

Astemes LUnit Global Fixture

Astemes - Anton Sundqvist

Anton Sundqvist

Copyright © 2024 - 2025 Astemes

Table of contents

1. LUnit Global Fixture	3
1.1 What?	3
2. Why?	3
3. How?	3
4. Prerequisites	4

1. LUnit Global Fixture

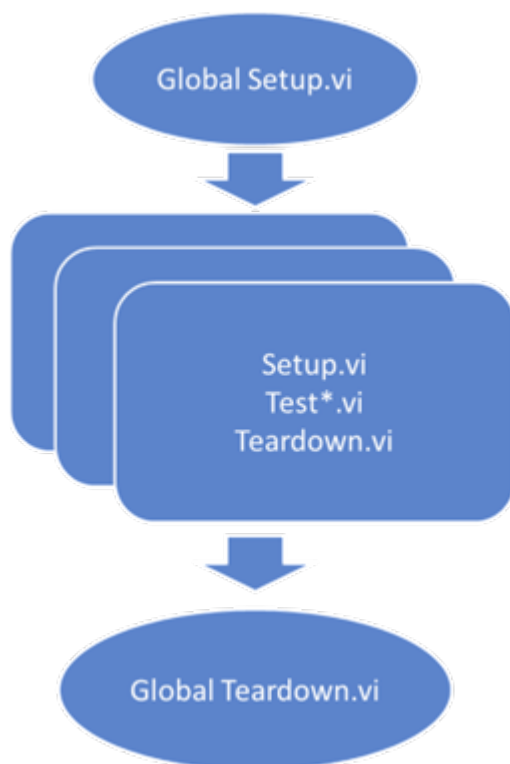
This repository contains a feature which may at some point get merged into LUnit, if there is enough interest. If you would like to see this feature in LUnit, please add star this repository to let us know.

This project requires LUnit 1.4.9 or later to work.

1.1 What?

A fixture is the environment in which a unit test is executed. In LUnit, the fixture may be either explicitly setup in the test method or implicitly using the Setup.vi and Teardown.vi overrides.

This project adds another hook for setting up a fixture around a test suite. In practice, this means that two new overrides (Global Setup.vi and Global Teardown.vi) are introduced and these are called before the first test (and after the last test) in a Test Case class.



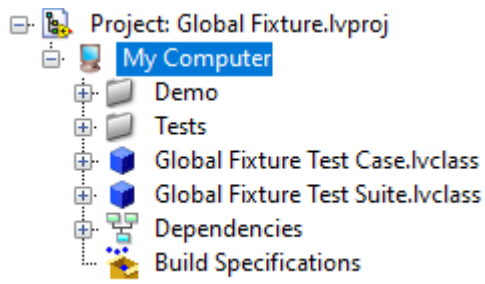
2. Why?

If the fixture is expensive to set up, test execution time may be improved significantly by not performing the setup/teardown cycle for each test. It is important that the fixture is not mutated by any of the case, or the tests may start to interact.

3. How?

This repository contains a class called Global Fixture Test Case.lvclass, which inherits from the LUnit Test Case.lvclass. To make a test case with global fixture, set the inheritance of the test case to point to this new class. Once you have inherited from the Global Fixture Test Case.lvclass class, there are two new VIs which you can override to add global setup and teardown functionality to your tests.

Under the hood a new derived test suite class is used, called Global Fixture Test Suite.lvclass. This class will need to be managed as a dependency for using the global fixture feature. The recommended way would be to just include the two classes in the client project test code.



4. Prerequisites

This project requires LUnit 1.4.9 or later to work. The code is in LabVIEW 2020.