

Hadoop client 代码阅读

Day01:

IdentityHashMap:

IdentityHashMap 有其特殊用途，比如序列化或者深度复制。或者记录对象代理。

举个例子，jvm 中的所有对象都是独一无二的，哪怕两个对象是同一个 class 的对象

，而且两个对象的数据完全相同，对于 jvm 来说，他们也是完全不同的，

如果要用一个 map 来记录这样 jvm 中的对象，你就需要用 IdentityHashMap，而不能使用其他 Map 实现

1.什么是 java 的 spi

SPI 全称为 (Service Provider Interface) ,是 JDK 内置的一种服务提供发现机制。目前有不少框架用它来做服务的扩展发现，简单来说，它就是一种动态替换发现的机制，举个例子来说，有个接口，想运行时动态的给它添加实现，你只需要添加一个实现。具体是在 JAR 包的"src/META-INF/services/"目录下建立一个文件，文件名是接口的全限定名，文件的内容可以有多行，每行都是该接口对应的具体实现类的全限定名。

2.运用场景

比如你想扩展一些框架，如 spring 的一些功能，就是要实现它接口，然后自己配置了。

一般使用接口的实现类都是静态 new 一个实现类赋值给接口引用，如下：

```
HelloService service = new HelloImpl();
```

如果需要动态的获取一个接口的实现类呢？全局扫描全部的 Class，然后判断是否实现了某个接口？代价太大，一般不会这么做。一种合适的方式就是使用配置文件，把实现类名配置在某个地方，然后读取这个配置文件，获取实现类名。JDK 给我们提供的 TestServiceLoader 就是这种方式。

总结:

1. 文件系统有大约 9 个实现，hdfs 只是其中之一
2. 文件系统对象获取的时候是通过反射获取，获取前先检查缓存中是否存在
3. 关闭客户端时，文件系统会被关闭调
4. 文件系统的统计信息 如读写次数 放在 map IdentityHashMap 中
5. DistuributeFilesystem 内置的 Dfs，是用来跟集群沟通的桥梁。

今日 debug 至 DistuributeFilesystem 的 create 方法