在Windows上编译PETSc

以下为在Windows系统上编译PETSc的一种途径：

编译准备：

1、PETSc的一个版本 同上述Linus下安装文件一样，从 上免费下载petsc-3.2-p6.tar.gz 。

2、MPI的实现 作为PETSc的底层结构之一，用户系统需要安装MPICH的一个版本（只使用PETSc串行功能的除外），一般从http://www.mcs.anl.gov/mpi/mpich 上可以下载其Windows下安装文件mpich2-1.3-win-ia32.msi，直接点击安装即可。安装完毕后需要将用户注册至MPICH2。

3、MS编译器，由于最终需要将PETSc编译成Windows系统下的库文件，故需要Windows的C编译器和Fortran编译器，本文安装使用Visual Studio 2005中的cl.exe编译器和IntelFortranCompiler10.1中的ifort.exe编译器，安装时注意需要将Inter的Fortran编译器集成到Visual Studio中去(Intel的Fortran能自动集成)。

4、Linux环境的模拟 为了得到Linux风格的环境，PETSc推荐使用Cygwin。Cygwin是一个用于 Windows 环境下的模拟 UNIX shell 环境，它由两个组件组成：一个 UNIX API库，它模拟 UNIX 操作系统提供的许多特性。另一部分为 Bash shell 的改写版本和许多 UNIX 实用程序，它们提供大家熟悉的 UNIX 命令行界面。前一个组件是一个 Windows 动态链接库 (DLL)。后一个组件是一组基于 Cygwin DLL 的程序，其中许多是用未经修改的 UNIX 源代码编译的。它们合在一起提供大家熟悉的 UNIX 环境。Cygwin具体操作如下：

Ⅰ 用户可以到主页http://www.cygwin.com 上下载Cygwin的安装向导setup.exe。该setup.exe文件可以用于安装软件，添加、修改或升级 Cygwin 配置的组件。

Ⅱ 运行setup.exe引导安装，由于Cygwin的完整安装版本很大，往往先将其下载至本地再安装，然后指定下载目录和下载链接。国内也有网站能提供镜像下载。

Ⅲ 下载完毕后再运行setup.exe，选择上本地安装，指定安装文件目录和安装目录后即可安装，建议完整版本的安装。整个安装过程将会持续1-2个小时。

编译阶段

1、从编译器的命令提示符中登录Cygwin 此步骤的目的是为了能在Cygwin的窗口中直接使用MS编译器。具体做法：在Start -> Programs -> Intel Software Development Tools -> Intel Fortran Compiler 10 -> Visual Fortran Build Enviornment 中打开命令提示符（cmd），登录cygwin：

C:\Documents and Settings\Administrator>$ G:\cygwin\bin\bash.exe –login

为了验证能否直接使用编译器及Cygwin组件是否安装上，可使用如下命令：

Administrator@ZFy ~

$ which cl

结果显示 /cygdrive/d/Program Files/Microsoft Visual Studio 8/VC/BIN/cl即能找到cl.exe。**对ifort，python，make，tar都运行一遍**。

2、解压文件

Administrator@ZFy ~

$ tar -zxf petsc-3.2-p6.tar

3、进入PETSc目录

Administrator@ZFy ~

$cd petsc-3.2-p6

4、配置系统 PETSc的配置提供了较多的功能，用户可以根据自己的情况选择不同的选项，比如对于已经拥有Blas和Lapack库的用户可以输入

Administrator@ ZFy ~/petsc-3.2-p6

$.config/configure.py --with-cc='win32fe cl'

--with-fc='win32fe ifort'

--with-cxx='win32fe cl'

--with-blas-lapack-dir=/home/ Blas-Lapack

--with-mpi-dir=/cygdrive/d/Program\ Files/MPICH2/

其中win32fe为MS编译器的接口，PETSc的程序包装已提供。目录/home/Administrtor/Blas-Lapack存有blas库和lapack库。/cygdrive/d /Program\ Files/MPICH2/则为MPICH2的安装目录。

对于没有这两个库的用户，可以让PETSc配置自动下载blas和lapck然后编译，用户只需提供网络连接即可，具体配置命令为：

Administrator@ ZFy ~/petsc-3.1-p8

.config/configure.py --with-cc='win32fe cl' --with-fc='win32fe ifort' --with-cxx='win32fe cl' -- download-f-blas-lapack=1 --with-mpi-dir=/cygdrive/c/Program\ Files/MPICH2/

5、编译源文件 此处需要配置两个环境变量：PETSC\_DIR与PETSC\_ARCH，其含义与Linux下安装相同。编译命令如下：

Administrator@ ZFy ~/petsc-3.2-p6

$ make PETSC\_DIR=/home/ petsc3.1

PETSC\_ARCH=arch-mswin-c-debug all

6、测试

Administrator@ ZFy ~/petsc-3.2-p6

$make test

编译结束后会在目录PETSC\_DIR/PETSC\_ARCH下生成Windows系统下PETSc库文件libpetsc.lib。

**特别的，**当添加--with-single-library=0配置选项后，PETSc将会被分别编译成各个模块，可以清楚的看到工具箱的构成：

libpetscsys.lib 基础程序 libpetscvec.lib 向量

libpetscdm.lib 索引对象 libpetscmat.lib 矩阵

libpetscksp.lib 线性求解器 libpetscsnes.lib 非线性求解器

libpetscts.lib 时间步进求解器

2.7.3 调用PETSc

编写基于PETSc的程序需要将上一节编译成的库文件以及头文件包含在程序的配置中，除此之外，并行用户还需要包含MPICH2的库文件和头文件。具体为：

Include ： PETSC\_DIR/PETSC\_ARCH/include

PETSC\_DIR/include

MPICH2-DIR/include

LIB ： PETSC\_DIR/PETSC\_ARCH/lib

MPICH2-DIR/lib

PETSc中包含的大量程序实例可以作为模板供用户学习和使用。

此外，PETSc3.1版本下载的是HYPRE 2.60b版本。

最后编译连接时出现的问题：

1. poisson.obj : error LNK2019: 无法解析的外部符号 \_PCHYPRESetType，该符号在函数 "int \_\_cdecl PoissonSolver\_MG(struct UserMG \*,struct IBMNodes \*,struct IBMInfo \*)" (?PoissonSolver\_MG@@YAHPAUUserMG@@PAUIBMNodes@@PAUIBMInfo@@@Z) 中被引用

下面的问题解决，需要用c语言版本的lapack库文件

1>HYPRE.lib(par\_relax.obj) : error LNK2019: 无法解析的外部符号 \_dgetrs\_，该符号在函数 \_hypre\_BoomerAMGRelax 中被引用

1>HYPRE.lib(schwarz.obj) : error LNK2001: 无法解析的外部符号 \_dgetrs\_

1>HYPRE.lib(par\_relax.obj) : error LNK2019: 无法解析的外部符号 \_dgetrf\_，该符号在函数 \_hypre\_BoomerAMGRelax 中被引用

1>HYPRE.lib(schwarz.obj) : error LNK2001: 无法解析的外部符号 \_dgetrf\_

1>HYPRE.lib(par\_gsmg.obj) : error LNK2019: 无法解析的外部符号 \_dgels\_，该符号在函数 \_hypre\_BoomerAMGFitVectors 中被引用

1>HYPRE.lib(ParaSails.obj) : error LNK2001: 无法解析的外部符号 \_dgels\_

1>HYPRE.lib(schwarz.obj) : error LNK2019: 无法解析的外部符号 \_dpotrs\_，该符号在函数 \_hypre\_ParMPSchwarzSolve 中被引用

1>HYPRE.lib(ParaSails.obj) : error LNK2001: 无法解析的外部符号 \_dpotrs\_

1>HYPRE.lib(schwarz.obj) : error LNK2019: 无法解析的外部符号 \_dpotrf\_，该符号在函数 \_hypre\_AMGNodalSchwarzSmoother 中被引用

1>HYPRE.lib(ParaSails.obj) : error LNK2001: 无法解析的外部符号 \_dpotrf\_

1>E:\VFS-Rivers\vsl3d\Debug\vsl3d.exe : fatal error LNK1120: 6 个无法解析的外部命令