

TFS/MapRed Version 0.1 Tutorial

1.1 功能描述

本软件包为 TFSMapRed 系统的客户开发包。

TFSMapRed 系统是搭建在 netlab cluster 上的一个 gfs/mapreduce-like 系统，用以支持海量数据处理。

要求具有 netlab cluster 帐号，才能访问搭建好的服务机群。

本发布包的库版本为 gcc 3.4.3，要求在对应版本的 cluster 节点上进行开发和任务提交。

以下以 build1 上登录的 webg 帐号为例。

1.2 获取软件包，安装

登录 netlab cluster。

软件包位置：/homeX/tfs/share/system/release/tplatform.####.tar.gz

解压缩展开，得到如下目录树：

```
[webg@build1 webg]$
cp /homeX/tfs/share/system/release/tplatform.20080815.tar.gz .
[webg@build1 webg]$ tar xzvf tplatform.20080815.tar.gz
platform
|——bin      ; 工具程序
|——include  ; 头文件
|——lib      ; 库
|——doc      ; API 手册
|   |——tfs
|   |——mapreduce
|——examples      ; 例程
|   |——tfimport    ; 把天网网页原始数据文件导入 TFS
|   |——tfsshell    ; 访问 TFS 的 shell
|   |——PageLen     ; 统计网页长度的 MapRed 程序
|   |——WordFrequency ; 统计中文词频的 MapRed 程序
|   |——PennySort    ; 并行排序的 MapRed 程序
```

1.3 例程编译与本地运行

开发环境：gcc 3.4，还需要安装 cmake 2.4.6

首先需要配置用户的环境，设置包括 PATH 和 LD_LIBRARY_PATH 环境变

量的设置，进入 tplatform 目录，如下：

```
[webg@build1tplatform] $ source ./configure
```

也可以根据需要把这两个设置，加到 ~/.bash_profile 里面。

对于某个帐号第一次使用 tfs/mapreduce cluster，就需要先在系统注册，如下：

```
[webg@build1tplatform] $ register.sh
Please hit Enter 3 times...
```

如果出现提示，按提示按三次 Enter 回车键即可。

注：当前系统没有用户安全检查，所有用户公用一个帐号访问资源，请注意避免删除其它用户的数据。

编译：进入 examples 目录，建立一个 build 目录，进入 build 目录，如下：

```
[webg@build1 examples]$ mkdir build
[webg@build1 examples]$ cd build/
[webg@build1 examples]$ cmake ..
[webg@build1 examples]$ make
```

编译成功后，有如下文件：

tfs_import	； 把天网网页原始数据文件导入 TFS
tfs_shell	； 访问 TFS 的 shell
libPageLen.so	； 统计网页长度的 MapRed 程序
libWordFrequency.so	； 统计中文词频的 MapRed 程序
libPennySort.so	； 并行排序的 MapRed 程序
ChkSort	； 校验 PennySort 排序结果
SortGen	； 生成 PennySort 排序的原始数据文件
TfsSortGen	； 在 TFS 上生成 PennySort 排序的原始数据文件

运行例程之前，首先需要得到一个 cluster 的客户端配置文件，运行 bin 目录下的 tlist.sh，如下：

```
[webg@build1 build]$ tlist.sh
Current useable clusters are:
dev
user
```

tlist.sh 命令列出当前系统可用的 cluster 名字

```
[webg@build1 build]$ tlist.sh -g dev
ok!, please check the configure file tfs_client.ini
```

tlist.sh -g 命令把指定的 cluster 的客户端配置文件复制到本地。

下面可以开始在本地运行例程。

运行 tfs_shell，如下：

```
[webg@build1 build]$ ./tfs_shell
```

```
tfs_shell:/> ls
drw-rw-rw-  1          0      0 2008-07-07 10:34 data
drw-rw-rw-  1          0      0 2008-07-11 13:10 output
tfs_shell:/> help
    ls
    pwd
    cd <tfs_dir>
    mkdir <tfs_dir>
    rm <tfs_file>
    rm -rf <tfs_dir>
    put <local_file> [<tfs_file> [<blocksize>]]
    get <tfs_file> [<local_file>]
    rename <src_file> <dest_file>
    quit
    !<system_command>
```

运行 tfs_import, 如下:

```
[webg@build1 build]$ ./tfs_import -f
    /home/webg/cwt200g/Web_Raw/Web_Raw.build1.build1.06543.00030
    -t /data/Test.dat
upload 63887 records done!
```

本地运行 PageLen, 首先得到一个本地天网网页原始数据文件 Pages.dat, 然后执行如下:

```
[webg@build1 build]$ mapred_serial ../PageLen/PageLen.ini --input=Pages.dat
INFO: 8-15 14:19:12 load jobSpec file from ../PageLen/PageLen.ini
INFO: 8-15 14:19:12 Start SetMapper on Pages.dat with 8 partitions
INFO: 8-15 14:19:26 End SetMapper return : recordCnt = 63887
INFO: 8-15 14:19:26 Start Sorting...
INFO: 8-15 14:19:27 End Sorting!
INFO: 8-15 14:19:27 Start SetReduce...
INFO: 8-15 14:19:28 End SetReducer return : recordCnt = 18299
```

输出结果是: Pages.dat.result

输入的配置文件的在 examples/PageLen/PageLen.ini

本地运行 WordFrequency, 首先需要把分词需要的词典文件复制到当前目录, 如下:

```
[webg@build1 build]$ cp ../WordFrequency/ WordFrequency/* .
[webg@build1 build]$ mapred_serial ../WordFrequency/WordFrequency.ini
--input=Pages.dat
INFO: 8-15 14:23:9 load jobSpec file from ../WordFrequency/WordFrequency.ini
INFO: 8-15 14:23:9 Start SetMapper on Pages.dat with 8 partitions
INFO: 8-15 14:26:55 End SetMapper return : recordCnt = 63887
INFO: 8-15 14:26:55 Start Sorting...
```

```
INFO: 8-15 14:28:31 End Sorting!
INFO: 8-15 14:28:31 Start SetReduce...
INFO: 8-15 14:28:52 End SetReducer return : recordCnt = 149469
```

输出结果是: Pages.dat.result

本地运行 PennySort, 首先用 SortGen 创建一个数据文件, 如下:

```
[webg@build1 build]$ ./SortGen 1000000 1M.dat
[webg@build1 build]$ mapred\_serial ../PennySort/PennySort.ini
--input=1M.dat
INFO: 8-15 14:29:50 load jobSpec file from ../PennySort/PennySort.ini
INFO: 8-15 14:29:50 Start SetMapper on 1M.dat with 8 partitions
INFO: 8-15 14:29:51 End SetMapper return : recordCnt = 1000000
INFO: 8-15 14:29:51 Start Sorting...
INFO: 8-15 14:29:55 End Sorting!
INFO: 8-15 14:29:55 Start SetReduce...
INFO: 8-15 14:29:58 End SetReducer return : recordCnt = 1000000
[webg@build1 build]$ ./ChkSort 1M.dat 1M.dat.result
ChkSort (V1.0) - Verify your PennySort result with its checksum and order.
    Author: Peng Liu, Tsinghua University, China.  pengliu@ieee.org  12/2002
Analysing file: 1M.dat
.....
Record Number: 1000000  Checksum: F34F988E
Analysing file: 1M.dat.result
.....
Record Number: 1000000  Checksum: F34F988E
Congratulations: Verification Passed!
```

1.4 在Cluster上运行

在 Cluster 上运行 MapRed 任务, 首先需要准备数据, 把待处理的数据放到 TFS 之上, 以 PageLen 为例。使用 tfs_import, 把待处理的数据传到 TFS 上, 如上 1.3 所示。

第二步需要把任务的 lib 文件和相关数据文件发布到 cluster 上去, 如下:

```
[webg@build1 build]$ mkdir PageLen
[webg@build1 build]$ cp libPageLen.so tfs_client.ini PageLen/
[webg@build1 build]$ tpush.sh dev PageLen
user <webg> start to push local job directory <PageLen> across the <dev> cluster
make job directory and distribute data across cluster...
***** dev_workers *****
----- build1-----
----- build2-----
```

```

----- build3-----
----- build4-----
----- build5-----
----- build7-----
----- build8-----
----- build9-----
----- build10-----

```

这里使用 dev 开发用的 cluster, 用户应该使用名字为 user 的 cluster。参见 tlist.sh。
然后可以用 tdir.sh 命令检查数据分发的情况, 如下:

```

[webg@build1 build]$ tdir.sh dev PageLen
webg's job directory  on dev:
***** dev_workers *****
----- build1-----
total 652
-rwxrwxr-x   1 tfs tfs 656318 Aug 15 14:42 libPageLen.so
-rw-rw-r--   1 tfs tfs   114 Aug 15 14:43 tfs_client.ini

```

第三步, 就是提交任务, 如下:

```

[webg@build1 build]$ mapred_job commit ../PageLen/PageLen.ini PageLen --detach
--input=/data/Test.dat
input_file:/data/Test.dat
output_file:/home/webg/output.dat
so_file:/home/tfs/tianwang//user/webg/PageLen/libPageLen.so
mapper_class:PageLenMapper
reducer_class:PageLenReducer
mapper_out_key_class:int
mapper_out_value_class:char*
do_local_combine:0
do_group_by_sort:1
do_concat_outputs:0
reduce_task_num:8
output_replication:1
mapper_in_key_class:
mapper_in_value_class:
mapper_record_reader_class:TianwangRawFileFeeder
reducer_out_key_class:
reducer_out_value_class:
reducer_record_writer_class:TextFileWriter<int,char*>

masterHost:masterPort as 222.29.154.21:25000
Job ID: 266807686

```

参数说明: 需要给出发布数据时用的 jobName, 并指定输入数据文件名

默认输出在 TFS 用户目录下/home/##username/下

执行中间状态，可以查询，如下：

```
[webg@build1 build]$ mapred_monitor --jobid=645668066
645668066          COMP  Detached  /data/Test.dat  /home/webg/output.dat
/home/tfs/tianwang//user/webg/PageLen/libPageLen.so  9  8  NotLocalCombine  8-15
16:3:28 8-15 16:3:38
All tasks for job 645668066 :
  MAP          IDLE      0
  MAP          INPR      0
  MAP          COMP      0
  MAP          CNCL      0
  TRN          IDLE      0
  TRN          INPR      0
  TRN          COMP      0
  TRN          CNCL      0
  RDC          IDLE      0
  RDC          INPR      0
  RDC          COMP      0
  RDC          CNCL      0
  CLN          IDLE      0
  CLN          INPR      0
  CLN          COMP      0
  CLN          CNCL      0
TOTAL Tasks: 0
```

第四步，可以查看执行日志，如下：

```
[webg@build1 build]$ tget.sh dev 645668066
```

这个命令把指定 jobId 对应的 log 数据取到本地，可以去浏览检查。