

Change-Based Testing

This section covers change-based testing. Parasoft's change-based testing helps you optimize your testing efforts by automatically identifying the test cases directly related to your most recent source code modifications. You can then rerun only the impacted test cases rather than retest the entire system after each modification.

In this section:

- [Correlating Code via Test Case Execution](#)
- [Correlating Tests](#)
- [Determining Where Retesting is Needed](#)

Correlating Code via Test Case Execution

Project Center helps you monitor which source code is exercised when a test executes, then correlate that source code with the appropriate task, requirement, or defect.

To do this, you need to:

1. Enable reporting to Parasoft Project Center - see [Connecting to Project Center](#).
2. Mark which task, requirement, or defect each test is related to - see [Indicating Code and Test Correlations](#).

Correlating Tests

If you are correlating code via test case execution (as described above) no additional configuration is necessary for this.

In addition to enabling change-based testing and requirements-based testing, correlating tests with tasks, requirements, and defects allows Project Center to track the number of project PRs/FRs/tasks that have test cases, track PR/FR/task test case status details (passed, failed).

Determining Where Retesting is Needed

From Project Center

You can determine where retesting is needed in the following ways:

- To see which manual test scenarios should be retested due to changes in correlated code, choose **Tests> Change-Based Testing> Test Scenarios**. This opens the [Change-Based Testing: Test Scenarios Report](#).
- To see which requirements/defects should be retested due to changes in correlated code, choose **Tests> Change-Based Testing> Requirements/Defects**. This opens the [Change-Based Testing: Requirements/Defects Report](#).

Change-Based Testing: Test Scenarios Report

The Change-Based Testing: Test Scenarios report indicates which manual test scenarios should be retested because there have been changes in the source code that was correlated to the previously-tested functionality. In other words, if a manual test scenario A was correlated to requirement B and the source code related to requirement B has changed, manual test scenario A will be marked as a recommended test.

The following Manual Test Scenarios should be retested:

Scenario/Manual Test	Last run status	Executed on	Performed by	Test duration	Relations	Priority ▲	Retest
Test recording alerts in Chrome	Pass	Apr 15, 2011	jakubiak	1m	REQ 1047 TASK 34696	Very high	
Validate Browser Contents Correct after Recording in Chrome	Pass	Apr 15, 2011	jakubiak	1m	REQ 1047 TASK 33779	Medium	
Test recording confirm dialogs in Chrome	Pass	Apr 15, 2011	jakubiak	1m	REQ 1047 TASK 34696	Medium	
Test recording prompt dialogs in Chrome	Pass	Apr 15, 2011	jakubiak	1m	REQ 1047 TASK 34696	Low	
Task 32257 - PR 87984: Test SOAP Client tool configuration wizard	Pass	Apr 15, 2011	truong	0d 0h 0m	FR 87984 TASK 32257	Low	
Task 34339 - Chrome - Test recording with onbeforeunload dialogs	Pass	Apr 15, 2011	jakubiak	0d 0h 0m	REQ 1047 TASK 34339	Very low	
Task 34487 - Handle case where Chrome process started when a leftover process still exists	Not Run				REQ 1047 TASK 34487	Very low	
Task 33781 - Support headless mode in Chrome	Not Run				REQ 1047 TASK 33781	Very low	
Task 36462 - PR 94925 - Javascript parsing error on object literal containing regex	Not Run				PR 94925 TASK 36462	Very low	

The **Priority** column sorts tests by priority. The more changes that are introduced to the specific functionality related to a test, the higher the priority is.

The **Relations** column shows project artifacts with which the scenario is linked.

The report date period specifies the period for which code changes are considered. You can adjust the date range by clicking the **Last 7 | 15 | 30 days** **Last 12 | 26 | 52 weeks** area—or by clicking the **Switch to range mode** icon then specifying the desired to/from date range.

Clicking the priority cell value will display code details. For example:

Scenario: Test recording alerts in Chrome
Version: 0
Scenario Recent Run: Apr 15, 2011

The following file revisions have been committed to repository since the scenario recent run.

File	Version	Revision Date	Relation
HttpUnitContext.java	1.24	Apr 19, 2011	REQ 1047
FFBrowserController.java	1.146	Apr 15, 2011	REQ 1047
AjaxTestingContext.java	1.59	Apr 19, 2011	REQ 1047
ProxyEventHandler.java	1.35	Apr 15, 2011	TASK 34696
ProxyEventHandler.java	1.35	Apr 15, 2011	REQ 1047
Connection.java	1.61	Apr 19, 2011	TASK 34696

The details report presents all the source code revisions that:

- Are associated with specific functionality and
- Were committed to source control after the recent manual test run.

Change-Based Testing: Requirements/Defects Report

This report indicates which requirements or defects should be retested because there have been changes in the source code that was correlated to them. In other words, if code correlated to requirement A has changed, requirement A will be marked as needing retesting.

The following Requirement(s)/Defect(s) should be retested:

Id	Summary	Priority ▲	#Revisions	Last modification
REQ 1047	Support Chrome	Very high	34	Apr 19, 2011
PR 95250	navigate test failing when using proxy that requires basic authentication	Very high	6	Apr 19, 2011
PR 94760	task 35606 - Make sure finish() is called each time an ErrorOutputBatchContext is created, to prevent memory leaks	High	3	Apr 15, 2011
PR 74683	task 35823 - Cardinal Health/ExactTarget/Verisign/EMC/Fidelity/Lockheed/CIBC - Environment Variables do not work for File Paths	Medium	2	Apr 15, 2011
PR 85984	task 35826 - Sabre - Allow Environment Variables in Delay fields for tests	Low	1	Apr 15, 2011
PR 93892	task 35627 - CCS - Unable to navigate through HTML report using japanese characters in IE	Low	1	Apr 20, 2011

Clicking the priority cell value will display details about the entity that changed. For example:

Entity Id: [PR 94760](#)
Summary: task 35606 - Make sure finish() is called each time an ErrorOutputBatchContext is created, to prevent memory leaks
Total Revisions: 4

The following file revisions have been committed to repository during last 7 days

File ▲	Revision	Last modification
BatchContextUtil.java	32.2	Apr 15, 2011
TestCleanupTest.java	32.2	Apr 15, 2011
ToolTest.java	32.308	Apr 15, 2011
ToolTest.java	32.309	Apr 15, 2011

The details report presents all the source code revisions that:

- Are associated with specific requirement/defect and
- Were committed to source control in the specified timeframe.