目录

[一、sphinx的基本介绍： 1](#_Toc403917646)

[1、使用背景： 1](#_Toc403917647)

[2、基本原理： 1](#_Toc403917648)

[3、Coreseek介绍： 2](#_Toc403917649)

[二、安装： 3](#_Toc403917650)

[1、下载解压，拷贝到我们指定一个目录下面。 3](#_Toc403917651)

[2、在etc目录下面，把mysql的模板配置文件拷贝到上级目录，并修改为sphinx.conf 3](#_Toc403917652)

[3、配置数据源（对哪些数据进行建立索引） 4](#_Toc403917653)

[4、配置索引，一个数据源，要对应一个索引。 5](#_Toc403917654)

[5、配置sphinx服务器 5](#_Toc403917655)

[6、根据配置文件里面的配置，生成索引文件。 6](#_Toc403917656)

[7、要安装启动sphinx服务。 7](#_Toc403917657)

[三、php中应用： 8](#_Toc403917658)

[四、sphinx的匹配，模式： 8](#_Toc403917659)

[五、可以把匹配到的单词加样式显示。 9](#_Toc403917660)

[六、增量索引： 10](#_Toc403917661)

[七、索引的修改 13](#_Toc403917662)

# 一、sphinx的基本介绍：

## 1、使用背景：

在mysql中优化的时候，对varchar,char,text对这些数据进行查询时，如果我们使用like ‘%单词’，是无法使用到索引，如果网站的数据量比较大，会拖垮网站的速度。比如在根据电影的剧情来查找电影的名称，比如根据歌词查找歌名。

利用第三方搜索软件:

Sphinx是一个独立的**全文索引引擎**，意图为其他应用提供**高速**、**低空间**

占用、搜索结果**高相关度**的**全文搜索**功能。Sphinx可以非常容易的与

SQL数据库和脚本语言集成。内置**MySQL**和PostgreSQL数据库数据源

的支持。搜索API支持**PHP**、Python、Perl、Rudy和Java。

## 2、基本原理：

比如有一张表：

archives内容表：

id title content

1 龙门飞甲 今天很热啊，有道夏天了

2 凤凰传奇 又出信息歌曲了

3 今天天气晴朗 明天可能下雪了

4 夏天

使用原理：

（1）先对数据源建立索引。采用分词技术，形成一个索引表：

龙门 1

飞甲 1

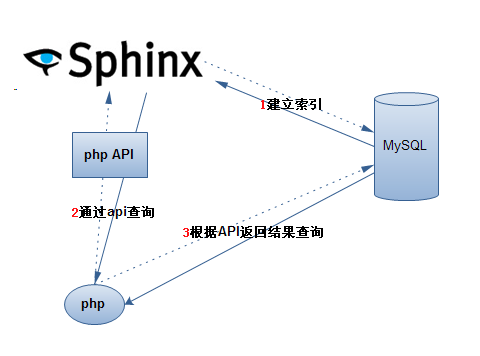
今天 1,3

夏天 1,4

（2）当查询某个单词的时候，先到sphinx建立的索引去查找、

比如查找“夏天“ ------》返回该词所在行的id

（3）mysql利用返回的id再进行从mysql 数据库里面查找。



## 3、Coreseek介绍：

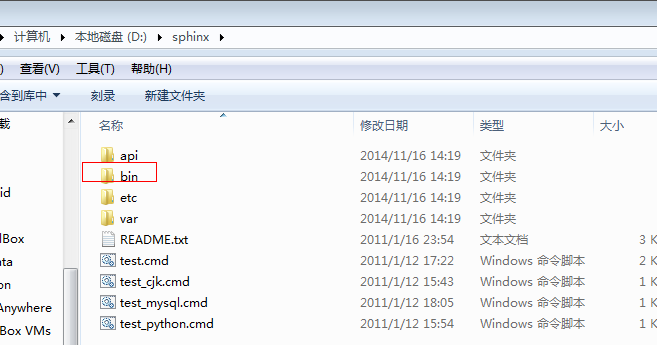
Coreseek 是一款**中文全文检索**/搜索软件,**基于Sphinx**研发并独立发布，

专攻**中文搜索**和信息处理领域，适用于行业/垂直搜索、论坛/站内搜索、

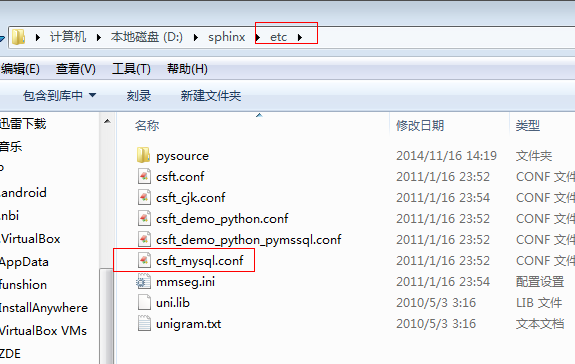
数据库搜索、文档/文献检索、信息检索、数据挖掘等应用场景

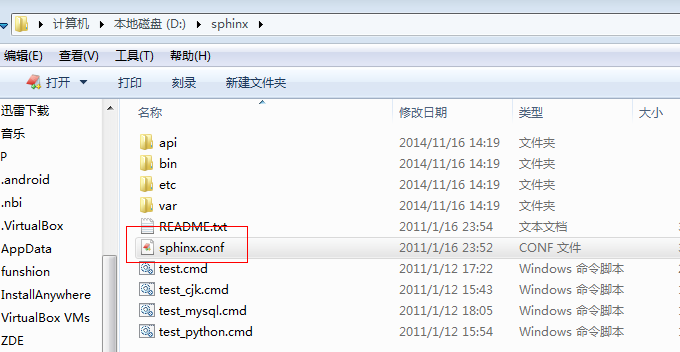
# 二、安装：

## 1、下载解压，拷贝到我们指定一个目录下面。



## 2、在etc目录下面，把mysql的模板配置文件拷贝到上级目录，并修改为sphinx.conf





## 3、配置数据源（对哪些数据进行建立索引）

打开配置文件。

建立索引源：在一个配置文件中，可以建立多个索引源的。

语法：source 索引源的名称。

source 名称{

//具体的配置

}



## 4、配置索引，一个数据源，要对应一个索引。

语法：index 索引名

index a67{

//配置项

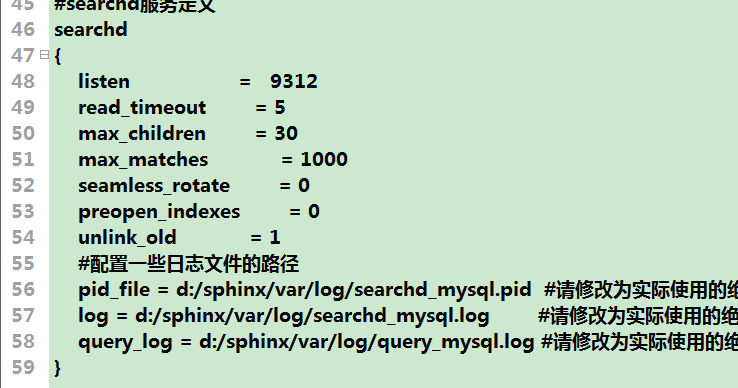
}



是生成的索引文件的名字，不是一个目录

要和数据源的名字必须一致

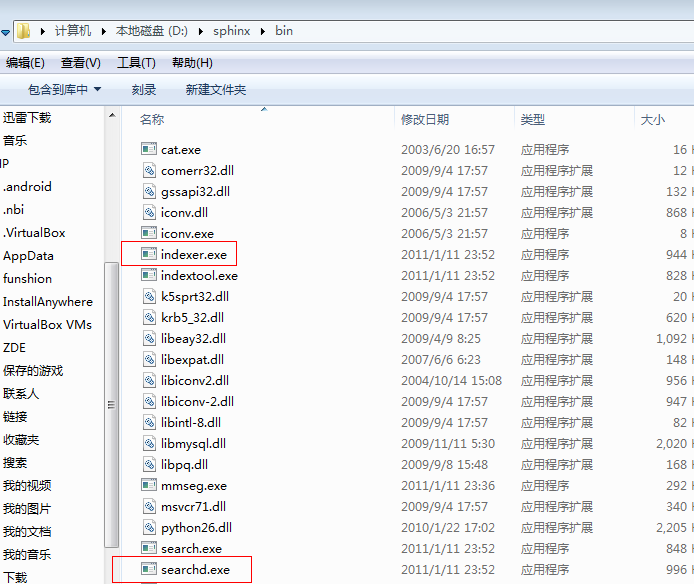
## 5、配置sphinx服务器



配置的信息是：数据源，数据源对应的索引，服务器端信息。

## 6、根据配置文件里面的配置，生成索引文件。

要以管理员的方式进入到cmd,进入 到sphinx下面的bin目录



安装该服务的命令

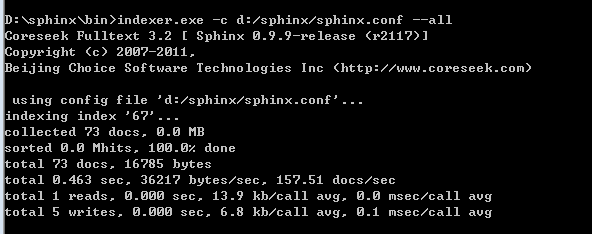
生成索引文件的命令

语法格式：

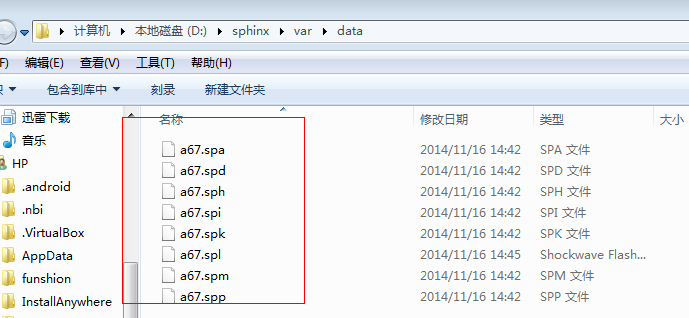
执行sphinx下的一个程序indexer.exe –c配置文件 –all | 索引的名字

--all:为配置文件中所有的索引创建索引文件

也可以使用索引的名字只为某一个索引创建索引文件。



生成的索引文件。



## 7、要安装启动sphinx服务。

语法：

searchd.exe –c 配置文件 --install

该命令对应的参数：

searchd开启服务端

searchd -c 配置文件 索引名称

服务器端默认监听 9312 端口。常用命令：

-c : 指定配置文件路径

--stop : 停止当前服务

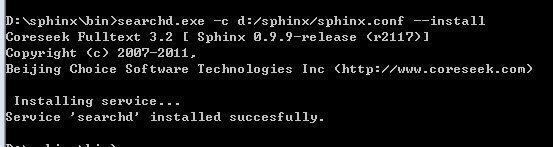
--status ： 查看当前状态

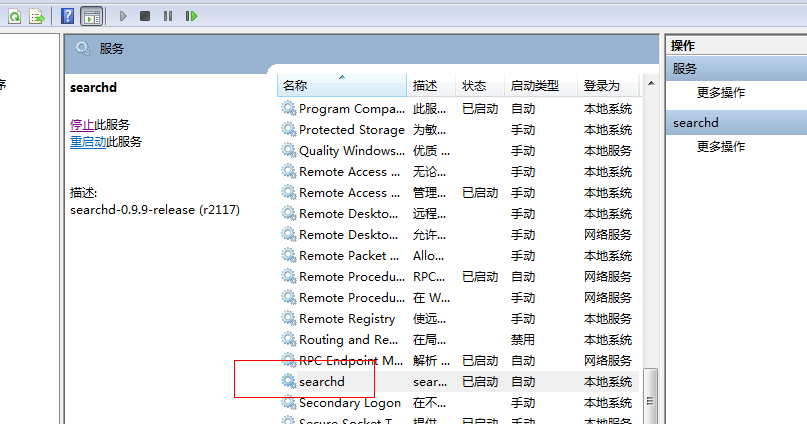
--install : 安装为 windows 服务

--delete: 删除windows服务

--port port: 监听的端口

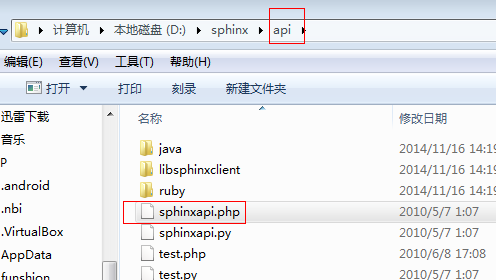
--index indexName : 只查询某个索引,默认查询所有索引

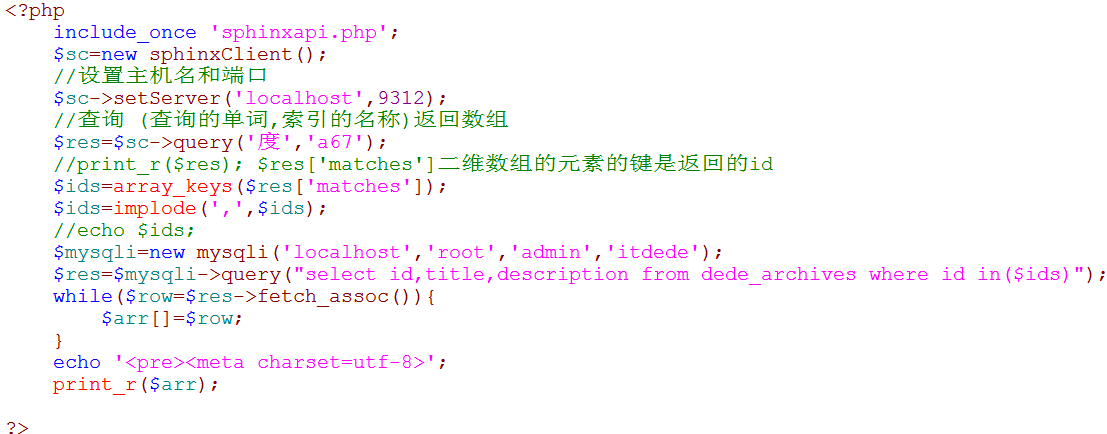




# 三、php中应用：

使用一个接口文件（sphinxapi.php）。把该文件拷到项目目录中，





相关的一些函数：

//$sc->setLimits(20,10);//设置返回结果集偏移量和数目

# 四、sphinx的匹配，模式：

$sc->setMatchMode(SPH\_MATCH\_ANY);

SphinxClient::setMatchMode — 设置全文查询的匹配模式

**（1）SPH\_MATCH\_ALL 完全匹配所有的词**

如“冬天 的 雪”，并**不会匹配** “我爱冬天”,

但**可以匹配** “我**的**朋友，爱**冬天**，和**雪**”。

因为“冬天的雪” 被分成 “冬天”，“的”，“雪”三个词，**匹配条件是同时包含**

这三个词，“我爱冬天”里只包含一个“冬天”

（2）**SPH\_MATCH\_ANY: 匹配任意一个词**

如“冬天 的 雪”，并**会匹配** “我爱冬天”。

"冬天的雪“ -》 ”冬天“ ”的“ ”雪“

因为“我爱冬天”里有一个“冬天”相匹配

（3）**SPH\_MATCH\_PHRASE: 必须匹配整个短语**

如“冬天的雪”，**不会匹配** “我**的**朋友，爱**冬天**，和**雪**”，虽然都包含同样的

需要严格匹配不再健忘，只匹配“冬天的雪”

（4）**SPH\_MATCH\_BOOLEAN 与，或，非，分组 &,or,!,()**

如：hello | world

查询“手机”，或“冬天”，：

<?php

$sc = new SphinxClient();

$res = $sc->query("手机|冬天");

（5）**SPH\_MATCH\_EXTENDED: 支持一些扩展的语法**

支持 @字段 查询

如，查询title包含 abc , content 包含 bcd的：

'@title abc @content bcd'

# 五、可以把匹配到的单词加样式显示。

使用该函数，给匹配到的内容添加样式显示：

public array **SphinxClient::buildExcerpts** ( array $docs , string $index , string $words [, array $opts ] )

有4个参数：

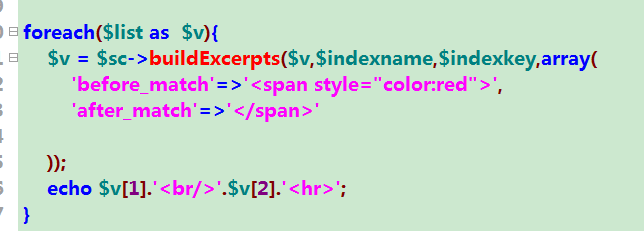
第一个参数：匹配到的内容源（数组）

第二个参数：索引的名字

第三个参数：匹配的词。

第四个参数：匹配的内容。

要注意：该函数运行完成后，**返回的是一个索引数组**



## 六、增量索引：

对与后面添加的数据，如果要重新建立索引，则非常耗费时间，我们可以只对添加的内容建立索引，建完索引后，把该索引合并到主索引里面即可。

思考：如何知道哪些数据是新添加的呢？

新建一个表，该表用于记录数据源表里面最大的id,

当创建增量索引时，只针对大于该表记录的id 的数据建立索引。

具体的步骤：

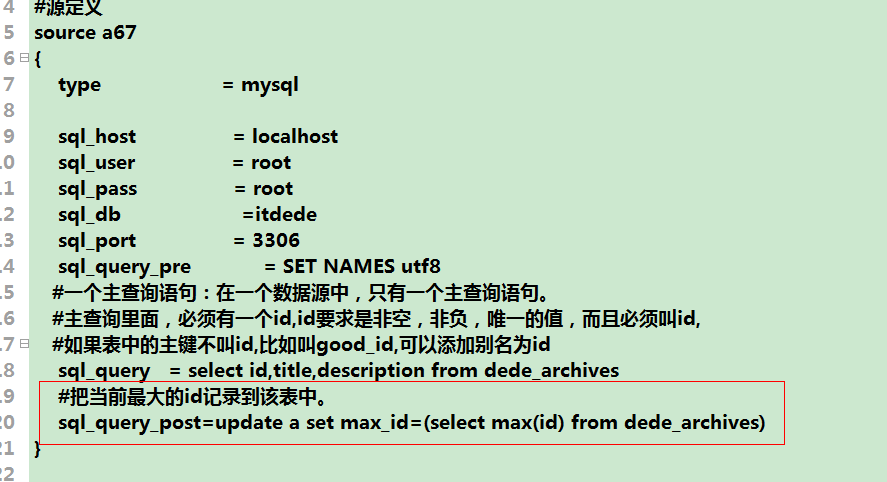
（1）创建一张表：

create table a(max\_id int);

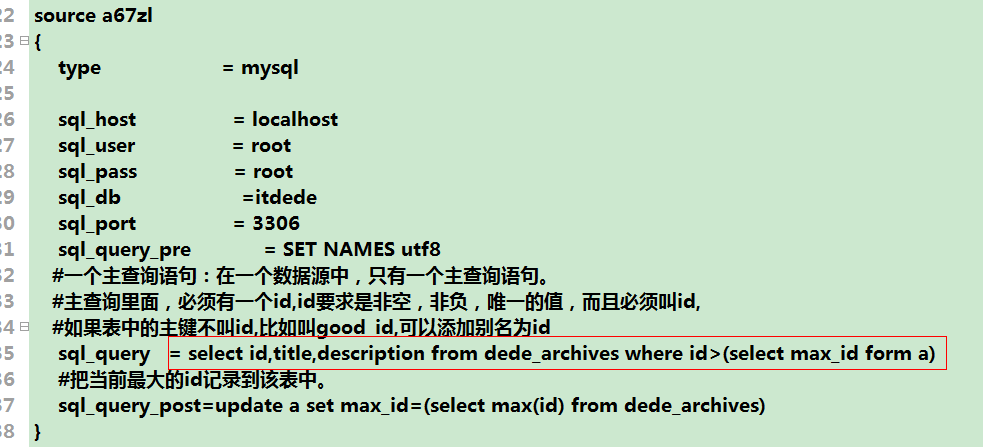
添加一条记录。

inser into a values(0)

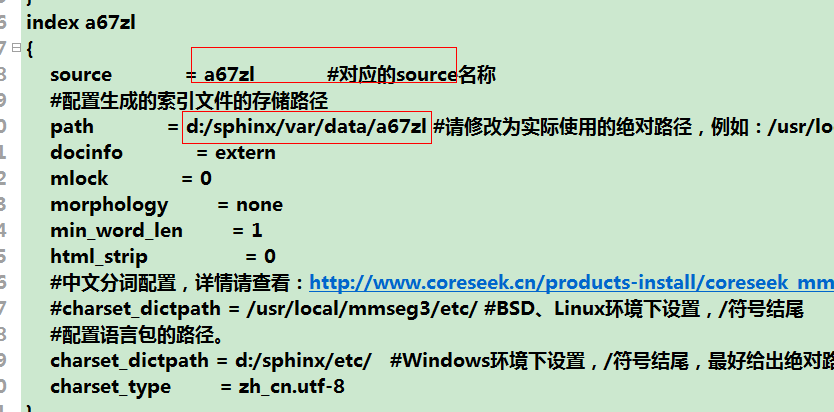
（2）修改配置文件中的主索引配置项，目的建完索引后，把最大的id存储到该表中。



（3）新建一个增量的数据源：



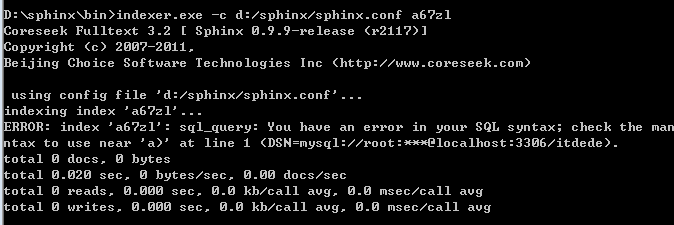
（4）新建一个增量数据源对应的索引。

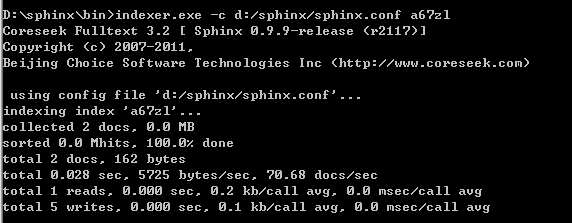


生成增量索引的名字

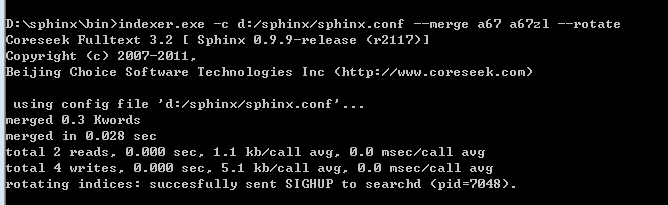
指定对应增量数据源

（5）重新建立主索引，要把最大的id值保存的a表里面





（6）要把增量索引合并到主索引上面。



<?php

//引入接口文件：

require 'sphinxapi.php';

$sc = new SphinxClient(); // 生成客户端

$sc->setServer('localhost', 9312); // 设置服务器

//$sc->query("查询的单词",'索引的名称');

//$sc->setLimits(20,10);//设置返回结果集偏移量和数目

$sc->setMatchMode(SPH\_MATCH\_ANY);

$indexname = 'a67';

$indexkey='八仙前传';

$res = $sc->query($indexkey, $indexname); // 在 mysql2 索引中查询 手机

//print\_r($res);

$ids = array\_keys($res['matches']);//返回的是查询到id值

//print\_r($ids);

$id = implode(",",$ids);

//echo $id;

$conn = mysql\_connect("localhost",'root','root');

mysql\_query("use itdede");

mysql\_query("set names utf8");

$sql="select id,title,description from dede\_archives where id in($id)";

$res = mysql\_query($sql,$conn);

$list=array();

while($row=mysql\_fetch\_assoc($res)){

$list[]=$row;

}

foreach($list as $v){

$v = $sc->buildExcerpts($v,$indexname,$indexkey,array(

'before\_match'=>'<span style="color:red">',

'after\_match'=>'</span>'

));

echo $v[1].'<br/>'.$v[2].'<hr>';

}

?>

# 七、索引的修改

在修改时，也是需要一张表，该表用于记录修改数据的id,再根据表里面记录的id,进行建立索引，建完索引后，再合并到主索引里面。