

Maple_feng

python哲学思想：“如果看起来像鸭子，听起来像鸭子，就可以把它当作鸭子”。因此python对象的接口力求简单而且统一，类似其他语言中面向接口编程思想。

博客园 首页 新随笔 联系 订阅 管理

随笔 - 184 文章 - 83 评论 - 46

python高并发的解决方案

一.cdn加速

简单说就是把静态资源放到别人服务器上

全称:Content Delivery Network或Content Ddistribute Network，即内容分发网络

基本思路:

尽可能避开互联网上有可能影响数据传输速度和稳定性的瓶颈和环节，使内容传输的更快、更稳定。通过在网络各处放置节点服务器所构成的在现有的互联网基础之上的一层智能虚拟网络，CDN系统能够实时地根据网络流量和各节点的连接、负载状况以及到用户的距离和响应时间等综合信息将用户的请求重新导向离用户最近的服务节点上。

目的:

解决因分布、带宽、服务器性能带来的访问延迟问题，适用于站点加速、点播、直播等场景。使用户可就近取得所需内容，解决 Internet网络拥挤的状况，提高用户访问网站的响应速度和成功率。

控制时延无疑是现代信息科技的重要指标，CDN的意图就是尽可能的减少资源在转发、传输、链路抖动等情况下顺利保障信息的连贯性。

CDN就是扮演者护航者和加速者的角色，更快准狠的触发信息和触达每一个用户，带来更为极致的使用体验。

二.精灵图

什么是精灵图：

css精灵,是一种网页图片应用处理技术。主要是指将网页中需要的零星的小图片集成到一个大的图片中

应用的原因：

- 1.减少对浏览器的请求次数，避免网页的延迟
- 2.方便小图标的统一管理

精灵图的制作：

- 1.软件：ps
- 2.制作方法：

新建透明图层

添加参考线将画布划分，将小图标放入划分好的格子中

三.后台数据库使用mysql+Redis

mysql是持久化存储，存放在磁盘里面，检索的话，会涉及到一定的IO，为了解决这个瓶颈，于是出现了缓存，比如现在常用的 redis。首先，用户访问缓存，如果未命中，就去访问mysql，之后将mysql中的数据复制到缓存中。

redis是缓存，并且是驻留在内存中运行的，这大大提升了高数据量web访问的访问速度。redis提供了大量的数据结构，比如string、list、set、hashset、sorted set这些，之后用户的访问就能直接从Redis的内存中去取数据了,那内存的读取速度远远大于硬盘

四.数据库的优化

- 1.sql的优化
- 2.索引的优化
- 3.分库分表
- 4.读写分离

公告

23:16:11
阿里云免费注册，免费试用，领取优惠券

昵称： Maple_feng
园龄： 2年2个月
粉丝： 26
关注： 4
+加关注

<	2020年9月						>
日	一	二	三	四	五	六	
30	31	1	2	3	4	5	
6	7	8	9	10	11	12	
13	14	15	16	17	18	19	
20	21	22	23	24	25	26	
27	28	29	30	1	2	3	
4	5	6	7	8	9	10	

搜索

找找看

谷歌搜索

常用链接

我的随笔
我的评论
我的参与
最新评论
我的标签

积分与排名

积分 - 157374
排名 - 4230

随笔分类

celery(1)
consul(1)
Django(15)
docker(7)
ElasticSearch(2)
excel(6)

五.使用缓存

例如python-django-restframework自带的缓存

六.集群化的部署

例如:django+uwsgi+nginx

详细部署内容请点击这里:<https://www.cnblogs.com/alex3714/p/6538374.html>

七.分布式+异步

celery:就是一个分布式异步的解决方案

这里详细的celery解释:<https://www.cnblogs.com/alex3714/p/6351797.html>

八.语言层面

例如:在一些并发量更多的地方,该用golang语言编写

好文要顶

关注我

收藏该文

Maple_feng
关注 - 4
粉丝 - 26
[+加关注](#)

« 上一篇: [django基于restframework的CBV封装](#)
» 下一篇: [django缓存机制](#)

posted @ 2019-02-26 19:27 Maple_feng 阅读(4276) 评论(1) 编辑 收藏

评论列表

#1楼 2019-10-24 16:14 金色旭光

回复 引用

这些解决办法似乎和python语言没多少关系呀

支持(0) 反对(0)

刷新评论 刷新页面 返回顶部

发表评论

编辑 预览

B

支持 Markdown

提交评论 退出 订阅评论

[Ctrl+Enter快捷键提交]

- flask(4)
- git(1)
- go(25)
- kakfa(1)
- kibana(2)
- Linux(8)
- MongoDB(1)
- mysql(3)
- numpy(1)
- oracle(1)
- pandas(4)
- python(68)
- RabbitMQ(2)
- Redis(6)
- splunk(1)
- supervisor(2)
- 爬虫(6)
- 前端(9)
- 算法与数据结构(6)
- 邮件(3)
- 正则(1)
- 支付宝支付(1)
- 知识图谱(1)

随笔档案

- 2020年9月(2)
- 2020年8月(1)
- 2020年7月(1)
- 2020年6月(2)
- 2020年5月(2)
- 2020年4月(4)
- 2020年3月(10)
- 2020年2月(6)
- 2020年1月(9)
- 2019年12月(9)
- 2019年11月(5)
- 2019年10月(6)
- 2019年9月(8)
- 2019年8月(11)
- 2019年7月(13)
- 2019年6月(21)
- 2019年5月(8)
- 2019年4月(16)
- 2019年3月(17)
- 2019年2月(19)
- 2019年1月(3)
- 2018年12月(1)
- 2018年11月(2)
- 2018年10月(4)
- 2018年9月(4)

最新评论

- 1. Re:爬虫库requests @Maple_feng 谢谢~... --带好吃的
- 2. Re:爬虫库requests @带好吃的 你可以去看源码，找到这个类RequestsCookieJar，这个类生成的对象就是cookies.那么这个类下面有很多方法，就包括get_dict()... --Maple_feng
- 3. Re:爬虫库requests get_dict()是从哪来的函数呢? --带好吃的
- 4. Re:docker快速安装rabbitmq 感谢分享

--溢性循环

5. Re:python制作动态排序图
运行报错..

--平芮宇

阅读排行榜

1. anaconda的安装教程和使用方法(20323)
2. docker快速安装rabbitmq(17940)
3. RBAC权限管理设计(8807)
4. vue 自定义指令input表单的数据验证(7833)
5. beego框架返回json数据(6582)
6. golang中使用gorm连接mysql操作(5809)
7. golang与vscode的安装与配置(5480)
8. python使用consul进行服务注册和发现(5193)
9. docker快速安装kibana(4725)
10. python高并发的解决方案(4276)

评论排行榜

1. python实现钉钉文件上传发送，钉钉机器人接收信息(17)
2. python基于redis实现分布式锁(5)
3. bs4修改html文件和保存(3)
4. golang模块viper读取配置文件(2)
5. golang中使用gorm连接mysql操作(2)

推荐排行榜

1. docker快速安装rabbitmq(4)
2. 爬虫四大金刚：requests，selenium，BeautifulSoup，Scrapy(2)
3. scrapy+pyppeteer指定搜索动态爬取头条(1)
4. python制作动态排序图(1)
5. django简单实现短url(1)

目录代码