
toc: true title: 《从0到1学习Flink》—— 你上传的 jar 包藏到哪里去了? date: 2019-03-13 tags:

- Flink
 - 大数据
 - 流式计算
-

前言

写这篇文章其实也是知识星球里面的一个小伙伴问了这样一个问题：

通过 flink UI 仪表盘提交的 jar 是存储在哪个目录下？

这个问题其实我自己也有问过，但是自己因为自己的问题没有啥压力也就没深入去思考，现在可是知识星球的付费小伙伴问的，所以自然要逼着自己去深入然后才能给出正确的答案。

跟着我一起来看看我的探寻步骤吧！小小的 jar 竟然还敢和我捉迷藏？

查看配置文件


首先想到的是这个肯定可以在配置文件中有所设置的地方的：

```
# zookeeper.sasl.login-context-name: Client
zhisheng@zhisheng > /usr/local/blink-1.5.1/conf cat flink-conf.yaml | grep jar
X zhisheng@zhisheng > /usr/local/blink-1.5.1/conf cat flink-conf.yaml | grep upload
# Directory to upload completed jobs to. Add this directory to the list of
zhisheng@zhisheng > /usr/local/blink-1.5.1/conf open flink-conf.yaml
zhisheng@zhisheng > /usr/local/blink-1.5.1/conf
```


谷歌大法好

虽然有个是 upload 的，但是并不是我们想要的目录！于是，只好动用我的“谷歌大法好”。

flink jar upload




Search Google for 'flink jar upload'
Search for 'flink jar upload' with default fallback



Search Amazon for 'flink jar upload'

⌘2



Search Wikipedia for 'flink jar upload'

zhisheng ⌘3

找到了一条，点进去看 Issue 如下：

APACHE SOFTWARE FOUNDATION
http://www.apache.org/

Dashboards

Projects

Issues

Flink

Issues

Reports

Components

Flink / FLINK-4308

Allow uploaded jar directory to be configurable

Details

Type: Improvement

Priority: Minor

Affects Version/s: None

Component/s: Runtime / Web Frontend

Labels: None

Status: CLOSED

Resolution: Fixed

Fix Version/s: 1.2.0

People

Assignee:

Reporter:

Votes:

Watchers:

Description

I notice sometimes it's preferable to have uploaded jars to be put into a configurable directory location instead only have it at runtime. In this case we can pre-load the directory with jars in a docker image and allows us to leverage the jobmanager restful interface to start/kill jobs.

WebRuntimeMonitor.java

String uploadDirName = "flink-web-upload-" + UUID.randomUUID();
this.uploadDir = new File(getBaseDir(config), uploadDirName);

Issue Links

links to

GitHub Pull Request #2335

Activity

All

Comments

Work Log

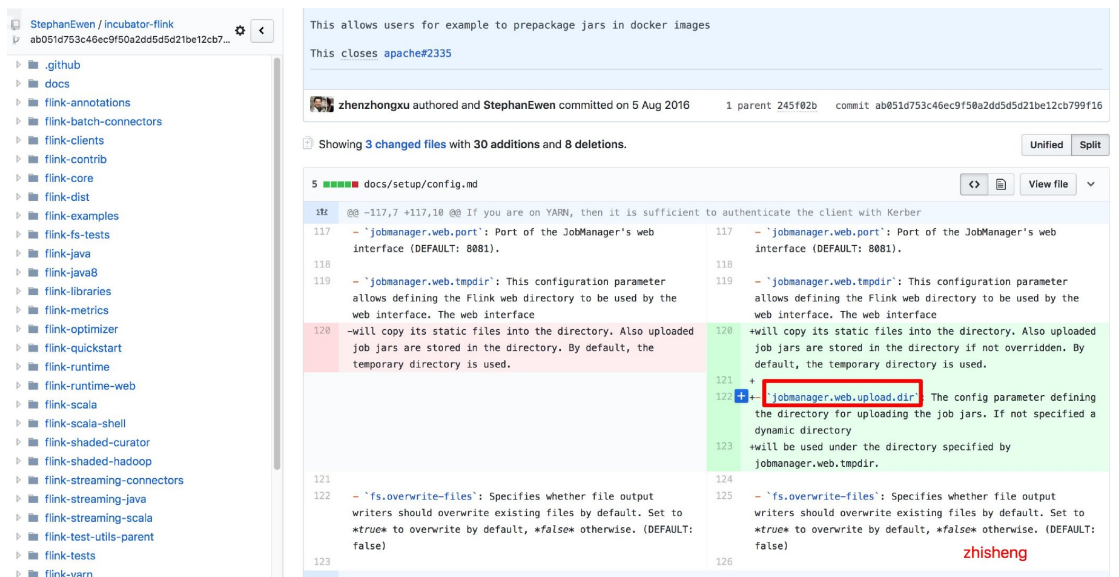
History

Activity

Transitions

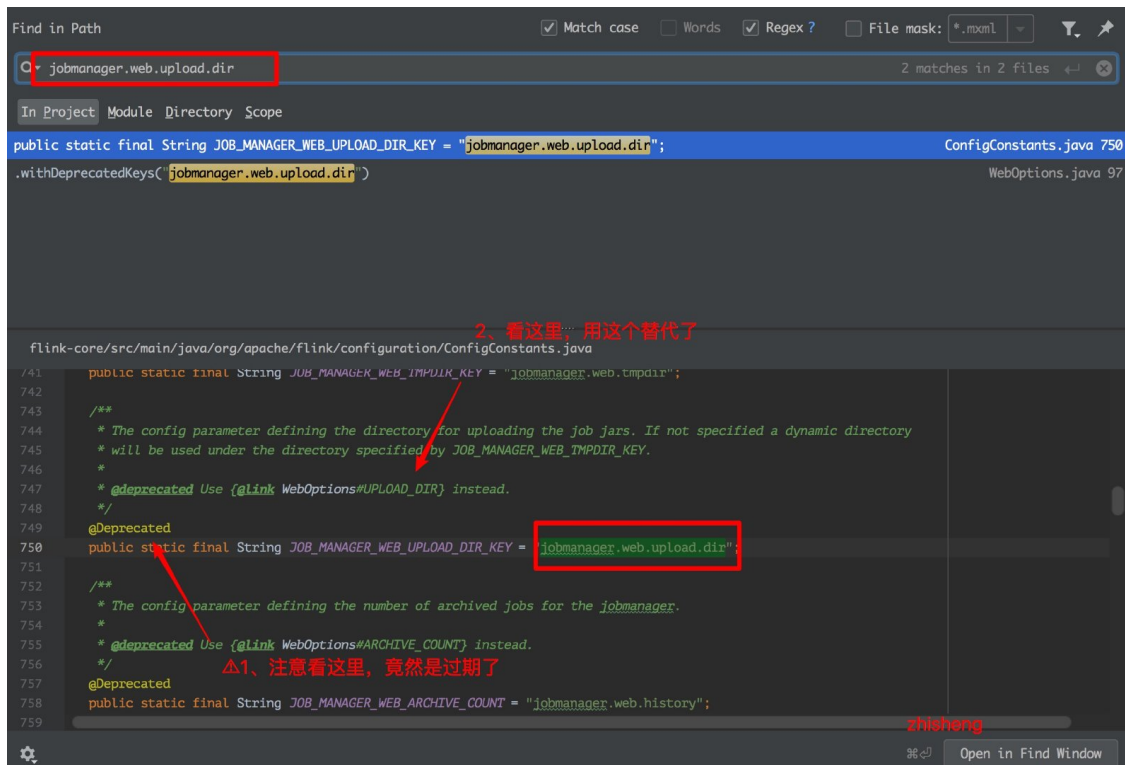
zhisheng

发现这 tm 不就是想要的吗？都支持配置文件来填写上传的 jar 后存储的目录了！赶紧点进去看一波源码：



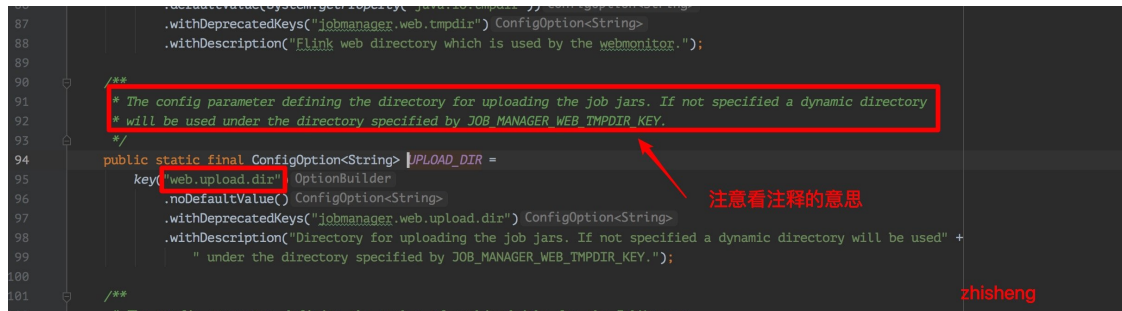
源码确认

这个 `jobmanager.web.upload.dir` 是不是？我去看下 1.8 的源码确认一下：



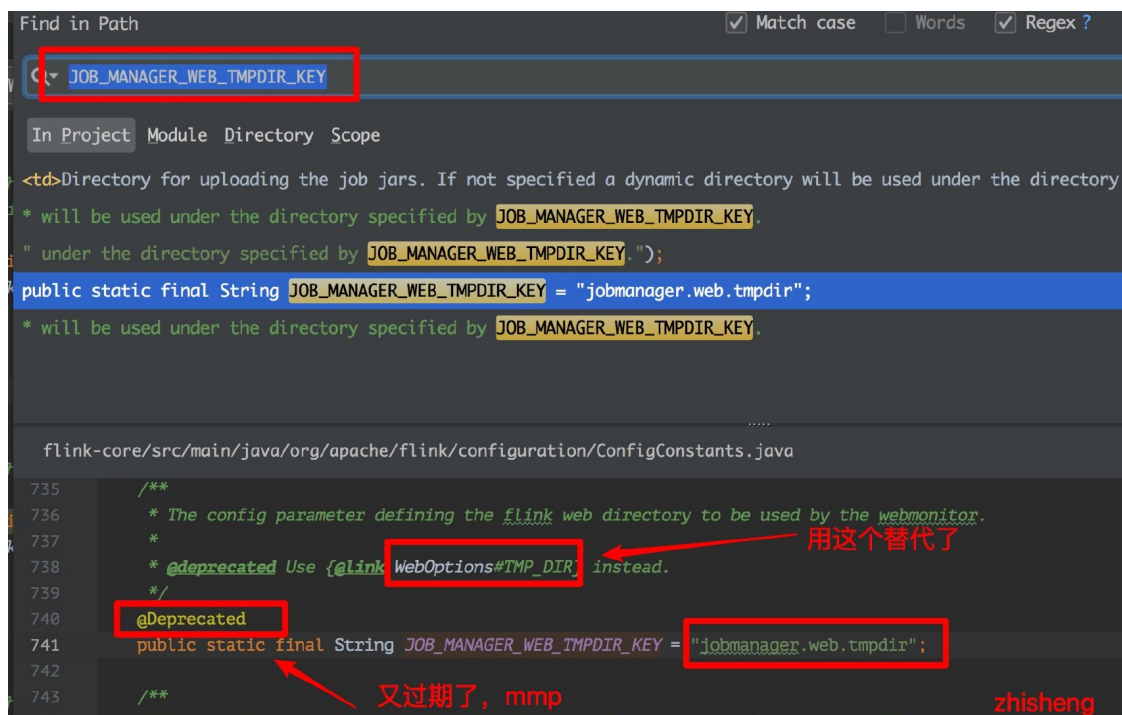
发现这个 `jobmanager.web.upload.dir` 还过期了，用 `WebOptions` 类中的 `UPLOAD_DIR` 替代了！

继续跟进去看看这个 `UPLOAD_DIR` 是啥玩意？



看这注释的意思是说，如果这个配置 `web.upload.dir` 没有配置具体的路径的话就会使用 `JOB_MANAGER_WEB_TMPDIR_KEY` 目录，那么来看看是否配置了这个目录呢？

确实没有配置这个 jar 文件上传的目录，那么来看看这个临时目录 `JOB_MANAGER_WEB_TMPDIR_KEY` 是在哪里的？



又是一个过期的目录，mmp，继续跟下去看下这个目录 `TMP_DIR`。

```

        .withDescription("Flag indicating whether to override SSL support for the JobManager Web UI.");

/**
 * The config parameter defining the flink web directory to be used by the webmonitor.
 */
@Documentation.OverrideDefault("System.getProperty(\"java.io.tmpdir\")")
public static final ConfigOption<String> TMP_DIR =
    key("web.tmpdir").optionBuilder()
        .defaultValue(System.getProperty("java.io.tmpdir")) ConfigOption<String>
        .withDeprecatedKeys("jobmanager.web.tmpdir") ConfigOption<String>
        .withDescription("Flink web directory which is used by the webmonitor.");

/**

```

如果配置了这个目录则会覆盖默认目录

默认的目录

zhisheng

我们查看下配置文件是否有配置这个 `web.tmpdir` 的值，又是没有：

```

zhisheng@zhisheng /usr/local/blink-1.5.1/conf > cat flink-conf.yaml | grep web.upload.dir
zhisheng@zhisheng /usr/local/blink-1.5.1/conf > cat flink-conf.yaml | grep web.upload.dir
zhisheng@zhisheng /usr/local/blink-1.5.1/conf > cat flink-conf.yaml | grep web.upload.dir
zhisheng@zhisheng /usr/local/blink-1.5.1/conf > cat flink-conf.yaml | grep web.tmpdir
zhisheng@zhisheng /usr/local/blink-1.5.1/conf > cat flink-conf.yaml | grep web.tmpdir
zhisheng@zhisheng /usr/local/blink-1.5.1/conf > cat flink-conf.yaml | grep web.tmpdir
zhisheng@zhisheng /usr/local/blink-1.5.1/conf > cat flink-conf.yaml | grep web.tmpdir

```

so，它肯定使用的是 `System.getProperty("java.io.tmpdir")` 这个目录了，

我查看了本地电脑起的 job 它的配置中有这个配置如下：

```
java.io.tmpdir /var/folders/mb/3vpbvkkx13l2jmpt2kmmt0fr0000gn/T/
```

swap.delay	90
diskioutil.alert.switch	true
java.io.tmpdir	/var/folders/mb/3vpbvkkx13l2jmpt2kmmt0fr0000gn/T/
java.version	1.8.0_152
java.vm.specification.name	Java Virtual Machine Specification

zhisheng

再观察了下 job，发现 jobManager 这里有个 `web.tmpdir` 的配置：

```
web.tmpdir /var/folders/mb/3vpbvkkx13l2jmpt2kmmt0fr0000gn/T/flink-
web-ea909e9e-4bac-452d-8450-b4ff082298c7
```

Version: 1.5.1 Commit: 185d730 @ 27.01.2019 @ 23:01:07

Key	Value
rest.port	8081
taskmanager.managed.memory.size	256
port	8789
metrics.reporter.jmx.class	org.apache.flink.metrics.jmx.JMXReporter
parallelism.default	1
jobmanager.rpc.address	localhost
jobmanager.heap.mb	1024
taskmanager.numberofTaskSlots	6
web.tmpdir	/var/folders/mb/3vpbvkkx13l2jmnt2kmt0fr0000gn/T/flink-web-ea909e9e-4bac-452d-8450-b4ff082298c7
taskmanager.heap.mb	2048
jobmanager.rpc.port	6123

zhisheng

发现这个 `web.tmpdir` 的就是由 `java.io.tmpdir` + “flink-web-” + UUID 组成的！

水落石出

进入这个目录发现我们上传的 jar 终于被找到了：

```

Last login: Thu Mar 14 10:53:15 on ttys007
zhisheng@zhisheng ~ % cd /var/folders/mb/3vpbvkkx13l2jmnt2kmt0fr0000gn/T/flink-web-ea909e9e-4bac-452d-8450-b4ff082298c7
zhisheng@zhisheng: /var/folders/mb/3vpbvkkx13l2jmnt2kmt0fr0000gn/T/flink-web-ea909e9e-4bac-452d-8450-b4ff082298c7 % ls
total 0
drwxr-xr-x  18 zhisheng  staff   576B  3 13 21:30 flink-web-ui
drwxr-xr-x   3 zhisheng  staff    96B  3 13 21:30 flink-web-upload
zhisheng@zhisheng: /var/folders/mb/3vpbvkkx13l2jmnt2kmt0fr0000gn/T/flink-web-ea909e9e-4bac-452d-8450-b4ff082298c7 % ls
total 240576
-rw-r--r--   1 zhisheng  staff   112M  3 13 21:30 70241fd1-e987-45b6-91f6-bf621fc9ddb3_c7.jar
zhisheng@zhisheng: /var/folders/mb/3vpbvkkx13l2jmnt2kmt0fr0000gn/T/flink-web-ea909e9e-4bac-452d-8450-b4ff082298c7 % ls
total 240576
-rw-r--r--   1 zhisheng  staff   112M  3 13 21:30 70241fd1-e987-45b6-91f6-bf621fc9ddb3_c7.jar
zhisheng@zhisheng: /var/folders/mb/3vpbvkkx13l2jmnt2kmt0fr0000gn/T/flink-web-ea909e9e-4bac-452d-8450-b4ff082298c7 %

```

配置上传 jar 目录确认

上面我们虽然已经知道我们上传的 jar 是存储在这个临时目录里，那么我们现在要验证一下，我们在配置文件中配置一下上传 jar 的固定位置，我们先在目录下创建一个 `jars` 目录，然后在配置文件中加入这个配置：

```
web.tmpdir: /usr/local/blink-1.5.1/jars
```



```

drwxr-xr-x  21 zhisheng  staff   672B  1 30 10:30 ont
zhisheng@zhisheng: /usr/local/blink-1.5.1$ mkdir jars
zhisheng@zhisheng: /usr/local/blink-1.5.1$ ll
total 56
-rw-r--r--  1 zhisheng  staff   18K  1 29 11:44 LICENSE
-rw-r--r--  1 zhisheng  staff  779B  1 29 11:44 NOTICE
-rw-r--r--  1 zhisheng  staff  1.3K  1 28 16:51 README.txt
drwxr-xr-x  15 zhisheng  staff   480B  1 30 10:30 batch_conf
drwxr-xr-x  26 zhisheng  staff   832B  1 30 10:30 bin
drwxr-xr-x  16 zhisheng  staff   512B  3 11 10:58 conf
drwxr-xr-x  7 zhisheng  staff   224B  1 30 10:30 examples
drwxr-xr-x  2 zhisheng  staff    64B  3 14 10:58 jars
drwxr-xr-x  7 zhisheng  staff   224B  1 30 10:30 lib
drwxr-xr-x 122 zhisheng  staff   3.8K  3 13 21:30 log
drwxr-xr-x  21 zhisheng  staff   672B  1 30 10:30 opt
zhisheng@zhisheng: /usr/local/blink-1.5.1$ cd jars
zhisheng@zhisheng: /usr/local/blink-1.5.1/jars$

```

更改之后再看 `web.tmpdir` 是这样的:

Apache Flink Dashboard		Version: 1.5.1	Commit: 185d730 @ 27.01.2019 @ 23:01:07
Overview	Configuration	Logs	Stdout
Jobs			
Running Jobs			
Completed Jobs			
Task Managers			
Job Manager			
Submit New Job			
Key	Value		
rest.port	8081		
taskmanager.managed.memory.size	256		
parallelism.default	1		
jobmanager.rpc.address	localhost		
jobmanager.heap.mb	1024		
taskmanager.numberOfTaskSlots	6		
web.tmpdir	/usr/local/blink-1.5.1/jars/ link-web-7a98165b-1d56-44be-be8c-d0cd9166b179		
taskmanager.heap.mb	2048		
jobmanager.rpc.port	6123		

从 Flink UI 上上传了三个 jar, 查看 `/usr/local/blink-1.5.1/jars/flink-web-7a98165b-1d56-44be-be8c-d0cd9166b179` 目录下就出现了我们的 jar 了。

```

log flink-ccc http://logging.apache.org/log4j/1.2/faq.html#config for more info.
Starting taskexecutor daemon on host zhisheng.
zhisheng@zhisheng: /usr/local/blink-1.5.1/bin$ cd ..
zhisheng@zhisheng: /usr/local/blink-1.5.1$ cd jars
zhisheng@zhisheng: /usr/local/blink-1.5.1/jars$ cd flink-web-7a98165b-1d56-44be-be8c-d0cd9166b179
zhisheng@zhisheng: /usr/local/blink-1.5.1/jars/flink-web-7a98165b-1d56-44be-be8c-d0cd9166b179$ ll
total 0
drwxr-xr-x  18 zhisheng  staff   576B  3 14 11:08 flink-web-ui
drwxr-xr-x   5 zhisheng  staff   160B  3 14 11:10 flink-web-upload
zhisheng@zhisheng: /usr/local/blink-1.5.1/jars/flink-web-7a98165b-1d56-44be-be8c-d0cd9166b179$ cd flink-web-upload
zhisheng@zhisheng: /usr/local/blink-1.5.1/jars/flink-web-7a98165b-1d56-44be-be8c-d0cd9166b179/flink-web-upload$ ll
total 717560
-rw-r--r--  1 zhisheng  staff  112M  3 14 11:10 0a1169de-df46-468d-91af-070f7347bdd6_..._jar
-rw-r--r--  1 zhisheng  staff  111M  3 14 11:10 1c58ce85-009d-4632-936e-47c19650a526_application_..._jar
-rw-r--r--  1 zhisheng  staff  112M  3 14 11:10 c1e25322-d8d9-4830-ac7f-4370051ce9f8_..._jar
zhisheng@zhisheng: /usr/local/blink-1.5.1/jars/flink-web-7a98165b-1d56-44be-be8c-d0cd9166b179/flink-web-upload$

```

我们重启 Flink，发现这三个 jar 又没有了，这也能解释之前我自己也遇到过的问题了，Flink 重启后之前所有上传的 jar 都被删除了！作为生产环境，这样玩，肯定不行的，所以我们还是得固定一个目录来存储所有的上传 jar 包，并且不能够被删除，要配置固定的目录（Flink 重启也不删除的话）需要配置如下：

```
web.upload.dir: /usr/local/blink-1.5.1/jars
```

这样的话，就可以保证你的 jar 不再会被删除了！

总结

本文从知识星球一个朋友的问题，从现象到本质再到解决方案的讲解了下如何找到 Flink UI 上上传的 jar 包藏身之处，并提出了如何解决 Flink 上传的 jar 包被删除的问题。


关注我

微信公众号：zhisheng


另外我自己整理了些 Flink 的学习资料，目前已经全部放到微信公众号了。你可以加我的微信：zhisheng_tian，然后回复关键字：**Flink** 即可无条件获取到。



更多私密资料请加入知识星球！




Flink 精进学习
星主: zhisheng



知识星球

微信扫码预览星球详情



Github 代码仓库

<https://github.com/zhisheng17/flink-learning/>

以后这个项目的所有代码都将放在这个仓库里，包含了自己学习 flink 的一些 demo 和博客。

相关文章

- 1、《从0到1学习Flink》—— Apache Flink 介绍
- 2、《从0到1学习Flink》—— Mac 上搭建 Flink 1.6.0 环境并构建运行简单程序入门
- 3、《从0到1学习Flink》—— Flink 配置文件详解
- 4、《从0到1学习Flink》—— Data Source 介绍
- 5、《从0到1学习Flink》—— 如何自定义 Data Source ?
- 6、《从0到1学习Flink》—— Data Sink 介绍
- 7、《从0到1学习Flink》—— 如何自定义 Data Sink ?
- 8、《从0到1学习Flink》—— Flink Data transformation(转换)
- 9、《从0到1学习Flink》—— 介绍Flink中的Stream Windows
- 10、《从0到1学习Flink》—— Flink 中的几种 Time 详解
- 11、《从0到1学习Flink》—— Flink 写入数据到 ElasticSearch
- 12、《从0到1学习Flink》—— Flink 项目如何运行?
- 13、《从0到1学习Flink》—— Flink 写入数据到 Kafka
- 14、《从0到1学习Flink》—— Flink JobManager 高可用性配置
- 15、《从0到1学习Flink》—— Flink parallelism 和 Slot 介绍
- 16、《从0到1学习Flink》—— Flink 读取 Kafka 数据批量写入到 MySQL
- 17、《从0到1学习Flink》—— Flink 读取 Kafka 数据写入到 RabbitMQ