tianocore

UEFI & EDK II Training

PLATFORM BUILD LAB WINDOWS EMULATOR

tianocore.org

See also LabGuide.md for Copy & Paste examples in labs

PLATFORM BUILD LABS

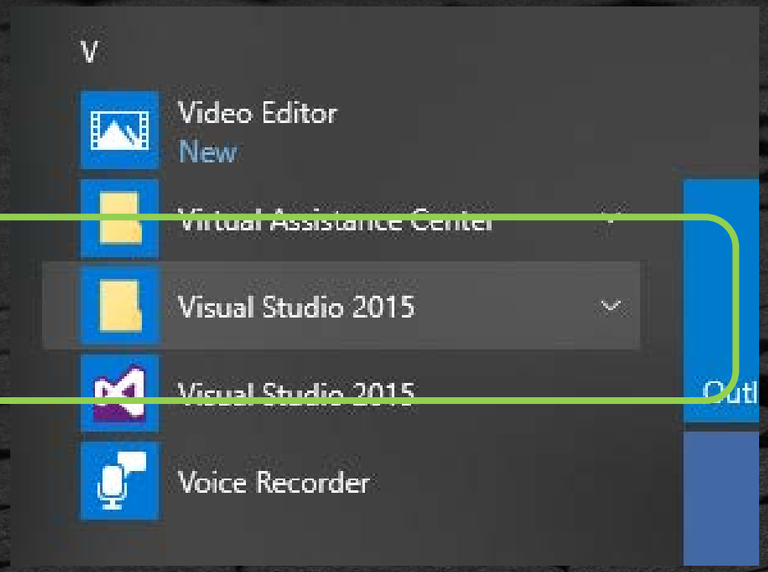
- Pin Visual Studio Command Prompt to Windows Task Bar
- Build a EDK II Platform using Emulator package
- Run the Emulator in Windows

tianocore

PIN VS COMMAND PROMPT

Pin the Visual Studio Command prompt to Windows Task Bar

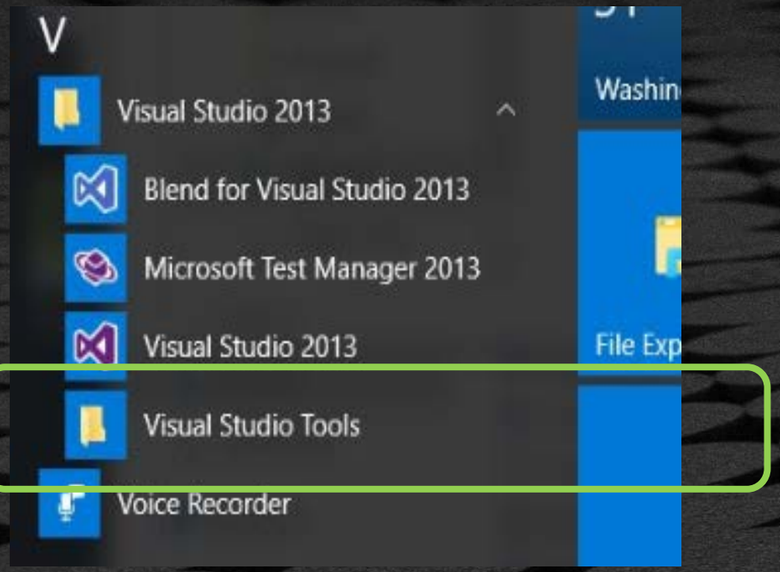
PIN VS COMMAND PROMPT



Steps to Pin Visual Studio Command Prompt to task bar for Windows 10

1. Using the Start menu in Windows 10, Left Click on "Windows Key" Lower Left
2. Scroll down from the scroll bar on the right until "**Visual Studio 201*n***"
3. Left Click "**Visual Studio 201*n***"

PIN VS COMMAND PROMPT



VS 2013

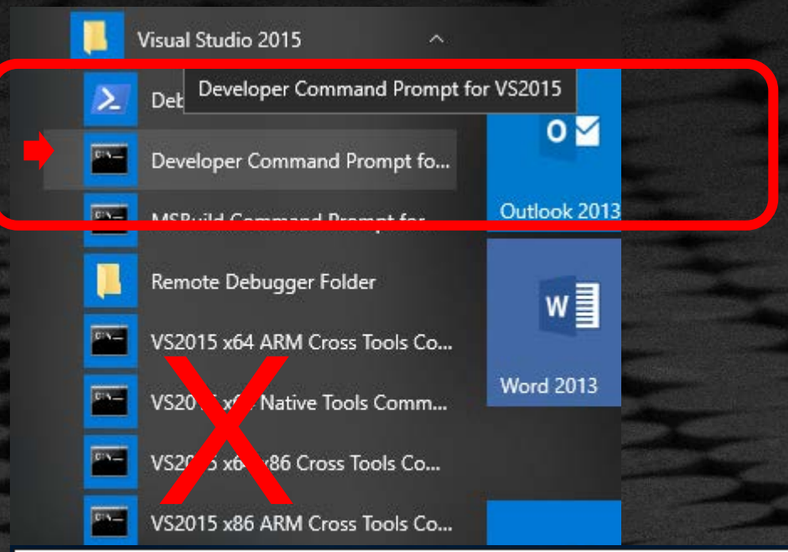
4. Left Click "**Visual Studio Tools**"

This will open another Windows file explorer window

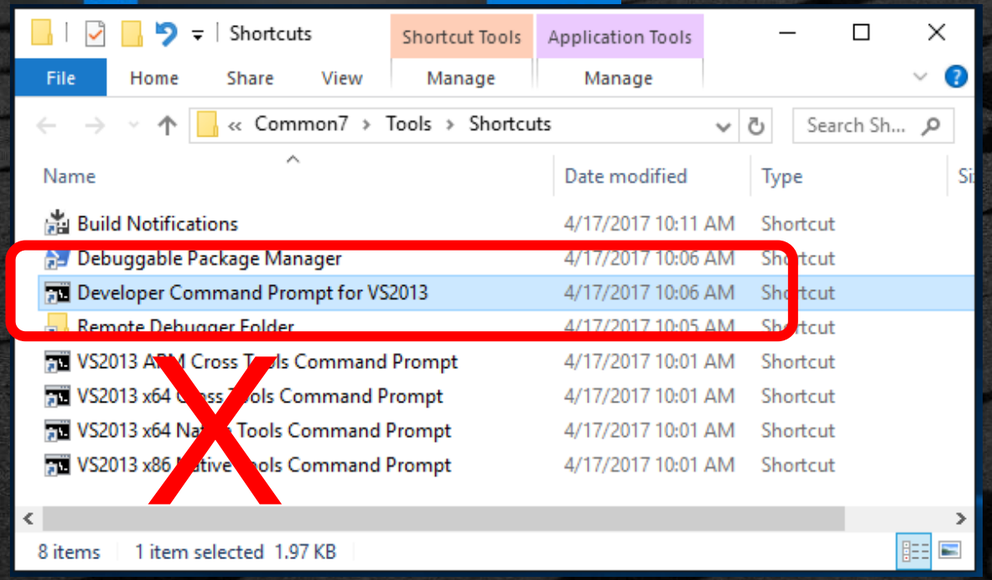
Note: *VS 2013 example, other version of VS maybe different*

PIN VS COMMAND PROMPT

VS 2015 VS 2017 is similar



VS 2013

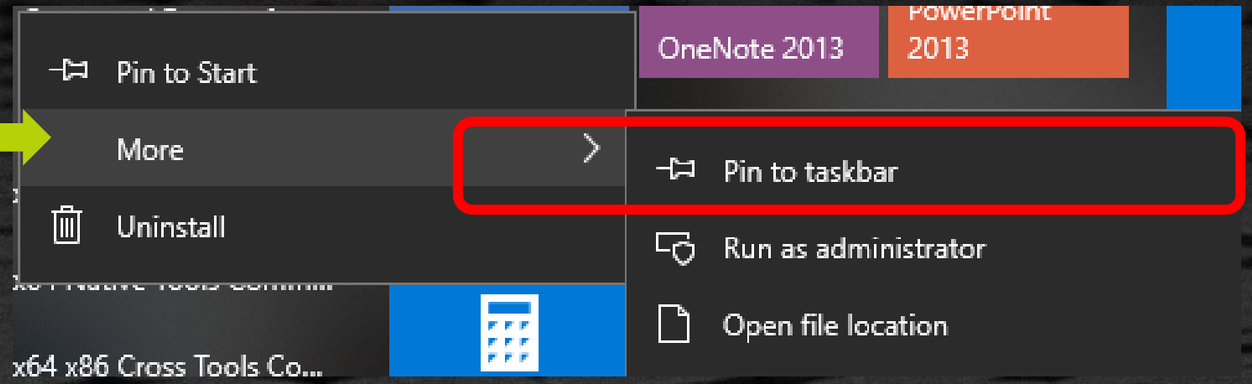


5. Select
“**Developer Command Prompt for VS201n**”

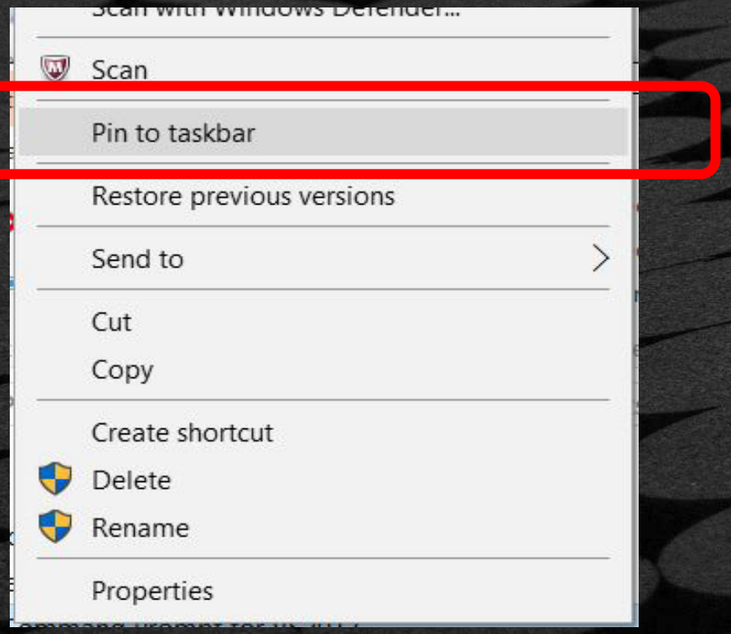
6. Right Click to open Windows dialog box

Do not use any of the other
“.. Command Prompts”

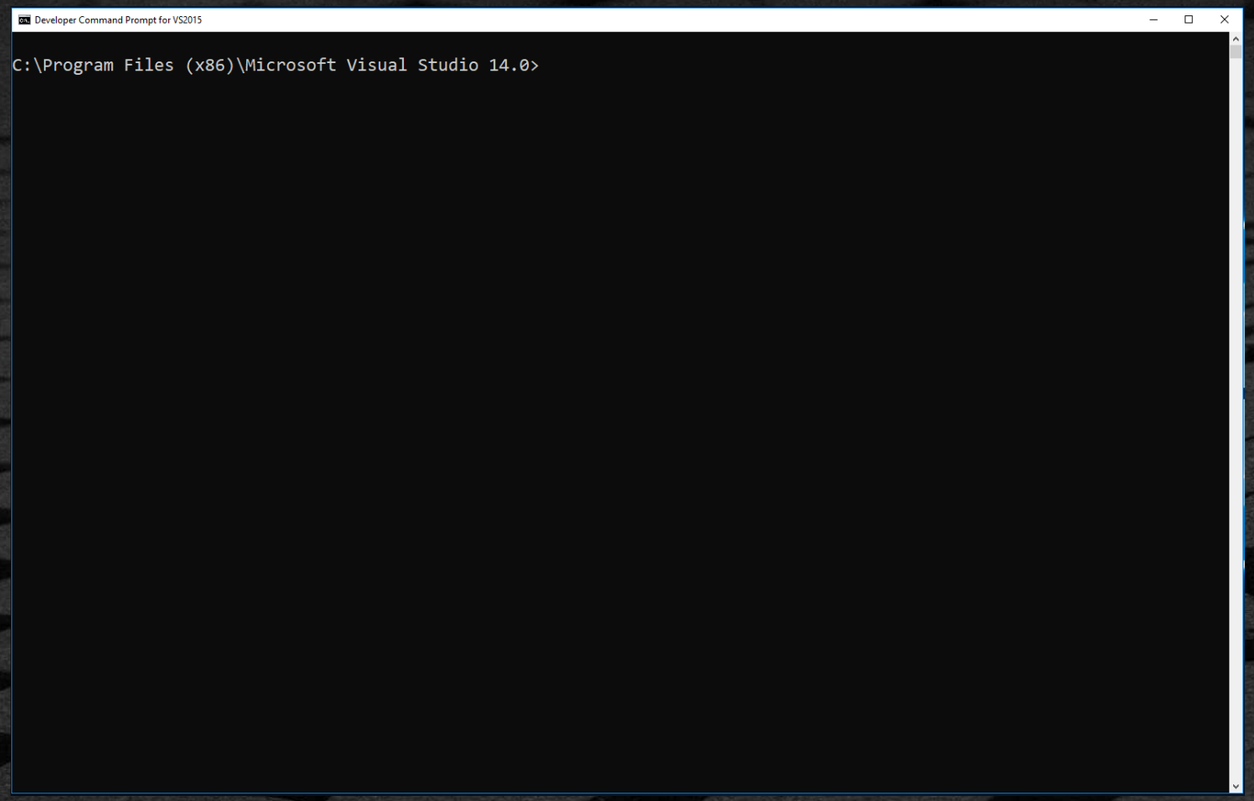
PIN VS COMMAND PROMPT



7. Left Click on “Pin to taskbar”



PIN VS COMMAND PROMPT



8. Open VS Command Prompt”

All Windows Labs use this short-cut to Build Edk II platforms and projects using Windows Visual Studio :
2010 / 2012 / 2013 / 2015 or 2017

END OF PIN VS PROMPT

BUILD EMULATOR

Setup EmulatorPkg to build and run emulation with Windows

Prerequisites

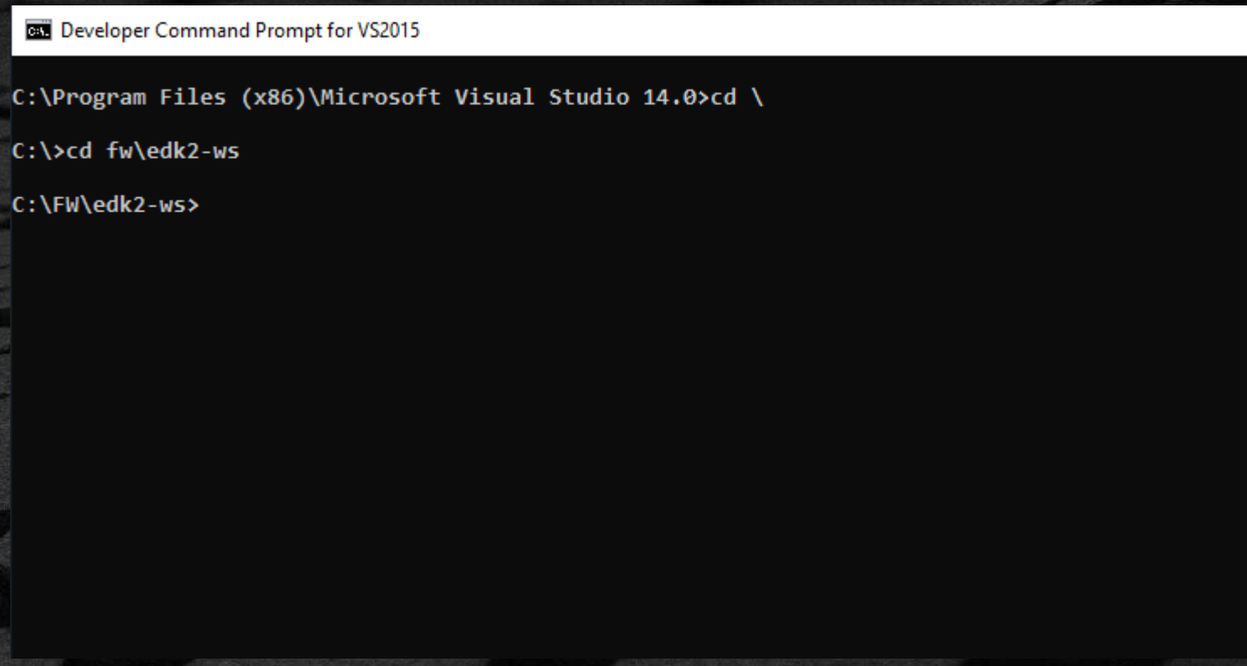
- Windows 10: – Done Before Class
  - Continuous Integration (CI) - Stuart CI Build with Visual Studio VS2017 or VS2019
  - Non-Stuart CI - Visual Studio VS2015, VS2017 or VS2019
  - Windows SDK (for rc)
  - Windows WDK (for Capsules)
- Python 3.7.x or greater and `/Scripts` directories on Path: Link to download
- Git for Windows on Path : Link
- NASM for Win64 : Link

Create Workspace Directory

Open Windows Command Prompt

Make new directory for Workspace:

```
$ cd /
$ Mkdir FW
$ cd FW
$ Mkdir edk2-ws
$ cd edk2-ws
```



Download the EDK II Source Code

Download the open source EDK II from Github

From the command prompt use “git clone” to download

```
C:\FW\edk2-WS> git clone https://github.com/tianocore-training/edk2.git
C:\FW\edk2-WS> git clone https://github.com/tianocore/edk2-libc.git
```

Download the Submodules and Checkout the Lab Branch

```
C:\FW\edk2-WS> Cd edk2
C:\FW\edk2-WS\edk2> git checkout LabBranch
C:\FW\edk2-WS\edk2> git submodule update --init
C:\FW\edk2-WS> Cd ..
```

Note if behind a firewall, set PROXYS FIRST

```
$ git config --global https.proxy=proxy.hf.intel.com:911
$ git config --global http.proxy=proxy.hf.intel.com:911
```

Download Lab Material

Download the Lab_Material_FW.zip from : github.com Lab_Matrial_FW.zip
OR
Use git clone to download the Lab_Material_FW

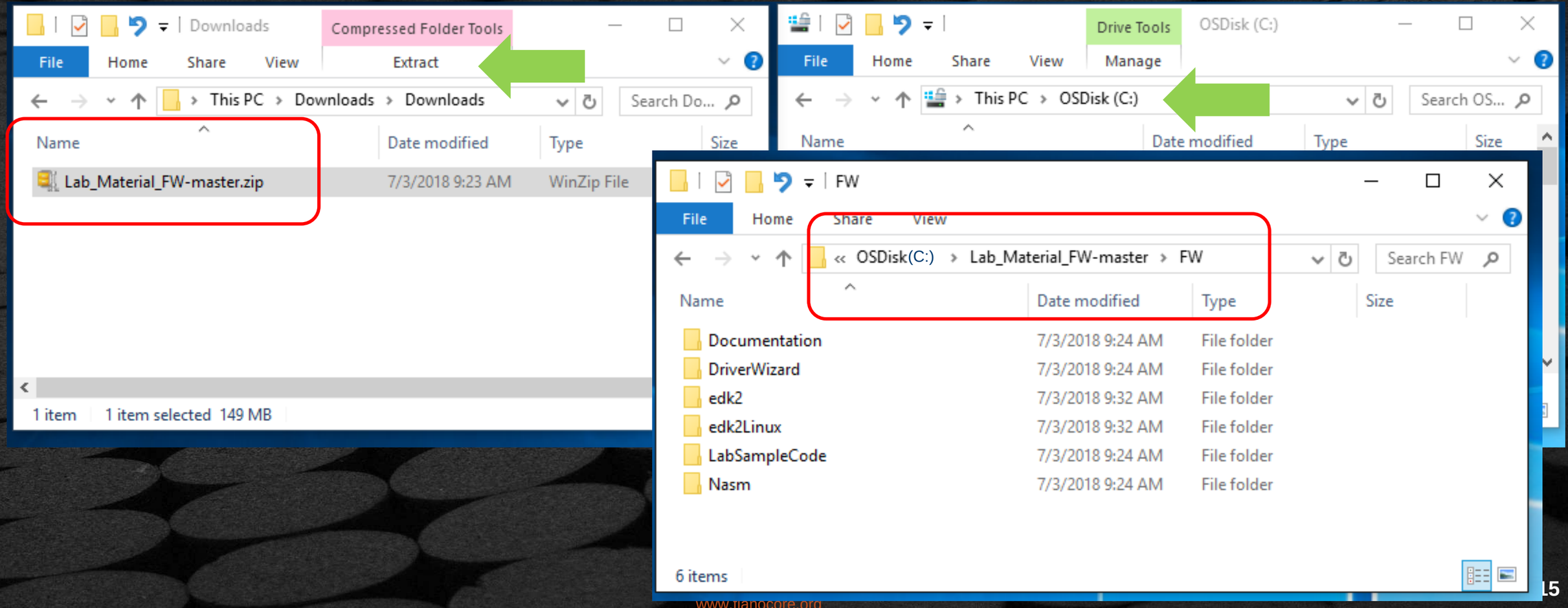
```
C:\> git clone https://github.com/tianocore-training/Lab_Material_FW.git
```

Directory Lab_Material_FW will be created

```
FW
  - Documentation
  - DriverWizard
  - edk2-ws
  - LabSampleCode
  - Nasm
```

Build EDK II
-Extract the Source

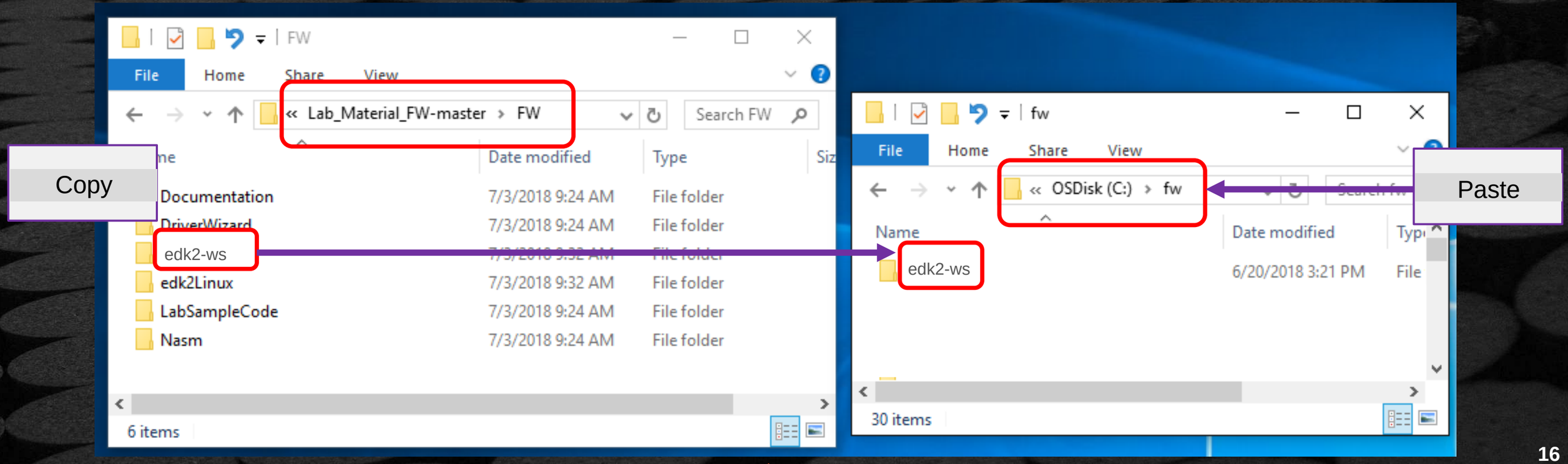
Extract the Downloaded `Lab_Material_FW-master.zip` to `C:\`



Build EDK II
- Copy edk2-ws

From the downloaded Lab_Material_FW folder, **copy** and **paste** folder "..\edk2-ws" to C:/FW

Note: Overwrite existing files and directories

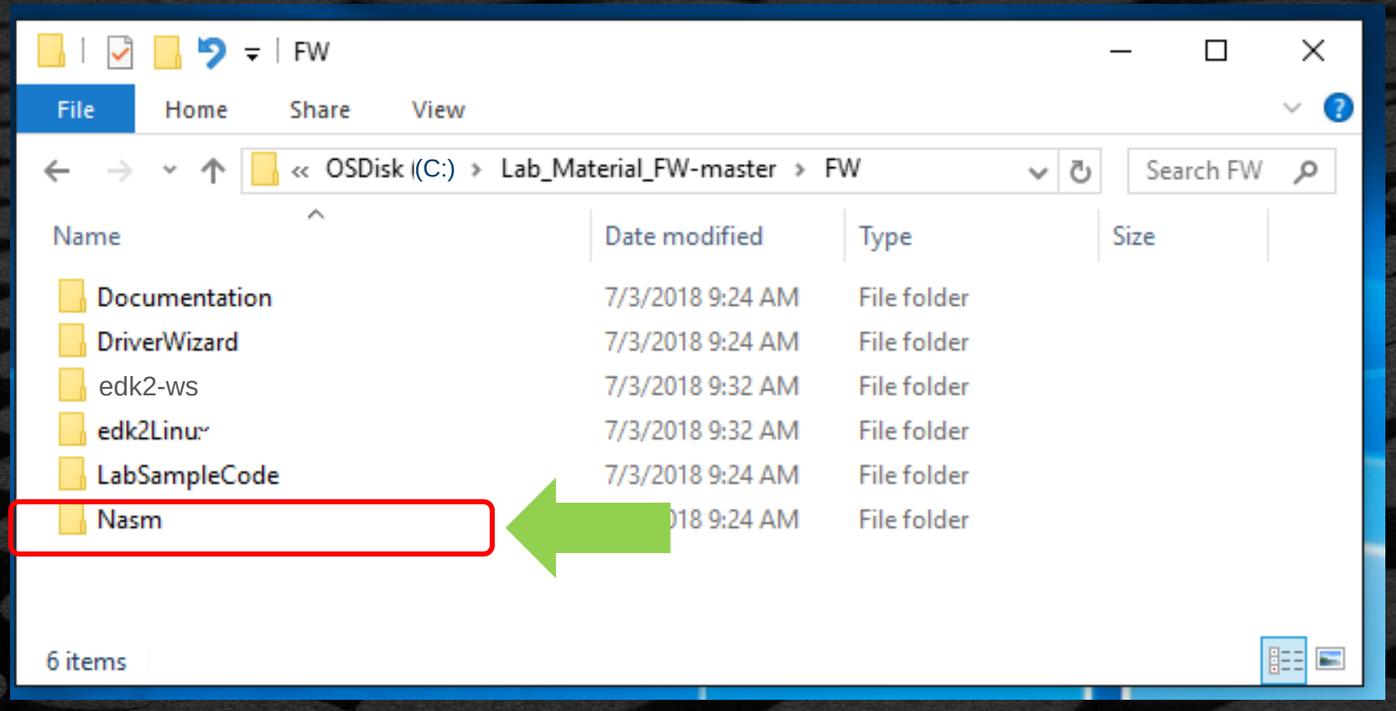


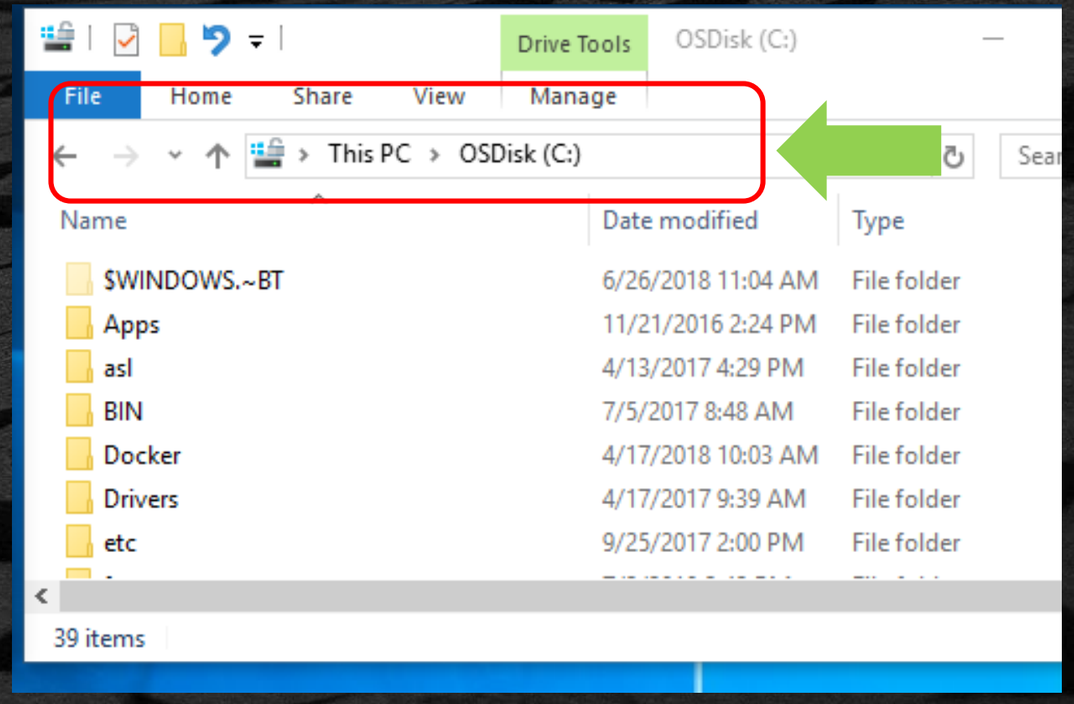
tianocore

Build EDK II
-Get Nasm

Copy `Nasm` directory to `C:\`

(creating `C:\Nasm` directory)





STUART CI BUILD EMULATORPKG

tianocore

Stuart CI Build EDK II

1. Install the pip requirements (Note, Proxy option needed behind a firewall)

```
$ pip install --upgrade -r pip-requirements.txt --proxy http://proxy-chain.intel.com:911
```

2. Get the code dependencies (done only when submodules change)

```
$ stuart_setup -c EmulatorPkg/PlatformCI/PlatformBuild.py TOOL_CHAIN_TAG=<Your TAG> -a X64
```

3. Update other dependencies (done on new VS Command Prompt)

```
$ stuart_update -c EmulatorPkg/PlatformCI/PlatformBuild.py TOOL_CHAIN_TAG=<Your TAG> -a X64
```

4. Build the BaseTools (done only when BaseTools change and first time)

```
$ python BaseTools\Edk2ToolsBuild.py -t <Your TAG>
```

5. Compile the EmulatorPkg

```
$ stuart_build -c EmulatorPkg/PlatformCI/PlatformBuild.py TOOL_CHAIN_TAG=<Your TAG> -a X64
BLD_*_ADD_SHELL_STRING=1
```

Where "<Your TAG>" is either VS2017 or VS2018

Output from CI Stuart Build

Developer Command Prompt for VS 2017

```
INFO -    Region Size = 0x2000
INFO -    Region Name = DATA
INFO -
INFO - Generate Region at Offset 0x590000
INFO -    Region Size = 0x10000
INFO -    Region Name = None
INFO -
INFO - GUID cross reference file can be found at d:\lab\fw\edk2-ws\edk2\Build\EmulatorX64\DEBUG_VS2017\FV\Guid.xref
INFO -
INFO - FV Space Information
INFO - FVRECOVERY [48%Full] 5767168 total, 2769848 used, 2997320 free
INFO -
INFO - - Done -
INFO - Build end time: 13:17:07, Jul.27 2020
INFO - Build total time: 00:01:33
INFO -
INFO - -----------------------------------------------
INFO - --------------Cmd Output Finished---------------
INFO - --------- Running Time (mm:ss): 01:33 ----------
INFO - ----------- Return Code: 0x00000000 ------------
INFO - -----------------------------------------------
PROGRESS - Running Post Build
DEBUG - Plugin Success: Windows RC Path Support
DEBUG - Plugin Success: Windows Visual Studio Tool Chain Support
INFO - Writing BuildToolsReports to D:\lab\FW\edk2-ws\edk2\Build\EmulatorX64\DEBUG_VS2017\BUILD_TOOLS_REPORT
DEBUG - Plugin Success: Build Tools Report Generator
PROGRESS - End time: 2020-07-27 13:17:07.515485  Total time Elapsed: 0:01:37
SECTION - Log file is located at: D:\lab\FW\edk2-ws\edk2\Build\BUILDLOG_EmulatorPkg.txt
SECTION - Summary
PROGRESS - Success
```

Finished build

NON-STUART CI BUILD EMULATORPKG

Skip if doing Stuart CI

Non-Stuart CI Build EDK II
- build BaseTools

Open VS Command prompt & Cd to workspace directory

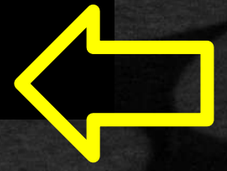
```
$> cd C:\FW\edk2-ws
```

Setup the local environment: (see batch file setenv.bat)

```
$> set WORKSPACE=%CD%
$> set PACKAGES_PATH=%WORKSPACE%\edk2;%WORKSPACE%\edk2-libc
```

Invoke `Edksetup.bat` from directory `C:/FW/edk2-ws/edk2` to Build `BaseTools`

```
$> cd edk2
$> edksetup.bat Rebuild
```



Building `BaseTools` only needs to be done once but setting up local environment and edksetup.bat needs to be done each new VS prompt session

Non Stuart CI Build EDK II

EmulatorPkg – Non Stuart CI Build with edk2

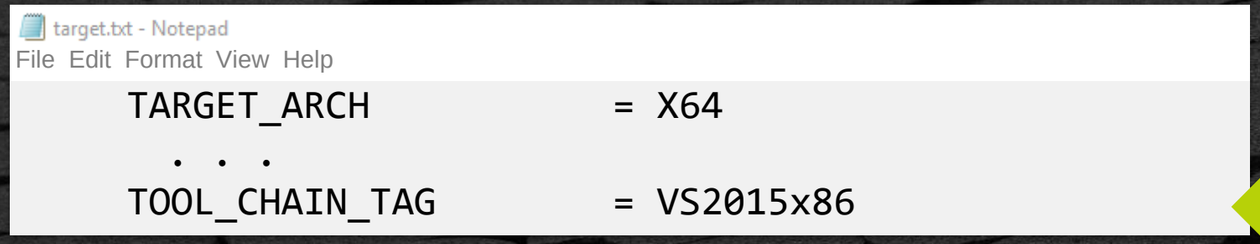
-Update Target.txt

Invoke Edksetup.bat

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

Edit the file Conf/target.txt ***(change*** TOOL_CHAIN_TAG***)***

```
notepad Conf/target.txt
```



VS version	TOOL_CHAIN_TAG
2015	**VS2015x86**
2017	**VS2017**
2019	**VS2019**

Save and Exit

Build EmulatorPkg

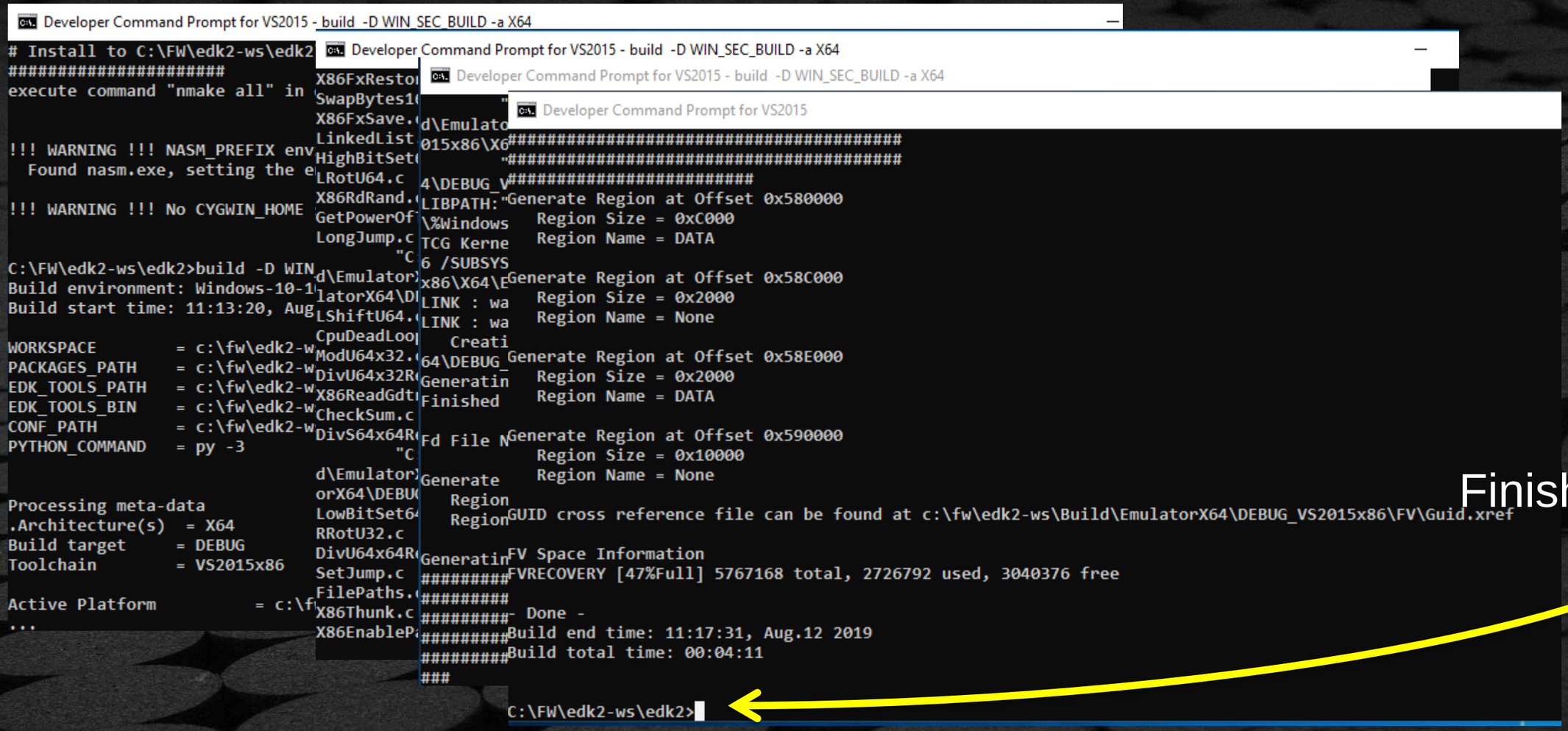
```
$> build -D ADD_SHELL_STRING -a X64
```

Possible Build Errors

1. If you get a BUILD Error: Error “`C:/Program` “ not found
   - First check that you have opened Visual Studio and installed the “C++”
   - Open Visual Studio and create a “C++” project
   - (This will take some time to install)
2. If you get a BUILD Error: Check if RC.Exe compiler not found is the error -here
3. If you get a BUILD Error: fatal error C1041: cannot open program database … Check here

tianocore

Build EDK II
-Inside VS Prompt



Finished build

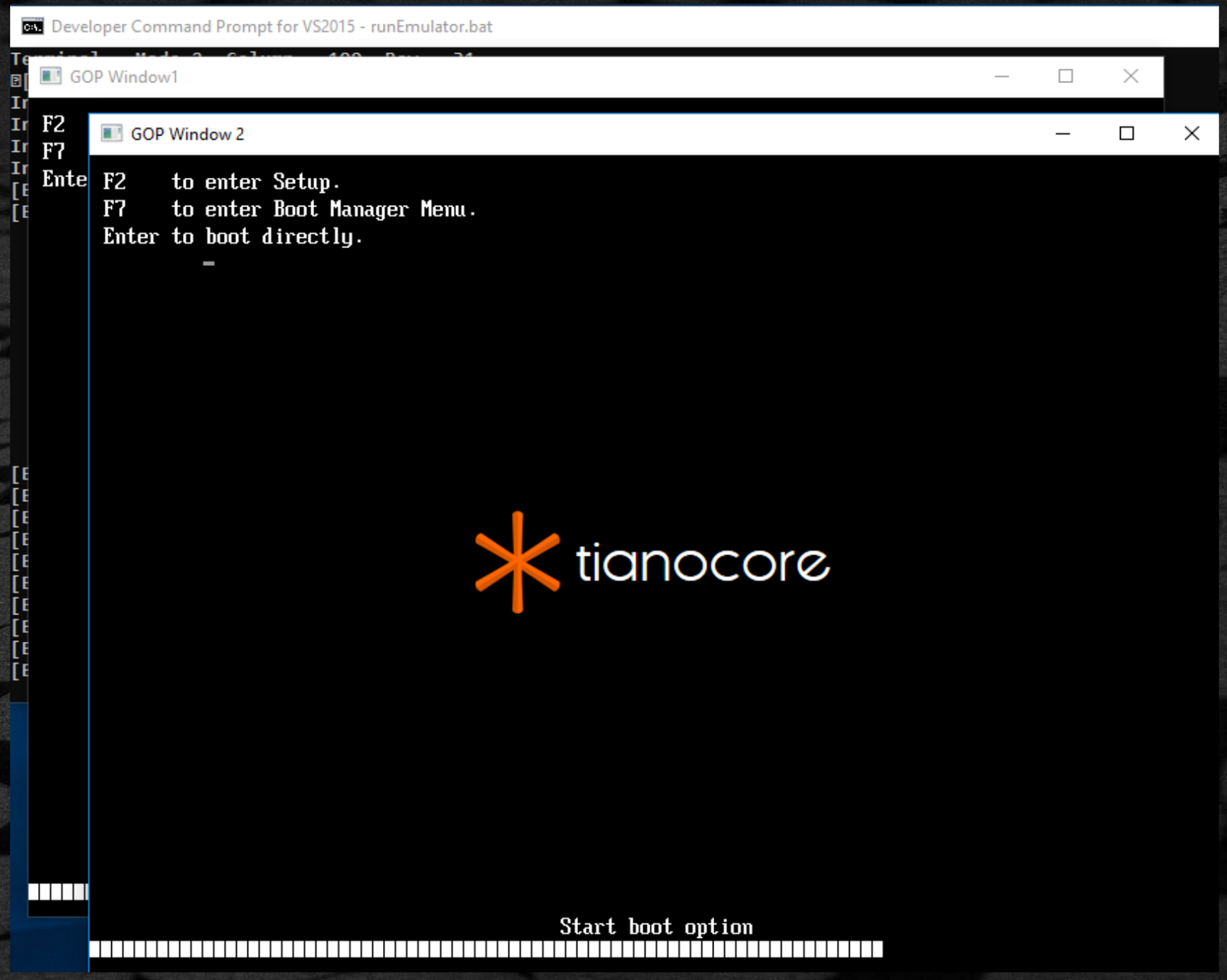
RUN THE EMULATOR

Invoke Emulation

From the command prompt

```
$> RunEmulator.bat
```

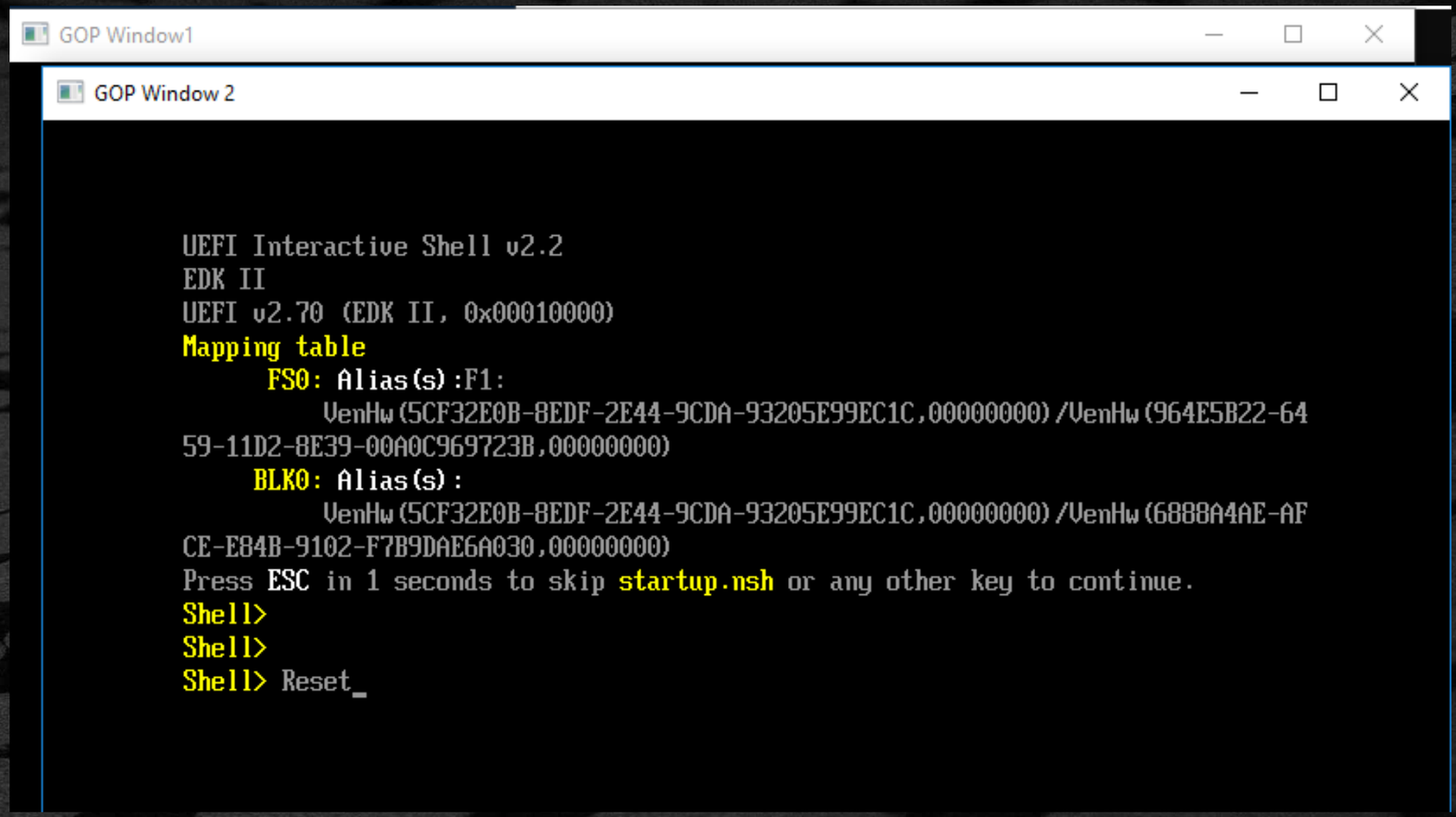
Or
run `WinHost.exe` from:
Build/ . . ./X64 directory



Notice 2 “GOP Window n" opened

EMULATOR AT SHELL PROMPT

Type : “Reset” to exit



tianocore

SUMMARY

- Pin Visual Studio Command Prompt to Windows Task Bar
- Build a EDK II Platform using Emulator package
- Run the Emulator in Windows

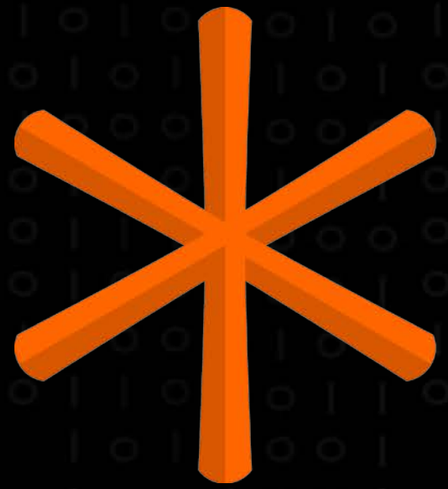
tianocore

Questions?

Return to Main Training Page

Return to Training Table of contents for next presentation

Link



tianocore

ACKNOWLEDGMENTS

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) 2020, Intel Corporation. All rights reserved.

tianocore

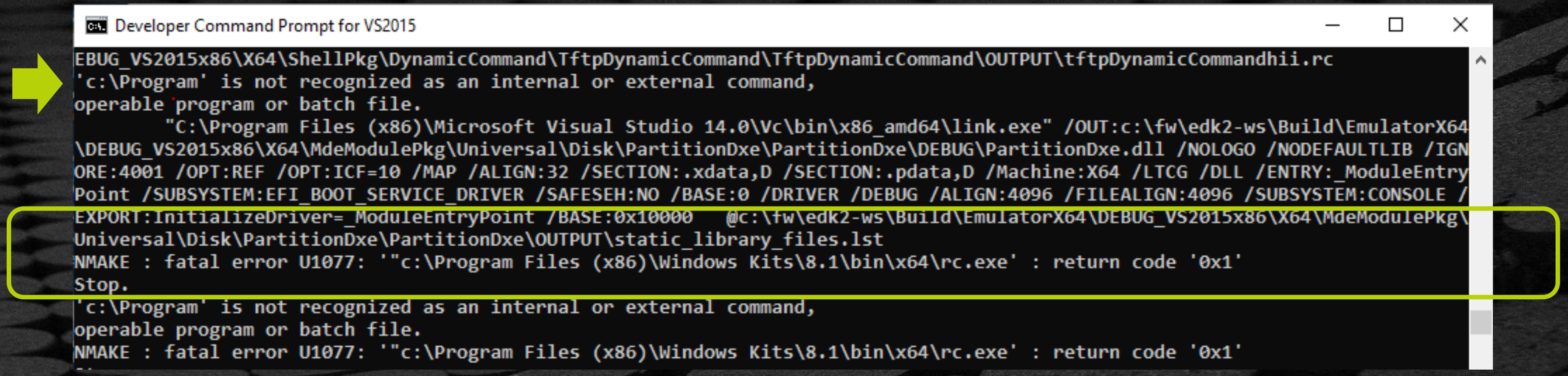
BACKUP

tianocore

BUILD ERRORS

Build Error- RC.exe

Error message:



Find where the RC.EXE is located on your VS Installation:

Example (VS 2015): The RC.exe is located on this machine:

```
C:\Program Files (x86)\Windows Kits\8.1\bin\x64
```

Edit Conf\tools_def.txt

Build Error- RC.exe Cont.

Edit Conf\tools_def.txt

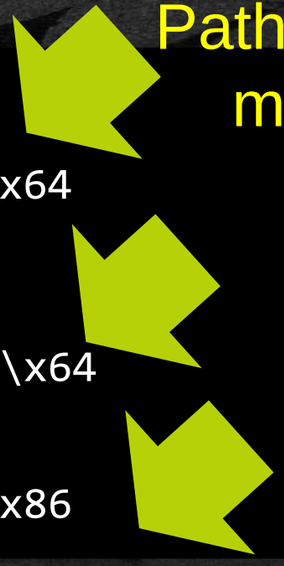
Search for your installation of Visual Studio (2013, 2015, 2017) “RC.EXE”
Probably in path C:\Program Files (x86)\Windows Kits\

Update according to the path for where the RC.EXE is found

```
# Microsoft Visual Studio 2013 Professional Edition
DEFINE WINSDK8_BIN       = c:\Program Files\Windows Kits\8.1\bin\x86\
DEFINE WINSDK8x86_BIN    = c:\Program Files (x86)\Windows Kits\8.1\bin\x64

# Microsoft Visual Studio 2015 Professional Edition
DEFINE WINSDK81_BIN       = c:\Program Files\Windows Kits\8.1\bin\x86\
DEFINE WINSDK81x86_BIN    = c:\Program Files (x86)\Windows Kits\8.1\bin\x64

# Microsoft Visual Studio 2017 Professional Edition
DEFINE WINSDK10_BIN       = C:\Program Files (x86)\Windows Kits\10\bin\x86
```



Copy and Paste RC error: Link

Build Error: fatal error C1041:

Build Error from fatal error C1041: cannot open program database

This Error is usually because the location you are building is being shared by another application in Windows. Example: Syncplicity may cause this

Error Message:

```
k:\fw\edk2\MdePkg\Library\BaseLib\LinkedList.c : fatal error C1041: cannot open program
database
'k:\fw\edk2\build\nt32ia32\debug_vs2013x86\ia32\mdepkg\library\baselib\baselib\vc120.pdb'; if
multiple CL.EXE write to the same .PDB file, please use /FS
NMAKE : fatal error U1077: '"C:\Program Files (x86)\Microsoft Visual Studio
12.0\Vc\bin\cl.exe"' : return code '0x2'
Stop.
```

Solution: Try using a Workspace that is not shared