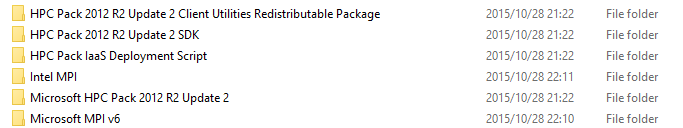
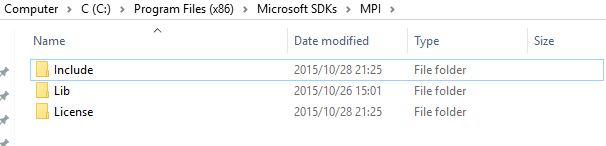
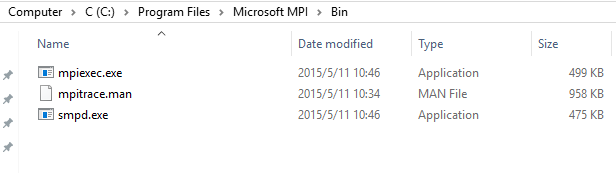
1. **MSMPI编程环境构建**
2. 安装Visual Studio 2013 update 5，作为开发MPI程序的IDE。
3. 在微软的网站上（<https://msdn.microsoft.com/en-us/library/bb524831.aspx>）下载如下的开发包（其中，Intel MPI是在Intel的官方网站上下载的，不用安装）：



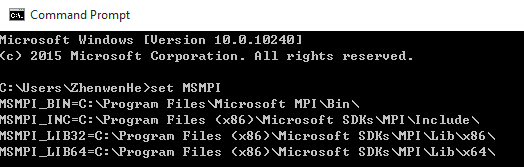
1. 安装完成后，MPI的头文件和库文件，被放在下面的位置：



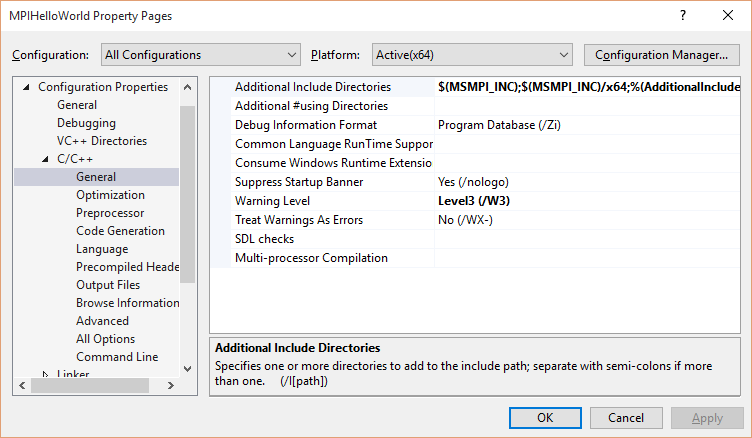
而mpiexec等执行文件，放置在下面的位置：



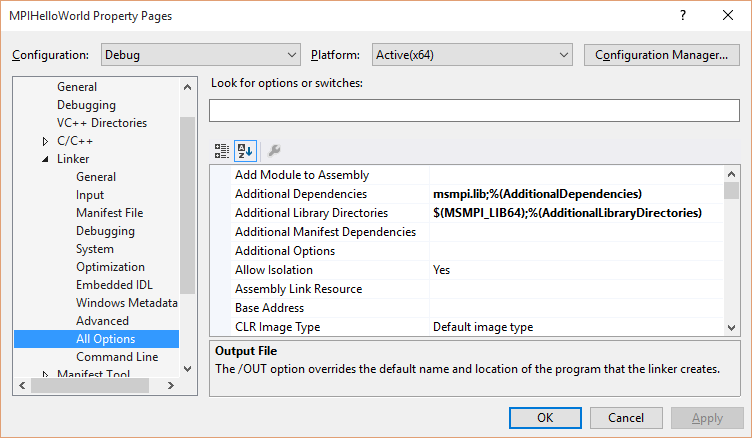
1. 运行命令set MSMPI，可以检查安装是否成功，也可以查看MSMPI开发的一些相关环境变量，如下图：



1. Run Visual Studio 2013，create Visual C++ Win32 Console Application Project. Let’s name it MPIHelloWorld and use default settings. Setup the include directories so that the compiler can find the MS-MPI header files. Note that we will be building for 64 bits so we will point the include directory to $(MSMPI\_INC);$(MSMPI\_INC)\x64. If you will be building for 32 bits please use $(MSMPI\_INC);$(MSMPI\_INC)\x86, show as follow figure:



Setup the linker lib (notice I add msmpi.lib to the additional dependencies and also add $(MSMPI\_LIB64) to the Additional Library Directories). Note that we will be building for 64 bits so we will point the Additional Library Directories to $(MSMPI\_LIB64). If you will be building for 32 bits please use $(MSMPI\_LIB32)



Code and build a simple Hello World program

#include "stdio.h"

#include "stdlib.h"

#include <string>

#include "mpi.h"

using namespace std;

int main(int argc, char\* argv[])

{

MPI\_Init(&argc, &argv);

int rank;

MPI\_Comm\_rank(MPI\_COMM\_WORLD, &rank);

if (rank == 0) {

char helloStr [] = "Hello World";

MPI\_Send(helloStr , \_countof(helloStr), MPI\_CHAR, 1, 0, MPI\_COMM\_WORLD);

}

else if (rank == 1) {

char helloStr[12];

MPI\_Recv(helloStr, \_countof(helloStr), MPI\_CHAR, 0, 0, MPI\_COMM\_WORLD, MPI\_STATUSES\_IGNORE);

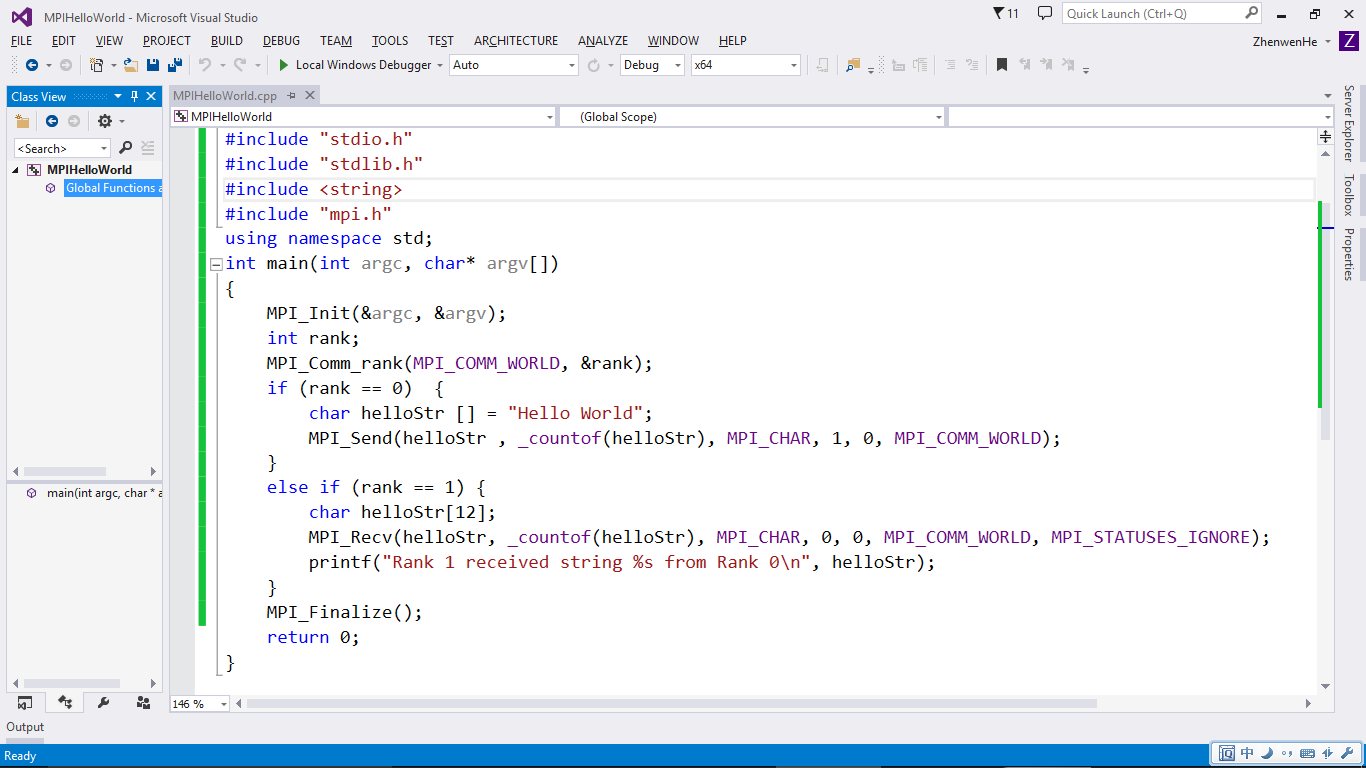
printf("Rank 1 received string %s from Rank 0\n", helloStr);

}

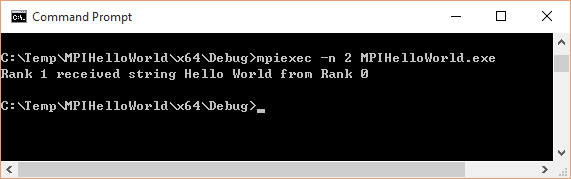
MPI\_Finalize();

return 0;

}



Test run the program on the command line



We recommend that our users use HPC Pack to run MPI across machines. However, you can still run jobs across different machines without HPC Pack, wherein you would need to install MS-MPI on all the machines and start SMPD daemon on each machine using the command smpd –d. Make sure you add the necessary firewall rules for your application. To launch the MPIHelloWorld.exe application with 2 processes, 1 on hostA and 1 on hostB, you can use the following command:

**mpiexec** -hosts 2 hostA 1 hostB 1 -wdir [\\hostA\c$\SomeDirectory MPIHelloWorld.exe](file:///\\hostA\c$\SomeDirectory MPIHelloWorld.exe)

两台主机的情况还没有测试。需要进一步研究。

Alternatively, you can use the command line to compile and link your program (replacing steps 1-6 above). Note that I have added “C:\Program Files (x86)\Microsoft Visual Studio 12.0\VC\bin\amd64” to my path environment variable so that cl.exe and link.exe are available.

To compile your program into .obj files:

cl /I"C:\Program Files (x86)\Microsoft SDKs\MPI\Include" /I"C:\Program Files (x86)\Microsoft SDKs\MPI\Include\x64" /I. /I"C:\Program Files (x86)\Microsoft SDKs\Windows\v7.1A\Include" /I"C:\Program Files (x86)\Microsoft Visual Studio 12.0\VC\include" /c MPIHelloWorld.cpp

Linking the .obj files:

link /machine:x64 /out:MpiHelloWorld.exe /dynamicbase "msmpi.lib" /libpath:"C:\Program Files (x86)\Microsoft SDKs\MPI\Lib\x64" /LIBPATH:"C:\Program Files (x86)\Microsoft Visual Studio 12.0\VC\lib\amd64" /LIBPATH:"C:\Program Files (x86)\Microsoft SDKs\Windows\v7.1A\Lib\x64" MPIHelloWorld.obj

Git环境配置 https://github.com/quantysoft/gtl

1. 在[www.github.com](http://www.github.com)上注册账号:zhenwenhe/\*\*\*\*\*\*\*\*\*\*\*,E-mail:zwhe@cug.edu.cn
2. 在Windows上安装Git\_V2.5.1\_64\_bit\_setup.exe，一路采用默认设置
3. 在打开git bash终端，键入：

**git config --global user.name "用户名"**

**git config --global user.email "邮箱地址"**

**ssh-keygen -t rsa -C "邮箱地址"**

执行前述命令后若成功则会提示在用户文件夹下生成了ssh公钥的文件。是否成功可以通过访问文件夹 .ssh 来确定，若有此文件夹则说明生成成功。在资源管理器中打开这个.ssh文件夹，在它下面会看到两个文件，选择后缀名为**.pub**的文件并用记事本打开，复制这个文件中的所有内容。

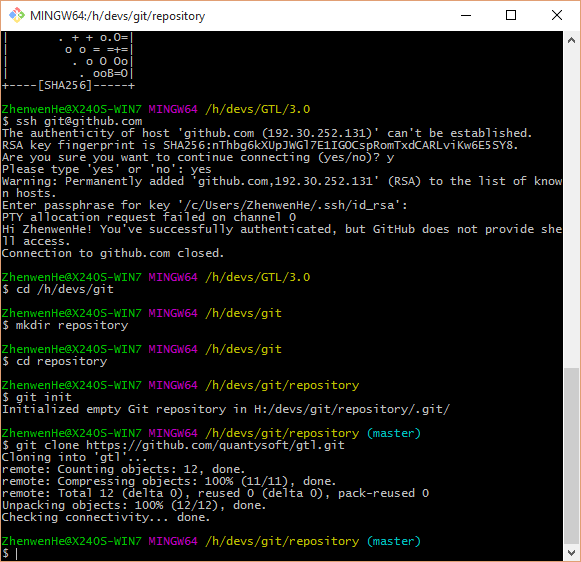
输入的passphrase for key: eaym091

打开浏览器登陆github，在自己的账户面板下找到SSH keys这一栏，打开后即会看到目前该账户下已进行过SSH认证的机器，选择Add SSH key之后，将前一步复制的内容粘贴至Key中，同时需要编辑一个Title来说明此Key认证的是哪一台机器，通常会使用计算机的名字。

1. 在git bash窗口输入：

ssh git@github.com

注意，这个命令需要按照提示输入前面设置好的passphrase for key。



1. MSMPI编程