



# Release Note

3.6 — Last update: Nov 16, 2021

Suresofttech

# Table of Contents

<b>1. Added Features</b> .....	<b>1</b>
1.1. Fault Injection Reconfiguration .....	2
1.2. Interaction with V-SPICE .....	3
1.3. Add Projects supporting DISCOVERY .....	5
1.4. The Others .....	6
<b>2. Improvements</b> .....	<b>8</b>
<b>3. Bug Fixed and Feature Renamed</b> .....	<b>10</b>
3.1. List of Fixed Bugs .....	11
3.2. List of Renamed Features .....	12

# 1. Added Features

---

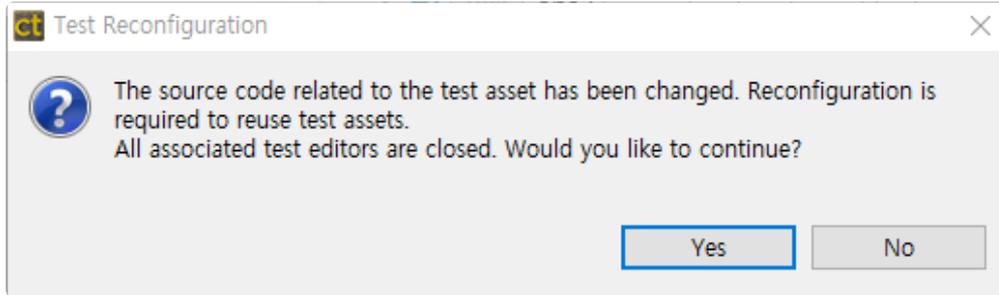
- [Fault Injection Reconfiguration](#)
- [Interaction with V-SPICE](#)
- [Add Projects supporting DISCOVERY](#)
- [The Others](#)

# 1.1. Fault Injection Reconfiguration

If fault-injected functions are changed, you can reuse the fault injection codes by using [Reconfiguring Fault Injection] feature.

## Run [Reconfiguring Fault Injection] on reanalysis

If you reanalyze the source codes after modifying, a dialog box asking if you want to run [Reconfiguring Fault Injection] appears.



## Run [Reconfiguring Fault Injection] in the Fault Injection View

If there are functions that need to be reconfigured in the Fault Injection View, you can run [Reconfiguring Fault Injection] through the pull-down menu(▽).

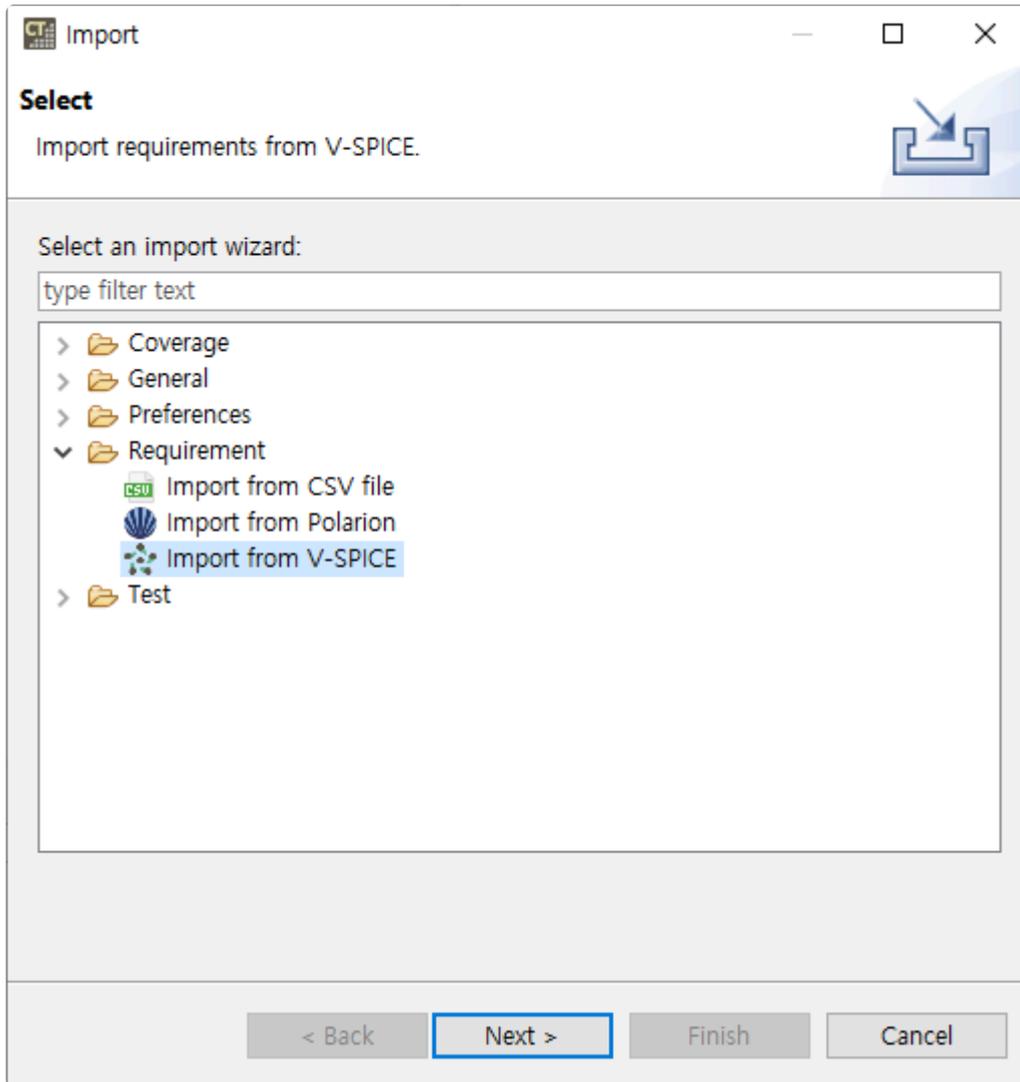


# 1.2. Interaction with V-SPICE

V-SPICE has been added to the tools that can interact with Controller Tester. {vspice}, A-SPICE certification support tool, support traceability and management for requirement and test cases. For details about this tool, please contact the technical support(help@suresofttech.com).

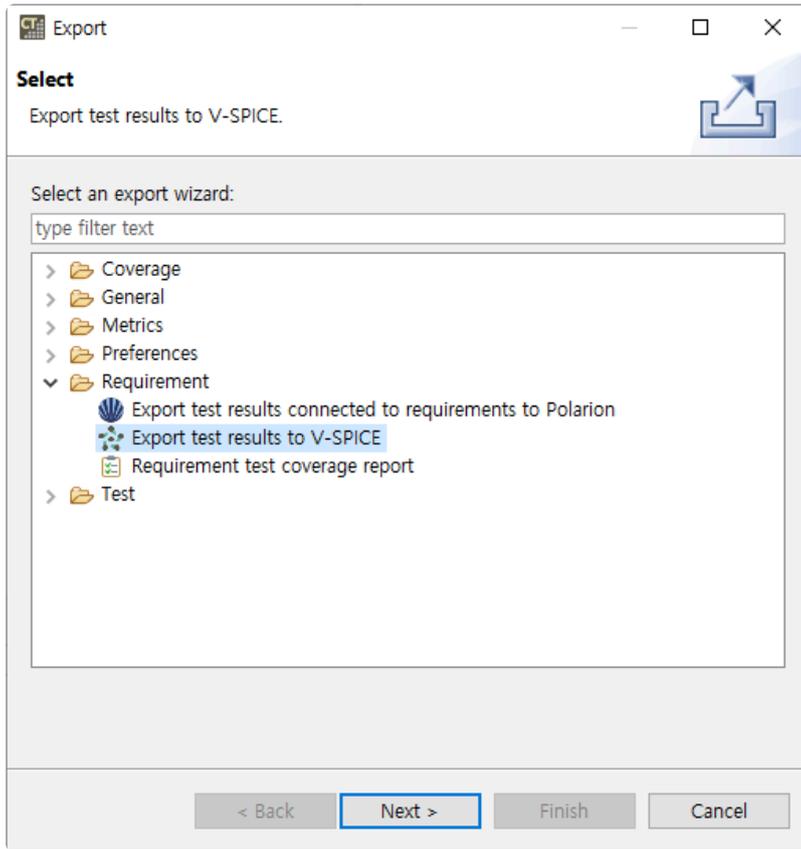
## Import requirements from V-SPICE

You can import requirements from V-SPICE



## Export test results to V-SPICE

You can export requirements and information about tests to V-SPICE



## 1.3. Add Projects supporting DISCOVERY

---

Projects that support DISCOVERY has been added.

- C++ projects
- Projects that use 64bit toolchain

# 1.4. The Others

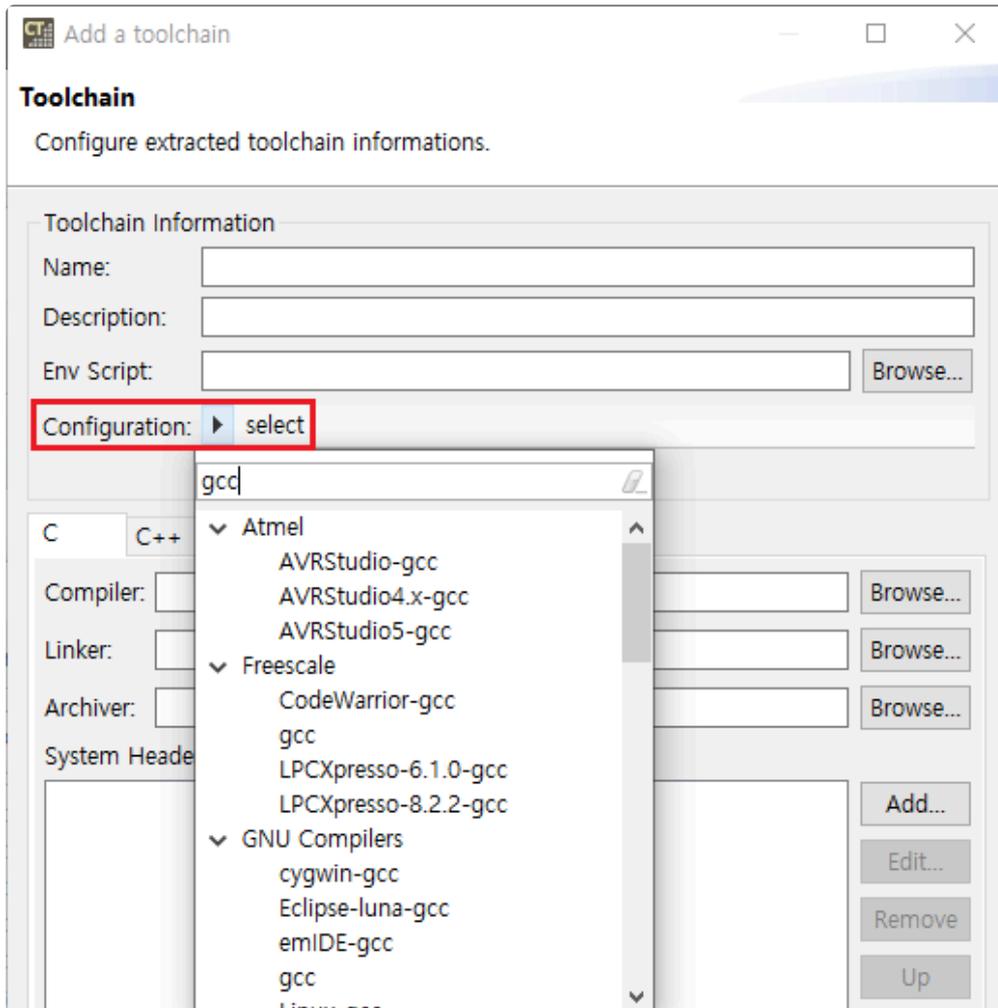
## Undo test data

When modifying test data in [Test Editor] > [Test Case Tab], you can use undoing feature.

- Undo: `Ctrl + Z`

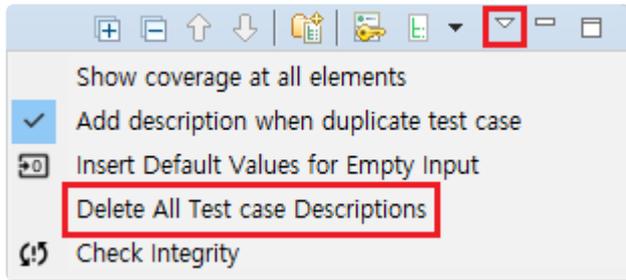
## Search configuration of toolchains

When add/edit a toolchain, you can find configuration by searching by name.



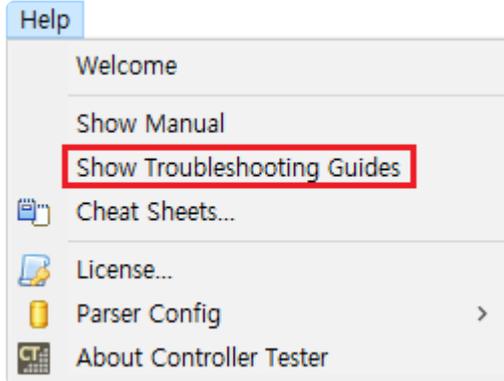
## Remove descriptions for test cases

You can remove descriptions for test cases all together in pull-down menu(▽) of the Unit/Integration Test View.



## Show troubleshooting guides.

[Show Troubleshooting Guides] feature has been added in [Help] menu.



## 2. Improvements

---

### The Fault Injection View

- You can inject faults line by line.
- You can check the lines you want through the filters.
  - You can check the non-empty lines through [Show Only Non-empty Fault Injections].
  - You can check the enable lines through [Show Only Enable Fault Injections].
- You can save the information of fault injections as Excel files through [Export Fault Injections].

### Import coverages

- When [Import Coverage] is performed, it has been improved so that it can be imported according to the conditions below.
  - by version of coverage shared file
  - by the conditional operator(?:) option of coverage shared file
  - coverage type of coverage shared file

### [Test reconfiguration] feature

- In the [Test reconfiguration] dialog, you can reconfigure information of the pointer type symbol with index data by dragging and dropping.
- In the [Test reconfiguration] dialog, if the input/output options of variables are different, they are displayed in red.
- In the [Test reconfiguration] dialog, you can reconfigure the input/output options of variables by dragging and dropping.

### Requirement test coverage report

- You can check the number of tests and test cases.
- You can check requirement consistency coverage and requirement traceability coverage.

### C++ tests

- The codes generated by default for std::ostream objects have been changed to console output types.
- If you have functions that return objects, it will be reflected in the automatically generated stub code.
- The test codes that automatically generate for template functions have been improved.

### The others

- When adding test target functions in [Test Editor] > [Test Information] > [Test Structure], you can select and add multiple.
- If you change the check box status of Global Variable in [Test Editor] > [Test Information] > [Test Structure] and save it, the selected global variable is raised to the top of the global variable list.

- When running DISCOVERY, if there are any changes in the Test Editor, a [Save and Launch] dialog will appear.

## 3. Bug Fixed and Feature Renamed

---

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

## 3.1. List of Fixed Bugs

---

- Fixed an issue that test run shortcut(F11, Alt + F11) do not work.
- Fixed an issue where symbols using input/output macros in unmanaged codes do not appear in the Test Case Tab if more than 4000 symbols.
- Fixed an issue where global variables were written incorrectly during code conversion.
- Fixed an issue that factory code for template class is created incorrectly in the Class Factory View.
- Fixed an issue where code line numbers were displayed up to 99 in views using the editors.
- Fixed class codes to avoid compile errors in the Class Factory View.
- Fixed an issue where test reports could not be generated using CLI.
- Fixed an issue where stubs were not generated when function names end in \_\_S or \_\_C.
- Fixed an issue where existing global variables could not be found when [Global variable reconfiguration] dialog.
- Fixed an issue that the conversion toolchain option(ini file) is not applied for toolchains created after Controller Tester 3.5.
- Fixed to not generate factory code when using inner class as template parameter.

## 3.2. List of Renamed Features

---

- [Help] in the [Help] menu has been renamed into [Show Manual].

