

Spustenie MPI programu

→ Nastavenie v CodeBlocks

Pre použitie MPI a OpenMP v programe treba mať nastavený príslušný argument `-fopenmp` a `-lmpi`.
Nezabudnúť `#include <mpi.h>` a `#include <openmp.h>`

OpenMP prepínač v CodeBlocks

project -> build options -> compiler settings -> other options -> argument **-fopenmp**
project -> build options -> linker setting -> other linker option

MPI prepínač v CodeBlocks

project -> build options -> linker settings -> other linker settings -> argument **-lmpi**

→ MPI na Windows

Buď použijete CodeBlocks s priloženou MPI distribúciou, alebo nainštalujete MPICH2 (MPI distribúcia pre Windows)

1. Inštalácia MPICH2

stiahnuť a nainštalovať mpich2 z adresy <http://www.mpich.org/downloads/versions/>

- staršia inštalácia v1.4
 - <http://www.mpich.org/static/downloads/1.4.1p1/mpich2-1.4.1p1-win-x86-64.msi>
- novšiu verziu inštalovať podľa návodu - Kapitola 11 (Cygwin potrebný) - inštalovať ako admin
 - <http://www.mpich.org/static/downloads/3.0.4/mpich-3.0.4-README.txt>
- po skončení inštalácie by mal v procesoch bežať MPICH služba

2. MPI spustenie bez codeblocks

1. MPI sa nachádza v PaPP\toolchain\bin
2. Odtiaľ treba skopírovať `mpi***.dll` do adresára, kde je vykompileovaný váš program
3. Staci spustiť projekt cez `mpiexec` príkaz. Napr.:
 - a. **`mpiexec.exe -n 2 -localroot`** (cesta k `.exe` buildnutému)
4. Aby počítač vedel, kde sa `mpiexec` nachádza, tak buď:
 - a. Pustajte PaPP2/console.bat, ktorý nastavenie ciest rieši už za vás
 - b. Pridajte si do PATH cestu k MPI binárkam (...mpich\bin, alebo ...toolchain\mpich\bin)
 - c. Použite vlastný .bat súbor s cestami
5. Treba mať spustený proces manažer (`smpd`) na pozadí
 - a. pri MPICH2 by mala bežať služba na pozadí
 - b. inak treba spustiť **`smpd -d`** nachádzajúci sa pri MPI

3. MPI spustenie cez codeblocks

1. Z PaPP\toolchain\bin skopírovať `mpi***.dll` k svojmu vykompileovanému .exe súboru
2. Spustiť cez `smpd` proces
 - `smpd -d` (pocúva na pozadí)
3. Run projekt v CodeBlocks
4. Nezabudnúť nastaviť vstupné argumenty
 - a. Project / Set programs' arguments
 - b. **`-mpi mpiexec -n 2`** príp. **`-mpi mpiexec -hosts`**

4. MPI spustenie na viacerých počítačoch

Z pohľadu programu MPI spúšťa váš kus kódu s daným počtom procesov, ktoré môžu a nemusia bežať na rôznych strojoch. Ďalšie nastroje a prostriedky MPI riešia za vás sieťovú komunikáciu. A jediný rozdiel je v tom, s akými argumentami spúšťate mpi.

Na všetkých počítačoch vykompilovaný program musí mať rovnaké umiestnenie.

Opat popis pre CodeBlocks

1. V properties / Build targets, vybrať program (napr. hello), nastaviť vlastný working dir
 - o napr. na C:\mpi_zadanie a output filename na C:\mpi_zadanie***hello***.exe
2. Spustiť s prepínačom **-hosts** v rámci ktorého špecifikujete počet procesov
 - o -hosts 2 127.0.0.1 10.62.120.225
 - o radšej IP ako DNS, nemusí fungovať
 - o poradie spustenia dane poradím adries, prvý localhost, kde sa spustí master thread
3. Opat smpd podľa potreby
 - o na windowse predpokladám je namiesto toho ssh serveru ten smpd daemon
4. Pridať do firewallu výnimku / netreba vypnúť

Argument "-localonly" slúži na spustenie MPI procesov na lokálnom stroji - defaultné správanie MPICH2

Note: Different MPI Process Managers

- **mpd (mpirun)**

MPD was the traditional process manager in MPICH. The file `mpich-3.0.4/src/pm/mpd/README` has more information about interactive commands for managing the ring of MPDs. The MPD process manager is now deprecated.

- **smpd (mpiexec)**

SMPD is a process manager for interoperability between Microsoft Windows and UNIX, where some processes are running on Windows and others are running on a variant of UNIX. For more information, please see `mpich-3.0.4/src/pm/smpd/README`.