互联网创业服务器运维工具集

米聊服务端开源工具选择实践

陈臻（54chen）@velocity 2011

开源

创业公司　事多枪少　怎么办？

米聊团队
创立于2010/11
<3人

米聊服务器端所使用的开源技术/项目：

Maven
Nexus
Rose
Nginx
Resin
Quercus
Zookeeper
Hadoop
Riak
……

如何hold住他们？

选择原则：

1）大公司在用的；

2）大公司没在用代码级别搞得定的。

细节是魔鬼

三个代表：

1）解决什么问题？

2）如何解决？

3）url在哪里？

一个一个来看

Nginx挡在前面，解决了平滑上线、快速接入的问题

细节：

静态过期设置

Upstream fallback设置

Error log定期检查

离用户最近的地方

静态过期设置

```
 location ~ .*\.(gif|jpg|jpeg|png|bmp|swf)$
  {
    expires      15d;
  }
```

upstream fallback设置

```
upstream backend_openapi.miliao.com {
   server   192.168.1.99:8089 weight=1 max_fails=2 fail_timeout=10s; #连续失败2次  暂时停止使用10s
   server   192.168.1.100:8089 weight=1 max_fails=2 fail_timeout=10s;
}
```

细节

[error] 4597#0: *512 readv() failed (104: Connection reset by peer) while reading upstream

上游在过程中干掉了链接，一般情况是后端的代码有问题。

Errlog里的常见问题

http://wiki.nginx.org/
http://wiki.nginx.org/Resources
http://wiki.nginx.org/HttpProxyModule
http://code.google.com/p/nginxsrp/wiki/NginxSRPProtocol
http://www.evanmiller.org/nginx-modules-guide-advanced.html

相关的url

它帮忙解决了我们的代码跑在哪里

CAUCHO

CAUCHO

Resin

CAUCHO公司的产品
application server
对servlet和JSP提供了良好的支持
resin自身采用JAVA语言开发

不选tomcat的原因

CAUCHO

OOM参数

MAT协助分析问题

<jvm-arg>-XX:+HeapDumpOnOutOfMemoryError</jvm-arg>

MAT=Memory Analyzer Tool

Resin注意事项

CAUCHO

Quercus

CAUCHO公司的产品
纯Java开发的一个PHP5引擎
基于开源授权协议GPL发布
性能超群
可以简单使用java扩展函数

不选php-fpm的原因

CAUCHO

Mysql连接编码问题

com.caucho.quercus.lib.db.JdbcConnectionResource: 94 348
com.caucho.quercus.lib.db.Mysqli:229
修改上述文件的**hard code**位置为**utf8**

地址与引用的问题

避免**php**里超过**3**层的引用
$obj->some->some->dosome()

Resin已经内置了**quercus,**修改代码注意替换

Quercus的问题

CAUCHO

http://www.caucho.com/

quercus记录：php和java的混合型项目建立手记
http://www.54chen.com/php-tech/quercus-notes-php-java-mixed-projects.html

相关url

ROSE

Rose：来自人人网的充分使用spring的框架

ROSE

简单DAO
Sharding(需要加工)
Restful规范
Portal
pipe:类似facebook big pipe的东西，内部支持多线程运行。

还在更加方便使用中。。。

ROS

Portal/pipe

简单例子，经得起历史的考验

ROSE

http://code.google.com/p/paoding-rose/wiki/Rose_SJ

相关url

由facebook带来的thrift解决了内网高效消息传递的问题

跨语言

facebook

.thrifty文件定义接口自动生成代码
多语言（**php**不能作服务端）
接口版本兼容最佳实践

facebook

Thrift简单例子

```
#!/usr/local/bin/thrift --gen java
namespace java com.chen
service Hello{
        i32 hello()
}

thrift --gen java chen.thrift
```

facebook

相关url

http://thrift.apache.org/

thrift最佳实践:版本与兼容
http://www.54chen.com/_linux_/thrift-versioning-compatibility.html

http://www.54chen.com/_linux_/thrift-quick-start.html

facebook

由facebook带来的scribe解决了大量日志收集的问题

Scribe:分布式日志系统
依靠Thrift传输日志

facebook

```
          'client'                          'central'
-----------------------------     ---------------------
| Port 1464                  |    | Port 1463         |
|        -----------------   |    | -----------------  |
|     -> | scribe server |--|--->| | scribe server | |
|        -----------------   |    | -----------------  |
|                |           |    |    |          |    |
|            temp file       |    |    |    temp file |
|----------------------------     |--------------------
                                     |
                                   |--------------------
                                    | /tmp/scribetest/ |
                                    --------------------
```

Scribe与thrift版本的问题

Thrift升级快 已经0.7.0?
Scribe慢

良好搭配：thrift 0.5.0

facebook

Scribe与log4j使用举例

cd facebook-scribe-2ee14d3/if/

thrift -r -l ../../thrift-0.5.0/contrib/（这里需要指到你的thrift的源码目录） -gen java scribe.thrift

产生java客户端

继承org.apache.log4j.AsyncAppendert 和 org.apache.log4j.WriterAppender完成异步打log的过程。

```
TSocket sock = new TSocket(new Socket(scribeHost, scribePort));
        transport = new TFramedTransport(sock);
        TBinaryProtocol protocol = new TBinaryProtocol(transport,
            false, false);
        client = new scribe.Client(protocol, protocol);

  String message = String.format("%s %s", hostname, layout
            .format(event));
        LogEntry entry = new LogEntry(scribeCategory, message);
        logEntries.add(entry);
        client.Log(logEntries);
```

facebook

相关url

https://github.com/facebook/scribe

分布式日志系统scribe使用手记
http://www.54chen.com/java-ee/log-server-scribe-helper.html

facebook

Scribe收集的日志怎么分析？

二者解决了大量日志和千变万化的分析挖掘需求

Scribe->file->crontab->hive->loadfile

Scribe日志写入hdfs与hive的协作还有些bug

Metadata保留在DB中，需要自行修改

```
update DBS set DB_LOCATION_URI=REPLACE(DB_LOCATION_URI,'old host','new host');
update SDS set LOCATION=REPLACE(LOCATION,'old host','new host');
```

hadoop hive安装手记
http://www.54chen.com/_linux_/hive-hadoop-how-to-install.html

用hadoop hive协同scribe log用户行为分析方案
http://www.54chen.com/java-ee/hadoop-hive-scribe-log.html

相关url

Maven

解决了项目依赖问题，使新人入手成本低

Maven

Jdk7下各种诡异问题

遇到的问题

解决了仓库可视化，傻瓜化

必须整体作战

Sonatype
Transforming Software Development

Maven

m2eclipse

Nexus的问题

Nexus里的snapshot为什么总是旧的

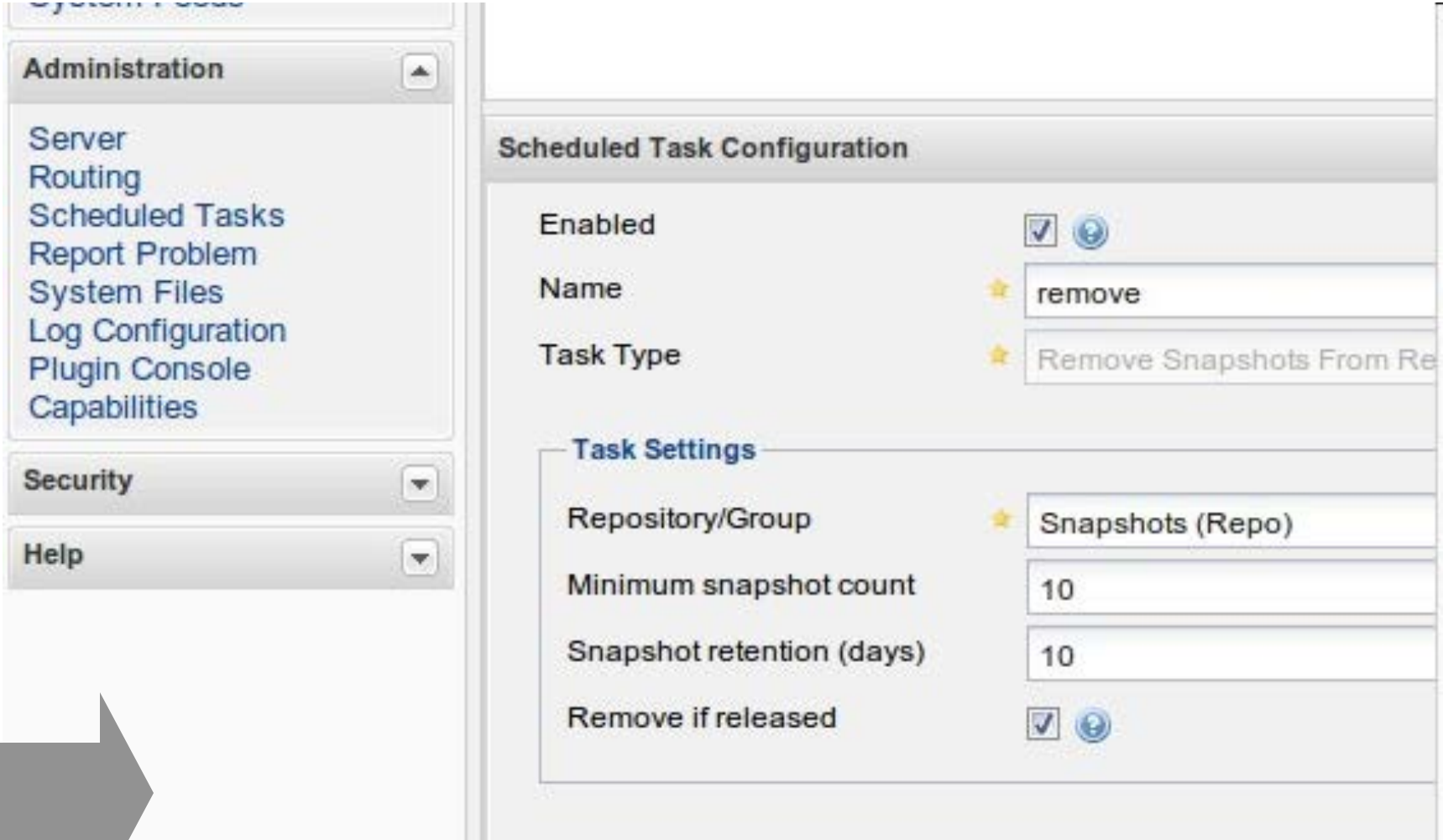
单个目录下文件过多

打开**nexus**的自动清理

Nexus的问题

Nexus里的snapshot为什么总是旧的

login->scheduled tasks->add



相关url

maven项目代码质量检测工具sonar使用手记
http://www.54chen.com/java-ee/open-source-quality-management-platform-sonar-notes.html

http://m2eclipse.sonatype.org/

http://maven.apache.org/

http://nexus.sonatype.org/

Hudson

持续构建工具

解决了自动编译发布snapshot包到maven库

sonar

项目代码质量检测工具

解决了静态代码质量检查，提升代码质量

Maven+hudson+sonar

自动编译打包质量评估

团队成员有空时就看看一sonar

Check-out Strategy: Use 'svn update' as much as possible

Use 'svn update' whenever possible, making the build faster. But this causes the

Repository browser: (Auto)

Build Triggers

- [x] Build whenever a SNAPSHOT dependency is built
- [ ] Build after other projects are built
- [x] Build periodically

Schedule: 0 00 * * *

- [ ] Poll SCM

Build

Root POM: pom.xml

Goals and options: -U clean package deploy sonar:sonar

具体设置

build中的goals and options
-U clean package deploy sonar:sonar 代码检测到sonar中

Maven

具体设置

需要在hudson所在的机器设置.m2/setttings.xml

```
<profile>
        <id>sonar</id>
        <activation>
           <activeByDefault>true</activeByDefault>
        </activation>
        <properties>
           <!-- EXAMPLE FOR MYSQL -->
           <sonar.jdbc.url>
             jdbc:mysql://maven.n.miliao.com:3306/sonar?
useUnicode=true&amp;characterEncoding=utf8
           </sonar.jdbc.url>
           <sonar.jdbc.driverClassName>com.mysql.jdbc.Driver</
sonar.jdbc.driverClassName>
           <sonar.jdbc.username>sonar</sonar.jdbc.username>
           <sonar.jdbc.password>sonar</sonar.jdbc.password>

           <!-- SERVER ON A REMOTE HOST -->
           <sonar.host.url>http://maven.n.miliao.com:8888</sonar.host.url>
        </properties>
     </profile>
```

具体设置

maven项目代码质量检测工具sonar使用手记
http://www.54chen.com/java-ee/open-source-quality-management-platform-sonar-notes.html

持续构建工具hudson使用手记
http://www.54chen.com/java-ee/ci-tool-hudson-usage.html

Maven

相关url

basho（相扑）是一家美国的技术公司，专营数据存储和管理软件，11年6月30日获得了750万美元的融资。basho将riak开源，通过收取riak的维护和管理界面软件的使用费用来赚钱，和resin的公司类似。

Riak:
Erlang开发
代码量少
标准的dynamo实现

Riak解决了kv存储的一些需求

basho

现在已经是1.0版本
优先使用leveldb
线上使用0.14.2基本0维护

Riak 0.14.2 java client存在一点bug

```
private DomainBucket<History> getDomainBucket(final IRiakClient client, String
bucketName) throws RiakRetryFailedException {
    final Bucket b = client.fetchBucket(bucketName).execute();

    return DomainBucket.builder(b, History.class).mutationProducer(new
MutationProducer<History>() {
        public Mutation<History> produce(History o) {
            return new HistoryMerger(o);
        }
    }).returnBody(true).withResolver(new
MergeHistoryResolver()).retrier(DefaultRetrier.attempts(3)).build();
  }
```

这是一个使用**java client**的例子，使用了**pb**协议，里面有：一个叫**History**的实体；不同的**bucket**保存不一样的数据，保证不冲 突；**MergeHistoryResolver**为冲突时解决办法； **HistoryMerger**为保存时对现存数据的处理办法（客户端做**append**操作）。官方文档大多是**http json**的，性能不高。

Riak java client example

http://www.54chen.com/tag/riak
http://basho.com/resources/downloads/

相关url

解决线上进程自动重启远程启停

Supervisor
hang on to your processes

Supervisord是用**Python**实现的一款非常实用的进程管理工具，在批量服务化管理时特别有效。

非**deamon**进程

它适应的进程

Supervisor
hang on to your processes

[program:xxxservice]
command=启动命令
directory=工作目录

最简配置

Supervisor status

REFRESH | RESTART ALL | STOP ALL

State	Description	N
running	pid 6758, uptime 2 days, 7:28:39	ap
running	pid 7191, uptime 31 days, 7:45:20	gl
running	pid 10002, uptime 3 days, 0:22:49	gr
running	pid 29950, uptime 1 day, 20:19:09	m
running	pid 32264, uptime 9 days, 7:32:11	m
running	pid 5397, uptime 2 days, 8:01:00	no
running	pid 19369, uptime 1 day, 0:29:49	oa
running	pid 31211, uptime 8:38:32	w
running	pid 2243, uptime 8:28:18	w

Supervisor
hang on to your processes

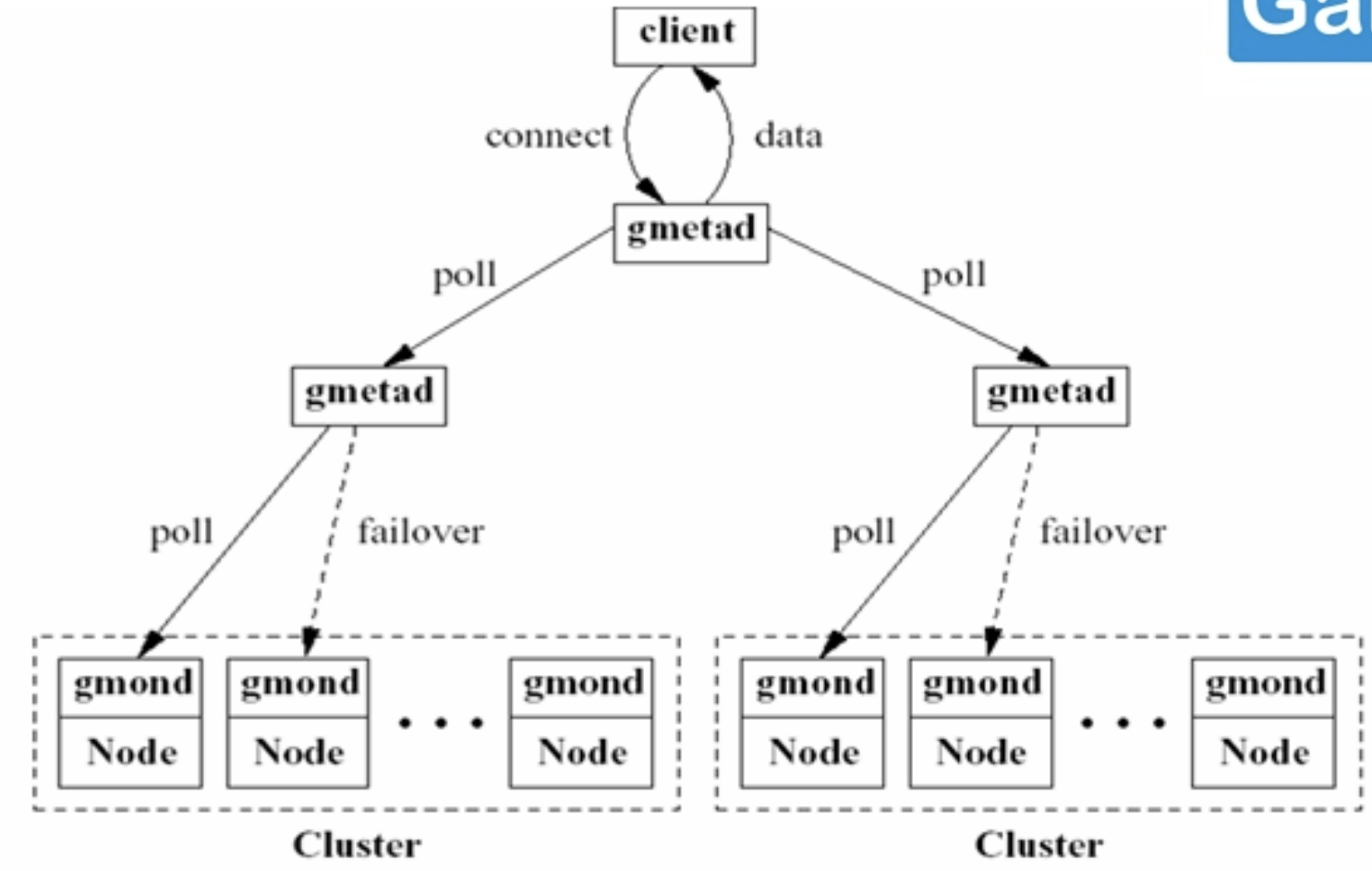
Web化管理进程

Supervisor
hang on to your processes

用**supervisord**管理杂乱的服务
http://www.54chen.com/java-ee/supervisord-manage-service.html

相关**url**

解决监控的问题



ganglia的理想汇报过程

多播可能存在风暴
拖死过华三交换机

ganglia的问题

http://ganglia.sourceforge.net

相关url

以上是过去一年的回顾，

我们正在成长中...

谢谢!

http://54chen.com

加入小米:czhttp@gmail.com