Introduction to Process Virtual Memory

Stanislas Plessia

2016

How does it look like ?

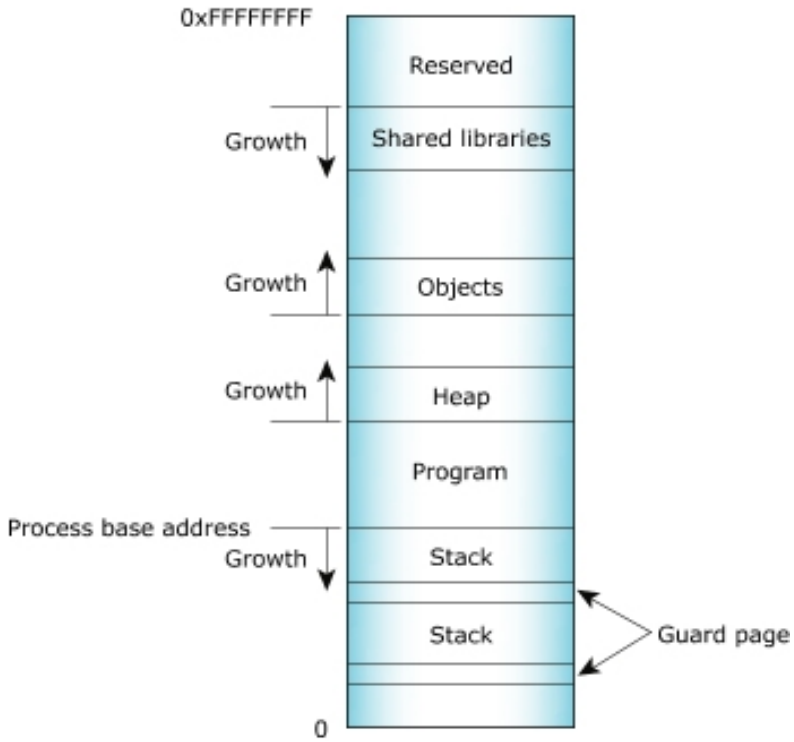

FIGURE – Process Memory

How is the physical/virtual Mapping done

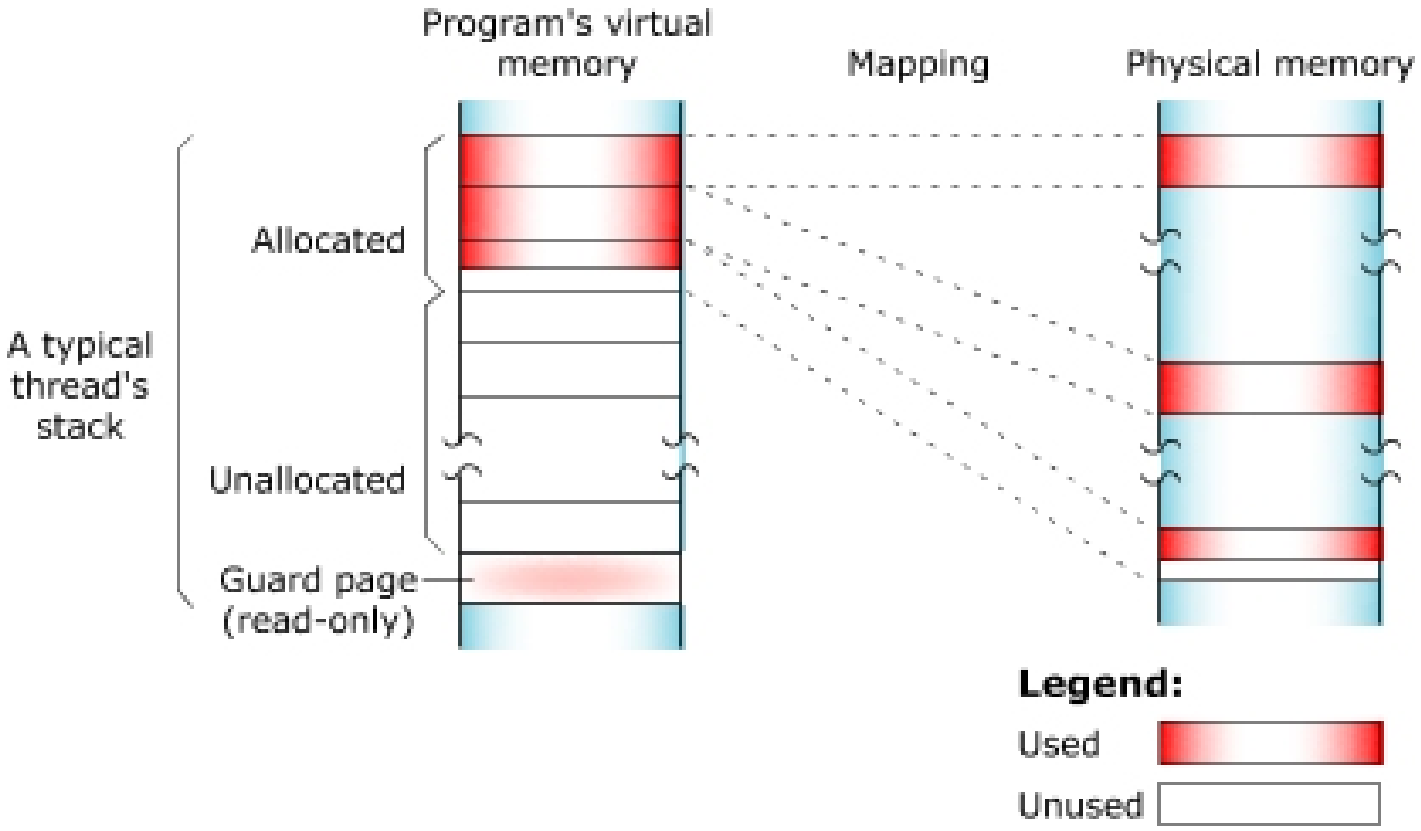

FIGURE – Physical Mapping

What it's used for

- Program : Contains the executable content of the program (code + data)
- Stack : Contains all the local variables and parameters needed for the program to run
- Libraries : Shared Libraries like libc
- Objects : To map hardware components memory into the process memory
- Heap : Dynamic memory used at runtime

What it's used for

- Program : Contains the executable content of the program (code + data)
- Stack : Contains all the local variables and parameters needed for the program to run
- Libraries : Shared Libraries like libc
- Objects : To map hardware components memory into the process memory
- Heap : Dynamic memory used at runtime

What it's used for

- Program : Contains the executable content of the program (code + data)
- Stack : Contains all the local variables and parameters needed for the program to run
- Libraries : Shared Libraries like libc
- Objects : To map hardware components memory into the process memory
- Heap : Dynamic memory used at runtime

What it's used for

- Program : Contains the executable content of the program (code + data)
- Stack : Contains all the local variables and parameters needed for the program to run
- Libraries : Shared Libraries like libc
- Objects : To map hardware components memory into the process memory
- Heap : Dynamic memory used at runtime

What it's used for

- Program : Contains the executable content of the program (code + data)
- Stack : Contains all the local variables and parameters needed for the program to run
- Libraries : Shared Libraries like libc
- Objects : To map hardware components memory into the process memory
- Heap : Dynamic memory used at runtime

What it's used for

- Program : Contains the executable content of the program (code + data)
- Stack : Contains all the local variables and parameters needed for the program to run
- Libraries : Shared Libraries like libc
- Objects : To map hardware components memory into the process memory
- Heap : Dynamic memory used at runtime

What it's used for

- Program : Contains the executable content of the program (code + data)
- Stack : Contains all the local variables and parameters needed for the program to run
- Libraries : Shared Libraries like libc
- Objects : To map hardware components memory into the process memory
- Heap : Dynamic memory used at runtime

What it's used for

- Program : Contains the executable content of the program (code + data)
- Stack : Contains all the local variables and parameters needed for the program to run
- Libraries : Shared Libraries like libc
- Objects : To map hardware components memory into the process memory
- Heap : Dynamic memory used at runtime

What it's used for

- Program : Contains the executable content of the program (code + data)
- Stack : Contains all the local variables and parameters needed for the program to run
- Libraries : Shared Libraries like libc
- Objects : To map hardware components memory into the process memory
- Heap : Dynamic memory used at runtime

What it's used for

- Program : Contains the executable content of the program (code + data)
- Stack : Contains all the local variables and parameters needed for the program to run
- Libraries : Shared Libraries like libc
- Objects : To map hardware components memory into the process memory
- Heap : Dynamic memory used at runtime

How the Program memory looks like

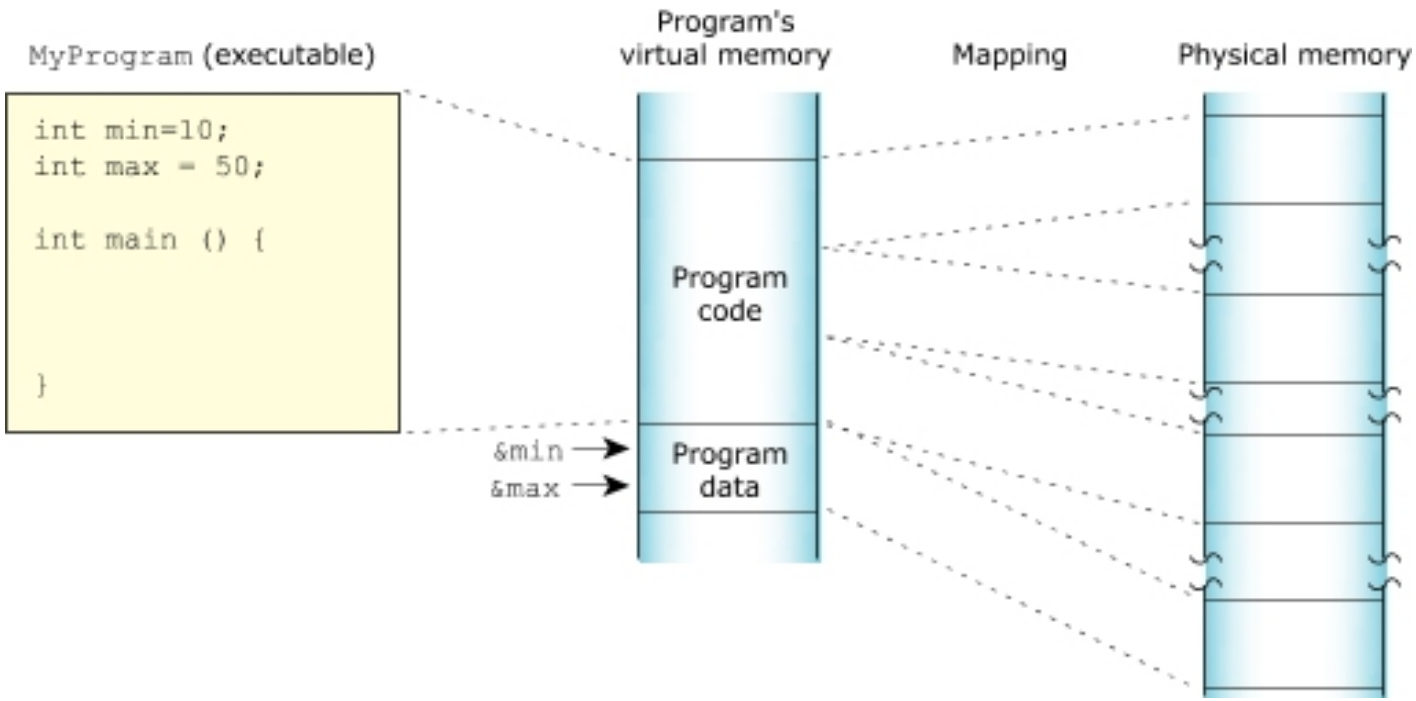

FIGURE – Program Memory

What the Stack memory looks like

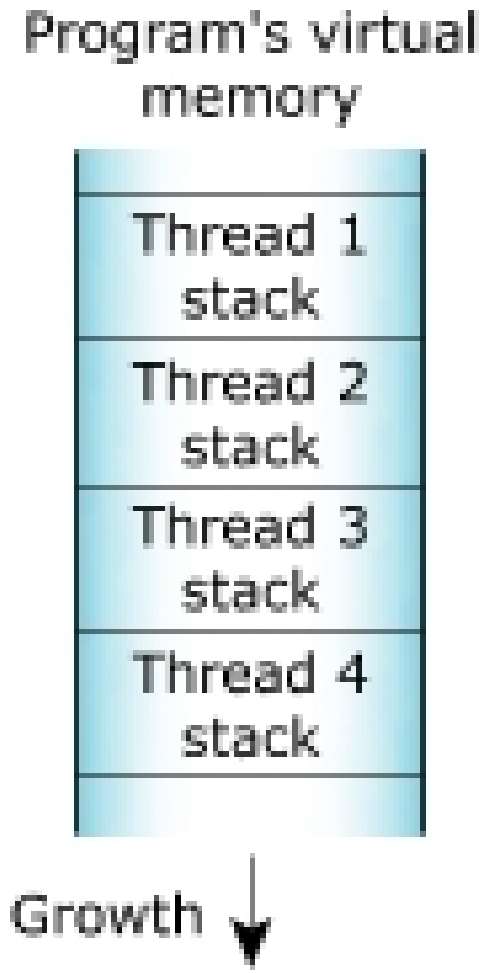

FIGURE – Stack Memory

What the Shared Libraries memory looks like

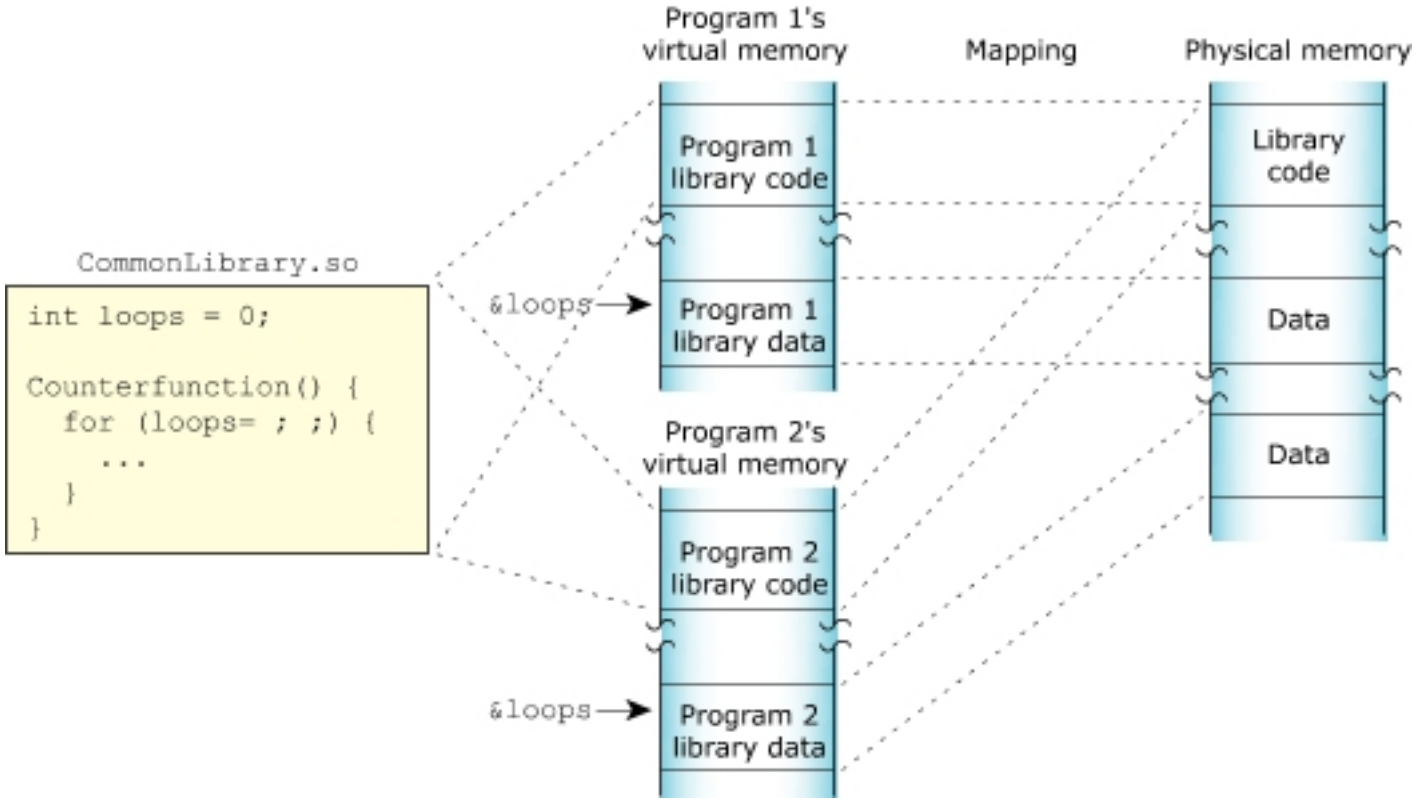

FIGURE – Libraries Memory

What the Objects memory looks like

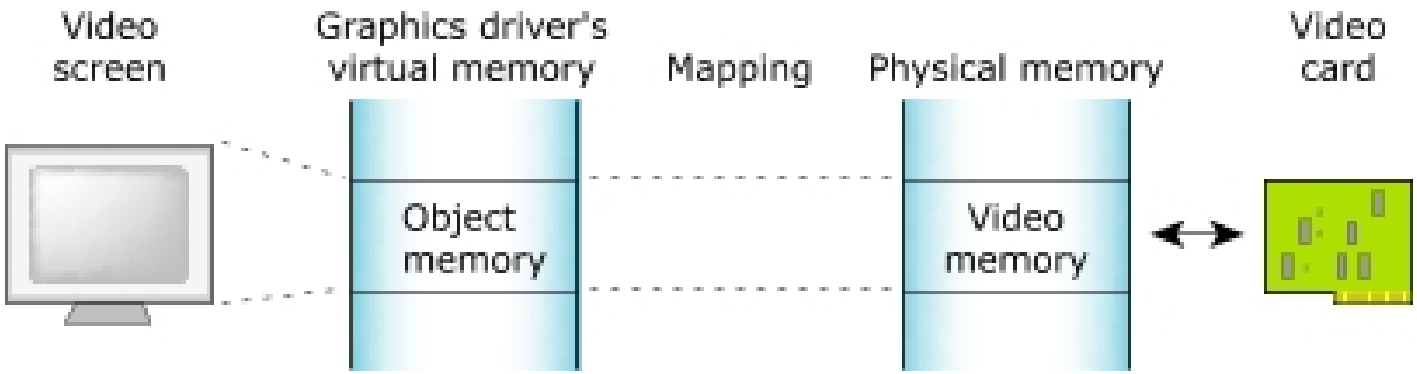

FIGURE – Objects Memory

What the Heap looks like and how it works

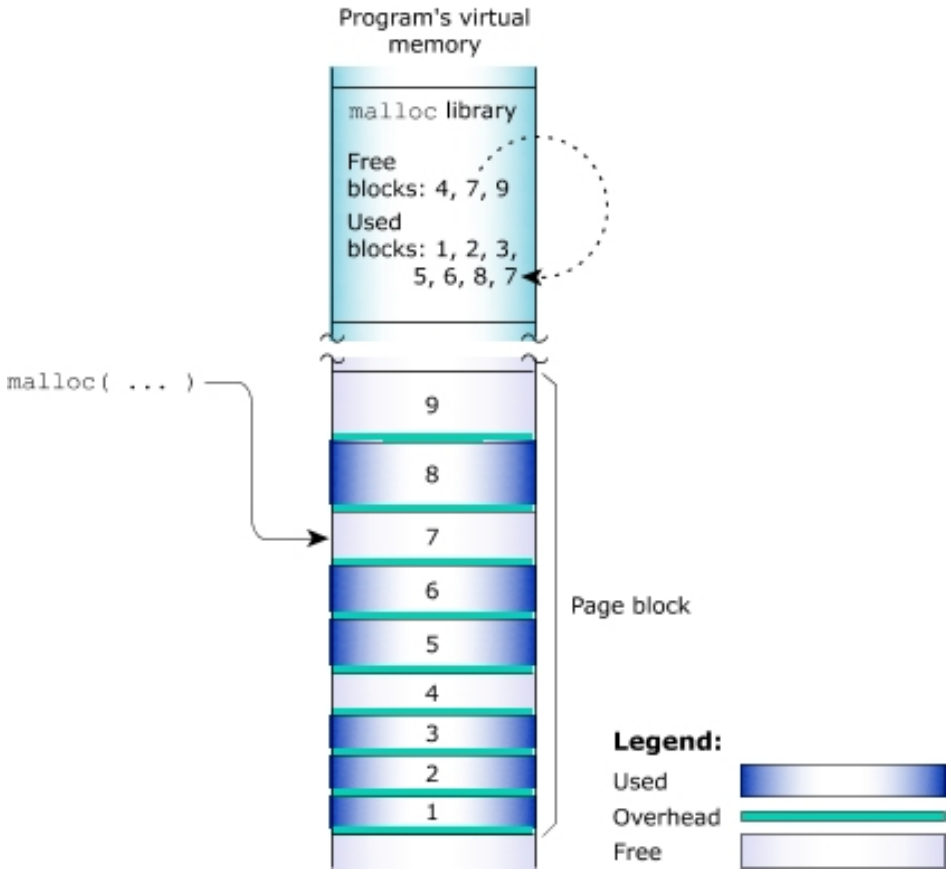

FIGURE – Heap Memory

All images where taken from QNX Website at `http://www.qnx.com`.

The User's Guide explain how memory works in QNX Neutrino IDE, but it's barelly the same everywhere.

Thank You for your attention.