



流计算大数据技术在运营商实时信令处理中的应用

董 斌¹, 杨 迪¹, 王 铮¹, 周文红²

(1. 中国电信股份有限公司上海研究院 上海 200122; 2. 中国电信集团公司 北京 100032)

摘 要: 基于 Hadoop 搭建的大数据平台采用离线批处理的方式, 无法满足对数据实效性敏感的业务要求。针对运营商动态数据信息开放大数据平台的实时信令处理要求, 对流式计算大数据组件进行了分析, 介绍了与流计算大数据相关的实时采集、汇聚和处理组件, 形成了端到端实时信令处理大数据技术解决方案, 并提出了融合批处理和实时计算的大数据平台解决方案, 提高了网络信令数据的时效性, 为业务创新提供更大空间, 带来更多利益。

关键词: 大数据; 流式计算; Storm; 实时信令数据处理

doi: 10.11959/j.issn.1000-0801.2015212

Steam Computing Analysis and Application on Operator Real-time Signal Big Data Processing

Dong Bin¹, Yang Di¹, Wang Zheng¹, Zhou Wenhong²

(1. Shanghai Research Institute of China Telecom Co., Ltd., Shanghai 200122, China;

2. China Telecom Corporation, Beijing 100032, China)

Abstract: Big data platform based on the Hadoop structures using offline batch processing mode, unable to meet the service requirements for data timeliness. In order to solve the real-time signal processing requirements in operator OIDD(open information of dynamic data)big data platform, the big-data system components of stream computing were discussed, real-time data collecting, distribution and processing components which formed the end-to-end real-time signal processing big data technology solutions were recommended, and the solutions of the OIDD big data platform which integrated batch computing and stream computing were proposed. It will improve the timeliness of the network signaling data and provide more space for business innovation and bring more benefits.

Key words: big data, stream computing, Storm, real-time signal data processing

1 引言

自从 Google 发布了基于云计算的分布式 MapReduce^[1] 大数据处理编程模型, 大数据技术和应用得到了广泛的应用, 开源的 Hadoop 分布式计算软件框架^[2]更是将大数据应

用推向了极限, 网页搜索、RTB(real time bidding)广告、精准营销等典型应用的成功使 Hadoop、MapReduce 成为大数据的象征。MapReduce 是一种离线的批处理方式, 可以成功处理 TB、PB 级海量数据, 但无法应对实时数据分析需求和对消息事件的实时响应, 大数据处理需要支持实时处

收稿日期: 2015-05-12; 修回日期: 2015-08-24

论文引用格式: 董斌, 杨迪, 王铮等. 流计算大数据技术在运营商实时信令处理中的应用. 电信科学, 2015212

Dong B, Yang D, Wang Z, *et al.* Steam computing analysis and application on operator realtime signal big data processing. Telecommunications Science, 2015212

理和迭代计算技术作为补充,因此流式计算成为大数据技术研究的新热点。流式计算来自于一个信念:数据的价值随着时间的流逝而降低,所以事件出现后必须尽快地对它们进行处理,发生一个事件进行一次处理,而不是缓存起来成批处理。

电信运营商是天然的大数据拥有者,丰富的数据资源已成为电信运营商的重要战略资产,而运营商数据也有鲜明特点,其在大数据产业链中处于数据传递和交换中心的地位,拥有的数据一般不是最终落地数据,主要由实时信令和流式数据构成,因此在运营商大数据应用实践过程中,对实时信令数据处理、分析的实时性需求越来越迫切。流式大数据技术的应用为运营商数据信息化提供了新的机遇和工具,并有助于提升数据价值和推进业务创新。

在运营商动态数据信息开放(open information of dynamic data,OIDD)平台的大数据集约运营应用中,实时信令采集、清理汇聚需要在省层面进行,全国层面再进行汇聚和加工处理开放。目前最快获取信令中实时信息(如用户状态、位置信息等)的时延能达到小时量级,这无法满足对数据实时性敏感的数据增值服务要求。以解决 OIDD 所面临的问题为出发点,本文提出了基于流式计算的全国 OIDD 平台的实时信令处理大数据技术解决方案。

2 基于 Hadoop 架构的动态数据信息开放平台

用户信息能力是电信能力的重要组成部分,具有丰富的潜在价值。动态数据信息开放平台是对用户动态信息进行实时采集、汇聚、挖掘和开放的平台,为网络优化和第三方应用提供“快数据”的增值服务。

2.1 OIDD 平台架构

通过网络信令接口可以实时获取到用户状态信息,包括终端能力信息、用户状态信息、用户位置信息、网络状态及 IP 地址和电话号码临时绑定关系等,OIDD 对从网络中实时采集的数据集约汇聚,并经过脱敏处理后,根据数据不同类型和特性封装成核心数据能力,面向第三方应用提供“快数据”能力服务和对网络状态进行实时的监控和优化。

目前,OIDD 平台基于 Hadoop 大数据平台建设,平台技术架构如图 1 所示。

- ETL(extraction transformation loading):负责将分散的、从异构数据源中收集的数据(如信令数据、位置数据、DPI 数据等)抽取到临时中间层后进行清洗、

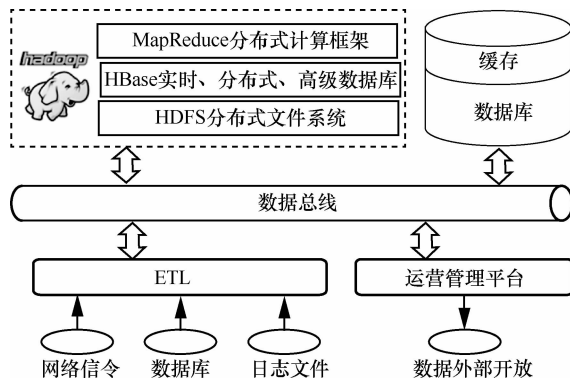


图 1 OIDD 平台架构

转换、集成,最后通过数据总线存入 Hadoop 分布式数据库。

- 运营管理平台(OP):系统数据开放接口的管理平台,提供数据路由、接入鉴权、数据访问权限、数据脱敏等功能。
- 数据总线:系统的数据总线,整合了系统 Hadoop、数据库、缓存等资源池访问、系统消息队列,并提供高速网络访问接口,使各模块可以分开部署,灵活扩容。
- 缓存(cache):OIDD 的高速数据缓存主体为内存数据库,在海量数据环境下它提供了数十倍于物理数据库的数据查询速度。
- 数据库(DB):保存数据挖掘和数据分析的结果性数据、被经常访问的数据以及其他的临时性数据等。

2.2 OIDD 平台多级组网

电信网络按地域部署,信令信息、DPI 信息都需要在省层面采集,并在集团层面进一步汇聚。OIDD 平台组网分为集团和省份 2 个层面,集团平台完成数据统一汇聚,省份部署采集系统,并可根据自身发展,决定是否建设省级 OIDD 平台。

图 2 是数据源采集和汇聚的过程,数据源可以分为 3 种类型:集团集中建设的平台系统(如业务平台、集团 IT 平台等)、已经具备数据集约的省份 OIDD 平台、按省建设但是集团具备统一数据集约的系统(如综合网管、集团信令系统、日志采集系统等)。

2.3 业务应用与发展问题

基于已建成的 OIDD 平台,封装成 API 和独立的服务地址,提供数据的查询、状态订阅和通知的功能。现已开展了用户漫游服务、企业名片挂机推送、行车路线定位监测等服务,并取得良好的经济效益。在平台运营实践中,虽然通过高速数据缓存能提升系统效率,显著降低外部应用对分析结果的查询时延,但由于需要经过层层汇聚和批处理,数据

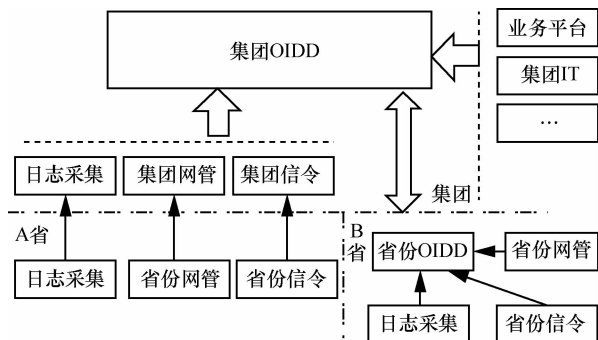


图2 动态信息开放平台的多级组网

时效性最快只能达到小时量级,限制了新业务创新,提升“快数据”实时性价值成为 OIDD 平台的未来发展重点。

根据现有 OIDD 平台架构和组网提高数据时效性,重点要解决以下 2 个方面问题。

- 原有平台基于 Hadoop 大数据处理技术,本质是批处理方式,实时信令数据采集需要积累到一定量或时间后再统一处理。平台要进一步支持对实时、单独的消息和事件进行处理,并且这个过程是消息/事件驱动和不间断的。
- 全国 OIDD 平台数据需要通过省、集团两个层面的汇聚,也是导致数据实时性无法保证的重要原因,有的在省公司可以开展的业务,由于数据时效性已过,集团层面难以开展。全国 OIDD 平台需要及时获得实时信令信息,尽量消除批量存储再转发的中间环节。

3 流式大数据处理技术及应用分析

流式数据处理系统和批量数据处理系统有着本质的差别,流式数据处理系统需要维护消息队列并进行实时消息的及时处理。分布式流式大数据处理技术虽然处于起步发展阶段,但由于市场广泛需求的驱动,成为关注和研究热点^[3]。当前具有代表性的流式处理系统有 Storm^[4]、S4^[5]以及 Spark Streaming^[6]。

流式计算组件是实时大数据技术平台的核心,但是几种技术的比较不是本文重点,本文目标是选择一款便于引入 OIDD 平台的实时大数据处理技术。Storm 主从方式比 S4 去中心方式更适合消息处理(保证消息顺序性)^[2],因此 Storm 越来越得到广泛关注和应用,如阿里巴巴、百度实时数据处理都采用了 Storm 架构。Spark Streaming 也是当前热点,其原理是将数据流分成小的时间片断(秒级),以类似批处理的方式处理数据,对于目前版本的 Spark Streaming 而言,其最小的批处理时间间隔选取为 0.5~2 s^[6],所以 Spark

Streaming 能够满足除了对实时性要求非常高之外的所有流式准实时计算场景。但相比 Storm 系统,Spark Streaming 计算时延大,Storm 目前最小的时延是毫秒级^[3]。基于上述分析,本文针对性地对如何将 Storm 引入 OIDD 平台的解决方案进行了分析。

大数据处理架构包括数据采集、数据接入、数据处理和数据输出,如图 3 所示。流式大数据处理也要选择和配置满足实时信令处理要求的数据采集、汇聚、处理和分析相关组件。



图3 大数据处理流程

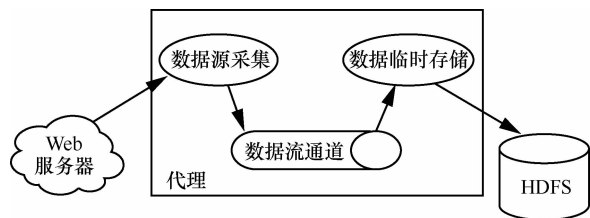
通过 Flume-ng+Kafka+Storm 可以搭建实现支持实时信令处理的大数据分析平台。Flume-ng 负责从各节点上实时采集数据;由于数据采集的速度和数据处理的速度不一定同步,因此需要一个消息中间件作为缓冲,Kafka 作为一种高吞吐量的分布式消息系统非常适合承担此项工作;流式数据处理作为关键组件,根据前述分析选择了 Storm;数据输出可以用 MySQL 和自定义 API (application programming interface,应用程序编程接口)。

3.1 数据收集服务组件 Flume-ng

Flume 目前是一个分布式、高可用、可扩展的海量信息收集工具,可以实时进行数据的收集和传输,Apache Flume 项目将 Flume1.0 以后的版本统称为 Flume-ng^[7]。为了简洁,后续涉及的 Flume 组件都是指 Flume-ng 版本。

Flume 架构如图 4 所示,以代理(agent)为最小的独立运行单位,由数据源采集(source)、数据临时存储(sink)和数据流通道(channel)3 层组成,一个 agent 就是一个 JVM(Java virtual machine,Java 虚拟机)。数据流的采集由事件(event)贯穿始终,事件是 Flume 的基本数据单位,运营商网络中信令消息和日志记录都可以看成一个个事件。

- source 负责接收事件,并进行简单处理后,写到定制的各种数据接收方。通过编程,可针对不同数据源或数据类别对 source 进行定制。
- channel 位于 source 和 sink 之间,用于缓存接收进来的事件。
- sink 负责取出 channel 中事件,传输到下一跳或最终目的,sink 也是可以通过编程自定义的。当 sink 成功地将事件发送到下一跳或最终目的地,事件从 channel 移除。

图 4 Apache Flume 架构^[7]

Flume 是分布式的,每一层均可以水平扩展,具有端到端的数据传送保障,非常适合实时性高、协议需要定制分析的网络信令的实时采集。

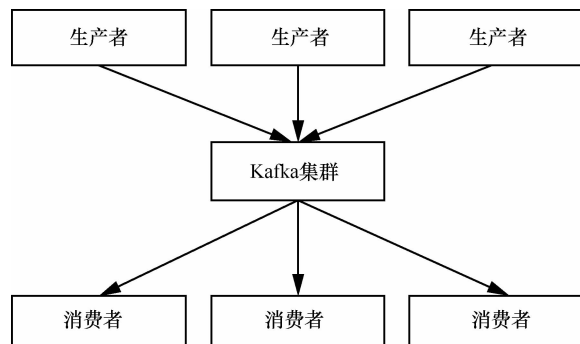
3.2 分布式消息队列组件 Kafka

Kafka 也是 Apache 下的开源消息系统项目^[8],是一种高吞吐量的分布式消息发布订阅系统,在普通的服务器上每秒也能处理几十万条消息,可用于低时延的收集和发送大量的事件和日志数据。

Kafka 架构如图 5 所示,它维护按照类别进行区分的信息^[8]。一个典型的 Kafka 集群包含若干生产者(producer)、处理服务器(broker)、消费者(consumer)以及一个 ZooKeeper 集群^[9]。producer 就是向 Kafka 发消息的客户端,consumer 则是从 Kafka 取消息的客户端,一台 Kafka 服务器就是一个 broker,负责消息的处理分发,ZooKeeper 管理 broker 与 consumer 的动态加入与离开,各组件都可以水平扩展。

Kafka 具有消息缓存能力,Kafka 集群可以在一个可配置的时间内保存所有发布上来的消息,不管这些消息有没有被消费。

在网络实时信令处理中,需要一个消息队列系统来缓冲实时信令采集客户端发送过来的消息,并且要求这个消息队列系统支持良好的扩展性和大规模的数据流,同时为了和下个环节的数据处理速度匹配。Kafka 组件非常适合承担这项工作,根据配置的订阅规则将缓存的消息转发到消息使用者,从而降低实时信令数据处理系统的复杂性。

图 5 Apache Kafka 架构^[8]

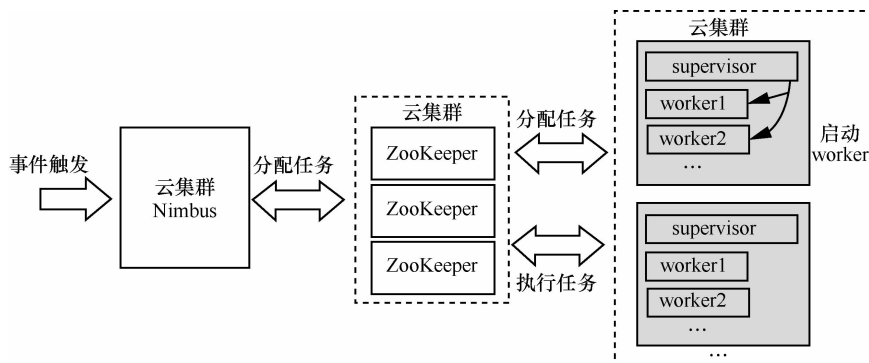
Flume-ng+Kafka+Storm 架构中,Flume Sink 作为 Kafka 的生产者,将消息事件传送到 Kafka 集群中,按照消息类别(例如按发布者区分)进行缓存,Kafka 根据配置的订阅规则转发到消费者客户端 Storm spout(参见第 3.3 节的 Storm 介绍)。

3.3 实时流计算组件 Storm

Hadoop 是基于批量的数据处理,需要等待数据的收集和任务调度,因此是离线的处理方式,通常的时间跨度在数十分钟到数小时。与 Hadoop 不同,Storm 是基于流式的数据处理,可以持续处理到达的数据,并且是基于内存级的计算,从而让处理进行得更加实时,处理时延在几十毫秒到几百毫秒量级。

Storm 也是一个分布式、高容错的实时计算系统,计算在多个线程、进程和服务器之间并行进行,节点可以方便地水平扩展^[4]。根据测试,单个节点服务器大约每秒可处理几万条消息或日志。

如图 6 所示,Storm 集群主要由一个主节点和一群工作节点组成,并通过 ZooKeeper 进行协调。主节点运行 nimbus 进程,负责任务调度和资源分配。每个工作节点都运行 supervisor 进程,supervisor 负责接受 nimbus 分配的任务,启动和停止属于自己管理的 worker 进程,nimbus 和 supervisor

图 6 Apache Storm 架构^[4]



不存在直接通信,两者的协调都是由 ZooKeeper 来完成的,任务状态和心跳信息等也都保存在 ZooKeeper 上。

拓扑是 Storm 中最关键的一个抽象概念,相当于一个实时数据流的处理逻辑,可以被提交到 Storm 集群任务中执行,图 6 的 worker 中执行的任务就是一个个拓扑。Storm 中的拓扑如图 7 所示^[4]。拓扑的基本元素是 spout 和 bolt。spout 是一个拓扑的数据源,通常情况下 spout 会从外部数据源中读取数据,然后转换为拓扑内部的源数据;bolt 是实时流数据处理逻辑的执行进程,用户所希望的对数据流的分析和处理操作在这里执行,复杂的消息流处理可能需要经过多个步骤,即经过多步 bolt。

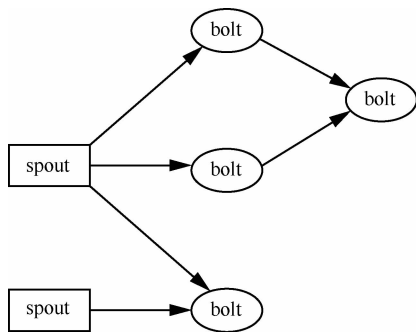


图 7 Storm 的流计算拓扑^[4]

实时信令数据的处理逻辑预先加载到 Storm 集群, Storm 根据从 Kafka 接收到的消息,按消息类别触发任务,进行信令数据处理分析,并输出结果。

4 融合实时处理的 OIDD 大数据平台解决方案

现有动态数据信息开放平台 OIDD 是 Hadoop 大数据处理平台,基于批处理方式。从第 2 节的分析可知,OIDD 平台亟需解决数据的时效性问题,通过引入流式计算大数据技术,提高数据处理实时性能力,同时要考虑同现有平台的融合方案。

4.1 实时信令处理大数据技术解决方案

实时信令处理大数据技术解决方案包括数据实时采集、流式的数据处理以及消息的调度和缓存,需要关注的关键技术包括组件的选取和部署、数据的处理时延和吞吐量以及与现有平台的融合。

根据第 3 节分析,实时采集采用 Flume 组件,流式数据处理采用 Storm,Kafka 作为消息中间件进行消息的缓存和调度。

(1) 相关组件关键技术及部署方案

信令实时采集组件 Flume:Flume 负责网络信令的采集以及 DPI 日志采集。通信网络信令是实时的消息接口,不同设备的接口消息协议也有差别,DPI 采集的则是一条条使用记录。针对不同的数据源,需要通过开发不同的定制化 Flume source 程序,Flume source 组件即嵌入网络设备,也可以部署在云资源池,通过网络接口连接。Flume agent 的 channel 集中部署在云资源池,需要同时处理不同数据源数据,应该采用内存处理模式,提高处理性能。Flume 部署则最好靠近数据源,因此建议在省层面云资源池部署 Flume 集群,通过管理平台为不同的数据源加载对应的 Flume source 程序,根据数据吞吐量弹性配置计算、存储资源。

分布式消息中间件 Kafka:Kafka 整体架构简单,部署方便,有内置的分区,这让 Kafka 成为了一个很好的大规模消息处理应用的解决方案。在实时信令的大数据处理中,Kafka 主要是作为消息缓存中间件,保证数据采集和数据分析处理的消息同步,Flume sink 作为 Kafka 的生产者,而 Storm spout 作为 Kafka 的消费者,Kafka 将从 Flume sink 接收到的数据缓存到不同分区,Kafka broker 对接收到的数据进行持久化处理,保存到存储服务器,基于订阅机制,将缓存的消息按需发送到不同的大数据处理平台。Kafka 集群也部署在省层面,同时为省 OIDD 平台、集团 OIDD 平台提供服务。

Storm 平台:Storm 平台是实时信令处理的大数据平台关键组件,通过 spout 从 Kafka 消息队列中读取数据,发送到相应的 bolt 中进行处理,bolt 则按业务需求配置对应的规则策略,可以针对不同数据源划分不同 bolt 类,根据吞吐量在云资源池分配相应的资源。Storm 还需要同集团/省的动态数据信息开放平台进行整合,与 Hadoop 平台实现资源共享,平台整合的解决方案在第 4.2 节中描述。

(2) 关键流程描述

图 8 是实时信令大数据技术方案中的主要环节和处理流程。

- 数据采集:通过 Flume 集群上部署的各类数据源采集接口机,采集 OIDD 所需的数据源信息,并进行数据格式转化,传送到 Kafka 消息队列。
- 建立消息队列:Kafka 将接收到的数据,按不同数据源建立多条分布式消息队列并将数据缓存,根据不同 OIDD 大数据平台对数据的订阅规则,传送给相

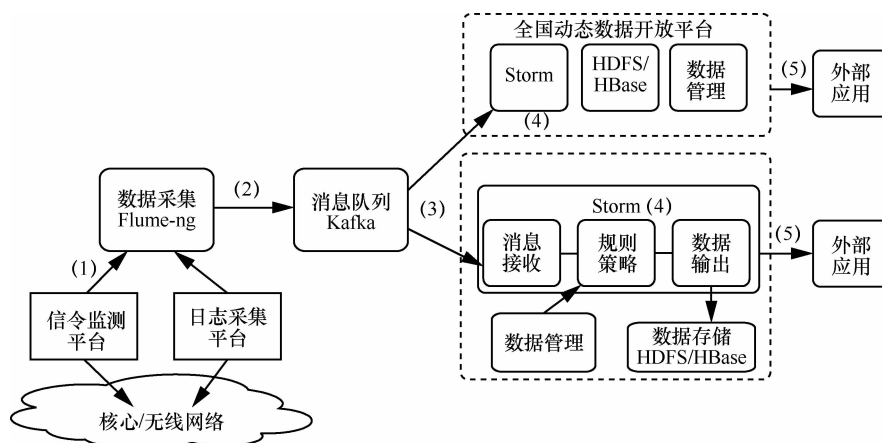


图 8 实时信令处理大数据技术方案

应的 OIDD 平台分别处理,省 OIDD 平台和集团 OIDD 平台的数据源可以不一致,消费者可以是 OIDD 的 Storm 平台,也可以是原来的 Hadoop 平台。

- 消息接收:不同数据源具有不同的数据分析处理逻辑,对于实时信令由 Storm 平台处理,Storm 的 spout 接收消息或文件内容信息,并送到对应的 bolt 处理。
- 规则匹配:由 Storm 与关联规则筛选符合规则的数据内容,这些规则可以是对原有数据的更新,也可根据外部应用对数据的订阅需求转化而来,然后输出结果。
- 数据输出:数据发送到数据库存储或者根据外部应用的订阅需求触发通知、查询等事件。

4.2 与 Hadoop 平台融合解决方案

Hadoop 2.0 平台中引入了新的资源管理系统 YARN^[10], MapReduce、Storm 和 Spark 等组件均可运行在 YARN 之上,这就为 Storm 与 Hadoop 大数据平台整合提供了解决方案。

将 Storm 运行在 Hadoop YARN 上,Storm 与原有 Hadoop 平台可共享整个集群中的资源,并实现了数据的复用,避免了在多个数据平台上保存同样的数据副本,从而节省了存储资源和跨群复制数据导致的网络开销,也避免了多个集群带来的维护成本。

OIDD 平台汇聚运营商网络中多种数据源,实时信令也是 OIDD 大数据平台需要采集的数据源,基于 YARN 架构,可以将负责实时信令处理的 Storm 大数据平台组件整合到原有 Hadoop 大数据平台中,Storm 平台输出的数据仍然入库到已经部署的 HDFS/HBase 数据库中,从而实现云资源、大数据组件和数据的共享。整合后的大数据平台,原有平台中的数据管理、安全控制、服务封装和业务生命周期管理等模块可以基本保持不变,但 Hadoop 平台的版本需要升级到 YARN,并提供对 Storm 模块的数据管理、规则配置的接口。融合的大数据平台和组网架构如图 9 所示。

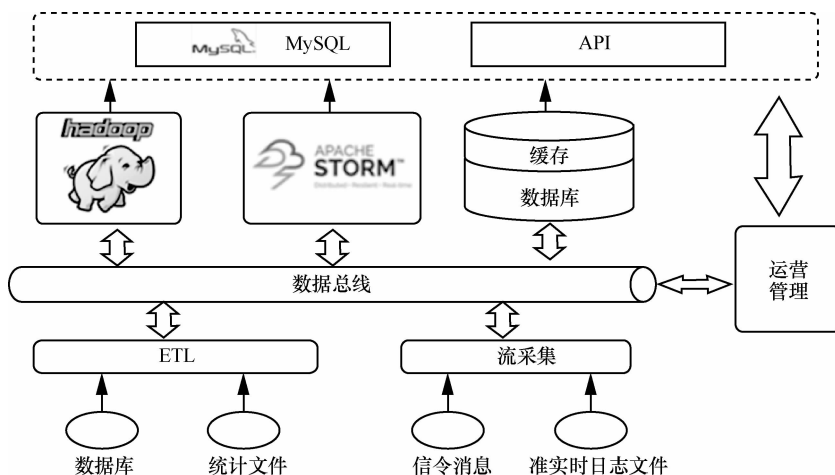


图 9 融合的大数据平台和组网架构



在新的平台组网架构中,数据采集分为 ETL 组件和流采集组件,实时性要求高的数据通过 Flume,由 Storm 流计算模块实时分析处理,满足对数据时效性要求高的数据业务需求;数据总线基于消息中间件调度,实现基于消息发布订阅的任务调度;大数据平台实现了资源共享,采集的源数据整合加工后仍然保存到共享的统一数据库,因此仍然能够满足原来基于 Hadoop 的大数据应用。

5 结束语

大数据流式计算和批量计算适用于不同的应用场景。批处理汇聚海量数据分析出的结果可能更精确,但对数据时效性要求严格而对历史数据积累并不非常关注的场景,流式计算具有明显的优势。批量计算和流式计算是有优劣互补的,因此在多种应用场合下可以将两者结合起来使用。

以 Hadoop 大数据技术为基础的 OIDD 平台已在现网应用,随着业务开展,业务创新对数据的时效性提出了更高要求。结合流式计算的研究,本文提出了融合流式计算和批量计算的 OIDD 解决方案,对所需的组件进行了分析和选择,从实验室搭建 Flume-ng+Kafka+Storm 大数据处理环境,10 万条数据可在几秒内完成采集、汇聚、处理端到端过程,大大提高了数据有效性。

大数据流式计算的研究和应用仍处于起步和尝试阶段,为了促进大数据流式计算平台的成熟,还需要在实践中进行试点和完善。

参考文献

- 1 Condi T, Alvaro P, Hellerstein J M, *et al.* MapReduce Online. UCB/EECS-2009-136, 2009
- 2 Apache Software Foundation. Welcome to Apache Hadoop. <http://hadoop.apache.org>, 2015
- 3 孙大为, 张广艳, 郑纬民. 大数据流式计算: 关键技术及系统实例. 软件学报, 2014, 25(4): 839~862
Sun D W, Zhang G Y, Zheng W M. Big data stream computing: technologies and instances. Journal of Software, 2014, 25 (4): 839~862
- 4 Neumeyer L, Robbins B, Nair A, *et al.* S4: distributed stream computing platform. Proceedings of the IEEE International Conference on Data Mining Workshops, Sydney, Australia, 2010
- 5 崔星灿, 禹晓辉, 刘洋等. 分布式流处理技术综述. 计算机研究与发展, 2015, 52(2): 318~332
Cui X C, Yu X H, Liu Y, *et al.* Distributed stream processing: a

survey. Journal of Computer Research and Development, 2015, 52(2): 318~332

- 6 Apache Software Foundation. Spark Streaming makes it easy to build scalable fault-tolerant streaming applications. <http://spark.apache.org/streaming/>, 2015
- 7 Apache Software Foundation. Apache Kafka: a high-throughput distributed messaging system. <http://kafka.apache.org/>, 2015
- 8 Apache Software Foundation. Storm official website. <http://storm.apache.org/>, 2015
- 9 Apache Software Foundation. Welcome to Apache ZooKeeper. <http://zookeeper.apache.org/>, 2015
- 10 Vavilapali V K, Murthy A C, Douglas C, *et al.* Apache Hadoop YARN: yet another resource negotiator. Proceedings of the 4th ACM Symposium on Cloud Computing, Santa Clara, California, USA, 2013

[作者简介]



董斌,男,博士,中国电信股份有限公司上海研究院高级工程师,主要研究方向为移动互联网平台技术及应用、下一代网络技术。



王铮,男,中国电信股份有限公司上海研究院工程师,主要研究方向为移动互联网、大数据等。



杨迪,男,中国电信股份有限公司上海研究院工程师,主要研究方向为移动互联网平台及应用、大数据应用。



周文红,女,中国电信集团公司网络运行维护事业部工程师,长期从事移动增值业务平台规划、建设和运营工作,在移动定位业务运营方面有深厚的研究和积累。