# Cài đặt thư viện PMDK.

## Build PMDK từ source trên Ubuntu 20.04:

**Trước khi tiến hành cài đặt PMDK ta cần cài đặt các gói sau:**

* autoconf
* pkg-config
* libndctl-devel (v63 trở lên)
* libdaxctl-devel (v63 trở lên)
* pandoc (cho document, cần thiết trong quá trình cài đặt)
* libfabric (v1.4.2 trở lên) – cần cho thư viện librpmem

**Build PMDK:**

Để build từ source, ta clone repository này về:

$ git clone <https://github.com/pmem/pmdk>

$ cd pmdk

Nếu ta muốn chọn một phiên bản ổn định, hãy xem xét release tag sau, nếu không ta có thể bỏ qua bước này để build phiên bản mới nhất:

$ git checkout tags/1.10

Và ta bắt đầu build bằng câu lệnh:

$ make

Sau khi build xong, ta tiến hành cài đặt vào vị trí mặc định trên hệ thống bằng câu lệnh:

$ sudo make install

## Build thư viện libpmemobj-cpp trên Ubuntu 20.04:

**Ta cần các gói sau trước khi cài đặt:**

* cmake >= 3.3
* libpmemobj-dev(el) >= 1.9
* compiler hổ trợ C++ 11: gcc >= 4.8.1

**Build libpmemobj-cpp:**

Trước tiên ta sẽ clone repository của libpmemobj-cpp về:

$ git clone <https://github.com/pmem/libpmemobj-cpp>

Sau đó chạy các dòng lệnh sau:

$ mkdir build

$ cd build

$ cmake ..

$ make [-DCMAKE\_INSTALL\_PREFIX=<path\_to\_installation\_dir>]

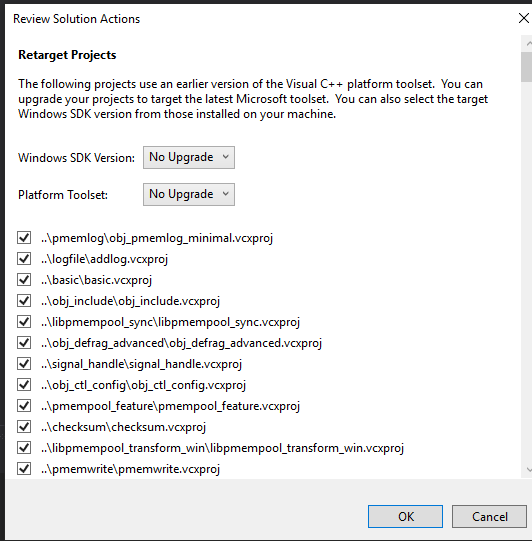
$ sudo make install

## Cài đặt thư viện PMDK trên Windows.

Hiện tại thư viện PMEM trên Windows có 2 cách để cài đặt và sử dụng

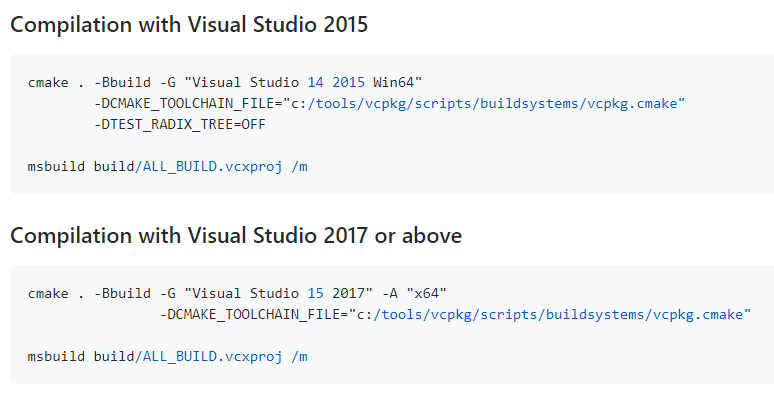
### Build trực tiếp từ project mà Intel cung cấp trên github:

* Clone source từ <https://github.com/pmem/pmdk.git>
* Mở thư mục src và mở project “PMDK.sln” và tiến hành build.
* Để build được từ project trên visual studio cần phải target solution đúng với version mà solution đang sử dụng.



* Cụ thể là SDK version cần cài đặt là 10.0.17134.0 và platform toolset là v140.
* Tiến hành build thư viện. Quá trình này sẽ diễn ra trong một khoảng thời gian. Sau khi build xong, các file thư viện sẽ nằm trong thư mục src của x64/Debug của project
* Để sử dụng những thư viện được build, ta cần phải cài đặt visual studio để có thể include các thư viện này vào project (bao gồm file header và file .lib).

Đối với thư viện libpmemobj++ ta có thể cài đặt thông qua các dòng lệnh sau:



### Cài đặt thông qua tool vcpkg:

* vcpkg là một tool mã nguồn mở từ Microsoft (https://github.com/Microsoft/vcpkg) giúp cài đặt và quản lý các thư viện trên các nền tảng khác nhau, gồm Windows, Linux, MacOS.
* Để cài đặt vcpkg ta cần chạy lần lượt hai dòng lệnh sau:

> git clone https://github.com/microsoft/vcpkg

> .\vcpkg\bootstrap-vcpkg.bat

* Để cài đặt thư viện PMDK ta cần chuyển đến thư mục vcpkg vừa clone về và chạy dòng lệnh:

> .\vcpkg integrate install

> .\vcpkg install pmdk:x64-windows

* Để cài đặt thư viện libpmemobj++ ta chạy câu lệnh:

vcpkg install pmdk:x64-windows

vcpkg integrate install

* Việc cài đặt thông qua tool vcpkg giúp cho việc cài đặt thư viện pmem đơn giản hơn, sau khi cài đặt ta không cần phải cấu hình visual studio, tuy nhiên các thư viện này chỉ hoạt động trên nền tảng x64.

# Thư viện libpmemobj++:

Thư viện libpmemobj++ là thư viện mã nguồn mở, được phát triển từ thư viện libpmemobj, dùng ngôn ngữ C++. Đây là thư viện quan trọng trong bộ thư viện PMDK, nó cung cấp các đối tượng như array, pool, vector,..và các phương thức cho các đối tượng này. Đặc biệt, libpmemobj++ cung cấp cho ta một con trỏ persistent\_ptr<> hoạt động tương tự như con trỏ bình thường.

# Chuyển đổi cấu trúc dữ liệu B + Tree sang persistent

Xác định những biến cần được persistent bao gồm các node, ExtNode, IntNode và một số biến lưu trữ lại nội dung của cây.

Trong cấu trúc dữ liệu của B+Tree có một biến m\_Path dùng để làm hàng đợi, vì vậy ta phải chuyển đổi cấu trúc của Stack này sang persistent để tiến trình đang hoạt động có thể được tiếp tục nếu xảy ra sự cố.

# Tài liệu tham khảo:

<https://github.com/pmem/pmdk>

<https://github.com/pmem/libpmemobj-cpp>