

UEFI & EDK II Training



Continuous Integration (CI) Unit Test Framework for
Developer Validation Lab – Simics® QSP

tianocore.org

See also [LabGuide.md](#) for Copy & Paste examples in labs

This lab will show how to build and run a unit test sample code in the host-based environment.

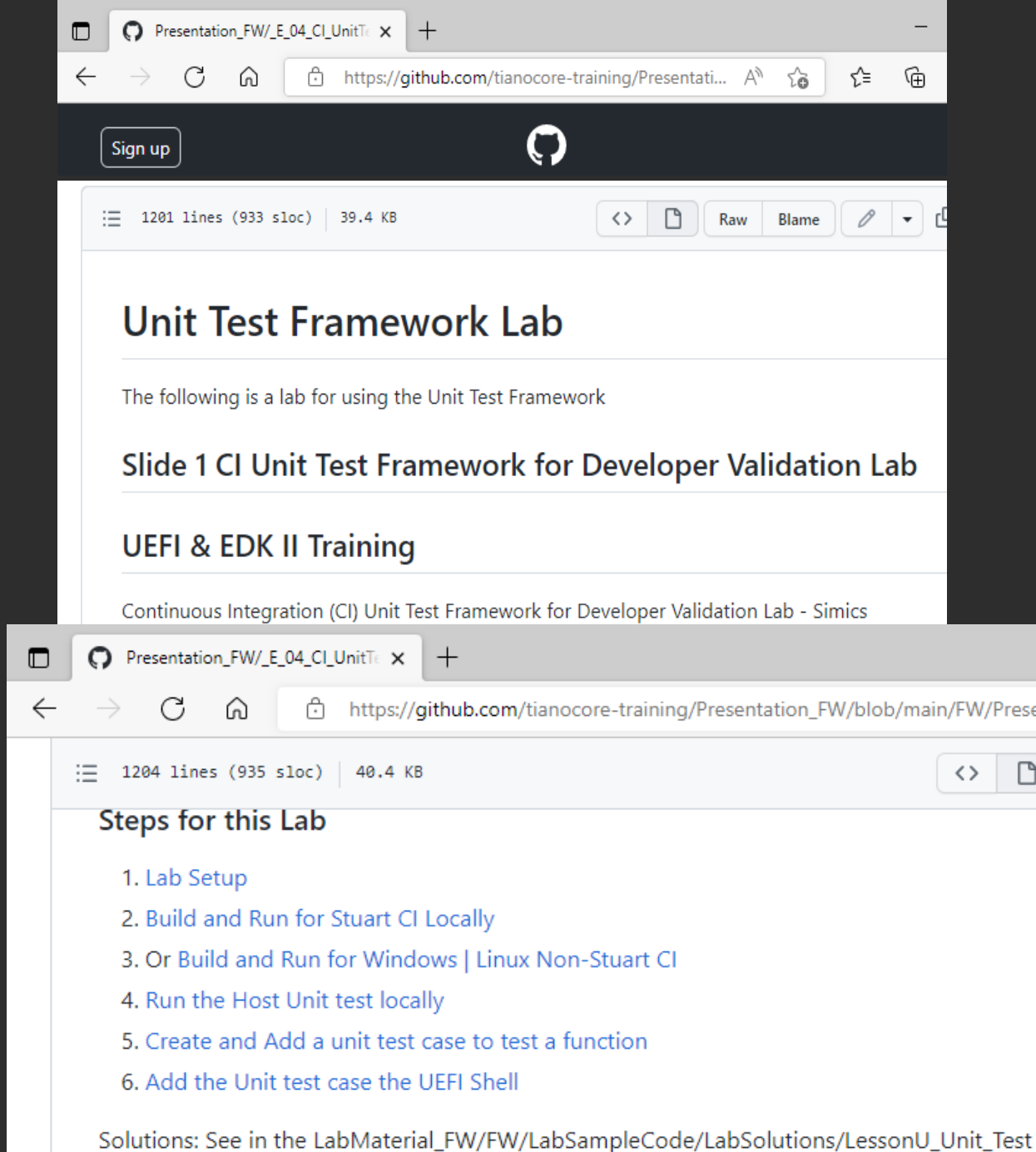
- Step by step guide for the Stuart CI build and run for the Sample Unit Test from `UnitTestFrameworkPkg`
- Steps to build for the Non-Stuart CI build and run
- Create a Host Unit Test Framework for a simple function
- Add a UEFI Shell Unit Test Framework using the Simics® Quick Start Platform (QSP)

- Windows 10: 
 - Stuart CI - Visual Studio VS2017 or VS2019
 - Non-Stuart CI - Visual Studio VS2015, VS2017 or VS2019
 - Windows SDK (for rc)
 - Windows WDK (for Capsules)
- Ubuntu 18.04 or Fedora 
 - GCC5 or greater
- Python 3.7.x or greater on Path
- Git on Path

Steps for this Lab

1. Lab Setup
2. Build and Run for Stuart CI Locally
3. OR Build and Run for Windows | Linux Non-Stuart CI
4. Run the Host Unit test locally
5. Create and Add a unit test case to test a function
6. Add the Unit test case the UEFI Shell

Solutions: See in the LabMaterial_FW/FW/
/LabSampleCode/LessonU_Unit_Test



The screenshot shows two overlapping browser windows displaying GitHub pages. The top window shows the 'Unit Test Framework Lab' page, which includes a 'Sign up' button, file statistics (1201 lines, 933 sloc, 39.4 KB), and a title 'Unit Test Framework Lab'. Below the title, it says 'The following is a lab for using the Unit Test Framework' and 'Slide 1 CI Unit Test Framework for Developer Validation Lab'. The bottom window shows the 'Steps for this Lab' page, which lists the same six steps as the text on the left. The URL in the address bar is 'https://github.com/tianocore-training/Presentation_FW/blob/main/FW/Prese'.

Unit Test Framework Lab

The following is a lab for using the Unit Test Framework

Slide 1 CI Unit Test Framework for Developer Validation Lab

UEFI & EDK II Training

Continuous Integration (CI) Unit Test Framework for Developer Validation Lab - Simics

Steps for this Lab

1. Lab Setup
2. Build and Run for Stuart CI Locally
3. Or Build and Run for Windows | Linux Non-Stuart CI
4. Run the Host Unit test locally
5. Create and Add a unit test case to test a function
6. Add the Unit test case the UEFI Shell

Solutions: See in the LabMaterial_FW/FW/LabSampleCode/LabSolutions/LessonU_Unit_Test

Setup Lab material from previous lab

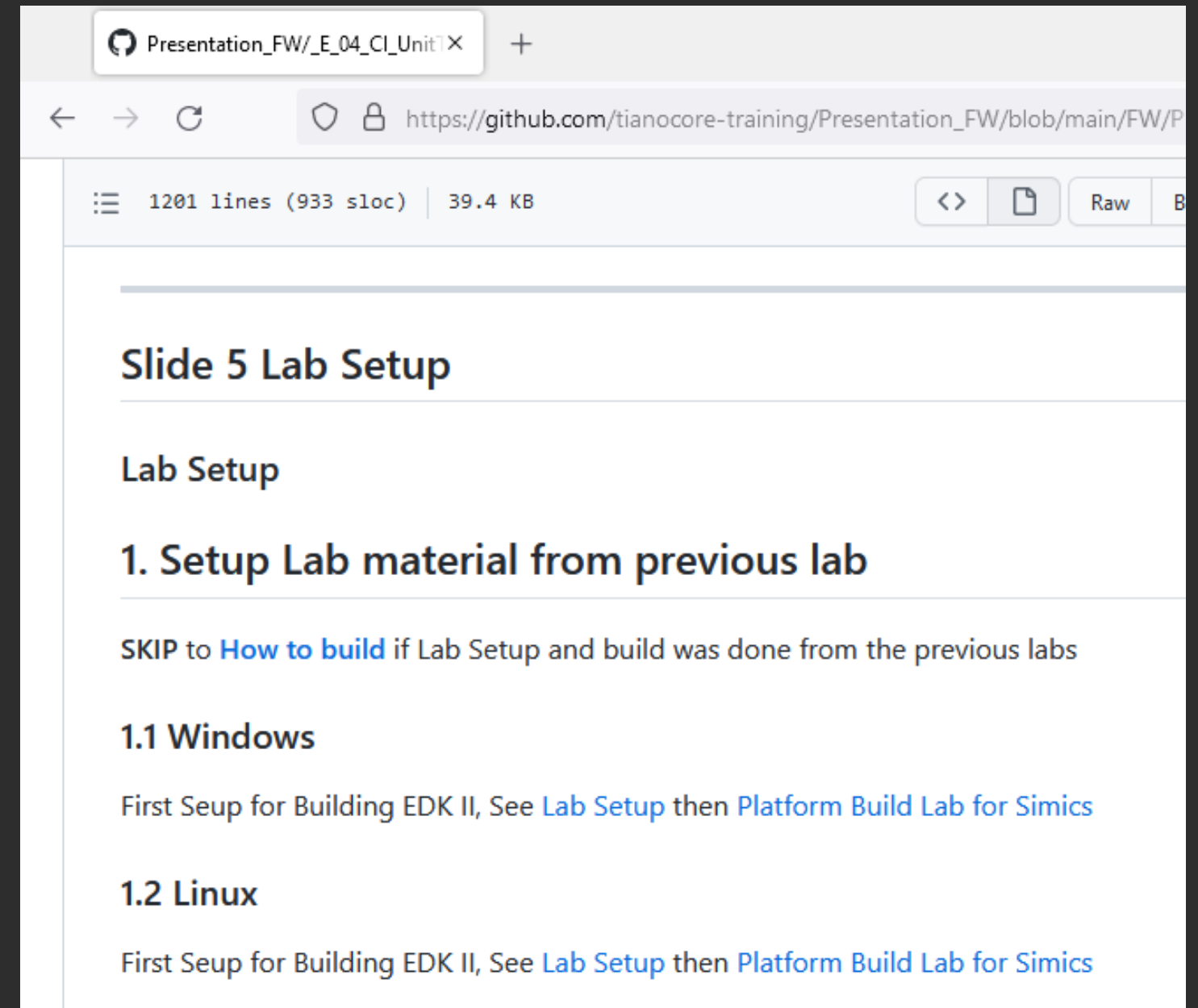
SKIP to [How to Build](#) if already done

Windows 

- First Setup for Building EDK II, See [Lab Setup](#) then [Platform Build Lab for Simics](#)

Linux 

- First Setup for Building EDK II, See [Lab Setup](#) then [Platform Build Lab for Simics](#)



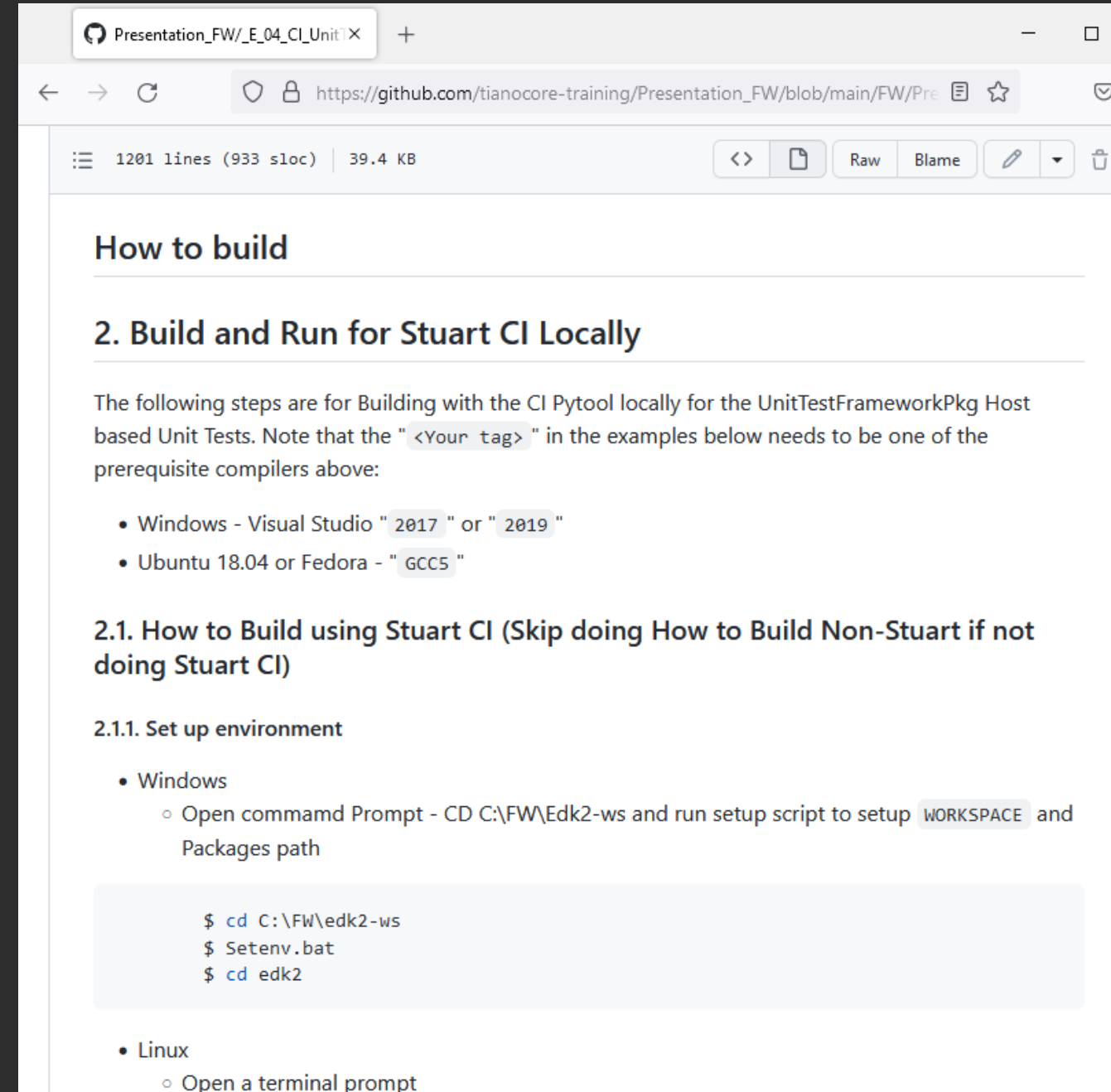
How to Build Unit Tests

Continue following the Lab Guide Starting at [How to Build](#)

Build either Stuart CI
or Non-Stuart CI

- Windows 

- Linux 



The screenshot shows a GitHub repository page for 'tianocore-training/Presentation_FW'. The file 'Presentation_FW/_E_04_CI_Unit' is selected, showing 1201 lines (933 sloc) and 39.4 KB. The page content is titled 'How to build' and includes a section '2. Build and Run for Stuart CI Locally'. The text explains that the following steps are for building with the CI Pytool locally for the UnitTestFrameworkPkg Host based Unit Tests. It notes that the '<Your tag>' in the examples below needs to be one of the prerequisite compilers above:

- Windows - Visual Studio "2017" or "2019"
- Ubuntu 18.04 or Fedora - "GCC5"

Section 2.1. How to Build using Stuart CI (Skip doing How to Build Non-Stuart if not doing Stuart CI)

2.1.1. Set up environment

- Windows
 - Open command Prompt - CD C:\FW\Edk2-ws and run setup script to setup WORKSPACE and Packages path

```
$ cd C:\FW\edk2-ws
$ Setenv.bat
$ cd edk2
```

- Linux
 - Open a terminal prompt

Unit Test Framework Package Overview

– [Link](#)

Continuous Integration (CI) Configuring for Unit Tests – [Link](#)

Code Examples of Unit Test Cases

- [Sample Unit Test](#)
- [BaseSafeIntLib Unit Test](#)
- [BaseLib Unit Test](#)
- [DxeResetSystemLib Unit Test](#)
- [MtrrLibUnitTest](#)
- Cmocka Edk II Unit Test ChefCook example: [link](#)

Questions?



Return to Main Training Page



Return to Training Table of contents for next presentation [link](#)



ACKNOWLEDGEMENTS

Redistribution and use in source (original document form) and 'compiled' forms (converted to PDF, epub, HTML and other formats) with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code (original document form) must retain the above copyright notice, this list of conditions and the following disclaimer as the first lines of this file unmodified.

Redistributions in compiled form (transformed to other DTDs, converted to PDF, epub, HTML and other formats) must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS DOCUMENTATION IS PROVIDED BY TIANOCORE PROJECT "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL TIANOCORE PROJECT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS DOCUMENTATION, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2021-2022, Intel Corporation. All rights reserved.