

## 如果只是想运行，那么只 **tar xvzf index.\***

on April 9, 2008 by 侯锐<hsw212 AT 163.com>

如果只是想运行，那么只需 `tar xvzf index.*` 这个包即可，不用 `tse.*` 包。因为 `index*` 包中的 `Data` 文件夹中有一个自带的天网文档集 `Tianwang.raw.2559638448`，以及对其切分，建立倒排索引等的文 `sun.iidx` 等。搭建的具体步骤：

1. `tar xvzf index.*`
2. `make`
3. 把 `linux` 的 `/var/www/html` 中的内容移出，并将 `make` 后的文件放到 `/var/www/html/yc-cgi-bin/index` 中
4. 把 `tar` 后 `index/public_html` 中的所有文件移到 `/var/www/html` 中
5. 建立 `/var/www/html/yc/TSE`，并把 `index/public_html` 中的文件发到其中
6. `vi /etc/httpd/conf/httpd.conf` 更改其中几项
  - a) `ScripAlias /cgi-bin/` “`/var/www/html/`” 改为 `ScripAlias /yc-cgi-bin/index/` “`/var/www/html/yc-cgi-bin/index/`”
  - b) `AddDefaultCharset ISO-8859-1` 改为 `AddDefaultCharset GB2312`

改后需要重启 Apache 服务器。

说明这些文件夹的名称和位置都是可改的，但是需要改动源码文件中的 `Shapshot.cpp`, `DisplayRst.cpp` 等并 `make`。我是按照闫老师源代码中的设置来建的文件夹，力求不改动源码。另，`make` 后的 `index` 文件夹也不一定要放在 `/var/www/html/` 下，但要改动 `httpd.conf` 中的 `DocumentRoot` 选项。

如果要进行重新爬取网页，应在以上的基础上添上如下步骤：

1. `tar xvzf tse*`
2. `make`
3. `nohup ./Tse -c tse_seed.pku &`

4. 爬取后会得到 10 个 Tianwang.raw.\*\*\*\*\*, 选取一个移到 tar 后的 index 文件中
5. 打开 index 中的 DocIndex.cpp, Comm.h, Snapshot.cpp 找到其中的 Tianwang.raw.2559638448 改成 Tianwang.raw.\*\*\*\*\*, 注意, \*\*\*\*的数字应为你之前移到 Index 中的那个。
6. 在 Index 中 make
7. ./DocIndex
8. 打开生成的 Doc.idx 记住最后的数字。
9. 打开 DocSegment.cpp 按源码的提示将 MAX\_DOC\_ID 的值改为此数字。
10. 再次 make
11. 然后按闫老师在 index 文件下所写的 readme.txt (为了方便大家, 我将其贴到了下面) 进行操作, 并将得到的 sun.iidx, Url.idx.sort\_uniq 放到 Data 文件夹中 (应先删除 Data 中自带的文件, 若不想则需修改 Comm.h 文件)

希望我的说明能帮助对 TSE 感兴趣的朋友在最短的时间把系统搭建起来。

版权归闫老师所有, 再次感谢闫老师的悉心指导。

学生 侯锐 Email: hsw212 AT 163.com

附录:

闫老师的 readme.txt 文档

1. The document index (Doc.idx) keeps information about each document. It is a fixed width ISAM (Index sequential access mode) index, ordered by docID. The information stored in each entry includes a pointer into the repository, a document length, a document checksum.

The url index (url.idx) is used to convert URLs into docIDs. It is a list of URL checksums with their corresponding docIDs and is sorted by checksum. In order to find the docID of a particular URL, the URL's checksum is computed and a binary search is performed on the checksums file to find its docID.

- ```
./DocIndex
  got Doc.idx, Url.idx, DocId2Url.idx
```
2. `sort Url.idx|uniq > Url.idx.sort_uniq`
  3. Segment document to terms, (with finding document according to the url)  
`./DocSegment Tianwang.raw.2559638448`  
`got Tianwang.raw.2559638448.seg`
  4. Create forward index (docid-->termid)  
`./CrtForwardIdx Tianwang.raw.2559638448.seg > moon.fidx`
  5. `# set | grep "LANG"`  
`LANG=en; export LANG;`  
`sort moon.fidx > moon.fidx.sort`
  6. Create inverted index (termid-->docid)  
`./CrtInvertedIdx moon.fidx.sort > sun.iidx`

-----  
provdng service

at <http://162.105.80.60/TSE/>

TSESearch CGI program for query  
Snapshot CGI program for page snapshot