

ZipSite

ZipSite Readme

ZipSite

关于ZipSite

ZipSite是这样个项目，它允许你通过打包压缩好的zip文件，快速的建立起一个基于Google App Engine（以下简称GAE）的互联网站点。

由于GAE对于其上传的文件有着种种限制，例如：

上传的文件最大不得大于1M

上传的文件数最多不得多于1000个

考虑到HTML文档大多可以通过zip方式压缩后获得更小的体积（部分情况下文件大小可以下降到压缩前的25%以下）；同时，zip本身支持目录结构化，对于层级性较好的HTML站点则可以通过对其目录分别打包甚至整站打包的方式大大减少文件数量。

版本更新历史

时间	版本号	更新内容
2010/11	Ver 0.1	原型版本： 完成程序原型 支持通过GAE自带的memcache和数据存储区域做双重缓冲,减少响应时间 导入了插件系统框架，支持灵活的插件修改原始文件内容
2011/03	Ver 0.2	SEO和大文件支持强化： 支持对于原始大于1M以上的文件进行缓冲 增加GAE taskqueue任务解决缓冲无法一次清空的问题 支持通过访问记录，自动生成符合标准的 /sitemap.txt 和 /sitemap.xml链接
2011/03	Ver 1.0	正式发布版本： 修复之前存在的bug 更加简单化的目录结构 导入配置文件website.cfg 取消了不够成熟的插件系统
2011/07	Ver 1.02	性能优化版本： 合理化目录和代码结构 部分算法优化 支持基于HTTP 304状态的本地缓存设置

实现原理

1. ZipSite内核实现基于GAE所支持的Python中的ZipFile类包，该类包旨在允许Python直接访问经过压缩后的zip文件。对于此类库的详细文档您可以访问Python官网的详细介绍：<http://www.python.org/doc/current/library/zipfile.html>

2. 考虑到GAE在对文件大小和数量限制的同时，还对整个网站沙盒的CPU Time和 API访问进行了一系列的限制，为在此基础上尽可能减少这些限制对于网站访问的限制，同时尽可能降低页面响应时间，Zipsite通过GAE API中的memcache和google.appengine.ext.db类包对解压后的数据做两次缓冲。
3. 对于首次访问，系统中会主动在Http头中添加“Cache-Control”（缓存时间），“Last-Modified”（文件最后修改时间）和“Expires”（缓存到期时间）标签，要求客户端浏览器对收到的页面进行缓存。在缓存期内，同一个页面浏览器会在发起请求的同时会主动添加“If-Modified-Since”（最后修改时间），系统会将此时间于服务器端文件生成时间进行比对。如果两者时间相同，则视为缓存仍然有效，返回Http 304状态，要求直接显示本地缓存，否则将直接下载服务器端文件并给予Http 200状态。
4. 默认情况下可以通过访问<http://YOUR-SITE/deletecache>清空DB Store和Memcache中所有缓存数据。

由于插件目前尚不成熟，原测试版本中的内容将不会出现在正式版本中。该Feature的导入时间有待评估。

使用方法

GAE准备

注册GAE账户

请访问[GAE网站首页](#)根据相应的提示完成GAE的注册，期间可能会需要真实手机号码验证。

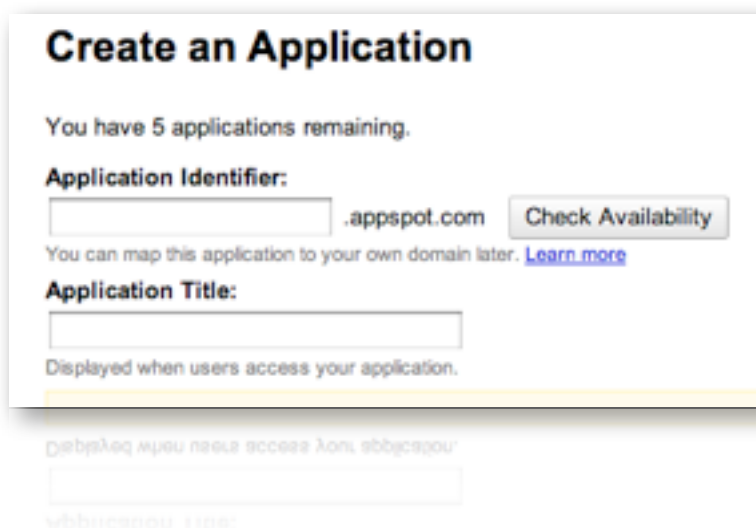
设置GAE应用名称

完成注册后，点击Create Application按钮创建一个属于自己的GAE应用

在如右图的界面中，Application Identifier 填写一个简单贴切的唯一标识，之后您可以通过<http://Your-ID.appspot.com>的方式访问您的网站。

Application Title为的应用标识，可以随意起一个简单的标识名称。

点击确认创建该应用。



创建SDK环境

访问<http://www.python.org/getit/> 页面，或者直接使用*nix的软件管理器获取的Python。由于Python3.0版本与之前版本有较大差异，在这里本人建议下载Python2系列的版本（最新的为2.7）。

访问<http://code.google.com/intl/zh-CN/appengine/downloads.html> 获取符合您操作系统的GAE SDK。

程序设置

站点配置文件的设置

请使用文本编辑器打开zipsite根目录下的app.yaml文件。

位于文件头部的

application参数，为您注册的GAE项目名称

version参数，为您目前站点的版本，如果您不确定站点的版本号，建议您不要修改此项目

文件中其余的项目不建议您做任何修改，除非您清楚的明白您操作的目的。

网站参数的设置

由于默认的设置已经可以满足大多数用户的需求，所以正常情况下，我们不建议您在不清楚具体参数含义的情况下贸然修改配置。

请使用文本文件编辑器打开zipsite根目录下的website.cfg文件。

请根据注释内容修改相应的参数设置。

网站文件准备

将您所有的网站文件保存在根目录的Website目录下（默认配置）

对每个目录分别进行zip压缩，压缩前后要保持目录结构前后一致

- 原目录结构为Website/a/b/c.html 则制作一个Website/目录下压缩包，文件名为a.zip。包内文件结构则必须以a/为根目录，即zip文件中以一个a/的文件夹为起点
- 由于系统支持结构化的操作，您同样可以在一个子目录中进行同样的操作，Website/a/b/c.html同样可以是在Website/a/下面建立一个名为b.zip的文件，文件以b/为起点。

为方便大家操作，减少不必要的差错，在这里本人推荐您可以通命令行操作：

For Windows，请先通过默认路径安装[7-zip](#)：

```
for /d %x in (*) do "c:\Program Files\7-Zip\7z.exe" a -tzip %x.zip %x
```

For Unix，需要安装zip包：

```
find ./ -type d -depth* 1 | xargs -l {} zip -r -9 {}.zip {}
```

*部分Linux系统不支持-depth参数，取而代之的是-maxdepth参数

上传站点

本文以Mac版本的GAE SDK为例，其余版本可供参考

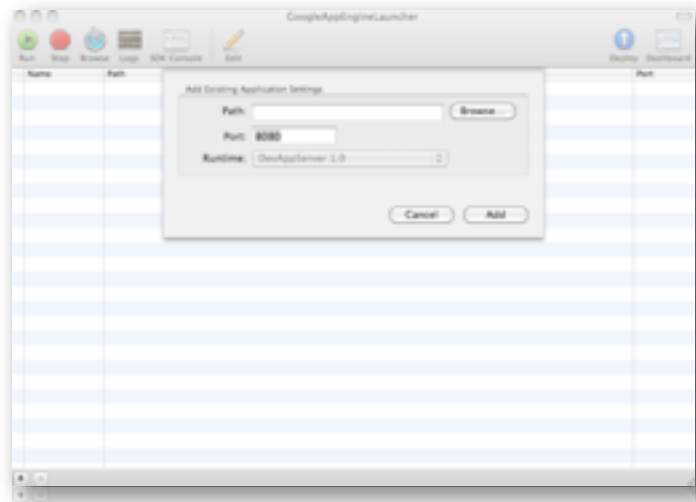
打开google程序主界面，File->add exist applicati，将path项指定到您的zipsite目录，点击Deploy按钮上传，期间会提示您输入账户和密码，请输入您之前注册的完整GAE账户（邮件地址）和密码。

需要注意的是整个操作需要完整的互联网连接，如果您尝试多次后仍然无法上传成功，建议您检查或者更换您的互联网连接。

等待上传结束后，可以通过<http://Your-ID.appspot.com> 的链接访问您的站点

清空缓存

对于更新原有的网站文件，请确保每次上传结束后，以管理员身份访问<http://Your-ID.appspot.com/deletecache>。系统将会清空之前所有的数据缓存，并对新上传的文件进行缓存。



附录：FAQ

我无法访问任何appspot中托管的网站，怎么办？

可以通过修改hosts的方法简单实现对于GAE的访问，操作如下：

对于windows用户使用记事本等文本编辑器打开c:\windows\system32\drivers\etc\hosts文件（注意写保护状态）。在文件末尾添加一行：203.208.39.104 android-sdk.appspot.com（请替换成你的项目网址）

对于Unix/Linux/Mac用户，请直接在控制台下以root身份执行 `#echo "203.208.39.104 android-sdk.appspot.com" >> /etc/hosts`

我的网站访问量较大，Zipsite项目采用了哪些技术以减少我支付给Google的费用？

1. Request方面：对不起，我们不可能采取任何限制手段禁止用户访问您的网站 :-)。
2. CPU时间方面：对于压缩包内的文件，我们只会进行1次解压操作，解压后的文件将保存在DB Store内，除非您手工清除DB Store，在此之前不会再次解压。
3. DB Store方面：对于缓存的文件，相对于45M/day的访问限额，1M以下的文件，我们确保Zipsite将首先访问相对廉价(192M/day)的Memcache服务，并可以手工进行参数优化。1M以上的文件，由于技术限制目前尚未实现此机制，我们承诺将会尽快提供支持。
4. 带宽方面：从Ver1.02之后，我们提供了304服务器响应的机制，对于在一定时间内（默认为7天）再次访问的用户，客户端将直接调用在客户端中缓存的数据，而不会再次发起下载请求。当然，前提是用户没有在此之前进行“清空浏览器缓存”操作。

Zipsite是否支持使用GAE的HTTPS特性访问？

是的，Zipsite完全支持GAE的HTTPS访问！

请用文本编辑器打开app.yaml文件，将默认第10行中的“secure: never”属性前加上“#”注释掉或者改为“secure: optional”，重新上传至GAE空间即可通HTTP方式以及更为安全的HTTPS加密方式同时访问。

同样的，如果修改为 secure: always则意味着该网站仅能使用HTTPS方式访问，而无法使用HTTP。

我可以使用Zipsite的部分或者全部的代码吗？

我们很乐意您这么做！

Zipsite项目遵循较为宽松的BSD协议，您可以自由的使用该项目中的任何资源（包括本文在内）。但在您使用的同时必须遵守以下3点：

- 1.如果再发布的产品中包含源代码，则在源代码中必须带有原来代码中的BSD协议
- 2.如果再发布的只是二进制类库/软件，则需要在类库/软件的文档和版权声明中包含原来代码中的BSD协议。
- 3.不可以用Zipsite的作者/机构名字和Zipsite做市场推广。

如果您直接使用Zipsite项目进行以盈利为目的的商业活动，请注明“该网站使用Zipsite项目驱动”并请增加链接地址至<http://code.google.com/p/zipsite>