Project Workflow Management[1]

Closing Process and User Acceptance Test [2]

By Dan Epstein

Note: *This article is based on the book Project Workflow Management: A Business Process Approach by Dan Epstein and Rich Maltzman, published by J Ross Publishing in 2014. The book describes PM Workflow® framework, the step-by-step workflow guiding approach using project management methods, practical techniques, examples, tools, templates, checklists and tips, teaching readers the detailed and necessary knowledge required to manage project "hands-on" from scratch, instructing what to do, when to do and how to do it up to delivering the completed and tested product or service to your client.*

The project workflow framework is the result of Dan's research into the subject, having the following objectives:

1. *Create the virtually error-free project management environment to ensure significant reduction of project costs*
2. *Reduce demands for highly qualified project managers using the step-by-step workflow guiding approach.*

While PM Workflow® is the continuous multi-threaded process, where all PM processes are integrated together, this article will attempt to describe the project closing group of processes as a stand-alone group that can be used independently outside of PM Workflow® framework. It will be difficult in this article not to venture into processes outside of the current subject, such as planning, quality, communications and other management processes, so they will be just mentioned. However, to get full benefit and the error free project management environment, the complete implementation of PM Workflow® is required. In order to understand how PM Workflow® ensures this environment, I strongly recommend reading my article Project Workflow Framework – An Error Free Project Management Environment*. in the PMI affiliated projectmanagement.com (*https://www.projectmanagement.com/articles/330037/Project-Workflow-Framework--An-Error-Free-Project-Management-Environment*)*

For more information, please visit my website www.pm-workflow.com

Please note, this is the last article in the series Project Workflow Management.

[1] *This series of articles is based on the book Project Workflow Management: A Business Process Approach by Dan Epstein and Rich Maltzman, published by J Ross Publishing in 2014. The book describes the PM Workflow® framework, a step-by-step approach using project management methods, practical techniques, examples, tools, templates, checklists and tips, teaching readers how to manage a project "hands-on" from scratch, including what to do, when and how to do it up to delivering a completed and tested product or service to a client.*

[2] How to cite this paper: Epstein, D. (2019). Closing Process and User Acceptance Test, Series on Project Workflow Management, *PM World Journal,* Vol. VIII, Issue IV (May).

Purpose

The Closing Frame is the frame in which the project ends. Its purpose is:

- Prepare environment, develop a test plan and execute acceptance test
- Develop training materials and conduct user training sessions
- Ensure that all project management processes and standards are followed
- Verify that all project deliverables are produced in accordance with the quality standards
- Keep records of all project events in order to avoid similar issues in future projects
- Roll out project to production or deliver all project materials to clients
- Release project resources
- Conduct and document Lessons Learned analysis
- Gather the Closing Frame statistics for the project management process improvement

Roles and Responsibilities

The following roles and responsibilities will be required in the Closing Frame:

Project Manager (PM) - responsible for ensuring all steps of the project management process are adhered to. The PM ensures that the delivery and the client teams are involved in all stages of the Closing Frame. The PM is responsible for the following Frame activities:

- Track the Frame implementation
- Manage the Acceptance Test Support Team activities
- Allocate the project team resources for the Acceptance Testing
- Close the project and reallocate all project resources
- Conduct the project Lessons Learned analysis
- Review the status on a weekly basis with the client and on monthly basis with the senior business and the senior delivery management. In some environments PM reports status of the project to the Senior Business Manager on a weekly, rather than monthly basis.

Client – This is a lead client and a single focal point to the business. The client must participate in all status review meetings scheduled by the project manager . The client is sometimes referred to as the Business Project Manager or the Business Area Lead. The client usually reports to the Senior Business Manager. During acceptance test client has the following responsibilities:

- Hold overall responsibility for development of the Acceptance Test Plan
- Schedule acceptance testing with all relevant project participants
- Obtain commitment of all resources needed for acceptance testing
- Ensure all steps of the testing procedure are followed and documented
- Ensure that all test scripts are documented

- Ensure that all defects are documented in the Acceptance Test Incident Recording Tool.
- Approve the Acceptance Test Completion Document
- Ensure that the Acceptance Test is completed according to the schedule, budget and quality.
- Resolve business issues, arising out of testing.
- Approve change requests arising out of testing.

Delivery Team Lead is responsible for coordinating Acceptance Test Support Team activities and allocating the project team resources for support of the acceptance testing

Test Manager is a member of the business team assigned by the Lead Client. The test Manager has the following responsibilities:

- Develop Acceptance Test Plan and ensure that it conforms to Test Strategy Document.
- Assure that the Test Strategy Document exists prior to Acceptance Testing
- Ensure that detailed test conditions or test scripts are produced
- Ensure that Entrance and Exit criteria are defined prior to Acceptance Testing
- Review Acceptance Test Plan with the Acceptance Test Team
- Regularly review test progress with the Acceptance Test Team
- Ensure that the Root Cause Analysis is performed for all defects found.
- Ensure that completion criteria are achieved for all tests before presenting the Acceptance Test Completion Document for signoff.
- Manage Test Cycles
- Manage the Resolution of Acceptance Test Team issues and queries
- Ensure that all defects and problems are reported immediately and followed up until they are resolved
- Participate in the daily defect review meetings with the Acceptance Test Support Team

Acceptance Test Team. Its members are assigned by the lead client from the business users' pool. They have the following responsibilities assigned to them by the Lead Client:

- Develop test cases, test scripts and obtain their approval by the Test Manager
- Develop test schedule/calendar, based on the Acceptance Test Plan for the assigned to them test cases
- Identify test data and assure that data is available and accessible during the acceptance test
- Execute test scripts
- Produce defect reporting
- Rerun test cases after correcting defects

Acceptance Test Support Team. This team is different from the Acceptance Test Team. Its members are selected from the delivery team pool. They have the following responsibilities assigned to them by a project manager:

- Provide timely support for the Acceptance Test Team
- Resolve defects
- Perform Root Cause Analysis for all defects
- Participate in the daily defect review meetings with the Acceptance Test Manager
- Provide test infrastructure
- Serve as supporting resources for running scheduled test
- Provide related test reports
- Provide interfaces to other systems according to the acceptance test plan

Test Coordinator is a member of Delivery Team, who receives defect notifications, assigns resources out of Test Support Team pool to fix defects, monitors incidents and assigns Root Cause Analysis tasks, as well as sending the defect resolution notification back to Test Manager. Test Coordinator is the focal point for communication with the Test Manager.

Senior Business Manager's responsibility is to review Monthly Status Report with the PM. Senior Business Manager usually reports to the Project Sponsor.

Senior Delivery Manager – person who owns a delivery budget, whose signature is required on the Statement of Work.

Project Sponsor is a major stakeholder. A Project Sponsor is responsible for the business success of the project, specifically, ensuring that the business objectives are met for which the project has been undertaken. The Project Sponsor is the owner of the overall project.

Inputs / Outputs

The Closing Frame processes interact with the rest of project frames via the entry and the exit points, as shown on the Closing Frame process flow diagram on Fig 16-1:

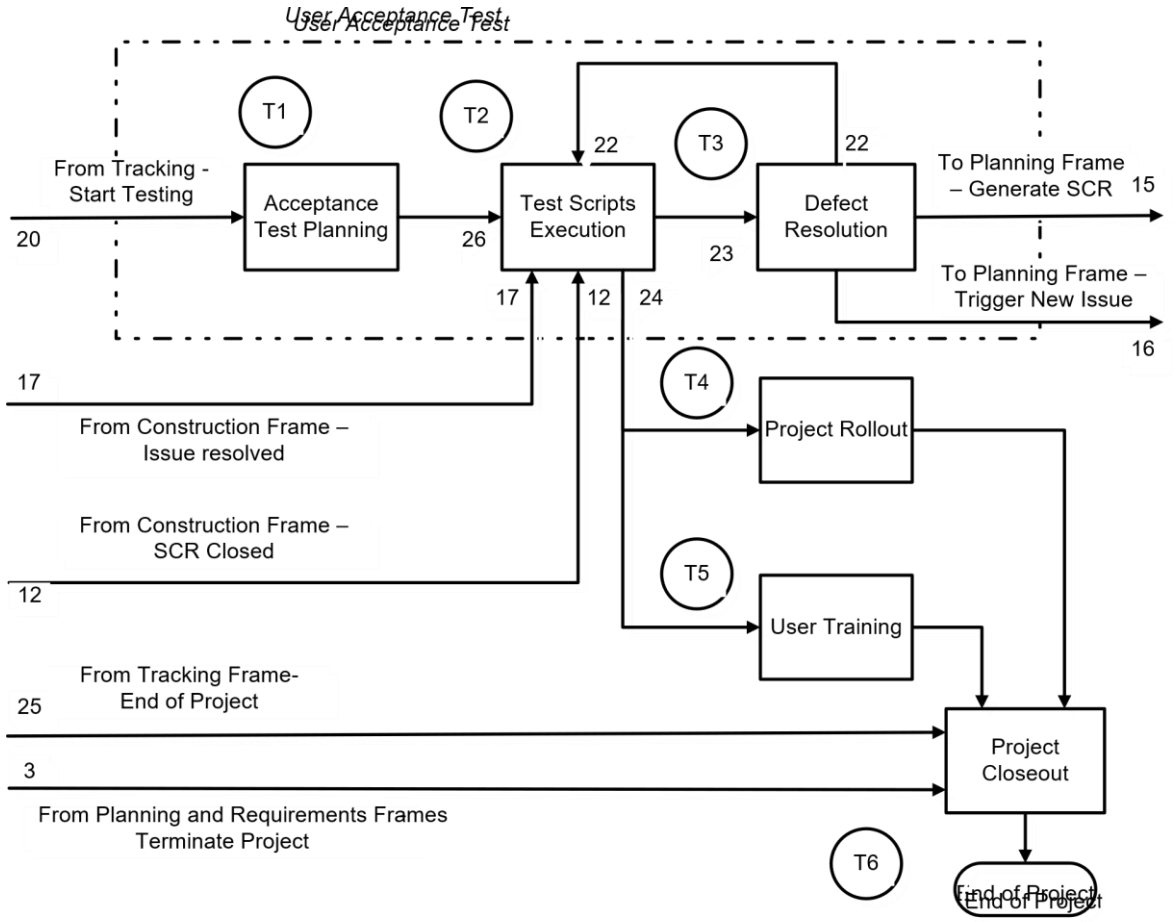


Fig 16-1 Closing Frame

Exit Point 15: To Planning Frame – Trigger scope change request
Exit Point 16: To Planning Frame – Trigger new issue
Entry Point 3: From Planning and Requirements Frames – Terminate project
Entry Point 12: From Construction Frame – SCR complete
Entry Point 17: From Construction Frame – Issue resolved
Entry Point 20: From Tracking Frame – Start acceptance test
Entry Point 25: From Tracking Frame – End of project

Closing Frame High Level Process Flow

The Closing/Testing Frame consists of the high-level processes as shown on Fig 16-1. They will be decomposed to the following detailed processes:

1. Acceptance Test Planning (T1).
   The purpose of this process is to describe steps necessary to run the acceptance test. The process includes test planning, test scripts development and the defect management planning. This is a business owned process.
2. Test Scripts Execution (T2)
   This process tests functionality of the developed product in accordance with the developed scenarios or scripts. This is a business owned process.
3. Defects Resolution (T3).

The Defect Resolution Process describes steps necessary to fix defect and produce Root Cause Analysis and steps needed for tracking of defects which occur before and after release of product into production. This is the delivery team owned process.

4. Project Rollout (T4).
   The purpose of this process is to establish methods and roll out the project to production environment or deliver the product to users.
5. User Training (T5).
   The purpose of this process is to identify training needs and establish user training.
6. Project Closeout (T6).
   The purpose of this process is to close project documentation, release resources and to analyze the finished project in order to identify successes and failures during the project execution and avoid their repeat in future.

The process is initiated when the request to start testing comes from the Tracking Frame via input 20. The Acceptance Test Planning process T1 will be executed first, where scripts are developed and required tests are scheduled. From there the process flow moves to the entry point 26 of the Test Scripts Execution process T2, where the next scheduled script execution tests the specific element of the product functionality to confirm compliance with the corresponding business requirement.

If a deviation from the expected results, also called an incident, occurred during the script execution, the process flow enters Defect Resolution process T3 via the entry point 23. When a defect resolution is completed, the process flow will loop back to the Test Script Execution process T2 to start execution of the same test script again to verify that the defect was resolved. In fact, as can be seen in the detailed process flow of the Defect Resolution process T3, the process flow may loop back to T2 even before the end of the defect resolution process.

If the scheduled script execution is complete without incidents, the next scheduled script will be executed, until there are no more scripts left to run, in which case, the user acceptance test is complete and the process flow via exit 24 enters Project Rollout process T4 and the User Training process T5 at the same time. During execution of theT4 process, deliverables are handed over to users in the production environment or warehouse. The T5 process will take care of developing training materials and conducting user training. When both processes are complete, the process flow enters the Project Closeout process T6, which must receive End of Project Notification via the entry point 25 in order to be executed. During T5 execution, signoffs are received, resources are released and all project documentation is archived. The project will be formally closed and the entire project process flow analyzed. Conclusions will be made about mistakes that occurred and how they could have been avoided. Upon closing the project, the project flow ends.

T6 will also be executed, even without the User Acceptance Test, if the Terminate Project notification comes from the Initiation or Planning Frames via entry point 3.

High level processes T1, T2 and T3 represent the set of User Acceptance Test processes. In the following chapters those processes will be broken down into detailed sets of processes.

User Acceptance Test Overview

The User Acceptance Test or UAT is a set of user developed and conducted tests to confirm that a completed product satisfies business requirements as outlined in the Business Requirements Document. Each individual test is conducted according to the developed script of test execution, which is the detailed step-by-step description of test activities. Each script represents a plan to test a specific business feature of the product in the environment, which is as close as possible to the real life operating conditions. The expected result is clearly described in each script and rated pass or fail. All scripts must be documented in the Script and Incident Repository.

The UAT scripts are developed and the tests conducted by the Acceptance Test Team, which consists of business users who are assigned by the business. Another team, the Acceptance Test Support Team, assigned by the delivery organization, will assist business users in that task. Their responsibility is also to resolve defects.

The UAT procedure described here may not be applicable to all projects, such as construction, process improvements, etc. For example, construction projects usually have their own test methods which are compliant to construction, electrical and other codes, as well as their own approval chain. It is not possible to provide specific UAT methods applicable to all project types. Therefore, processes described below provide general UAT steps and guidelines, but no specific examples of implementation. Some process names, such as “Identify test conditions for each BRD requirement” or “Identify test cases for each BRD requirement” give the general idea of this concept, but have no examples or details, since details are very specific for every business or practice area. The following will be achieved during the User Acceptance Test:

- A clear understanding of the testing scope
- The comprehensive test plan is developed, which provides schedule, reporting requirements, analysis and tracking activities
- Test scripts are developed to provide steps and conditions for testing of the product functionality, one test script for one element of functionality
- All business-related functions and requirements, as described in BRD, are tested
- The product performance is tested
- The product interfaces with other products and users are tested
- The product tests use actual work scenarios
- All product defects found during script execution, which affect product functionality, are fixed and documented
- The root cause analysis of every defect is established and documented, unless agreed by all sides otherwise
- Nonfunctional requirements, such as safety, security and others are met
- All possible efforts are made to reduce number of defects discovered after release into production

- Training plans are developed based on actual user operations
- Communication plan is developed and implemented for communication between Delivery Team, Client, Business users and Management during all test activities
- All change requests are identified and prioritized

Note: *The most complicated and comprehensive test scripts are developed for the software, the information technology and to a lesser degree to projects in electronics and electromechanical fields. Those scenarios incorporate various types of inputs, including intentionally erroneous, while testing expected outputs. They often simulate unexpected inputs and try to break the existing project logic. Therefore, many details of test scripts described below are applicable only to specific business areas.*

The User Acceptance Test includes the following high-level sequential processes:

1. Acceptance Test Planning (T1)
2. Test Script Execution (T2)
3. Defect Resolution (T3)

Defects

A defect is a quality gap between the specified quality characteristic and the actual quality, which results in a product or service not satisfying its intended use.

Defects are identified when script execution conditions are not met successfully. The most probable reasons for defects discovered during the Acceptance Test:

1. A business requirement was not appropriately defined or documented, resulting in initiating a Change Request
2. A business requirement was not correctly implemented due to misunderstanding or errors in design or implementation
3. Requirements cannot be fully implemented with the selected technology or materials
4. A misunderstanding or mistake in testing occurred
5. Incorrect setup of the test environment
6. Insufficient communications between the delivery team and business Lack of required skills in the delivery team
7. Lack of the required skills in the Acceptance Test or the Acceptance Test Support Teams
8. Project planning errors
9. Not all quality audits and reviews have been performed during the course of the project
10. Incorrect assumptions are made during design or test

A defect must be corrected before continuing script execution. A defect is a problem, which will consistently show up in a similar situation in every script execution. For example, a bug

in the software program code or design error in the mechanical/electrical assembly is a defect.

Sometimes an incident may be defined as a defect only after multiple occurrences. If, for example, the tester complains about a perceived problem, which turns out to be an incorrect use of a test tool, this is not a defect. However, if more than one tester complains about the same problem, then there may be a true defect in the test tool training. In order to reduce the number of future defects, every defect requires a Root Cause Analysis. The assigned delivery team members will perform the Root Cause Analysis when an incident is determined to be a defect. Root Cause Analysis is the activity to figure out what is really causing the defect to occur and remove it, so the situation does not occur again, rather than simply continuing to deal with symptoms. If the incident is not a defect, then it is an issue, which must be resolved in the framework of the issue resolution.

Every incident of a problem must be documented in the Script and Incident Repository, along with the Root Cause Analysis.

Quality

Testing process does not build quality into the product. Testing is a means of determining the quality of the tested product and removing the defects found. Since it is often impractical to test every single detail due to schedule and cost constrains, some defects may be discovered after the system is released into production, if quality is not built in.

By PMI's definition, quality is the characteristic of an entity that affects its ability to satisfy stated or implied business needs and its fitness for use. The Quality Management process embeds the quality into the project and thus the product from day one of project development. By the time the testing of the product is performed, it is way too late to improve the quality of the deliverable.

Defects may be found any time during the project lifecycle, and not only during UAT. The cost of defect repairs increases proportionately the later in the project lifecycle they are detected. To correct a problem at the Requirements Frame may not cost much. To correct the problem post-implementation may cost tenfold and more.

V- Model

The V-Model, as shown on Fig 16-2 is the most commonly used approach to testing. The V-Model represents the Project Life Cycle. It shows the various stages in development and testing as well as relationships between various stages. The verification or validation of test activities relates to their corresponding requirements or specifications. The related testing activities to ensure that the right functions are performed are called ***validation*** activities; while other activities to ensure the correct or reliable performance of these specified functions are called ***verification*** activities.

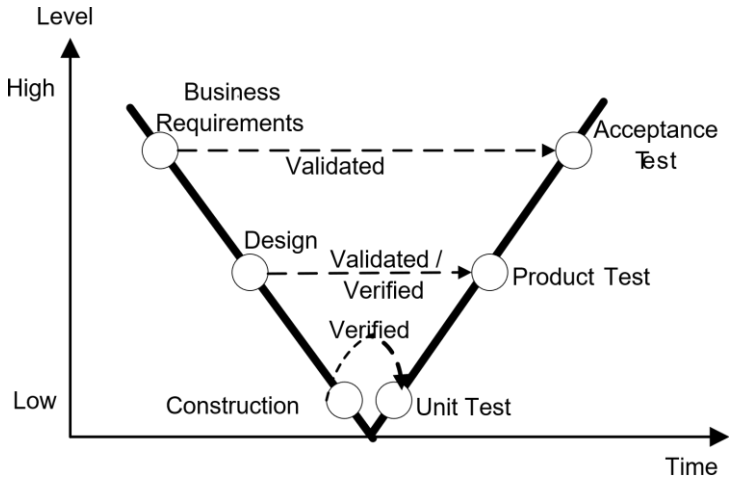


Fig 16-2 V-Model

If validation of activities is performed on a specific level of details, the verification of activities will be performed on exactly the same level of details and vice versa.

Every product under development consists of a number of elements, called units. Thus, in a software project a product may be called system and software modules called units. In a mechanical engineering project a product may be called assembly and units are subassemblies.

In the top level of the project view the business requirements established at the beginning of the project may be validated by the acceptance test, which is developed and performed by the business at the end of the project. On a lower or more detailed project level the product design produced at the initial stages of the project must be first validated and then verified by the product test. The product test is developed by an analyst or engineer and performed by several members of the delivery team at the last stages of the project before the product is scheduled for handover to the client for acceptance test.

On the lowest and most detailed level during the project construction, the unit verification is done by the unit test which is developed and performed immediately by the person who created the unit. A nit is a smallest testable part of the product. A Unit Test is performed by the individual developer who developed that unit/module/subassembly. It verifies that unit behaves exactly as required when tested independently of other units.

The next higher level of testing is the product test, when all tested units are assembled together. For a software development project the product test combines the system and integration tests. The product test will verify and then validate the existing quality elements:

- Correctness
- Reliability
- Usability
- Maintainability
- Testability
- Reusability

Both unit and product tests are very specific for each business area and product type. In order to understand the unit and product testing methods and processes, the reader must be a technical specialist in that specific area. Therefore the above tests are out of scope of this article.

Remember that the UAT is a business requirements validation process conducted by business users.

Script and Incident Repository

All scripts and every incident must be documented. There are many off-the-shelf tools which do this available. However if for any reason such a tool is not available, it is possible to build the repository the same way as the Project Control Book was built, which is an example of the document repository. It is not recommended to expand the PCB in order to include scripts and incidents, because the PCB is tool used mostly by the project manager and the Delivery Team, while the Scripts and Incidents Repository is mainly a user tool for User Acceptance Test. Script and Incidents Entry forms are provided in the Entry Forms section of this chapter.

Acceptance Test Planning (T1) Process Flow

Test activities must be planned like any other project activities, so that the test execution may be estimated, scheduled and tracked. Keep in mind that some of test activities are carried out by members of the client team, which must be planned, but they are not chargeable to client and are not a part of the budget. However, if those activities are not done in accordance with the plan, then the delivery team activities will cost extra, when supporting the client team test tasks. If this happens, the SCR must be submitted as described in the corresponding section.

In order to plan test activities, a good understanding of those activities, a detailed schedule/calendar, the script sequence and dependencies must be entered in the overall project plan. The planning process must have the following planning tasks allocated for implementation in the overall Project Plan:

- Planning of approach to testing
- Planning Test Methods
- Risk Assessment
- Detailed list of Acceptance Test tasks
- Task Dependencies
- Resource Assignment
- Acceptance Test Schedule/Calendar
- Communication Plan during the Acceptance Test
- Acceptance Test Exit/Completion Criteria planning
- Test cases and test script planning

The Acceptance Test Planning, shown on Fig 16-3, consists of the following processes:

1. Analyze BRD and Prepare Test Plan (T1-1)
2. Build, document and review test cases (T1-2)
3. Prepare Test Execution Checklist (T1-3)

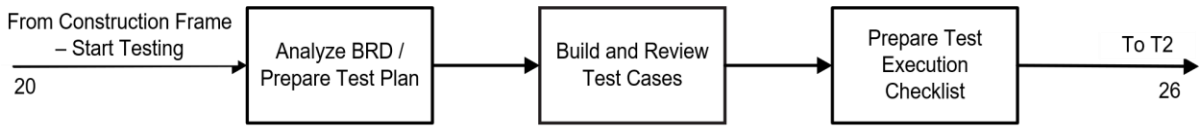


Fig 16-3 Acceptance Test Planning

Analyze BRD and Prepare Test (T1-1)

The process consists of the following tasks:

1. Analyze BRD (T1-1-1)
2. Identify test conditions for each BRD requirement (T1-1-2)
3. Identify test cases for each BRD requirement (T1-1-3)
4. Break down each test case for detailed elementary tests steps (T1-1-4)
5. Document test cases (T1-1-5)

Build, Document and Review Test Cases (T1-2)

The acceptance test cases are developed to validate that the system is performing in accordance with the Business Requirements Document. Test cases comprise the following elements:

Inputs - Requirements to be tested based on BRD should be identified and a combination of input scenarios to test all requirements should be documented. Attempts should be made to "break" the system by setting up illogical conditions and invalid data entry, such as entering smaller than minimum values, larger than maximum values or non-numeric values when numeric are expected. The customer name entry may have non-alpha characters. The expected results from the incorrect data entry are the rejection of data entry by the system, but the system should never crash.

Transactions - Each business requirement must be tested as specified in BRD by generating several transactions with both valid and invalid conditions. Ensure that transactions cover the complete requirement and then test what happens if the order of their execution is incorrect. If transaction deletes some data, upon completion of test case, restore the data, since test cases developed by other testers are based on a predetermined state of the data. Under no circumstances may the system crash, but rather must provide an acceptable and informative error message.

Outputs - Expected Results should identify the following elements:

- Results of each step in the test script. If expected results are numeric values, the range must be precisely identified or calculated.
- System status during script execution and upon its completion, such as ability or inability to run some transactions, display computer screens etc.
- Data modifications during a test script execution. As mentioned earlier, a script execution is complete; data must be restored to the state before script execution, unless planned otherwise.

Prepare Test Execution Checklist (T1-3)

Before running tests, the following checklist must be used to verify readiness:

- The test case execution schedule is available
- The test environment is ready
- Resources are committed
- Back-up & recovery procedures for software projects are in place
- The test data, if any, is loaded and accessible

Regression Testing is re-testing of previously working test scripts following defect resolution to ensure that defects have not been introduced as a result of the changes made. Typically, Regression Testing is required if modifications have been introduced to an existing product or to the environment. In this case tests should run to ensure that all existing processes that may possibly be affected still work after the change. If the original script was executed and closed, the copy of the script should run. In this case, the script number will consist of the original script number, then dash and then the copy number. For example if the original script number was 100, then the first copy will be 100-1, the second copy 100-2, etc.

Before the beginning of the Test Script execution after defect removal, it has to be determined whether a Regression Test is required. If it is, the Regression Test must be conducted after removing defects before the subsequent script execution.

Execute Test Script (T2) Process Flow

The Execute Test Script process T2 is decomposed to the following processes, as displayed in Table 16-1. Prior to executing process T2, all scripts must be entered into the Script and Incident Repository using the Script Entry Form in Table 16-2.

Table 16-1 Execute Test Script Process Decomposition

Process #	Process Name	Owner / Executor
T2-1	Open Next Script	Test Team Member
T2-2	Execute Script	Test Team Member
T2-3	Approve Incident	Test Team Member
T2-4	Open New Incident Report	Test Team Member
T2-5	Document Script Execution	Test Manager
T2-6	Receive Defect Resolution Notification	Test Manager
T2-7	Close Incident	Test Manager
T2-8	Reschedule Test	Test Manager

Table 16-2 Script Entry Form

Script Entry Form	
Project Name:	
Project Manager:	
Test Manager:	
Test Coordinator:	
Script Author:	
Script Owner:	
Script Number: _____ Execution #: _____ Script Status: □ Pending □ Defect Resolution □ Completed □ Escalated	
Script Title: _____	
Description: _____	
Script Open Date: _____ Script Close Date: _____ Closing Status: □ Ended with no incidents □ Defect fixed □ New SCR opened □ New Issue Opened	
Products/Projects Impacted: □ None 1. _____ 2. _____ 3. _____	
Script Comments: _____ _____	

Step # 1 Assigned To: _____Step Status: _____ Description: _____ Step Expected Results: _____ Step Actual Results: _____ Step Comments: _____
Step # 2 Assigned To: _____ Step Status: _____ Description: _____ _____ Step Expected Results: _____ Step Actual Results: _____ Step Comments: _____
Continue with steps 3, 4, 5 etc

The process starts when the Open Next Script process T2-1 triggered by one of the following:

1. The schedule of the test execution, indicating which specific script must be executed next.

2. The Reschedule Test process T2-8**,** which means that the defect that occurred during the previously executed script has been fixed. In this case the schedule also must indicate the correct timing to re-run the script and run regression tests.

3. After previous script is successfully completed and documented in the Document Script Information process T2-5 and the control point question (Acceptance Test Complete?) is answered NO, the process flow loops back to the Open Next Script process T2-1, starting execution of the next script in a queue.

4. An incident has been approved by the Test Manager in the Approve Incident process T2-3 and while a new incident is being opened in the Open New Incident process T2-4, the next script in the queue is opened for execution. The incident approval by the Test Manager is required in order to reduce the number of false defects, when it is clear that perceived defects are not defects, but rather problems which are not directly related to the quality of the deliverable. The Execute Test Script process flow is shown on Fig 16-4.

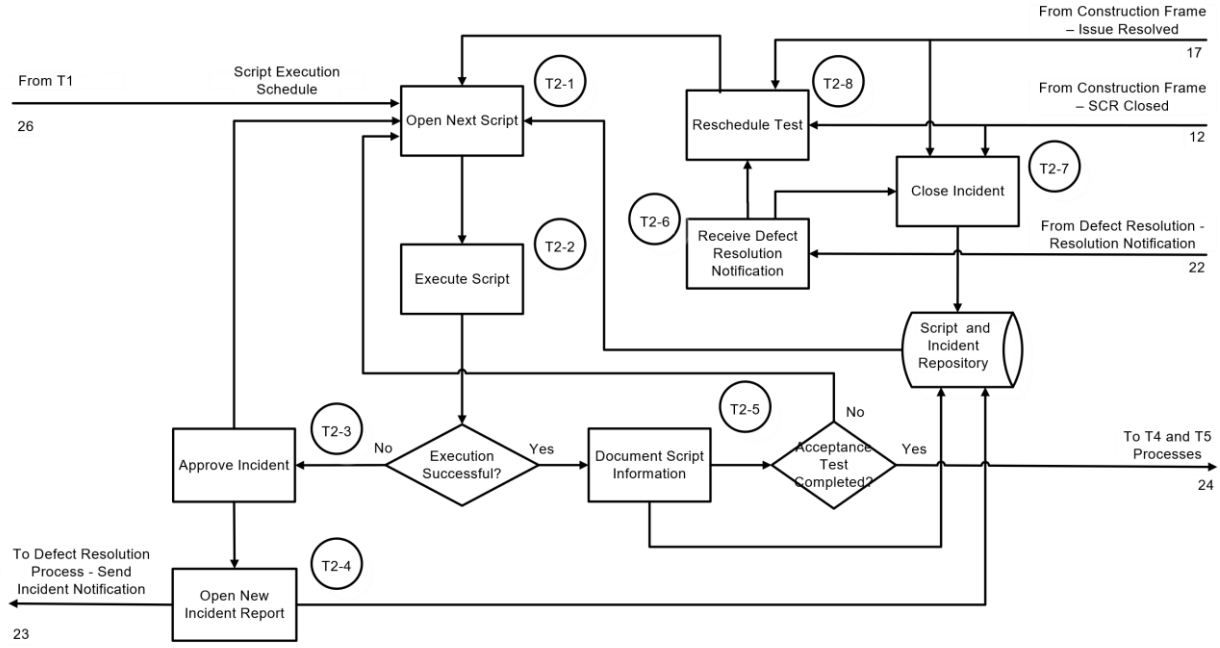


Fig 16-4 Execute Test Script

The process retrieves the next scheduled script information from the Script and Incident Repository and the script gets executed step by step in the Execute Script process T2-2 until the script execution is complete or an incident occurred during the execution. If the script execution is successful, the answer to the control point question (Execution Successful?" is YES and the process flow is directed to the Document Script Information process T2-5, where the Test Manager is notified, the script is closed and the script information is stored in the Script and Incident Repository. At the next control point question (Acceptance Test Complete?) the process flow loops back to the Open Next Script process T2-1 if the answer is NO, which means that the User Acceptance Test of the product is not yet complete and the next scheduled script must be executed. Otherwise, if the answer is YES, the process flows to the Project Rollout process T4 and User Training process T5 outside of the Script Execution process.

If the answer to the control point question (Execution Successful?) is NO, then the Test Manager approves opening the incident in the Approve Incident process T2-3 and new incident report is opened in the Open New Incident process T2-4. The Incident form is filled out as shown in the Incident Entry Form in Table 16-4 and stored in the Script and Incident Repository. The incident number consists of the script number, dash and then step number. At the same time an incident notification is sent to the Test Coordinator via exit point 23 to assign a test support team member to resolve the incident and perform Root Cause Analysis, which is done in the Defect Resolution process T3. From there the flow is looped back to the Open Next Script process T2-1 to start execution of the next script.

Table 16-4 Incident Entry Form

Incident Entry Form	
Project Name:	
Project Manager:	
Test Manager:	
Test Coordinator:	
Incident Owner:	
Incident Number:	
Script Number: _____ Step Number: _____	
Incident Open Date: _____ Incident Close Date: _____	
Resolution Priority: □ **Immediate** □ **1-2 weeks** □ **Delayed Pending Notification**	
Script Title: _____	
Status: □ **Owner Assigned** □ **Opened** □ **Closed**	
Incident Description: _____ _____	
List of Incident Attachments: _____ _____	
Reasons for Closing: □ **Defect Removed** □ **SCR Opened** □ **Issue Opened** Incident Closing Comment: _____ _____	
Defect Root Cause Analysis	
Defect Description: _____ _____ _____	
Analysis Description: _____ _____ _____ Performed by: _____ Date: _____	

The incident will only be approved by the Test Manager when there is a certainty that the incident did not occur due to a tester error, the test data is correct, there was no power failure, etc. Otherwise, the problem must be fixed first. Then the process flow will loop back to the Open Next Script process T2-1 to continue the same script execution when possible or start the script from the beginning. If the incident is approved, the process T2-4 opens a new incident report and then follows the flow as described above.

When the defect is resolved, the resolution notification comes from the Defect Resolution process T3 via entry point 22. The process flow enters the Receive Defect Resolution Notification process T2-6, where the Test Manager receives notice and confirms with the defect owner that defect was indeed resolved. The incident will be closed in the Close Incident process T2-7 and documented in the Script and Incident Repository. At the same time the script execution will be rescheduled in the Reschedule Test process T2-8 and the decision will be made whether regression tests are required.

During the Defect Resolution process T3 a new Scope Change Request or new issue may be opened. When the scope change is complete or issue is resolved, the corresponding notifications come via entry point 12 or 17 accordingly, to the Close Incident process T2-7 and also to the Reschedule Test process T2-8 in order to rerun the failed script again.

The Script Owner is responsible for the initial data entry into the Incident Entry Form. It is also the Script Owner's responsibility to keep the script data in the Script Entry Form, as shown in Table 16-2 above up to date; updating it daily if new information is available.

Script Entry Form

1. Project Name: name
2. Project Manager - name
3. Test Manager - name
4. Test Coordinator - name
5. Script Author – Name of the person who developed the script.
6. Script Owner - This is the name of the Acceptance Test team member who is assigned responsibility to execute the script.
7. Script Number – The first part of the script number is the sequential number assigned to the script. The second part is the number which shows the execution number. For the first script run it is 1, which increments for each subsequent script run. Under the normal circumstances the script runs only once. If the incident occurs during a script run, after taking care of the problem, the script is executed again. Example of the script number: 215-02.
8. Execution # - Shows whether the script runs the first time, the second etc.
9. Script Status – This is the status of the last step executed. Upon entering script information, script status is Pending, but later may be as follows:
   Pending: Script is being executed.
   Defect Resolution: Incident occurred, which is being resolved.
   Closed: Script execution is successfully completed.
10. Script Title – Script title should relate to the business function being validated.
11. Script Description – Detailed description of the script.
12. Script Open Date - The date when the script is entered.
13. Script Close Date – The date when the script is closed.
14. Closing Status -
    Execution ends without incidents
    Defect is fixed
    New Scope Change Request is opened
    New Issue is opened

15. Products/Projects Impacted – List of products or projects which may be impacted by this script execution.
16. Script Comments – Any helpful information to understand the script.
17. Step # - Each script may have a number of steps, which are sequentially numbered, beginning with number 1.
18. Step Owner – This is the name of the Acceptance Test team member who is assigned responsibility to execute the step. Usually, the Step Owner is the same as the Script Owner
19. Step Status – See script status.
20. Step Description – Description of the executed step, unless already included in the script description.
21. Step Expected Results – Description of expected results, which must include numbers, names of files with expected screens etc.
22. Step Actual Results - Actual results of the script execution. This information is entered at the time of script execution. If expected and actual results variance is acceptable, the script execution rating is PASSED. Otherwise, it is FAILED.
23. Step Comments – Any useful information to understand the step.

Defect Resolution (T3) Process Flow

The Defect Resolution high level process is decomposed to the following processes, as shown in Table 16-3.

Table 16-3 Defect Resolution Process Decomposition

Process #	Process Name	Owner / Executor
T3-1	Get Incident Notification	Test Coordinator
T3-2	Review Incident	Test Coordinator
T3-3	Clarify Info	Test Coordinator
T3-4	Assign Defect Owner	Test Coordinator
T3-5	Resolve Defect	Test Support Team Member
T3-6	Track Resolution	Test Coordinator
T3-7	Escalate	Test Coordinator
T3-8	Perform Root Cause Analysis	Test Support Team Member
T3-9	Notify Test Manager	Test Coordinator

This process starts when the incident notification comes from the Script Execution process T3 via the entry point 23. The Test Coordinator receives email notification in the Get Incident Notification process T3-1. The incident information is pulled from the Script and Incident Repository and reviewed with members of the Test Support Team in the Review Incident process T3-2. T3-2 process also makes a detailed analysis to establish whether the incident is a defect. If the incident information is insufficient for further actions, the answer to the control question (Info Complete?) is NO and the inquiry is made to the Test Manager to provide additional information in the Clarify Info process T3-3. When clarifications are received, the process flow loops back to the Review

Incident process T3-2 and incident is reviewed again. If the answer is YES, the next control question asked is (Is This A Defect?) If it is determined that the incident is not a defect, then this is an issue that must be resolved using issue resolution process described in the Planning Frame section. In this case the answer is NO and request is forwarded to the Planning Frame via exit point 16 to open a new issue. If the answer is YES, then the incident is a defect and the process flow enters Assign Defect Owner process T3-4, where the Test Coordinator assigns a member of the Test Support Team to fix the defect. While defect is being fixed in the Resolve Defect process T3-5, the progress will be tracked by the Test Coordinator in the Track Resolution process T3-6. If defect cannot be resolved without scope changes to the project, then the answer to the control question (Scope Change Required?) is YES and a request is issued to the Planning Frame to issue new SCR via exit point 15. Otherwise the next control question is (Resolved?) If the defect is not resolved yet, the answer is NO and the process flow enters the next control point (Deadline Missed?) If the deadline for defect resolution is not missed, the answer is NO and the defect resolution tracking continues by looping back to the Track Resolution process T3-6. However, if deadline is missed, the answer is YES and the problem is escalated to the project manager in the Escalate process T3-7. If the project manager is not successful in remediation of the problem, he or she must escalate it to the Senior Delivery Manager.

After escalation, the process flow loops back to the Review Incident process T3-2, where the Test Coordinator reviews the incident again and the defect resolution starts all over again, as described above. Going back to the Review Incident process, rather than continuing execution of the Resolve Defect process T3-5 gives the opportunity to assign a different defect owner in the process T3-4.

Note*: It is always a good idea to assign a different Test Support Team member for resolution of that specific incident, even if the current one did not make any mistakes. This is done to alleviate tensions in the business team due to delay and demonstrate to them that the delivery team makes visible efforts to resolve the problem.*

When the defect is fixed, the answer to the control question (Resolved?) is YES and the flow enters Notify Test Manager process T3-9, where the test schedule is updated to allow the failed script to run again. T3-9 process also sends resolution notification to the Script Execution process via exit point 22, so that the failed script may run again in accordance with new script execution schedule. At the same time, when the Test Manager is notified that the defect is fixed, the process flow enters Perform Root Cause Analysis process T3-8, where the root of the defect is determined. This process is required to avoid future problems due to the same reason and to build a knowledge database, which may assist with quick resolution of similar problems in the future. The Root Cause Analysis may take up to 72 hours of investigation. At the end of the T3-8 process, the Root Cause Analysis is documented in the Script and Incident repository. The Defect Resolution Process flow diagram is shown on Fig 16-5.

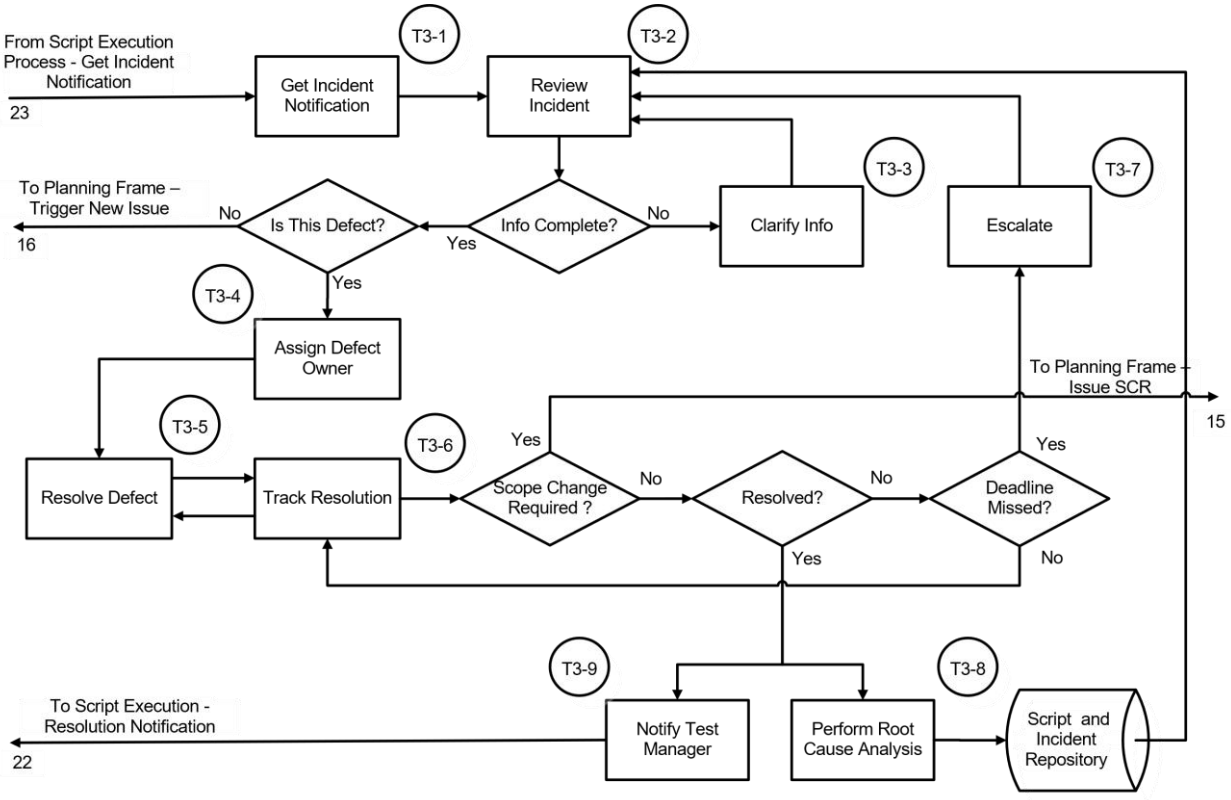


Fig 16-5 Defect Resolution Process Flow

Incident Entry Form

As shown in the Incident Entry Form Table 16-4, the Incident entries are:

Table 16-4 Incident Entry Form

Incident Entry Form	
Project Name:	
Project Manager:	
Test Manager:	
Test Coordinator:	
Incident Owner:	
Incident Number:	
Script Number: _____ Step Number: _____	
Incident Open Date: _____ Incident Close Date: _____	
Resolution Priority: □ **Immediate** □ **1-2 weeks** □ **Delayed Pending Notification**	
Script Title: _____	
Status: □ **Owner Assigned** □ **Opened** □ **Closed**	

Incident Description: _____ _____
List of Incident Attachments: _____ _____
Reasons for Closing: □ **Defect Removed** □ **SCR Opened** □ **Issue Opened** Incident Closing Comment: _____ _____
Defect Root Cause Analysis
Defect Description: _____ _____ _____
Analysis Description: _____ _____ _____ Performed by: _____ Date: _____

1. Project Name - name
2. Project Manager - name
3. Test Manager - name
4. Test Coordinator - name
5. Incident Owner - Name of the test support team member in charge of resolving incident

6. Incident Number – The number consists of the sequential number, always beginning with 1 for each new script.
7. Script Number - Script number is the sequential number assigned to script at the time of the script generation.
8. Step Number - Each script may have a number of steps, which are sequentially numbered, beginning with number 1.
9. Incident Open Date - Date when the incident was open.
10. Incident Close Date - Date when the incident was closed. The incident will be closed when one of the following happens:
    * *Defect was resolved*
    * *New SCR was open*
    * *New issue was open*
11. Resolution Priority – A resolution priority is assigned at the time of the incident opening. There are four priority levels:
    * *Immediate*
    * *1-2 weeks*
    * *Delayed pending notification*
12. Script Title – The script title relates to the business function being validated.
13. Status – The incident status may have one of three values:
    * *Opened*
    * *Defect Owner Assigned*
    * *Active* (Defect being resolved)
    * *Closed*

14. Incident Description – Detailed description of the problem.
15. List of Incident Attachments – List of attachments needed to understand the problem.
16. Reasons for Closing – Incident may be closed due to one of the following reasons:
    * *Defect Removed*
    * *New SCR Opened*
    * *New Issue Opened*
17. Incident Closing Comment – Comments required to understand the reason for closing. If the reason for closure is opening new SCR or Issue, it must be explained here.

Defect Root Cause Analysis:

1. Defect Description – If this is different from the incident description above, this is the place to enter it.
2. Analysis Description – This is the root cause analysis report.
3. Performed by - Name of the test support team member who performed the root cause analysis.
4. Date - Date of the root cause analysis.

Metrics

During the script development and execution, at a minimum, the following metrics will be recorded for the eventual storage in PCB:

- Number of test scripts and test steps planned and prepared.
- Number of tests scripts and test steps planned for execution up to-date.
- Number of tests scripts and test steps executed up to-date.
- Number of defects discovered and fixed for each test script.
- Total number of defects discovered and fixed.

In addition, the effort and duration of the test planning, preparation and execution stages, as well as defect fixing and the root cause analysis, must be recorded and tracked.

Project Rollout (T4) Process Flow

The objective of the process is to:

1. Ensure that user acceptance test has been successfully completed and the deliverable product meets the documented business requirements.
2. Product is moved/installed and fully functional in the users' or production environment.
3. User acceptance/approval is received and documented.
4. Client receives all required authorities to use the product on a daily basis or to transfer or sell the product to a third party.

Upon completion of the user acceptance test, the evidence must exist that all scripts run successfully, all incidents are closed and no outstanding defects, scope change requests

or issues exist. The Test Manager must ensure that all scripts are successfully executed and documented in the Script and Incident Repository. This provides evidence that the UAT is successfully completed.

Moving the product to the user's environment involves methods specific to the industry. Thus, in the information systems area, the product moves to the production library, having new sets of programs, libraries and databases initialized with the production data. The cutoff to the production environment takes place on the planned date and time. In addition, security arrangements must be made to ensure the end users' access to the product, at the same time relinquishing the delivery team's access, except for the maintenance personnel. Other business areas use other methods to ensure that the product will meet business requirements when operated in user's environment by the assigned end users of the product.

It's at this point where the product of the project moves to the steady state. For sustainable design, it is important that the project manager keep this moment in mind throughout the planning and execution Frames. By doing so, the PM keeps in mind long-term effects. For example, if the product of the project is a single-serve coffeemaker, questions about the recyclability of the coffee pods used in everyday use are thought through at an early enough time where they can have an impact. It is important for a project manager to think this way to align the sustainability and environmental assertions of the corporation with the project itself.

Just before transferring authorities to the client, the project manager will forward a Project Acceptance Certificate to the client for approval. Once approved, the certificate will serve as a basis for administrative and financial closure of the project. The Project Acceptance Certificate, along with the approval documentation, will be stored in the Project Control Book. From that moment on, the delivery organization will provide warranty repairs in accordance with the documented agreement between delivery organization and the client.

In case the client withholds the project approval for any reason, an attempt is made to resolve the issue between the project manager and the client. An action plan is developed to resolve the issue by a certain deadline. If approval is still not received by the deadline, the project manager will escalate the issue with the senior delivery manager, who, in turn, will escalate it to the senior business manager and from there to a higher level management, arbitration or the court of law. As long as there is no approval, the client and business users will have no authority to use or operate the product.

Project Acceptance Certificate

The Project Acceptance Certificate is displayed in Table 16-5.

Table 16-5 Project Acceptance Certificate

Delivery Organization			
Client Organization			
Project Name			
Client Name			
Project Manager			
Submitted Date		Rejected Date	
Reason for Rejection:			
Resubmitted Date:		Rejected Date:	
Reason for Rejection:			
Accepted Date:		Accepted by:	
Deliverables List:			
Comments:			

User Training (T5) Process Flow

When end users are well trained in the use of the product, it will have two obvious benefits:

- The delivery organization will save time and expenses by limiting false warranty service calls.
- The Client's organization will increase productivity and reduce the product downtime.

Therefore, it is important to plan and conduct the end-user training in the way which establish the fastest possible learning curve and gain the hands-on practical experience.

User Training process, as shown on Fig 16-6, can be decomposed into the following sub-processes:

1. Obtain List of Trainees (T5-1)
2. Assess User Needs. (T5-2)
3. Develop Training Materials (T5-3)
4. Deliver Training (T5-4)

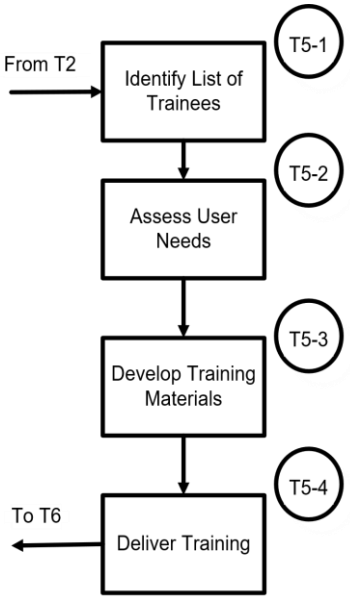


Fig 16-6 User Training Process Flow

Obtain List of Trainees (T5-1)

A list of trainees may be obtained from managers of those business units that are planned to operate the developed product. Those business units or their representatives must have participated in developing business requirements and desirably, performed user acceptance tests.

When list of trainees is received, there should be an indication how many users can participate in training at the same time without affecting daily operation of business units. This information must be taken into consideration when identifying training methods and scheduling training.

If the delivered product is a brand new product, then end users will have to learn everything from the business background to the practical use of the product until they attain a satisfactory proficiency in the product use. Sometimes the newly developed product comes in place of the old existing product, which, upon successful implementation will increase performance, accuracy, etc. The latter may require less training, even though that is not always true. If the end users develop specific habits of operating the old product, while the new method of operation is substantially different from the old one, the upgrade may be a difficult task to learn.

Assess User Needs (T5-2)

The assessment of user needs may require study of incidents that occurred during the UAT due to users' errors and evaluation of their skills. The assessment may require producing checklists to understand end user readiness to learn both the business part as

well as the product operation. Always keep in mind that users have different skills, different education and different attitudes.
While it is not always practically possible, it may be beneficial to provide at least two levels of training to avoid novice users getting confused or advanced users bored.

Sometimes different groups of users use different features of the product. In those cases a separate training is mandatory for each group of end users.

Develop Training Materials (T5-3)

Training material will depend on the method of training delivery and on the evaluation of user needs, as well as on the number of trainees. Development of training materials is best achieved by the combined teams of the delivery team and the business team members having both technical and business input into the training. Therefore, before beginning this task, a team for development of training materials must be formed that reports to both the project manager and a client.

The most effective method of training is when an individual hands-on instructor trains a group of 15 to 20 users. Unfortunately, this is the most expensive method to deliver training materials. The least effective and the least expensive method of training is providing users with the product manual in hope that they figure it out by themselves. Other training methods may include:

- Live demonstration of product features and methods of operating individual product features.
- Computer Based Training, also called CBT, is either a web-based training online or the offline use of the training software package distributed to each trainee. CBT may offer interactive self-paced lessons with frequent quizzes to ensure that user understands training materials.

Training must include, among other things:

- Differences between the old product, if it exists, and the new one.
- What is the business purpose of the new product and advantages of its use?
- All business operations performed with the product.
- Possible issues and the common errors to avoid when using the product.

After developing the training materials, it is useful to conduct a pilot training ofa few specially selected users to evaluate the effectiveness of the training and adjust training materials if necessary. If the list of trainees is large, a “train-the-trainer” program may be used, when an advanced group of trained users conduct training with the rest of them. Otherwise, an agreed group of the delivery team members or well-trained business team members are assigned to conduct the training.

Deliver the Training (T5-4)

In order to deliver the training, the following steps must be followed:

1. Schedule each training session and send notifications to trainees.
2. Assign trainees for each training session
3. Distribute training materials and conduct training sessions

With the exception of CBT training, training sessions must be scheduled in advance to avoid interruption of the existing business. All trainees and their managers must be notified and the acceptance must be confirmed. The entire training schedule should be sent to trainees, so they may request an alternative training date, if they cannot attend at the scheduled date. Their managers must release them from other duties during the training session and assign other personnel to perform their duties during that time. If a trainee misses part of the training, he or she must not receive training certification and must not be allowed to operate the product, unless the complete training is received at the alternative training session.

The training schedule will be considered confirmed after the trainee's acceptance is received, and training rooms and training equipment are booked for all training sessions.

Training sessions should preferably be delivered by trained members of the business team. Training sessions should not extend beyond 6 hours/day with 15 minute breaks every two hours. Having longer sessions will reduce the effectiveness of training. If training sessions are longer than three hours, there should be two trainers, alternating every one or two hours.

Project Closeout (T6) Process Flow

The closeout process starts when both Project Rollout (T4) and User Training (T5) processes are complete. The major purpose of project closeout includes the following key processes, as displayed on Fig 16-7.

- Assign maintenance / operation support team (T6-1).
- Release project resources (T6-2).
- Close the project financial documentation (T6-3).
- Perform Lessons Learned Analysis (T6-4).
- Archive all project documentation, contained in PCB (T6-5).

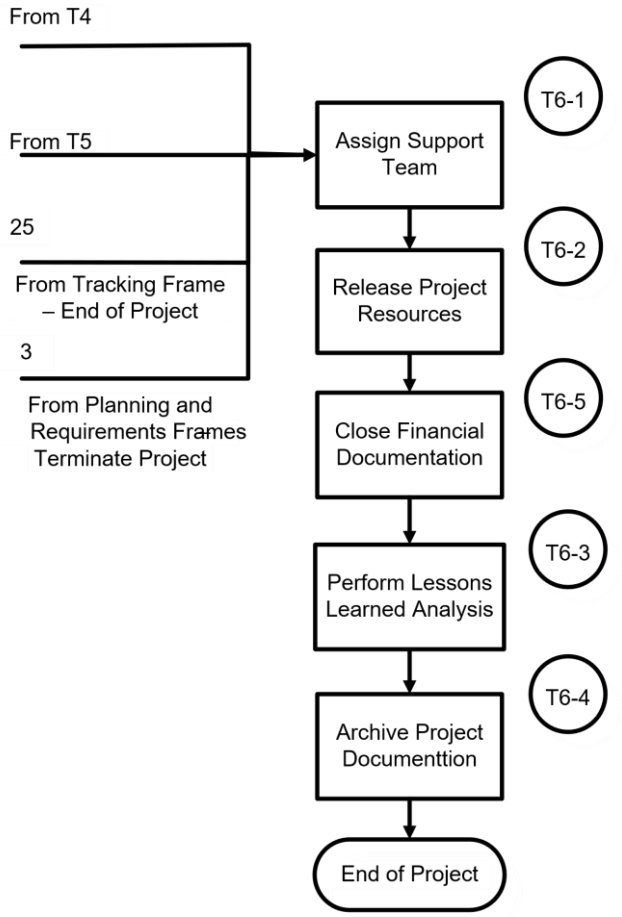


Fig 16-7 Project Closeout Process Flow

Assign Maintenance/Operation Support Team (T6-1)

When the project cut-off to the production or business environment is concluded in the Project Rollout process T4, the special assigned team must continue to exercise some control of the product for maintenance and operation support even after the project delivery team no longer exists. To enable such a team to perform its duty, the transfer of knowledge and documentation should be completed at this point.

Release Project Resources (T6-2)

Project resources, such as project staff, equipment and facilities must be released in T6-2 process in the following sequence:

1. All delivery team staff should complete the end-of-project feedback using the End-Of-Project Review Form, which may provide valuable insight of the inner project issues.
2. Collect all project documents and equipment from team members they used for project work.

3. Release the remaining project delivery team members and return them to the resource pool for use in other projects.
4. Submit performance evaluation of all delivery team members to resource owners and in some cases provides recommendations for recognition of performance.
5. Return equipment and facilities to owners.

End-Of-Project Feedback Form

The End-of-Project Review Form is shown in Table 16-6.

Table 16-6 End-of-Project Review Form

End-Of-Project Review Form		
Project Name: _____		Review Completed On: _____
Project Manager: _____		Team Member's Name: _____
Note: If any of your responses to statements were FALSE or N/A, please provide explanations		
#	**Statement**	**Response**
1	Your project duties were clearly defined and agreed with you. _____	☐ **TRUE** ☐ **FALSE** ☐ **N/A**
2	All your assignments and tasks were clear and unambiguous. _____	☐ **TRUE** ☐ **FALSE** ☐ **N/A**
3	You completed most of your assignments on time. _____	☐ **TRUE** ☐ **FALSE** ☐ **N/A**
4	Most predecessor tasks assigned to others were completed on time. _____	☐ **TRUE** ☐ **FALSE** ☐ **N/A**
5	Most of your task estimates were accurate. _____	☐ **TRUE** ☐ **FALSE** ☐ **N/A**
6	Most project team meetings were effective and efficient. _____	☐ **TRUE** ☐ **FALSE** ☐ **N/A**
7	You had never have personal issues, which affected your work. _____	☐ **TRUE** ☐ **FALSE** ☐ **N/A**
8	You had been almost never required to perform other unscheduled duties or excessive participation in meetings, conferences etc. _____	☐ **TRUE** ☐ **FALSE** ☐ **N/A**
9	You had almost never have to rework earlier completed work. _____	☐ **TRUE** ☐ **FALSE** ☐ **N/A**
10	You had never done any work without project manager's knowledge. _____	☐ **TRUE** ☐ **FALSE** ☐ **N/A**
11	Development or project management processes never hampered your work. _____	☐ **TRUE** ☐ **FALSE** ☐ **N/A**
12	If you had to start the project all over, you would do everything the same way. _____	☐ **TRUE** ☐ **FALSE** ☐ **N/A**
13	You have never been frustrated with the project. _____	☐ **TRUE** ☐ **FALSE** ☐ **N/A**

14	Your interaction with project manager was reasonable and business-like. _____	☐ **TRUE** ☐ **FALSE** ☐ **N/A**
15	You felt that your project work was appreciated and recognized. _____	☐ **TRUE** ☐ **FALSE** ☐ **N/A**
Please feel free to provide your comments: _____ __		

Close Financial Documentation (T6-3)

The project manager will issue advice to the accounting department or equivalent, that the project has been complete. Based on this, they should close and archive all project financial documentation, such as bills, payments, purchase orders as well as the project contract. At the same time a new contract may be established for the product or service maintenance. A formal letter to the business must be produced, announcing contract completion. In case of premature project termination, the reason for termination must be stated.

Lessons Learned Analysis (T6-4)

A Lessons Learned analysis session must be conducted for all projects, whether they were successful or not, in order to avoid making the same mistakes in other projects. Major participants in the Lessons Learned Analysis session are project manager, technical lead and client.

The collective wisdom of project managers who have preceded you is a valuable resource for your organization. Do not lose this opportunity to capture wisdom for practical use.

All events that occurred during the course of the project are analyzed and the determination is made as to whether the applied response was the correct one. This should be recorded in the PCB as it happens, not only at the end when memory is faded or distorted, key contributors have left, and the pain (or pleasure) is not present in the moment.

For incorrect responses or actions taken during the course of the project, identify what should have been done instead. Along with events, project management processes should be scrutinized and their effectiveness checked. In addition, the end-of-project forms completed by the delivery team members should be thoroughly reviewed. Every problem should be discussed and documented in the following format:

1. Present the problem.
2. Present corrective actions taken to resolve the problem.
3. Review results of the problem resolution and analyze undesirable effect occurred, if any.
4. Discuss what would be a correct course of action.
5. Provide recommendation about the overall better approach to the project.

6. Review the end-of-project feedback from the team and identify significant issues.

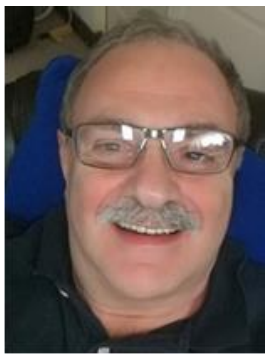
The Lessons Learned Analysis must be documented in the PCB. The analysis should be made available to delivery management for follow up to avoid similar mistakes in future projects – or to build on successes of the project under analysis. It is useful to place these findings in a searchable database to have corrective actions for many problems listed in order to use them in other projects.

Archive all project Documentation (T6-5)

All project documentation which is contained in PCB should be archived for future and for auditing purposes. This information does not have to be available for immediate retrieval, but the copy may be ordered from archive, while the original stays in archive.

Before archiving PCB, all documentation must be made “read only” to avoid modifications. The retrieved copy should retain the “read-only” attribute. When the copy is no longer needed, it should be discarded by the administrator.

About the Author



Dan Epstein

New York, USA

Dan Epstein combines over 25 years of experience in the project management field and the best practices area, working for several major Canadian and U.S. corporations, as well as 4 years teaching university students project management and several software engineering subjects. He received a master's degree in electrical engineering from the LITMO University in Leningrad (today St. Petersburg, Russia) in 1970, was certified as a Professional Engineer in 1983 by the Canadian Association of Professional Engineers – Ontario, and earned a master's certificate in project management from George Washington University in 2000 and the Project Management Professional (PMP®) certification from the Project Management Institute (PMI®) in 2001.

Throughout his career, Dan managed multiple complex interdependent projects and programs, traveling extensively worldwide. He possesses multi-industry business analysis, process reengineering, best practices, professional training development and technical background in a wide array of technologies. In 2004 Dan was a keynote speaker and educator at the PMI-sponsored International Project Management Symposium in Central Asia. He published several articles and gave published interviews on several occasions. In the summer of 2008 he published "Methodology for Project Managers Education" in a university journal. His book, *Project Workflow Management - The Business Process Approach*, written in cooperation with Rich Maltzman, was published in 2014 by J. Ross Publishing.

Dan first started development of the Project Management Workflow in 2003, and it was used in a project management training course. Later this early version of the methodology was used for teaching project management classes at universities in the 2003–2005 school years. Later on, working in the best practices area, the author entertained the idea of presenting project management as a single multithreaded business workflow. In 2007–2008 the idea was further refined when teaching the project management class at a university. Since 2009, Dan has continued working full time in Project Management. Dan can be contacted at dan@pm-workflow.com.