System den Zeitaufwand für automatisierte Testfälle. Dadurch werden Lösungsdokumentationsknoten mit automatisierten Testfällen deutlich effektiver als Knoten mit manuellen Testfällen. Da das System die Knoten für die Optimierung nach ihrer Effizienz sortiert, werden Knoten mit automatisierten Testfällen bevorzugt behandelt. Diese Sortierung wirkt sich auch auf die Reihenfolge der im Diagramm angezeigten Knoten aus.
- Wenn Sie das Ankreuzfeld entmarkieren, berücksichtigt das System sowohl für automatisierte als auch für manuelle Testfälle den jeweils angegebenen Zeitaufwand.

## 7.7.2.4 Testmultiplizitäten für geändert Objekte festlegen

### Kontext

Definieren Sie im Bereich *Optimierungsoptionen* die Testmultiplizitäten für geänderte Objekte:

### Vorgehensweise

1. Klappen Sie den Bereich *Optimierungsoptionen* auf.
2. Wählen Sie die Registerkarte *Allgemeine Optionen*.
3. Legen Sie die Testmultiplizität für geänderte Objekte nach Klassifikationsart fest. Standardmäßig werden die Testmultiplizitäten für alle Klassifikationsarten auf 1 gesetzt.
  - Die Testmultiplizität legt fest, wie oft ein geändertes Objekt getestet werden muss, damit es von der Testumfangsoptimierung als abgedeckt betrachtet wird. Die Standardeinstellung 1 bedeutet, dass geänderte Objekte mindestens einmal getestet werden müssen, um als abgedeckt betrachtet zu werden.
  - Testmultiplizitäten größer als 1 bieten mehr Sicherheit, falls sich Objekte an unterschiedlichen Standorten unterschiedlich verhalten. Der Nachteil dieser erhöhten Sicherheit ist eine Zunahme des Testaufwands, da Objekte mehrmals getestet werden müssen, um eine gewisse Testabdeckung zu erzielen.
  - Wenn die definierte Testmultiplizität höher ist als die Anzahl der von einem geänderten Objekt betroffenen Knoten, dann wird die Anzahl der betroffenen Knoten als Testmultiplizität für das Objekt verwendet.
  - Höhere Testmultiplizitäten sind für die Klassifikationsarten *Programme/Kundeneigene Objekte* und *Tabellen-Content* relevant.

## 7.7.2.5 Lösungsdokumentationsoptionen

### Voraussetzungen

Sie haben Attribute für Objekttypen definiert, die Attribute den Objekttypen zugeordnet und optional eine Signaturstrategie zugeordnet. Weitere Informationen finden Sie im Customizing für *SAP Solution Manager* unter ► *Capabilities (Optional)* ► *Einführung/Upgrade* ►.

### Kontext

In der Lösungsdokumentation können Sie Strukturattribute Knoten zuordnen und anhand dieser Attribute den Testumfang festlegen, indem Sie Testfälle, die ausgeführt werden müssen, unabhängig vom Optimierungsalgorithmus als nicht auszuführen oder bevorzugt auszuführen kennzeichnen.

Die Knoten im *Zwangseinschlussbereich* sind im Testumfang immer vorhanden.

Die Knoten im Vorzugsbereich werden als erste in den Testumfang aufgenommen, wenn die Vorgaben zu Testabdeckung und Zeitaufwand dies zulassen. Sämtliche Vorgaben bezüglich der Testabdeckung oder des Zeitaufwands wirken sich nur auf die übrigen Knoten aus, die Sie nicht anhand der Strukturattribute in den Testumfang aufgenommen bzw. daraus ausgeschlossen haben.

Die Knoten im *Zwangausschlussbereich* sind im Testumfang nicht vorhanden.

### Vorgehensweise

1. Klappen Sie den Anzeigebereich *Optimierungsoptionen* auf.
2. Wählen Sie die Registerkarte *Lösungsdokumentationsoptionen*.
3. Wählen Sie *Attribut hinzufügen*.
4. Geben Sie in der Spalte *Verwendete Attribute* das Attribut an, nach dem Sie filtern wollen.
5. Geben Sie in der Spalte *Wert* den Wert ein, den das Attribut haben muss, damit das System die zugehörigen Knoten in den Testumfang aufnimmt, bevorzugt behandelt oder vom Testumfang ausschließt.
6. Optional: In der Spalte *Wertebereich* können Sie komplexe Bedingungen für den Attributwert angeben.
7. Wählen Sie in der Spalte *Bereich* aus, ob Knoten mit den angegebenen Attributwerten in den Testumfang eingeschlossen, bevorzugt behandelt oder aus dem Testumfang ausgeschlossen werden sollen.
8. Wählen Sie *Attributwerte hinzufügen*, um weitere Werte für das markierte Attribut anzugeben.

Sie können die neuen Knoten beispielsweise je nach Attributwerten unterschiedlichen Bereichen zuordnen: Wert 1 – immer aufnehmen; Wert 2 – bevorzugt behandeln, Wert 3 – ausschließen.

9. Um ausgewählte Einträge aus der Liste der verwendeten Attribute zu entfernen, wählen Sie *Einträge löschen*.

10. Wählen Sie *Werte zurücksetzen*, um die Werte der ausgewählten Attribute zurückzusetzen.

11. Wählen Sie *Übernehmen und sichern*.

Der Testumfang wird im Diagramm unter *Testumfangsoptimierung: Rangfolge* durch einen hellgrünen Hintergrund gekennzeichnet. Die durch Strukturattribute hinzugefügten Knoten sind durch eine schwarze Linie von den übrigen Knoten getrennt. Diese Linie kennzeichnet den minimalen Testumfang. Die Knoten links dieser Linie können Sie nicht ändern, indem Sie auf einen weiter links liegenden Knoten im Diagramm klicken oder eine andere Testabdeckung oder einen anderen Zeitaufwand vorgeben.

12. Um Knoten aus dem Testumfang zu entfernen, die durch ein Strukturattribut hinzugefügt wurden, können Sie folgende Aktionen aus dem Testumfang durchführen:

- Strukturattribute ändern
- Knoten durch manuelles Übersteuern vom Testumfang ausschließen

## 7.7.2.6 Testfalloptionen

### Verwendung

Mit der Definition von Testfalloptionen können sie bestimmen, welche Testfälle für die im Testumfang enthaltenen Knoten ausgeführt werden sollen. Definieren Sie einen Testfallfilter. Die Testfalloptionen unterteilen sich in zwei Kategorien:

- Bereichsregeln für Testfälle: Diese Regeln werden auf alle Testfälle eines bestimmten Bereiches (Einschluss oder Optimierung) angewendet.
- Attributregeln für Testfälle.

### Voraussetzungen

Sie haben Testfallattribute für alle Testfälle gepflegt.

### Vorgehensweise

#### Testfallbereichsregeln angeben

1. Klappen Sie den Anzeigebereich *Optimierungsoptionen* auf.
2. Wählen Sie die Registerkarte *Testfalloptionen*.
3. Markieren Sie die entsprechenden Ankreuzfelder, um die Testfälle für Knoten im jeweiligen Bereich zu filtern:
  - *Automatisch*  
Wenn Sie dieses Ankreuzfeld markieren, schließt das System nur automatische Tests für diese Knoten in den Testumfang ein.  
Wenn Sie das Ankreuzfeld entmarkieren, berücksichtigt das System keine automatisierten Testfälle.
  - *Manuell*

Wenn Sie das Ankreuzfeld markieren, schließt das System nur manuelle Tests für diese Knoten im Testumfang ein.

Wenn Sie das Ankreuzfeld zurücksetzen, führt das System keine manuellen Testfälle aus.

- *Nur Automatisch (sofern verfügbar)*

Wenn Sie dieses Ankreuzfeld markieren, prüft das System, ob ein Knoten automatisierte oder manuelle Testfälle enthält. Wenn der Knoten mindestens einen automatisierten Testfall aufweist, werden nur die automatisierten Testfälle dieses Knotens berücksichtigt. Wenn der Knoten keine automatisierten Testfälle aufweist, werden alle manuellen Testfälle des Knotens berücksichtigt. Wenn Sie das Ankreuzfeld entmarkieren, berücksichtigt das System die Testfälle entsprechend den übrigen Ankreuzfeldern.

- *Nur mit Zuordnung zu Testobjekt (sofern verfügbar)*

Wenn Sie dieses Ankreuzfeld markieren, prüft das System für jeden Knoten in der Lösungsdokumentation, ob einem Testfall mindestens ein Testobjekt zugeordnet ist. Wenn das der Fall ist, berücksichtigt das System nur Testfälle mit zugeordnetem Testobjekt für diesen Knoten. Wenn Sie in einem Knoten keinem Testfall ein Testobjekt zugeordnet haben, berücksichtigt das System alle Testfälle.

Wählen Sie diese Option, wenn Sie Varianten von ausführbaren Einheiten verwenden und nur die Testfälle für die Variante auswerten möchten.

### **i** Hinweis

Um eine hundertprozentige Testabdeckung zu erzielen, müssen alle von der Analyse betroffenen ausführbaren Einheiten getestet werden. Wenn Sie das Ankreuzfeld *Nur mit Zuordnung zu Testobjekt (sofern verfügbar)* markiert haben, müssen Sie für eine hundertprozentige Testabdeckung Folgendes sicherstellen:

- Sie haben jede von der Analyse betroffene ausführbare Einheit in mindestens einem Testfall als Testobjekt zugeordnet.  
— oder —
- Sie haben keine der von Analyse betroffenen ausführbaren Einheiten in keinem Testfall als Testobjekt zugeordnet.

Wenn Sie nur einzelne ausführbare Einheiten nicht zugeordnet haben, führt diese Funktion möglicherweise dazu, dass Sie keine hundertprozentige Testabdeckung erzielen.

Wenn Sie das Ankreuzfeld entmarkieren, berücksichtigt das System die Testfälle entsprechend den übrigen Ankreuzfeldern.

### **Testfälle über Testfallattribute wählen**

Zusätzlich zu den Testfallregeln können Sie die Testfälle auch über Testfallattribute filtern:

1. Klappen Sie den Anzeigebereich *Optimierungsoptionen* auf.
2. Wählen Sie die Registerkarte *Testfalloptionen*.
3. Wählen Sie aus, ob Sie die Filterbedingungen für verschiedene Attribute mit **AND** oder **OR** verknüpfen möchten.
4. Wählen Sie *Attribut hinzufügen*.
5. Wählen Sie in der Spalte *Verwendete Attribute* das Attribut aus, nach dem Sie filtern möchten.
6. Geben Sie Attributwerte an:
  - Geben Sie in der Spalte *Attributwert* den Wert ein, den das Attribut haben muss, damit das System den zugehörigen Testfall berücksichtigt.
  - Geben Sie in der Spalte *Wertebereich* Bedingungen für den Attributwert an.

- In der Spalte *Sprache* können Sie die Sprache angeben, in der Sie die sprachabhängigen Attributwerte festlegen möchten. Wenn Sie das Feld leer lassen, sucht das System in allen verfügbaren Sprachen.
  - In den Spalten *Gültig für Zwangseinschlussbereich* und *Gültig für Optimierungsbereich* geben Sie an, ob Sie Testfälle in diesen Bereichen filtern wollen.
7. Wählen Sie *Attributwerte hinzufügen*, um weitere Werte für das markierte Attribut anzugeben. Sie können Testfälle mit spezifischen Attributwerten in den beiden Bereichen unterschiedlich behandeln. Sie können z. B. Testfälle mit den Prioritäten 1 und 2 in den *Zwangseinschlussbereich* und ausschließlich Testfälle mit Priorität 1 in den Optimierungsbereich einbeziehen.
  8. Um ausgewählte Einträge aus der Liste der verwendeten Attribute zu entfernen, wählen Sie *Einträge löschen*.
  9. Wählen Sie *Werte zurücksetzen*, um die Werte der ausgewählten Attribute zurückzusetzen.
  10. Wählen Sie *Übernehmen und sichern*.

## 7.7.2.7 Gewichtungsoptionen

### Voraussetzungen

Sie haben den Objekten in den TBOMs Gewichtungsangaben zugeordnet. Weitere Informationen finden Sie unter [Bearbeiten der TBOM-Gewichtung \[Seite 156\]](#).

### Kontext

Geben Sie an, in welchem Ausmaß die Gewichtung eines geänderten Objekts bei der Optimierung des Testumfangs berücksichtigt wird.

### Vorgehensweise

1. Klappen Sie den Anzeigebereich *Optimierungsoptionen* auf.
2. Wählen Sie die Registerkarte *Gewichtungsoptionen*.
3. Legen Sie fest, ob Gewichtungen bei der Optimierung berücksichtigt werden sollen.

Wenn Sie das Ankreuzfeld *Keine Gewichtungen bei der Optimierung verwenden* markieren, betrachtet das System alle Objekte als gleichwertig.

Nur wenn Sie die Auswahl des Ankreuzfelds aufheben, können Sie auf der Registerkarte *Gewichtungsoptionen* weitere Einstellungen vornehmen.

4. Wählen Sie einen Schwellenwert.

Nur Knoten, deren Objekte eine relevante Gewichtung haben, werden in den Testumfang aufgenommen. Wenn ein Knoten ausschließlich Objekte mit einer Gewichtung unterhalb des gewählten Schwellenwerts hat, verschiebt das System diesen Knoten in den *Zwangsausschlussbereich*.

5. Um den Einfluss einzelner Gewichtungen auf die Optimierungsreihenfolge zu ändern, wählen Sie *Multiplikatoren einblenden*.

Das System verwendet diese Zahlen bei der Berechnung der Test-Effektivität der einzelnen Geschäftsprozesshierarchie-Knoten. Die Test-Effektivität berechnet sich zunächst aus dem Verhältnis Anzahl der Objekte zu benötigtem Zeitaufwand. Wenn Sie jedoch die Gewichtungen mit berücksichtigen, multipliziert das System die Anzahl der Objekte mit den angegebenen Faktoren, je nach Gewichtung eines Objekts. Dadurch werden Knoten mit vielen kritischen Objekten deutlich effektiver als Knoten mit vielen unkritischen Objekten. Je effektiver ein Knoten ist, desto weiter vorne wird er in der Optimierungsreihenfolge eingereiht. Mit diesen Multiplikatoren können Sie also den Einfluss der einzelnen Gewichtungen auf die Optimierungsreihenfolge bestimmen.

## 7.7.2.8 Vereinfachungselement-Optionen

### Voraussetzungen

Sie haben eine BPCA-Analyse für einen SAP-S/4HANA-Release-Wechsel durchgeführt.

### Kontext

Geben Sie an, in welchem Ausmaß die Lösungsdokumentationsknoten, die von SAP-S/4HANA-Vereinfachungselementen betroffen sind, von der Testumfangoptimierung berücksichtigt werden.

### Vorgehensweise

1. Klappen Sie den Anzeigebereich *Optimierungsoptionen* auf.
2. Wählen Sie die Registerkarte *Vereinfachungselement-Optionen*.
3. Wählen Sie, ob alle betroffenen Objekte, einschließlich Einflüsse aus Vereinfachungselementen, berücksichtigt werden sollen oder ob die Testumfangoptimierung nur auf die Einflüsse aus Vereinfachungselementen eingeschränkt werden soll.
4. Wählen Sie pro Vereinfachungskategorie, ob Lösungsdokumentationsknoten, die durch Vereinfachungselemente einer bestimmten Kategorie betroffen sind, in einen bestimmten Bereich gestellt werden sollen.



Pro Kategorie können folgende Bereiche gewählt werden:

1. Zwangseinschlussbereich
2. Vorzugsbereich
3. Optimierungsbereich: Wenn Sie diese Option wählen, erhalten die passenden Lösungsdokumentationsknoten keine spezielle Behandlung, es sei denn, sie werden wegen anderer Optionen in einen anderen Bereich gestellt.

## 7.7.2.9 Testfallempfehlungen

### Verwendung

Sie können Empfehlungen für Testfälle generieren und den Aufwand für verschiedene Szenarios berechnen. Sie können beispielsweise den Aufwand für die folgenden Aktivitäten berechnen:

- Anlegen (manueller oder automatisierter) Testfälle und Verbessern der Testabdeckung von geänderten Objekten
- Verringern des Aufwands für die Ausführung von Testfällen durch Ersetzung manueller Testfälle durch automatische Testfälle

### Voraussetzungen

- Optional: Sie haben die Vorschlagswerte für den durchschnittlichen Aufwand zur Erstellung und Ausführung automatisierter oder manueller Testfälle angepasst. Weitere Informationen finden Sie unter [Bearbeiten von Standardzeiten zum Anlegen und Ausführen von Testfällen \[Seite 184\]](#).

### Vorgehensweise

Gehen Sie dazu folgendermaßen vor:

1. Definieren Sie die Objekte, für die die Testautomatisierung sinnvoll sein könnte und für die Sie Testfallempfehlungen wünschen.

#### ❖ Beispiel

Um Testfallempfehlungen für die geschäftskritischsten Prozesse zu erhalten, fügen Sie deren Knoten anhand von Strukturattributen in den *Zwangseinschlussbereich* auf der Registerkarte [Lösungsdokumentationsoptionen \[Seite 188\]](#) ein.

2. Geben Sie unter *Zieltestabdeckung mit automatisierten Tests* einen Wert für die Testabdeckung ein, beispielsweise 99 %. Weitere Informationen finden Sie unter [Anpassen des Testumfangs in einer Grafik \[Seite 196\]](#).  
Der erforderliche Testaufwand wird im Ergebnisbereich auf der Registerkarte *Testumfangoptimierung: Rangfolge* angezeigt.

### ❁ Beispiel

Wenn Sie alle Objekte testen, die mit geschäftskritischen Geschäftsprozessen zusammenhängen, erzielen Sie eine Testabdeckung von 30 % aller geänderten Objekte. Geben Sie diesen Wert als Zielabdeckung für automatisierte Testfälle ein.

- Um den für die Testautomatisierung erforderlichen und daraus resultierenden Aufwand zu berechnen, gehen Sie folgendermaßen vor:  
Wählen Sie *Aufwand für das Anlegen automatisierter Testfälle bis zu u.g. Abdeckung berechnen*. Geben Sie im Feld *Zieltestabdeckung mit automatisierten Tests* den Wert für den Prozentsatz an Objekten ein, der durch automatisierte Testfälle abgedeckt werden soll.

### ❁ Beispiel

Um alle Objekte abzudecken, die mit geschäftskritischen Geschäftsprozessen zusammenhängen, geben Sie den Wert **30** ein. Als Ergebnis sollen 30 % aller Objekte durch vollautomatisierte Testfälle abgedeckt werden.

Wenn manuelle Testfälle vorhanden sind, geht das System davon aus, dass sie durch automatisierte Testfälle ersetzt werden.

- Berechnen Sie den Aufwand, der erforderlich ist, um Testfälle für den **übrigen** Teil der Testabdeckung anzulegen und auszuführen.

### i Hinweis

Das System berechnet Folgendes:

- den Aufwand, der bei fehlenden Testfällen zum Anlegen neuer Testfälle erforderlich ist
- den Gesamtaufwand für die Ausführung vorhandener und neuer Testfälle

In unserem Beispiel berechnen Sie den Aufwand für die Differenz zwischen der Gesamtabdeckung von 99 % und der Zielabdeckung von 30 % der Objekte. Diese erreichen Sie, wenn Sie für die *Zwangseinschluss*-Geschäftsprozesse automatisierte Testfälle haben. Der Rest beträgt 69 % aller geänderten Objekte.

Wenn Sie nicht möchten, dass vorhandene manuelle Testfälle durch automatisierte Testfälle ersetzt werden, markieren Sie nicht das Ankreuzfeld *Aufwand zum Anlegen automatisierter Testfälle berechnen*. Legen Sie alternativ das Ziel für die automatisierte Testfallabdeckung auf 0 fest. Wenn Sie keine **vollständige** Testautomatisierung wünschen, bleibt die gewünschte Gesamtabdeckung.

Sie können die folgenden Szenarios ausprobieren:

- *Aufwand für das Anlegen automatisierter Testfälle berechnen (empfohlen)*: Wenn es für die übrigen 69 % keine Testfälle gibt, empfiehlt das System in unserem Beispiel das Anlegen automatisierter Testfälle. Das System berechnet den Aufwand zum Anlegen und Ausführen dieser Testfälle. Das System empfiehlt nicht, vorhandene manuelle Testfälle durch automatisierte Testfälle zu ersetzen. Folgendes Szenario wird empfohlen: Auf lange Sicht wird der Testaufwand minimiert, obwohl der anfängliche Aufwand, automatisierte Testfälle anzulegen größer ist, als der Aufwand, manuelle Testfälle anzulegen.
- *Aufwand für das Anlegen manueller Testfälle berechnen*: In unserem Beispiel gilt Folgendes: wenn es keine Testfälle gibt, empfiehlt das System, für die übrigen 69 % manuelle Testfälle anzulegen, da diese schneller als automatisierte angelegt werden können. Das System berechnet den Aufwand zum Anlegen und Ausführen dieser Testfälle.

- *Keine Empfehlungen*: Sie möchten die vorhandene Testfallabdeckung nicht ändern. In unserem Beispiel berechnet das System nur den Aufwand für die 30 % Zielabdeckung für die Testautomatisierung.
5. Wählen Sie *Übernehmen und sichern*.  
Auf der Registerkarte *Testfallempfehlungen* werden im Ergebnisbildbereich die aktuelle Situation und der berechnete Aufwand zum Anlegen und Ausführen zusätzlicher Testfälle in grafischer und tabellarischer Form angezeigt.
  6. Optional: Um manuelle oder automatisierte Testfälle anzulegen und diese der Lösungsdokumentation zuzuordnen, können Sie unter *Testfallempfehlungen - Details* eine Liste der empfohlenen Testfälle exportieren.
  7. Um die Auswirkungen neu angelegter und zugeordneter Testfälle anzuzeigen, wählen Sie *Testfalldaten aktualisieren*.

## 7.7.3 Festlegen von Testumfang oder Zeitaufwand

### Voraussetzungen

- Sie haben für Ihre Testfälle Zeitaufwände angegeben.
  - Sie haben die erforderliche Zeit direkt in die einzelnen Testfälle eingetragen oder
  - Standardzeitaufwände für die verschiedenen Testfalltypen eingegeben.
- Weitere Informationen finden Sie unter [Bearbeiten von Standardzeiten zum Anlegen und Ausführen von Testfällen \[Seite 184\]](#).

### Kontext

Mit dem Testumfang bestimmen Sie den Anteil der Objekte, die in mindestens einem getesteten Knoten der Lösungsdokumentation enthalten sind. Der Zeitaufwand hängt von den Testfällen ab, die Sie den Knoten zugeordnet haben. Je höher die Abdeckung ist, desto höher ist auch der Zeitaufwand, wobei Sie oft schon eine annähernd vollständige Abdeckung (z. B. 95 %) mit einem deutlich geringeren Aufwand (z. B. die Hälfte des Gesamtaufwands) erreichen können.

Im Anzeigebereich *Definition für Testumfang* sehen Sie eine Balkendarstellung der aktuell gewählten Testabdeckung und des erforderlichen Zeitaufwands, unterteilt in manuelle und automatisierte Testfälle. Die Markierungen innerhalb der Balkendiagramme kennzeichnen die Werte für Abdeckung bzw. Zeitaufwand, die Sie durch den Test einzelner Knoten erreichen können. Da das System immer die Gesamtheit aller Objekte und Testfälle eines Knotens zählt, sind nur bestimmte Werte für Abdeckung und Zeitaufwand möglich.

## Vorgehensweise

1. Wählen Sie die Zeiteinheit aus, und geben Sie den erforderlichen Wert in das Feld *Wert* neben einem Balken ein (z.B. dem Balken *Gesamter Testaufwand*), oder klicken Sie direkt auf den Balken.
2. Bestätigen Sie die Eingabe mit .

Das System berechnet die nächsthöhere mögliche Testabdeckung und den dazugehörigen Zeitaufwand. Die Balkendarstellungen werden automatisch aktualisiert. Auf der Registerkarte *Testumfangoptimierung: Rangfolge* wird der Testumfang im Diagramm unter *Grafische Sicht* durch einen hellgrünen Hintergrund angezeigt.

### i Hinweis

Wenn Sie keine Testabdeckung von 100 % erreichen, bedeutet das, dass es Objekte in Knoten ohne Testfälle gibt, oder dass Sie die Testfälle über die Optimierungsoptionen ausgeschlossen haben. In diesem Fall ist es nicht mehr möglich, alle geänderten Objekte in einem Knoten zu testen. Das System berücksichtigt solche Knoten zunächst nicht bei der Optimierung. Es reiht sie in der Optimierungsreihenfolge am Ende des Optimierungsbereiches ein. Um diese Knoten trotzdem in die Optimierung mit einzubeziehen, haben Sie folgende Möglichkeiten:

- Ordnen Sie den Knoten Testfälle zu.
- Ändern Sie die Testfalloptionen so, dass das System weitere Testfälle berücksichtigen kann.
- Binden Sie die zu testenden Transaktionen in den Testplan ein. Gehen Sie dazu folgendermaßen vor:
  1. Erweitern Sie die *Optionen für die Testplangenerierung*.
  2. Markieren Sie das Ankreuzfeld *Transaktionen ohne Testfälle einschließen*.
  3. Wählen Sie *Übernehmen und sichern*.

Sie können die Objekte auch an den Knoten ohne Testfälle testen. Sie werden vom System optimiert. Wenn Sie einen Testplan erzeugen, bindet das System die Transaktion in den Testplan mit ein. Sie können die Transaktion aus dem Testplan heraus starten.

## 7.7.4 Anpassen des Testumfangs in einer Grafik

### Verwendung

Alle Knoten werden in der Lösungsdokumentation in der Reihenfolge ihrer Testeffektivität grafisch dargestellt. Wenn Sie den Cursor über einen Knoten bewegen, werden folgende Werte angezeigt:

- Aufwand für automatische Tests in Minuten (Balkenanteil in grün)
- Aufwand für manuelle Tests in Minuten (Balkenanteil in orange)
- Testabdeckung (blauer Datenpunkt)

Die im Testumfang enthaltenen Knoten sind durch einen hellgrünen Hintergrund gekennzeichnet. Eine schwarze Linie am linken Rand kennzeichnet Knoten, die dem Testumfang manuell über Strukturattribute hinzugefügt wurden und die daher getestet werden müssen. Eine schwarze Linie am rechten Rand kennzeichnet Knoten, die manuell aus dem Testumfang entfernt wurden. Diese werden nicht getestet.

## Vorgehensweise

Um die grafische Darstellung aufzurufen, wählen Sie die Registerkarte *Testumfangsoptimierung: Rangfolge*.

Um den Testumfang zu vergrößern, wählen Sie einen Knoten aus, der noch nicht im Testumfang enthalten ist. Das System nimmt alle Knoten bis zum ausgewählten Knoten in den Testumfang auf. Das System passt die Angaben zu Abdeckung und Zeitaufwand automatisch an. Wenn am rechten Rand eine schwarze Linie dargestellt wird, können Sie Knoten nur bis zu dieser Markierung zum Testumfang hinzufügen.

Um den Testumfang zu verkleinern, wählen Sie den gewünschten letzten Knoten im Testumfang. Das System entfernt daraufhin alle folgenden Knoten aus dem Testumfang und passt die Angaben zu Abdeckung und Zeitaufwand automatisch an. Wenn am linken Rand eine schwarze Linie dargestellt wird, können Sie Knoten erst ab dieser Markierung entfernen.

### Anzeige der Aufwandswerte ändern

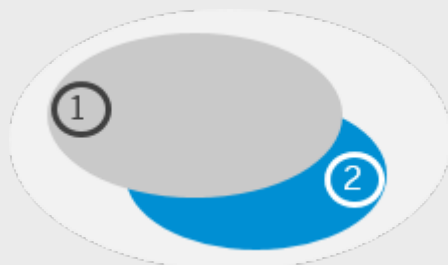
1. Wählen Sie unter *Aufwandswert* eine der folgenden Optionen:
  - *Kumuliert*: Die Werte für die Testaufwände und die Testabdeckung werden abgestimmt.
  - *Einfach*: Die Werte für die Testaufwände und die Testabdeckung jedes einzelnen Knotens werden als Absolutwerte angezeigt.

#### ❖ Beispiel

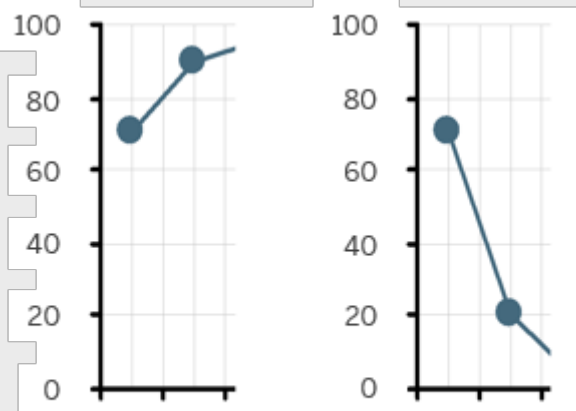
Die Art der Anzeige beeinflusst nicht die Reihenfolge der einzelnen Knoten, sondern nur die Darstellung der Aufwände und der Testabdeckung. Die grafische Sicht zeigt zwei Knoten in der Lösungsdokumentation, etwa mit folgenden Details:

	Knoten 1	Knoten 2
<b>Testabdeckung</b>	70%	20%
<b>Automatisierter Testaufwand (Minuten)</b>	30	20
<b>Manueller Testaufwand (Minuten)</b>	10	20

Die Testabdeckung ist die Differenz zum vorherigen Knoten. Wenn also zwei Knoten eine ähnlich große Anzahl von Objekten verwenden und viele Objekte gleichzeitig in beiden Knoten verwendet werden, liefert Knoten 1 eine deutlich größere Abdeckung (z. B. 70 %) als Knoten 2 (z. B. 20 %). In diesem Beispiel sind nur 20 % aller Objekte noch nicht durch andere Tests abgedeckt und werden in Knoten 2 verwendet.



Testabdeckung: Knoten 1 = 70 %, Knoten 2 = 20 %

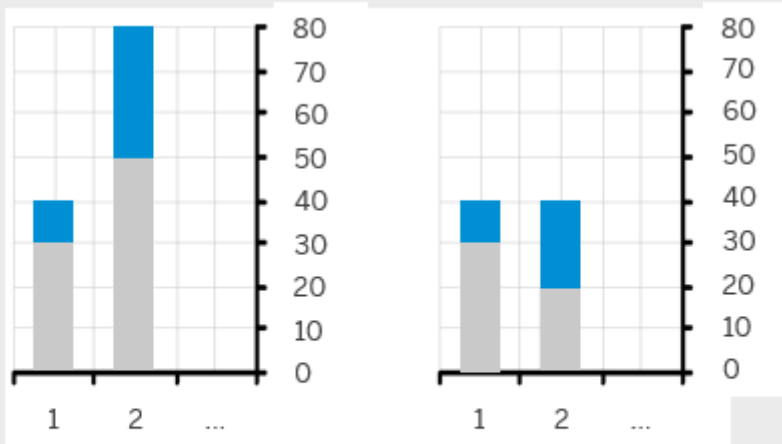


Kumuliert

Ledig

Anzeige der Testabdeckung als einfache oder kumulierte Werte

Die Aufwände für manuelle und automatische Tests ergeben sich aus den Testfalldetails, sollte es keine Testfalldetails geben, werden die Aufwände aus den Standardwerten abgeleitet.



Kumuliert

Ledig

Anzeige der automatisierten und manuellen Aufwände als einfache oder kumulierte Werte

## 7.7.5 Testumfang manuell übersteuern

## Kontext

Sie können Knoten der Lösungsdokumentation in den Testumfang aufnehmen, auch wenn diese eigentlich nicht getestet werden müssten. Sie können auch Knoten aus dem Testumfang entfernen.

## Vorgehensweise

1. Wählen Sie die Registerkarte *Lösungsdokumentation*.
2. Wählen Sie *Alles expandieren*.

Alle Zeilen mit Knoten, die aktuell im Testumfang enthalten sind, sind markiert.

3. Navigieren Sie zu der Zeile mit dem relevanten Knoten.
4. Um einen Knoten zum Testumfang hinzuzufügen: Drücken Sie `Strg`, und markieren Sie die bisher nicht markierte Zeile.
5. Um einen Knoten aus dem Testumfang zu entfernen: Drücken Sie `Strg`, und entmarkieren Sie die bisher markierte Zeile.
6. Wählen Sie *Übernehmen* und sichern Sie.

Am Status in der Spalte *Bereich eines Lösungsdokumentationsknotens* können Sie erkennen, ob ein Knoten manuell übersteuert wurde. Diese Spalte enthält außerdem auch Statusangaben für Knoten, die aufgrund von Lösungsdokumentationsoptionen oder Gewichtungsoptionen vom System in einen bestimmten Bereich verschoben wurden.

Wenn Sie die manuelle Übersteuerungsfunktion verwenden, ändern Sie die Optimierungsreihenfolge, sodass der Testumfang der vorgegebenen Werte für Testabdeckung und Zeitaufwand neu berechnet wird.

Die grafische Sicht wird automatisch aktualisiert. Alle manuell hinzugefügten Knoten sind am linken Rand des Diagramms durch eine schwarze Linie von den übrigen Knoten getrennt. Alle manuell entfernten Knoten sind am rechten Rand des Diagramms durch eine schwarze Linie von den übrigen Knoten getrennt.

7. Optional: Um alle manuellen Änderungen wieder zu entfernen, wählen Sie *Manuelle Markierung zurücksetzen*.

## 7.7.6 Prüfen von Objekten, die nicht im Testumfang enthalten sind

### Kontext

Im Anzeigebereich *Objekte nicht im Testumfang vorhanden* finden Sie eine Tabelle mit Objekten, die nicht im Testumfang enthalten sind. Sie können Objekte markieren und die zugehörigen Knoten der Lösungsdokumentation in den Testumfang aufnehmen.

## Vorgehensweise

1. Wählen Sie die Registerkarte *Lösungsdokumentation*.
2. Expandieren Sie den Anzeigebereich *Objekte nicht im Testumfang vorhanden*.  
Die tabellarische Übersicht zeigt die prozentuale Verteilung der nicht im Testumfang enthaltenen Objekte.
3. Navigieren Sie in der Tabelle zu der Hierarchieebene (logische Komponentengruppe, Softwarekomponente oder Paket), deren Objekte Sie prüfen möchten.
4. Wählen Sie den Hyperlink in der ausgewählten Hierarchieebene.  
In der detaillierten Analyse sind alle betroffenen Objekte aufgeführt.
5. Markieren Sie in der detaillierten Analyse die Objekte, die Sie dem Testumfang hinzuzufügen möchten.
6. Wählen Sie *In Testumfang einbeziehen*.

## Ergebnisse

Das System nimmt den Knoten der Lösungsdokumentation, der die ausgewählten Objekte mit der höchsten Testeffektivität enthält, in den Testumfang auf. Das System verhält sich so, als ob Sie diesen Knoten manuell übersteuert hätten:

- Unter *Lösungsdokumentation* wird die grafische Sicht automatisch aktualisiert.
- Unter *Lösungsdokumentation* ist der Knoten als *Manuell markiert* gekennzeichnet.

## Nächste Schritte

[Testumfang manuell übersteuern \[Seite 198\]](#)

[Ausschließen von Objekten von der Optimierung \[Seite 200\]](#)

## 7.7.7 Ausschließen von Objekten von der Optimierung

### Verwendung

Das System optimiert den Testumfang unter der Voraussetzung, dass ein Objekt vollständig getestet ist, sobald es mindestens einmal getestet wurde. D.h., auch wenn ein Objekt in mehreren Geschäftsprozessen verwendet wird, reicht das Testen eines Knotens in der Lösungsdokumentation für die Optimierungsstrategie „Jedes Objekt nur einmal testen“ aus. Wenn Sie bestimmte Objekte im jeweiligen Knotenkontext testen möchten, schließen Sie sie von der Optimierungsstrategie aus.



## Vorgehensweise

1. Wählen Sie in der Sicht *Verwaltung - Änderungseinfluss-Analyse* den Pfad ► *BPCA-Testumfangsoptimierung* ► *Befreiungen für Testumfangsoptimierung festlegen* ►.
2. Wählen Sie *Neue Einträge*.
3. Geben Sie die erforderlichen Daten ein, und sichern Sie Ihre Eingaben.  
Sie können den Platzhalter (\*) verwenden, um mehrere Objekte desselben Pakets, derselben Softwarekomponente oder desselben Objekttyps auszuschließen. Beispiel: Um alle Funktionsgruppen in der Softwarekomponente `SAP_BASIS` als Ausnahmen anzugeben, geben Sie Folgendes ein:
  - PG.ID = **R3TR**
  - Objekttyp = **FUGR**
  - Objektname = **\***
  - Softwarekomponente = **SAP\_BASIS**
  - Paket = [leer lassen]
4. Sichern Sie Ihre Eingaben.

## Ergebnis

Wenn Sie ein Objekt von der Optimierungsstrategie ausgeschlossen haben und dieses Objekt in anderen Knoten der Geschäftsprozesshierarchie verwendet wird, wird das Objekt in jedem Knoten separat behandelt. Um eine Testabdeckung von 100 % zu erreichen, müssen Sie alle Knoten, die dieses Objekt verwenden, dem Testumfang hinzufügen.

Wenn mehrere Knoten, die ein solches Objekt verwenden, noch nicht im Testumfang enthalten sind, so wird dieses Objekt unter *Objekte nicht im Testumfang vorhanden* mehrfach angezeigt. Wenn Sie das Objekt mindestens einmal markieren und *In Testumfang einbeziehen* wählen, fügt das System automatisch alle Knoten, die das Objekt verwenden, zum Testumfang hinzu.

## Weitere Informationen

[Prüfen des TBOM-Contents \[Seite 179\]](#)

[Analyseergebnisse auswerten und Testpläne erstellen \[Seite 176\]](#)

## 7.7.8 Mit Optimierungsansätzen arbeiten

### Verwendung

Sichern Sie alle Einstellungen, die Sie zur Optimierung des Testumfangs vornehmen. Diesen Optimierungsansatz können Sie wiederverwenden oder mit anderen Benutzern teilen.

Beim Durchführen einer Einflussanalyse können Sie einen Optimierungsansatz zuweisen. Die Einstellungen des Optimierungsansatzes werden in die Optimierungseinstellungen der Analyse übernommen. Daher wirken sich die Änderungen an einem Optimierungsansatz nicht auf vorhandene Analysen aus.

Der Optimierungsansatz enthält folgende Einstellungen:

- Optionen für die Testplangenerierung
- Optimierungsoptionen
- Gewünschte Testabdeckung
- Testumfangsauswahl über die grafische Sicht
- Manueller Einschluss oder Ausschluss von Knoten

## Vorgehensweise

### Optimierungsansatz anlegen

1. Führen Sie eine Änderungseinflussanalyse durch, und optimieren Sie den Testumfang.
2. Um die Optimierungseinstellungen zu ändern, wählen Sie [Testumfang optimieren](#).
3. Passen Sie die Einstellungen Ihren Anforderungen an.
4. Wählen Sie [Sichern](#), um die Einstellungen des Testumfangs zu speichern.
5. Wiederholen Sie die Schritte 2 und 3 so lange, bis Sie die gewünschten Optimierungsergebnisse erreicht haben.
6. Wählen Sie [Sichern als](#), um die Optimierungseinstellungen als wiederverwendbaren Optimierungsansatz zu speichern.  
Der Dialog [Neuer Optimierungsansatz](#) wird angezeigt.
7. Geben Sie einen technischen Namen und eine kurze Beschreibung für den neuen Optimierungsansatz an.
8. Optional: Markieren Sie das Ankreuzfeld [Globaler Optimierungsansatz](#), um anderen Benutzern das Verwenden Ihres Optimierungsansatzes zu erlauben.
9. Optional: Wählen Sie [Beschreibung](#), um den Optimierungsansatz genauer zu beschreiben.
10. Sichern Sie Ihre Eingaben.  
Die Einstellungen Ihrer Änderungseinfluss-Analyse werden in Ihrem Optimierungsansatz gespeichert.

### Optimierungsansatz importieren

1. Wählen Sie [Einstellungen importieren](#), um einen Optimierungsansatz in eine Änderungseinfluss-Analyse zu importieren.
2. Wählen Sie den Optimierungsansatz aus, und öffnen Sie ihn mit [Öffnen](#).
3. Sichern Sie Ihre Eingaben.  
Die Einstellungen Ihres Optimierungsansatzes werden auf Ihre Änderungseinfluss-Analyse angewendet.
4. Um die Beschreibung des Optimierungsansatzes anzuzeigen, wählen Sie [Beschreibung](#).

### Einen Optimierungsansatz bearbeiten, kopieren oder löschen

Wählen Sie [Einstellungen importieren](#), um einen Optimierungsansatz zu bearbeiten, zu kopieren oder zu löschen.

Alternativ können Sie auf den Optimierungsansatz über das Selektionsbild [Änderungseinfluss-Analyse](#) zugreifen und ihn bearbeiten. Wählen Sie unter [6. Optionale Parameter definieren](#) den Optimierungsansatz aus und klicken Sie auf [Bearbeiten](#).

## 7.7.9 Anlegen oder Erweitern von Testplänen

### Verwendung

Wie bei den Analyseergebnissen können Sie sogar während der Optimierung des Testumfangs jederzeit einen Testplan anlegen oder erweitern.

### Vorgehensweise

Sie haben folgende Möglichkeiten:

- Wählen Sie ► [Testplan](#) ► [Testplan anlegen](#) ►, um einen Testplan für alle Testfälle anzulegen, die sich aktuell im Testumfang befinden.
- Wählen Sie ► [Testplan](#) ► [Testplan erweitern](#) ►, um alle Testfälle, die sich aktuell im Testumfang befinden, einem vorhandenen Testplan hinzuzufügen.  
Das System ermöglicht das Erweitern aller Testpläne für den Branch, den Sie ausgewählt haben. Alle Testfälle aus SAP Solution Manager, die sich aktuell im Testumfang befinden, werden dem Testplan hinzugefügt.

## 7.8 Verwenden von Testplänen und Testfällen aus Partnerwerkzeugen

### Verwendung

Der Business Process Change Analyzer (BPCA) kann Testpläne und Testfälle aus SAP-Systemen und aus Partner-Testverwaltungswerkzeugen auswerten.

### Integration

Während der Änderungseinfluss-Analyse ermittelt der BPCA Testfallinformationen aus dem Partnerwerkzeug. Diese Informationen werden während der Testumfangsoptimierung an folgenden Stellen verwendet:

- Bereichsregeln für Testfälle definieren
- Testfallauswahl über Testfallattribute beeinflussen
- Testumfang durch Definition von Testaufwänden optimieren

## Voraussetzungen

- Sie nutzen ein Partner-Testverwaltungswerkzeug, das eine Integration in den BPCA unterstützt.
- In der *SAP-Solution-Manager-Konfiguration* haben Sie unter *Test-Suite* das Partner-Testmanagement integriert.
- Das Partnerwerkzeug ist im BPCA vorhanden.
- Die Auswertung von Testplänen und Testfällen ist nur so weit möglich, wie diese vom Partnerwerkzeug unterstützt werden.

## Funktionsumfang

Sie haben folgende Möglichkeiten, mit Partnerwerkzeugen zu arbeiten:

- Sie können bei der Änderungseinfluss-Analyse ein Partner-Testverwaltungswerkzeug angeben, in dem Sie Testfälle vorbereitet haben. Nach der Analyse können Sie aus den Testfällen Testpläne im Partnerwerkzeug erstellen oder erweitern.
- Sie können bei der Optimierung des Testumfangs festlegen, in welchem Umfang die Testfälle aus dem Partnerwerkzeug berücksichtigt werden sollen.
- Sie können die Testfälle aus dem Partnerwerkzeug abhängig von ihren Attributwerten für den *Zwangseinschlussbereich*, *Zwangsausschlussbereich* oder *Optimierungsbereich* berücksichtigen.
- Sie können aus den Testfällen im Partnerwerkzeug Testpläne erstellen oder erweitern.

## Weitere Informationen

[Durchführen der Änderungseinfluss-Analyse mit dem Partnerwerkzeug \[Seite 207\]](#)

[Anlegen oder Erweitern von Testplänen \[Seite 203\]](#)

## 7.8.1 Partner-Testwerkzeug registrieren

### Kontext

Um die von einem bestimmten Partnerwerkzeug für das Testmanagement unterstützten Testfalltypen, Testfallattribute und Attributwerte auswerten zu können, müssen Sie das Werkzeug für den Business Process Change Analyzer (BPCA) registrieren.

## Vorgehensweise

1. Wählen Sie in der *SAP-Solution-Manager-Konfiguration* den Pfad [▶ Test-Suite](#) [▶ SAP Quality Center/SAP TAO](#) [▶ Partner-Testmanagement](#) [▶](#). Gehen Sie zum Schritt [▶ BPCA integrieren](#) [▶ PTM-System in BPCA registrieren](#) [▶](#).
2. Fügen Sie einen neuen Eintrag hinzu, und geben Sie den Namen und eine Beschreibung des Partnerwerkzeugs für das Testmanagement ein.
3. Wählen Sie über die Wertehilfe den in der Konfiguration angelegten logischen Port zum Partner-Testmanagementwerkzeug aus.
4. Sichern Sie Ihre Eingaben.

Über die Drucktaste *Ping* können Sie die Verbindung zum Partner-Testmanagementwerkzeug überprüfen.

Wählen Sie *Erneut registrieren*, um alle während der Registrierung mit dem Partner-Testmanagementwerkzeug erfolgten Kommunikationsschritte zu wiederholen. Dies ist z.B. dann sinnvoll, wenn im Partner-Testmanagementwerkzeug neue Testfalltypen oder neue Testfallattribute hinzugefügt wurden.

## Ergebnisse

Der BPCA erkennt nun die vom Partnerwerkzeug gegenwärtig verwendeten Testfalltypen, Testfallattribute und Attributwerte.

## 7.8.2 Testfalltypen und -attribute für Partnerwerkzeuge

### Vorgehensweise

1. Rufen Sie die Anwendung *Verwaltung - Änderungseinfluss-Analyse* auf.
2. Wählen Sie [▶ BPCA-Integration](#) [▶ Partner-Testmanagementwerkzeug für BPCA registrieren](#) [▶](#).
3. Wählen Sie das relevante Partnerwerkzeug aus.
4. Wählen Sie die Sicht *Partner-Testmanagement-Testfalltypen*.

In dieser Sicht sehen Sie alle vom Partnerwerkzeug unterstützten Testfalltypen. Der Typ `3PTM_DEFAULT_TC_TYPE` wird automatisch während der Registrierung hinzugefügt. Er dient als Fallback-Typ, falls das Partnerwerkzeug während der Änderungseinfluss-Analyse nicht registrierte Testfalltypen meldet.

Für jeden Testfalltyp werden folgende Attribute angezeigt:

- Schlüssel des Typs
- Gruppe des Testfalltyps (manuell oder automatisiert)

- Beschreibung des Testfalltyps
  - Standard-Ausführungsaufwand in Minuten  
Diesen Standardaufwand verwendet der BPCA, wenn er während der Änderungseinfluss-Analyse zu einzelnen Testfällen keine Ausführungsaufwände ermitteln kann. Wenn vom Partnerwerkzeug kein Standard-Testaufwand gemeldet wird, können Sie hier einen Wert eingeben.
5. Wählen Sie die Sicht *Partner-Testmanagementattribute*.
- In dieser Sicht sehen Sie alle vom Partnerwerkzeug unterstützten Testfallattribute.
6. Wählen Sie die Sicht *Partner-Testmanagementattribute*.
- In dieser Sicht sehen Sie alle erlaubten Attributwerte zu einem ausgewählten Testfallattribut. Wenn diese Sicht für ein Attribut leer ist, gibt es keine Einschränkungen bezüglich der Attributwerte.

## 7.8.3 Zu einem Testplan oder Testfall im Partnerwerkzeug navigieren

### Voraussetzungen

Das Partner-Testverwaltungswerkzeug unterstützt die Navigation zu einem Testplan oder zu einem Testfall.

### Vorgehensweise

#### Zu einem Testplan im Partnerwerkzeug navigieren

1. Wählen Sie in der BCPA-Ergebnisliste den gewünschten Ergebnisdatensatz.
2. Wählen Sie in der oberen Detailtabelle (Ergebnis-ID ...) das Projekt oder die Lösung aus, das/die Sie auswerten wollen.
3. Wählen Sie **► Testplan ► Testplan erweitern ►**, um eine Liste aller Testpläne zu dem Projekt oder der Lösung zu erhalten.
4. Markieren Sie die Testplanbeschreibung.

Dies hat zum Ergebnis, dass der Testplan im Partner-Testverwaltungswerkzeug angezeigt wird.

#### Zu einem Testfall im Partnerwerkzeug navigieren

1. Wählen Sie in der BCPA-Ergebnisliste den gewünschten Ergebnisdatensatz.
2. Wählen Sie in der oberen Detailtabelle (Ergebnis-ID ...) das Projekt oder die Lösung aus, das/die Sie auswerten wollen.
3. Wählen Sie *Testumfang optimieren*.
4. Blenden Sie im Anzeigebereich die *Geschäftsprozesshierarchie* ein.
5. Navigieren Sie zu der Zeile mit dem relevanten Knoten.
6. Expandieren Sie den Knoten *Testfälle*.
7. Klicken Sie mit der rechten Maustaste auf den Testfall, zu dem Sie navigieren wollen, und wählen Sie *Springen Partnertestfall* aus dem Kontextmenü.

Dies hat zum Ergebnis, dass der Testfall im Partner-Testverwaltungswerkzeug angezeigt wird.

## Weitere Informationen

[Durchführen der Änderungseinfluss-Analyse mit dem Partnerwerkzeug \[Seite 207\]](#)

[Anlegen oder Erweitern von Testplänen \[Seite 203\]](#)

## 7.8.4 Durchführen der Änderungseinfluss-Analyse mit dem Partnerwerkzeug

### Voraussetzungen

Sie haben mindestens ein Partner-Testmanagementwerkzeug im Business Process Change Analyzer registriert.

### Kontext

Der Business Process Change Analyzer wertet bei der Änderungseinfluss-Analyse Informationen zu Testfällen aus. Diese Testfälle können entweder in einem SAP-System oder in einem Partner-Testverwaltungswerkzeug vorliegen.

### Vorgehensweise

1. Folgen Sie den allgemeinen Anweisungen für die Analyse der von Änderungen betroffenen Geschäftsprozesse.
2. Wählen Sie unter **6.** die Option *Details* aus.
3. Wählen Sie unter *Testfall-Speicherort* im Feld *Partnertestfälle* das relevante Partnerwerkzeug.

Die Liste enthält sämtliche im Business Process Change Analyzer registrierten Partnerwerkzeuge. Wenn Sie hier *Test Organizer* wählen, berücksichtigt das System nur Testfälle aus SAP-Systemen.

### Nächste Schritte

[Verwenden von Testplänen und Testfällen aus Partnerwerkzeugen \[Seite 203\]](#)

# 8 Umfangs- und Aufwandsanalyse

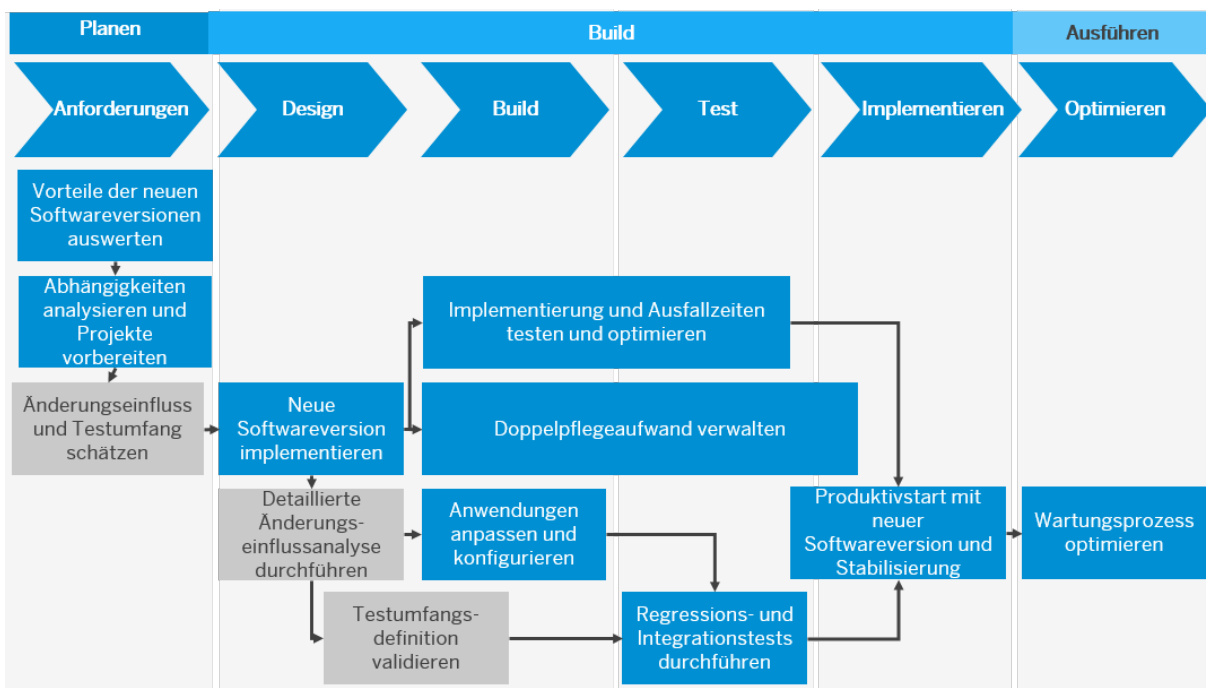
## Verwendung

Mit der *Umfangs- und Aufwandsanalyse* können Sie den Umfang und den Aufwand von Aktivitäten analysieren, bevor Sie mit der eigentlichen Implementierung von Enhancement Packages (EHP) und Support Packages (SP) beginnen.

Die *Umfangs- und Aufwandsanalyse* bietet Ihnen umfassende Analysemöglichkeiten bei geringstmöglichem Eingabeaufwand für den Kunden. Nachdem Sie die erforderlichen Eingaben vorgenommen haben, werden alle Analyseschritte der *Umfangs- und Aufwandsanalyse* im Hintergrund ausgeführt. Zu den wichtigsten und zeitaufwendigsten Analyseaktivitäten vor einer EHP- und SP-Implementierung gehören:

- Ermittlung der betroffenen kundeneigenen Entwicklung und Modifikationen sowie der erforderlichen Anpassungen im Kundensystem, da ein EHP oder SP immer eine Aktualisierung der SAP-Standardobjekte umfasst
- Testumfangsermittlung, Testplanung, Anlegen der fehlenden Testfälle und Ausführen von manuellen Tests

Anhand der Analyseergebnisse der *Umfang- und Aufwandsanalyse* können Sie die Änderungsauswirkungen auf kundeneigene Objekte und Modifikationen ermitteln und den Nachbearbeitungsaufwand für kundeneigene Objekte und Modifikationen sowie den Aufwand für Regressionstests für die betroffenen Geschäftsprozesse abschätzen.



Die Umfangs- und Aufwandsanalyse führt vor dem Projektstart eine Änderungseinflussanalyse durch, um den Aufwand abzuschätzen und den Testumfang zu ermitteln.



## Voraussetzungen

### Konfiguration in der SAP-Solution-Manager-Konfiguration

Bevor Sie mit der *Umfangs- und Aufwandsanalyse* arbeiten können, müssen Sie in der SAP-Solution-Manager-Konfiguration das Szenario *Umfangs- und Aufwandsanalyse* unter dem Szenario *Test-Suite* konfigurieren.

### Berechtigungen


Für die Verwendung der *Umfangs- und Aufwandsanalyse* benötigen Sie Berechtigungen im Berechtigungsobjekt `SM_SEA` für folgende Aktivitäten:

- Anwendung im Anzeigemodus starten
- Analyse anlegen
- Analyse ändern
- Analysen löschen

### Rollen

Um die Funktion nutzen zu können, verwenden Sie eine der beiden Sammelrollen, die für Endanwender der *Umfangs- und Aufwandsanalyse* relevant sind:

- `SAP_SEA_ALL_COMP`  
Sammelrolle „Umfangs- und Aufwandsanalyse“ (Volle Berechtigung)
- `SAP_SEA_DIS_COMP`  
Sammelrolle „Umfangs- und Aufwandsanalyse“ (Anzeigeberechtigung)

Weitere Informationen finden Sie im Security Guide des SAP Solution Manager auf dem SAP Support Portal unter [https://help.sap.com/viewer/p/SAP\\_Solution\\_Manager](https://help.sap.com/viewer/p/SAP_Solution_Manager) im Abschnitt **Security** > **Authorization** *Concept Security Guide* .

## Funktionsumfang

Auf dem Einstiegsbild der *Aufwands- und Umfangsanalyse* können Sie Folgendes tun:

- Vorhandene Analysen anzeigen
- Eine neue Analyse anlegen
- Analysen löschen
- Sicht einer vorhandenen Analyse exportieren
- Analysenverarbeitungsstatus anzeigen
- Analyseergebnisse anzeigen

Mit der Option „Durchschnittswerte“ können Sie die Standardwerte zur Berechnung des Entwicklungs- und Testaufwands ändern.

### Hinweis

In der *Umfangs- und Aufwandsanalyse* ist für jeden Schritt und für jedes Bild ein umfassender Hilfetext verfügbar.

## Aktivitäten

Sie können die *Umfangs- und Aufwandsanalyse* über das SAP Solution Manager Launchpad starten.

### 8.1 Anlegen einer Analyse

Zum Anlegen einer Analyse bietet die *Umfangs- und Aufwandsanalyse* eine geführte Aktivität mit den folgenden Schritten:

1. System auswählen und Wartungsplan zuordnen  
In diesem Schritt definieren Sie das System, das Sie auf eine neuere Softwareversion aktualisieren und für das Sie die Auswirkungsanalyse anlegen möchten.  
Nach Angabe des Systems für das geplante Upgrade geben Sie die aktuelle Produktversion und eine Transaktion für den *Maintenance Planner* ein. Mit diesem Wartungsvorgang wird die Ziel-Softwareversion festgelegt, auf die das ausgewählte System aktualisiert werden soll. Dabei berechnet die Anwendung, welches Software-Paket für das Update erforderlich ist, und ermittelt alle ABAP-Repository-Objekte, die beim Update geändert werden. Der Wartungsvorgang muss vor Anlage der *Umfangs- und Aufwandsanalyse* angelegt werden, und muss den Status *Abgeschlossen* aufweisen. Ein Wartungsvorgang kann in mehreren Instanzen der *Umfangs- und Aufwandsanalyse* verwendet werden.  
Außerdem legen Sie einen Namen für die Analyse fest.
2. Zusätzliche Systeme angeben  
In diesem Schritt geben Sie drei zusätzliche Systeme an, die von der Anwendung herangezogen werden, um die erforderlichen Informationen zu lesen und bestimmte Berechnungen durchzuführen. Lesen Sie für jede Systemrolle die ausführliche Beschreibung, um sich über den Zweck des jeweiligen Systems und die in der Regel zu verwendenden Systeme zu informieren.
3. Lösungsdokumentation angeben  
In diesem Schritt wählen Sie die Lösung und die Branche aus, in denen die Lösungsdokumentation enthalten ist. Bei Bedarf können Sie auch eine Site und den Umfang auswählen.
4. Testumfang angeben  
In diesem Schritt können Sie Ihre Testverwaltungsanwendung auswählen sowie den gewünschten Ansatz zur Testumfangsoptimierung, deren Berechnung im Hintergrund durch den Business Process Change Analyzer (BPCA) erfolgt.
5. Review  
In diesem Schritt wird zusammengefasst, was Sie in den vorigen Schritten eingegeben bzw. ausgewählt haben. Falls Sie daran Änderungen vornehmen möchten, kehren Sie zum entsprechenden Schritt zurück. Sind Ihre Eingaben richtig, können Sie die Analyse durchführen.

## 8.2 Analyseergebnisse

### Verwendung

Die *Umfangs- und Aufwandsanalyse* umfasst die folgenden Seiten, die Sie im Aktionsmenü auswählen können:

- Seite *Übersicht*
- Seite *Details – Modifikationen / Kundeneigene Entwicklungen*
- Seite *Details – Testmanagement*
- Seite *Einstellungen und Verarbeitungsprotokoll*

### Funktionsumfang

#### Seite „Übersicht“

Auf der Seite *Übersicht* finden Sie Übersichtsregisterkarten für die folgende Bereiche:

- Zusammenfassung  
Auf dieser Registerkarte erhalten Sie eine allgemeine Übersicht über die Ergebnisse der *Umfangs- und Aufwandsanalyse*. Die Übersicht zeigt die erwartete Entwicklung und den Testaufwand für die geplante Wartungsaktivität.

#### i Hinweis

Sonstiger Projektaufwand zur Implementierung einer neuen SAP-Softwareversion, wie Systemadministration, Aktualisierung von Betriebssystem- oder Datenbankversionen, Doppelpflegeaufwand oder andere Aktivitäten, ist in dieser Analyse nicht enthalten. Der in der Zusammenfassung angezeigte Aufwand ist also nicht der für die geplante Wartungsaktivität erforderliche Gesamtaufwand.

- Modifikationen / Kundeneigene Entwicklungen  
Auf dieser Registerkarte finden Sie eine Übersicht der Auswirkungen der geplanten Wartungsaktivität auf Ihre kundeneigenen Objekte und Modifikationen.  
**Modifikationen**  
In diesem Block finden Sie Ihre Modifikationen, geordnet nach Auswirkungskategorien. Eine Beschreibung der Auswirkungskategorien und der entsprechenden empfohlenen Aktionen finden Sie auf der Seite *Details - Modifikationen / Kundeneigene Entwicklungen* in der Hilfe zur Registerkarte *Modifikationen*.  
**Kundeneigene Entwicklungen**  
In diesem Block finden Sie Ihre kundeneigenen Entwicklungen, geordnet nach Auswirkungskategorien. Eine Beschreibung der Auswirkungskategorien und der entsprechenden empfohlenen Aktionen finden Sie auf der Seite *Details – Modifikationen / Kundeneigene Entwicklungen* in der Hilfe zur Registerkarte *Kundeneigene Entwicklungen*.
- Testmanagement  
Auf dieser Registerkarte finden Sie eine Übersicht des Testumfangs und des Aufwands für die geplante Wartungsaktivität. Mit dem Auswahlknopf oben auf dem Bild können Sie Testumfang und Aufwand für alle betroffenen Prozesse anzeigen oder die Anzeige auf ausführbare Einheiten beschränken, die kundeneigene Objekte und Modifikationen enthalten.

- Aktualisierte SAP-Objekte  
Auf dieser Registerkarte finden Sie eine Übersicht der SAP-Repository-Objekte, die mit der Erweiterung bzw. den Support Packages, die Sie implementieren möchten, ausgeliefert werden.

## Seite „Details – Modifikationen / Kundeneigene Entwicklungen“

### Modifikationen

Auf dieser Registerkarte werden Details zu den Auswirkungen der geplanten Wartungsaktivität auf Ihre Modifikationen angezeigt. Die komplette Liste der Modifikationen können Sie im Modification Browser anzeigen (Transaktion SE95). In dieser Sicht sind nur die betroffenen Modifikationen zu sehen.

### Kundeneigene Entwicklungen

Auf dieser Registerkarte werden Details zu den Auswirkungen der geplanten Wartungsaktivität auf Ihre kundeneigenen Entwicklungen angezeigt.

Zusätzlich zur Dokumentation in der Anwendung finden Sie weitere Informationen unter [Weitere Empfehlungen für kundeneigene Entwicklungen \[Seite 213\]](#).

## Seite „Details – Testmanagement“

### Ergebnisübersicht

Auf dieser Registerkarte finden Sie eine Übersicht des Testumfangs und des Aufwands für die geplante Wartungsaktivität. Mit dem Auswahlknopf oben auf dem Bild können Sie Testumfang und Aufwand für alle betroffenen Prozesse anzeigen oder die Anzeige auf ausführbare Einheiten beschränken, die kundeneigene Objekte und Modifikationen enthalten.

### Testumfangsoptimierung: Rangfolge

Auf dieser Registerkarte finden Sie mehrere Optionen zum Steuern der Testumfangsoptimierung.

### Lösungsdokumentation

Auf dieser Registerkarte wird die gesamte Lösungsdokumentation hierarchisch dargestellt. Betroffene Geschäftsprozesse, Prozessschritte und ausführbare Einheiten werden hervorgehoben.

### Testfallempfehlung

Auf dieser Registerkarte kann die Anwendung auf Grundlage der verfügbaren Informationen Empfehlungen zum Anlegen fehlender Testfälle und Ersetzen manueller durch automatisierte Testfälle abgeben.

## Seite „Einstellungen und Verarbeitungsprotokoll“

### Durchschnittswerte für Aufwände

Auf dieser Registerkarte sehen Sie die Parameter, mit denen bei der [Umfangs- und Aufwandsanalyse](#) Entwicklungs- und Testaufwand berechnet werden. Außerdem können Sie die Durchschnittswerte zur Berechnung des Anpassungsaufwands ändern. Alle Änderungen gelten jeweils nur für die aktuelle Analyse. Der Anpassungsaufwand wird nach dem Sichern der Änderungen neu berechnet. Die Durchschnittswerte für den Testaufwand können nicht bearbeitet werden, da sonst die gesamte Analyse neu gestartet werden müsste. Um die Durchschnittswerte für den Testaufwand zu ändern, müssen Sie die Standardwerte auf dem Einstiegsbild der [Umfangs- und Aufwandsanalyse](#) anpassen, bevor Sie eine Analyse anlegen.

## 8.2.1 Weitere Empfehlungen für kundeneigene Entwicklungen

Auf der Registerkarte *kundeneigene Entwicklungen* werden Details zu den Auswirkungen der geplanten Wartungsaktivität auf Ihre kundeneigenen Entwicklungen angezeigt. Zusätzlich zur Dokumentation in der Anwendung erhalten Sie in diesem Abschnitt weitere Empfehlungen für kundeneigene Entwicklungen nach Auswirkungskategorie.

Es liegen weitere umfassende Empfehlungen für die folgenden Auswirkungskategorien vor.

### Kundeneigene Objekte mit gelöschten SAP-ABAP-Objekten

#### Empfehlung: Anpassung erforderlich

Für gelöschte SAP-DDIC-Objekte, auf die Sie in Ihren Programmen verweisen, müssen Sie entsprechende Objekte in Ihrem Kundennamensraum anlegen. Für SAP-Programme, -Funktionen oder -Methoden müssen Sie nach anderen SAP-Programmen suchen, die die gelöschten Programme ersetzen und die für die Kundennutzung freigegeben sind. Alternativ legen Sie die erforderlichen Funktionen in Ihrem Kundennamensraum an. Dabei dürfen Sie das gelöschte SAP-Objekt **nicht** in Ihren Namensraum kopieren (oder dort einen Clone anlegen). Dies kann zu Folgeproblemen führen wie in den nachfolgenden Kategorien für geklonte Programme näher beschrieben.

### Clones mit geänderten SAP-Referenzen und -Originalen

Diese Auswirkungskategorie zeigt Clones an, wenn die folgenden Bedingungen erfüllt sind:

- Der Clone umfasst SAP-Repository-Objekte (oder ruft diese auf), die in der neuen SAP-Software-Version einer Änderung unterliegen
- Die Original-SAP-Objekte, für die der Clone angelegt wurde, werden in der neuen SAP-Software-Version geändert, oder der Clone umfasst weitere geklonte SAP-Objekte, für die sich das Original-SAP-Objekt nach dem Upgrade ändert

#### Empfehlung: Anpassung erforderlich

Es wird empfohlen, Clones in dieser Auswirkungskategorie sorgfältig in einem Upgrade-Testsystem zu untersuchen und zu testen. Mit dem Werkzeug Clone Finder können Sie diese Analyse in Ihrem Entwicklungssystem ausführen. Den Clone Finder rufen Sie über die Transaktion `CC_APPS` auf.

### Kundeneigene Objekte mit geänderten obsoleten SAP-Funktionen

#### Empfehlung: Anpassung erforderlich

SAP kann nicht für die ordnungsgemäße Funktionsweise von veralteten Funktionen in aktualisierten Softwareversionen garantieren. Sie müssen diese Funktionen durch die entsprechenden freigegebenen Nachfolger ersetzen.

### **i Hinweis**

Die Umfangs- und Aufwandsanalyse kann keine Funktionen ermitteln, die mit Einführung des neuen Release veraltet sind. Deshalb müssen Sie die Referenzanalyse verwenden, die Sie in Transaktion `CC_APPS` starten können, um nach weiteren veralteten Funktionen in einem Upgrade-Testsystem zu suchen.

## **Kundeneigene Objekte mit geänderten nicht veröffentlichten SAP-Funktionen**

### **Empfehlung: Anpassung erforderlich**

SAP kann nicht für die ordnungsgemäße Funktionsweise von nicht freigegebenen Funktionen in aktualisierten Softwareversionen garantieren. Sie müssen diese Funktionen durch die entsprechenden freigegebenen SAP-Funktionen ersetzen. Zumindest müssen Sie die geänderten SAP-Funktionen eingehend prüfen und testen.

## **Kundeneigene Objekte mit Syntaxfehlern nach Upgrade**

### **Empfehlung: Anpassung erforderlich**

Führen Sie eine erweiterte Syntaxprüfung für alle kundeneigenen Objekte in einem aktualisierten Testsystem aus, um die betroffenen kundeneigenen Objekte zu ermitteln und einen Arbeitsvorrat für Ihre Entwickler anzulegen. Sie können die Syntaxprüfung als Massenverarbeitungsjob mit dem ABAP Test Cockpit oder mit dem Code Inspector ausführen.

## **Kundeneigene Objekte mit geänderten SAP-Referenzen**

### **Empfehlung: Anpassung erforderlich**

Die meisten kundeneigenen Objekte, die von einem Enhancement-Package- oder Support-Package-Upgrade betroffen sind, befinden sich in dieser Kategorie. In der Regel sind die meisten ausgelieferten SAP-Änderungen nicht kritisch und erfordern keinen zusätzlichen Entwicklungsaufwand. Jedoch kann mit der Umfangs- und Aufwandsanalyse keine detaillierte Auswirkungsanalyse auf Codeebene durchgeführt werden, da der entsprechende neue Quellcode zum Zeitpunkt der Analyse nicht verfügbar ist.

Führen Sie eine detaillierte Auswirkungsanalyse für Ihre kundeneigenen Objekte durch, sobald ein aktualisiertes Testsystem zur Verfügung steht, um kritische Änderungen zu ermitteln und einen Arbeitsvorrat für Ihre Entwickler anzulegen. Sie können diese Analyse mit dem Custom Development Management Cockpit durchführen.

Zusätzlich müssen Sie Regressionstests für alle Geschäftsprozesse durchführen, die kundeneigene Objekte mit geänderten Referenzen verwenden. Weitere Informationen finden Sie im Abschnitt zur Testumfangsoptimierung für diese Umfangs- und Aufwandsanalyse.

## **Kundeneigene Objekte mit obsoleten SAP-Funktionen**

### **Empfehlung: Anpassung erforderlich**

Sie müssen im Verlauf des geplanten Wartungsprojekts veraltete Funktionen durch die entsprechenden freigegebenen SAP-Funktionen ersetzen. Auf diese Weise werden zukünftige Konflikte vermieden.

## **Clones mit geändertem SAP-Original**

### **Empfehlung: Anpassung erforderlich**

Wenn sich keine anderen Teile des geklonten Programms ändern, sind in der Regel keine Auswirkungen auf die Funktionsweise des Clone zu erwarten. Sie müssen also keine sofortige Entwicklungsmaßnahme ergreifen. Sie müssen jedoch einen Regressionstest für die Geschäftsprozesse mit diesen Clones durchführen, um zu prüfen, ob die entsprechenden Clones weiterhin wie gewünscht funktionieren. Weitere Informationen finden Sie im Abschnitt zur Testumfangoptimierung für diese Umfangs- und Aufwandsanalyse.

# 9 Test-Repository: Testskripte und Testkonfigurationen

## Verwendung

Die Anwendungen *Test-Repository - Testkonfigurationen* und *Test-Repository - Testskripte* ermöglichen Ihnen den Zugriff auf die Umgebung für die Testzusammenstellung im SAP Solution Manager. Hier können Sie Testskripte und Testkonfigurationen für automatisierte Testfälle anlegen und bearbeiten:

- Eine **Testkonfiguration** enthält einen Systemdatencontainer, einen Testdatencontainer und ein Testskript. Die Daten aus diesen Elementen können in mehreren Testkonfigurationen wiederverwendet werden.
- Ein **Testskript** beschreibt technisch detailliert die Komponenten und die Reihenfolge der zu testenden Aktivitäten für einen ausgewählten Geschäftsprozess. Das Testskript kann einen oder mehrere Schritte enthalten, die Sie in die richtige Reihenfolge bringen können.  
Sie können Testskripte mit im System konfigurierten Testwerkzeugen automatisch anlegen, und zwar unabhängig davon, ob es sich dabei um eCATT, die komponentenbasierte Testautomatisierung (CBTA) oder Drittanbieter-Testwerkzeuge handelt.  
Sie können vorhandene Testskripte wiederverwenden, modulare Testskripte anlegen und durchgängige Prozesse testen. Weitere Informationen finden Sie unter [Anlegen eines Testskripts \[Seite 222\]](#).
- Ein **Testfall** ist die Zuordnung einer Testkonfiguration zu einem Strukturknoten (z.B. einem Geschäftsprozessschritt) in Ihrer Lösungsdokumentation. Weitere Informationen finden Sie unter [Lösungsdokumentation: Anlegen von Testfällen \[Seite 232\]](#)  
Um den korrekten Kontext für SAP Solution Manager zu erhalten, wird empfohlen, die Testkonfiguration aus der Lösungsdokumentation anzulegen.

## Voraussetzungen

- Um ein Drittanbieter-Testwerkzeug zu verwenden, haben Sie das Testwerkzeug auf dem Client-Rechner installiert, auf dem das SAP GUI für das SAP-Solution-Manager-System und für die Systeme unter Test installiert ist, und Sie haben das Testwerkzeug im SAP Solution Manager registriert.
- Sie haben Ihre Lösungsdokumentation definiert und dort die relevanten ausführbaren Einheiten angelegt. Die Auswahl einer ausführbaren Einheit beim Anlegen eines automatisierten Tests ist ein obligatorischer Schritt, da auf diese Weise das System unter Test abgeleitet wird.
- Optional: Sie haben einen Testdatencontainer angelegt. Weitere Informationen finden Sie unter [Testdatencontainer im Testautomatisierungs-Framework anlegen \[Seite 23\]](#).



## Aktivitäten

Führen Sie folgende Aktivitäten aus:

1. Legen Sie eine Testkonfiguration an, die einen Testdatencontainer und einen Systemdatencontainer enthält, oder bearbeiten Sie eine solche Testkonfiguration. Weitere Informationen finden Sie unter [Anlegen von Testkonfigurationen und Testskripten \[Seite 217\]](#).  
Ein leeres Testskript, das noch keine Testskriptschritte enthält, wird automatisch angelegt. Ein leeres Testskript (ohne Testkonfiguration) legen Sie z. B. dann an, wenn Sie schnell einen Test erstellen möchten.
2. Um z.B. die Anwendungskomponente oder die Verantwortlichen anzugeben, pflegen Sie die SAP-Attribute der Testkonfiguration oder des Testskripts. Weitere Informationen finden Sie unter [Pflegen von Attributen \[Seite 220\]](#).
3. Um das Testskript zu bearbeiten, zeichnen Sie ein Testskript auf oder definieren die Testskriptschritte manuell. Weitere Informationen finden Sie unter [Anlegen eines Testskripts \[Seite 222\]](#).  
Sie können eine durchgängige Testkonfiguration von Schritten erstellen, die Sie in verschiedenen Testwerkzeugen angelegt haben: eCATT, komponentenbasierte Testautomatisierung, Standardregressionstests oder Drittanbieter-Testwerkzeuge.
4. Führen Sie das Testskript manuell aus, und beheben Sie vorhandene Fehler. Weitere Informationen finden Sie unter [Testkonfigurationen und Testskripte ausführen \[Seite 230\]](#).  
Um die Ausführung eines Tests im Detail zu überwachen, können Sie einen Ausführungsreport anzeigen. Weitere Informationen finden Sie unter [Ausführungs-Reports anzeigen \[Seite 64\]](#).
5. Verwenden Sie die Lösungsdokumentation in der Anwendung *Test-Suite - Testvorbereitung*, um einen Testfall für einen Geschäftsprozessschritt anzulegen. Während Sie den Testfall anlegen, ordnen Sie ihm eine Testkonfiguration zu oder legen eine neue an. Weitere Informationen finden Sie unter [Lösungsdokumentation](#).
6. Um den Testfall zur Ausführung einzuplanen, ordnen Sie ihn einem Testpaket in einem Testplan zu. Weitere Informationen finden Sie unter [Testplanverwaltung \[Seite 235\]](#).

## 9.1 Anlegen von Testkonfigurationen und Testskripten

### Verwendung

In der Anwendung *Test-Repository* können Sie Testkonfigurationen und Testskripte anlegen und bearbeiten. Ein Testskript ist Bestandteil jeder Testkonfiguration und beschreibt die technischen Komponenten, die zur Ausführung des Geschäftsprozessschritts erforderlich sind. Weitere Informationen finden Sie unter [Komponenten automatisierter Testfälle \[Seite 18\]](#).

Sie ordnen einer Lösung, die mit einem Systemdatencontainer verknüpft ist, eine Testkonfiguration und ein Testskript zu. Sie können einen Testdatencontainer einer Testkonfiguration zuordnen.

Sie legen Sie eine Testkonfiguration an, um einen automatisierten Testfall anzulegen. Da ohne ein Testskript keine Testkonfiguration möglich ist, müssen Sie parallel das entsprechende Testskript anlegen. Zu einem späteren Zeitpunkt können Sie ein zugeordnetes Testskript bearbeiten oder durch ein anderes Testskript ersetzen.

Sie können das Testskript automatisch anlegen, indem Sie beispielsweise mit dem CBTA-Werkzeug eine Abfolge von Schritten mit Bezug zu einer ausführbaren Einheit (z.B. einem Vorgang oder einer Web-Dynpro-Anwendung) aufzeichnen. Sie können auch eCATT- oder Drittanbieter-Testskripte verwenden.

## Voraussetzungen

Weitere Informationen erhalten Sie in den Voraussetzungen unter [Test-Repository: Testskripte und Testkonfigurationen \[Seite 216\]](#).

### ⚠ Achtung

Bevor Sie eine bestehende Testkonfiguration bearbeiten, prüfen Sie, in welchen Testplänen, Testpaketen oder Testskripten sie verwendet wird. Markieren Sie hierzu die Testkonfiguration, und wählen Sie [Verwendungsnachweis](#).

## Vorgehensweise

### Anlegen von Testkonfigurationen in der Lösungsdokumentation (empfohlen)

Wir empfehlen aus folgenden Gründen, eine Testkonfiguration aus der Lösungsdokumentation in der Anwendung [Test-Suite - Testvorbereitung](#) heraus anzulegen:

- In der Anwendung [Test-Suite - Testvorbereitung](#) können Sie zunächst einen bestimmten Prozessschritt in der richtigen Lösung und im richtigen Branch auswählen und den Testfall dann direkt mit der zugehörigen Testkonfiguration anlegen. Somit ist die richtige Zuordnung von Anfang an vorhanden.
- Sie können das ausführbare Objekt für den Prozessschritt in der Lösungsdokumentation auswählen oder die zugeordnete ausführbare Einheit direkt abrufen.
- Sie können einem Prozessschritt mehrere Testkonfigurationen zuordnen und Ihre Tests entsprechend organisieren.
- Sie können den Business Process Change Analyzer (BPCA) für die Lösung verwenden.

Nach der Erstellung können Sie Testkonfigurationen auch in der Anwendung [Test-Repository - Testkonfigurationen](#) bearbeiten.

### Legen Sie folgendermaßen eine Testkonfiguration an:

Öffnen Sie in der Anwendung [Test-Suite - Testvorbereitung](#) eine Lösung und einen Branch.

Führen Sie im [Browser](#) (kann oben im neuen Fenster ausgewählt werden) einen Drilldown auf den Geschäftsprozessschritt aus, für den Sie einen Test anlegen möchten.

In der Tabelle der Elemente sehen Sie die ausführbaren Einheiten und Testfälle, die bereits zugeordnet wurden.

Um dem Geschäftsprozessschritt eine neue ausführbare Einheit zuzuordnen, klicken Sie mit der rechten Maustaste in diese Tabelle, und wählen Sie **► Neu ► Ausführbare Einheiten ►**. Solch eine ausführbare Einheit kann eine Transaktion, ein Programm, eine SAP-GUI-Transaktion, eine CRM-Anwendung (unter Verwendung der Rolle und der logischen Verknüpfungparameter), eine Web-Dynpro-ABAP-Anwendung oder eine URL sein.

Aus diesem Grund ist die ausführbare Einheit verfügbar, um in den nächsten Schritten einem Testfall zugeordnet zu werden.

Um einen neuen Testfall anzulegen und dem Geschäftsprozessschritt zuzuordnen, klicken Sie mit der rechten Maustaste in die Tabelle mit Elementen, und wählen Sie ► [Neu](#) ► [Testfälle](#) ► [Testkonfiguration \(Anlegen\)](#) ⌵:

1. Der Editor für die Testkonfiguration wird geöffnet.  
Füllen Sie die Pflichtfelder aus, und sichern Sie die Angaben.
2. Nehmen Sie die Einstellungen auf der Registerkarte [Attribute](#) vor.  
Weitere Informationen finden Sie unter [Pflegen von Attributen \[Seite 220\]](#).
3. Ordnen Sie die zu testende ausführbare Einheit zu, da das System für die Testausführung aus der ausführbaren Einheit abgerufen wird.
4. Wenn Sie die CBTA (komponentenbasierte Testautomatisierung) als Testwerkzeug verwenden, wird die Zuordnung der ausführbaren Einheit auch während der Aufzeichnung Ihres Tests verwendet. Nach Auswahl der ausführbaren Einheit können Sie [CBTA starten](#) wählen, um die Erstellung des Testskripts direkt zu starten, indem Sie die ausgewählte ausführbare Einheit in CBTA ausführen.
5. Wenn die Testkonfiguration abgeschlossen ist, wählen Sie [Sichern](#) und [Schließen](#).

Wählen Sie im Editor für die Geschäftsprozesse den neu angelegten Testfall aus.

Auf der rechten Seite können Sie weitere Merkmale definieren:

- [Testklassifizierung](#): Sie können den Testfall beispielsweise als Regressionstest oder Benutzertest definieren. Dies ist nützlich, wenn Sie später Testpläne für diesen spezifischen Zweck anlegen.
- [Testdatenvariante](#): Sie können diese für automatische Testfälle auswählen. Die Testdaten enthalten eine Reihe von Werten, die für den automatischen Test verwendet werden sollen. Die Variante definiert, welche Werte verwendet werden sollen.
- [Gruppe log. Zielkomponenten](#): Wenn Sie eine Testkonfiguration anlegen, müssen Sie die logische Komponentengruppe definieren. Zusammen mit dem Branch und der Systemrolle ermöglicht dies, dass der Test im richtigen Zielsystem ausgeführt werden kann. Die hier angegebene logische Komponentengruppe entspricht der Gruppe, die für die Testkonfiguration ausgewählt wird.

Sichern Sie Ihre Angaben, und schließen Sie das Fenster.

Sie können Testkonfigurationen in der Anwendung [Test-Repository - Testkonfigurationen](#) oder in der Lösungsdokumentation in der Anwendung [Test-Suite - Testvorbereitung](#) bearbeiten.

### **Anlegen von Testskripten im Test-Repository (für einfache Tests)**

Wir empfehlen, Testkonfigurationen einschließlich Testskript in der Anwendung [Test-Suite - Testvorbereitung](#) anzulegen. Um jedoch schnell einen sehr einfachen Test anzulegen, können Sie das leere Testskript direkt in der Anwendung [Test-Repository - Testskripte](#) oder der Anwendung [Test-Repository - Testkonfigurationen](#) anlegen:

1. Starten Sie die Anwendung [Test-Repository - Testskripte](#) oder die Anwendung [Test-Repository - Testkonfigurationen](#). Wenn Sie eine Testkonfiguration anlegen oder bearbeiten, wird das Testskript automatisch eingeschlossen.
2. Wählen Sie [Anlegen](#).
3. Wählen Sie eine Lösung aus.
4. Sie können die Systemrolle direkt im Test-Repository definieren. Zusammen mit der Lösung und dem Branch ermöglicht sie die automatische Ableitung des physischen Systems, auf das die logische Komponentengruppe verweist. Durch die Änderung der Systemrolle wird das System, auf dem die Tests aufgezeichnet und ausgeführt werden, geändert. Dies ist nützlich, wenn Sie eine Reihe von Landschaften definieren, die mehrere Systeme wie etwa Entwicklungs-, Test-, Vorproduktiv- und Produktivsystem enthalten.

5. Wenn Sie eine Testkonfiguration anlegen, gehen Sie folgendermaßen vor: Geben Sie im Feld *Testkonfiguration* den Namen der Testkonfiguration ein.

#### **i** Hinweis

Wenn Sie eine lokale Testkonfiguration (die nicht in andere Systeme transportiert werden soll) anlegen möchten, geben Sie einen technischen Namen ein, der mit **z** beginnt (abhängig von dem in der Systemkonfiguration festgelegten Namensraum). Wenn Sie zum Beispiel alle Testfälle auf einem einzigen SAP-Solution-Manager-System ablegen, wird der Namensraum nach dem Aufzeichnen einer Bildkomponente automatisch zugeordnet. Weitere Informationen zum Konfigurieren des Namensraums finden Sie in SAP-Hinweis [1763697](#).

6. Wählen Sie im Feld *Testwerkzeug* das gewünschte Testwerkzeug aus.
7. Wenn Sie eine Testkonfiguration anlegen, geben Sie den Namen für das Testskript ein, das angelegt werden soll, oder ordnen Sie ein bereits vorhandenes Testskript zu.

#### **i** Hinweis

Wenn Sie kein vorhandenes Testskript zuordnen, wird beim Anlegen einer Testkonfiguration automatisch ein Testskript angelegt. Der Name des Testskripts entspricht dem Namen der Testkonfiguration.

8. Optional: Ändern Sie die Standardversionsnummer.
9. Geben Sie im Feld *Titel* einen Namen ein.
10. Wählen Sie im Feld *Paket* ein Transportpaket aus, oder wählen Sie *Lokales Objekt*.
11. Bestätigen Sie Ihre Eingaben.

## 9.1.1 Definieren von Testkonfigurationsattributen

### Voraussetzungen

- Sie verfügen über die Berechtigung eines Entwicklungsberaters.
- Wenn Sie das Werkzeug für die komponentenbasierte Testautomatisierung verwenden, haben Sie ein Testprofil angelegt. Weitere Informationen finden Sie unter [Konfigurieren eines SUT mit Testprofil \[Seite 47\]](#).

### Kontext

Sie können eine wie unter [Anlegen von Testkonfigurationen und Testskripten \[Seite 217\]](#) beschrieben angelegte Testkonfiguration bearbeiten.

## Vorgehensweise

1. Klicken Sie in der Anwendung *Test-Repository - Testkonfiguration* auf eine Testkonfiguration, um diese zu bearbeiten.
2. Auf der Hauptregisterkarte *Attribute* werden auf der Unterregisterkarte *Testkonfigurationsattribute* die wichtigsten Attribute angezeigt.

**(Hinweis:** Wenn Sie ein Testskript mithilfe eines Drittanbieter-Testwerkzeugs angelegt haben, ist auch die untergeordnete Registerkarte *Testwerkzeugattribute* vorhanden. Sie ist schreibgeschützt und die Attribute werden bei der Aufzeichnung ausgefüllt.)

Geben Sie zuerst die Testkonfigurationsattribute ein, und definieren Sie danach die Testskriptattribute.

3. Ausführbare Einheit zuordnen: Ordnen Sie die ausführbare Einheit zu, die getestet werden soll. Weitere Informationen finden Sie unter [Typen von ausführbaren Einheiten \[Seite 221\]](#).

Wenn Sie die ausführbare Einheit aus der zugehörigen Lösung zuordnen, ordnet das System automatisch die logische Komponentengruppe zu.

4. Die Anwendungskomponente wird aus der ausgewählten ausführbaren Einheit abgeleitet. Ist die Anwendungskomponente nicht verfügbar, kann sie manuell eingegeben werden. In der Regel ist es die für die getesteten Anwendungen.
5. Geben Sie folgende Daten zur Beschreibung des Testskripts bzw. der Testkonfiguration ein:
  - *Verantwortlicher*: Standardmäßig ist der aktuelle Benutzer vordefiniert. Diesen können Sie ändern.
  - Ordnen Sie ein Testprofil zu, das in der Verwaltung der Systeme unter Test definiert wurde. Weitere Informationen finden Sie unter [Konfigurieren eines SUT mit Testprofil \[Seite 47\]](#).
  - Um die Anmeldung beim Zielsystem zu testen, klicken Sie auf die Verknüpfung mit dem Zielsystem.
  - *Status*: Der Status ist standardmäßig auf `aktiv` gesetzt. Nur aktive Skripte werden in automatischen Tests berücksichtigt. Wenn Sie nicht möchten, dass das Testskript ausgeführt wird (z.B. weil es noch nicht abgeschlossen ist oder von Ihnen derzeit bearbeitet wird), setzen Sie den Status auf „Inaktiv“.
  - *Zeitaufwand*: Geben Sie die für die Testkonfiguration voraussichtlich erforderliche Zeit ein.
  - *Priorität*

### 9.1.1.1 Ausführbare Einheiten zu Testskripten zuordnen

#### Verwendung

Ausführbare Einheiten werden in der [Lösungsdokumentation](#) angelegt. Wenn Sie also einer Lösung einen Testfall zuordnen, können Sie die ausführbare Einheit der Lösung dem Testfall zuordnen.

Diese Zuordnung ist obligatorisch, da sie die Bestimmung des Zielsystems ermöglicht. Das System stellt eine Suchhilfe zum Abrufen der ausführbaren Einheiten bereit, die im Rahmen Ihres Lösungsdokumentationskontexts verfügbar sind.

## Aktivitäten

1. Wählen Sie in der Sicht *Test-Repository - Testskripte* ein Testskript zur Bearbeitung aus.
2. Ordnen Sie auf der Registerkarte *Attribute* eine ausführbare Einheit zu.  
Definieren Sie logische Komponentengruppe, Art der ausführbaren Einheit und ausführbare Einheit, und wählen Sie *Suchen*.  
Die Suchergebnisse werden aus dem zugehörigen Branch der Lösung abgerufen.
3. Klicken Sie auf ein Suchergebnis, um es dem Testskript zuzuordnen.
4. Sichern Sie Ihre Eingaben und schließen Sie das Fenster.

## 9.1.2 Anlegen eines Testskripts

### Verwendung

Das Testskript ist der zentrale Bestandteil einer Testkonfiguration. Hier werden die Komponenten des Geschäftsprozesses, die getestet werden sollen, und deren Reihenfolge im Test im Detail beschrieben. In den *Test-Repository*-Sichten können Sie Testkonfigurationen und die darin enthaltenen Testskripte anlegen und bearbeiten. Um Testskripte anzulegen, können Sie die Funktion zum Aufzeichnen von Tests verwenden, um Testskripte automatisch anzulegen und manuell zu bearbeiten. Es wird empfohlen, das Testwerkzeug der komponentenbasierten Testautomatisierung zu verwenden, da es im Unterschied zu zahlreichen anderen Testwerkzeugen die automatische Aufzeichnung und manuelle Bearbeitung unterstützt.

Um eine Testkonfiguration und das zugehörige Testskript im richtigen SAP-Solution-Manager-Kontext anzulegen, empfehlen wir Ihnen, die Testkonfiguration aus der Lösungsdokumentation in der Anwendung *Test-Suite - Testvorbereitung* des SAP Solution Manager heraus anzulegen.

### Voraussetzungen

Sie haben die Testkonfigurationsattribute wie unter [Definieren von Testkonfigurationsattributen \[Seite 220\]](#) beschrieben definiert.

### Vorgehensweise

1. Öffnen Sie in der Anwendung *Test-Suite - Testvorbereitung* eine Lösung und einen Branch.
2. Navigieren Sie zum Geschäftsprozess oder Schritt, für den Sie den Test anlegen möchten.  
Klicken Sie in der Liste der Elemente mit der rechten Maustaste auf den unteren Bereich, wählen Sie **► Neu ► Testfälle ◄**, und legen Sie eine neue Testkonfiguration an, oder ordnen Sie eine vorhandene Testkonfiguration zu.
3. Wenn Sie die Testkonfigurationsattribute definiert haben, wählen Sie **<Testwerkzeug> starten**.
4. Das Aufzeichnungswerkzeug wird gestartet.

- Wenn Sie das Testwerkzeug der komponentenbasierten Testautomatisierung (CBTA) verwenden, wird der Assistent zum Anlegen eines Tests gestartet.  
Weitere Informationen finden Sie unter [Erfassung von Geschäftsszenarios \[Seite 55\]](#).
- Wenn Sie eCATT als Testwerkzeug verwenden, gelangen Sie zum Bild „Editor Testskript eCATT“. Weitere Informationen finden Sie in der Hilfe zur Anwendung *eCATT: extended Computer Aided Test Tool* im SAP Help Portal (<http://help.sap.com>) unter **► Testskripte entwickeln ► Testskripte anlegen ► Testskript-Editor ►**.
- Wenn Sie ein Drittanbieter-Testwerkzeug verwenden, wird der Prozess zum Anlegen des Testskripts gestartet.

### Kombinieren von Testskriptschritten aus unterschiedlichen Werkzeugen

Sie können eine End-to-End-Testkonfiguration aus Schritten zusammensetzen, die Sie in unterschiedlichen Testwerkzeugen angelegt haben: komponentenbasierte Testautomatisierung (CBTA), eCATT oder Drittanbieter-Testwerkzeuge.

In der Regel definieren Sie die Reihenfolge, in der die Testskriptschritte ausgeführt werden sollen. Dann definieren Sie die Parameterwerte für die Schritte. Weitere Informationen finden Sie unter [Bearbeiten eines Testskripts \[Seite 223\]](#).

Um ein zusammengesetztes Testskript anzulegen, legen Sie eine Testkonfiguration oder ein Testskript wie unter [Anlegen von Testkonfigurationen und Testskripten \[Seite 217\]](#) beschrieben an.

Wählen Sie *Zusammengesetzter Test* im Feld *Testwerkzeug*. Weitere Informationen finden Sie unter [Definieren von Testkonfigurationsattributen \[Seite 220\]](#).

## 9.1.3 Bearbeiten von Testskripten

### Voraussetzungen

Sie haben eine Testkonfiguration einschließlich Testskript angelegt und die Testkonfigurationsattribute gepflegt, wie in [Definieren von Testkonfigurationsattributen \[Seite 220\]](#) und [Anlegen eines Testskripts \[Seite 222\]](#) beschrieben.

### Kontext

Wenn Sie ein Testskript aufgezeichnet haben, können Sie es bearbeiten und vervollständigen:

- Sie können Komponenten eines Testskripts bearbeiten, das mit der komponentenbasierten Testautomatisierung in der Sicht *Test-Repository* aufgezeichnet wurde. Weitere Informationen finden Sie unter [Bearbeiten von Testskripten \[Seite 59\]](#).
- Sie können Schritte (Module) aus einem eCATT-Testskript oder aus einem eCATT-Testskript in Kombination mit Drittanbieter-Testwerkzeugschritten bearbeiten. Weitere Informationen finden Sie unter [Kombinieren von Testskriptschritten aus unterschiedlichen Werkzeugen](#) in [Anlegen eines Testskripts \[Seite 222\]](#).

- **Einschränkungen:**

- Ein in eCATT modifiziertes Testskript können Sie in den *Test-Repository*-Sichten nicht mehr bearbeiten. Um den Testfall zu bearbeiten, öffnen Sie die Testkonfiguration, und wählen *eCATT starten*.
- Wenn Sie ein Drittanbieter-Testwerkzeug verwenden, wird die Registerkarte *Testskript* im Editor für Testkonfigurationen nicht angezeigt. Der Inhalt des Testskripts kann nur vom Drittanbieterwerkzeug – nicht vom SAP Solution Manager – interpretiert werden.

## Vorgehensweise

1. Wählen Sie in der Sicht *Test-Repository - Testkonfiguration* die Testkonfiguration aus, die das Testskript enthält, das Sie bearbeiten möchten.
2. Wählen Sie die Registerkarte *Testskript*.

Die Testskriptschritte werden angezeigt. Bei einem eCATT-Testskript oder einem zusammengesetzten Testskript kann es sich bei den Testskriptschritten um Testskripte einer beliebigen Art von Testwerkzeug handeln.

3. Unter *Testskriptschritte* können Sie Testskriptschritte hinzufügen:
  - Geben Sie für ein eCATT-Testskript oder ein zusammengesetztes Testskript die folgenden Daten ein:
    1. Wählen Sie in der Spalte *Typ* den Typ des Testwerkzeugs aus, mit dem das Testskript angelegt wurde.
    2. Wählen Sie in der Spalte *Name* das Testskript aus.
    3. Geben Sie in der Spalte *Version* die relevante Version ein, die in den meisten Fällen **1** ist.

Sichern Sie Ihre Eingaben. Leere Zeilen werden automatisch entfernt.

4. Um die Reihenfolge der Testskriptschritte zu definieren, verwenden Sie die Drucktasten *Nach oben* und *Nach unten*.

Um mehrere Schritte mit einem Testskript zu duplizieren, können Sie die Funktion zum Kopieren und Einfügen verwenden.

Um einen überflüssigen Schritt zu entfernen (z.B. weil Sie ihn versehentlich zweimal ausgeführt haben), wählen Sie *Entfernen*.

5. Konfigurieren Sie die Parameter unter **▶ Testskript ▶ Parameter ▶**:
  - Um die an die Testskriptschritte übergebenen Werte zu definieren (Import- und Exportparameter), bearbeiten Sie die Unterregisterkarte *Parameter*:
    1. Legen Sie im Feld *Verwendung* die Verwendungsart fest:
      - *Kein Wert*: Kein Parameterwert wird an den Parameter des Testskripts übergeben. Der Testskriptparameter verfügt über keinen Standardwert.
      - *Fest*: Ein Konstantenwert wird an den Parameter übergeben. Der Konstantenwert ist nicht von der Ausführung des Testskripts abhängig.

### ❁ Beispiel

Wenn Sie immer einen Kundenauftrag mit 5 Pumpen anlegen möchten, geben Sie **5** für den Parameter *Menge* in die Spalte *Wert* ein. Wenn Sie für die Verwendung die Option *Fest* wählen, aber keinen Wert eingeben, wird der Wert NULL im Testskript verwendet. Zum Definieren einer leeren Zeichenfolge verwenden Sie den Wert **%BLANK%**.



- **Exponiert:** Der Parameter wird dem Parameter des übergeordneten Skripts zugeordnet. Der Parameterwert wird vom Parameter des übergeordneten Skripts übernommen (Importparameter) bzw. an den Parameter des übergeordneten Skripts (Exportparameter) übergeben.  
Importparameter können die Werte vom Testdatencontainer empfangen.  
Mehrere exponierte Importparameter können ihre Werte vom selben Parameter eines übergeordneten Skripts empfangen.  
Ein einzelner exponierter Exportparameter kann seinen Wert an mehrere Exportparameter des übergeordneten Skripts übergeben.

#### ♣ Beispiel

Sie möchten ein Haupttestskript, das 3 Schritte (Skripte) umfasst, mit den folgenden Parametern anlegen:

- Angebot anlegen  
Importparameter: Kunde, Material, Menge  
Exportparameter: Angebotsnummer
- Kundenauftrag aus Angebot anlegen  
Importparameter: Angebotsnummer  
Exportparameter: Kundenauftragsnummer
- Lieferung aus Kundenauftrag anlegen  
Importparameter: Kundenauftragsnummer  
Exportparameter: Lieferungsnummer

Sie legen 3 Schritte in der Testsequenz an. Um das Hauptskript für einen anderen Kunden, ein anderes Material und eine andere Menge ausführen zu können, müssen Sie die Lieferungsnummer kennen, die zum Schluss angelegt wird. Setzen Sie dazu für die folgenden Parameter die Verwendungsart auf **Exponiert**:

- Importparameter Kunde, Material und Menge des Skripts „Angebot anlegen“
- Exportparameter Lieferungsnummer des Testskripts „Lieferung aus Kundenauftrag anlegen“

- **Lokal:** Der an den Parameter übergebene Wert wird dem Parameter eines anderen Schritts zugeordnet. Der während der Ausführung des Testskripts tatsächlich verwendete Wert hängt von der Ausführung des übergeordneten Skripts und den in den vorigen Schritten angelegten Werten ab.

#### ♣ Beispiel

Sie möchten den Wert des Parameters „Angebotsnummer“, der durch den Testskriptschritt „Angebot anlegen“ erstellt wurde, an den Testskriptschritt „Kundenauftrag aus Angebot anlegen“ übergeben. Setzen Sie dazu die Verwendungsart auf **Lokal**, und führen Sie ein Mapping der beiden Parameter zwischen den beiden Schritten durch. Ordnen Sie dazu in der Spalte **Referenzparameter** die Namen der Parameter einander zu.

2. Definieren Sie die anderen Spalten nach Bedarf.
- Zum Anzeigen von SAP-Attributen eines eCATT-Testskripts oder eines Testskripts, das mithilfe eines Drittanbieter-Testwerkzeugs angelegt wurde, wählen Sie die Unterregisterkarte **Attribute** unter **Testskript**.  
Um die Attribute unter **Testskriptschritte** in der Spalte **Name** zu ändern, wählen Sie das Testskript.

- Zum Anzeigen der Beschreibung (sofern vorhanden) wählen Sie die Unterregisterkarte *Beschreibung*. Um die SAP-Beschreibung oder die Drittanbieter-Testwerkzeugbeschreibung für Testskriptschritte anzuzeigen, die mithilfe eines Drittanbieter-Testwerkzeugs angelegt wurden, markieren Sie jeweils den entsprechenden Auswahlknopf.
  - Zum Anzeigen des Screenshots eines Testskriptschritts (sofern vorhanden) wählen Sie die Registerkarte *Bild*.
  - Um zu prüfen, wo ein Testskript verwendet wird, markieren Sie das entsprechende Testskript, und wählen Sie *Verwendungsnachweis*.
  - Um einen Testskriptschritt in einer neuen Sitzung anzuzeigen oder zu bearbeiten, wählen Sie in der Spalte *Name* das entsprechende Testskript aus.  
Wenn das Drittanbieter-Testwerkzeug, mit dem das Skript angelegt wurde, nicht verfügbar ist, werden Sie zum eCATT-Testskript (Wrapper) weitergeleitet.
6. Um den Test auszuführen, wählen Sie *Ausführen*.

## 9.1.4 Definieren von Eingabe- und Ausgabeparametern

### Voraussetzungen

Sie haben ein Testskript oder eine Testkonfiguration angelegt. Weitere Informationen finden Sie unter [Anlegen von Testkonfigurationen und Testskripten \[Seite 217\]](#).

### Kontext

Im Editor für Testkonfigurationen können Sie auf der Registerkarte *Parameter* die Import- und Exportparameter für Testskriptschritte bearbeiten, um die Testskriptschnittstelle zu definieren. Diese Testskriptparameter werden als Parameter für die Testkonfiguration angezeigt, der das Testskript zugeordnet wurde. Sie können sie anschließend einem Testdatencontainer zuordnen, der die erforderlichen Testparameterwerte enthält.

### Vorgehensweise

1. Wählen Sie in der Sicht *Test-Repository - Testkonfiguration* eine Testkonfiguration aus.
2. Wählen Sie die Registerkarte *Parameter*.
3. Um die Parameter in der Reihenfolge der entsprechenden Testschritte im Testskript anzuzeigen, wählen Sie *Parameter: Standardreihenfolge*.
4. Sie können die Parameter von Testskriptschritten bearbeiten, die mit eCATT oder der komponentenbasierten Testautomatisierung aufgezeichnet wurden. Sie können z.B. aussagekräftigere Parameternamen und -beschreibungen zuordnen, was die Verwendung des Testskripts erleichtert.

Weitere Informationen finden Sie unter [Bearbeiten eines Testskripts \[Seite 223\]](#).

- Um zu den Testskriptschritten und den zugehörigen Parametern zu navigieren, wählen Sie die Komponente oder die Komponentenparameter.

Dadurch gelangen Sie zu den entsprechenden Einträgen auf der Registerkarte *Testskript*.

## 9.1.5 Weitere Einstellungen für Testkonfigurationen und Testskripte

### Verwendung

In den *Test-Repository*-Sichten können Sie weitere Attribute für Testkonfigurationen oder Testskripte festlegen. Sie können beispielsweise Testdatencontainer zur Wiederverwendung oder Testdatenvarianten zuordnen.

### Voraussetzungen

Sie haben die Testkonfiguration wie unter [Definieren von Testkonfigurationsattributen \[Seite 220\]](#) beschrieben definiert.

Sie haben das Testskript wie unter [Bearbeiten von Testskripten \[Seite 223\]](#) und [Definieren von Eingabe- und Ausgabeparametern \[Seite 226\]](#) beschrieben definiert.

### Vorgehensweise

#### Hinzufügen von Testdatencontainern

Um eine Testkonfiguration mit geänderten Werten wiederzuverwenden, können Sie mindestens einem Testdatencontainer Referenzen hinzufügen.

- Wählen Sie auf der Registerkarte *Testdaten* die Option *Hinzufügen*.
- Geben Sie in der Spalte *Alias* einen Alias ein.  
Bei der Abbildung der Parameter des Testskripts auf die Parameter eines Testdatencontainers werden die Referenzen angezeigt. Wenn der Name eines Testdatencontainers zu lang ist, kann die Erkennung der Referenzen Probleme bereiten. Um das Mapping überwachen zu können, geben Sie einen alphanumerischen Namen mit einer Länge von bis zu drei Zeichen für den Testdatencontainer ein.
- Wählen Sie in der Spalte *Testdatencontainer* den Testdatencontainer aus.  
Im Feld *Titel* wird der Name des Testdatencontainers angezeigt.  
Wenn der Testdatencontainer externe Testdatenvarianten enthält, wird das entsprechende Ankreuzfeld markiert, und der Name der Datei mit den externen Testdatenvarianten wird angezeigt.
- Sichern Sie Ihre Eingaben.  
Der Testdatencontainer wird Ihrer Testkonfiguration zugeordnet.
- Um Testdatenvarianten für eine Testkonfiguration anzulegen, gehen Sie wie folgt vor:**
  - Sie können wählen, ob der Parameterwert oder der Testdatencontainer-Alias und die Nummer der Variante, welcher der Parameterwert entnommen wird, angezeigt werden.

- Um den Parameterwert anzuzeigen, wählen Sie *Wert*. Es wird z. B. 200 angezeigt.
- Um den Testdatencontainer-Alias und den Parameternamen anzuzeigen, wählen Sie *Referenz*:

### ❁ Beispiel

Wenn Sie einen Testdatencontainer mit dem Alias TD1 statt dem Wert 200 verwenden, wird `<Parametername> (<Testdatencontainer-Alias>, <Nummer der Testdatencontainer-Variante>)` angezeigt, z.B. `I_ORDER_TYPE (TD1, 1)`.

- **Um den Parametern Ihrer Testkonfiguration eine Testdatenvariante zuzuordnen, gehen Sie wie folgt vor:**
  1. Wählen Sie die Drucktaste *Testdaten: Zuordnungsassistent*. Die Drucktaste ist nur aktiv, wenn das Testskript Parameter aufweist.  
Der Assistent zur Variantenpflege wird gestartet.
  2. Geben Sie die erforderlichen Daten ein.
  3. Wählen Sie *Ergebnis an Konfiguration übergeben*.
  4. Wählen Sie auf der Registerkarte *Testdaten* die Option *Aktualisieren*.  
Sie gelangen auf ein Dialogfenster.
  5. Zum Bestätigen wählen Sie *Ja*.  
Unter *Testdatenvarianten* wird die Testdatenvariante angezeigt.
  6. Damit die Testdatenvariante zur Ausführung ausgewählt werden kann, markieren Sie das Ankreuzfeld *Ausführen*.  
Wenn Sie mehrere Testdatenvarianten auswählen, können Sie beim Ausführen des Testfalls die gewünschte Testdatenvariante auf dem Bild *Startoptionen* angeben.  
Weitere Informationen finden Sie in der Hilfe zur Anwendung *eCATT: extended Computer Aided Test Tool* im SAP Help Portal (<http://help.sap.com>) unter **► Testskripte ausführen ► Startoptionen ►**.
  7. Sichern Sie Ihre Eingaben.

### Definieren von Suchbegriffen

Um die Suche nach Testskripten und Testkonfigurationen zu erleichtern, können Sie Suchbegriffe zuordnen. Suchbegriffe können beispielsweise einen bestimmten Benutzer, einen Anwendungsfall oder ein Szenario beschreiben, für das der Test ausgeführt werden soll.

1. Wählen Sie in den *Test-Repository*-Anwendungen des SAP Solution Manager eine Testkonfiguration oder ein Testskript aus.  
Suchbegriffe für ein Testskript gelten auch für die zugehörige Testkonfiguration und umgekehrt.
2. Wählen Sie die Registerkarte *Suchbegriffe*.
3. Geben Sie unter Einhaltung der in Ihrer Organisation geltenden Namenskonventionen Suchbegriffe ein. Sie können nicht mehr als 20 Zeichen eingeben.  
Die als Suchbegriffe eingegebenen Werte werden in Großbuchstaben umgewandelt.
4. Sichern Sie Ihre Angaben, und schließen Sie das Fenster.

### Ersetzen eines Testskripts in einer Testkonfiguration

Sie haben folgende Möglichkeiten, um ein Testskript zu ersetzen, das automatisch einer Testkonfiguration zugeordnet wurde:

- Wenn Sie die Kopie eines Skripts zuordnen möchten, das bereits vorhanden ist und geändert werden soll, wählen Sie **► Skript zuordnen ► Vorhandene kopieren ►**.

- Um einen Verweis auf ein bereits vorhandenes Skript anzulegen, wählen Sie ► [Skript zuordnen](#) ► [Vorhand.](#) ►.
- Um ein neues Testskript anzulegen, wählen Sie ► [Skript zuordnen](#) ► [Neu](#) ►.

### Verwenden von Experteneinstellungen für eCATT

Optional: Um erweiterte Funktionen für die Testskriptbearbeitung nutzen zu können, wählen Sie ► [Springen](#) ► [Expertenmodus](#) ► (Transaktion SECATT).

#### **i** Hinweis

Testkonfiguration oder Testskripte, die über die Transaktion SECATT bearbeitet werden, können in der Umgebung für die Testkomposition später nicht mehr bearbeitet werden.

## 9.1.6 Verwalten von Komponenten automatisierter Testfälle

### Kontext

Sie können folgende Komponenten von Testkonfigurationen bearbeiten, kopieren, umbenennen oder löschen:

- Testskripte (siehe [Bearbeiten von Testskripten \[Seite 223\]](#))
- Testdatencontainer
- Systemdatencontainer

In SAP Solution Manager wird zum Bearbeiten von Testskripten und Testkonfigurationen die Verwendung der [Test-Repository](#)-Sichten empfohlen. Nur wenn dort keine Änderungen möglich sind, sollten Sie die Schnittstelle für die erweiterte Testautomatisierung in der Transaktion eCATT verwenden.

Systemdaten-Container (SDCs) werden automatisch angelegt, wenn Sie die Lösungslandschaft ändern. SDCs werden nur verwendet, um die Kommunikation mit der eCATT-Schicht zu ermöglichen. Sie legen die Verbindungseinstellungen über die [SUT-Verwaltung \[Seite 17\]](#) fest. Das Aktualisieren von SDCs in der erweiterten Testautomatisierung ist nicht mehr relevant.

### Vorgehensweise

1. Wählen Sie in der Transaktion SECATT die Testkonfiguration, das Testskript, den Testdatencontainer oder den Systemdatencontainer aus, die/das/den Sie pflegen möchten.
2. Sie haben folgende Möglichkeiten:
  - Wählen Sie [Anzeigen](#) oder [Bearbeiten](#), um den Editor zu öffnen. Der entsprechende Editor wird geöffnet.
  - Wählen Sie [Objekt löschen](#). Sie gelangen auf ein Dialogfenster.

- Wenn Sie eine Testkonfiguration ausgewählt haben, können Sie wählen, ob Sie nur die Testkonfiguration selbst, die Testkonfiguration mit der aktuellen Version oder die Testkonfiguration mit allen Versionen des Testskripts auswählen möchten. Wenn Sie nur die Testkonfiguration selbst löschen, können Sie das entsprechende Testskript beim Anlegen einer neuen Testkonfiguration wiederverwenden.
- Wenn Sie einen Test- oder Systemdatencontainer ausgewählt haben, der einer Testkonfiguration zugeordnet ist, müssen Sie zuerst die Zuordnung entfernen. Wählen Sie dazu den *Verwendungsnachweis*, um die entsprechenden Objekte anzuzeigen und zu bearbeiten.

Wählen Sie *Ja*.

- Um ein Objekt zu kopieren oder umzubenennen, gehen Sie folgendermaßen vor:
  1. Wählen Sie *Objekt kopieren* oder *Objekt umbenennen*. Sie gelangen auf ein Dialogfenster.
  2. Geben Sie den Namen des Objekts ein, das kopiert oder umbenannt werden soll.
  3. Um die Testkonfiguration zu kopieren oder umzubenennen, ohne das entsprechende Testskript zu kopieren oder umzubenennen, entmarkieren Sie das Ankreuzfeld *Einschl. Testskript*. Wenn Sie einen Testfall kopieren, wird das Testskript des Quelltestfalls dem Zieltestfall zugeordnet. Wenn Sie einen Testfall umbenennen, wird das Testskript des Quelltestfalls dem Zieltestfall zugeordnet, ohne umbenannt zu werden.
  4. Wählen Sie *Kopieren* oder *Umbenennen*.
- 3. Sichern Sie Ihre Eingaben.
- 4. Beenden Sie die Funktion.

## 9.2 Testkonfigurationen und Testskripte ausführen

### Verwendung

Sie können die Ausführung einer Testkonfiguration in einem Testplan einplanen. In einem Testplan können Testkonfigurationen automatisch oder manuell von Testern ausgeführt werden.

Sie können außerdem eine Testkonfiguration oder ein Testskript manuell ohne Zeitplan ausführen, um sicherzustellen, dass der Test ordnungsgemäß funktioniert.

### Vorgehensweise

#### Test manuell ausführen

1. Wählen Sie in den *Test-Repository*-Sichten ein Testskript oder eine Testkonfiguration aus.
2. Wählen Sie *Ausführen*.  
Es wird ein neues Fenster geöffnet, in dem Sie die Startoptionen bearbeiten können. Weitere Informationen zu eCATT-Tests finden Sie in der Hilfe zur Anwendung *eCATT: extended Computer Aided Test Tool* unter <http://help.sap.com> unter **▶ Testskripte ausführen ▶ Startoptionen**.
3. Wählen Sie *Ausführen* (oder *Ohne Startoptionen ausführen*).

4. Prüfen Sie den Ausführungsreport. Weitere Informationen finden Sie unter [Ausführungs-Reports anzeigen](#) [Seite 64].

### **Einplanen der Ausführen von Tests in Testplänen und Testpaketen**

Weitere Informationen finden Sie unter [Testplanverwaltung](#) [Seite 235].

# 10 Testfälle

## Verwendung

Testfälle, in denen die Testkonfigurationen enthalten sind, werden in der Lösungsdokumentation zu Geschäftsprozessen zugeordnet. Dieser Schritt erfolgt in der Anwendung [Test-Suite - Testvorbereitung](#), wo Sie folgende Möglichkeiten haben:

- Auswählen von Lösungen und Branches aus der Hierarchie und Anzeigen von Geschäftsprozessen, Schnittstellen, ausführbaren Einheiten usw.
- Starten von Reports für einen ausgewählten Branch, um z.B. technische Stücklisten (TBOMs) und Testfälle auszuwerten
- Anlegen von Testfällen

Weitere Informationen finden Sie unter [Lösungsdokumentation](#).




## Aktivitäten

### Anlegen von Testfällen

In SAP Solution Manager 7.2 legen Sie Testfälle für einen Geschäftsprozessschritt in einer Lösung an.

1. Öffnen Sie in der Anwendung [Test-Suite - Testvorbereitung](#) eine Lösungsdokumentation.
2. Führen Sie einen Drilldown auf das Element (z.B. auf einen Geschäftsprozessschritt) aus, für das Sie einen Testfall anlegen möchten.

In der nachfolgenden Tabelle sehen Sie die eingeschlossenen Elemente, z.B. ausführbare Einheiten und Testfälle, die bereits zugeordnet wurden.

Klicken Sie mit der rechten Maustaste in die Tabelle mit Elementen, und wählen Sie  **Neu**  **Testfälle** .

Dann können Sie aus den folgenden Optionen wählen:

- **Zuordnen einer vorhandenen Testkonfiguration.**  
Wenn Sie diese Option wählen, müssen Sie eine Testkonfiguration auswählen.  
Dann definieren Sie weitere Informationen für den Testfall:
  - Verantwortlicher
  - Testklassifizierung, z.B.:
    - Funktionaler Integrationstest
    - Generischer Funktionstest
    - Regressionstest
    - Einzelner Funktionstest
    - Benutzertest
  - Testdatenvariante
  - Suchbegriffe: Dies kann nützlich sein, wenn Sie viele Testfälle haben und weitere Merkmale zuordnen möchten, nach denen Sie in der Kriterienschnellpflege suchen können. Suchbegriffe werden nicht in der Lösung, sondern in der Testkonfiguration definiert.



- Logische Zielkomponenten
- **Anlegen einer neuen Testkonfiguration.**  
Mit dieser Option können Sie eine neue Testkonfiguration anlegen und ein neues Testskript aufzeichnen (oder ein vorhandenes Testskript zuordnen). Wenn Sie diese Option wählen, werden der Editor für Testkonfigurationen und das Dialogfenster geöffnet, in dem Sie die Kopfdaten definieren. Weitere Informationen finden Sie unter [Anlegen von Testkonfigurationen und Testskripten \[Seite 217\]](#). Wenn Sie die Testkonfiguration abgeschlossen haben, wird sie dem ausgewählten Geschäftsprozessschritt zugeordnet.  
Dann definieren Sie die Merkmale für den Testfall wie oben beschrieben.
- **Anlegen eines neuen Testdokuments für manuelle Tests aus einer Vorlage.**  
Auch für Testdokumente definieren Sie weitere Testfallmerkmale:
  - Status
  - Sensitivität
  - Priorität
  - Stichwörter
  - Inhaber
  - Verantwortlicher
  - Testklassifizierung (siehe obige Beschreibung)
  - Dauer
- **Hochladen eines vorhandenen Testdokuments für manuelle Tests.**  
Sie können jede lokale Textdatei oder URL mit einer schriftlichen Beschreibung hochladen, die den Tester durch einen Testfall führt.
- **Zuordnen eines vorhandenen Testdokuments für manuelle Tests.**  
Sie können dasselbe Dokument zuordnen, das bereits für einen anderen Testfall hochgeladen wurde, oder Sie können ein vorhandenes Dokument kopieren, um eine unabhängige Version anzulegen.

### **i** Hinweis

Wenn eine Signatur einem Testdokument innerhalb eines Testplans zugeordnet ist, werden die Signaturdaten auch in den Testbericht aufgenommen. Dies ist relevant für Testfälle und alle Testdokumente.

3. Neu angelegte Testfälle werden auf der Elementregisterkarte für den Geschäftsprozess angezeigt.
4. Sichern Sie Ihre Angaben, und schließen Sie das Fenster.

Nun können Sie unter [Testplanverwaltung](#) die Testfälle einem Testpaket zuordnen. Weitere Informationen finden Sie unter [Testplanverwaltung \[Seite 235\]](#).

### **Testfalltypen**

Beim Anlegen eines Tests können Sie die folgenden Testfalltypen verwenden:

- [Testdokument-URL](#): bezieht sich auf manuelle Tests.
- [Testkonfiguration](#): bezieht sich auf automatische Tests für alle automatischen Werkzeuge, die in Ihrem System registriert sind. (Werkzeuge für zusammengesetzte Tests und das eCATT-Testwerkzeug müssen nicht explizit registriert werden).

### **i** Hinweis

Sie können bestimmte Testfalltypen deaktivieren, zum Beispiel wenn die Lizenz für ein Drittanbieter-Testwerkzeug abgelaufen ist.

In diesem Fall müssen Sie die Registrierung für das entsprechende Drittanbieter-Testwerkzeug aufheben. Weitere Informationen finden Sie unter [Konfiguration des Testmanagements \[Seite 121\]](#).

# 11 Testplanverwaltung

## Verwendung

Testpläne sind Sammlungen aller Testfälle, die Sie für eine spezifische Testphase benötigen. Sie können die folgenden Objekte umfassen:

- manuelle Testfälle
- Testkonfigurationen (eCATT, zusammengesetzte Tests sowie alle registrierten Testwerkzeuge von Drittanbietern)
- Ausführbare Einheiten

Mit der App „Testplanverwaltung“ können Sie Testpläne basierend auf der Lösungsdokumentation anlegen. Jeder Testplan ist für eine bestimmte Kombination aus Lösung, Branch und Umfang definiert. Die Arbeit mit einem Standardumfang ist nicht empfehlenswert, da dies zu einem schnellen Anstieg des Testvolumens und Performance-Problemen führen kann.

Im Testplan wählen Sie genau die Testfälle aus, die Sie in einer Testphase abdecken möchten. Sie definieren für jeden Testplan Testpakete, die aus einer Teilmenge aller Testfälle des Testplans bestehen. Den Testpaketen werden Tester zugeordnet.

Über das Menü [Springen](#) können Sie einen Testreport erzeugen und ausdrucken, der z. B. die Testfallbeschreibungen, zu den Testfällen angelegte Meldungen und den Status enthält.

Sie können Testsequenzen zum Sortieren der Testfälle definieren, um eine schrittweise Ausführung zu gewährleisten, wobei jeder Schritt einem anderen Benutzer zugeordnet werden kann. Zur besseren Kontrolle der Testausführung können Sie einen Workflow einrichten, der auf einem [Freigabestatusschema \[Seite 248\]](#) basiert.

## Integration

Organisieren Sie Tests, nachdem Sie eine Lösung angelegt haben. Weitere Informationen finden Sie unter [Lösungsdokumentation](#).

Wenn Sie einen Testplan für eine bestimmte Testphase anlegen, basiert die Testplanstruktur standardmäßig auf der Struktur der Lösungsdokumentation.

## Voraussetzungen

- Sie haben in der Landschaftsverwaltung eine Testsystemlandschaft eingerichtet.
- Sie haben die Lösung mit einem Branch und einem Umfang definiert.
- Sie haben der Kombination aus Lösung, Branch und Umfang die Testfälle und ausführbaren Einheiten zugeordnet, die Sie testen möchten.

## Funktionsumfang

Die *Testplanverwaltung* bietet folgende Möglichkeiten:

- In der Anwendung *Testplanverwaltung* können Sie Testpläne anlegen, bearbeiten, anzeigen und kopieren. Sie können mehrere Testpläne auswählen und eine Massenaktualisierung der zugehörigen Attribute ausführen.  
Sie können Testpläne auch löschen; dies wird jedoch nur für z.B. Dummy-Testpläne empfohlen. Echte Testpläne dürfen nicht gelöscht werden, da sie ein Beleg für korrekte Testverfahren sind.
- Reports für ausgewählten Testplan starten (wählen Sie ► *Springen* ► *Testfallanalyse* ►):
  - Testpaketanalyse
  - Testfallanalyse
  - Fehlerübersicht
  - Testreport anlegen
- Historie eines Testplans aufrufen (wählen Sie ► *Springen* ► *Testplanhistorie* ►)
- Testplan der Favoritenliste hinzufügen oder daraus entfernen (wählen Sie *Favoriten*)
- Zu einem ausgewählten Testplan können Sie die Liste der Testpakete anzeigen und Aktivitäten bei den Testpaketen ausführen:
  - Testpakete anlegen, aus Testsequenz anlegen, bearbeiten, anzeigen und kopieren
  - Per Massenerstellung mehrere Testpakete gleichzeitig anlegen (mit vordefinierten Kopfdaten, z.B. allgemeine Daten und Einstellungen, für alle Testpakete)
  - Mehrere Testpakete auswählen und eine Massenaktualisierung der zugehörigen Attribute ausführen
  - Testpakete löschen
  - Zugeordnete Tester sperren oder entsperren (betrifft alle den Testpaketen zugeordnete Tester)
  - Zu den Testpaketanlagen navigieren
  - Reports für das ausgewählte Testpaket starten (wählen Sie „Springen“ -> „Testfallanalyse“)
  - Fehlerübersicht
  - Historie eines Testpakets aufrufen (wählen Sie ► *Springen* ► *Testpakethistorie* ►)
- Zu einem ausgewählten Testplan können Sie die Liste der Testsequenzen anzeigen und Aktivitäten in Bezug auf die Testsequenzen selbst ausführen:
  - Testsequenzen anlegen, bearbeiten, anzeigen und kopieren
  - Testsequenzen löschen

### **i** Hinweis

Wenn ein Test bereits ausgeführt wird, werden durch das Löschen eines Testplans, einer Testsequenz oder eines Testpakets auch alle zugehörigen Informationen wie Teststatus, Testnotizen, Testergebnisse und Fehlerzuordnungen gelöscht.

## 11.1 Workflow und Benachrichtigungen

### Verwendung

Der Testorganisator kann für Testpläne Workflows für den automatischen Versand von E-Mail-Benachrichtigungen zu ausgewählten Ereignissen aktivieren. So kann das System einen Tester beispielsweise über freigegebene und gesperrte Testpakete informieren oder – bei einer Reihe von Testfällen – darüber, dass ein Tester einen vorhergehenden Testfall abgeschlossen hat.

Der Workflow wird durch folgende Aktivitäten ausgelöst:

- **Wenn Sie den Freigabestatus eines Testplans ändern** und dadurch die Testausführung zulassen oder verbieten und wenn die Zuordnung des betreffenden Testers nicht gesperrt ist, wird der Tester in einer E-Mail darüber informiert, welche Testpakete in seinem Arbeitsvorrat testbereit sind bzw. für welche Testpakete keine Tests zulässig sind.
- **Wenn Sie die Zuordnung eines Testers zu einem Testpaket sperren oder entsperren** und der Freigabestatus die Testausführung zulässt, wird der Tester in einer E-Mail darüber informiert, dass er einen Testfall ausführen kann bzw. das Testen beenden muss.
- **Wenn ein Tester einem Testpaket zugeordnet oder seine Zuordnung zu einem Testpaket aufgehoben wird** und der Testplan zum Testen freigegeben ist, wird der Tester in einer E-Mail darüber informiert, dass er Tests ausführen kann bzw. das Testen beenden muss.
- **Wenn ein Tester einen Testfall aus einer Reihe von Testfällen abschließt**, wird der Tester per E-Mail über den nächsten Testfall informiert, dessen Test gestartet werden kann. Der Tester, der dem ersten testbereiten Testfall in einer Testsequenz zugeordnet ist, wird per E-Mail über eine der oben genannten Aktivitäten informiert.

Im Anschluss an diese Benachrichtigung erhält der Testplaner eine Bestätigungs-E-Mail mit Details zu Benachrichtigungen und Fehlern.

### Voraussetzungen

- Sie haben in der [Testplanverwaltung](#) einen Testplan mit mindestens einem Testpaket angelegt und dem Testpaket mindestens einen Tester zugeordnet.
- Sie haben im Customizing (Transaktion `SPRO`) für SAP Solution Manager unter [► Test-Suite-Vorbereitung ► Spezielle Konfiguration ► Workflow und E-Mail-Benachrichtigungen ►](#) die Aktivitäten unter [Workflow](#) ausgeführt.  
Weitere Informationen finden Sie in der Dokumentation zu den Customizing-Aktivitäten.
- Sie haben in der Anwendung [Testplanverwaltung](#) auf der Registerkarte [Einstellungen](#) der Testpläne Folgendes definiert:
  - Sie haben das Kennzeichen [Workflow](#) gesetzt.
  - Sie haben das Freigabeschema und den Freigabestatus definiert.  
Weitere Informationen finden Sie unter [Freigabestatusschema \[Seite 248\]](#).
- Sie haben eine E-Mail-Adresse für die Geschäftspartner definiert, die als Tester oder als Verantwortliche für Testpläne oder Testpakete fungieren.

## Prozess

1. Wählen Sie in der Anwendung *Testplanverwaltung* den Testplan aus, in dem Sie die relevanten Testfälle ausgewählt und ein oder mehrere Testpakete angelegt haben.  
Öffnen Sie das Testpaket des ausgewählten Testplans.
2. Auf der Registerkarte *Tester* ordnen Sie Tester zu.  
Wenn das Testpaket mit einer Testsequenz als Referenz angelegt wurde, werden die entsprechenden Testfälle auf der Registerkarte *Tester* in der korrekten Reihenfolge aufgelistet, und Sie können jedem Testfall einen Tester zuordnen.  
Sie können einen Testfall auch neu zu einem anderen Benutzer zuordnen. In diesem Fall werden entsprechend auch alle verfügbaren Statusinformationen dem neuen Tester zugeordnet.
3. Wenn die Testertzuordnungen abgeschlossen sind, ändern Sie den Freigabestatus Ihres Testplans in einen Status, der die Testausführung erlaubt.
4. Bei Standardtestpaketen erhalten alle den Testpaketen zugeordneten Tester die E-Mail-Benachrichtigung (sofern ihre individuellen Zuordnungen nicht gesperrt sind).
5. Bei Testpaketen, die sich auf eine Testsequenz beziehen, erhält nur der Tester, der dem ersten Testfall zugeordnet ist, die E-Mail-Benachrichtigung (sofern seine Zuordnung nicht gesperrt ist). Wenn die Testausführung bereits in einer vorhergehenden Phase gestartet wurde, wird die Benachrichtigung an den Tester gesendet, der dem jeweils testbereiten Testfall in der Sequenz zugeordnet ist. (Wenn z.B. der erste und der zweite Testfall bereits ausgeführt und abgeschlossen sind, ist der dritte Testfall der testbereite Testfall.)  
In einer Testsequenz bearbeitet der erste Tester den ihm zugeordneten Testfall (bzw. die ihm zugeordneten Testfälle). Nach Beendigung des Vorgangs setzt der Tester den Status auf *Test beendet*.  
Das System informiert anschließend den folgenden Tester darüber, dass dieser Testfall nun zum Test bereitsteht.  
Die Benachrichtigung wird nur versendet, wenn es sich bei dem Tester des nächsten Testfalls nicht um dieselbe Person wie beim ersten Testfall handelt.
6. Um Änderungen am Testplan durchzuführen, ohne den Test zu sperren, wählen Sie einen Freigabestatus aus, der gleichzeitig sowohl Änderungen als auch die Testausführung zulässt (z.B. *Test in Anpassung (dringend)*), und sichern Sie den Testplan.  
Wenn Sie einen Tester sperren oder wieder freigeben, sendet das System eine entsprechende Benachrichtigung an den betreffenden Tester, jedoch nicht an die übrigen Tester im Testpaket.  
Wenn Sie die Testertzuordnung ändern, informiert das System alle Tester, die von der Änderung betroffen sind.

Der Testorganisator erhält eine E-Mail-Benachrichtigung, wenn der Workflow für den Testplan oder das Testpaket durch die Änderung eines Status oder einer Zuordnung initiiert wird. Er erhält außerdem dann eine E-Mail-Benachrichtigung, wenn die Benachrichtigung des nächsten Testers in einer Sequenz fehlgeschlagen ist.

Benachrichtigungen können unabhängig vom Kennzeichen *Workflow* im Testplan auch im Rahmen der automatisierten Testausführung versendet werden.

## 11.2 Workflow für Testdokument-Signaturen

Sie können das bereitgestellte API zum Anlegen eines eigenen Testdokument-Signatur-Workflows verwenden, z.B., um Benutzer zu informieren, wenn ihre digitale Signatur für Testdokumente, wie z.B. Testpläne,

Testpaketanlagen, Testergebnisse oder Testnotizen, erforderlich ist. Die Schnittstelle stellt eine Methode zum Anzeigen des Dokumentinhalts sowie eine weitere Schnittstelle für den Zugriff auf die Dokumentübersichtsanwendung zur Verfügung, auf der Benutzer das Dokument mit einer digitalen Signatur versehen können.

Sie können einen Workflow beispielsweise so anlegen, dass wenn ein bestimmter Status von einem Benutzer gesetzt wird, der neue Status nur wirksam wird, wenn der Testmanager den neuen Status mit seiner digitalen Signatur bestätigt. Sie können den Workflow so konfigurieren, dass der Testmanager eine E-Mail-Benachrichtigung erhält, in der er informiert wird, dass seine Signatur erforderlich ist. Die E-Mail könnte eine Verknüpfung zur Anzeige des Testdokuments sowie eine weitere Verknüpfung zur direkten Navigation zu der jeweiligen Anwendung, in der der Testmanager seine digitale Signatur vornehmen kann, enthalten.

#### **i Hinweis**

Beachten Sie, dass kein vordefinierter Dokumentensignatur-Workflow von SAP ausgeliefert wird. Sie müssen das API zur Entwicklung eines benutzerspezifischen Workflows nach Ihren Bedürfnissen verwenden.

Weitere Informationen finden Sie im entsprechenden Wissensdatenbankartikel unter: [2769575](#)

## **11.3 Anlegen und Bearbeiten von Testplänen**

### **Verwendung**

Sie können Testpläne für eine Teilmenge Ihrer Lösungsdokumentation in SAP Solution Manager anlegen. Ein Testplan bezieht sich auf eine Lösung, einen Branch und einen Umfang. Zum Anlegen des Testplans können Sie die Testfälle und ausführbaren Einheiten auswählen, die Sie der Lösungsfamilie, dem Branch und der Sicht auf einer beliebigen Ebene zugeordnet haben. Allerdings sind Testfälle und ausführbare Einheiten normalerweise einem Prozessschritt zugeordnet. Tester führen die Testfälle aus, indem Sie die zugeordneten ausführbaren Einheiten aufrufen und die Testfälle ausführen.

#### **i Hinweis**

In SAP Solution Manager können Testpläne nur auf einer Lösung, einem Branch und einem Umfang basieren.

### **Voraussetzungen**

- Sie haben die Lösungsdokumentation und die entsprechende Systemlandschaft in SAP Solution Manager definiert.
- Sie haben in der Lösungsdokumentation die ausführbaren Einheiten definiert, die Sie testen oder automatisieren möchten.
- Sie haben in der Lösungsdokumentation Testfälle für die zu testenden Prozesse angelegt oder zugeordnet. Die Testfälle umfassen die Testkonfigurationen und Testdokumente.
- Sie haben in der Lösungsdokumentation auf die Lösung, den Branch oder den Umfang zugegriffen, die bzw. die Sie testen möchten.

Sie sollten die Testkonfigurationen direkt aus der Lösungsdokumentation heraus anlegen. Dies erleichtert die Zuordnung von ausführbaren Einheiten.

## Aktivitäten

### Anlegen eines Testplans

1. Wählen Sie in der Anwendung [Testplanverwaltung](#) ► [Testplan](#) ► [Anlegen](#) ►.  
Der Editor für Testplanverwaltung kann alle relevanten Daten in mehreren Registerkarten eingeben. Sie müssen einige obligatorische Informationen unter [Allgemeine Daten](#) und [Einstellungen](#) definieren, bevor Sie den Testplan sichern können. Nachdem Sie den Testplan gesichert haben, können Sie diesen zu einem späteren Zeitpunkt weiterhin ändern oder vervollständigen.
2. Registerkarte [Allgemeine Daten](#):
  - Geben Sie unter [Kopfdaten](#) eine Testplan-ID ein. Dies ist die ID Ihres Testplans, die in SAP Solution Manager eindeutig sein muss. Namenskonventionen können die Organisation von Testplänen und Benutzerzugriff über Berechtigungsprüfungen unterstützen.  
Geben Sie eine Beschreibung und den Verantwortlichen ein. Standardmäßig wird der aktuelle Benutzer als Verantwortlicher ausgewählt. Jeder Benutzer mit der erforderlichen Berechtigung kann auf die Testpläne zugreifen.
  - Wählen Sie unter [Lösungsdokumentation](#) die Lösung, den Branch, den Umfang und die Systemrollen-ID, zu denen der Testplan gehört.

#### i Hinweis

Diese Auswahl kann so lange geändert werden, bis ein erstes Testpaket oder eine erste Testsequenz für den Testplan angelegt wird. Durch Änderung dieser Auswahl werden die Hierarchie und die möglicherweise bereits vorgenommene Auswahl von Testfällen zurückgesetzt.

Wählen Sie die Systemrolle, um zu definieren, in welchen Systemen der Testplan ausgeführt werden soll.

#### i Hinweis

Wenn Sie die Systemrolle ändern, wird die Anzeige der Systeme unter [Systemlandschaft](#) aktualisiert. Dies betrifft das Zielsystem, in dem der Test ausgeführt werden soll.

- Wählen Sie Lösung, Branch und Umfang zum Testplan.

#### i Hinweis

Nachdem der Testplan gesichert und eine Testsequenz oder ein Testpaket angelegt wurde, können Sie die Auswahl nicht mehr ändern.

- Site: Falls Sites für den ausgewählten Branch aktiviert waren, müssen Sie die Site angeben, für die der Testplan verwendet wird. Die ausgewählte Site wirkt sich auf Folgendes aus:
  - Zielsystem zur Testausführung: Wenn Sites für die logische Komponentengruppe aktiviert wurden, wird das Zielsystem auf Basis der angegebenen Site bestimmt. Andernfalls wird der Test auf Basis der Landschaft „Global“ ausgeführt. Falls „Global“ explizit als Site ausgewählt wird, wird nur die Landschaft „Global“ verwendet, d.h., site-spezifische Daten werden ignoriert.



- Testfallhierarchie-Inhalt: Wenn „Global“ ausgewählt wird, enthält die Hierarchie nur Testfälle, die nicht Site-spezifisch sind. Testfälle, die eine spezifische Site in der Lösungsdokumentation enthalten (oder eine Site von übergeordneten Testfällen übernommen haben) werden also ausgeschlossen. Falls Sie eine bestimmte Site angeben (d.h., eine andere Site als „Global“), enthält die Hierarchie nur Testfälle, die für diese Site gültig sind. Dies beinhaltet Testfälle, deren Site-Attribut (oder Site-Attribut der übergeordneten Testfälle) die angegebene Site enthält, sowie Testfälle, die für alle Sites gültig sind.

### i Hinweis

Für Testfälle, die angelegt wurden, bevor die Site „Global“ in der Test-Suite unterstützt wurde oder bevor Sites für den Branch aktiviert wurden, zeigt die Hierarchie alle Testfälle an, gleichgültig, ob sie site-spezifisch oder relevant für alle Sites sind.

- Unter *Systemlandschaft* bestimmen die Systemlandschaft, die in der Lösung definiert ist, und die Systemrollen-ID gemeinsam, welche Systemlandschaft angezeigt wird. Beide werden unter *Lösungsdokumentation* ausgewählt. Die Testfälle werden in den Systemen ausgeführt, die in der Liste enthalten sind.
  - Unter *Erstellung/Änderung* werden Datum und Zeitstempel der Erstellung und Änderung sowie der Änderer angezeigt. Außerdem können Sie sehen, ob es seit der letzten Aktualisierung des Testplans Änderungen an der Prozessdokumentation gegeben hat (Änderungen an der Struktur, der Version der Testdokumentation und der Testdatenvariante).
3. Registerkarte *Einstellungen*:
- Definieren Sie die Freigabeinformationen und weitere Attribute des Testplans.
- Unter *Freigabestatus* legen Sie fest, welche Aktionen mit dem Testplan möglich sind: Wenn der Testplan änderbar ist, können Sie beispielsweise Testfälle auswählen, Attribute ändern oder Testpakete anlegen. Wenn der Testplan ausführbar ist, können Tester den Status definieren und Hinweise und Fehlerkorrekturen anlegen. Die folgenden Freigabestatuswerte lassen keine Änderungen oder Ausführung zu:
    - Zwischenstatus: Warten auf mehrere Signaturen
    - Endstatus: Der Testplan ist eingefroren. In der Lösungsdokumentation vorgenommene Änderungen werden nicht mehr verfolgt und diese Aktion kann nicht rückgängig gemacht werden. Verwenden Sie diesen Status nur am Ende der Testphase. Standardmäßig ist ein Testplan mit Endstatus in der Analyse oder Testplanliste nicht sichtbar.
  - Unter *Attribute* definieren Sie Folgendes:
    - Der Wert für *Ausführbare Einheiten in Testfallauswahl* legt fest, wie die ausführbaren Einheiten angezeigt werden, wenn ein Benutzer Testfälle in einem Testplan auswählt. Es gibt drei Möglichkeiten:
      - Nur Testfall anzeigen* bedeutet, dass die ausführbaren Einheiten nicht in einem Testplan auswählbar sind, wenn mit ihnen kein Testfall verknüpft ist. Ein Tester kann die ausführbaren Einheiten daher nicht ohne Testfall prüfen. Ausführbare Einheiten, denen Testfälle zugeordnet sind, werden in der Testplanhierarchie angezeigt. Sie werden oberhalb des zugeordneten Testfalls angezeigt. Der Tester kann den Status im Testfall testen und dokumentieren.
      - Ausführbare Einheit anzeigen, wenn Testfall fehlt* bedeutet, dass die ausführbare Einheit zur Ausführung ausgewählt werden kann, wenn eine ausführbare Einheit, jedoch kein Testfall zugeordnet ist, und der Tester die ausführbare Einheit prüfen kann.
      - Testfälle und ausführbare Einheiten anzeigen*: Ausführbare Einheiten ohne zugehörige Testfälle werden wie in der vorangehenden Option behandelt. Ausführbaren Einheiten und die zugehörigen Testfälle sind verfügbar und können für die Ausführung ausgewählt werden.

- *Nur freigegebene Dok. zulässig* legt fest, ob die Testdokumente im Testplan berücksichtigt werden. Ist das Kennzeichen gesetzt, werden Dokumente mit einem anderen Dokumentstatus als „Freigegeben“ (oder die als freigegebenes Dokument definiert sind) ignoriert. Dies trifft auch für eine neue „nicht freigegebene Version“ von Dokumenten zu. Diese Einschränkung ist nur möglich, wenn *Nur Testfall anzeigen* für *Ausführbare Einheiten in Testfallauswahl* ausgewählt wurde.
- Die *Testklassifizierung* ordnet den Testplan einer Kategorie zu.
- Sie können eine *Testreihe* auswählen, die im Customizing für SAP Solution Manager unter **SAP Solution Manager > Capabilities > Test-Suite > Test-Suite für SAP Solution Manager > Vorbereitung > Setup > Testreihe definieren** definiert wurde. Diese hat keine Auswirkungen auf den Testplan, Sie können sie aber beispielsweise in Reports als Filterkriterium verwenden. Auf diese Weise können Sie Reports leicht für alle Testpläne ausführen, die derselben Testreihe zugeordnet sind.
- Die *Dokumentart für Testnotiz/Testergebnisse* definiert die Dokumentart, die während der Testausführung angelegt werden kann. Die Werte in der Dropdown-Selektion sind abhängig von den Dokumentarten, die mit der Lösung verknüpft sind. Die Vorlagen der Dokumentenarten werden in der *Lösungsdokumentation* unter *Dokumentenarten-Verwaltung* verwaltet.
- Unter *Testplan-Anlagen* können Sie Anlagen zum Testplan direkt anlegen oder zuordnen. Titel, Dokumentart, Status und Sprache sind obligatorisch. Zur Zuordnung von Anlagen stehen Ihnen folgende Funktionen zur Verfügung:
  - *Von Upload anlegen* zum Hochladen einer Datei aus einer lokalen Quelle.
  - *Aus URL anlegen* zum Anlegen einer neuen Anlage mittels Vorlagen, die der Administrator für verschiedene Dokumentenarten hochgeladen hat.
 Die Anlagen, die Sie hier zuordnen, können später beim Erstellen von Testpaketen wiederverwendet werden. Dort können Sie einen Link auf eine Anlage verwenden oder die Anlage kopieren.
- Definieren Sie die Planungsebene und die geplanten Start- und Enddaten unter *Daten*. Zur Planung auf Testplanebene können Sie auch den geplanten Aufwand erfassen. Der tatsächliche Aufwand wird während der Ausführungsphase eines Testplans automatisch aktualisiert. Das tatsächliche Enddatum können Sie am Ende des Ausführungszeitraums manuell festlegen.

#### 4. Registerkarte *Testfallauswahl*:

Je nach ausgewählter Lösung können Sie die einbezogenen Prozesse expandieren. Auf der niedrigsten Ebene können Sie alle Testfälle und ausführbaren Einheiten sehen, die Sie für jeden Prozess definiert haben.

Wählen Sie die Testfälle aus, die in den Testplan einbezogen werden sollen. Sie können den Haken beispielsweise für einen vollständigen Prozess setzen. In diesem Fall werden alle untergeordneten Testfälle einbezogen. Wenn Sie nur einzelne Testfälle auswählen möchten, setzen Sie den Haken für den ausgewählten Testfall.

Verwenden Sie die *Filterfunktion*, um die sichtbare Hierarchie einzuschränken und die Auswahl des gewünschten Testfalls zu erleichtern.

Sie können ein *Filterprofil* für den späteren Gebrauch sichern, verschiedene *Filtereinstellungen* zum Bearbeiten von Knoten verwenden, nach *Strukturknoten in der Lösungsdokumentation* zum Hinzufügen von Kriterien filtern und *zusätzliche Filter für Testfälle* auswählen, in denen sich Testfallkriterien direkt auf mehreren Registerkarten angeben lassen.

Wenn Sie Ihren Testplan fertiggestellt und alle Testpakete definiert haben, können Sie die Auswahl ganz einfach bereinigen. Filtern Sie dazu beispielsweise nach nicht zugeordneten Testfällen, wählen Sie alle Ergebnisse aus, und wählen Sie *Entfernen*. Testfälle, die keinem Testpaket zugeordnet sind, werden aus der Testplanauswahl entfernt.

Sie können eine beliebige Komponente aus der Testfallauswahl markieren und *Zuordnen zu* für ein Testpaket oder eine Testsequenz wählen.

Registerkarte *Testsequenzen*:

Wenn Sie den Testplan gesichert haben, können Sie Testsequenzen anlegen, die den Testplan in logische, kleinere Arbeitseinheiten aufteilen. Die Testsequenz beschreibt normalerweise ein durchgängiges Szenario, das in der vorgegebenen Reihenfolge getestet wird. Dies ist beispielsweise sinnvoll, wenn im Rahmen eines Geschäftsprozesses ein Testfall auf den Ergebnissen eines anderen Testfalls aufbaut. Es gibt zwei Möglichkeiten:

- Ein **als Referenz aus einer Testsequenz angelegtes Testpaket** verwendet dieselbe Sequenz wieder. Die Sequenz legt fest, dass ein Testfall erst dann zum Test bereitsteht, wenn der Vorgängertestfall abgeschlossen ist. In diesem Fall ist das Kennzeichen *Bereit zum Test* gesetzt. Sie können die Testfallselektion im Testpaket nicht ändern und Änderungen an der Testsequenz werden automatisch mit dem Testpaket synchronisiert.
- Ein **aus einer Sequenz als Vorlage angelegtes Testpaket** hat anfangs dieselbe Auswahl an Testfällen, ausführbaren Einheiten und Sequenzen, es kann jedoch nach der Erstellung nicht mehr geändert werden und ist unabhängig von den Änderungen in der Testsequenz.

Innerhalb eines Testplans können Sie ein oder mehrere Testsequenzen anlegen. In der Testsequenz wählen Sie eine Teilmenge der Testfälle und ausführbaren Einheiten aus dem Testplan aus und legen ihre Sequenz fest:

1. Wählen Sie **Testsequenz > Anlegen**. Der Editor für Testsequenzen wird in einem neuen Fenster geöffnet.
2. Definieren Sie zusätzlich zu der abgerufenen Testplan-ID Folgendes:
  - Wählen Sie auf der Registerkarte *Testfallsequenz* die Testfälle und ihre Reihenfolge. Sie können ein durchgängiges Dokument zuordnen, das das vollständige Szenario beschreibt. Das Dokument ist ein Testdokument im Testplan und kann auch als Information für die Testsequenz verwendet werden.
  - Auf der Registerkarte *Testpakete* können Sie ein Testpaket aus der Testsequenz anlegen, jedoch nur dann, wenn die Testsequenz sich im Anzeigemodus befindet und wenn der Testplanstatus Änderungen zulässt. Die Registerkarte zeigt die Testpakete an, die von der Testsequenz abhängen (Testpakete mit Testsequenz als Referenz).

Sie können die laufende Nummer verwalten (wenn der Testplanstatus Änderungen zulässt), die Testfallauswahl und andere Informationen ändern.

Auf der Registerkarte *Allgemeine Daten* können Sie sehen, ob die Testausführung für referenzierte Testpakete bereits begonnen hat. Änderungen können zum Zurücksetzen der Testausführungsdaten führen.

Änderungen an Testsequenzen werden sofort auf die zugehörigen Testpakete angewendet. Falls die Testausführung bereits gestartet wurde, müssen Sie entscheiden, inwieweit die Ausführungsergebnisse angepasst werden sollen:

- *Option zurücksetzen*:
  - *Ab erster Änderung zurücksetzen* (Standardoption): Die Zurücksetzung beginnt erst ab dem ersten betroffenen Testfall.
  - *Alle zurücksetzen*: Die Zurücksetzung wird auf alle Testfälle angewendet, auch wenn die Änderungen sich nicht auf sie auswirken (z.B. würde sich die Entfernung des dritten Testfalls nicht auf den für den ersten und zweiten Testfall ausgeführten Test auswirken).
- *Umfang zurücksetzen*:
  - *Status*: Dies ist der Mindestumfang, in dem der Status der Testfälle auf *noch nicht getestet* zurückgesetzt wird.

- **Testnotiz:** Wenn *Testnotiz* nicht für den Umfang ausgewählt wird, bleiben die Testnotizen verfügbar.
- **Testergebnisse:** Wenn *Testergebnisse* nicht für den Umfang ausgewählt wird, bleiben sie verfügbar.
- **Fehler:** Wenn Fehler kein Bestandteil des Umfangs sind, bleiben sie verfügbar.

Standardmäßig sind alle vier Bestandteil des Umfangs. Fehler und Dokumente sind noch im System enthalten, werden jedoch nicht mehr auf den Teststatus bezogen.

#### 5. Registerkarte *Testpakete*:

Nachdem Sie den Testplan gesichert haben, können Sie Testpakete anlegen, die den Testplan in logisch zugehörige, kleinere Arbeitseinheiten für die einzelnen Tester aufteilen.

Innerhalb eines Testplans können Sie ein oder mehrere Testpakete anlegen. Ein Testpaket ist eine Teilmenge des Testplans und fasst die Testfälle, die ein Tester in einem ausgewählten Zeitraum ausführen soll, zu einer Gruppe zusammen.

Wählen Sie ► *Testpaket* ► *Anlegen* ►. Der Editor für Testpakete wird in einem neuen Fenster geöffnet. Im Editor für Testpakete rufen Sie dieselben Informationen ab, die Sie auch für den Testplan festgelegt haben, wie z.B. Testplan-ID, Prozessdokumentation und Einstellungen.

Zusätzlich definieren Sie Folgendes:

- Testpaket-ID, Beschreibung, Verantwortlicher
- Geplante Start- und Enddaten, die in den Zeitrahmen passen müssen, den Sie für den Testplan definiert haben.
- Geplanter Aufwand, den Sie definieren müssen, wenn *Testpaket* als Planungsebene festgelegt wurde.

#### 6. Registerkarte *Testfallauswahl*: Sie sehen die Testfallhierarchie für den Testplan und können die Auswahl für das ausgewählte Testpaket einschränken.

#### 7. Registerkarte *Tester*: Sie können dem Testpaket Tester zuordnen, die Testerzuordnung sperren und entsperren und Details des Testfallstatus für einen ausgewählten Tester anzeigen.

#### 8. Registerkarte *Status*: Sie sehen die aktuelle Testauswertung für alle Testfälle. Der Gesamtstatus für einen Testfall in einem Testpaket hängt von der im Customizing definierten Auswertungsregel ab und berücksichtigt entweder das Ergebnis der letzten Ausführung oder den schlechtesten Status.

**Sie können auch ein Testpaket aus einer Testsequenz anlegen.** Wählen Sie ► *Testpaket* ► *Aus Testsequenz anlegen* ►. Wählen Sie im Popup-Fenster die Anlegart wie oben unter 5 beschrieben aus.

- Wenn Sie den **Vorlagenmodus** ausgewählt haben, befinden Sie sich in derselben Situation wie oben unter 5 beschrieben, mit einer vordefinierten Auswahl von Testfällen.
- Wenn Sie den **Referenzmodus** ausgewählt haben, gibt es einige Unterschiede:  
Die Registerkarte *Allgemeine Daten* zeigt die referenzierte Testsequenz (ID und Beschreibung) an. Die Registerkarte *Testfallauswahl* befindet sich lediglich im Anzeigemodus und stellt die Registerkarte *Testfallsequenz* aus der Testsequenz dar.  
Die Registerkarte *Tester* zeigt die Liste der Testfälle an und ermöglicht die Zuordnung von Testern auf Testfallebene. Sie können die Testerzuordnung ersetzen, so dass die vom ursprünglichen Tester festgelegten Testausführungsdaten dem neuen Test zugeordnet werden. Die Zuordnung von Testern zum Testpaket ist noch relevant und wird von der einzelnen Zuordnung abgeleitet.

#### 9. Registerkarte *Testdatensätze*: Sie können Ihrem Testplan Testdatensätze zuordnen. Dies sind Verweise auf Testdatencontainer. Der auf Testplanebene zugeordnete Testdatensatz kann in Testpakete propagiert werden.

Nachdem der Testdatensatz einem Testpaket zugeordnet wurde, ist er in der Anwendung „Manuelle Testausführung“ sichtbar und der Tester kann direkt darauf zugreifen.

## Bearbeiten eines Testplans

Sie können die Attribute eines Testplans ändern, z.B. den Freigabestatus, das geplante Start- und Enddatum oder den Verantwortlichen für den Testplan. Sie können Ihren vorhandenen Testplänen auch Testfälle hinzufügen.

In der Anwendung *Testplanverwaltung* können Sie einen Testplan auswählen und bearbeiten. Der Editor für Testpläne wird geöffnet.

Je nach Freigabestatus können Sie die meisten Attribute, die Testfallauswahl sowie die Anlagen und Testpakete ändern. Testplan-ID, Lösung, Branch, Umfang, Freigabeschema und ausführbare Einheiten können nicht mehr geändert werden, nachdem ein Testpaket oder eine Testsequenz für den Testplan angelegt wurde.

Wenn Änderungen nicht zulässig sind, können Sie nur den Freigabestatuswert ändern, um letztendlich die Änderungen wieder zu ermöglichen. Es ist nicht mehr möglich, Anlagen, Testpakete oder andere Daten zu ändern.

Ist der Testplanstatus als „Final“ gekennzeichnet, können keine Änderungen mehr vorgenommen werden und der Status kann nicht mehr geändert werden.

## Registerkarte

*Allgemeine Daten*

## Funktionen

### Achtung

Wenn Sie die Systemrolle eines Testplans ändern, kann dies bei einem bereits bearbeiteten Testplan zu Inkonsistenzen in den Testergebnissen führen, weil Sie nun auf Testfälle in anderen Systemen zugreifen. Wir empfehlen, die Systemrolle nur in sorgfältig durchdachten Ausnahmefällen zu wechseln.

Sie können anzeigen, wer den Testplan angelegt und zuletzt geändert hat, zusammen mit den zugehörigen Zeitangaben.

Sie können sehen, ob Änderungen in der Lösungsdokumentation auftreten, auf der der Testplan basiert.

- Strukturänderungen: Neue Knoten wurden hinzugefügt oder zum Testplan gehörende Knoten wurden entfernt.
- Testdokumentversion: Für die Lösung wurde eine neue Version hochgeladen.
- Es wurde eine neue Testdatenvariante für eine Testkonfiguration ausgewählt.

## Registerkarte

### Einstellungen

## Funktionen

Sie können beispielsweise die Dokumentart für Testnotizen bearbeiten oder den Freigabestatus ändern.

Sie können festlegen, auf welcher Ebene der erwartete Testaufwand verwaltet werden soll, und die Werte für den erwarteten Testaufwand anzeigen oder spezifizieren. Istdaten von Aufwänden (von Testern gemäß tatsächlichem Zeitaufwand erfasst) zeigt das System automatisch an. Weitere Informationen finden Sie unter [Testzeiten und -aufwände planen \[Seite 252\]](#).

Sie können Dokumente anhängen, die für die Testplanung und -ausführung relevant sind, z.B. MS-Excel-Tabellen mit Testdaten.

Das tatsächliche Enddatum Ihres Testplans können Sie am Ende der Testausführungsphase festlegen.

### Testfallauswahl

Sie können Testfälle auswählen oder verwerfen.

In der Spalte [Ändern](#) können Sie sehen, welche Testfälle und Knoten in der Lösung verglichen mit dem letzten im Testplan verwendeten Bild geändert wurden. Bei Änderungen an der Testdokumentversion oder der Testdatenvariante können Sie die Änderung im Testplan übernehmen. In diesem Fall wird der Teststatus für die entsprechenden Testfälle zurückgesetzt.

Statusinformationen für Testfälle oder ausführbare Einheiten, die bereits in den Testplan aufgenommen wurden, gehen verloren, wenn Sie die Auswahl des Testfalls aufheben und Ihren Testplan und Ihr Testpaket sichern.

## Bearbeiten mehrerer Testpläne

Über die Funktion für [Massenaktualisierung](#) können Sie mehrere Testpläne auswählen und eine Massenaktualisierung der zugehörigen Attribute ausführen.

Sie können in der Anwendung [Testplanverwaltung](#) auch mehrere Testpläne auswählen und dann [▶ Testplan ▶ Massenaktualisierung ▶](#) wählen.

In den Attributen für die [Massenaktualisierung von Testplänen](#) können Sie mehrere Parameter von mehreren Testplänen gleichzeitig aktualisieren. Dies gilt nur, wenn keine Signatur erforderlich ist.

## 11.4 Testpläne kopieren

## Kontext

Um Testpläne wiederzuverwenden, können Sie einen oder mehrere kopieren und angeben, in welchem Ausmaß kopiert werden soll, beispielsweise mit oder ohne Testpakete. Testsequenzen sind automatisch in der Kopie der Testpläne enthalten.

- Vorherigen Status der Lösung beibehalten: Ist diese Option nicht ausgewählt, werden die neuen Testpläne identisch mit den Quelltestplänen sein.
- Lösungsinhalt automatisch auf aktuellen Status aktualisieren: Ist diese Option ausgewählt, gibt der neue Testplan den aktuellen Status der Lösungsdokumentation wieder:
  - Knoten, die entfernt wurden, sind nicht Bestandteil der neuen Testpläne
  - In den neuen Testplänen werden die aktuellen Versionen von Testdokumenten verwendet
  - In den neuen Testplänen werden die aktuellen, für Testkonfigurationen angegebenen Testdatenvarianten verwendet

Sie können wählen, ob Sie alle Testfälle oder nur Testfälle mit dem Status „Offen“ kopieren möchten. Testfälle gelten als offen, wenn ihr Gesamtstatus nicht grün (OK) ist. Sie können Testfälle nach Bedarf auch in Sequenzen kopieren, um unvollständige Sequenzen zu vermeiden.

Beim Kopieren eines Testplans werden alle Informationen aus dem Quelltestplan in den neuen Testplan übertragen (Attribute und Testfallauswahl). Sie können einige Attribute für den Zieltestplan ändern:

- Sie müssen eine neue, eindeutige Testplan-ID definieren.
- Die Testplanbeschreibung wird aus dem Quelltestplan wiederverwendet, Sie können sie jedoch ändern.

Die folgenden weiteren Attribute können einmalig für alle Zieltestpläne definiert werden. Wenn Sie dies nicht tun, werden die Werte aus dem Quelltestplan wiederverwendet.

- Verantwortlicher
- Soll-Startdatum
- Soll-Enddatum
- Testmenge
- Testklassifizierung
- Freigabestatusschema  
Bei den Zieltestplänen wird der Freigabestatuswert automatisch auf den Initialstatus festgelegt, der für das Freigabestatusschema definiert wurde.
- Workflow

Darüber hinaus können Sie den Umfang der Kopie erweitern und folgende Informationen einschließen:

- Testplananhänge: Neue Dokumente werden durch Kopieren der Quelltestplan-Anhänge angelegt.
- Testdatensätze: Für die neuen Testpläne werden Verweise auf Testdatensätze angelegt
- Testpakete: Die Kopie legt automatisch Testpakete für den Zieltestplan an. Wenn Sie ein geplantes Start- und Endedatum definiert haben, werden diese auch für die Testpakete verwendet. Wenn die Testdatensätze ebenfalls ausgewählt wurden, wird der auf Testpaketebene referenzierte Testdatensatz auch auf den Zieltestpaketen referenziert.
- Beim Kopieren von Testpaketen können Sie die Zieltestpakete umbenennen und ihre Beschreibungen ändern.  
Sie können auch die zugeordneten Tester und ihre Anhänge kopieren.
- Falls die Testzuordnung kopiert wird, können Sie auch die Fehler kopieren. Standardmäßig werden nur offene Fehler kopiert, die einem aktiven Prozesstyp für das Testmanagement entsprechen. Sie können

diese Standardeinstellung ändern, indem Sie die Prozessarten und -status auswählen, die für die Kopie relevant sind.

- Beschädigt: Sie können Beschädigte in der Kopie einschließen. Standardmäßig werden nur offene Beschädigte kopiert, die einer aktiven Prozessart für das Testmanagement entsprechen. Sie können diese Standardeinstellung ändern, indem Sie die für die Kopie relevanten Prozessarten und Status auswählen. Wenn Sie auch Testpakete zum Kopieren markiert haben, werden zusätzlich zu dem auf Testplanebene gemeldeten Testpaket die auf Testpaketebene gemeldeten beschädigten Testpakete kopiert.

Wenn Sie Testpläne kopieren, wird die Anmeldesprache als Originalsprache für die Kopie verwendet.

## Vorgehensweise

1. Wählen Sie in der Anwendung *Testplanverwaltung* die zu kopierenden Testpläne.
2. Wählen Sie ► *Testplan* ► *Kopieren* ⌵.  
Sie gelangen zur Sicht *Kopie Massentestplan*.
3. Setzen Sie unter *Parameter kopieren* die entsprechenden Haken, um den Umfang für die Kopie festzulegen.
4. Mithilfe der Felder *Suchen:* und *Ersetzen durch:* können Sie die Spalte *Zieltestplan-ID* automatisch ausfüllen. Das System sucht nach den Werten in den ursprünglichen Testplan-IDs (nur letzte Zeichen) und ersetzt diese, um die neue Zieltestplan-ID zu erzeugen. Geben Sie die relevanten Werte in die *Suffix*-Felder ein und wählen *Anwenden*.
5. Wählen Sie *Kopieren*.

Bestätigen Sie die Frage im Dialogfenster, um die Kopie zu starten. Sie können die Attribute nicht mehr bearbeiten.

## 11.5 Freigabestatusschema

### Verwendung

Als Testorganisator können Sie Testobjekte wie z.B. Testpläne und Testpakete sperren oder freigeben, sodass sie von Testern nicht geändert oder ausgeführt werden können. Das Freigabestatusschema definiert ein Reihe von Freigabestatuswerten, die Sie im Testplan zuordnen können.

Sie können einen [Workflow \[Seite 237\]](#) als Freigabestatusschema verwenden. Damit können Sie die Testausführung noch effektiver steuern und überwachen. Beispielsweise informiert das System die Tester, wann die ihnen zugeordneten Testpakete zum Test bereitstehen.



## Voraussetzungen

- Sie haben im Customizing für SAP Solution Manager unter [▶ Test-Suite](#) [▶ Test-Suite-Vorbereitung](#) [▶ Spezielle Konfiguration](#) [▶ Freigabestatusschema](#) [▶ Freigabestatusschema einrichten](#) [▶](#) Ihre Freigabestatusschemata definiert und verwenden die Freigabestatuswerte Ihres Statusschemas.
- Sie haben im Customizing für SAP Solution Manager unter [▶ Test-Suite](#) [▶ Test-Suite-Vorbereitung](#) [▶ Spezielle Konfiguration](#) [▶ Freigabestatusschema](#) [▶ Freigabestatusschema einrichten](#) [▶](#) Ihren Anforderungen entsprechend benutzerdefinierte Freigabestatuswerte definiert.

## Funktionsumfang

Abhängig von den Einstellungen der Freigabestatuswerte können Sie die Freigabe von Testplänen und Testpaketen wie folgt beeinflussen:

- **Auswählbare Freigabestatuswerte:** Jedem Freigabestatuswert ist eine Ordnungsnummer zugeordnet. Die Ordnungsnummern legen die Reihenfolge der Freigabestatuswerte in dem Freigabestatusschema fest. Über die höchste und die niedrigste Ordnungsnummer, die jedem Freigabestatuswert zusätzlich zugeordnet ist, ermittelt das System die in den Testplanattributen auswählbaren Freigabestatuswerte und zeigt diese in der Eingabehilfe an.
- **Final:** der auf „Final“ gesetzte Statuswert ist der letzte Status im Testphasenzyklus. Er lässt weder zusätzliche Änderungen am Testplan noch deren Ausführung zu. Ist dieser Status für den Testplan gesetzt, kann er nicht mehr rückgängig gemacht werden. Testpläne mit finalelem Status werden nicht standardmäßig in den Testplanlisten und Analyse-Reports angezeigt (können jedoch bei Bedarf aufgerufen werden). Außerdem haben Änderungen an ihrer Quelllösung keine Auswirkungen auf sie, und sie verhindern nicht das Löschen ihres Quell-Branch.
- **Änderungssperre:** Wenn ein auswählbarer Freigabestatuswert in den Testplanattributen für Änderungen gesperrt ist, können Attribute und Struktur des Testplans sowie die zugehörigen Anlagen, Testsequenzen und Testpakete nicht geändert werden.
- **Ausführungssperre:** Wenn ein auswählbarer Freigabestatuswert in den Testplanattributen für die Ausführung gesperrt ist, dann kann ein Tester die ihm zugeordneten Tests nicht ausführen und den Teststatus nicht ändern.
- **Signaturstrategie:** Eine Kombination aus Einzelsignaturen, die zur Unterzeichnung eines Vorgangs erforderlich sind. Sie gibt Folgendes an:
  - Erforderliche Einzelsignaturen, d.h. welche Benutzergruppen den Vorgang unterzeichnen müssen
  - Reihenfolge, in der die Einzelsignaturen geleistet werden müssen
  - Signatur bzw. Signaturen, mit der bzw. denen der Signaturprozess abgeschlossen werden kann
  - Verwendete Signaturmethode, z.B. Benutzersignatur oder Systemsignatur mit Autorisierung durch Benutzer-ID und Kennwort

Wenn dem in den Testplanattributen gesetzten Freigabestatuswert eine Signaturstrategie zugeordnet ist, dann müssen zuerst Personen unterzeichnen, um die Signaturstrategie zu erfüllen. Wenn die Signaturstrategie erfüllt ist, dann setzt das System automatisch den zugeordneten Endstatuswert. Wenn die Signaturstrategie abgebrochen wird, setzt das System automatisch den Abbruchstatuswert. Sie können Signaturstrategien den Freigabestatuswerten zuordnen. Dies ist nur bei den Status mit definiertem End- und Abbruchstatuswert sinnvoll, d.h. im Standard-Freigabestatusschema bei den Review-Status.

Standardmäßig können Sie eines der folgenden Freigabestatusschemata verwenden:

**Default-Freigabestatusschema (DEFAULT)**

Statusordnungs-nr.	Freigabe-status	Initial-status*	Final		Höchste Statusordnungs-nr.	Niedrigste Statusordnungs-nr.	Änderung ver-boten*	Ausfüh-rung ver-boten*	Signa-turstra-tegie**	Endsta-tus der Signa-tur**	Ab-bruch-status der Sig-natur**
10	Test in Vorbereitung	X	-	-	20	20	-	X	-	-	-
20	Freigegeben für Test	-	-	-	30	40	X	-	-	-	-
30	Geschützt	-	-	-	50	50	X	X	-	-	-
40	Test in Anpassung	-	-	-	20	20	-	-	-	-	-
50	Test in Anpassung	-	-	-	20	20	-	X	-	-	-

**einfachstes Schema (SIMPLEST)**

Statusordnungs-nr.	Freigabe-status	Initial-status*	Final		Höchste Statusordnungs-nr.	Niedrigste Statusordnungs-nr.	Änderungs-sperre*	Ausfüh-rungs-sperre*	Signa-turstra-tegie**	Endsta-tus der Signa-tur**	Ab-bruch-status der Sig-natur**
10	Freigegeben für Test	X	-	-	20	20	-	-	-	-	-
20	Geschützt	-	-	-	10	10	-	-	-	-	-

**Detailliertes Freigabeschema (DETAILED)**

Statu- sord- nungsnr.	Freigabe- status	Initial- status*	Final		Höchste Statu- sord- nungsnr.	Niedrig- ste Sta- tusord- nungsnr.	Ände- rung ver- boten*	Ausfüh- rung ver- boten*	Signa- turstra- tegie**	Endsta- tus der Signa- tur**	Ab- bruch- status der Sig- natur**
10	Test in Vorberei- tung	X	-	-	20	20	-	X	-	-	-
20	Freigege- ben für Test	-	-	-	30	60	X	-	-	-	-
30	Ge- schützt	-	-	-	50	60	X	X	-	-	-
40	Review zur An- passung (drin- gend)	-	-	-	20	20	-	-	-	-	-
50	Test in Anpas- sung	-	-	-	20	20	-	X	-	-	-
60	Abge- schlos- sen	-	X	-	60	60	X	X	-	-	-

\* x = Ankreuzfeld ist markiert, - = Ankreuzfeld ist nicht markiert

\*\* - = Eingabefeld ist leer

## Aktivitäten

Sie können das Freigabestatusschema beim Anlegen eines neuen Testplans auf der Registerkarte „Einstellungen“ des Testplaneditors zuordnen. Weitere Informationen finden Sie unter [Testpläne anlegen und bearbeiten \[Seite 239\]](#). Hier können Sie auch einen der unterstützten Freigabestatus zuordnen.

### i Hinweis

Sie können in den Testplanattributen ein Freigabestatusschema auswählen, solange ein Testplan den Status *Neu* aufweist. Dies ist der Fall, wenn der Testplan noch keinen Teststatus hat und wenn er kein Statusschema hat oder wenn er ein Statusschema hat und der Freigabestatus noch *NEW* lautet.

Wenn dem Freigabestatuswert eine Signaturstrategie zugeordnet ist, stößt das System automatisch die Signaturstrategie an.

Wenn für den aktuellen Testplan-Freigabestatus eine Ausführungssperre aktiviert ist, zeigt das System die Sperre auch im Arbeitsvorrat des Testers am entsprechenden Testpaket an.

Sie können nun den Freigabestatuswert erneut ändern und jederzeit die Daten in der Änderungshistorie überprüfen.

## 11.6 Planen von Testzeitplänen und -aufwänden

### Verwendung

Als Testkoordinator können Sie den Testzeitrahmen einplanen und die Testplanung abhängig vom erwarteten Aufwand durchführen.

#### i Hinweis

Für die Planungsebene haben Sie folgende Optionen, die sich in Aufwand und der Planungsgenauigkeit unterscheiden:

- Sie planen auf **Testfallebene** und können damit relativ genau den erwarteten Aufwand summieren.
- Sie planen auf **Testpaketebene** und können damit ihre Zeitaufwände grob abschätzen.
- Sie planen auf **Testplanebene**. Damit ist nur eine sehr grobe Schätzung möglich, bei der Sie ausgehend von der Anzahl der Testfälle einen erwarteten Gesamtaufwand angeben.

### Voraussetzungen

- Geben Sie für eine genaue testfallspezifische Auswertung im Vorfeld in den Testfallattributen des jeweiligen Testfalls den erwarteten Testaufwand ein.  
Hier können Sie den tatsächlichen Aufwand oder die aggregierten Daten aus Testpaketen ermitteln.
- Andernfalls können Sie eine der folgenden Alternativen nutzen:
  - Planung mit Standardwerten: Standardwerte können Sie im Customizing (Transaktion `SPRO`) für SAP Solution Manager unter [Test-Suite](#) > [Test-Suite-Vorbereitung](#) > [Spezielle Konfiguration](#) > [Zentrale Einstellungen](#) > [Standardausführungszeit angeben](#) definieren.
  - Geben Sie eine grobe Aufwandsschätzung pro Testplan oder Testpaket (in den Testpaketattributen) ein.
- Um einen Vergleich des geplanten und tatsächlichen Aufwands durchführen zu können, muss jeder Tester bei der Testausführung den tatsächlichen Zeitaufwand erfassen.

### Vorgehensweise

#### Planen des Testzeitrahmens

1. Öffnen Sie in der Anwendung *Testplanverwaltung* einen Testplan für die Bearbeitung.
2. Geben Sie auf der Registerkarte *Einstellungen* auf der linken Seite des Bilds Folgendes ein:
  - Die Ebene (Testplan, Testpaket oder Testfall), auf der Sie den erwarteten Aufwand festlegen möchten. Bei den Ebenen *Testpaket* oder *Testfall* können Sie den Aufwand vom System berechnen lassen.
  - Start- und Enddatum für den erwarteten Aufwand

### **i** Hinweis

Unabhängig von der Ebene, auf der Sie planen, müssen Sie ein geplantes Startdatum und ein geplantes Enddatum festlegen. Sie können den geschätzten Gesamtaufwand im Feld *Geplanter Aufwand* eingeben.

Das System zeigt die aktuellen Daten auf der rechten Bildseite an.

## 11.7 Testpakete anlegen und bearbeiten

### Verwendung

Testfälle werden für die Tester in Testpaketen zusammengefasst. Ein Testpaket basiert auf einem Testplan. Testfälle werden aus dem Testplan für das Testpaket ausgewählt. Wenn Sie einen Testplan für einen Prozess in SAP Solution Manager angelegt haben, können Sie auch Testfälle und ausführbare Elemente aus dem Testplan auswählen, die für ein Testpaket getestet werden sollen.

### Aktivitäten

#### Testpakete anlegen

1. Wählen Sie in der Anwendung *Testplanverwaltung* einen Testplan.
2. Unter *Testpaketliste* wählen Sie ► *Testpaket* ► *Anlegen* ⌵.  
Die Anwendung zur Testpaketverwaltung wird geöffnet.
3. Geben Sie die erforderlichen Daten ein.
  - Füllen Sie die Mussfelder auf der Registerkarte *Allgemeine Daten*:
    - *Testpaket-ID*: Dies ist die Kennung für Ihr Testpaket. Sie muss innerhalb Ihres Testplans einmalig sein.
    - Beschreibung
    - Verantwortlicher
    - Priorität

Die anderen Informationen auf dieser Registerkarte, mit Ausnahme der Erstellungs- und Änderungsinformationen, werden aus dem Testplan vererbt und sind schreibgeschützt.

  - Auf der Registerkarte *Einstellungen* geben Sie das *Geplante Startdatum* und das *Geplante Enddatum* für das Testpaket an. Diese sollten in den Zeitrahmen passen, der für den Testplan definiert wurde. Termine für das Testpaket werden standardmäßig mit den Terminen belegt, die für den Testplan definiert wurden.

Wenn Sie eine Planungsebene für das Testpaket definieren, können Sie den Gesamtressourcenbedarf unter *Planaufwand* erfassen.

Sie können Anlagen hinzufügen, um den Testprozess oder -kontext zu dokumentieren. Die Anlagen, die Sie für den Testplan angelegt haben, können hier ebenfalls verwendet werden.

- Wählen Sie auf der Registerkarte *Testfallauswahl* die Testfälle, die für das Testpaket relevant sind. Weitere Informationen finden Sie unter *Testpläne anlegen und bearbeiten* [Seite 239].
- Auf der Registerkarte *Testdatensatz* können Sie Ihrem Testpaket Testdatensätze zuordnen. Dies sind Verweise auf Testdatencontainer. Der auf Testplanebene zugeordnete Testdatensatz kann ausgewählt werden, so dass er im Testpaket sichtbar ist. Nachdem der Testdatensatz einem Testpaket zugeordnet wurde, ist er in der Anwendung „Manuelle Testausführung“ sichtbar und der Tester kann direkt darauf zugreifen.
- Ordnen Sie auf der Registerkarte *Tester* die Personen zu, die die Testfälle ausführen. Jeder Tester muss in SAP Solution Manager als Geschäftspartner registriert sein. Ihre technischen Benutzer und die Systeme, auf denen sie arbeiten dürfen, müssen entsprechend in ihrem Geschäftspartnerprofil definiert sein. Sie können ein zugeordnetes Testpaket vorübergehend vor einem Tester verbergen, indem Sie die Benutzerzuordnung sperren. Der Tester kann dann das Testpaket im Testerarbeitsvorrat nicht sehen. Diese Sperre ist benutzerspezifisch und unabhängig vom Testplan-Statuswert.

4. Sichern Sie Ihre Angaben, und schließen Sie das Fenster.

Testpakete können **abhängig von den vorhandenen Testsequenzen** angelegt werden. Weitere Einzelheiten finden Sie unter *Testpläne anlegen und bearbeiten* [Seite 239].

### Massenerstellung von Testpaketen

1. Wählen Sie in der Anwendung *Testplanverwaltung* einen Testplan aus.
2. Unter *Testpaketliste* wählen Sie ► *Testpaket* ► *Massenerstellung* ⌵. Geben Sie die Anzahl der Testpakete an, die Sie anlegen möchten, und wählen Sie *Anlegen*. Die Anwendung zur Verwaltung der Massenerstellung von Testpaketen wird geöffnet.
3. Die ID und die Beschreibung der Testpakete werden automatisch festgelegt. Die Testpaket-ID ist wie folgt aufgebaut: Testplan-ID + „\_“ + Nummerierung ab dem ersten verfügbaren Wert. Als Beschreibung des Testpakets wird die Testpaket-ID verwendet.
4. Die Standardeinstellungen sind wie folgt: Der Geschäftspartner des aktuellen Benutzers wird als Verantwortlicher für Testpakete festgelegt, die Priorität wird auf *Mittel* gesetzt, und das Plandatum für Testpakete wird aus den im Testplan gepflegten Plandaten abgerufen.
5. Wenn als Planungsebene des Testplans „Testpaket“ festgelegt ist, bietet die Anwendung Felder für den geplanten Aufwand und die Einheit des geplanten Aufwands an.
6. Sie können Felder direkt aktualisieren oder über die Funktion *Attribute setzen* mehrere Zeilen gleichzeitig ändern.
7. Wählen Sie *Sichern*, um die Erstellung der entsprechenden Testpakete anzustoßen.

Anschließend können Sie die einzelnen Testpakete bearbeiten, um Testfälle auszuwählen und Tester zuzuordnen.

### Testpakete bearbeiten

Als Testkoordinator haben Sie die Möglichkeit, den Verantwortlichen oder die Priorität der Pakete zu ändern.

#### **i** Hinweis

Nachdem die Testausführung gestartet wurde, geht die Status- und Fehlerzuordnung verloren, wenn Sie den Tester löschen oder die Auswahl eines Testfalls aufheben.

Diese Informationen werden wiederhergestellt, wenn Sie den Tester hinzufügen oder den Testfall erneut auswählen.

1. Wählen Sie in der Anwendung *Testplanverwaltung* den entsprechenden Testplan.
2. Wählen Sie im Bereich *Testpakete* ein Testpaket, und wählen Sie ► *Testpaket* ► *Bearbeiten* ►.
3. Nehmen Sie im Editor für Testpakete Änderungen vor, und sichern Sie Ihre Einträge.

#### Massenaktualisierung von Testpaketen

Als Testkoordinator können Sie die wichtigsten Attribute mehrerer Testpakete gleichzeitig aktualisieren, um beispielsweise einen neuen Verantwortlichen zuzuordnen oder die Priorität zu ändern. Wählen Sie dazu

► *Testpaket* ► *Massenaktualisierung* ►.

## 11.8 Kopieren von Testpaketen und Testsequenzen

### Kontext

Zusätzlich zum Kopieren von Testpaketen und Testsequenzen beim Kopieren eines gesamten Testplans können Sie diese auch einzeln kopieren, entweder innerhalb eines Testplans oder in einen anderen vorhandenen Testplan.

### Kopieren von Testpaketen/-sequenzen innerhalb eines Testplans

1. Wählen Sie in der Anwendung *Testplanverwaltung* einen Testplan.
2. Wählen Sie unter *Liste der Testpakete* oder *Testsequenzliste* ein Paket oder eine Sequenz.
3. Wählen Sie ► *Testpaket/Testsequenz* ► *Kopieren* ►.
4. Geben Sie unter Verwendung der verschiedenen Optionen an, welche Daten Sie in die neue Version einbeziehen möchten. Bestätigen Sie Ihre Auswahl mit *Kopieren*.

### Kopieren von Testpaketen/-sequenzen in einen anderen Testplan

1. Wählen Sie in der Anwendung *Testplanverwaltung* einen Testplan.
2. Wählen Sie unter *Liste der Testpakete* oder *Testsequenzliste* ein Paket oder eine Sequenz.
3. Wählen Sie ► *Testpaket/Testsequenz* ► *In anderen Testplan kopieren* ►.
4. Geben Sie unter Verwendung der verschiedenen Optionen an, welche Daten Sie in die neue Version einbeziehen möchten. Bestätigen Sie Ihre Auswahl mit *Kopieren*.

## 11.9 Zuordnen von Testern

### Verwendung

Ein Testpaket wird in der Regel einem Tester zugeordnet, der es dann aus seinem Arbeitsvorrat in der App *Meine Aufgaben - Tester-Arbeitsvorrat* aufrufen kann. Sie können demselben Testpaket jedoch auch mehrere Tester zuordnen. Beispiel: Wenn Sie einen Testfall zweimal testen (also zwei voneinander unabhängige Tests durchführen) möchten, können Sie demselben Testpaket zwei Tester zuordnen und beide bitten, alle Testfälle zu testen. Sie müssten dann nicht dasselbe Testpaket zweimal anlegen. Da der Status eines Testfalls vom Tester abhängig ist, wechselt der Fall nur dann zum Status OK, wenn beide Tester diesen Status setzen (Regel „Worst Wins“ – der schlechteste Status gilt). Wenn einer der Tester mit dem Test nicht beginnt, wird sein Status ignoriert.

#### i Hinweis

Bei automatischen Testfällen, die im Hintergrund ausgeführt werden, entfällt dieses Vier-Augen-Prinzip, da aufgrund der Aggregationsregel „Last Wins“ (der zuletzt gesetzte Status gilt) jeder Tester das gleiche Ergebnis erhalten würde. Daher ist die Zuordnung mehrerer Tester zum selben automatischen Testfall nicht sinnvoll.

Die Auswertung des Gesamtstatus des Testfalls berücksichtigt die einzelnen von den Testern gesetzten Testfallstatus. Wenn dann mehrere Tester demselben Testpaket zugeordnet werden, muss jeder Tester einen Status für jeden Testfall setzen. Wenn derselbe Testfall zu unterschiedlichen Testpaketen gehört, wird der Gesamtstatus des Testplans basierend auf der Aggregationsregel ausgewertet, die auf Testplanebene definiert wurde. Diese Regeln werden in der Anwendung *Konfiguration - Test-Suite* für SAP Solution Manager (Transaktion SPRO) definiert: ► [Test-Suite](#) ► [Test-Suite-Vorbereitung](#) ► [Spezielle Konfiguration](#) ► [Zentrale Einstellungen](#) ► [Testplanauswertung anpassen](#) ►

### Voraussetzungen

- Alle, die am Test beteiligt sind, werden als Geschäftspartner registriert, und ein Systembenutzer wurde ihren Geschäftspartnern zugeordnet. Der Geschäftspartner muss die Rolle *Mitarbeiter* haben.
- Wenn Sie zusätzlich den [Workflow \[Seite 237\]](#) nutzen möchten, muss für den Geschäftspartner eine gültige E-Mail-Adresse hinterlegt sein.

### Aktivitäten

#### i Hinweis

Beachten Sie, dass die folgende Vorgehensweise für Standardtestpakete gilt. Wenn ein Testpaket zu einer vordefinierten Testsequenz (Testsequenz als Referenz) gehört, werden Änderungen an der Testsequenz auch in das Testpaket übernommen. Weitere Informationen finden Sie unter [Testpläne anlegen und bearbeiten \[Seite 239\]](#).



Sie können in der Funktion [Testpaketverwaltung](#) Tester zuordnen, löschen, ersetzen, sperren und freigeben. Stattdessen können Sie diese Aktionen auch als Massenänderungen bei mehreren Testern in der App [Tester zuordnen](#) ausführen.

Um diese Aktionen für einen Tester gleichzeitig auszuführen, gehen Sie wie folgt vor:

1. Wählen Sie in der App [Testplanverwaltung](#) einen Testplan.
2. Wählen Sie ein Testpaket, und wählen Sie dann [▶ Testpaket ▶ Bearbeiten ▶](#).
3. Auf der Registerkarte [Tester](#) haben Sie folgende Möglichkeiten:
  - **Zuordnung Testpaket zu Tester** – wählen Sie [Tester zuordnen](#), um einen neuen Tester auszuwählen. Der Tester wird in die Liste der Tester aufgenommen. Wenn Sie einen Tester aus der Liste auswählen, sehen Sie alle Testfälle aus dem Testpaket, die dem Tester automatisch zugeordnet werden.
  - **Ersetzen der Testerzuordnung in Testpaketen** – Wählen Sie den zu ersetzenden Tester und dann [Tester zuordnen](#) aus. Wählen Sie dann einen neuen Tester aus der Liste aus. Der neue Tester wird in die Liste der Tester aufgenommen. Der vorherige Tester wird aus der Liste entfernt und sämtliche Testausführungsdaten dieses Testers werden dem neuen Tester erneut zugeordnet.
  - **Entfernen der Testpaketzuordnung für einen Tester** – Wählen Sie den Tester aus, den Sie sperren möchten, und wählen Sie [Zuordnung sperren \(entsperren\)](#). Sie können einen Tester auch sperren und entsperren, indem Sie das Sperrsymbol in der Spalte [Zuordnung](#) wählen. Um alle Testerzuordnungen zu sperren oder entsperren, wählen Sie der Anwendung [Testplanverwaltung](#) ein Testpaket und dann [▶ Zuordnung ▶ Sperren ▶](#) oder [Freigeben](#) aus. Wenn alle Tester gesperrt wurden, wird das Symbol in der Paketliste als Sperrsymbol angezeigt. Wenn nur ausgewählte zugeordnete Tester gesperrt wurden, wird beim Benutzer ein Sperrsymbol angezeigt.

#### Hinweis

Wenn Sie Testpakete aus dem Arbeitsvorrat der Tester entfernen möchten, **ohne die Testerzuordnung zu verlieren**, können Sie die relevanten Zuordnungen sperren. Dies ist sinnvoll, wenn ein Test abgeschlossen ist und der Tester den Status nicht mehr ändern soll.

- **Löschen der Testpaketzuordnung für einen Tester** – Wählen Sie den Tester aus, den Sie löschen möchten, und wählen Sie [Zuordnung aufheben](#). Sie können die Zuordnung eines Testers zu einem Testpaket einschließlich der bisher gesammelten Testdaten (Teststatus, Testergebnisse und Link zu Fehlern für einen Tester) löschen.

#### Achtung

Löschen Sie die Zuordnung eines Testers zu einem Testpaket nicht, wenn es bereits verarbeitet wurde. Ansonsten gehen sämtliche gesammelten Daten verloren.

- **Zuordnen von Testfällen zu einem Tester** – Sobald ein Tester einem Testpaket zugeordnet wurde, ist er automatisch allen Testfällen im Testpaket zugeordnet. Der Tester wird außerdem automatisch allen Testfällen zugeordnet, die dem Testpaket nachträglich hinzugefügt werden.

4. Sichern Sie Ihre Angaben, und schließen Sie das Fenster.

Um diese Aktionen als Massenänderungen durchzuführen, gehen Sie wie folgt vor:

1. Wählen Sie in der App [Tester zuordnen](#) einen Testplan.
2. Wählen Sie mindestens ein Testpaket und dann eine der folgenden Aktionen aus:
  - [Tester zuordnen](#) – Suchen Sie im neuen Fenster nach dem Tester, wählen Sie ihn aus, und bestätigen Sie die Auswahl. Der neue Tester wird in der Spalte [Neuer Testername](#) grün hervorgehoben.

## i Hinweis

Gesperrte Geschäftspartner werden in den Suchergebnissen nicht angezeigt.

- *Tester ersetzen* – Geben Sie den neuen Tester an, indem Sie den Namen oder Geschäftspartner eingeben. Legen Sie dann fest, ob alle oder nur bestimmte Tester ersetzt werden sollen.
  - *Zuordnung des Testers aufheben* – Die betroffenen Tester werden rot markiert.
  - *Massenoperation* – Im neuen Fenster werden alle den ausgewählten Testpaketen zugeordneten Tester angezeigt. Wählen Sie Tester aus, und füllen Sie die Spalten der Tabelle, um mehrere Tester gleichzeitig zuzuordnen, zu ersetzen oder zu entfernen.
  - *Zuordnung sperren (entsperren)* – siehe Spalte *Zuordnungssperre*. Änderungen werden gelb hervorgehoben.
3. Sichern Sie Ihre Änderungen.

## 11.9.1 Zuordnen von Tester-Teams

Sie können Geschäftspartner zum Repräsentieren eines Tester-Teams verwenden und dieses Team zu Testpaketen zuordnen.

Es gelten die folgenden Voraussetzungen:

- Der Geschäftspartner ist vom Typ *Gruppe* oder *Organisation*.
- Für den Geschäftspartner wurde entweder eine *Beinhaltet die Tester*- oder eine *Ist Tester von*-Beziehung definiert.
- Wenn Sie die Workflow-Funktion verwenden, ist eine gültige E-Mail-Adresse auf Teamebene definiert.
- Da die App „Testausführung“ nur für Geschäftspartner vom Typ *Person* zur Verfügung steht, muss mindestens ein Mitglied des Teams die Geschäftspartnerzuordnung *Person* haben.

Sie können dieses Team Testpaketen zuordnen, anstatt einzelne Tester zuzuordnen. Wenn ein Team einem Testpaket (oder einem Testfall in einem Testpaket mit einer Reihenfolge) zugeordnet wird, teilen alle dem Team zugeordneten Personen dieselben eindeutigen Testausführungsdaten und greifen auf diese zu. Sie können auch eine *Ist Tester für*-Beziehung zwischen diesem Team und einem anderen Team anlegen.

Weitere Informationen zum Anlegen und Bearbeiten von Geschäftspartnern finden Sie unter [Geschäftspartner konfigurieren](#)

## 11.9.2 Vertreter zuordnen

Sie können einen stellvertretenden Tester zuordnen, der bestimmte Tester ersetzt, wenn diese nicht verfügbar sind, z.B. weil sie im Urlaub oder auf Geschäftsreise sind. Außerdem können Sie sich selbst als Vertretung für einen oder mehrere Kollegen zuordnen.

Um einem Tester einen Stellvertreter zuzuordnen, definieren Sie entweder eine *Wird ersetzt durch*- oder eine *Ersetzt*-Beziehung zwischen den jeweiligen Geschäftspartnern im System.

Im Tester-Arbeitsvorrat sieht der Stellvertreter alle den Testern zugeordneten Testpakete, für die er als Stellvertreter bestimmt wurde. Während des Vertretungszeitraums erhält der Stellvertreter auch Kopien der an die ursprünglichen Tester gesendeten Benachrichtigungs-E-Mails.

Weitere Informationen zur Definition von Geschäftspartnerbeziehungen finden Sie unter [Geschäftspartner konfigurieren](#)

# 12 Meine Aufgaben - Tester-Arbeitsvorrat

## Verwendung

Die Anwendung *Meine Aufgaben - Tester-Arbeitsvorrat* enthält eine Übersicht über alle Testpakete, die dem aktuellen Benutzer zugeordnet wurden. Hier beginnen die Tester tatsächlich mit der Ausführung ihrer Tests. In der Liste werden auch die Testpakete angezeigt, die einem Tester-Team zugeordnet wurden, zu dem der aktuelle Benutzer gehört. Sie können die Liste filtern, um nur die Testpakete anzuzeigen, die Ihnen oder einem Ihrer Teams zugeordnet sind.

Die folgenden Queries sind unter *Meine Pakete* verfügbar:

- *Freigegeben*: Zeigt die Testpakete, denen Sie zugeordnet sind und bei denen der Freigabestatus des Testplans die Ausführung zulässt.
- *Zugeordnet*: Zeigt die Testpakete, denen Sie zugeordnet sind, unabhängig vom aktuellen Testplanstatus.
- *Geschützt*: Zeigt die Testpakete, denen Sie zugeordnet sind und bei denen der Freigabestatus des Testplans die Ausführung nicht zulässt.
- *Bereit zum Testen*: Zeigt die Testpakete, denen Sie zugeordnet sind und bei denen ein testbereiter Testfall vorhanden ist.

Wenn die Zuordnung zu einem Testpaket zurzeit **gesperrt** ist, beachten Sie, dass das Testpaket dem zugeordneten Tester **nicht angezeigt** wird. Es wird wieder angezeigt, wenn der Test Organizer die Sperre der Zuordnung aufhebt. Diese Funktion kann verwendet werden, um z.B. die Ausführung der Tests vorübergehend anzuhalten oder nur einen Tester zu sperren, ohne die Ausführung des gesamten Testplans zu sperren.

Als Tester können Sie folgende Objekte im Arbeitsvorrat anzeigen:

- das Testpaket, dem Sie zugeordnet sind
- die Testpakete, denen Ihr Team zugeordnet ist
- die Testpakete, denen der Tester zugeordnet ist, den Sie ersetzen

Sie können *Tester-ID* wählen, um nur die Testpakete anzuzeigen, die dieser spezifischen Tester-ID zugeordnet sind.

Wenn Sie die Rolle „Test Organizer“ haben, haben Sie keinen Zugriff auf den Tester-Arbeitsvorrat. Sie können jedoch weiterhin alle Testpakete in der Anwendung *Testplanverwaltung* [Seite 235] anzeigen. Weitere Informationen dazu finden Sie unter *Tester zuordnen* [Seite 256].

## Voraussetzungen

- Dem aktuell angemeldeten Benutzer wurden Testpakete zugeordnet.

Um einen Test durchzuführen, müssen Sie folgende Voraussetzungen erfüllen:

- Sie sind dem Testfall als Tester zugeordnet. In Testsequenzen werden Ihnen auch die Testfälle angezeigt, die anderen Testern zugeordnet sind.
- Der Freigabestatus des Testplans lässt die Testausführung zu.

- Der Test ist im Kontext einer Testsequenz testbereit. Das heißt, die vorhergehenden Testfälle wurden ausgeführt, und der Status *Test beendet* wurde für alle Testfälle mit Ausnahme des aktuellen Testfalls festgelegt.

## Funktionsumfang

In dieser Anwendung können Sie folgende Aktionen ausführen:

- Auswahl eines Testpakets zur Anzeige der Testfälle im Bereich *Details*, das Sie mit *Ausführen* ausführen können.
- Zugriff auf die Testfallhierarchie eines gegebenen Testpakets, um den Test auszuführen. Die in dieser Anwendung verfügbaren Funktionen sind identisch mit denen, auf die Sie im Detailbereich des Tester-Arbeitsvorrats zugreifen können. Wechsel von der Listenansicht zur hierarchischen Ansicht, um die Abhängigkeiten zwischen Testpaketen und Testfällen zu sehen.
- Navigation zu den Anlagen eines Testpakets, um Informationen zum Testprozess zu erhalten.
- Wählen Sie *Automatischer Test*, um automatische Tests für das gesamte Testpaket einzuplanen:
  - Planen Sie automatische Tests im Vordergrund oder im Hintergrund ein. (Alle Testpakete, die nicht Teil einer Testsequenz sind, werden zur Ausführung im selben Job eingeplant. Allerdings plant das System einen einzelnen Job für jedes Testpaket ein, das zu einer Testsequenz gehört. Dies bedeutet, dass das System den Job für eine einzelne Testsequenz stoppen kann, wenn einer der Schritte fehlschlägt oder nicht ausgeführt werden kann.)
  - Auswahl automatischer Tests für die direkte Ausführung
- Wählen von *Springen*, um zur Jobübersicht zu navigieren, um alle Jobs anzuzeigen, die für die Ausführung automatischer Tests eingeplant sind
- Wählen von *Exportieren*, um eine Liste der Testpakete in ein Excel-Arbeitsblatt zu exportieren

Die Testfälle, die dem Testpaket zugeordnet sind, werden im unteren Bildbereich angezeigt.

### i Hinweis

Die Priorität der einzelnen Tests wird im Tester-Arbeitsvorrat angezeigt. Damit können Sie leichter entscheiden, welche Tests zuerst auszuführen sind. Außerdem lässt sich der Arbeitsvorrat nach Priorität filtern.

Sie können folgende Aktionen ausführen:

- Ausführen eines Tests:
  - Direkter Start der Ausführung eines manuellen Testfalls (relevant für Testdokumente und ausführbare Einheiten)
  - Navigieren zu den Optionen für den automatischen Teststart, um einen automatischen Test zu starten
- Ändern des Status eines Testfalls (Massenaktualisierung für mehrere Testfälle ist möglich)
- Anlegen eines Fehlers (Anwendungsfehler) oder Zuordnen eines vorhandenen Fehlers
- Melden eines Testfallfehlers oder Zuordnen eines vorhandenen Fehlers
- Anlegen eines Hinweises oder Verwalten eines vorhandenen Hinweises. Hinweise können aus mehreren Quellen angelegt werden:
  - Aus einer Vorlage, wenn eine Vorlage für Hinweise und Testergebnisse für die Dokumentenart definiert und dem Testplan zugeordnet wurde
  - Aus einem Testdokument, wenn der ausgewählte Testfall ein Testdokument ist

- Durch Hochladen eines lokalen Dokuments als Hinweis
- Aus einer URL
- Auswählen der Testfall-ID, um den Testfall zu öffnen. Abhängig vom Testfalltyp können Sie folgende Aktionen ausführen:
  - Testdokument im Anzeigemodus öffnen
  - Umgebung für die Testzusammenstellung für einen automatischen Test öffnen
  - Ausführbare Einheit im Zielsystem öffnen

Unten im Bild können Sie die Attribute des ausgewählten Testfalls und die Liste der zugeordneten Fehler sehen. In der Liste können Sie verschiedene Aktionen ausführen, z.B. Zuordnung der Fehler aufheben, wenn notwendig. Beachten Sie, dass der Fehler selbst dabei nicht gelöscht wird.

Der *Ausführungsstatus* ist testerspezifisch. Der Ausführungsstatus, der in der Liste der Testfälle angezeigt wird, wird daher vom aktuellen Tester gesetzt.

Der *Gesamtstatus* wird basierend auf einer Aggregation aller Status ausgewertet, die von den Testern festgelegt werden, die demselben Testpaket zugeordnet sind. Die Regeln sind im Customizing für SAP Solution Manager (Transaktion SPRO) festgelegt: [▶ Test-Suite](#) [▶ Test-Suite-Vorbereitung](#) [▶ Spezielle Konfiguration](#) [▶ Zentrale Einstellungen](#) [▶ Testplanauswertung anpassen](#) [▶](#)

Hinweise hängen vom Testfall und Testpaket ab. Alle Tester, die dem Testpaket zugeordnet sind, können auf dasselbe Dokument zugreifen und es bearbeiten. Bei Testsequenzen hängt der Hinweis vom Tester ab, da auf der Testfallebene nur ein Tester zugeordnet wird.

Testergebnisse und -fehler hängen vom Tester ab. Jeder Tester ordnet seine eigenen Fehler zu und dokumentiert die Ergebnisse seiner eigenen Tests.

Die Zahlen in der Liste der Testpakete und die Zahlen auf der Testfallebene beziehen sich auf den spezifischen Tester, mit Ausnahme des Gesamtstatus und der Änderungsinformationen. Auf den Gesamtstatus bezogene Ergebnisse können auf Testpaketebene hinzugefügt werden.

## 12.1 Manuelle Testfälle ausführen

### Verwendung

Beim Ausführen manueller Testfälle haben Sie folgende Möglichkeiten:

- Einen Status zuordnen, damit die Testfälle weiterverarbeitet werden können
- Den aktuellen Aufwand messen
- Einen Fehler melden und Informationen zur Reproduktion von Fehlern eingeben, z.B. die Schritte, die zu einem Dump geführt haben.

### Voraussetzungen

Der Testingenieur hat Ihrem Benutzer (oder einem Team, zu dem Sie gehören) ein Testpaket zugeordnet.

## Vorgehensweise

1. Wählen Sie in der Anwendung *Meine Aufgaben - Tester-Arbeitsvorrat* ein Testpaket aus. Die Testfälle, die dem Testpaket zugeordnet sind, werden in der zweiten Tabelle angezeigt.
2. Wählen Sie einen manuellen Testfall aus. Manuellen Testfällen ist das Symbol *Testdokument* in der Spalte *Art* zugeordnet.
3. Wählen Sie *Ausführen*. Die Anwendung zur Ausführung manueller Testfälle wird geöffnet, sodass Sie den Test ausführen und den Status aktualisieren können.
4. Die Anwendung verfügt über zwei verschiedene Layouts: die Tabellensicht (Standardsicht) und die Listenansicht. Wenn Sie die Listenansicht verwenden möchten, sollten Sie sie in Ihrem Personalisierungsprofil auswählen.
5. Wählen Sie *Testfall anzeigen*, um das Testdokument zu öffnen.
6. Wenn dem Testfall eine ausführbare Einheit zugeordnet ist, können Sie in einer Dropdown-Auswahl festlegen, wie das TBOM-Trace gehandhabt wird:
  - Wählen Sie *Anlegen*, dann wird ein Dialogfenster zur Ausführung mit TBOM angezeigt.
  - Wählen Sie *Nicht anlegen*, um die ausführbare Einheit direkt auszuführen, ohne das TBOM-Trace zu starten.

(Um die Dropdown-Auswahl zu aktivieren, haben Sie den TBOM-Trace in Ihrem Personalisierungsprofil angefordert.)

Unter *Testfall* werden die wichtigsten Attribute des Testfalls angezeigt (Beschreibung, Priorität, Verantwortlicher).

Unter *Ausführbare Einheiten* werden die dem Testfall zugeordneten ausführbaren Einheiten sowie Informationen wie Art, logische Komponentengruppe und Zielsystem angezeigt. Sie können auf die ID der ausführbaren Einheit klicken, um diese direkt im Zielsystem zu starten.

Unter *Fortschritt* sehen Sie den Status. Sie können Status, Testaufwand und Kommentar ändern und die Einschränkungskennzeichen setzen.

7. Um die Messung des Aufwands für die Ausführung des aktuellen Testfalls zu starten, wählen Sie *Ausführung starten*.
8. Nachdem Sie den Testfall ausgeführt haben, wählen Sie *Ausführung anhalten*. Der Testaufwand wird angezeigt, und Sie können ihn ggf. ändern.
9. Wählen Sie den entsprechenden Teststatus aus. Die folgenden Werte sind von SAP vordefiniert:

Situation	Status	Aktion
Sie haben noch keinen Test durchgeführt.	Nicht getestet	-
Ein Test wurde vorübergehend unterbrochen.	In Bearbeitung	-
Der Test wurde erfolgreich ausgeführt.	OK	-
Fehler gefunden, die den Einsatz der Software geringfügig beeinträchtigen (z. B. ergonomische Schwächen, mangelnde Benutzerführung).	OK mit Einschränkungen	Um eine Erklärung einzugeben, legen Sie eine Testnotiz oder ein Testergebnis an. Legen Sie einen Fehler an, um das Issue von der Testentwicklung beheben zu lassen.

Situation	Status	Aktion
Programmfehler oder gravierende Mängel gefunden	Fehler: Nachtest erforderlich	Um eine Erklärung einzugeben, legen Sie eine Testnotiz oder ein Testergebnis an. Legen Sie einen Fehler an, um das Issue von der Testentwicklung beheben zu lassen.
Beim Nachtest wurde der ursprüngliche Fehler behoben, und es sind keine neuen Fehler aufgetreten.	Nachtest O.K.	Um eine Erklärung einzugeben, legen Sie eine Testnotiz oder ein Testergebnis an. Legen Sie einen Fehler an, um das Issue von der Testentwicklung beheben zu lassen.

### **i** Hinweis

Welche der genannten Status möglich sind, hängt von der Konfiguration Ihres Systems ab. Wenn Sie sich über die Bedeutung eines Status unsicher sind, fragen Sie Ihren Systemadministrator.

Sie können die Status in der Anwendung *Verwaltung - Test-Suite* unter [► Vorbereitung](#) [► Testfallstatus](#) definieren.

10. Geben Sie den Testaufwand ein.
11. Geben Sie unter *Einschränkungen* an, welche Art von Objekt getestet wurde. Wenn die Ausführung des Tests beispielsweise dahingehend eingeschränkt war, dass die Funktionen, nicht jedoch die Dokumentation überprüft wurden, wählen Sie das Feld *Funktionen*.
12. Unter *Fehler zuordnen* können Sie einen Fehler anlegen, einen vorhandenen Fehler zuordnen oder die Zuordnung aufheben.
13. Zum Hinzufügen von Informationen haben Sie folgende Möglichkeiten:
  - Fügen Sie eine Testnotiz mit Textinformationen hinzu. Sie können eine Notiz aus dem Testdokument, der Vorlage oder per Upload anlegen. Sie können vorhandene Notizen bearbeiten. Sie können per Drag&Drop ein neues Dokument hochladen, um eine Testnotiz anzulegen oder zu aktualisieren.
  - Fügen Sie Testergebnisse hinzu. Im Gegensatz zu Testnotizen sind Testergebnisse testerspezifisch, und Sie können mehrere für einen Testfall anlegen.

### **i** Hinweis

Falls es nötig ist, eine Testnotiz oder ein Testergebnis obligatorisch zu machen, können Sie dies in der Anwendung *Verwaltung - Test-Suite* unter

*Globale Einstellungen pflegen* aktivieren:

Unter *Sonstige Einstellungen* können Sie festlegen, dass ein Tester einen Testnotiz- oder Testergebniseintrag anlegen muss, wenn der Test einen definitiven Status hat (also rot oder grün ist). Der Tester kann keine Sicherung ausführen, ohne den entsprechenden Eintrag angelegt zu haben.

14. Wenn die Ausführung des Tests erfolgreich war und das Feld *Test beendet* nicht automatisch ausgewählt wird, wählen Sie es aus.



### **i** Hinweis

Ob das Feld *Test beendet* automatisch ausgewählt wird oder nicht, hängt davon ab, wie Ihr System konfiguriert ist.

Weitere Informationen finden Sie unter [Administration \[Seite 123\]](#).

15. Sichern Sie Ihre Angaben, und schließen Sie das Fenster.

### **i** Hinweis

Wenn Sie eine ausführbare Einheit unter *Meine Aufgaben - Tester-Arbeitsvorrat* auswählen, sieht die Anwendung etwas anders aus: Der Bereich *Testfall* und die Drucktaste *Testfall anzeigen* sind nicht vorhanden. Unter *Ausführbare Einheiten* wird nur die ausführbare Einheit angezeigt, an der Sie derzeit arbeiten.

In der Anwendung zur Ausführung des manuellen Tests sind zwei zusätzliche Bereiche vorhanden:

- *Testpaketanlage*: angehängte Dokumente
- *Zugeordnete Tester*: Tester, die demselben Testfall zugeordnet sind
- *Allgemeine Info*: Verwaltungsdetails zu Testplan und Testpaket

## **Ergebnis**

Sie haben das Ergebnis Ihres Testfalls dokumentiert. Der Testorganisator hat einen Überblick über den Status aller Testfälle in Ihrem Testpaket.

## **12.2 Ausführen automatisierter Testfälle**

### **Voraussetzungen**

Der Testkoordinator oder -organisator hat Ihrem Benutzer ein Testpaket zugeordnet.

### **Vorgehensweise**

1. Wählen Sie in der Anwendung *Meine Aufgaben - Tester-Arbeitsvorrat* ein Testpaket aus.  
Die Testfälle, die dem Testpaket zugeordnet sind, werden angezeigt.
2. Wählen Sie im Bereich *Testfälle* einen automatisierten Testfall aus. Automatisierten Testfällen ist das Symbol *Automatischer Test* in der Spalte *Art* zugeordnet.

3. Wählen Sie *Ausführen*.

Sie gelangen auf das Bild *Startoptionen*.

4. Wählen Sie die geeigneten Startoptionen aus.

Weitere Informationen finden Sie unter *eCATT: extended Computer Aided Test Tool* unter <http://help.sap.com> unter ► *Testskripte ausführen* ► *Startoptionen* ►.

5. Wählen Sie *Ausführen*.

Das Ausführungsprotokoll des automatisierten Testfalls wird angezeigt.

Weitere Informationen finden Sie in der Hilfe zur Anwendung *eCATT: extended Computer Aided Test Tool* unter <http://help.sap.com> unter ► *Protokolle* ► *Protokolle verwenden* ►.

6. Überprüfen Sie das Protokoll auf Fehler.

7. Beenden Sie die Funktion.

8. Wenn Fehler aufgetreten sind, fügen Sie Informationen hinzu:

1. Kehren Sie zur Anwendung *Meine Aufgaben - Tester-Arbeitsvorrat* zurück.
2. Wählen Sie den entsprechenden Testfall aus.
3. Wählen Sie *Status bearbeiten*.

Sie gelangen in das Dialogfenster *Statusbearbeitung*.

1. Ändern Sie den Status manuell oder ordnen Sie einen Status manuell zu, sofern die Einstellung nicht automatisch erfolgt.

### **i** Hinweis

Der Status wird automatisch gesetzt, wenn die Option *Status in die TWB kopieren* bei den *Startoptionen* nicht abgewählt ist.

2. Geben Sie im Bereich *Einschränkungen* an, welche Art von Objekt getestet wurde. Wenn die Ausführung des Tests beispielsweise dahingehend eingeschränkt war, dass die Funktionen, nicht jedoch die Dokumentation überprüft wurden, wählen Sie das Feld *Funktionen*.
3. Sichern Sie Ihre Eingaben.
4. Um Informationen hinzuzufügen, können Sie im Bereich *Details* des Tester-Arbeitsvorrats für den ausgewählten Testfall folgende Aktion ausführen:
  - Um eine Notiz anzulegen, wählen Sie in der Spalte *Testnotiz* die Option *Notes Management*.
  - Um einen Fehler anzulegen, wählen Sie *Fehler anlegen*.
  - Um einen vorhandenen Fehler zuzuordnen, wählen Sie *Fehler zuordnen*.
  - Um einen Fehler aus dem Teststatus zu entfernen, wählen Sie *Zuordnung aufheben*.

Hierdurch wird nur die Zuordnung gelöscht und nicht der Fehler. Passen Sie daher ggf. den Fehler an, wenn Sie dessen Zuordnung aufheben, beispielsweise können Sie ihn schließen, wenn er nicht mehr relevant ist.

## Ergebnisse

Sie haben das Ergebnis Ihres Testfalls dokumentiert. Der Testorganisator hat einen Überblick über den Status aller Testfälle in Ihrem Testpaket.




## 12.3 Bearbeiten von Testnotizen

### Kontext

Sei können Testnotizen über die App „Meine Aufgaben - Tester-Arbeitsvorrat“ anlegen und bearbeiten.

Um eine vorhandene Testnotiz zu verwalten, gehen Sie folgendermaßen vor:

### Vorgehensweise

1. Wählen Sie in der App „Meine Aufgaben - Tester-Arbeitsvorrat“ ein Testpaket aus.
2. Wählen Sie den relevanten Testfall aus.
3. Wählen Sie  [Testnotiz](#)  [Verwalten](#) . Das Bild [Testnotizverwaltung](#) wird geöffnet.
4. Sie haben mehrere Möglichkeiten zum Bearbeiten von Testnotizen:
  - **Testnotizen online bearbeiten**
    1. Wählen Sie [Online bearbeiten](#), um die Testnotiz als Word-Dokument in einem neuen Fenster zu öffnen. Die Testnotiz ist vorübergehend gesperrt, um zu verhindern, dass andere Benutzer gleichzeitig Änderungen daran vornehmen.
    2. Nehmen Sie die erforderlichen Änderungen vor, und schließen Sie anschließend das Fenster. Sie werden in einer Meldung gefragt, ob Sie die Änderungen sichern möchten.
    3. Bestätigen Sie die Änderungen. Die Testnotiz wird direkt aktualisiert.
  - **Testnotizen offline bearbeiten**
    1. Wählen Sie [Auschecken](#).
    2. Wenn Sie dazu aufgefordert werden, laden Sie die Testnotiz als Word-Dokument herunter. Die Testnotiz ist vorübergehend gesperrt, um zu verhindern, dass andere Benutzer gleichzeitig Änderungen daran vornehmen.
    3. Öffnen Sie das Word-Dokument und nehmen die erforderlichen Änderungen vor.
    4. Sichern Sie das Word-Dokument wie gewohnt.
    5. Wenn Sie fertig sind, kehren Sie zum Bild [Testnotizverwaltung](#) zurück und wählen [Einchecken](#).
    6. Wenn Sie dazu aufgefordert werden, wählen Sie das Word-Dokument aus, in dem Sie die Änderungen vorgenommen haben. Die Testnotiz wird im System aktualisiert und automatisch entsperrt.  
Falls Sie Ihre Meinung ändern und die Änderungen verwerfen möchten, können Sie auch [Auschecken zurücknehmen](#) wählen. Die Testnotiz wird entsperrt und bleibt unverändert. Anschließend können Sie das Word-Dokument, das Sie heruntergeladen haben, löschen.
5. Wählen Sie das Signatursymbol, um die Signatur direkt aufzurufen. Die Auswahl des Signatursymbols führt zu einer der folgenden Reaktionen:
  - Anzeige des Signaturverlaufs, wenn die Anwendung sich im Anzeigemodus befindet, oder wenn der aktuelle Benutzer das Dokument selbst nicht unterzeichnen kann, z.B. wenn er es bereits unterzeichnet hat
  - Anzeige des Popups zum Unterzeichnen des Dokuments, wenn der aktuelle Benutzer zum Unterzeichnen berechtigt oder verpflichtet ist.

## 12.4 Anlagen von Testpaketen anzeigen

### Verwendung

Sie können Dokumente, die an Testpakete angefügt sind, anzeigen.

Ein Testkoordinator kann z. B. mehrere Tester demselben Testpaket zuordnen. In einem Dokument, das dem Testpaket angefügt wurde, kann der Testkoordinator weitere Informationen zum Test angeben, z.B. zu den Eingabedaten und den erwarteten Ausgabedaten.

### Voraussetzungen

Der Testorganisator hat Ihrem Benutzer ein Testpaket zugeordnet.

### Vorgehensweise

1. Wählen Sie in der Anwendung *Meine Aufgaben - Tester-Arbeitsvorrat* ein Testpaket aus.
2. Wählen Sie *Anlagen*.  
Im Bild *Testpaketanlagen* werden Ihnen die Dokumente angezeigt, die der Testorganisator bereitgestellt hat.

Tester können Testpaketanlagen während der Ausführungsphase verwalten. Sie können Informationen zu Testfällen anderen Personen über eine Testfallnotiz zur Verfügung stellen.

## 12.5 Einplanen automatischer Tests

### Voraussetzungen

Der Testingenieur hat Ihrem Benutzer ein Testpaket zugeordnet.

Das Testpaket sollte nicht mit einer Testsequenz verknüpft sein, da die Ausführungsreihenfolge während der Einplanung ohnehin nicht verwaltet werden kann.

## Kontext

Sie können die Ausführung von automatisierten Testfällen für mehrere Testpakete einplanen.

Sie können auswählen, ob die Testfälle im Vorder- oder Hintergrund verarbeitet werden sollen. Bei der Verarbeitung im Vordergrund haben Sie die Möglichkeit, jeden Schritt anzuzeigen und manuell Änderungen an den Systemvorgabewerten vorzunehmen. Vergewissern Sie sich, dass der Ausführungsmodus `Background` nur dann ausgewählt wird, wenn die getesteten Vorgänge keine Benutzungsoberfläche haben.

## Vorgehensweise

1. Wählen Sie in der Anwendung *Meine Aufgaben - Tester-Arbeitsvorrat* ein Testpaket aus, das automatische Testfälle enthält.

Die Testfälle, die dem Testpaket zugeordnet sind, werden angezeigt.

2. Wählen Sie **► Automatischer Test ► Ausführung einplanen ►**.

Sie gelangen auf das Dialogfenster *Einplanung*.

3. Wenn es sich um eine Probeausführung handelt und der Status dieser Probeausführung nicht in die Statusanalyse der Test-Suite übernommen werden soll, entmarkieren Sie *Status in Test Workbench kopieren*.
4. Wenn Sie TBOMs anlegen und die Anwendung *Business Process Change Analyzer* verwenden möchten, markieren Sie *TBOM-Aufzeichnung aktivieren*.
5. Um die Testfälle im Hintergrund auszuführen, wählen Sie *Hintergrund* in der Dropdown-Box *Ausführungsmodus*.
6. Definieren Sie den Startauslöser: sofort, Startdatum und -zeit, nach Job oder nach Ereignis.
7. Wenn Sie *Periodischer Job* markieren, müssen Sie die Periode festlegen: stündlich, täglich, wöchentlich oder monatlich.
8. Wählen Sie *OK*, um den Job einzuplanen.

## 12.6 Zuordnen von Vorgangsarten für Test-Suite

### Voraussetzungen

- Sie haben die Standardvorgangsarten im Customizing (Transaktion `SPRO`) für SAP Solution Manager unter **► Capabilities ► IT-Servicemanagement ► Vorgänge ► Vorgangsarten definieren ►** angepasst.
- Sie haben die Berechtigung als Administrator.

## Kontext

Zum Anlegen von Fehlern für Testfälle hat SAP für jede Fehlerart eine Meldungsart vordefiniert.

Sie können kundeneigene Vorgangsarten aus dem IT-Servicemanagement den Fehlerarten im Testmanagement zuordnen.

## Vorgehensweise

1. Gehen Sie zum Customizing für SAP Solution Manager und wählen Sie **►► Test-Suite ► Test-Suite-Vorbereitung ►► Spezielle Konfiguration ► Fehler ► Geschäftsvorgangsarten und Vorlagen zuordnen** **▢**.
2. Die Vorgangsart `SMIN` (Meldung im IT-Servicemanagement) ist standardmäßig den Anwendungsfehlern zugeordnet.

Sie können den Standard mit Ihren kundeneigenen Vorgangsarten ersetzen, die Ihren Meldungsarten für kundeneigene Vorgänge entsprechen.

Sie können verschiedene Vorgangsarten für Fehler in der Test-Suite auswählen. In diesem Fall können Sie neue Einträge der Anwendungsart *Anwendungsfehler* anlegen und eine neue Vorgangsart definieren.

3. Die Vorgangsart `SMDT` (Testfallfehler) wird standardmäßig Testfallfehlern zugeordnet.

Sie können den Standard mit Ihren kundeneigenen Vorgangsarten ersetzen, die Ihren Meldungsarten für kundeneigene Vorgänge entsprechen.

Sie können verschiedene Vorgangsarten für Fehler in der Test-Suite auswählen. In diesem Fall können Sie neue Einträge der Anwendungsart *Testfallfehler* anlegen und eine neue Vorgangsart definieren.

Sichern Sie Ihre Angaben, und schließen Sie das Fenster.

## 13 Tester-Arbeitsvorrat überprüfen

Hier können Testkoordinatoren auf den Tester-Arbeitsvorrat eines Testers zugreifen. Er greift auf den Tester-Arbeitsvorrat zu und kann den Tester auswählen.

Die Anwendung ist dieselbe wie der Tester-Arbeitsvorrat an sich und der Testkoordinator kann sämtliche Testausführungsdaten (Testnotiz, Testergebnisse, manuelle Testausführung) für den ausgewählten Tester anzeigen.

Der Testkoordinator kann auch Änderungen vornehmen, als ob er selbst der Tester wäre.

# 14 Personalisierung

In dieser Anwendung können Sie Personalisierungsprofile anlegen, um häufig verwendete Werte zu speichern. Sie können mehrere Personalisierungsprofile anlegen, jedoch muss eines als Standard festgelegt werden.

Das Anwendungsbild wird von den Test-Suite-Anwendungen mit den im Standardpersonalisierungsprofil verfügbaren Werten vorbelegt, ohne zusätzliche Pop-up- oder Selektionsoption

- Voraussetzungen:  
Anwendungen wie Testplanverwaltung, BPCA usw. enthalten zahlreiche Felder, die für die Eingabe erforderlich sind, bevor die Anwendungen verwendet werden können. Die Erstellung kann längere Zeit in Anspruch nehmen und falls Sie bestimmte Werte häufig verwenden, sollten Sie diese Werte zentral speichern und beim Erstellen einer neuen Transaktion wiederverwenden. Falls Sie nur eine Lösung verwenden, sollten Sie dieselbe Lösung nicht bei jeder Erstellung eines Testplans eingeben müssen, sofern Sie den Lösungsnamen in Ihrem Personalisierungsprofil gespeichert haben.
- Details:  
Jedes Personalisierungsprofil enthält zwei Arten von Parametern:
  - Anwendungsübergreifend verwendete Parameter: Nur Lösungsdokumentationsparameter, d.h. Lösung, Branch und Sicht werden über andere Anwendungen wie BPCA, Testplanverwaltung usw. hinweg verwendet.
  - Innerhalb einer Anwendung verwendete Parameter: Für alle Parameter, die Sie innerhalb einer Anwendung definieren. Sie können z.B. in BPCA die Systemrolle für die TBOM-Erstellung pflegen. Dies wird nur innerhalb von BPCA verwendet
- Verwendung:  
Legen Sie ein Personalisierungsprofil an und legen Sie eines davon als Standard fest. Die Anwendungen und Daten vom Personalisierungsprofil werden automatisch vorbelegt. Falls Sie andere Daten oder Einstellungen verwenden möchten, können Sie das aktuell als Standard definierte Profil aktualisieren oder ein neues Profil erstellen und als Standard festlegen.



# 15 Beschädigte Testfälle

Der Arbeitsvorrat für beschädigte Testfälle unterstützt die Verwaltung des Reparaturprozesses für beschädigte Testfälle.

Als Testingenieur können Sie

- die Fehlermeldungen zu beschädigten Testfällen bearbeiten und verwalten
- Routinen zur Analyse der Gefahrenursache starten
- zusätzliche Informationen von den Meldenden anfordern

## Voraussetzungen

- Sie besitzen die notwendigen Berechtigungen für die Test-Suite.
- Der Administrator hat für Sie einen Systembenutzer angelegt und ihm einen Geschäftspartner zugeordnet.

## Funktionsumfang

Ihnen steht eine Liste der Testfallfehler zur Verfügung, die Sie beispielsweise nach dem Status filtern können.

Wenn Sie die Berechtigung als Testingenieur haben, können Sie

- Testskripte bearbeiten
- Testfälle ausführen
- Wechseln Sie zur Verwendungsstelle eines Testfalls in der Lösung. Sie können direkt zum Geschäftsprozessschritt wechseln, für den der Testfall relevant ist und den Kontext prüfen, aus dem ein Testfall gestartet wird.
- Wechseln Sie zur Statuspflege eines Testfalls, um Meldungen oder die Ausführungshistorie anzuzeigen.
- Rufen Sie die CRM-Transaktion auf, um die technischen Details einer Support-Meldung anzuzeigen.

Meldende und Testingenieure haben stets Zugriff auf die Testfallfehler-Inhalte und die Testfälle.

## Abfragen

Sie können komplexe Suchen durchführen, indem Sie Abfragen definieren oder direkt Selektionskriterien angeben. Beispielsweise können Sie nach Status, Testskript und Erstellungsdatum filtern.

Sie können nach bestätigten Testfallfehlern suchen, die aus der Standardliste beschädigter Testfälle entfernt werden.

Der Testkoordinator hat Zugriff auf dieselbe Sicht und kann sich eine Gesamtübersicht über beschädigte Testfälle verschaffen, die von einem der Testpläne, für die er verantwortlich ist, gemeldet wurden.

Der Testkoordinator kann einen vorhandenen Testfallfehler an alle Testpläne und Testpakete, in denen derselbe Testfall verfügbar ist, verteilen.

# 16 Testausführungsreports

Im SAP Solution Manager Launchpad sind die folgenden Anwendungen zum Ausführen von Testreports verfügbar:

## Test-Suite - Übersicht

In der Anwendung *Test-Suite - Übersicht* wird auf den folgenden Registerkarten der Status manueller und automatischer Tests angezeigt:

- **Testplanung**  
Zeigt die Testpläne sowie die Anzahl der Testfälle, der ausführbaren Einheiten, der manuellen und automatischen Tests, der offenen Testfälle und der offenen Fehler an. Wenn Sie einen Testplan auswählen, wird der Status der darin enthaltenen Testpakete und Testfälle angezeigt.
- **Testausführung**  
Zeigt dem Benutzer den Status der eigenen Testpakete in einem Testplan an.

## Testausführung - Jobs

Die Liste enthält die Hintergrundjobs, die durch die Einplanung von automatischen Tests im Testerarbeitsvorrat angelegt wurden.

Es sind zwei Abfragen verfügbar: *Meine Jobs* und *Alle Jobs*. Sie können komplexe Suchvorgänge durchführen, indem Sie neue Abfragen definieren oder die Selektionskriterien der vorhandenen Abfragen ändern.

Sie können zu den Jobdetails navigieren, indem Sie auf den Jobnamen klicken.

## Automatisierte Tests – Ergebnisse

Diese Anwendung zeigt die Protokolle der ausgeführten automatischen Tests an.

In der Protokollübersicht können Sie Ihre Suchkriterien wie gewünscht definieren, um den Status der ausgeführten automatischen Tests zu prüfen.

Sie können zu den Einzelprotokolldetails navigieren, indem Sie auf die Protokoll-ID klicken. Zum Untersuchen von Testausführungsfehlern werden in dieser Sicht detailliertere Informationen zu folgenden Elementen angezeigt:

- Ausführungsstatus
- Ausführungskontext
- Solution-Manager-Kontext

# 17 Test-Suite - Analysen

## Verwendung

In der Anwendung *Test-Suite - Analysen* der Test-Suite können Sie mithilfe von Reports Informationen zu Tests auswerten.

## Voraussetzungen

Sie haben die Aggregationsregeln zur Berechnung des Testfallstatus für Testpläne und Testpakete im Customizing für SAP Solution Manager unter ► *Capabilities* ► *Testmanagement* ► *Test-Suite für SAP Solution Manager* ► *Vorbereitung* ► *Setup* ► *Zentrale Test-Suite-Einstellungen* ► *Testplanauswertung anpassen* ► definiert.

## Aktivitäten

1. Wählen Sie im Launchpad die Anwendung *Test-Suite - Analysen*.
2. Wählen Sie unter den folgenden Gruppen den Report aus, den Sie ausführen möchten:
  - [Vollständigkeits- und Lücken-Reports \[Seite 275\]](#)
  - [Testausführungsanalyse \[Seite 276\]](#)
  - [Status- und Fortschrittsanalyse \(BW-basiert\) \[Seite 279\]](#)
3. Ein neues Fenster wird geöffnet, in welchem Sie Suchkriterien definieren und Ergebnisse dazu erhalten können.

### i Hinweis

Für Analysereports, die die Hierarchie zeigen (Testplan – Testfallanalyse, Testpaket - Testfallanalyse), können Sie als Format eine flache Liste auswählen, um die Daten anzuzeigen.

## 17.1 Vollständigkeits- und Lücken-Reports

### Verwendung

Mit den Reports in diesem Bereich können Sie feststellen, ob Ihre [Lösungsdokumentation](#) ausreichend mit TBOM-Zuordnungen und Testfällen abgedeckt ist. Sie können prüfen, welche Testpläne von Änderungen in der Lösungsdokumentation betroffen sind.

## Aktivitäten

Wählen Sie im Launchpad [Test-Suite - Analysen](#).

Unter [Vollständigkeits- und Lücken-Reports](#) können Sie die folgenden Reports verwenden:

- [TBOMs und Testfälle nach ausführbaren Einheiten](#)
- [Testfallzuordnung zu Lösungsdokumentation](#)

Dieser Report zeigt, ob Testfälle in Ihrer Lösungsdokumentation gemäß den von Ihnen angegebenen Selektionskriterien verfügbar sind (positive Liste) oder fehlen (negative Liste) oder beides. Die Auswahl für Struktur- und Testfallattribute kann über erweiterte Suchkriterien im Filterprofil angepasst werden.
- [Testfälle](#)
- [Testpläne - Knoten- und Testfalländerungen](#)

Mit diesem Report können Sie Testpläne ermitteln, bei denen sich nach dem Generieren des Testplans der zugrunde liegende Satz aus Lösung, Branch und Sicht geändert hat. Das System ermittelt, ob Testpläne aufgrund der folgenden Änderungen von der aktuellen Lösungsdokumentation abweichen:

  - Die Struktur der Lösungsdokumentation hat sich geändert, beispielsweise wurde ein neuer Prozess mit Testfällen hinzugefügt oder Testfälle oder Strukturknoten, die in den Testplan eingebunden waren, wurden gelöscht.
  - Es sind neue Versionen von Testdokumenten vorhanden, die in den Testplan eingebunden sind.
  - Einer Testkonfiguration wurde eine neue Testdatenvariante zugeordnet.

Solche Testpläne können Sie überprüfen und aktualisieren, indem Sie die Änderungen integrieren und die Testpläne erneut sichern. In den Reportergebnissen können Sie die Testpläne durch Doppelklick auf den Titel im Änderungsmodus öffnen. Bevor Sie den Testplan sichern, können Sie in der Testplanstruktur beispielsweise prüfen, welche Strukturelemente entfernt oder hinzugefügt wurden. (Das System kennzeichnet diese Änderungen mit Symbolen.)
- [Testfall nicht in Testplan eingeschlossen](#)

Dieser Report zeigt an, welche Testfälle (mit Testdokument oder Testkonfiguration) für eine Prozessdokumentation (Lösung, Branch und Sicht) in keinen Testplan bzw. kein Testpaket eingefügt wurden. Wenn der Testfall in kein Testpaket eingefügt wurde, können Sie über den entsprechenden Link die Liste der Testpläne öffnen, denen der Testfall zugeordnet ist. Sie können zum ausgewählten Testplan navigieren und ein Testpaket zuordnen.

## 17.2 Testausführungsanalyse

### Verwendung

Sie können Statusanalysen für ein(en) oder mehrere Testpläne oder Testpakete mit unterschiedlichen Arten von Analysen durchführen.

## Funktionsumfang

Sie können die Reports über das Launchpad in der Launchpad-Gruppe [Test-Suite](#) starten. Unter [Testausführungsanalyse](#) sind folgende Reports verfügbar:

### Mehrfachtestplan-Status

In Ihren Suchergebnissen sehen Sie alle Testpläne entsprechend Ihren Suchkriterien mit aggregierten Informationen zur Testausführung. Sie können z.B. die Anzahl der Testfälle, die gerade ausgeführt werden (im Status kein Ergebnis, Fehler, OK) und die Anzahl der offenen Fehler anzeigen.

Für den Gesamttestplan können Sie den Testplanstatus, den Verantwortlichen, die Anzahl an Testpaketen und Testern sowie die Anzahl an nicht zugeordneten Testfällen anzeigen.

Wenn Sie einen Testplan auswählen, geben mehrere Grafiken den Status an:

- Testfälle pro Typ
- Testfälle pro Status
- Fehler pro Priorität

Sie können auch auf die Testfallanalyse für den ausgewählten Testplan zugreifen.

### Details zum Mehrfachtestplan-Status

Die in den Suchergebnissen aufgelisteten Testpläne liefern detaillierte Informationen zum Testausführungsstatus von Testpaketen, Testfällen und Testern.

### Testplan – Testpaketanalyse

Das Suchergebnis zeigt eine Hierarchie mit den folgenden Ebenen an: Testplan, Testpakete und zugeordnete Tester.

Für die Testplan- und Testpaketzeilen können Sie aggregierte Werte zur Testauswertung anzeigen, z.B. die Anzahl an beteiligten Testern, die Anzahl an Testfällen je spezifischem Status, die Anzahl an Fehlern und die Anzahl an Testfällen pro Typ.

Für die Testerzeile können Sie den Geschäftspartner des Testers sehen und ob seine Zuordnung des Partners gesperrt oder entsperrt ist. Wählen Sie [Springen](#), um zu den anderen Reports zu navigieren, die für die ausgewählte Zeile verfügbar sind.

Wählen Sie [Personalisierung](#), um weitere Spalten hinzuzufügen. In der Hierarchie ist die Knotenbeschreibung ein Link, mit dem Sie zum relevanten Objekt navigieren können, z.B. zum Testplan oder Testpaket.

### Testplan – Testfallanalyse

Das Suchergebnis zeigt die Testplanhierarchie mit Statusinformationen an. Auf Testplanebene sehen Sie aggregierte Werte zum Testfall-Ausführungsstatus: Anzahl an nicht zugeordneten Testfällen, Anzahl an Testfällen pro Status und Anzahl an Fehlern.

Auf Testfallebene können Sie einzelne Statusinformationen, wie Datum und letzter Tester, und einige Testfalldaten, wie die Testfallklassifizierung und -priorität, sehen.

Wählen Sie [Personalisierung](#), um weitere Spalten hinzuzufügen. In der Hierarchie ist die Knotenbeschreibung ein Link, mit dem Sie zum relevanten Objekt navigieren können.

Klicken Sie auf das Statussymbol eines Testfalls, um ein Dialogfenster mit allen Teststatus pro Paket und Tester zu öffnen. Diese Status werden in den Gesamttestfall-Status aggregiert.

Klicken Sie auf das Testnotiz- oder das Testergebnis-Symbol, um die [Testdokumentübersicht](#) auf einer neuen Registerkarte aufzurufen. Hier können Sie Ihre Testnotiz oder Ihr Testergebnis anzeigen und abzeichnen

### Testpaket - Testfallanalyse

Das Suchergebnis zeigt die Testpakethierarchie mit Statusinformationen an. Auf Testpaketebene sehen Sie aggregierte Werte zum Testfall-Ausführungsstatus: Prozentsatz der noch fehlenden Ergebnisse pro Status und Anzahl der Fehler.

Auf Testfallebene können Sie einzelne Statusinformationen wie Datum und letzter Tester und z.B. die Testfallklassifizierung und -priorität sehen.

Wählen Sie [Personalisierung](#), um weitere Spalten hinzuzufügen. In der Hierarchie ist die Knotenbeschreibung ein Link, mit dem Sie zum relevanten Objekt navigieren können.

Klicken Sie auf das Statussymbol für einen Testfall, um ein Dialogfenster mit allen Teststatus pro Paket und Tester zu öffnen. Diese Status werden in den Gesamttestfall-Status aggregiert.

### Testplan – Fehleranalyse

Dieser Report liefert eine Übersicht über die Fehler, die in einem Testplan angelegt wurden. Wenn ein Test nicht durchgeführt werden konnte, legen Tester Fehler an.

1. Sie können mehrere Kriterien definieren, um eine Analyse für alle Testpläne zu erhalten.
  - Verwenden Sie die folgenden Listen, um die Granularität des Reports zu bestimmen:
    - [Fehleranalyse](#): Details zum Fehler und dem zugehörigen Testfall. Wenn ein Fehler mehreren Testfällen zugeordnet wurde, wird er also mehrmals in der Liste aufgeführt.
    - [Fehlerübersicht](#): Fehlerinformationen in einem aggregierten Modus, sodass jeder Fehler nur einmal angezeigt wird.
    - Sie können Fehlervorgangsarten auswählen.
2. Abhängig von den Suchkriterien zeigt das System eine Analyse oder eine Übersicht der Fehler für die Testpläne mit nützlichen Informationen wie Fehler-ID, Status, Priorität und Meldendem an.
3. Um die Meldung für einen Fehler in Ihrem Incident Management System zu öffnen, klicken Sie auf die Fehler-ID. Hier erhalten Sie weitere Informationen und bearbeiten die Meldung im [IT-ServiceManagement](#).

Fehlerinformationen werden direkt aus dem Meldungssystem abgerufen und sind daher stets aktuell.

Alternativ können Sie auf die durch einen Test geplanten Reports in der Anwendung [Testplanverwaltung](#) zugreifen. Wählen Sie einen Testplan aus, und wählen Sie [Springen](#) [Fehlerübersicht](#).

### Testpaket - Fehleranalyse

Dieser Report zeigt die für Testfälle innerhalb eines bestimmten Testpakets angelegten Fehler an. Der Report ähnelt der Fehleranalyse für Testfälle, er zeigt jedoch nur die Fehler für ein ausgewähltes Testpaket an.

### Testergebnisdokument

Mit diesem Report können Sie ein Microsoft-Word-Dokument mit Informationen zu einem Testplan generieren. Diese umfassen Testpaketinformationen, Fehler und Landschaftsdetails.

Wählen Sie einen Testplan und die Informationen, die das Dokument enthalten soll, aus. Wählen Sie anschließend [Ausführen](#), und sichern Sie die Datei lokal.

### Rückverfolgbarkeitsmatrix

Mit dem Rückverfolgbarkeitsmatrix-Report können Sie den gesamten Lebenszyklus Ihrer Projekte, IT-Anforderungen, Änderungsanträge und Änderungsdokumente sowie Transportschleifen, die die Projekte bis zu

ihrem Abschluss durchlaufen, verfolgen. Für diesen Report typische Benutzer sind Projektmanager, Qualitätsmanager und Testmanager.

Nachdem Sie einen Filter für Ihre Projekte gesetzt haben, findet die Verfolgung durch den Report beginnend mit den Projekten bis hin zu den relevanten Prozessen statt und die relevanten Ergebnisse werden angezeigt.

Sie finden die KPI-Definition und Berechnung im KPI-Katalog.

Die verfolgten Prozesse sind:

- PPM-Projekt
- IT-Anforderung
- Geschäftsanforderung
- Änderungsantrag
- Änderungsdocument
- Testplan, Testpaket, Testfall.
- Fehler
- Transport
- Änderungszyklus
- Lösungsdokument

#### **Voraussetzung**

Führen Sie in der *SAP-Solution-Manager-Konfiguration* folgende Schritte durch:

- Konfiguration des Projektmanagements
- Konfiguration des IT-Servicemanagements
- Konfiguration des Change-Control-Managements
- Konfiguration der Test-Suite
- Konfiguration des Dashboard Builders

## **Weitere Informationen**

[Dashboard Builder](#)

[KPI-Katalog](#)

## **17.3 Status- und Fortschrittsanalyse (BW-basiert)**

### **Verwendung**

Der Test-Suite-Status und BW-basierte Fortschrittsberichte (Dashboards und Tabellen) geben Einblick in manuelle Tests auf Basis einer Testplanstruktur, einschließlich aller ausgewählten Testfälle der Lösungsdokumentation.

Dadurch können Sie Ihre Analyse am besten durchführen und so tiefgehend wie gewünscht. Sie können z.B. die Detailsicht in Tabellenform einsehen, sie mit Attributfiltern auf verschiedenen Ebenen anpassen und abstimmen und die Funktion im Kontextmenü verwenden.

## i Hinweis

Die Daten werden durch die letzte Datenextraktion aktualisiert, standardmäßig einmal täglich. Sie können die Extraktionshäufigkeit in der [SAP-Solution-Manager-Konfiguration](#) ändern.

## Voraussetzungen

Führen Sie in der [SAP-Solution-Manager-Konfiguration](#) folgende Schritte durch:

- Richten Sie im Szenario [Infrastrukturvorbereitung](#) das System [SAP BW](#) ein.
- Konfigurieren Sie [Analysen](#) im Szenario [Test-Suite-Vorbereitung](#), einschließlich der folgenden Schritte:
  - BW-Content aktivieren
  - Extraktionseinstellungen sichern
  - Vorgangsarten auswählen.
- Konfigurieren Sie in der [Test-Suite-Vorbereitung](#) unter [Standardkonfiguration](#) die Option [Vorlagebenutzer anlegen](#).

## Aktivitäten

Starten Sie [Test-Suite - Analysen](#) über das Launchpad. Unter [Status- und Fortschrittsanalyse](#) finden Sie die folgenden Reports:

### Manuelles Testen von KPIs

Dieses Dashboard enthält verschiedene KPIs für manuelle Tests, die automatisch wichtige Statistiken und Reports für manuelle Tests zusammenstellen, anstatt manueller Berechnungen. Mithilfe der globalen Filter können Selektionen wie Lösung und Branch gesetzt werden.

### Statusreports

Hier erhalten Sie eine Übersicht über den Status der Testfälle zu einem bestimmten Zeitpunkt. Standardmäßig verwendet das System das Datum der letzten Datenextraktion als Filter. Folgende Reports stehen Ihnen zur Verfügung:

- Testplan - Statusanalysen
- Testpaket - Statusanalysen
- Testfall - Statusanalysen

### Fortschrittsreport

Die [Fortschrittsanalyse](#) gibt Ihnen eine Übersicht über den Fortschritt der Testfall-Statusangaben über einen bestimmten Zeitraum. Standardmäßig verwendet das System die letzten 30 Tage als Filter.



## Fehlerreports

Hier erhalten Sie eine Übersicht über Anzahl, Priorität und Bearbeitungsstatus der Meldungen zu einem bestimmten Zeitpunkt. Der Standardfilter ist auf das geplante Startdatum des Testplans innerhalb der letzten 30 Tage gesetzt. Folgende Reports stehen Ihnen zur Verfügung:

- Fehleranalysen - Status
- Fehleranalysen - Priorität
- Fehleranalysen - Vorgang

## Testaufwandsreports

Hier erhalten Sie eine Übersicht über die Verteilung von Testfällen sowie deren Testaufwand zusammen mit dem geplanten und tatsächlichen Testaufwand im zeitlichen Verlauf. Folgende Reports stehen Ihnen zur Verfügung:

- Testaufwandanalysen
- Testaufwand im Zeitverlauf

## Weitere Informationen

[Dashboard Builder](#)

[KPI-Katalog](#)



[Floorplan Manager for Web Dynpro ABAP](#)

<https://www.sap.com/community/topic/web-dynpro-abap-and-floorplan-manager.html>

# Ausschlussklauseln und rechtliche Aspekte

## Hyperlinks

Einige Links werden durch ein Symbol und/oder einen Quick-Info-Text klassifiziert. Über diese Links erhalten Sie weitere Informationen. Informationen zu den Symbolen:

- Links zum Symbol : Sie rufen eine Website auf, die nicht von SAP gehostet wird. Durch die Nutzung solcher Links stimmen Sie Folgendem zu (sofern sich nicht aus Ihren Vereinbarungen mit SAP etwas anderes ergibt):
  - Der Inhalt der verlinkten Site ist keine SAP-Dokumentation. Basierend auf diesen Informationen ergibt sich für Sie keinerlei Produkthaftungsanspruch gegen SAP.
  - Weder widerspricht SAP dem Inhalt auf der verlinkten Site noch stimmt SAP ihm zu. Außerdem übernimmt SAP keine Gewährleistung für dessen Verfügbarkeit und Richtigkeit. SAP übernimmt keine Haftung für Schäden, die durch die Nutzung solchen Inhalts verursacht wurden, es sei denn, dass diese Schäden von SAP grob fahrlässig oder vorsätzlich verursacht wurden.
- Links zum Symbol : Sie verlassen die Dokumentation für das jeweilige SAP-Produkt oder den jeweiligen SAP-Service und rufen eine von SAP gehostete Website auf. Durch die Nutzung solcher Links stimmen Sie zu (sofern sich nicht aus Ihren Vereinbarungen mit SAP etwas anderes ergibt), dass sich basierend auf diesen Informationen für Sie keinerlei Produkthaftungsanspruch gegen SAP ergibt.

## Beta und andere experimentelle Funktionen

Experimentelle Funktionen sind nicht Teil des offiziellen Lieferumfangs, den SAP für künftige Releases garantiert. Dies bedeutet, dass experimentelle Funktionen von SAP jederzeit, aus beliebigen Gründen und ohne vorherige Ankündigung geändert werden können. Experimentelle Funktionen sind nicht zur Nutzung in einem Produktivsystem vorgesehen. Die experimentellen Funktionen dürfen nicht für Demonstrationen, Tests, Untersuchungen, Bewertungen oder anderweitige Zwecke in einer Produktivumgebung oder in Verbindung mit Daten, die nicht ausreichend gesichert wurden, verwendet werden.

Der Zweck der experimentellen Funktionen besteht darin, frühzeitig Feedback zu erhalten und so Kunden und Partnern die Möglichkeit zu geben, das zukünftige Produkt entsprechend zu beeinflussen. Durch die Abgabe von Feedback (z.B. über SAP Community) stimmen Sie zu, dass die geistigen Eigentumsrechte der Beiträge oder daraus abgeleiteten Werke im ausschließlichen Besitz von SAP verbleiben.

## Beispielcode

Bei dem Quelltext und/oder den Code-Snippets handelt es sich ausschließlich um beispielhafte Darstellungen. Sie sind nicht zur Nutzung in einem Produktivsystem vorgesehen. Der Beispielcode dient ausschließlich dem Zweck, Syntax- und Verphrasungsregeln besser zu erläutern und zu visualisieren. SAP übernimmt keine Gewährleistung für die Richtigkeit und Vollständigkeit des Beispielcodes. SAP übernimmt keine Haftung für Fehler oder Schäden, die durch die Nutzung des Beispielcodes verursacht wurden, es sei denn, dass diese Fehler oder Schäden von SAP grob fahrlässig oder vorsätzlich verursacht wurden.

## Geschlechtsneutrale Sprache

Sofern möglich, wird geschlechtsneutral formuliert. Je nach Kontext und zur besseren Lesbarkeit kann SAP die männliche Flexionsform verwenden, um sich auf alle Geschlechter zu beziehen.

## Videos, die auf externen Plattformen gehostet werden

Einige Videos verweisen möglicherweise auf Video-Hosting-Plattformen von Drittanbietern. SAP kann die zukünftige Verfügbarkeit von Videos, die auf diesen Plattformen gespeichert sind, nicht garantieren. Außerdem unterliegen alle Werbungen und anderen Inhalte, die auf diesen Plattformen gehostet werden (z.B. empfohlene Videos oder Navigation zu anderen gehosteten Videos auf derselben Site), nicht der Kontrolle oder Verantwortlichkeit von SAP.



© 2020 SAP SE oder ein SAP-Konzernunternehmen Alle Rechte vorbehalten.

Weitergabe und Vervielfältigung dieser Publikation oder von Teilen daraus sind, zu welchem Zweck und in welcher Form auch immer, ohne die ausdrückliche schriftliche Genehmigung durch SAP SE oder ein SAP-Konzernunternehmen nicht gestattet. In dieser Publikation enthaltene Informationen können ohne vorherige Ankündigung geändert werden.

Die von SAP SE oder deren Vertriebsfirmen angebotenen Softwareprodukte können Softwarekomponenten auch anderer Softwarehersteller enthalten. Produkte können länderspezifische Unterschiede aufweisen.

Die vorliegenden Unterlagen werden von der SAP SE oder einem SAP-Konzernunternehmen bereitgestellt und dienen ausschließlich zu Informationszwecken. Die SAP SE oder ihre Konzernunternehmen übernehmen keinerlei Haftung oder Gewährleistung für Fehler oder Unvollständigkeiten in dieser Publikation. Die SAP SE oder ein SAP-Konzernunternehmen steht lediglich für Produkte und Dienstleistungen nach der Maßgabe ein, die in der Vereinbarung über die jeweiligen Produkte und Dienstleistungen ausdrücklich geregelt ist. Keine der hierin enthaltenen Informationen ist als zusätzliche Garantie zu interpretieren.

SAP und andere in diesem Dokument erwähnte Produkte und Dienstleistungen von SAP sowie die dazugehörigen Logos sind Marken oder eingetragene Marken der SAP SE (oder von einem SAP-Konzernunternehmen) in Deutschland und verschiedenen anderen Ländern weltweit. Alle anderen Namen von Produkten und Dienstleistungen sind Marken der jeweiligen Firmen.

Zusätzliche Informationen zur Marke und Vermerke finden Sie auf der Seite <https://www.sap.com/germany/about/legal/trademark.html>.