# Typographic Conventions

<table>
<thead>
<tr>
<th>Type Style</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Example</strong></td>
<td>Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Textual cross-references to other documents.</td>
</tr>
<tr>
<td><strong>Example</strong></td>
<td>Emphasized words or expressions.</td>
</tr>
<tr>
<td>EXAMPLE</td>
<td>Technical names of system objects. These include report names, program names, transaction codes, table names, and key concepts of a programming language when they are surrounded by body text, for example, SELECT and INCLUDE.</td>
</tr>
<tr>
<td>Example</td>
<td>Output on the screen. This includes file and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.</td>
</tr>
<tr>
<td><strong>Example</strong></td>
<td>Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.</td>
</tr>
<tr>
<td><code>&lt;Example&gt;</code></td>
<td>Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.</td>
</tr>
<tr>
<td><strong>EXAMPLE</strong></td>
<td>Keys on the keyboard, for example, F2 or ENTER.</td>
</tr>
</tbody>
</table>
## Document History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>2014-12-12</td>
<td>First version created</td>
</tr>
</tbody>
</table>
# Table of Contents

1. **SAP Standards for E2E Solution Operations** ................................................................. 5  
   1.1 Control Center Approach ....................................................................................................... 6  

2. **Overview of the Standard for Test Management** .................................................................. 9  
   2.1 Principles of SAP Test Management ...................................................................................... 10  
   2.2 Functionality and Tools for Test Management ..................................................................... 11  
   2.3 Options for Test Management .............................................................................................. 11  
      2.3.1 Test Option 1 .................................................................................................................... 15  
      2.3.2 Test Option 2 .................................................................................................................... 22  
      2.3.3 Test Option 3 .................................................................................................................... 26  

3. **Lifecycle of Test Management** ............................................................................................ 27  
   3.1 Plan ..................................................................................................................................... 27  
      3.1.1 Define the Test Strategy ..................................................................................................... 27  
      3.1.2 Prepare the Tests ............................................................................................................. 28  
      3.1.3 Identify the Test Scope ..................................................................................................... 29  
   3.2 Build ..................................................................................................................................... 30  
   3.3 Run ....................................................................................................................................... 31  
      3.3.1 Execution and Documentation of Tests .......................................................................... 31  
   3.4 Optimize ............................................................................................................................... 32  

4. **Driving Continuous Improvement** ....................................................................................... 33  
   4.1 Quality Assurance Tasks ....................................................................................................... 33  
   4.2 Quality Targets and KPIs ....................................................................................................... 33  

5. **Training** ............................................................................................................................... 36  
   5.1 Expert Guided Implementation Sessions .............................................................................. 36  

6. **More information** .................................................................................................................. 38
1 SAP Standards for E2E Solution Operations

IT organizations face new challenges every day as they attempt to remain effective and future safe while also keeping costs for day-to-day operations as low as possible. They are also being challenged more than ever to demonstrate their value to businesses. Therefore, it is important to optimize the day-to-day tasks that have less obvious business value and to use KPI and benchmark-based reporting to make IT processes more visible, demonstrating the real value that IT can provide.

In order to minimize the costs of IT, it is necessary to standardize and automate IT processes end-to-end (E2E) without reducing the SLAs required by the business, such as stability, availability, performance, process and data transparency, data consistency, IT process compliance, and so on.

Based on the experience gained by SAP Active Global Support (AGS) while serving more than 36,000 customers, SAP has defined process standards and best practices to help customers set up and run E2E solution operations for their SAP-centric solutions.

The Build phase of SAP best practices supports a Build SAP Like a Factory approach, consisting of the following processes:

- Custom code management
- Change, test, and release management
- Incident, problem, and request management
- Solution documentation
- Remote supportability

During the Run phase of a solution, adapting your IT infrastructure to a Run SAP Like a Factory operation impacts both application operations and business process operations. Therefore, operations processes, such as technical monitoring, end-to-end root-cause analysis, technical administration, and data volume management need to be optimized to achieve state-of-the-art application operations. In business process operations, the same applies to business process and interface monitoring (including performance optimization), data consistency management, and job scheduling management.

Quality management processes and tasks need to be established throughout the lifecycle to guarantee continuous improvement of the end-to-end operations processes while simultaneously ensuring the flexibility needed to react to changing requirements.
This figure shows an organizational model for solution operations that aligns SAP best practice topics and E2E standards with SAP’s control center approach.

The Operations Control Center executes and controls the Run SAP Like a Factory processes, while the Innovation Control Center ensures optimal custom code management and a smooth transition to production with integration validation procedures. SAP connects to these control centers from the Mission Control Center to ensure that professional support is available to the customer. The following Application Lifecycle Management (ALM) functions are not provided directly in one of the control centers because they must be handled across different areas:

- Change, test, and release management
- Incident, problem, and request management
- Solution documentation
- Remote supportability

The quality management methodologies are an essential part of SAP’s Advanced Customer Center of Expertise (CoE) concept and ensure that the KPI-driven processes are continuously improved across all processes and teams. In addition, the quality manager roles ensure consistent and value-centric reporting to the business and management. This unified reporting platform is known as the Single Source of Truth.

### 1.1 Control Center Approach

The Operations Control Center (OCC) is the physical manifestation of the Run SAP Like a Factory philosophy. The OCC allows for automated, proactive operations, which simultaneously reduces operational costs while increasing the quality of IT services, leading to improved business satisfaction. The OCC also drives continuous improvement of business processes and IT support. To achieve these goals, it relies on a close interaction with both the Innovation Control Center (ICC) and the SAP Mission Control Center (MCC).
The OCC is a central IT support entity at the customer site, which monitors the productive SAP environment as well as important non-SAP applications. During operation, the OCC requires a workforce of 2 full-time equivalents (FTE) per shift to ensure that incidents are detected and resolved as quickly as possible. The OCC is equipped with large screens that display the status of business processes, IT landscape components, as well as exceptions and alerts. If problems occur, you use a video link to get live support from SAP and partners. The customer usually sets up the room with assistance from SAP Active Global Support (AGS). The customer is responsible for managing the OCC and the team of technical and functional IT operators who act on the alerts.

The OCC is most effective when closely integrated with other IT processes, such as IT Service Management (ITSM) and Change Management. Central monitors and dashboards based on application and business process operations display the current status of business and IT-related processes. This data can also be used to drive continuous improvement.

An effective system monitoring and alerting infrastructure is fundamental to the success of an OCC.
The OCC is most effective when closely integrated with other IT processes, such as IT Service Management (ITSM) and Change Management. Central monitors and dashboards based on application and business process operations display the current status of business and IT-related processes. This data can also be used to drive continuous improvement.

An effective system monitoring and alerting infrastructure is fundamental to the success of an OCC. For Job Scheduling Management, the OCC supervises all background monitoring processes, SAP controls and legacy background operations. It reacts to job monitoring alerts according to predefined error-resolution activities, and triggers follow-up activities for error handling if the relevant task are not completed within a certain timeframe.

Figure 3: OCC Concept
2 Overview of the Standard for Test Management

This document describes the main guidelines and tools provided by SAP to improve the level of test management for SAP customers. Given that SAP Business Suite is used every day by thousands of companies, it should, theoretically contain very few errors. However, the way SAP and, above all, SAP customers coordinate and make changes to SAP solutions that have already been implemented means that it is important to perform effective tests, even on business processes modeled with SAP solutions.

Most SAP software is used to map highly integrated and often complex business processes. When SAP standard software is initially implemented, it is adapted to accommodate customer-specific business processes by performing the following basic activities:

- Configure business processes and how they are managed, for example, pricing.
- Enter the master data and attributes that are needed to execute business processes, for example, vendors, materials, services, and customers.
- Implement and configure software from SAP partners or third parties and integrate it with SAP solutions using standard or customer-specific interfaces.
- Where appropriate, the end users choose the user interface for running business processes.
- If required, create and implement composite applications to handle extremely customer-specific processes. These applications are based on the enterprise service-oriented architecture (SOA) and use Web services to communicate with SAP Business Suite applications.
- If required, include custom code in SAP standard solutions to map extremely customer-specific functions or processes.

For the majority of SAP customers, the outcome of these activities is a comprehensive solution that encompasses complex interactions between SAP, partner, and third-party applications and interfaces, as well as custom code and customer developments.

Of course, the overall solution at each SAP customer is unique. Therefore, the solution must be tested thoroughly before going live. Such testing includes tests of individual functions, integration tests between SAP and non-SAP applications, scenario tests in which entire business processes are executed from beginning to end, and performance and acceptance tests by end users.

After going live, there are a number of reasons why SAP customers need to make changes to their SAP-centric solution. In many cases, these changes result from maintenance work or functional enhancements of business processes that are required by different departments. These changes are the main cause of further testing activities subsequent to going live.

The required tests cannot be performed entirely by the software manufacturers because all tests must take into account the customer environment with its configuration, master data, and interfaces.

Typical challenges and pain points with regards to testing include the following:

- Functional and integration tests in heterogeneous system landscapes
  - Integration testing for business processes that require several SAP applications and system instances
  - Testing of interfaces that connect SAP solutions with third-party applications
- Lack of analysis concerning the impact of changes
Changes to SAP solutions, for example, configuration changes, often impact a number of critical business processes without the application management being aware.

Due to a lack of options for analysis, these effects cannot be detected, which prompts many companies to delay or avoid necessary adjustments, for example, activating SAP support packages.

The task of defining the required test coverage is also severely hindered by the lack of analysis opportunities. As a result, test sets analyze unchanged areas and changes are often made without all critical processes first being covered by a regression test.

- Provision of test environments and test data
  - Test environments should resemble the production environment and reflect the changes to be tested as closely as possible. Due to the large volume of data and sensitivity of the data, test environments cannot be created simply by making a copy of the production system. Consequently, many companies’ test systems contain obsolete configurations and data.
  - Problems are compounded by the fact that providing suitable test data for manual and automatic tests can be difficult. For example, the customer credit-rating check in the order-to-cash process can only be carried out if special data conditions exist.

- Test automation
  - While automated tests can reduce the number of manual testers, they require specially trained experts with knowledge and experience of the testing tools to reproduce the required dynamics.
  - Automated tests demand high levels of maintenance. The effort varies depending on the tool because changes to the SAP solution often damage the automated test cases.

SAP provides the following solutions to these test management challenges:

- Testing is included in the methodology for implementing, operating, and adapting SAP-centric solutions.
- The methodology and the test management processes are based on best practices.
- SAP Solution Manager and integrated partner applications provide test capabilities.

### 2.1 Principles of SAP Test Management

SAP’s strategy for test management is guided by the following key principles that influence the existing level of test functions and their further development:

- **SAP Solution Manager as the central platform for test management**
  SAP Solution Manager as the central platform for E2E process testing delivers all major functions to handle all aspects of test management for SAP-centric solutions. This can be using either its own tools or those of integrated partner solutions.

- **Risk-based test planning**
  Risk-based test planning helps you to analyze the impact of SAP software changes on critical business processes and to provide adequate test plan proposals.

- **More options for SAP customers**
  Customers have the choice of using a combination of the test tools provided by SAP Solution Manager and test solutions from partners or third parties.
2.2 Functionality and Tools for Test Management

SAP Solution Manager provides a central platform with the following functions for handling all aspects of test management and related activities across the entire lifecycle:

- Documentation of business processes and assignment to SAP systems
- Dynamic analysis of the impact of software changes on SAP business processes
- Planning of manual and automated tests
- Test execution and documentation
- Handling of incidents and defects
- Monitoring of test execution
- Test status reporting
- Validation of integration tests

These functions are supplemented by a range of additional SAP products, such as:

- SAP Test Data Migration Server (TDMS)
- SAP Quality Center by HP
- SAP Test Acceleration and Optimization (TAO)
- SAP LoadRunner by HP

Most of these tools are highly integrated with SAP Solution Manager. Test applications from other suppliers, such as IBM Rational Software, can be integrated with the tools provided in SAP Solution Manager using existing interfaces.

2.3 Options for Test Management

Using the many tools available, there are a variety of options for testing SAP-centric business processes. The following options are proven and recommended by SAP:

- Option 1 uses functions and tools provided mostly by SAP Solution Manager and SAP Test Data Migration Server (TDMS).
- Option 2 uses functions and tools provided mostly by SAP Solution Manager, SAP TDMS, SAP Quality Center by HP, and SAP Test Acceleration and Optimization (TAO).
- Option 3 uses functions and tools provided mostly by SAP Solution Manager, SAP TDMS, and IBM Rational.
These options support the entire testing process using a variety of tools. However, all of the options are based on the initial foundation of the Business Blueprint, Business Process Change Analyzer (BPCA), and Test Data Migration Server (TDMS).

**Business Blueprint**

Using the Business Blueprint functions, you can design, document, and hierarchically catalog your business processes. The documentation is extremely important because it contains business information from the business units, technical information about logical and physical systems, and interface information. The business units can store their business requirements from a test management viewpoint. Later, you use the Business Blueprint to assign the manual and automatic test cases to the business processes. This provides a holistic view of the business processes. All technical and business aspects are entered centrally using the Business Blueprint and then adjusted centrally if changes are made.
For more detailed information about SAP Solution Manager Business Blueprint, see the *SAP E2E Standard for Solution Documentation*.

**Business Process Change Analyzer**

When a change is made in an existing SAP solution, important applications or critical business processes might be affected. Regardless of whether SAP source code is adjusted due to SAP maintenance packages, customer enhancements, or changes to the configuration or interfaces, it is crucial to identify which areas of the solution are affected by the change. SAP Solution Manager provides the Business Process Change Analyzer (BPCA) for precisely analyzing the SAP objects that are used when business processes are executed. In the event of an intentional change, the objects contained in the transport are compared with the technical bill of material (TBOM) of the target system to identify the affected business processes and areas. This helps the customer to decide whether to make the intended change as planned or a later point, for example, if there is not enough time to run a satisfactory regression test on all affected areas.

![Figure 6: SAP Solution Manager - Change Impact Analysis Approach](image)

To ensure the highest level of accuracy, the analysis is not performed statically on the source code; instead, it is performed dynamically at run time to record all the SAP objects used in the business process variant. The customer executes the critical business processes in an appropriate system and, as the processes are being executed, the BPCA records all the SAP objects (module pools, function modules, configuration and master data tables, interfaces, and so on) that are used and generates a technical bill of material (TBOM), which is then assigned to the business process using the Business Blueprint.
If SAP support packages, SAP enhancement packages, custom developments, or configuration adjustments might be implemented in the future, you can use the BPCA to identify business processes that could potentially be affected. You can save these analyses with a time stamp and use the detailed results for subsequent test applications.

**Recommendation**

SAP recommends assigning all test cases to the relevant business processes using the Business Blueprint. You can then automatically generate a test plan containing all affected processes on the basis of the BPCA – Test Workbench integration (Test Option 1) or the integration with HP or IBM tools as described in Test Option 2 and Test Option 3 respectively.

**SAP Test Data Migration Server**

SAP Test Data Migration Server (TDMS) accelerates the process of automatically building test systems and other, non-productive SAP systems. TDMS creates the target system using the configuration, master, and transaction data from a source system, such as the production system. In addition to creating an initial test system, you can also add data or replace data in an existing system.

**Note**

SAP TDMS is a separate product and, therefore, must be purchased separately.

As customers are keen to keep the volume of data in their test system to a minimum, creating a system using TDMS differs considerably from simply copying a system. For example, sensitive data should be avoided or at least modified. For this purpose, a number of different methods are used in TDMS. When the system is first built, the TDMS extracts repository and cross-client information to create a system shell. Client-specific configuration data and master data can then be transferred to the test system. A range of methods are used to reduce the transaction data. For example, you can limit the transaction data that is transferred based on one or more periods or organizational units. The flexible TDMS framework also supports customer-specific methods. For sensitive data, for example from SAP HR, you can apply preconfigured data anonymization methods.
Once the initial build is complete, TDMS allows you to update specific items of data. You can select documents from the source system and copy them to the target system together with all preceding documents. This means that you can perform integration tests even for complex business processes. TDMS is designed to achieve high data throughput and performance levels while putting as little strain as possible on the source system.

**Note**

In all options, SAP Solution Manager plays a pivotal role in bringing together technical and business aspects. All options use comprehensive test management functions that offer optimal support for coordinating and executing tests.

To achieve the greatest levels of efficiency and transparency, companies should choose the option most suited to their particular situation and use all of the tools provided.

### 2.3.1 Test Option 1

Almost all of the functions and tools for testing offered by Test Option 1 are provided by SAP Solution Manager and SAP Test Data Migration Server (SAP TDMS).

**Note**

SAP TDMS is a separate product and, therefore, must be purchased separately.
You determine the test focus based on the planned changes. You can then identify the test cases and create test plans.

**SAP Solution Manager Test Workbench**

In Test Option 1, you use the SAP Solution Manager Test Workbench to create test packages and assign them to the correspondent tester.

When designing and documenting business processes, you can store requirements that are relevant for creating test cases. For example, you can create test documents and instructions to help manual testers perform tests. These documents can be entered directly as text or created in a word processing program and uploaded to the test case. To direct the tester straight to the relevant transaction and system, you can also include test objects. In addition, each test case can be linked with the business requirements, which means that you can check that all requirements have been covered in each test case.

> **Recommendation**

SAP recommends assigning all test cases directly to the business processes. When the design stage is complete, you can configure the business processes in SAP Solution Manager.

All relevant test cases are grouped together in one or more test plans for a test cycle. You can create test plans manually by selecting the appropriate test cases. Alternatively, the test plan can be generated automatically using the business process hierarchy or test case attributes (for example, keywords). For monitoring and reporting reasons, several test cases can be organized in a test series.

The test cases are assigned to testers using test packages. Like test plans, you can either create the packages manually from the test cases found in the test plan or they can be generated by selecting test case attributes. One or more testers can be assigned to each package.

**Figure 9:** Capabilities of Test Option 1
Executing tests on complex process chains often requires different testers with different knowledge of the processes. For this reason, you can create test sequences for packages and define which test cases are performed by which testers. When each step of a test sequence is complete, the next tester is notified by e-mail that they can begin testing.

To ensure that your internal requirements are met and enable a formal release procedure, you can use a status release schema with the Test Workbench. In this schema, you can specify that test plans and test packages may
only be used once they have been explicitly released. Once test activities have been completed, the test plan and test packages can be locked.

![Diagram](image)

**Figure 12:** SAP Solution Manager Test Workbench — Release Status Schema

You can manage any defects or incidents that occur during testing in SAP Solution Manager using the integrated IT Service Management (ITSM) component. The test case and incident message are linked to each other and can be called directly from status reporting for further analysis.

The Test Workbench provides a wide range of hierarchical and list-based reports for monitoring the status. Test coordinators can use these reports to track the progress of testing in real-time, even if tests are being conducted in different locations around the world. It is possible to analyze the number and priority of incidents that have occurred. Customer-specific status reports, drilldown analyses, and dashboards are enabled using the new integration of SAP Solution Manager Test Workbench and SAP Business Intelligence (SAP BI).

**Test Automation Framework**

The Test Automation Framework (TAF) helps you to create and execute automated test cases. The TAF is integrated into the SAP Solution Manager System Landscape (LMDB), complementing the Extended Computer Aided Test Tool (eCATT) and SAP Component Based Test Automation (CBTA). This simplifies creating test configurations, test data containers, and system data containers.

Just like manual tests, automated tests using the TAF can be started from the test package. Once the test has been executed, the application automatically sets the test status. You manage the extensive test logs in the Test Workbench. The logs can be called directly from the status reports.
The Test Automation Framework supports multiple third-party test automation tools from certified independent software vendors (ISVs). For a list of certified ISVs, see the SAP Partner Directory at: http://www.sap.com/ecosystem/customers/directories/SearchSolution.epx.

This provides considerable advantages throughout the entire lifecycle of test cases because you can create, manage, maintain, execute, and evaluate the results of the test cases in one environment. For instance, automated test scripts can be created from CRM or Web shop applications using partner applications that are invoked at an appropriate place and time with an eCATT or CBTA test script.

In addition, the TAF provides the functionality for scheduling automated test for a specific time, for example, at night time (light-out tests), and on remote machines (either physical computers or virtual machines).
Component Based Test Automation

In Test Option 1, the SAP Component Based Test Automation (CBTA) tool is used to create automated test cases. For testing user interfaces that are not based on SAP technology, CBTA must be supplemented with tools from partners and third parties using the Test Automation Framework. These tools enable you to create automated tests for extensive, system-wide, and heterogeneous business process chains. You can also test Web services and their use in composite applications. The following functions can be used to plan automated tests with CBTA:

- **CBTA Script**
  
  To generate the initial test script, the user executes a business process while CBTA records which SAP objects are used and which activities performed by the user. This record serves as foundation for compiling a test script. You then add parameters so that test data can be used dynamically and transferred between process steps. CBTA scripts enable you to perform extensive checks of the test results. For example, after a test is complete, the accuracy of data posted in the respective SAP tables can be verified using CBTA check functions.

- **Test Data Container**
  
  Test data for complex business process scenarios, for example, order-to-cash, can be planned flexibly with the Test Data Container. It also allows business process variants to be executed because the creation of variants is often mapped to multiple test data records based on variations in test data.

- **System Data Container**
  
  CBTA scripts are very flexible and can be used for a range of system landscapes because they are cataloged independently of systems. The system data container reveals the logical and physical system in which the automated tests will be executed.
Figure 16: SAP Solution Manager CBTA – Flow for Creating Automated Test Scripts

- Test Composition Environment (TCE)
  The TCE provides all functions to create and maintain CBTA test scripts and test configurations. It enables you to create attributes for test configurations and test scripts and to compose E2E process tests, maintain parameters, and assign test data.

SAP Solution Manager Test Management Work Center

SAP Solution Manager supplies a range of tools for planning, executing, and analyzing functional tests. The Test Management work center enables Web-based access to all test management functions. This means the functions relevant for each role are accessible through a single point of entry.
2.3.2 Test Option 2

The functionality and tools for testing offered by Test Option 2 are provided by SAP Solution Manager, SAP TDMS, SAP TAO, SAP Solution Manager Adapter for SAP Quality Center by HP, and SAP Quality Center by HP.

1. Note

These are all separate SAP products for which additional charges may apply.
SAP Quality Center by HP

As in Test Option 1, the first step is to define and hierarchically catalog the business processes in the Business Blueprint. However, there is an additional key element. In Test Option 2, you can transfer the business process hierarchy and related information to the SAP Quality Center using the SAP Solution Manager Adapter for SAP Quality Center by HP. Related information can be documents, specifications, business requirements, links, and test objects, such as transactions, implementation guide activities, or custom programs. To do so, you need to add business requirements or test objects to the business processes before the transfer. These can then be expanded into test requirements in the Requirements module of SAP Quality Center.

On the basis of the processes that are defined and managed in SAP Solution Manager, quality managers use this data to plan and structure their test projects efficiently.

During test execution, test coordinators and quality managers gain insight into defects and incidents that occurred using the integration of Quality Center Defect Management with SAP Solution Manager IT Service Management (ITSM). Test results are also transferred back to SAP Solution Manager to allow test status reporting within the SAP Solution Manager environment.

Figure 19: SAP Solution Manager Adapter for SAP Quality Center by HP

**Recommendation**

SAP recommends using the Business Process Change Analyzer (BPCA) to analyze the impact of changes on business processes in SAP solutions. Using the BPCA, test coverage can be centered more precisely on the affected processes and components and you can generate test plans directly in SAP Quality Center by HP based on the BPCA results. It is also much easier to establish which automated tests could potentially be damaged by changes. The affected test components can then be regenerated rapidly using SAP Test Acceleration and Optimization (TAO).

As in the context of Test Option 1, TDMS facilitates the process of initially building a test system and then regularly updating it with test data that is representative of the business.

SAP Quality Center by HP contains all the functions needed to manage manual and automated tests. The Quality Center consists of the following modules:

- Requirements

  The Requirements module is where test requirements for the business processes can be stored. If you transfer the process hierarchy from SAP Solution Manager using the SAP Solution Manager Adapter for SAP
Quality Center by HP, you can use the requirement documents from the Business Blueprint as the basis for the test requirements. The hierarchy maintained in the Requirements module is used to generate test cases in the Test Plan module automatically.

- Business Components
  This module is important for creating component-based test cases for automatic testing with SAP TAO.

- Test Plan
  You create manual test cases in the Test Plan module. You can define a range of attributes for each test case and describe the flow and expected results for each script. In addition, individual test steps can be presented as lists. You have the option of assigning other documents, for example, if they contain details of test data to be used or screenshots of the application being tested. Every test case can be linked to the previously entered test requirements, which means that the extent to which requirements are covered remains transparent at all times.

- Test Lab
  The Test Lab module is used to define test cycles and execute the tests. You can select tests from the Test Plan module and present them in a completely new hierarchy. Using their assigned test cases, the testers can perform the tests in the respective SAP systems and enter the results directly in the test log, adding screenshots if necessary. All of the tests are carefully logged. The results are available immediately so that you can perform substantial list-based and graphical evaluations right away. This applies to both manual and automated test cases. If you are using the SAP Solution Manager Adapter for SAP Quality Center by HP, all test results and defects are transferred to SAP Solution Manager, where they can be processed further using the analysis tools.

- Defects
  If problems arise when a test is being performed, the Defects module can be used to enter incident messages directly. These messages are linked to the test case and can be assessed in the overview reports and detailed reports. Using the SAP Solution Manager Adapter for SAP Quality Center by HP, you can send these messages to ITSM in SAP Solution Manager, thereby ensuring that they can be further handled in an SAP context.

SAP Test Acceleration and Optimization

Many companies want to use automated test cases, in particular to handle regression tests, because automated testing reduces the number of manual testers while simultaneously increasing test coverage. Unfortunately, automated test cases are easily damaged by changes to the user interface or the process flows of SAP applications, which results in a considerable maintenance effort.

This is where integrating SAP Test Acceleration and Optimization (TAO) can help. TAO is a separate SAP product that offers functions to automatically create test components for SAP GUI-based transactions and then model test cases based on these test components. If the user interface is changed later, repairing the test case is simply a matter of regenerating the test component and entering test data.
Note

TAO is a separate product and, therefore, must be purchased separately.

SAP LoadRunner by HP

If important changes are made to an SAP solution, load tests should be scheduled to assess the following risks:

- System availability under load
- Functional stability under load
- Adequate end user response times
- Performance bottlenecks in the IT infrastructure and the applications

SAP LoadRunner by HP supports load tests in an SAP context for SAP GUID, Web Dynpro, HTTP(S), and e-mail protocols.

The basic process of performing load tests using SAP LoadRunner by HP is as follows:

1. Select the business processes for the load test.
2. Create test scripts to simulate user behavior.
3. Perform the load test with SAP LoadRunner by HP while gradually increasing the number of virtual users.
4. Evaluate the results by comparing the simulated performance with predefined target values.
5. If performance does not meet the requirements, adjust the corresponding infrastructure or applications.
6. Once the changes have been implemented, repeat the load test to verify the effectiveness of the changes.
2.3.3 Test Option 3

With SAP Solution Manager 7.1, a third test option is available that connects SAP Solution Manager with IBM Rational. As in Test Option 2, SAP Solution Manager still acts as the central platform where all information is stored.

SAP Solution Manager Connector for IBM Rational transfers business requirements documents and test objects to IBM Rational Quality Manager. The connector can also supply the applications of IBM Rational ClearQuest with service desk messages and supply IBM Rational Requisite Pro with business requirements from SAP Solution Manager.

Generally speaking, Test Option 3 integrates SAP Solution Manager with the IBM Rational suite in the same way as Test Option 2 integrates Solution Manager with SAP Quality Center by HP. For more information about IBM Rational and related tools, see the documentation provided by IBM.

Note

The tools provided by the IBM Rational suite are not SAP products and, therefore, must be licensed from IBM directly.
3 Lifecycle of Test Management

3.1 Plan

3.1.1 Define the Test Strategy

In general, software is tested to check its functionality and quality in order to find errors and to identify gaps. Effective testing ensures that software is handed over to the productive environment almost free of errors to avoid malfunctions. Testing should also verify whether the initial requirements specified by the business are adequately fulfilled by the solution provided by IT.

Experience with SAP customers has shown that the definition of a test strategy that meets the expectations of both business and IT is the starting point for all test related activities. An effective test strategy describes the approach for test management, provides guidance, and should describe at least the following aspects:

- Test organization principles and communication channels
- Test types that have to be considered in various change scenarios
- Roles and responsibilities for test management, such as business process expert, test coordinator, and tester
- Rules and guidelines for the documentation of business requirements, test cases, and reports
- Tools to be used to support the scoping, planning, execution, and analysis of tests, including defect and incident tracking
- General test start and exit criteria by test type

Depending on the change event and project itself, and in context with the company’s overall test strategy, different test types are used. Therefore, the terminology used in this context must be described first. SAP adopts the standard V-model for requirement-based testing.

Typically, the following test types are performed during implementation projects:

- Unit tests
  Module or object tests based on the technical design or specification.
- Functional tests
  Tests of single business process steps or transactions based on a functional design or specification.
- Integration tests
  Tests that examine the correct execution of business processes and scenarios, crossing various SAP applications and interfaces to partner or third-party party applications. Integration tests are based on the Business Blueprint documents.
- User acceptance tests
  Functional tests performed by key users of the changed functionality. Primary goals of user acceptance tests are to receive the formal acceptance of the business units and to identify usability problems.
- Regression tests
Tests that verify whether system changes affect the mission-critical business process behavior and their results. Usually, the effects of changes are known and can be predicted but sometimes the effects are not completely transparent.

- Technical developer tests
  Tests carried out by software developers at the lowest technical level during the Build phase with a focus on technical aspects, such as correct handling of interfaces.

- Technical system tests
  System tests are known in the SAP environment as technical system tests. Instead of looking at the functionality of the software in isolation, technical system tests check the entire system, including data consistency, databases, application servers, interfaces, network, and so on.

- Performance tests
  Technical system tests that measure the throughput and response times of the system.

- Security tests
  Tests that check user access and authorizations, segregation of duty (SOD), data security, and other security-related aspects.

3.1.2 Prepare the Tests

During the preparation phase, the test managers and their teams should adhere to the following basic process:

1. Review the test strategy in the context of the planned test project. Depending on the type of change, the suitable test types and approach must be determined and documented in a test concept.

2. Describe the goal of the test activities using a clear set of test objectives.

3. Specify the requirements for starting test activities, quality goals, and exit criteria to guide test preparation and manage expectations.

4. Identify the tools needed to support scoping, planning, execution, and analysis of the tests. This step depends largely on tools already licensed by your company.

   **Example**

   Although SAP recommends using Test Option 1, if your company already uses SAP Quality Center by HP as the standard tool for planning and execution, it might be better to integrate SAP Solution Manager as described in Test Option 2.

5. Plan the budget for the test project regarding financial resources, availability of manual testers, test systems, equipment, and so on.

6. Plan the important milestones for the project. For example:
   - Start and end dates
   - Quality gates
   - Deadlines for code freeze, emergency corrections, and final transport

7. Clearly identify and define all roles and their responsibilities, from which you can create a user role and authorization concept. A typical test project includes the following roles:
   - Test manager
   - Test coordinators
   - Manual testers
- IT support team
- Decision board members

8. If necessary, plan the relevant training for the tools based on roles.

3.1.3 Identify the Test Scope

The process for identifying the test scope differs depending on whether you plan to implement a new SAP solution or change an existing solution.

When a new SAP solution is implemented, for example, SAP CRM as an enhancement to an existing SAP ERP installation, a number of business processes are adapted, configured, and documented in the Business Blueprint. The test scope is derived from the list of business processes that have been adapted.

However, if you are changing an existing SAP solution, a different approach is required. After the planned change has been approved, an initial risk analysis determines the impact of the change on critical business processes. A second, more detailed risk analysis is then performed both during and after implementation of the change in the development system, and tests are run in the test system. The BPCA can support the change impact analysis by identifying the impact on critical business processes. This potentially reduces the amount of effort associated with testing and enables test resources to be targeted more precisely at risk areas. This process requires proper documentation of business processes and application objects in use. For more information about documentation and classification of business processes, see the SAP Standard for Solution Documentation.

Figure 22: Test Scope Identification
3.2 Build

Once you have identified the test scope for each individual test type, you can plan the actual tests. The typical basic process for the Build phase is as follows:

1. The user department collects the business requirements from the Business Blueprint. These can then be supplemented with test requirements by the QA department.
2. Based on these requirements, create manual and automated test cases, or adapt existing ones.
3. Assign all test cases to business processes of the Business Blueprint or topic-related test catalogs.
4. Identify suitable test cases and collect them in a test plan. These can be generated using the BPCA so that test resources are only targeted at areas that have been changed.
5. If necessary, define the test sequences.
6. Assign test cases to testers.
7. Equip all manual testers with a handout document that answers the following questions:
   - Where and how do I access my personal test package?
   - How do I access test case descriptions and the related test data?
   - How do I access the functionality of the test in the relevant test systems?
   - How do I document my test results?
     - What options do I have to set the status for results documentation and how do I make the settings?
     - How do I report and document incidents and defects?
   - How is the test organized (rooms, timeframe, and so on)?
8. Assign authorizations to all manual testers so that they can access the relevant systems and execute their test cases.
9. Update the tests systems so that they resemble the production systems but with the newly added changes already implemented. This often presents a major challenge because simply making a copy of the production system is not always possible for a variety of reasons.
10. Provide test data for the test cases.
    This is a complex task, especially when testing process chains involving multiple steps, which requires carefully coordinated configuration, master, and transaction data. Therefore, SAP recommends using SAP TDMS to improve efficiency when creating and maintaining test data.
11. Ensure that the following prerequisites for test execution are fulfilled:
    - The relevant system roles are defined and assigned to the related project and solution landscape.
    - The relevant objects for used executables (transactions, reports, and so on) are assigned to a functional process structure or E2E process structure.
    - The relevant test scripts (manual or automated) are assigned to test objects.
3.3 Run

3.3.1 Execution and Documentation of Tests

Once the tests have been planned and test data is available on the test systems, testing can begin. The typical basic process for the Run phase is as follows:

1. Software developers perform unit tests in the development systems. Depending on the type and scope of the test cycle, various functional tests are performed.
2. Manual testers are provided with the tester handout document and receive details regarding their test package by e-mail.
3. Automated tests are scheduled or started directly.
4. Every test that is executed is logged and documented with test notes and a test status is set manually or automatically.
5. If the system responds in an unexpected way during manual testing, for example, if an error message appears, the tester records the incident in the corresponding ITSM system, attaching screenshots, log messages, and so on. Usually, this also has to be done manually even for automated tests.
6. The incident is sent to the persons responsible for the analysis and categorization of defects, who then correct the defect in the development system.
7. The correction is transported to the test system according to the existing arrangements and timelines, where it is then retested.

Integration Validation

Given the complexity and heterogeneity of modern software solutions, SAP recommends performing integration validation, especially for important business processes. This involves gathering and subsequently evaluating a substantial amount of data from the software applications that are active while a given business process is being executed. This type of validation also allows you to identify the hidden warnings and error messages that frequently occur at the interfaces between applications.

Load Testing

If large-scale changes are made or new software solutions are implemented, load tests should be performed before these are used in production. These tests simulate a situation in which the expected load (known number of users and background load in a load-peak situation) is simulated. While doing so, system behavior in handling large data volumes can be inspected. Throughout the entire test cycle, test coordinators monitor the test status and progress, as well as the processing status of incidents that have been reported.

Deployment

Once all tests and criteria of the test concept are met, the quality gate can be scheduled. This meeting brings together all of the parties responsible, who evaluate the outcome of the tests and decide whether to release the changes to the production system. If they reach a positive decision, the solution is deployed in the production system.
3.4 Optimize

The main advantage, and at the same time the main disadvantage, of manual testing is the fact that it is performed by people. This is an advantage in that experienced key users in your organization are able to give you valuable feedback regarding the system under test, for example, concerning the usability of the system or the completeness of the documentation. However, the human factor can also be a disadvantage, not least because you have to recruit your team of testers, give them an induction into the project, provide them with support, and so on. People also make mistakes during their tests, leading to defect reports where you cannot always be sure whether the problem was caused by the software or the tester.

Given the disadvantages of manual testing, many people involved in the test process have looked to automated testing to support them during regression tests. Automated testing has some advantages that are immediately evident: for example, the test case can be run quickly, reliably, and repeatedly at the click of a button and fewer people are needed during the execution phase of a test project if at least some of the tests are automated.

However, there are drawbacks to automated testing that must be taken into consideration. First of all, there are the licensing and maintenance fees for test automation tools, not to mention the training costs for the test automation team. A further aspect of automated testing that is underestimated by many is the fact that creating automated tests often takes longer than creating a similar manual test case. The increased speed of automated testing compared with manual tests is found exclusively in the repeated execution of the tests.

Therefore, test automation is only beneficial if you create reusable test cases. This requires the clear modeling of an overall test approach before you start developing automated test cases. Despite the initial overhead associated with automated testing, there are considerable savings to be made over the course of several test cycles, as long as the test automation is restricted to test cases that can be reused in more than one scenario. For example, a posting inbound goods test case could also be used to test an order processing scenario or warehouse management.

However, it is important to note that complete test automation is unlikely to be a realistic goal in a test project. There will always be test cases that do not need to be reused, or where automation is hard to achieve, for example, because of the user interface of the application. Especially in the early stages of test automation, it is better to set a modest automation goal and expand the efforts in a later project.
4 Driving Continuous Improvement

It is important that your organization learns from each test cycle so that your tests become increasingly effective and efficient. This section describes some key steps you can take to drive continuous improvement and increased value for your test management processes.

4.1 Quality Assurance Tasks

From a quality management perspective the key tasks are as follows:
- Ensure quality of testing (plan, scope, and execution) to mitigate risks for business disruption in production within appropriate timeline and cost
- Evaluate quality of testing activities (for example, number of test cases that pass the quality gate)
- Evaluate test execution time and degree of test automation to identify potential efficiency improvements
- Track the number of incidents after going live that are related to test management issues. Based on the analysis, define a continuous improvement plan for test management.
- Ensure adequate documentation quality within a Single Source of Truth

4.2 Quality Targets and KPIs

To ensure continuous improvement of the test management process and drive the value recognition of IT, the most important quality targets are as follows:

To assess the quality of the test management process, clearly defined parameters and measurable objectives are required. The key parameters should be collated and evaluated in regular reports. The historical data that is created in this way can be used to identify trends and then derive the necessary measures to take.

The following table describes the main challenges for each of these quality targets and which KPIs can be used to measure them:

<table>
<thead>
<tr>
<th>Quality Targets</th>
<th>Main challenges</th>
<th>KPIs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deliver business innovations faster</td>
<td>• Speeding up test scope identification, test execution, and defect management</td>
<td>• Average time and effort spent for test scope identification</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Average time and effort spent for test execution</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Average time and effort spent for defect management</td>
</tr>
<tr>
<td>Improve stability and reliability of business processes and reduce business risks</td>
<td>• Reduce the number of critical defects and incidents that occur in the production environment by setting up your system</td>
<td>• Percentage of incidents caused by unknown issues after go-live</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• A high percentage of incidents after go-live indicates that you</td>
</tr>
<tr>
<td>Quality Targets</td>
<td>Main challenges</td>
<td>KPIs</td>
</tr>
<tr>
<td>-----------------</td>
<td>----------------</td>
<td>------</td>
</tr>
<tr>
<td>test management processes so that as many as possible are addressed during the test phase.</td>
<td>need to assess the quality of test scope, test cases, and defect management. These KPIs can be based on the number, nature, and severity of defects entering production system.</td>
<td></td>
</tr>
<tr>
<td>• Prioritizing incidents and following up on them accordingly</td>
<td>• Percentage of test case coverage in relation to incidents arising after go-live. A low test coverage and high number of incidents after go-live indicates that you need to increase the scope of your test cases.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Ratio of critical business processes steps tested for performance regularly to identify issues in solution transition quality.</td>
<td></td>
</tr>
<tr>
<td>Increase efficiency</td>
<td>• Ensuring quality while reducing costs</td>
<td>• Automation helps reduce testing timelines and increase test case coverage. From a QM perspective, reusability is also a factor.</td>
</tr>
<tr>
<td></td>
<td>• Increasing speed This can be achieved by replacing costly, time-consuming manual testing processes with automated tools, minimizing the required resources for both IT and business, while still ensuring that key issues are still addressed. Reusable scripts, a cohesive test framework, and automated test processes result in quicker benefits and reduced costs.</td>
<td>o Ratio of automated test cases to total test cases performed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o Percentage of test cases that are reusable</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Quality process that includes updated test cases to reduce the amount of incidents in the production environment</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o Percentage of test cases reviewed or changed after errors are identified</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o Number of new functional or technical defects identified during integration test phase</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o Percentage of test processes executed according to policy</td>
</tr>
<tr>
<td>Quality Targets</td>
<td>Main challenges</td>
<td>KPIs</td>
</tr>
<tr>
<td>----------------</td>
<td>----------------</td>
<td>------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o  Percentage of reported incidents that were not in test scope</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o  Percentage of reported incidents that passed the quality gate for production</td>
</tr>
</tbody>
</table>
5 Training

SAP offers the following training courses for Test Management:

- SMI310 SAP Solution Manager: Implementation Tools in Detail
- CA611 Test Management with eCATT
- TDMS10 Implementing and Configuring SAP TDMS 4.0
- TDMS20 Using SAP TDMS for BPL, HCM and Data Scrambling
- E2E220 End-to-End Test Management Overview
- E2E060 RunSAP Customer Center of Expertise - QM

5.1 Expert Guided Implementation Sessions

Expert guided implementation (EGI) is a new delivery methodology. The methodology balances the combination of training, practical experience, and expertise on demand. The focus is to enable the customer to execute complex activities with the help of SAP experts. It is important that during the delivery itself, the activity is executed as described in the delivery slot. Examples activities include performing an update, building a Customizing, or executing a service.

Expert guided implementation enables you to execute activities without being a subject matter expert. The methodology closes the gap between classroom training and consulting. Expert guided implementation provides guidance for the execution phase of your project.

The goal is to execute all relevant steps that are necessary to complete a defined activity during the delivery time of expert guided implementation. After the delivery, the targeted activity should be complete.

For an overview of available EGIs, see the SAP Enterprise Support Academy on Service Marketplace at http://service.sap.com/esacademy.
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Upgrade to 7.1</td>
<td>Solution Documentation I: Basic, Initial Upload, Verification</td>
<td>Test Management II: SAP Test Workbench</td>
<td>Change Control Management II: CTS+</td>
<td>Incident, Problem and Request Management (ITSM)</td>
<td>Business Process Monitoring and Stabilisation</td>
<td>PI Monitoring &amp; Troubleshooting</td>
<td>Authorization Concept &amp; Roles</td>
</tr>
<tr>
<td>Migration Charm 7.0 to 7.1</td>
<td>Solution Documentation II: Reverse Process Documentation</td>
<td>Test Management II: Business Process Change Analyzer</td>
<td>Change Control Management II: Quality Gate Management</td>
<td></td>
<td>Business Process Analytics and Improvement</td>
<td>BI Monitoring &amp; Troubleshooting</td>
<td>Configuration Validation</td>
</tr>
<tr>
<td>Basic Configuration</td>
<td>Solution Implementation</td>
<td>Test Management III: Automation with HPQC</td>
<td>Change Control Management III: Change Request Management</td>
<td></td>
<td></td>
<td></td>
<td>Custom Development Mgmt Cockpit</td>
</tr>
<tr>
<td>SAP BusinessObjects BI (4.0) Basic Configuration and Setup EWA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Software Quality and Clones</td>
</tr>
<tr>
<td>Setup Remote Support Component for SAP Business Objects</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 23:** Overview EGI
## 6 More information

<table>
<thead>
<tr>
<th>Documentation</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>IT Service Management in SAP Solution Manager</td>
<td>SAP PRESS published by Galileo Press</td>
</tr>
<tr>
<td>Application Lifecycle Management on SAP Service Marketplace</td>
<td><a href="http://service.sap.com/aim">http://service.sap.com/aim</a></td>
</tr>
<tr>
<td>SAP E2E Integration Testing</td>
<td><a href="http://service.sap.com/testing">http://service.sap.com/testing</a></td>
</tr>
<tr>
<td>SAP TAO</td>
<td><a href="http://service.sap.com/saptao">http://service.sap.com/saptao</a></td>
</tr>
<tr>
<td>SAP Solution Manager</td>
<td><a href="http://service.sap.com/solutionmanager">http://service.sap.com/solutionmanager</a></td>
</tr>
<tr>
<td>SAP Solution Manager Enterprise Edition</td>
<td><a href="http://service.sap.com/enterpriseedition">http://service.sap.com/enterpriseedition</a></td>
</tr>
<tr>
<td>SAP E2E Solution Operations</td>
<td><a href="http://service.sap.com/e2e">http://service.sap.com/e2e</a></td>
</tr>
<tr>
<td>Run SAP</td>
<td><a href="http://service.sap.com/runsap">http://service.sap.com/runsap</a></td>
</tr>
<tr>
<td>SAP Enterprise Support</td>
<td><a href="http://service.sap.com/enterprisesupport">http://service.sap.com/enterprisesupport</a></td>
</tr>
<tr>
<td>SAP Support Services</td>
<td><a href="http://service.sap.com/supportservices">http://service.sap.com/supportservices</a></td>
</tr>
</tbody>
</table>