# 大数据

## 什么是大数据？

大数据就是海量数据的处理。

大数据（big data），指无法在一定时间范围内用常规软件工具进行捕捉、管理和处理的数据集合，是需要新处理模式才能具有更强的决策力、洞察发现力和流程优化能力的海量、高增长率和多样化的信息资产（资源）。

在[维克托·迈尔-舍恩伯格](https://baike.baidu.com/item/%E7%BB%B4%E5%85%8B%E6%89%98%C2%B7%E8%BF%88%E5%B0%94-%E8%88%8D%E6%81%A9%E4%BC%AF%E6%A0%BC" \t "_blank)及肯尼斯·库克耶编写的《[大数据时代](https://baike.baidu.com/item/%E5%A4%A7%E6%95%B0%E6%8D%AE%E6%97%B6%E4%BB%A3/15434499" \t "_blank)》[2]  中大数据指不用随机分析法（[抽样调查](https://baike.baidu.com/item/%E6%8A%BD%E6%A0%B7%E8%B0%83%E6%9F%A5" \t "_blank)）这样捷径，而采用所有数据进行分析处理。大数据的5V特点（IBM提出）：Volume（大量）、Velocity（高速）、Variety（多样）、Value（低价值密度）、Veracity（真实性）。

## 为什么要学习大数据?

拓宽发展面

大数据已经成为基础技术，现在最火的人工智能，物联网，就建立在大数据的基础上。

面向面试，提高认知深度，拓宽知识广度，增加面试筹码，提高薪资。

## 课程设计

以日志分析系统项目贯穿。

做到当堂消化，不占用课余时间，空出时间复习前边知识，。

1. 了解什么是大数据，分析项目需求设计架构，分析指标讲解，数据获取，项目搭建。
2. Hadoop详解，使用
3. Flume详解，使用，将数据收集到Hadoop中。
4. Hive详解，使用
5. 基于Hive实现离线分析
6. 基于Storm实现实时分析
7. Hadoop进阶，完全分布式的搭建

学习要求：

每个技术的概念及设计思想。

熟悉项目流程，了解大数据处理的重要指标。

## 大数据相关技术

### 海量数据获取

爬虫（分析别人家的事）：Python，java…

日志（分析自己的事）：log4j、flume（数据的收集聚集加载）

历史数据（分析以前的事）：mysql、oracle、ETL（数据的提取转化加载）

### 海量数据存储

HDFS（Hive、Hbase、Spark底层存储都是HDFS）Hadoop distributed file system

### 海量数据清洗

数据清洗没有我们想的那么复杂，方式有很多：

爬虫过程中可以用java代码直接清洗，flume可以清洗，Hive可以清洗，Storm可以清洗。整个流程中几乎每个步骤都可以做到数据的清洗。

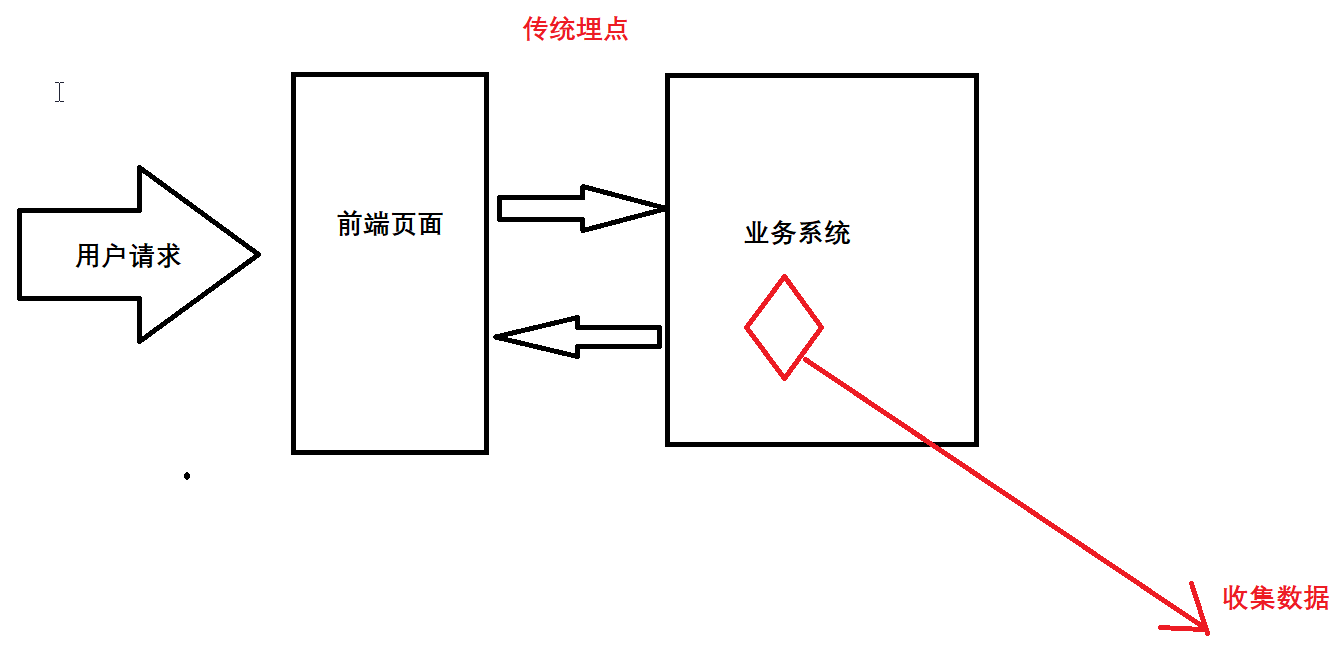
### 海量数据处理

离线数据处理：MapReduce（hadoop的一部分）、Hive（底层是MR，业界主流）、SparkSQL（学习Hive，甚至底层就是Hive）

流式数据处理：Storm（速度最快，亚秒级）、SparkStreaming（速度没有Storm快，但是吞吐量高）

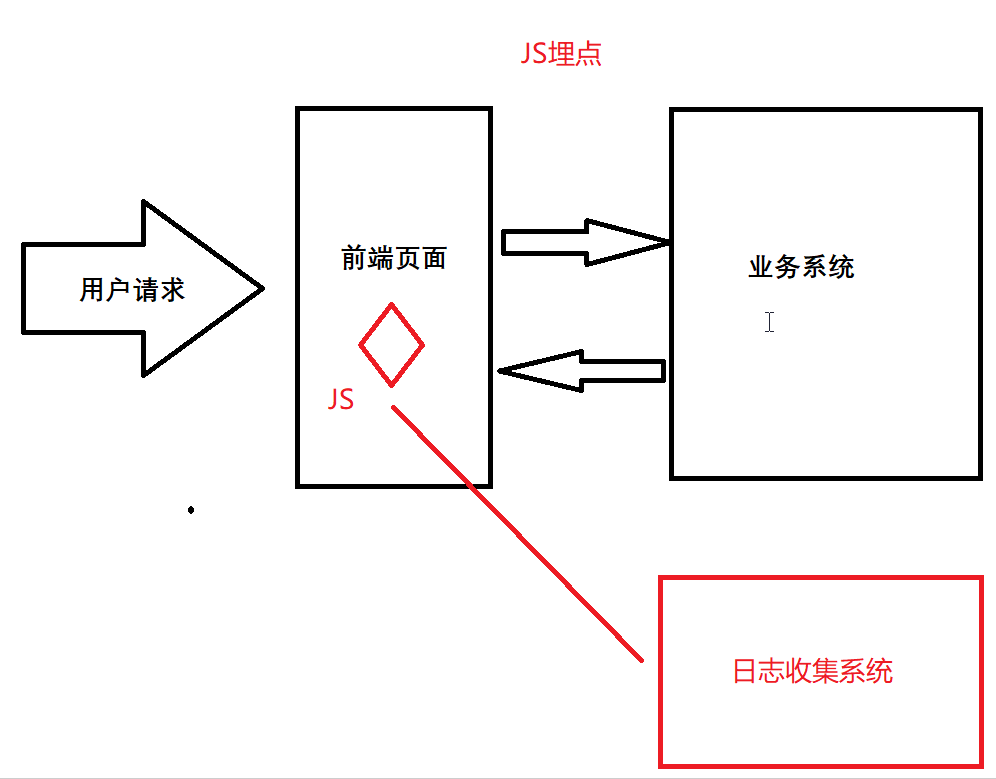
# 网站日志分析系统

## 数据收集：JS埋点



传统埋点方式的缺点：

直接嵌入业务系统，代码不好维护，影响业务系统性能。



JS埋点优点：

日志收集系统与业务系统相互独立，互不影响，不占用业务系统性能,能收集到更多更详细的用户信息。

### 前段页面如何将数据发送到后端系统？

Post提交、get提交

Form表单：组织起来比较麻烦，很多东西用不到

Get请求：https://www.jd.com/?cu=true&utm\_source=baidu-pinzhuan&utm\_medium=cpc&utm\_campaign=t\_288551095\_baidupinzhuan&utm\_term=0f3d30c8dba7459bb52f2eb5eba8ac7d\_0\_bd79f916377147b6aef8164d97d9abac

综合考虑，选择Get请求方式发送数据到后端系统

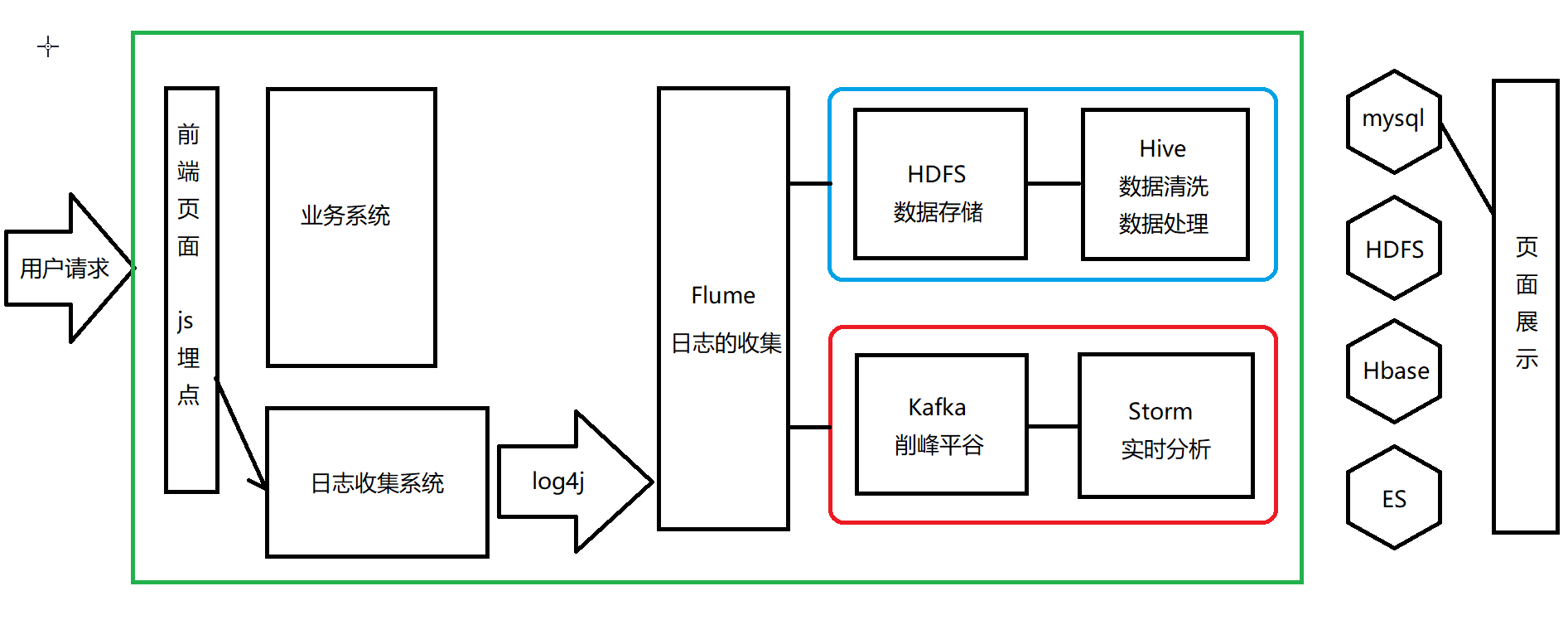
### 在哪里发起Get请求？

跨域请求

1.Script标签中src实现跨域访问，但是每个页面插入这样的代码影响页面逻辑。比较乱。

2.img标签中src实现跨域访问，将该图片插入到真个页面最后，像素1，边框0，用户看不到。

## 系统架构设计



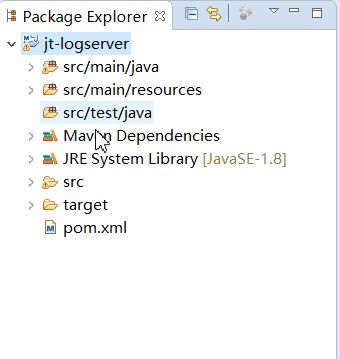
离线分析：js埋点🡪日志收集系统（log4j）🡪flume🡪HDFS🡪Hive

实时分析：js埋点🡪日志收集系统（log4j）🡪flume🡪Kafka🡪Storm

## 八个重要指标

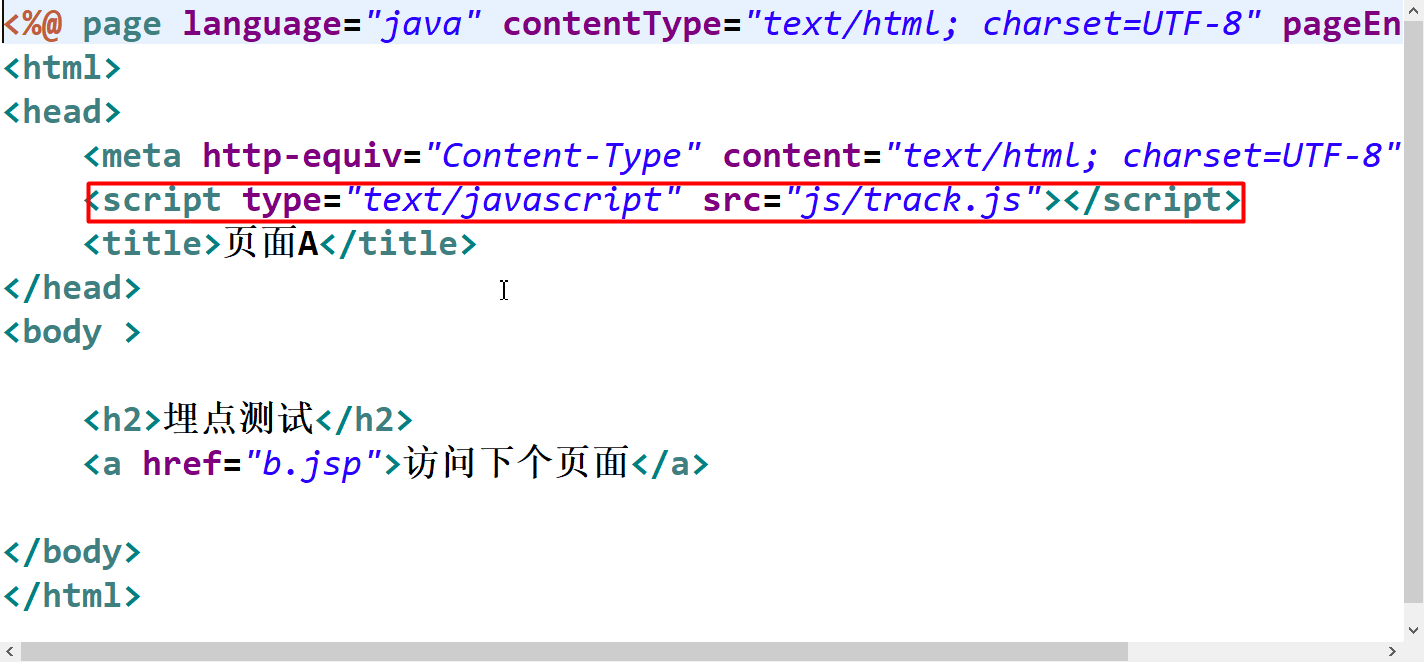
1. PV：点击量，一条日志就是一个点击量，越高越好。
2. UV：独立访客数，通过cookie记录uvid，用户每次访问带着cookie，系统识别是新的还是之前的，越高越好。
3. VV:独立会话数，通过cookie记录ssid，记录不同的ssid数量，体现时间段内所有用户一共访问了多少次，越高越好。
4. BR：跳出率，只访问一个页面就走了的访客/所有的独立访客，越低越好。
5. NewIP：新增独立IP，历史数据中（ip）没有出现过的访客IP地址数量， 越高越好，反映用户增长量，反映系统生命力。
6. NewCust：新增独立访客，历史数据中（uvid）没有出现过的独立访客，越高越好。
7. AvgTime：平均在线时长，所有会话的时长总和/会话数量，越高越好。
8. AvgDeep：平均访问深度，一个会话的访问页面数量叫做访问深度，每个会话的访问深度之和/会话数量。越高越好。

## 项目搭建



### Js

页面引入：



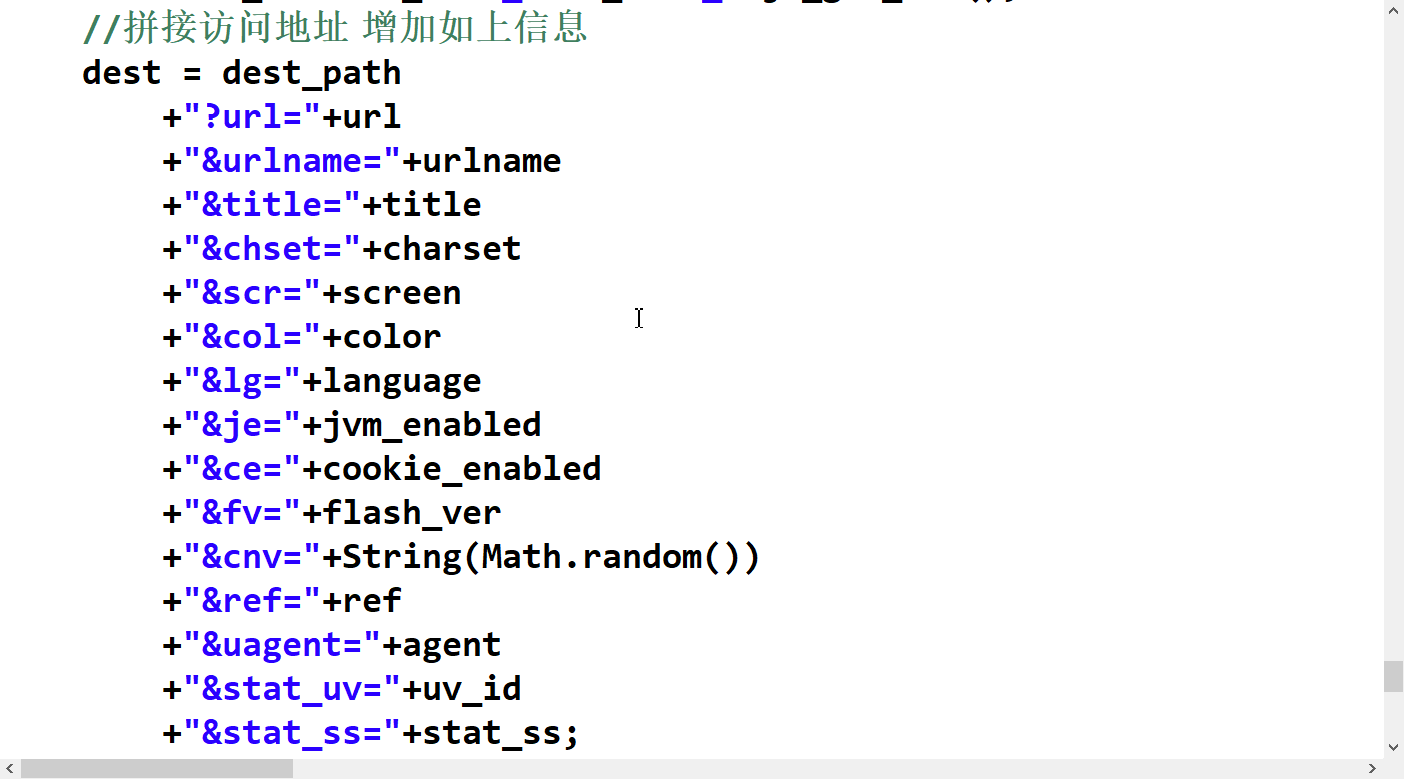
Js加载：



主函数：



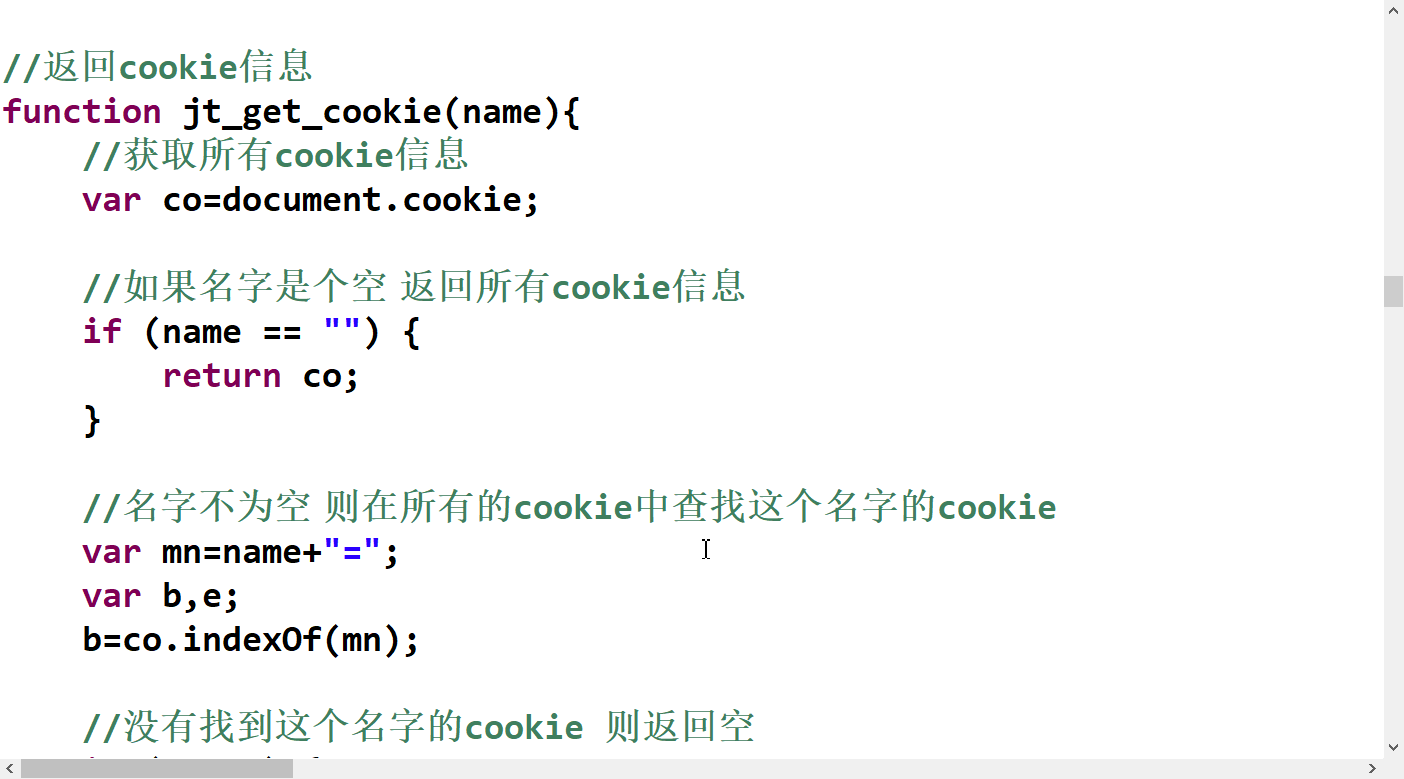
URL拼接：



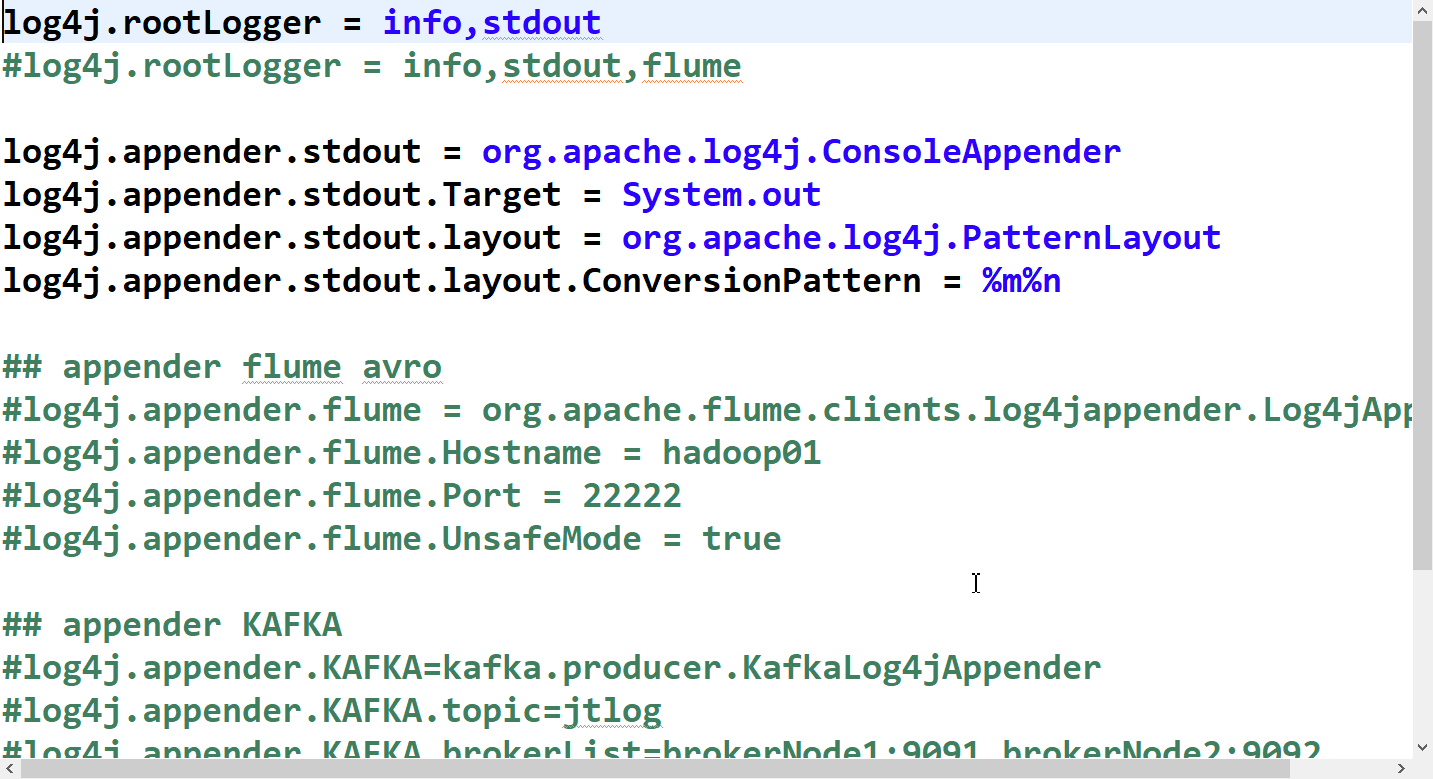
设置cookie：



获取cookie：



### Log4j



### Servlet



负责数据流入日志前的处理，获取请求参数，重新组织，并输出到Log4j中。

## 集群环境准备

### 准备虚拟机

centOS7.0克隆三个虚拟机hadoop01、hadoop02、hadoop03均为NAT模式，其中hadoop01内存设置为1G，hadoop02和hadoop03为512M。

### 修改为静态IP

修改配置文件

cd /etc/sysconfig/network-scripts #进入网络配置目录

dir ifcfg\* #找到网卡配置文件

ifcfg-ens33 #找到版本最新的文件并修改

vim ifcfg-ens33

或者

vim /etc/sysconfig/network-scripts/ifcfg-ens33

（\*输入字母i进入编辑模式：修改黄色的ip内容，修改完后ctrl+c一下，再按 :wq保存退出\*，如果发现修改错了:q!是不保存退出\*）

配置文件内容

TYPE=Ethernet

BOOTPROTO=static #改成static，针对NAT

NAME=eno16777736

UUID=4cc9c89b-cf9e-4847-b9ea-ac713baf4cc8

DEVICE=eno16777736

DNS1=114.114.114.114 #和网关相同

ONBOOT=yes #开机启动此网卡

IPADDR=192.168.65.161 #固定IP地址

NETMASK=255.255.255.0 #子网掩码

GATEWAY=192.168.65.2 #网关和NAT自动配置的相同，不同则无法登录

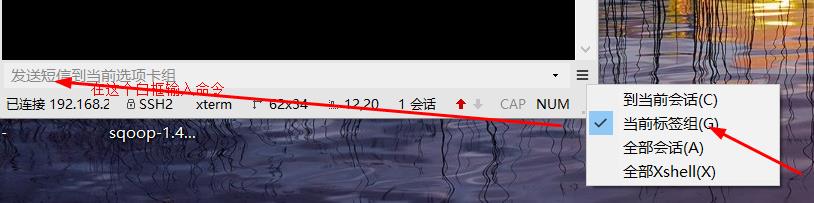
重启网络

systemctl restart network.service #重启网络centos7

查看IP

ip addr #查看IP地址 ip add

### 关闭防火墙（\*三台一起修改：启动连接xshell后再相同目录下执行\*）



systemctl stop firewalld.service #关闭防火墙服务

systemctl disable firewalld.service #禁止防火墙开启启动

systemctl restart iptables.service #重启防火墙使配置生效

systemctl enable iptables.service #设置防火墙开机启动

检查防火墙状态

[root@hadoop01 ~]# firewall-cmd --state #检查防火墙状态

not running #返回值，未运行

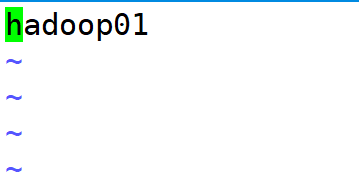
下载安装上传文件插件

yum install -y lrzsz

### 修改主机名

vim /etc/hostname

删掉原内容，改为：hadoop01



Hostname为内核属性需要重启才能生效，不想重启可以定义零时主机名

执行以下命令

hostname hadoop01

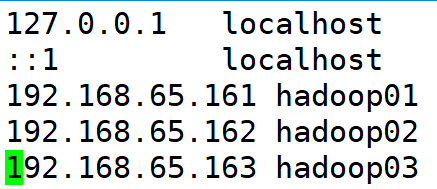
查看主机名

hostname

### 修改hosts文件（把ip替换为你自己的，设置好之后再打开检查是否设置正确）

vi /etc/hosts #127.0.0.1 dredis

删掉原来所有内容，改为以下



127.0.0.1 localhost

::1 localhost

192.168.202.161 hadoop01

192.168.202.162 hadoop02

192.168.202.163 hadoop03

### 设置免密登录（也是切换到标签组模式下执行命令）

每台都执行以下操作

在白框任意路径执行：

ssh-keygen

然后三次直接回车不输入任何内容

得到秘钥后如图



执行：

ssh-copy-id root@hadoop01

ssh-copy-id root@hadoop02

ssh-copy-id root@hadoop03

输入以上每一行命令的过程中提示你输入yes和密码root，输完即可；

测试

ssh hadoop02

不需要输入密码直接进入说明成功，exit退出

（\*配置失败后可以在重新设置一次\*）

### 安装JDK

在用户目录 ~ 中编辑go文件（\*这个也可以不设置，就是快速找到src目录下\*）

vim go

编辑内容为

cd /usr/local/src

保存后执行 . go即可到达指定目录

/usr/local/src下创建目录java

mkdir java

上传安装包并解压

tar -xvf 安装包

删除压缩包后远程发送给其余俩台 scp -r java/ root@hadoop02:/usr/local/src

修改环境变量

vim /etc/profile

最后添加

#set java environment

JAVA\_HOME=/usr/local/src/java/jdk1.7.0\_51

JAVA\_BIN=/usr/local/src/java/jdk1.7.0\_51/bin

PATH=$JAVA\_HOME/bin:$PATH

CLASSPATH=.:$JAVA\_HOME/lib/dt.jar:$JAVA\_HOME/lib/tools.jar

export JAVA\_HOME JAVA\_BIN PATH CLASSPATH

加载环境变量

source /etc/profile

测试

java -version

远程发送给其余俩台 scp /etc/profile root@hadoop02:/etc/profile

### 修改windows中的hosts文件

192.168.202.161 hadoop01

192.168.202.162 hadoop02

192.168.202.163 hadoop03

### 安装ZooKeeper（作为了解内容，本次项目不使用）

/usr/local/src下创建目录zk

上传解压包并解压，在conf目录下

cp zoo\_sample.cfg zoo.cfg

修改vim zoo.cfg配置文件zoo.cfg 中高亮部分

tickTime=2000 #tickTime心跳时间，

clientPort=2181 #访问端口

dataDir=/usr/local/src/zk/zookeeper-3.4.8/tmp #设置日志路径

在配置文件最后添加以下三行代码

server.1=hadoop01:2888:3888 #集群最少3个节点，可按机器名

server.2=hadoop02:2888:3888 #2888指follower连leader端口

server.3=hadoop03:2888:3888 #3888指定选举的端口

根目录下创建tmp文件夹，并在其中创建文件myid编辑1；

发送到hadoop02和hadoop03分别修改myid文件里的值2和3；

scp -r zk/ root@hadoop02:/usr/local/src/

操作命令三台bin目录下启动

./zkServer.sh start #启动ZK服务

./zkServer.sh start-foreground #第二种启动方式（日志启动方式）

./zkServer.sh stop #停止ZK服务

./zkServer.sh restart #重启ZK服务

./zkServer.sh status #查看ZK状态