SEE LINK BELOW FOR INSTRUCTIONS ON COMPATIBILITY TOOL DEMO
https://www.intel.com/content/www/us/en/developer/articles/training/intel-dpcpp-compatibility-tool-training.html

Intel® DPC++ Compatibility Tool Self-Guided Jupyter Notebook Tutorial

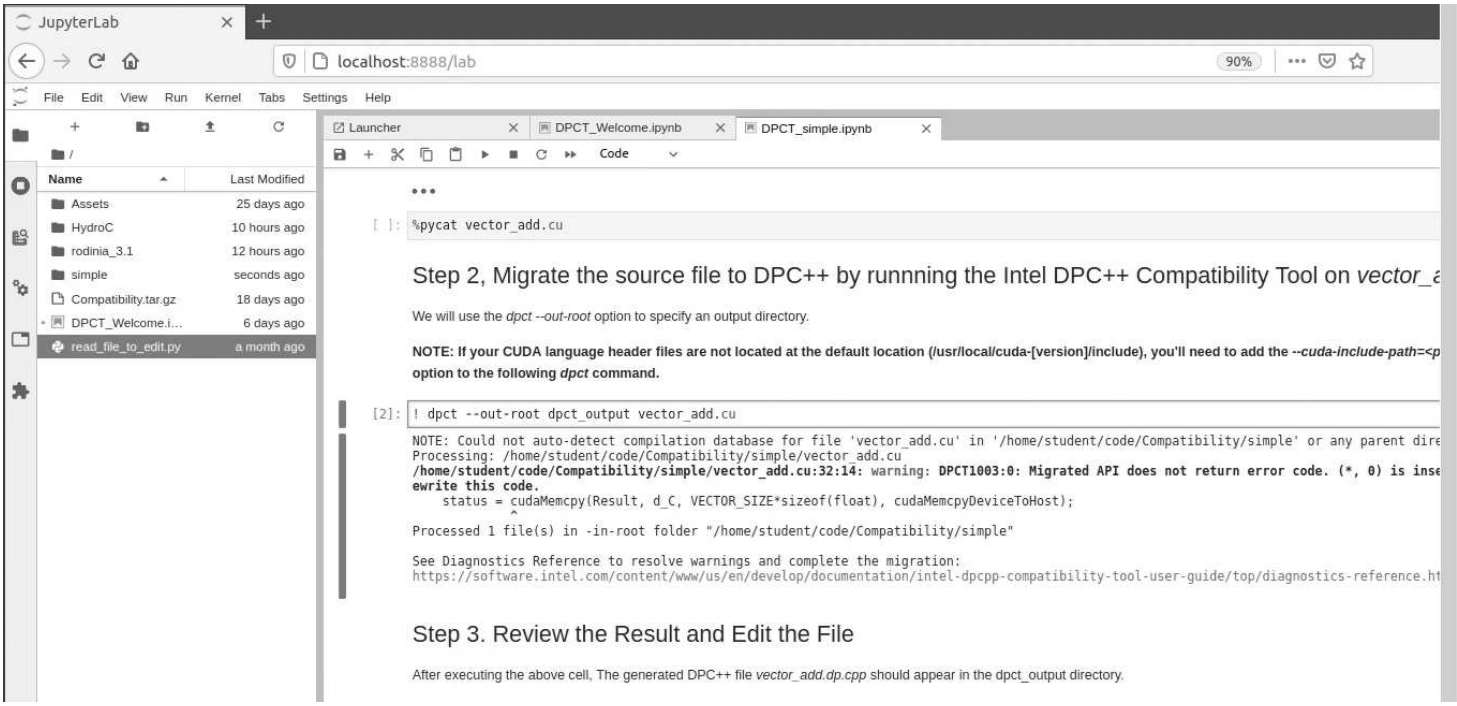
Published: 11/02/2020

By Xinyu Karl Qi

Files:	dpct-notebook-master.tar.gz
Licenses:	MIT Rodinia License CeCILL
Software:	Intel® DPC++ Compatibility Tool Intel® oneAPI DPC++ Compiler
OS:	Linux
Prerequisites:	Familiarity with SYCL*

The Intel® DPC++ Compatibility Tool (Compatibility Tool) assists in the migration of a developer's program that is written in CUDA* to a program written to SYCL* standards.

Run this self-guided tutorial on your local machine to learn about the Compatibility Tool in an interactive JupyterLab environment. These Jupyter notebooks will guide through the migration of a simple example as well as two real-world examples.



Run the self-guided tutorial

This tutorial requires the Intel® oneAPI Base Toolkit and JupyterLab. Certain CUDA language header files may also need to be accessible to the Compatibility Tool. If those components are not already installed, follow the direction in the “Install necessary tools” section first.

1. Download the tutorial files dpct-notebook-master.tar.gz file.
2. Uncompress the file into a directory of your choosing.
   - tar -xzf dpct-notebook-master.tar.gz
3. Ensure the oneAPI environment is set by running “source /opt/intel/oneapi/setvars.sh”.
4. Start JupyterLab and open the notebooks.
   - jupyter-lab DPCT_Welcome.ipynb
5. Follow the steps in the self-guided Jupyter notebook.

Install the necessary tools

1. The Intel® DPC++ Compatibility Tool and Intel® oneAPI DPC++ Compiler are part of the Intel® oneAPI Base Toolkit. If you haven't already done so, Install the Intel® oneAPI Base Toolkit by following the instructions in the Installation Guide .
2. Certain CUDA language header files may need to be accessible to the Intel® DPC++ Compatibility Tool. The Compatibility Tool looks for the CUDA header files in the default /usr/local/cuda[-version]/include directory. If your CUDA headers are not in the default directory, make note of CUDA include path so that you’ll be ready to supply the path in the Compatibility Tool command line.
3. Set up the oneAPI environment by running “source /opt/intel/oneapi/setvars.sh”
4. This tutorial requires JupyterLab, install JupyterLab if you haven’t already done so. From the command line:
   - conda install -c conda-forge jupyterlab

References

Visit the Intel® DPC++ Compatibility Tool User Guide for additional information about the tool. See the Release Notes for known issues and the most up-to-date information.