



日志收集系统 Flume



Apache Flume

Apache Flume

- Apache Flume是一款高性能高可用的分布式日志收集系统
- 同类的日志收集系统还有scribe和chuwka

- Flume在节点出现故障时，日志能够被传送到其他节点上而不会丢失
- 三种级别的可靠性保障：
 - end-to-end (Agent把事件写到磁盘上，数据传送成功再删除)
 - store on failure (数据接收方崩溃时，将数据写到本地，待恢复后继续发送)
 - Best Effort (数据发送到接收方后不会进行确认)

Apache Flume



Flume采用三层架构

- Agent, collector, storage, 每一层都可以水平扩展
- agent和collector由Master统一管理, master有许多个使用zookeeper和gossip保证动态配置数据的一致性和负载均衡, 避免单点
- 用户在master上查看各个数据源或数据流执行情况, 可以用web或者shell对数据源进行配置和加载, 对数据流进行管理

单击此处添加标题

功能可扩展性

系统架构

Flume NG

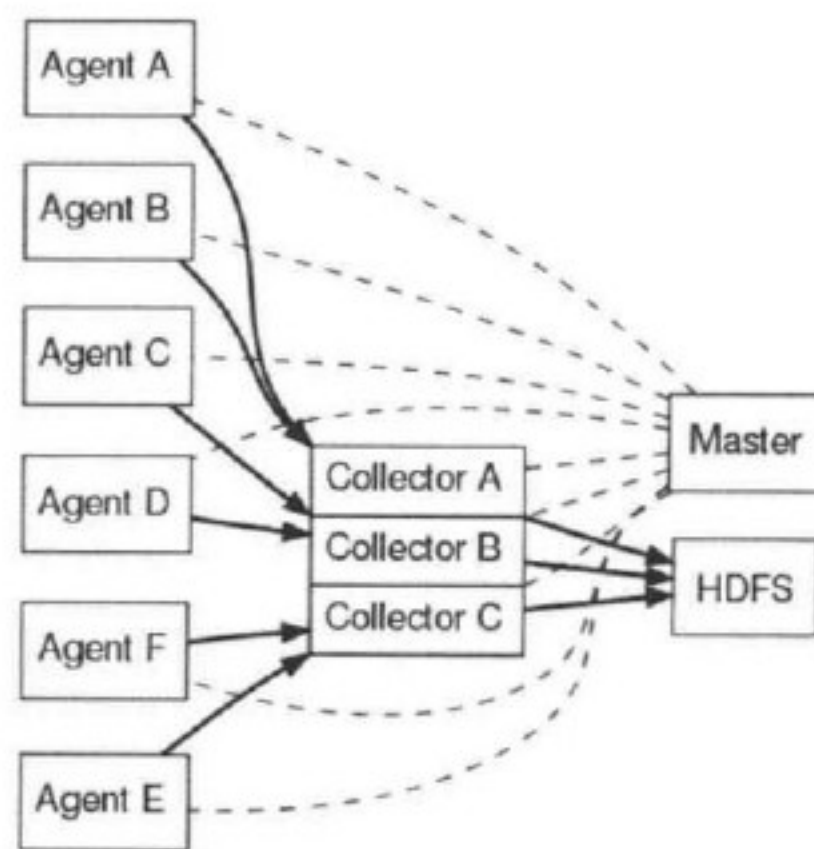
Master和Node组件

用户根据自己需要添加各种Agent (如file, syslog)、collector和storage (file, hdfs)

Agent层----->collector层----->storage层

采用分层架构，其中Agent和collector均由source和sink组成

Node根据在Master shell或web的配置决定作为agent还是collector



Agent tier → Collector tier → Storage tier

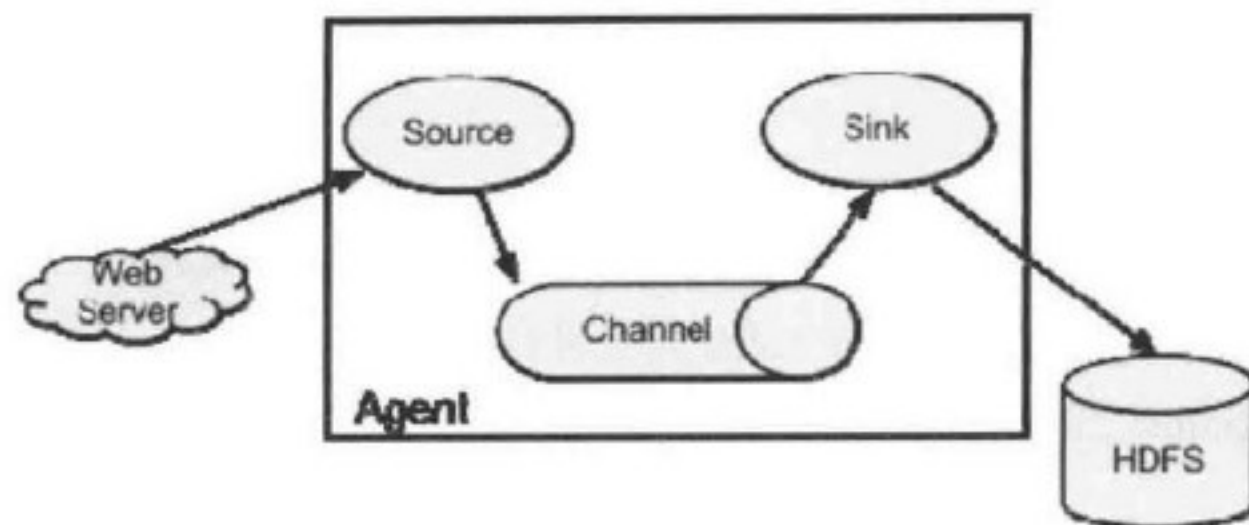
Agent



作用是将数据源的数据打给 collector，自带的数据源：

- 文件数据源
`text("filename")`,
- 探测新数据按行发送
`tail("filename")`
- 监听TCP端口发送接收的数据
`syslogTcp(5140)`

- `tailDir("dirname"[,fileregex=".*"[,startFromEnd=false[,recurseDepth=0]])`: 监听目录中的文件末尾，使用正则表达式选定需要监听的文件（不包含目录），`recurseDepth` 为递归监听其下子目录的深度，同时提供了很多 Sink，如 `console("format")`，直接将数据显示在 `console` 上。
- `text("txtfile")`: 将数据写到文件 `txtfile` 中。
- `dfs("dfsfile")`: 将数据写到 HDFS 上的 `dfsfile` 文件中。
- `syslogTcp("host",port)`: 将数据通过 TCP 传递给 `host` 节点。
- `agentSink(["machine"[,port]])`: 等价于 `agentE2ESink`，如果省略 `machine` 参数，则默认使用 `flume.collector.event.host` 与 `flume.collector.event.port` 作为默认 `collector`。
- `agentDFOSink(["machine"[,port]])`: 本地热备 Agent。Agent 发现 Collector 节点故障后，不断检查 Collector 的存活状态以便重新发送 Event，在此期间产生的数据将缓存到本地磁盘中。
- `agentBESink(["machine"[,port]])`: 不负责的 Agent。如果 Collector 出现故障，将不作任何处理，它发送的数据也将被直接丢弃。
- `agentE2EChain`: 指定多个 Collector，以提高可用性。当向主 Collector 发送 Event 失效后，将转向第二个 Collector 发送；当所有的 Collector 都失效后，它还会再发送一遍。



单击此处添加标题



collector

将多个Agent的数据汇总后加载到storage中，source和sink与Agent类似



sink

- collectorSink, 数据通过collector汇聚后发送到hdfs
- customdfs自定义格式dfs



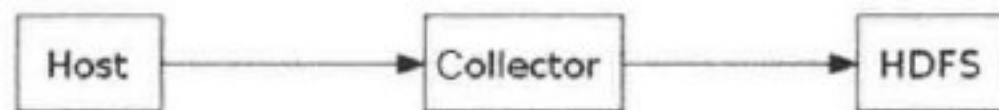
source

- collectorSource[5547]监听端口汇聚数据
- autoCollectorSource通过master协调节点汇聚数据
- logicalSource:由Master分配端口并监听rpcSink



Master

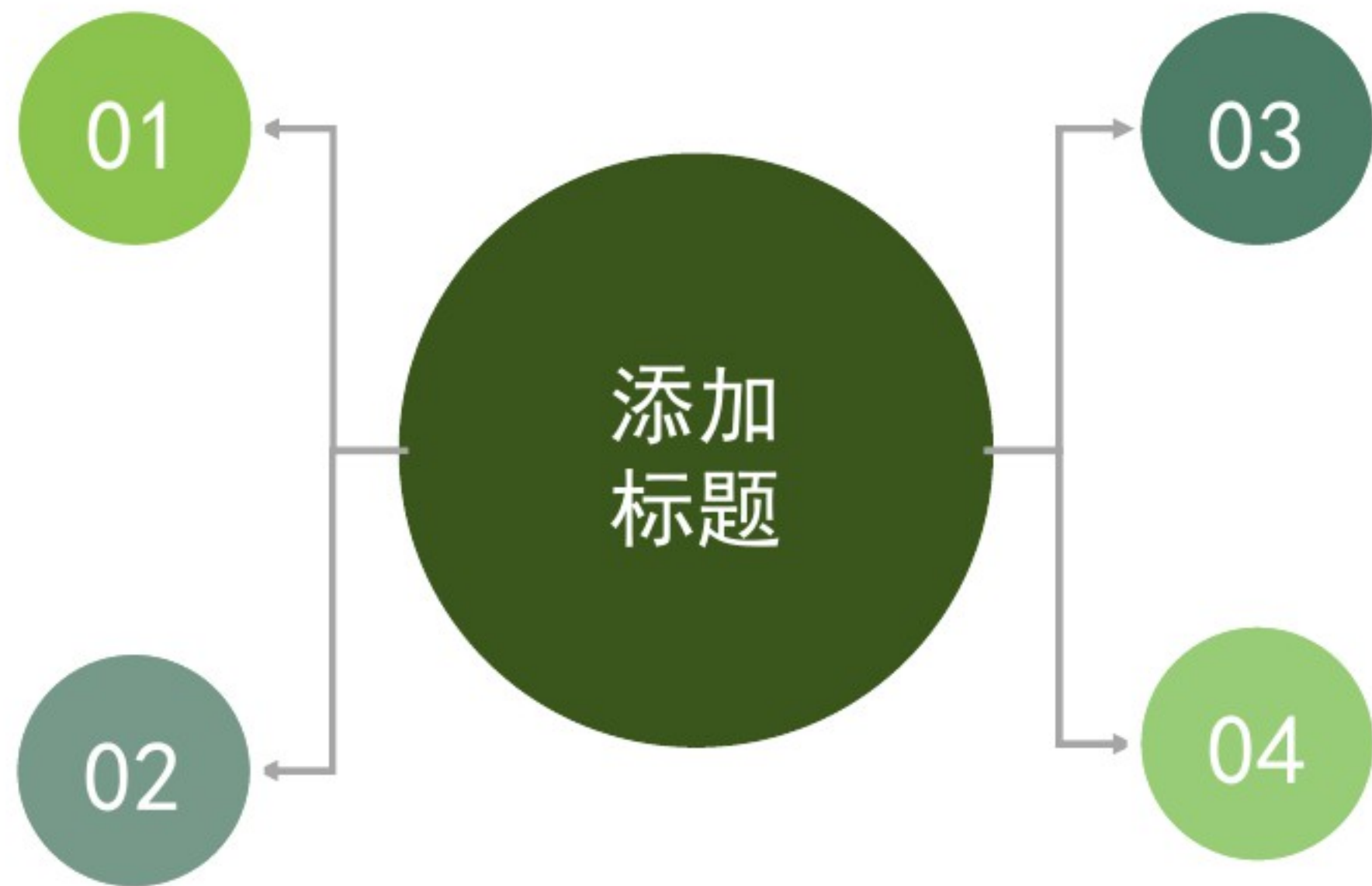
Agent数据流从哪里得到数据就把数据发到哪个collector
collector接收Agent发来的数据发送到目标机器



Agent tier $\xrightarrow{35872}$ Collector tier $\xrightarrow{\times\times\times}$ Storage tier

组件介绍

Flume对Hadoop和zk的依赖只存在于jar包上



默认rpc协议是avro, nettyavrorpcclient和thriftrpcclient分别实现了rpcclient接口

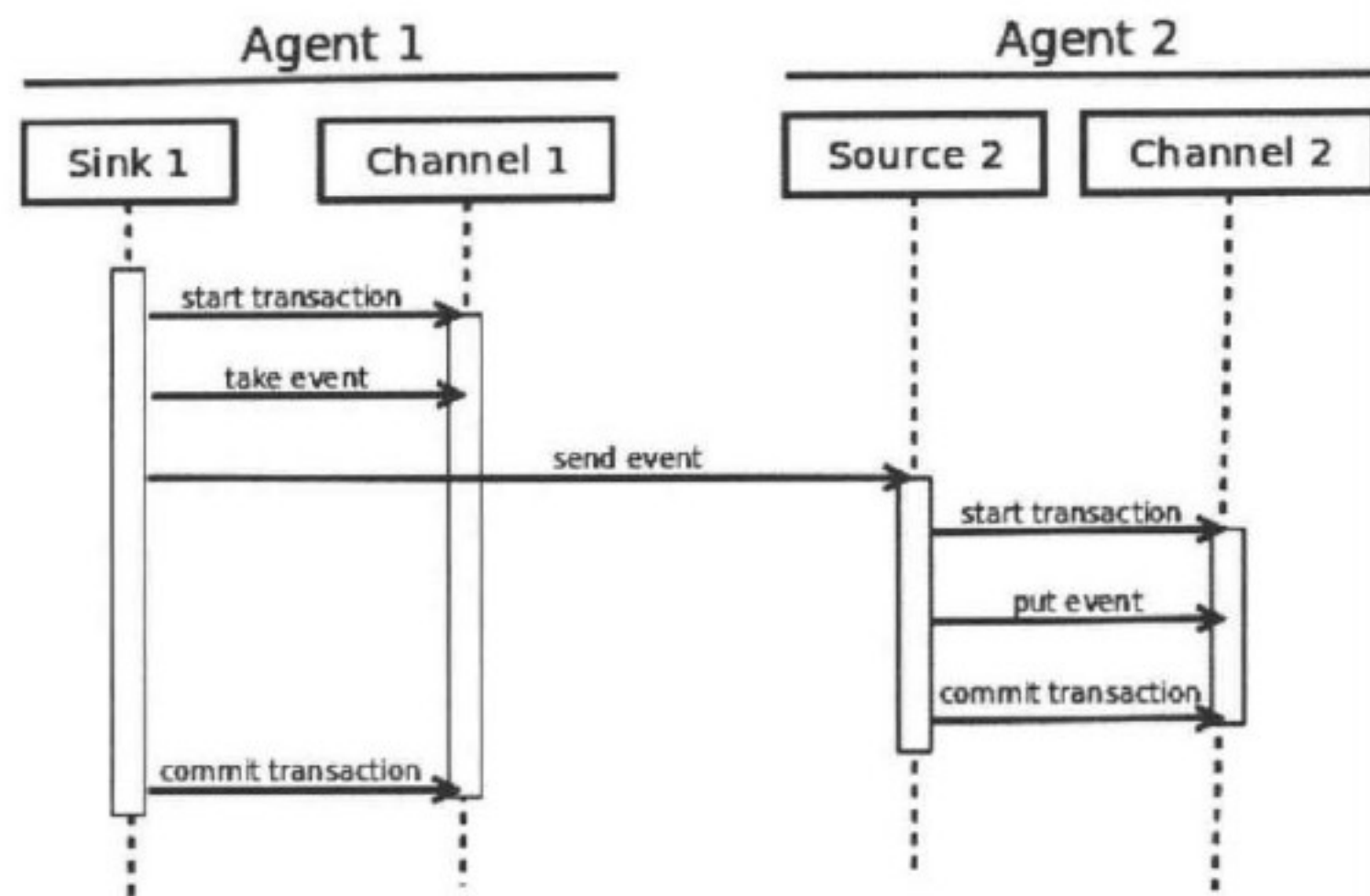
client sdk是一个基于rpc的sdk库, 可以通过rpc使应用与Flume直接连接, 直接调用sdk的api不用关心底层数据交互

FailoverRpcClient实现了主备切换, 采用host:port形式, 一旦当前连接失败就自动寻找下一个

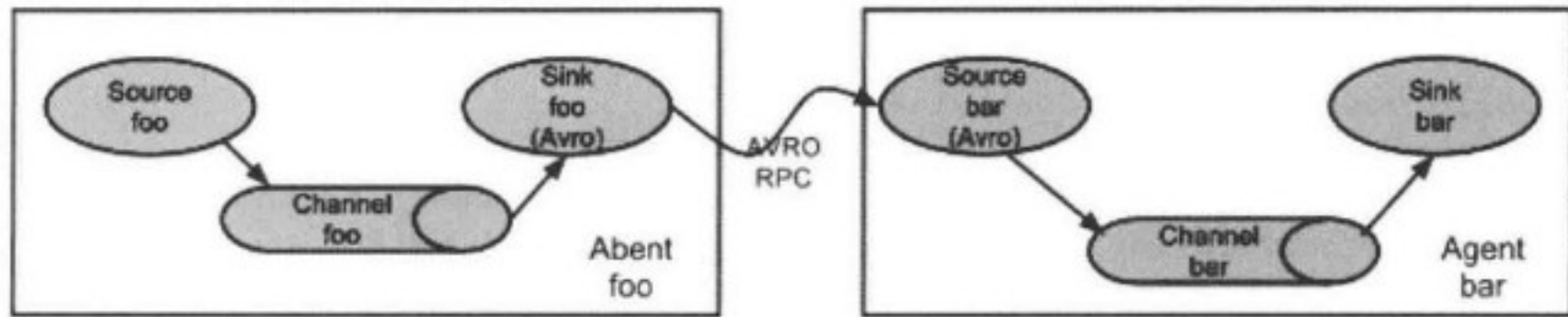
LoadBalancingRpcClient在有多个host的时候负载均衡

内嵌的agent不支持所有的source sink channel

source、sink、channel都要使用transaction进行消息收发，在channel类实现transaction接口，source和sink只要连上channel就必须先获取transaction对象

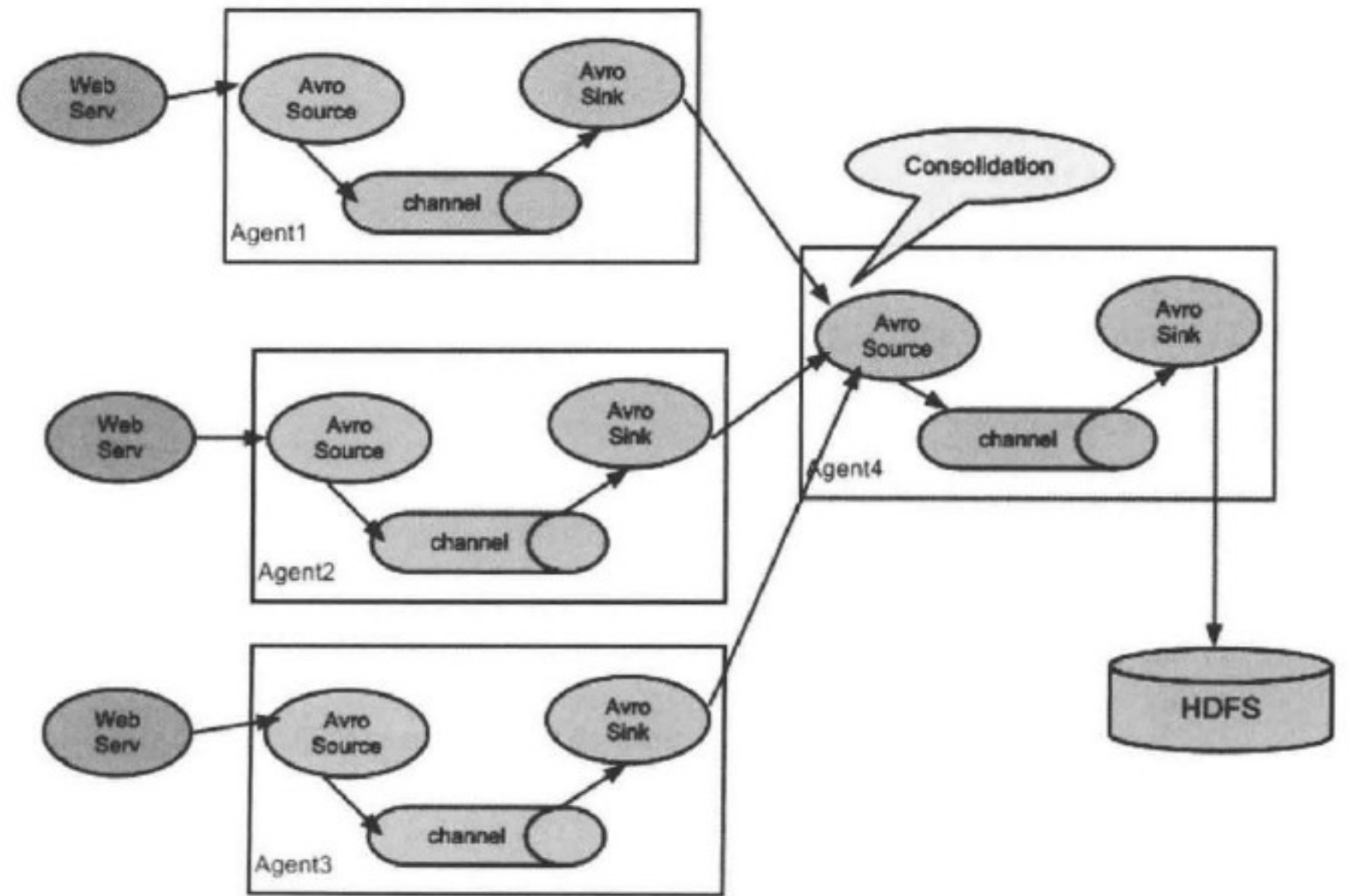
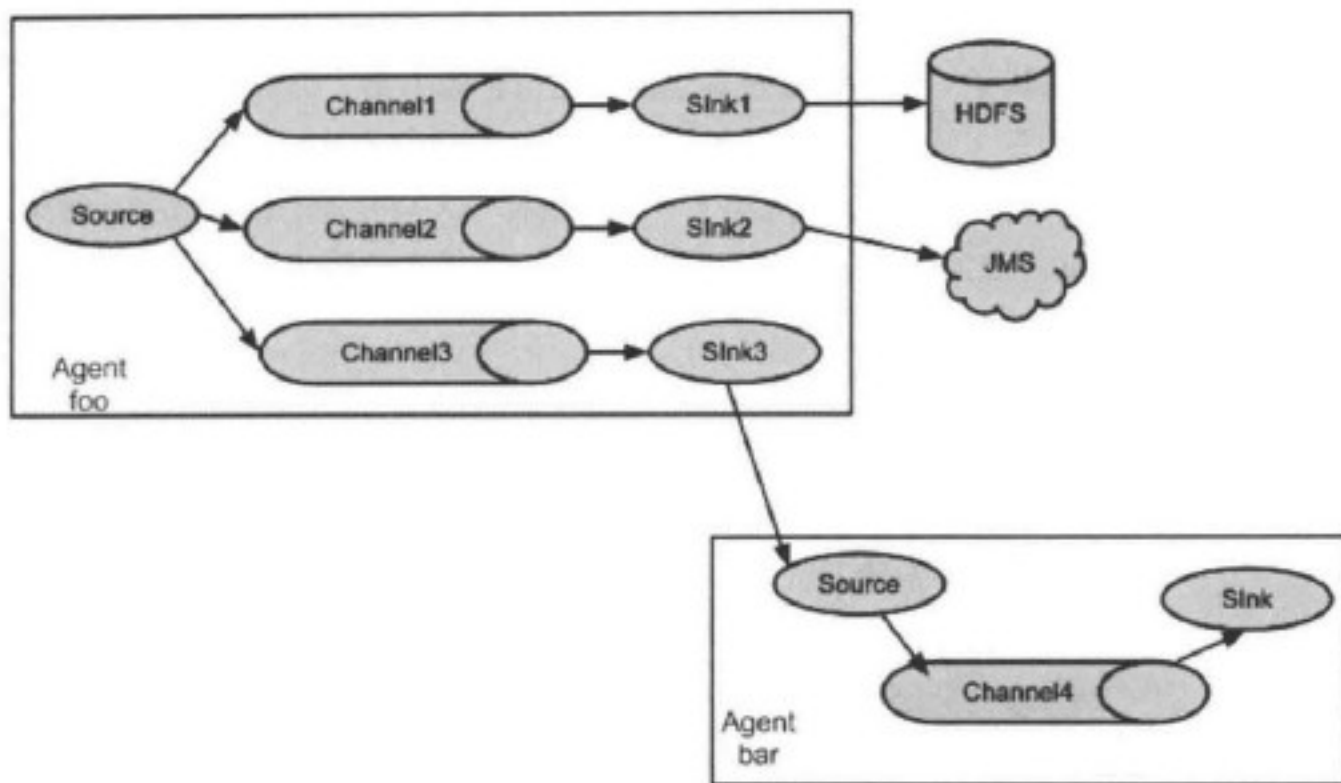


Fume使用模式



多Agent串联

单source的各种处理



多agent合并

Scribe 和 Chukwa 也是不错的日志收集组件。

- Scribe 主页: <https://github.com/facebook/scribe>。
- Chukwa 主页: <http://incubator.apache.org/chukwa/>。



谢谢观看

THANK YOU