



### Integration PM/TM

**Integration of TM into PM**

The test management framework (TMF) integrates seamlessly into the project management framework (PMF) and forms the basis for the software development project test activities. The test management framework supports the test activities carried out during the course of projects by specifying a fundamental testing methodology, providing useful instructions and laying down unified procedures and definitions. To maintain a clear distinction between the tasks and responsibilities of PMF and TMF we have created a responsibilities matrix which specifies and describes the various roles and overall responsibilities.

**Responsibilities matrix for the various project roles**

	Project head	Test manager	Test consultant	Tester	Test auditor	Developer
1 Test initialisation	Overall responsibility	Execution	Execution	Assistance	Assistance	Assistance
2 Test preparation	Assistance	Execution	Execution	Information	Assistance	Assistance
3 Test execution	Assistance	Execution	Execution	Information	Assistance	Assistance
4 Test approval	Assistance	Execution	Execution	Information	Assistance	Assistance
5 Test completion	Overall responsibility	Execution	Execution	Assistance	Assistance	Assistance

### Reporting

**Reporting Levels**

The standardisation of test management guarantees the efficient compilation of all information necessary in connection with a given test status. The various reporting requirements and reporting views can thereby be covered, right down to daily status reports. This ensures that the reporting offers material support for the coordination and steering of IT projects.

**Field**

- IT
- over-arching
- Program
- Program/Release
- Project/Release
- Sub-project
- Work package
- Test function/Service

**Tool support**

- Mobile Apps
- Web services
- MS-Report Services
- HP-QC/ALM and others

**Presentation level**

- Mobile Devices
- Presentation
- Reports
- Live analyses

### Resource planning

**Planning foundations**

- Infrastructure (back end)
- Tester (IT)
- Infrastructure (workplace)
- Tester (specific field)
- Test cases
- Project plan
- Test data

**Resource planning**

When planning resources for a test project various central points must be considered in order to ensure the successful execution of the testing:

- When selecting the testers all the skills required must be borne in mind
- The necessary test infrastructure must be available
- All test phases (integration test, overall integration test, acceptance testing, etc.) must be considered at the planning stage
- Complex testing chains (processes) must be borne in mind
- Planned software deliveries must be borne in mind

**Final result:**

### Test automation

**Process chains:** The automated testing of process chains with suitable tool support (e.g. LoadRunner©) facilitates the highest possible degree of comparability and measurability of test results.

**Stress-/Performance Testing:** Automated stress and performance testing offers a structured, repeatable approach to testing the performance of a system under stress.

**Regression testing:** To permit the testing of test objects which are not directly affected but which could be affected by the side effects of changes elsewhere, automated regression testing provides an efficient means of quality measurement.

**Unit testing:** Automated unit tests (e.g. JUnit) permit the detection of initial defects even during the development process and allow their correction before the actual test is carried out, thereby enhancing the stability of the test object during the ensuing test phases.

**Test data:** The automated compilation of test data allows us to arrive at a defined starting point for each test, on which all subsequent tests can be based in structured fashion.

**Tool support (examples):**

- LoadRunner
- Rational Functional Tester
- app.test
- JUnit

### Test management process flow

**1 Test initialisation**

During the test initialisation phase the foundations are laid for successful subsequent testing. The phase involves the evaluation, specification and preparation of the test content and scope, as well as the overall test environment.

**2 Test preparation**

To ensure successful test execution, the organisational premises are established during the test preparation phase. Painstaking test preparation is the key to the optimum achievement of the specified test objectives. The test preparation phase may be further broken down into the segments *Organisational*, *Test planning*, *Test design*, *Tester deployment planning* and *Test infrastructure*. This supports the various responsible parties in the structured performance of the tasks assigned to them, in handling the associated challenges and risks as well as in the targeted planning of resource deployment. Effective test preparation also comprises across-the-board risk management, which must be closely coordinated with the overall project management. Only then can we take suitable countermeasures to respond swiftly and proactively to any risks which may arise.

**3 Test execution**

Test execution comprises all activities to be carried out when processing the prepared test cases. This includes training the testers, proactive variation management and the entire field of reporting and documentation.

**4 Test approval**

The evaluation of approval criteria and degree of fulfilment of the objectives set at the test initialisation stage is conducted during the acceptance testing process. After ensuring that the practical and technical software preconditions for going live have been met, the decision to wind down project activities can be made at this point.

**5 Test completion**

During the test completion phase all activities conducive to the successful completion of testing are carried out. Alongside the compiling of a completion report, this also includes a final project completion meeting. Further activities include the archiving of the documents generated during the course of the project as well as knowledge transfer in the form of lessons learned and feedback.

**Activities and Deliverables:**

- Test initialisation:** Evaluation of start factors, Initialisation of test profile, Initialisation of action item list for the test, Initialisation of milestone planning, Initialisation of resource and capacity planning, Initialisation of costs and budget estimate, Initialisation of risk management, Test initialisation phase completion presentation, Assessment and evaluation of end factors.
- Test preparation:**
  - Organisational:** Evaluation of start factors, Establishment of organisational structure, Conducting kick-off sub-project test, Defining and detailing sub-projects, External supplier management, Providing tester manual, Preparing start of kick-off test, Risk list, Detailing risk management.
  - Test planning:** Specification of test planning, Finalising milestone planning, Finalising resource and capacity planning, Appointing test management team, Create RACI-Matrix, Finalising costs and budget estimate, Compiling test-specific reports, Establishing approval criteria, Devising test plan, Acceptance test plan, Test planning presentation.
  - Test design:** Detailed requirements, Identification affected processes and systems, Identification test scenarios, Identification test scenarios, Definition test data, Defining requisition to test environment, Identification regression effort, Assessing test coverage, Specified test case portfolio.
  - Tester deployment planning:** Devising test cases, Detailing tester deployment planning, Requisition and assignment of resources, Tester deployment planning, Arranging tester training.
  - Test infrastructure:** Test infrastructure, Planning and arranging test data, Installing and setting up test tool, Planning and appointing test laboratory, Planning and approving individual powers, Assessment and evaluation of end factors.
- Test execution:** Evaluation of start factors, Test start kick-off, Acceptance test, Execution of test, Steering test, Continuous variance management, Ongoing test documentation, Ongoing test status reporting, Test infrastructure management, Assessment and evaluation of end factors.
- Test approval:** Evaluation of start factors, Analysis of test results, Assessment of test results, Test acceptance testing, Assigning remaining items for roll-out, Arranging tester training.
- Test completion:** Evaluation of start factors, Compilation of project completion report, Drafting of lessons learned, Filing test results, Cutting back and dismantling test infrastructure, Assessment and evaluation of end factors.

**Key:**

- Activity (Blue circle)
- Critical activity (Red circle) (Criticality = priority, time or complexity)
- Start point (Blue circle)
- End point (Red circle)

### Role concept

Role <sup>1)</sup>	Project size			Function	
	Large	Medium	Small		
Test Manager	Execution	X	X	X	Responsibility for the overall testing process (as Test sub-project leader) or a defined part of the testing
Test analyst (practical and technical)	Execution	X	X		Responsibility for operational testing activities (all phases)
Tester	Information	X	X	X	Responsibility for operational test execution activities
Test auditor	Information	X			Formal acceptance testing of test cases
Test infrastructure coordinator	Execution	X			Responsibility for activities relating to testing environments
Test consultant	Execution	X	X		Advising test management
Defect manager	Execution	X	X		Management of reported variances, interfaces for implementation of test page
Dispatcher	Execution	X			Interface for test on implementation page, assignment of defects to developers
Developer	Execution	X	X	X	Provides working software and remedies defects

<sup>1)</sup> One person may simultaneously perform more than one of these roles.

### Setting objectives

**Motivations for standardisation**

Our software application life cycle requirements remain as high as they have ever been. However, in response to the ever-growing demand for the swift and timely production and launching of software, as well as permanently changing development processes, the need for structured testing grows ever greater. For instance, whereas in the past simple, sequential program functions determined the testing process, these days Web services, scrum-based models and wholly new application architectures with ever-evolving sourcing models dominate our testing requirements. Thus business processes are tested beyond applications, 'end-to-end', for which purpose new techniques, procedures, methods and tools are needed to facilitate the testing of a loosely coupled construct of Web services, possibly even one located in a public cloud. Today 'testing' means standardisation and managing complexity.

**Test management objective**

The objective of test management framework (TMF) is to put in place a binding, practically oriented framework for the testing process. The purpose of this framework is to safeguard software quality, lay down working guidelines for test management projects and provide structured support during all phases of the testing process. Its key components are:

- A standardised testing approach including the use of standardised terminology
- Specification of all testing activities to be carried out
- Ensuring the provision of all documentation necessary for such purposes as revision
- Efficient and effective deployment of resources
- Across-the-board tool support for all test activities, permitting the option of aggregating all testing information
- Definition of the test resourcing model (near offshore, test factories, in-house test organisation and quality management units)