



**CODESCROLL**  
**CONTROLLER TESTER**

**SURESOFT**

---

# **CODESCROLL** **CONTROLLER TESTER**

**RELEASE NOTE**

Software for safe world

[www.suresofttech.com](http://www.suresofttech.com)



# Release Note

2023 — Last update: Jul 18, 2023

Suresofttech

# Table of Contents

- 1. Added Features ..... 4
  - 1.1. Team Testing ..... 5
  - 1.2. Class Reconfiguration ..... 8
  - 1.3. New Toolchain and Support Environment..... 10
  - 1.4. The Others ..... 11
- 2. Improvements ..... 13
- 3. Bug Fixed and Feature Deleted ..... 16
  - 3.1. List of Fixed Bugs ..... 17
  - 3.2. List of Deleted Features ..... 18

# 1. Added Features

---

- [Team Testing](#)
- [Class Reconfiguration](#)
- [New Toolchain and Support Environment](#)
- [The Others](#)

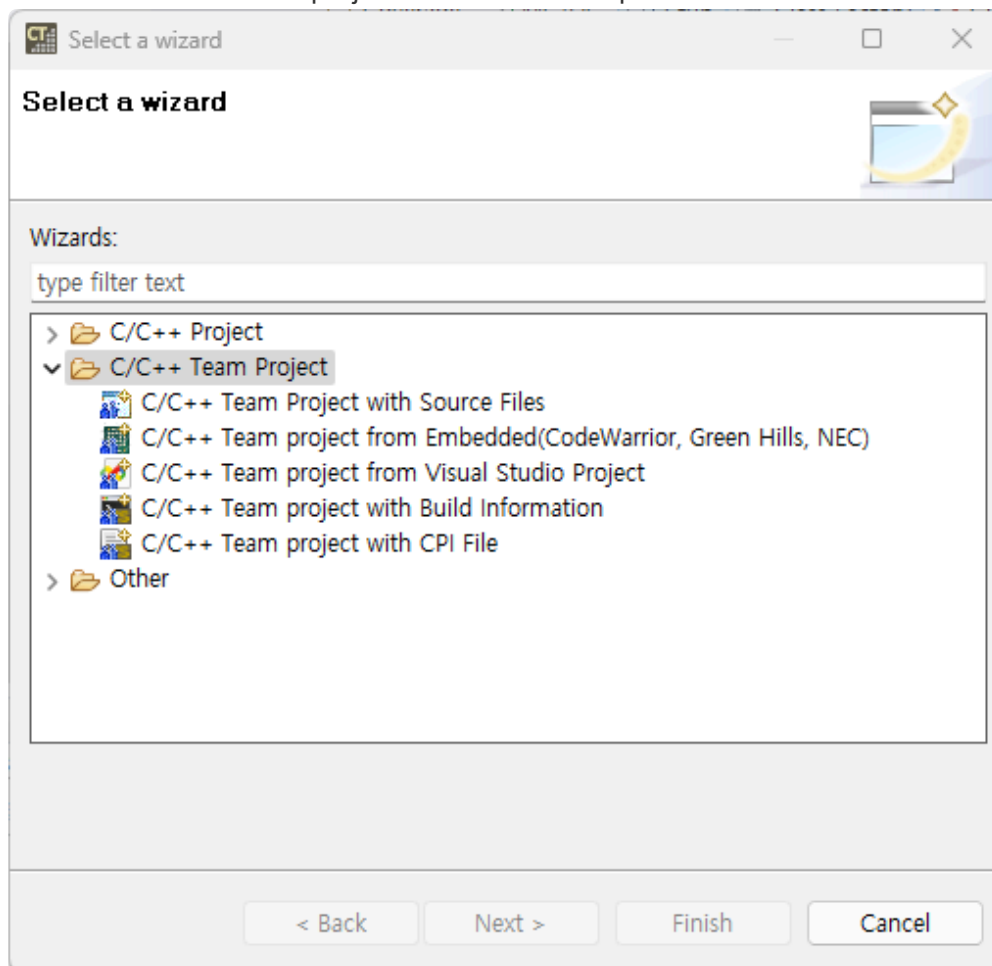
# 1.1. Team Testing

As software trends change, the source code grows and the update cycle shortens, reducing the time available for testing. This creates a requirement to perform many tests in a short period. For efficient testing in these situations, CT 2023 provides team testing feature.

- Update the target project onto Team Testing Server to synchronize configuration information while testing the project.
- Avoid duplicate creation by sharing stubs and class codes with all users.
- Provides the feature to resolve conflicts when multiple users modify a single resource and conflicts occur.
- Merge test results so you can check them live on your Team Testing Dashboard.

## Team project

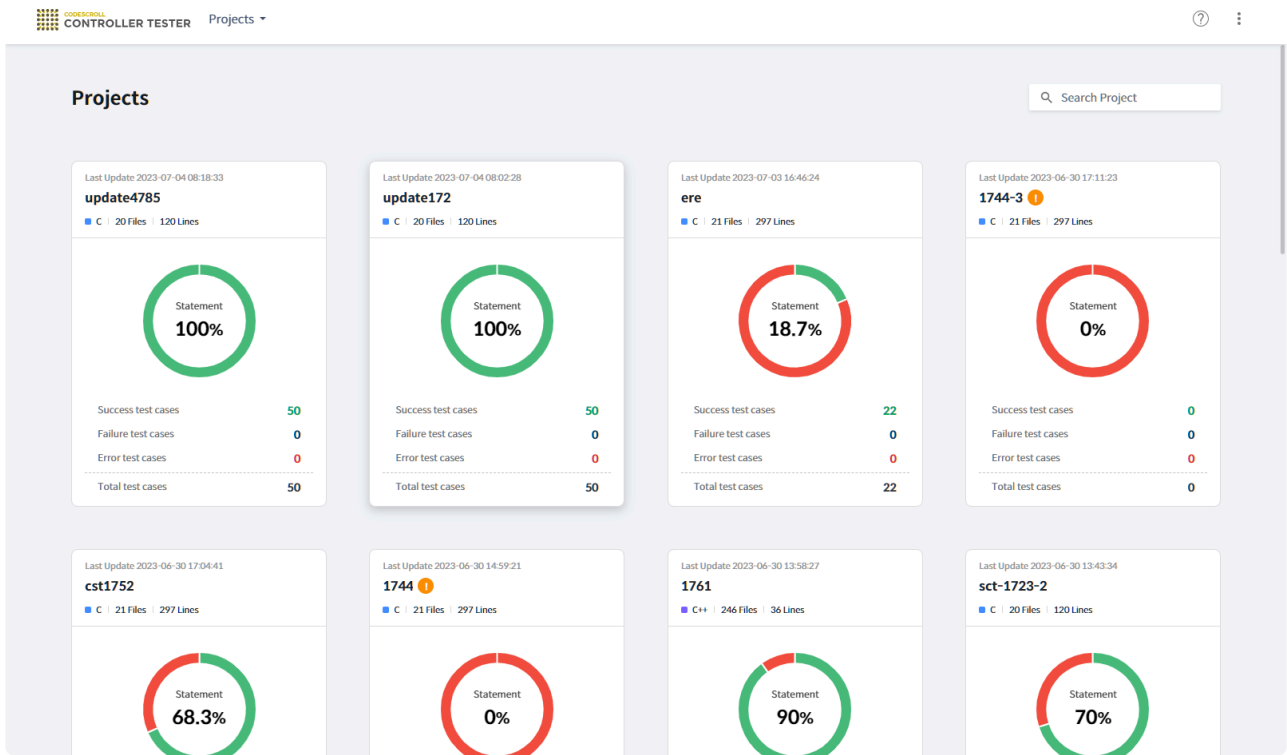
You can create a team project to test with multiple users.



- Added commit and update features to upload or download tests and resources to the server.
- Added a feature to resolve resource conflicts.
- Added team test import feature to import tests committed by other users.

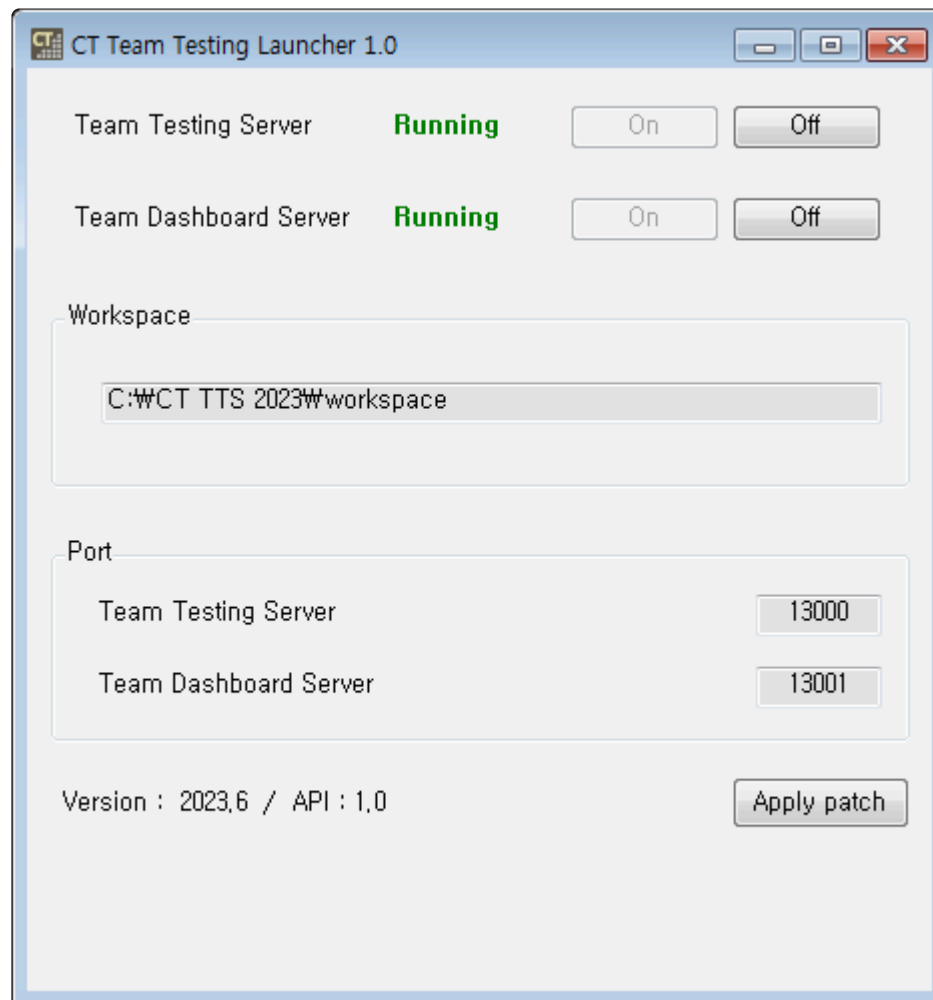
# Add Team Testing Server and Team Testing Dashboard

- Team Testing Server has been added to manage project information, tests, test resources and test results.
- Team Testing Dashboard has been added to show integrated test results.



# Add Launcher

Launcher has been added to manage Team Testing Server and Team Testing Dashboard.

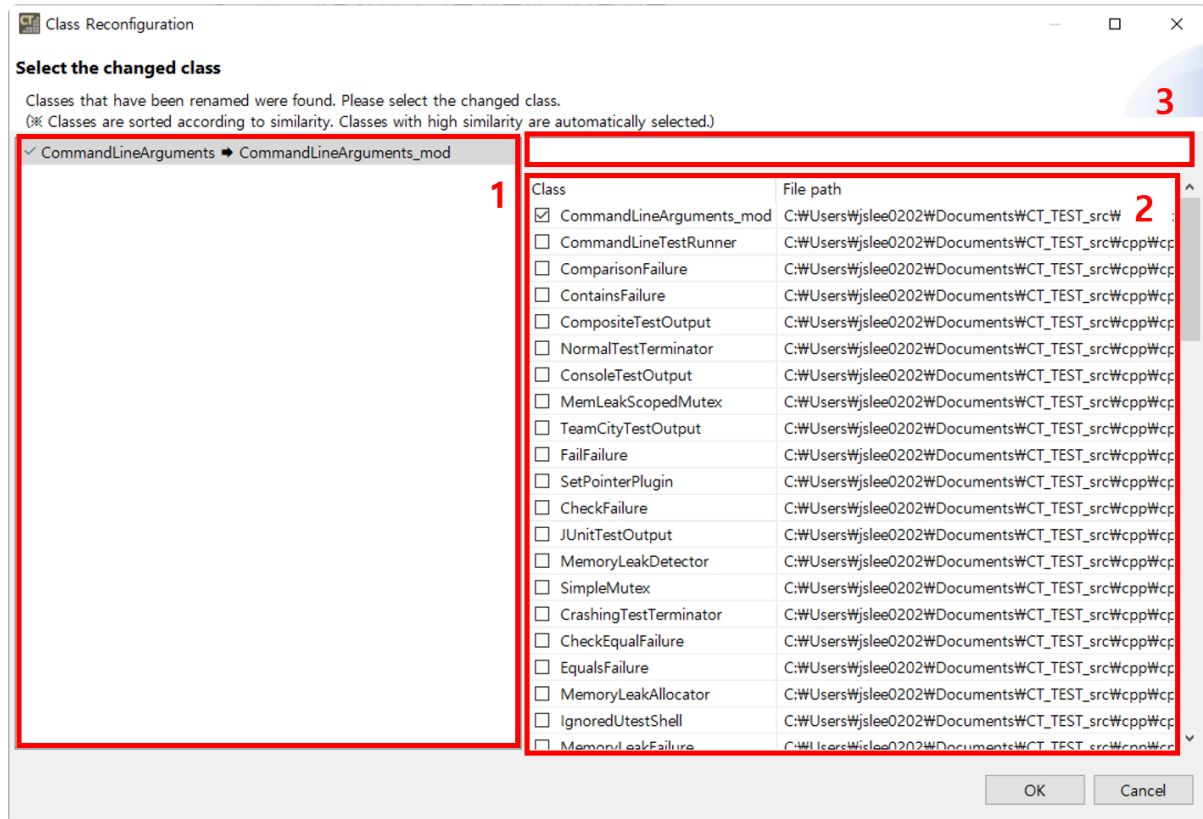


## 1.2. Class Reconfiguration

CT provides the feature to reconfigure existing tests, stubs, and fault injections when the source code modifies. To respond to C++ projects, CT 2023 added a class reconfiguration feature that allows you to reconfigure class codes and tests if a class or member function is renamed.

### Modify the name of class

- If the class name has changed, you can reconfigure the class code to reflect the change.



- In section 1, a list of classes that need to be reconfigured appears.
- Select the renamed class in section 2.
- Search by class name in section 3.

### Modify the name of member function

- If a member function is renamed, you can reconfigure the class code or tests to reflect the change.





- In section 1, a list of class codes that need to be reconfigured appears.
- Check the reconfigured class code in section 2. Click [OK] to save the right class code.

## 1.3. New Toolchain and Support Environment

---

### New support environment

Windows 11 is added to tool support environment.

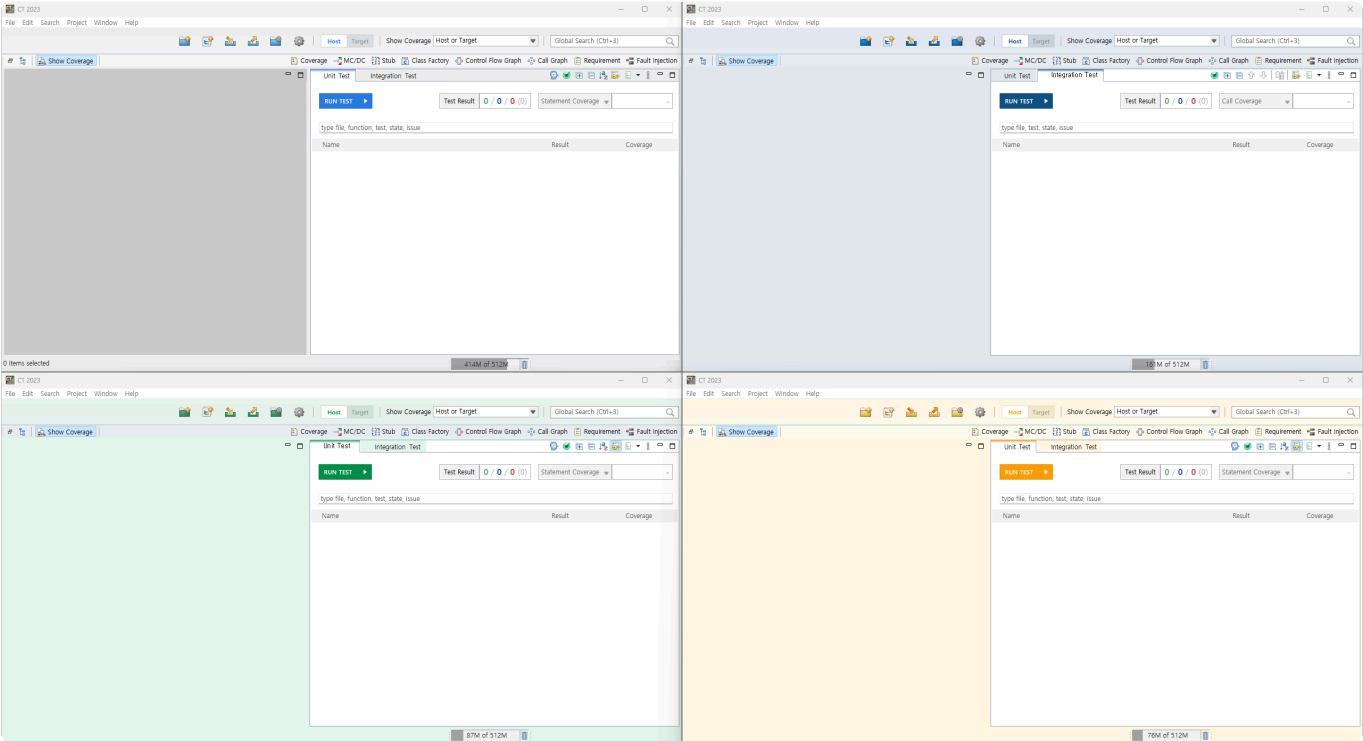
### New toolchain

Visual Studio 2022 are added to the list of supported toolchains. Visual Studio 2022 toolchains can be added using [Add Toolchain] feature on the toolchain configuration page.

# 1.4. The Others

## Theme

You can select 4 color themes in [Preferences] > [Theme].

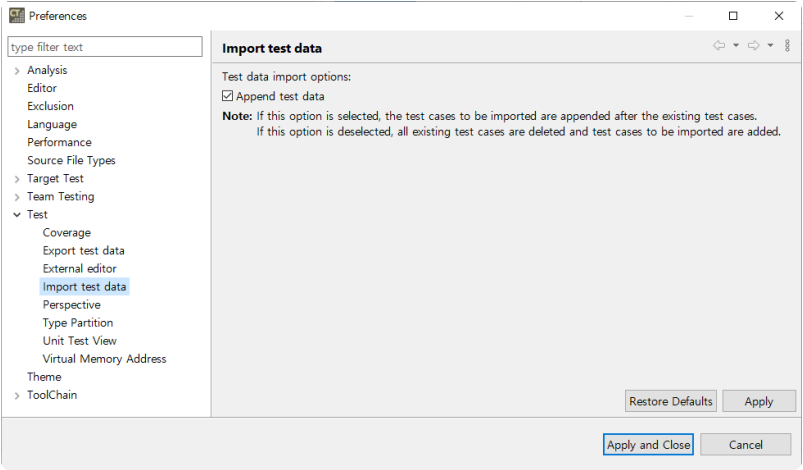


## Isolated stub

Added the feature to automatically generate stubs for calling functions when creating tests.

## [Import Test] preference

You can choose to append or overwrite the imported tests.



# Fault Localization

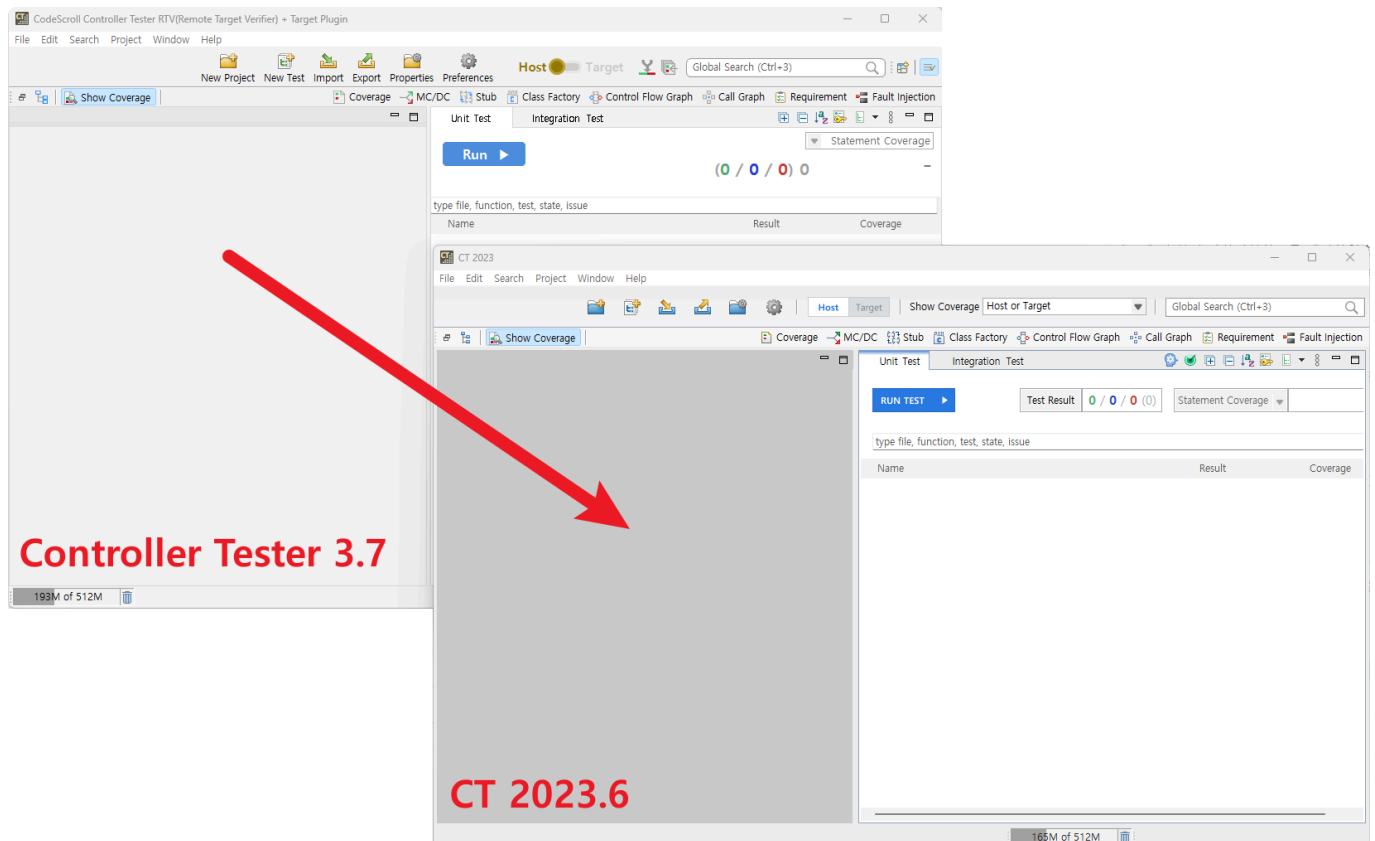
Apply Spectrum-Based Fault Localization (SBFL) technology to find where the fault is occurring.

## 2. Improvements

### UI

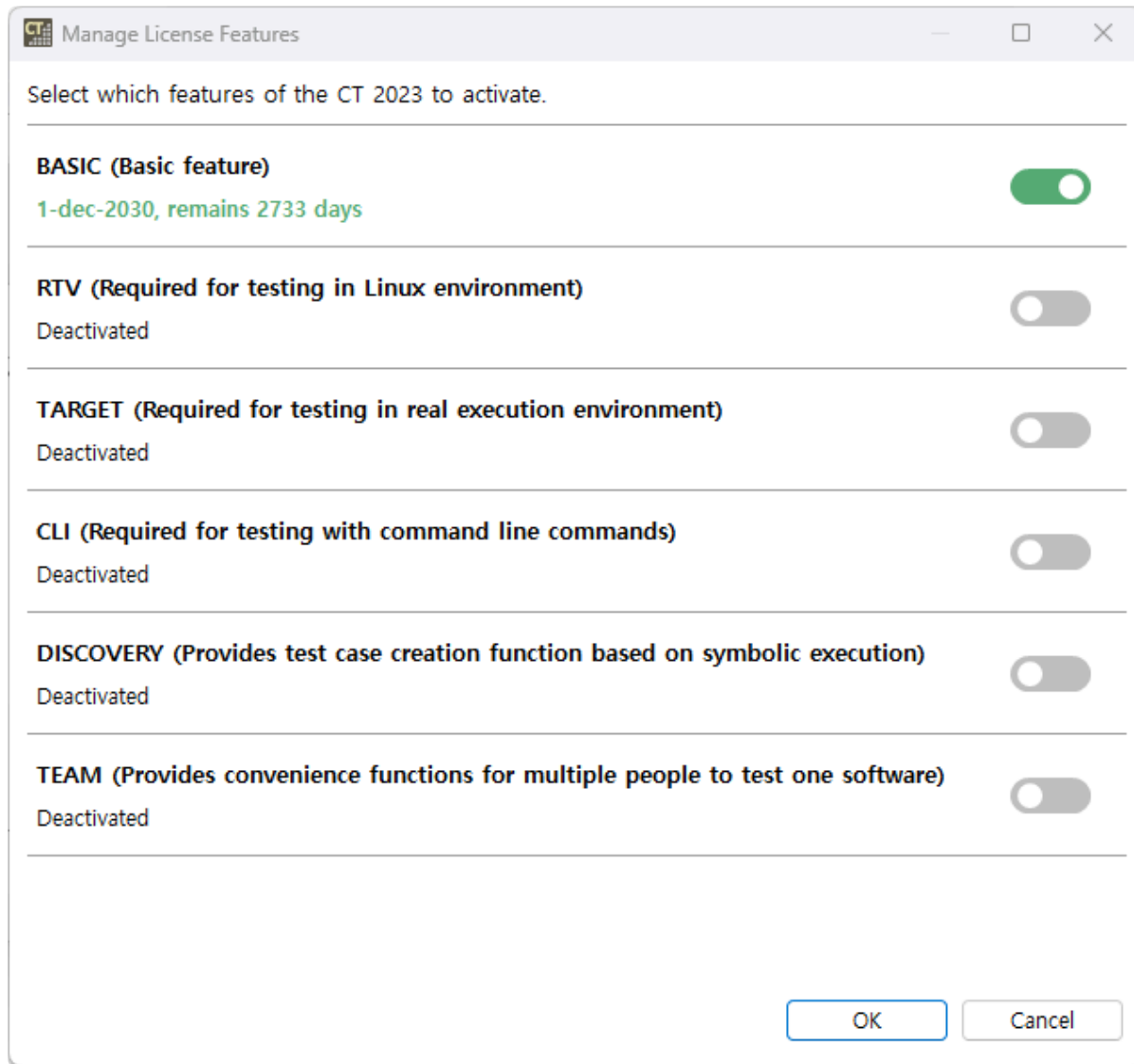
The overall UI has been improved to bring consistency to the tools UI.

- Change icons to be consistent
- Changed top dashboard UI
- Change Test view UI



### Package management

4 types of tool packages (Host, RTV, Target, RTV+Target) and 2 types of separate installers (CLI, DISCOVERY) are improved to integrate into one package and to activate/deactivate through license.



## Generate test report

When generating test reports, you can select coverages to export.

**Export**

**Export Test Result**

Select Project and Option

Project:

☐ Test

☐ vs

☒ zlib

Select All

Deselect All

1 of 3 selected.

**Coverage**

☒ Unit Test

☒ Integration Test

☒ Statement

☒ Function

☒ Branch

☒ Function Call

☒ MC/DC

**Option**

☒ File coverage report (format: XLS)

☐ Report for each test (format: XLS)

☐ Test stub report (format: XLS)

☐ Report with external coverage

☐ Uncovered function report (format: XLS)

< Back   Next >   Finish   Cancel

## Analyzer

- Support analyzing for lambda statements.
- Support most of C++17 and part of C++20 features via including latest parser

## Import/export in CLI

- You can use the CLI to import and export projects including toolchains and source code.
- When importing a project, you can set the path.

## The others

- Improved to warn when exporting projects larger than 5GB.

## 3. Bug Fixed and Feature Deleted

---

- [List of Fixed Bugs](#)
- [List of Deleted Features](#)



## 3.1. List of Fixed Bugs

---

- Fixed message in case of invalid when selecting configuration for toolchain
- Fixed an issue where the compile flag path was incorrectly replaced when importing a project
- Fixed an issue when editing the editor of a class type in the test structure tree.
- Fixed to decompress even when the length of the path exceeds 260 characters, when creating an RTV project,
- Fixed an issue where the list of workspaces was not initialized
- Fixed to handle exceptions when XML exception characters are included in the compile flags of RTV projects
- Fixed to perform large-capacity processing when performing tests in the RTV project
- Fixed to generate warning comments and code returning null if concrete class code for abstract class cannot be generated
- Fixed an issue where the array operator's test code was generated incorrectly
- Fixed the issue that tree-type function call information is displayed incorrectly when there is another function call in the code before/after the function call.
- Fixed to configure the number of symbols in the test
- Fixed to detect the change of the fault-injected function, when the source code is changed
- Fixed an issue where the values of previously selected symbols were applied in batches when [Apply the test case in a lump...]
- Modified test driver code to comply with C90 standard
- Fixed to export including subdirectories of system headers, when exporting toolchain with system headers
- Fixed an issue where default constructor stubs were not generated for some classes
- Fixed to import after warning when there is an invalid path in [Import Project] wizard.
- Fixed an issue that initialize the identifier entered by the user when reconfiguring the test
- Fixed an issue where wide string type input macros are created incorrectly
- Fixed an issue where function definitions in class code are incompletely created
- Fixed an issue where significant figures were not properly applied when the output value of the test case was less than the expected value
- Fixed an issue caused by shorten function name when build stub was set to [Disable]
- Fixed an issue that the position of the cursor changes when entering text in [Fault Injection] View
- Modified the display order of structure member numbers in the test structure tree to be sorted
- Fixed an issue that stub code input values are missing when importing a project

## 3.2. List of Deleted Features

---

- [Generate test cases automatically] > [Search-Based Test case Generation (CS Labs)] feature in the context menu of the test has been deleted.