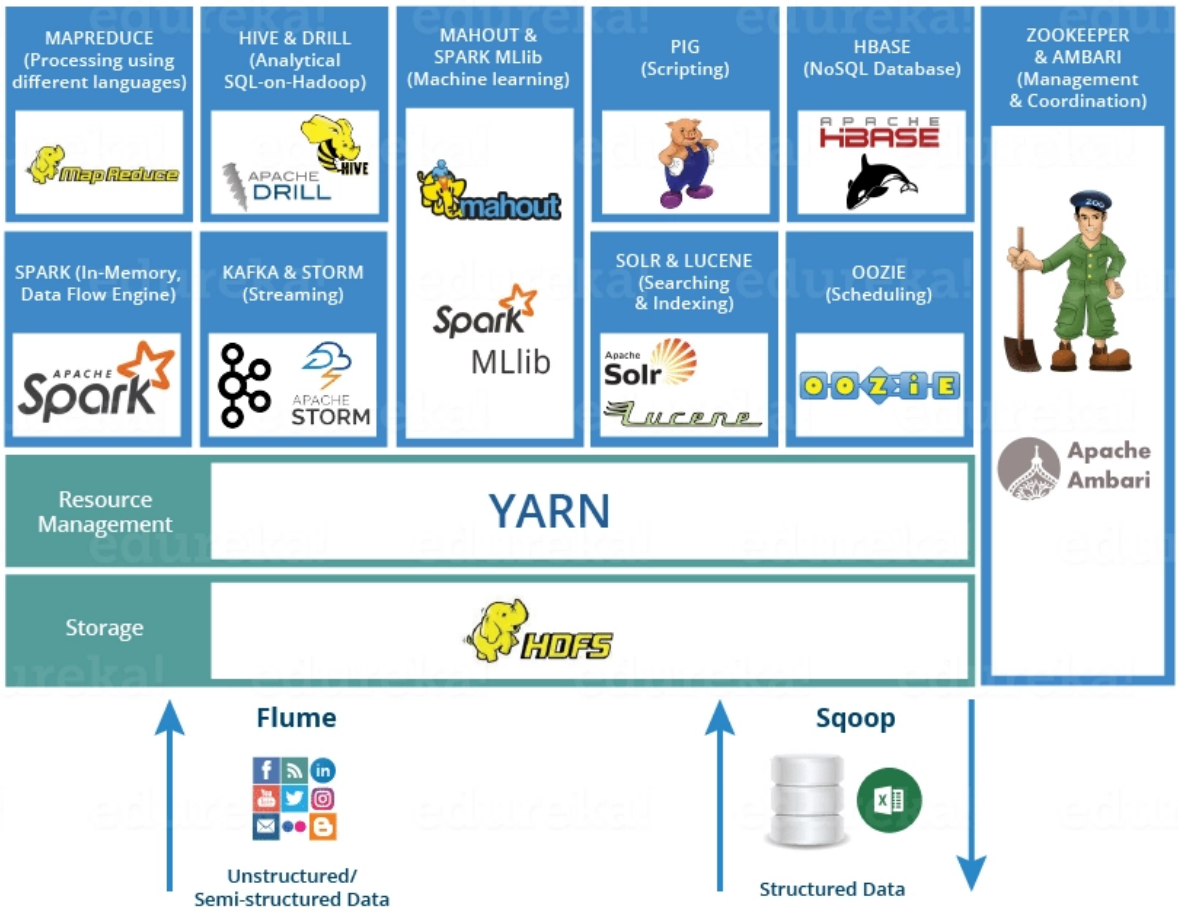
# Flume日志采集工具

## 1. Flume概述

Flume是一个高可用，高可靠的分布式的海量日志采集、聚合和传输的软件。

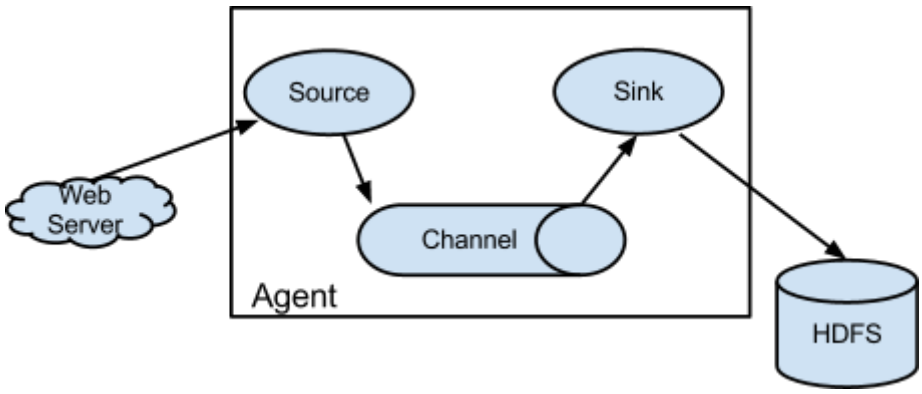
Flume的核心是把数据从**数据源(source)**收集过来，再将收集到的数据送到指定的目的地(sink)。为了保证输送的过程一定成功，在送到**目的地(sink)**之前，会先缓存数据**(channel),**待数据真正到达目的地(sink)后，flume在删除自己缓存的数据。

Flume支持定制各类数据发送方，用于收集各类型数据；同时，Flume支持定制各种数据接受方，用于最终存储数据。一般的采集需求，通过对flume的简单配置即可实现。针对特殊场景也具备良好的自定义扩展能力。因此，flume可以适用于大部分的日常数据采集场景。




## 1.2 运行机制

Flume系统中核心的角色是***agent***，agent本身是一个Java进程，一般运行在日志收集节点。




每一个agent相当于一个数据传递员，内部有三个组件：

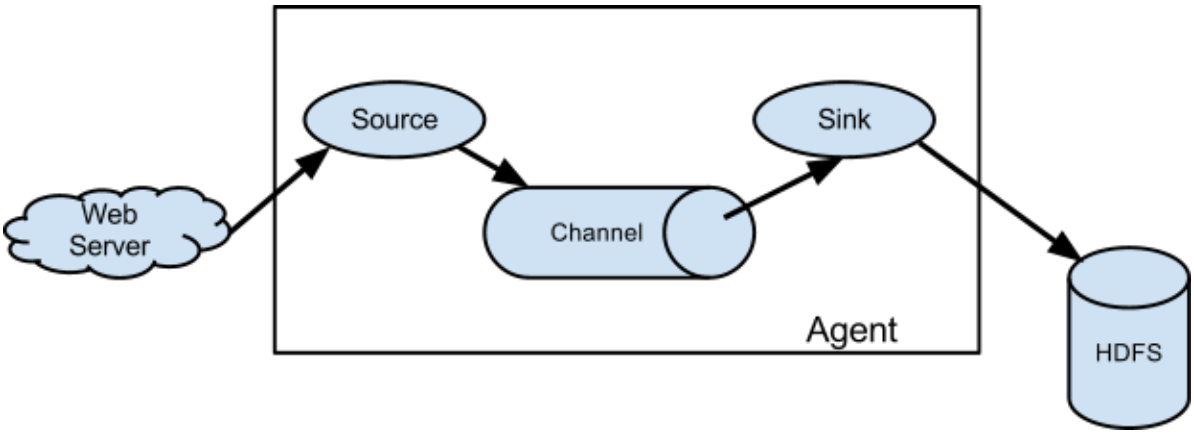
- Source：采集源，用于跟数据源对接，以获取数据；
- Sink：下沉地，采集数据的传送目的地，用于往下一级agent传递数据或者往最终存储系统传递数据；
- Channel：agent内部的数据传输通道，用于从source将数据传递到sink；

在整个数据的传输的过程中，流动的是***event***，它是Flume内部数据传输的最基本单元。event将传输的数据进行封装。如果是文本文件，通常是一行记录，event也是事务的基本单位。event从source，流向channel，再到sink，本身为一个字节数组，并可携带headers(头信息)信息。event代表着一个数据的最小完整单元，从外部数据源来，向外部的目的地去。

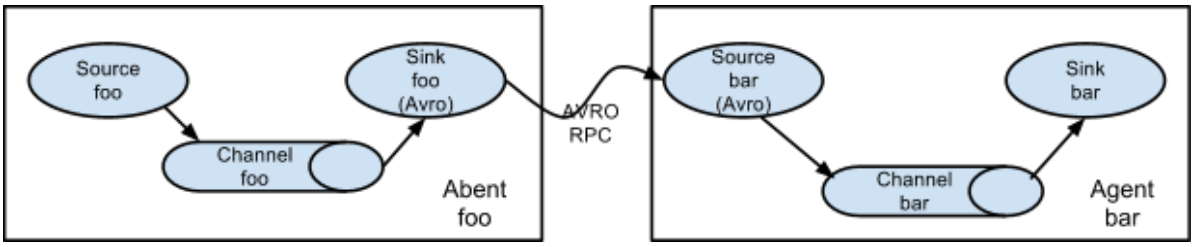
一个完整的event包括：event headers、event body、event信息，其中event信息就是flume收集到的日记记录。

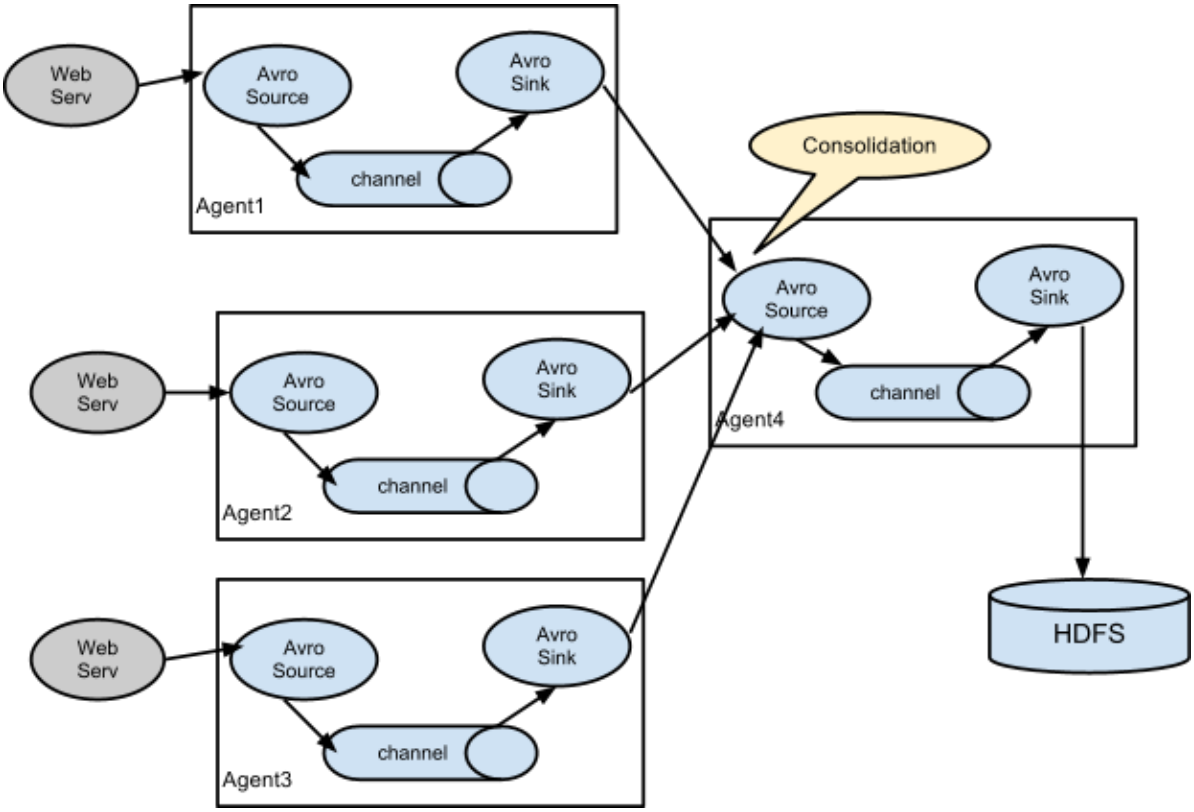
## 1.3 常见采集结构

简单结构




多级agent之间串联






## 1.4 安装部署

1. 上传安装包到数据源所在节点上
2. 解压

```
tar -zxvf apache-flume-1.9.0-bin.tar.gz -C /opt/server
```

3. 进入flume的目录，修改conf下的flume-env.sh，配置JAVA_HOME

```
cd /opt/server/apache-flume-1.9.0-bin/conf
# 先复制一份flume-env.sh.template文件
cp flume-env.sh.template flume-env.sh
# 修改
vim flume-env.sh
export JAVA_HOME=/opt/server/jdk1.8.0_131
```

# 2. 采集Nginx日志数据至HDFS

## 2.1 安装nginx

```
yum install epel-release
yum update
yum -y install nginx
```

nginx命令

```
systemctl start nginx #开启nginx服务
systemctl stop nginx #停止nginx服务
systemctl restart nginx #重启nginx服务
```

网站日志文件位置

```
cd /var/log/nginx
```

## 2.2 编写配置文件

将 lib 文件夹下的 guava-11.0.2.jar 删除以兼容 Hadoop 3.1.0  flume1.9

```
cp /opt/server/hadoop-3.1.0/share/hadoop/common/*.jar /opt/server/apache-flume-1.9.0-bin/lib
cp /opt/server/hadoop-3.1.0/share/hadoop/common/lib/*.jar /opt/server/apache-flume-1.9.0-bin/lib
cp /opt/server/hadoop-3.1.0/share/hadoop/hdfs/*.jar /opt/server/apache-flume-1.9.0-bin/lib
```

https://flume.apache.org/releases/content/1.9.0/FlumeUserGuide.html#taildir-source

创建配置文件，taildir-hdfs.conf

监控 /var/log/nginx 目录下的日志文件

```
a3.sources = r3
a3.sinks = k3
a3.channels = c3

# Describe/configure the source
a3.sources.r3.type = TAILDIR
a3.sources.r3.filegroups = f1
# 此处支持正则
a3.sources.r3.filegroups.f1 = /var/log/nginx/access.log
# 用于记录文件读取的位置信息
a3.sources.r3.positionFile = /opt/server/apache-flume-1.9.0-bin/tail_dir.json

# Describe the sink
a3.sinks.k3.type = hdfs
a3.sinks.k3.hdfs.path = hdfs://server:8020/user/tailDir
a3.sinks.k3.hdfs.fileType = DataStream
# 设置每个文件的滚动大小大概是 128M，默认值：1024，当临时文件达到该大小（单位：bytes）时，滚动成目标文件。如果设置成0，则表示不根据临时文件大小来滚动文件。
a3.sinks.k3.hdfs.rollSize = 134217700
# 默认值：10，当events数据达到该数量时候，将临时文件滚动成目标文件，如果设置成0，则表示不根据events数据来滚动文件。
a3.sinks.k3.hdfs.rollCount = 0
# 不随时间滚动，默认为30秒
a3.sinks.k3.hdfs.rollInterval = 10
# flume检测到hdfs在复制块时会自动滚动文件，导致roll参数不生效，要将该参数设置为1；否则HFDS文件所在块的复制会引起文件滚动
a3.sinks.k3.hdfs.minBlockReplicas = 1
# Use a channel which buffers events in memory
a3.channels.c3.type = memory
a3.channels.c3.capacity = 1000
a3.channels.c3.transactionCapacity = 100

# Bind the source and sink to the channel
```

```
a3.sources.r3.channels = c3
a3.sinks.k3.channel = c3
```

## 2.3 启动Flume

```
bin/flume-ng agent -c ./conf -f ./conf/taildir-hdfs.conf -n a3 -
Dflume.root.logger=INFO,console
```