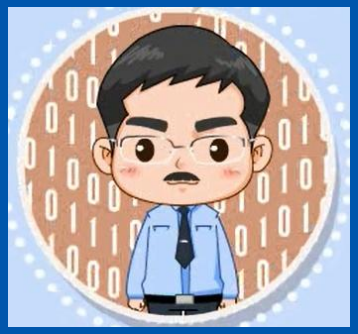


《Flink编程基础（Scala版）》

教材官网： http://dblab.xmu.edu.cn/post/flink/

温馨提示：编辑幻灯片母版，可以修改每页PPT的厦大校徽和底部文字

第3章 Flink的设计与运行原理

（PPT版本号：2021年3月版本）

扫一扫访问教材官网

林子雨

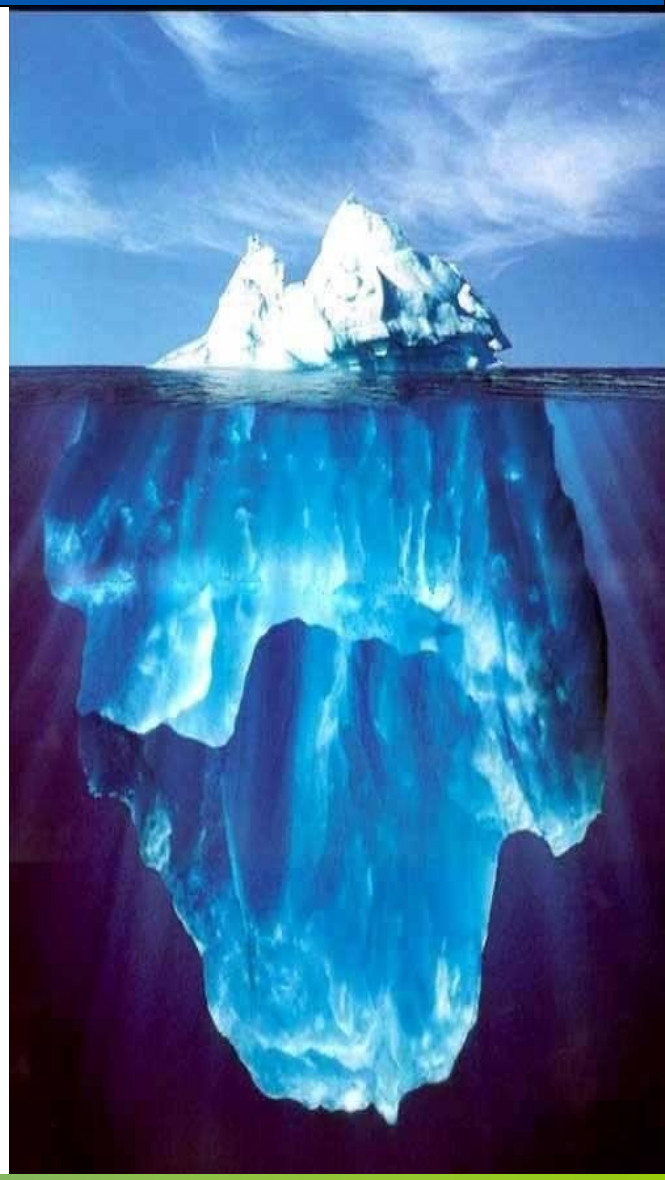
厦门大学计算机科学系

E-mail: ziyulin@xmu.edu.cn

主页：http://dblab.xmu.edu.cn/linziyu

提纲

3.1 Flink简介
3.2 为什么选择Flink
3.3 Flink应用场景
3.4 Flink中的统一数据处理
3.5 Flink技术栈
3.6 Flink工作原理
3.7 Flink编程模型
3.8 Flink应用程序结构
3.9 Flink中的数据一致性



3.1.Flink简介

•Flink是Apache软件基金会的一个顶级项目，是为分布式、高性能、随时可用以及准确的流处理应用程序打造的开源流处理框架，并且可以同时支持实时计算和批量计算。
•Flink起源于Stratosphere 项目，该项目是在2010年到2014年间由柏林工业大学、柏林洪堡大学和哈索普拉特纳研究所联合开展的。
•2014年4月，Stratosphere代码被贡献给Apache软件基金会，成为Apache软件基金会孵化器项目。

3.1.Flink简介

•2014年12月，Flink项目成为Apache软件基金会顶级项目。目前，Flink是Apache软件基金会的5个最大的大数据项目之一，在全球范围内拥有350多位开发人员，并在越来越多的企业中得到了应用。
•在国外，优步、网飞、微软和亚马逊等已经开始使用Flink。
•在国内，包括阿里巴巴、美团、滴滴等在内的知名互联网企业，都已经开始大规模使用Flink作为企业的分布式大数据处理引擎。
•在阿里巴巴，基于Flink搭建的平台于2016年正式上线，并从阿里巴巴的搜索和推荐这两大场景开始实现。目前，阿里巴巴所有的业务，包括阿里巴巴所有子公司都采用了基于Flink搭建的实时计算平台，服务器规模已经达到数万台，这种规模等级在全球范围内也是屈指可数。
•阿里巴巴的Flink平台内部积累起来的状态数据，已经达到PB级别规模，每天在平台上处理的数据量已经超过万亿条，在峰值期间可以承担每秒超过4.72亿次的访问，最典型的应用场景是阿里巴巴“双11”大屏。

3.1.Flink简介

•Flink具有十分强大的功能，可以支持不同类型的应用程序。Flink的主要特性包括：批流一体化、精密的状态管理、事件时间支持以及精确一次的状态一致性保障等。

•Flink 不仅可以运行在包括 YARN、 Mesos、Kubernetes等在内的多种资源管理框架上，还支持在裸机集群上独立部署。在启用高可用选项的情况下，它不存在单点失效问题。

•事实证明，Flink 已经可以扩展到数千核心，其状态可以达到 TB 级别，且仍能保持高吞吐、低延迟的特性。世界各地有很多要求严苛的流处理应用都运行在 Flink 之上。

3.2 为什么选择Flink

3.2.1 传统数据处理架构
3.2.2 大数据Lambda架构
3.2.3 流处理架构
3.2.4 Flink是理想的流计算框架
3.2.5 Flink的优势

3.2.1 传统数据处理架构

传统数据处理架构的一个显著特点就是采用一个中心化的数据库系统，用于存储事务性数据。

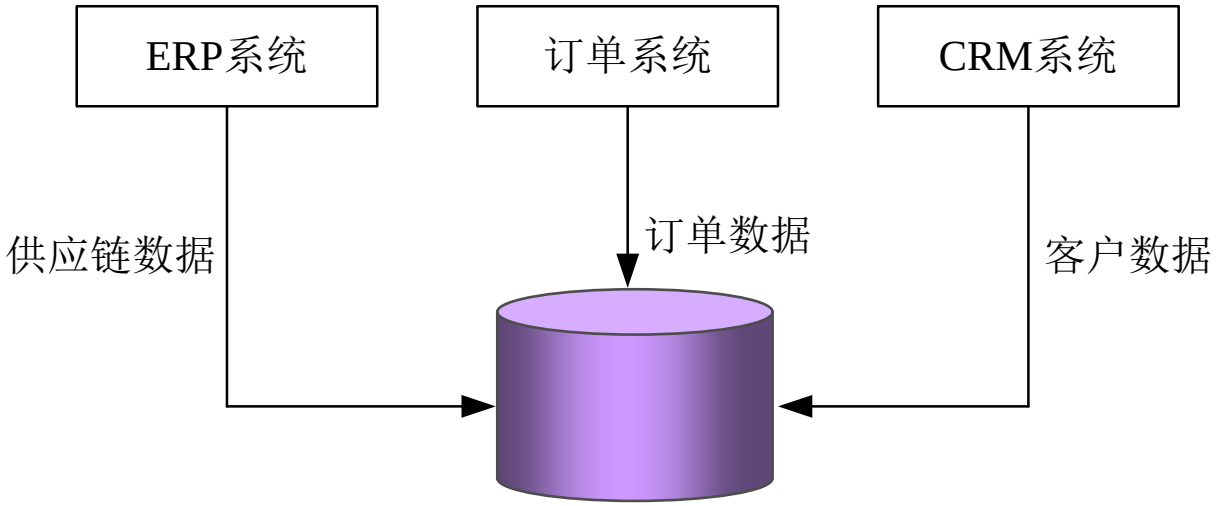


图 传统数据处理架构

3.2.2 大数据Lambda架构

大数据Lambda架构主要包含两层，即批处理层和实时处理层，在批处理层中，采用MapReduce、Spark等技术进行批量数据处理，而在实时处理层中，则采用Storm、Spark Streaming等技术进行数据的实时处理。

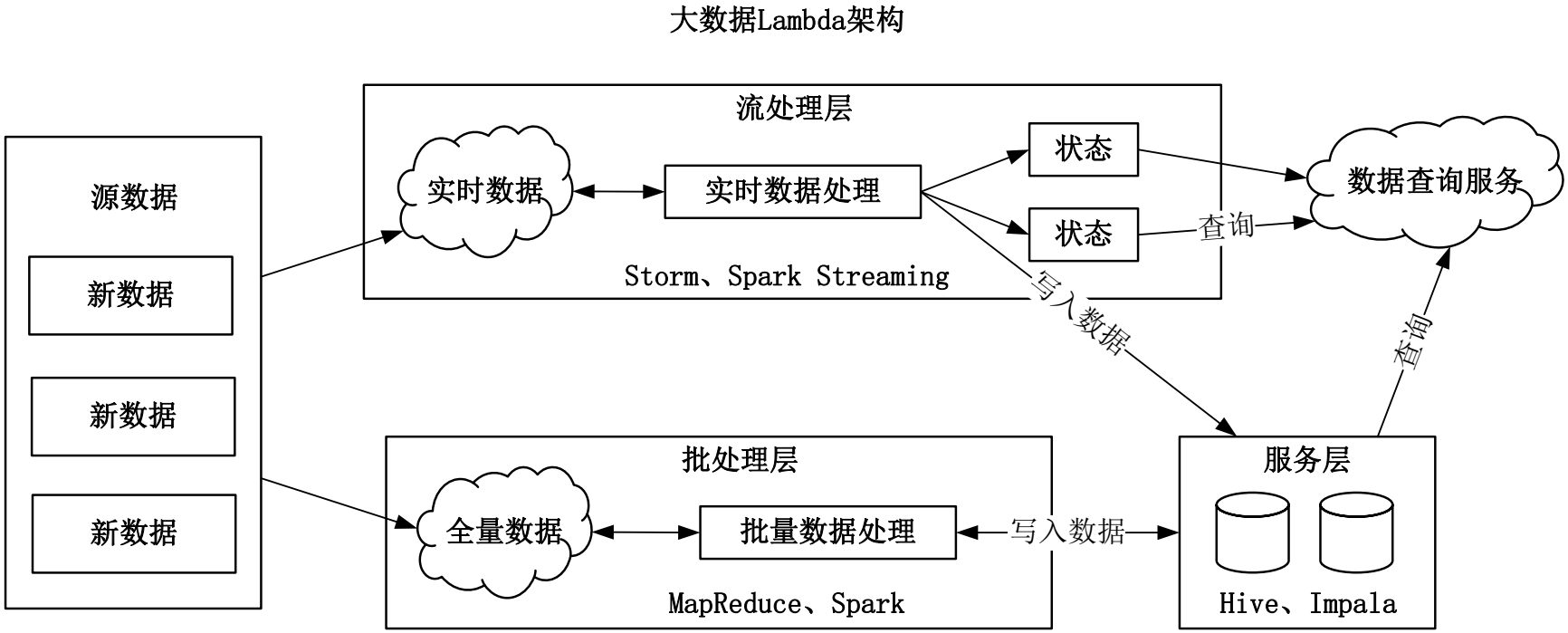


图 大数据Lambda架构

3.2.3流处理架构

为了高效地实现流处理架构，一般需要设置消息传输层和流处理层（如图所示）。消息传输层从各种数据源采集连续事件产生的数据，并传输给订阅了这些数据的应用程序；流处理层会持续地将数据在应用程序和系统间移动，聚合并处理事件，并在本地维持应用程序的状态。

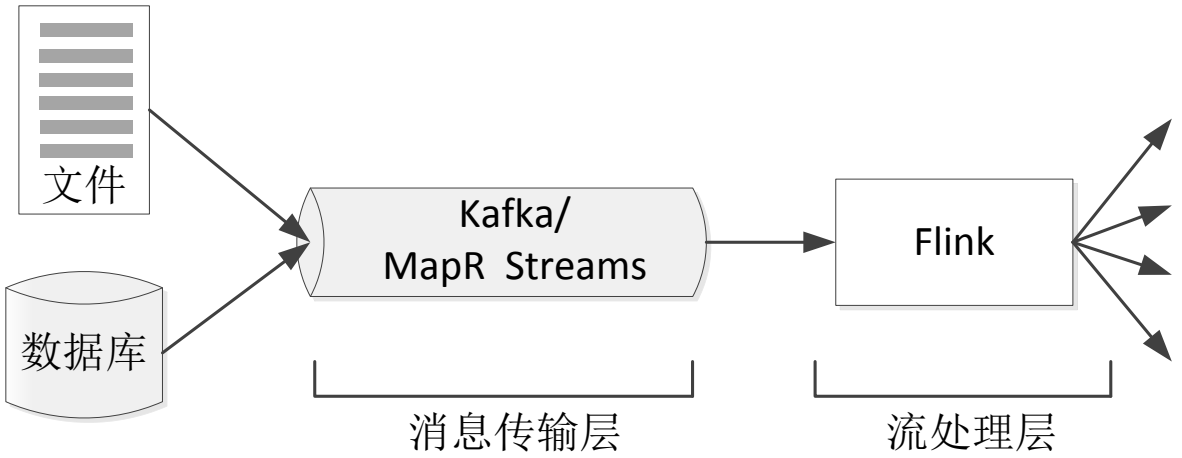


图 流处理架构

3.2.3流处理架构

流处理架构的核心是使各种应用程序互连在一起的消息队列，消息队列连接应用程序，并作为新的共享数据源，这些消息队列取代了从前的大型集中式数据库。如图所示，流处理器从消息队列中订阅数据并加以处理，处理后的数据可以流向另一个消息队列，这样，其他应用程序都可以共享流数据。在一些情况下，处理后的数据会被存放到本地数据库中。

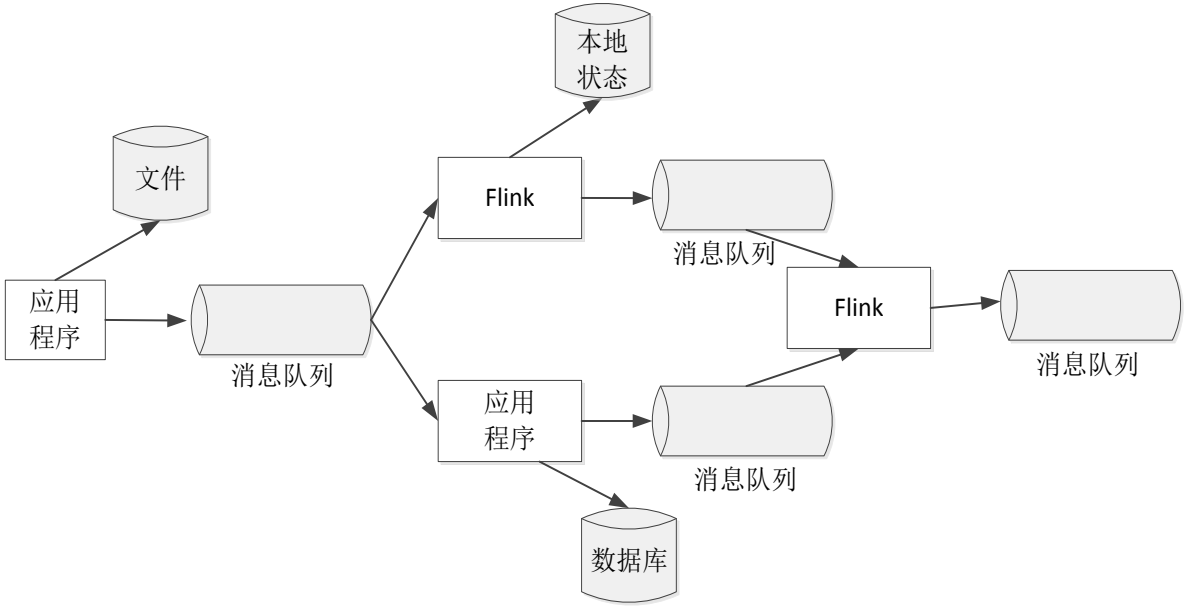


图 流处理架构中的消息队列

3.2.3流处理架构

•流处理架构正在逐步取代传统数据处理架构和Lambda架构，成为大数据处理架构的一种新趋势。
•一方面，由于流处理架构中不存在一个大型集中式数据库，因此，避免了传统数据处理架构中存在的“数据库不堪重负”的问题。
•另一方面，在流处理架构中，批处理被看成是流处理的一个子集，因此，就可以用面向流处理的框架进行批处理，这样就可以用一个流处理框架来统一处理流计算和批量计算，避免了Lambda架构中存在的“多个框架难管理”的问题。

3.2.4 Flink是理想的流计算框架

•流处理架构需要具备低延迟、高吞吐和高性能的特性，而目前从市场上已有的产品来看，只有Flink可以满足要求。
•Storm虽然可以做到低延迟，但是无法实现高吞吐，也不能在故障发生时准确地处理计算状态。
•Spark Streaming通过采用微批处理方法实现了高吞吐和容错性，但是牺牲了低延迟和实时处理能力。
•Spark的另一个流计算组件Structured Streaming，包括微批处理和持续处理两种处理模型。采用微批处理时，最快响应时间需要100毫秒，无法支持毫秒级别响应。采用持续处理模型时，可以支持毫秒级别响应，但是，只能做到“至少一次”的一致性，无法做到“精确一次”的一致性。
•Flink实现了Google Dataflow流计算模型，是一种兼具高吞吐、低延迟和高性能的实时流计算框架，并且同时支持批处理和流处理。此外，Flink支持高度容错的状态管理，防止状态在计算过程中因为系统异常而出现丢失。因此，Flink就成为了能够满足流处理架构要求的理想的流计算框架。

3.2.4 Flink是理想的流计算框架

表3-1 不同流计算框架的对比

产品	消息保证机制	容错机制	状态管理	延时	吞吐量
Storm	至少一次	Acker 机制	无	低	低
Spark Streaming	精确一次	基于RDD的检查点	基于DStream	中	高
Flink	精确一次	检查点	基于操作	低	高

3.2.5 Flink的优势

总体而言，Flink具有以下优势：

（1）同时支持高吞吐、低延迟、高性能

（2）同时支持流处理和批处理

（3）高度灵活的流式窗口

（4）支持有状态计算

（5）具有良好的容错性

（6）具有独立的内存管理

（7）支持迭代和增量迭代

3.3.Flink应用场景

3.3.1 事件驱动型应用
3.3.2 数据分析应用
3.3.3 数据流水线应用

3.3.1 事件驱动型应用

（1）什么是事件驱动型应用

事件驱动型应用是一类具有状态的应用，它从一个或多个事件数据流中读取事件，并根据到来的事件做出反应，包括触发计算、状态更新或其他外部动作等。事件驱动型应用是在传统的应用设计基础上进化而来的。如图所示，在传统的设计中，通常都具有独立的计算和数据存储层，应用会从一个远程的事务数据库中读写数据。而事务驱动型应用是建立在有状态流处理应用的基础之上的。在这种设计中，数据和计算不是相互独立的层，而是放在一起的，应用只需访问本地（内存或磁盘）即可获取数据。系统容错性是通过定期向远程持久化存储写入检查点来实现的。

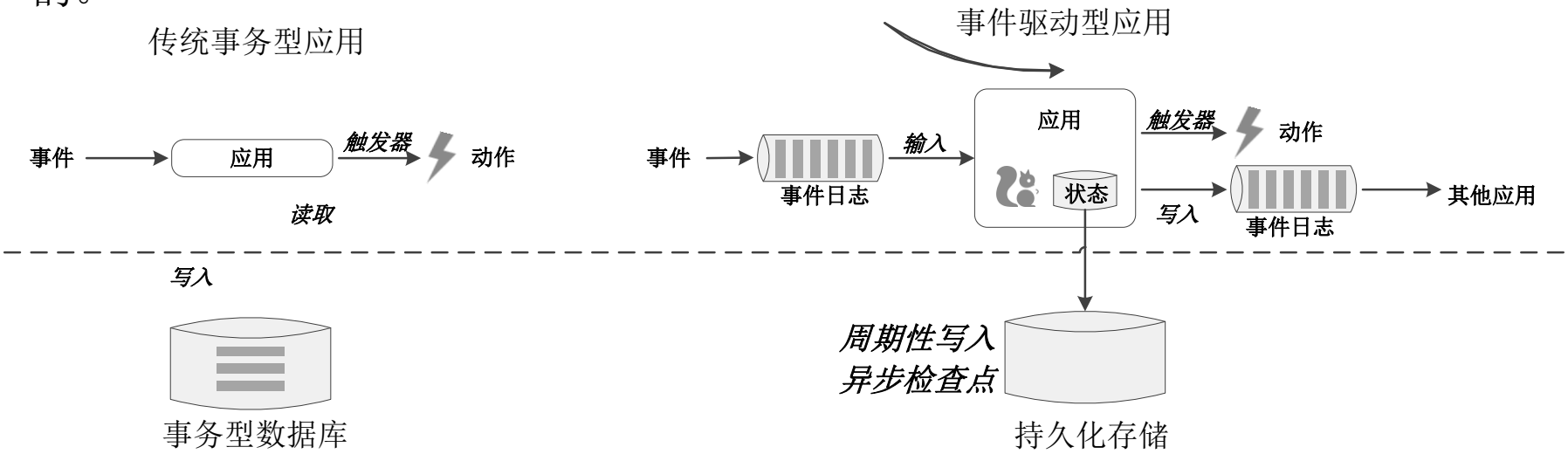


图 传统应用和事件驱动型应用架构的区别

3.3.1 事件驱动型应用

典型的事件驱动型应用包括反欺诈、异常检测、基于规则的报警、业务流程监控、Web 应用（社交网络）等。

（2）事件驱动型应用的优势

事件驱动型应用都是访问本地数据，而无需查询远程的数据库，这样，无论是在吞吐量方面，还是在延迟方面，都可以获得更好的性能。向一个远程的持久化存储周期性地写入检查点，可以采用异步和增量的方式来实现。因此，检查点对于常规的事件处理的影响是很小的。事件驱动型应用的优势不仅限于本地数据访问。在传统的分层架构中，多个应用共享相同的数据库，是一个很常见的现象。因此，数据库的任何变化，比如，由于一个应用的更新或服务的升级而导致的数据布局的变化，都需要谨慎协调。由于每个事件驱动型应用都只需要考虑自身的数据，对数据表示方式的改变或者应用的升级，都只需要很少的协调工作。

3.3.1 事件驱动型应用

（3）Flink是如何支持事件驱动型应用的

•一个流处理器如何能够很好地处理时间和状态，决定了事件驱动型应用的局限性。Flink许多优秀的特性都是围绕这些方面进行设计的。Flink提供了丰富的状态操作原语，它可以管理大量的数据（可以达到TB级别），并且可以确保“精确一次”的一致性。而且，Flink还支持事件时间、高度可定制的窗口逻辑和细粒度的时间控制，这些都可以帮助实现高级的商业逻辑。Flink还拥有一个复杂事件处理（CEP）类库，可以用来检测数据流中的模式。

•Flink中针对事件驱动应用的突出特性当属“保存点”（savepoint）。保存点是一个一致性的状态镜像，它可以作为许多相互兼容的应用的一个初始化点。给定一个保存点以后，就可放心对应用进行升级或扩容，还可以启动多个版本的应用来完成 A/B 测试。

3.3.2 数据分析应用

（1）什么是数据分析应用

分析作业会从原始数据中提取信息，并得到富有洞见的观察。如图所示，传统的数据分析通常先对事件进行记录，然后在这个有界的数据集上执行批量查询。为了把最新的数据融入到查询结果中，就必须把这些最新的数据添加到被分析的数据集中，然后重新运行查询。查询的结果会被写入到一个存储系统中，或者形成报表。

一个高级的流处理引擎，可以支持实时的数据分析。这些流处理引擎并非读取有限的数据集，而是获取实时事件流，并连续产生和更新查询结果。这些结果或者被保存到一个外部数据库中，或者作为内部状态被维护。仪表盘应用可以从这个外部的数据库中读取最新的结果，或者直接查询应用的内部状态。

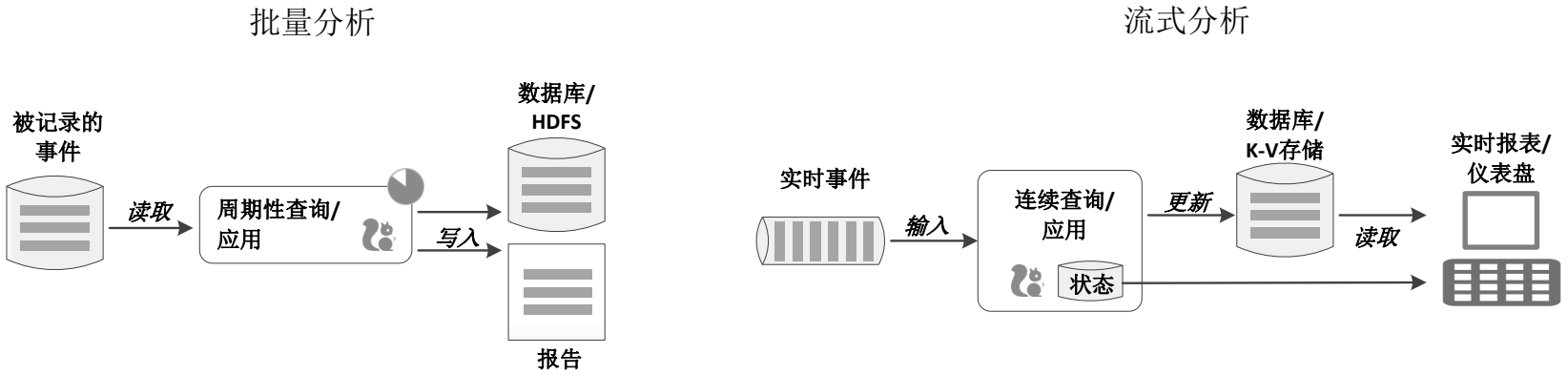


图 Flink同时支持流式及批量分析应用

3.3.2 数据分析应用

典型的数据分析应用包括电信网络质量监控、移动应用中的产品更新及实验评估分析、消费者技术中的实时数据即席分析、大规模图分析等。

（2）流式分析应用的优势

与批量分析相比，连续流式分析的优势是，由于消除了周期性的导入和查询，因而从事件中获取洞察结果的延迟更低。此外，流式查询不需要处理输入数据中的人为产生的边界。

另一方面，流式分析具有更加简单的应用架构。一个批量分析流水线会包含一些独立的组件来周期性地调度数据提取和查询执行。如此复杂的流水线，操作起来并非易事，因为，一个组件的失败就会直接影响到流水线中的其他步骤。相反，运行在一个高级流处理器（比如Flink）之上的流式分析应用，会把从数据提取到连续结果计算的所有步骤都整合起来，因此，它就可以依赖底层引擎提供的故障恢复机制。

3.3.2 数据分析应用

（3）Flink是如何支持数据分析应用的

Flink可以同时支持批处理和流处理。Flink提供了一个符合ANSI规范的SQL接口，它可以为批处理和流处理提供一致的语义。不管是运行在一个静态的数据集上，还是运行在一个实时的数据流上，SQL查询都可以得到相同的结果。Flink还提供了丰富的用户自定义函数，使得用户可以在SQL查询中执行自定义代码。如果需要进一步定制处理逻辑，Flink的DataStream API和DataSet API提供了更加底层的控制。此外，Flink 的Gelly 库为基于批量数据集的大规模高性能图分析提供了算法和构建模块支持。

3.3.3 数据流水线应用

（1）什么是数据流水线

如图所示，Extract-transform-load（ETL）是一个在存储系统之间转换和移动数据的常见方法。通常而言，ETL作业会被周期性地触发，从而把事务型数据库系统中的数据复制到一个分析型数据库或数据仓库中。

数据流水线可以实现和ETL类似的功能，它们可以转换、清洗数据，或者把数据从一个存储系统转移到另一个存储系统中。但是，它们是以一种连续的流模式来执行的，而不是周期性地触发。因此，当数据源中源源不断地生成数据时，数据流水线就可以把数据读取过来，并以较低的延迟转移到目的地。比如，一个数据流水线可以对一个文件系统目录进行监控，一旦发现有新的文件生成，就读取文件内容并写入到事件日志中。再比如，将事件流物化到数据库或增量构建和优化查询索引。

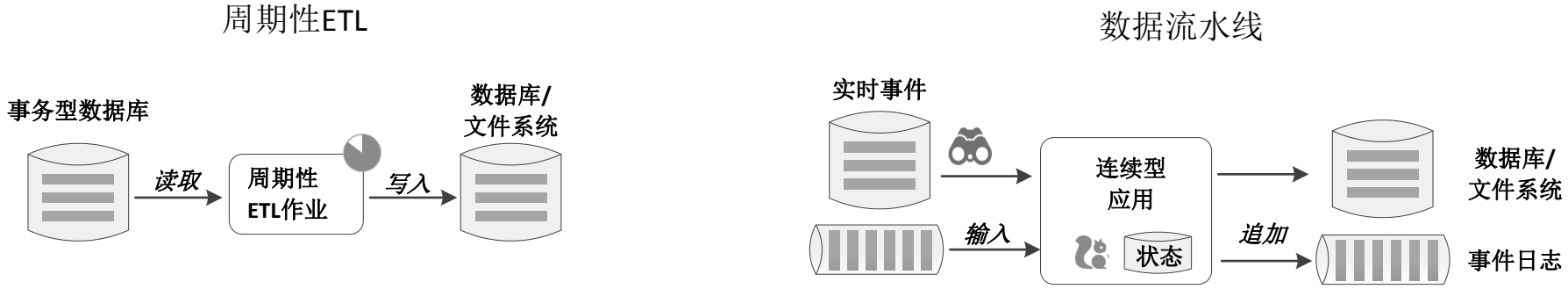


图 周期性 ETL 作业和持续数据流水线的差异

3.3.3 数据流水线应用

典型的数据流水线应用包括电子商务中的实时查询索引构建、电子商务中的持续ETL等。

（2）数据流水线的优势

相对于周期性的ETL作业而言，连续的数据流水线的优势是，减少了数据转移过程的延迟。此外，由于它能够持续消费和发送数据，因此用途更广，支持用例更多。

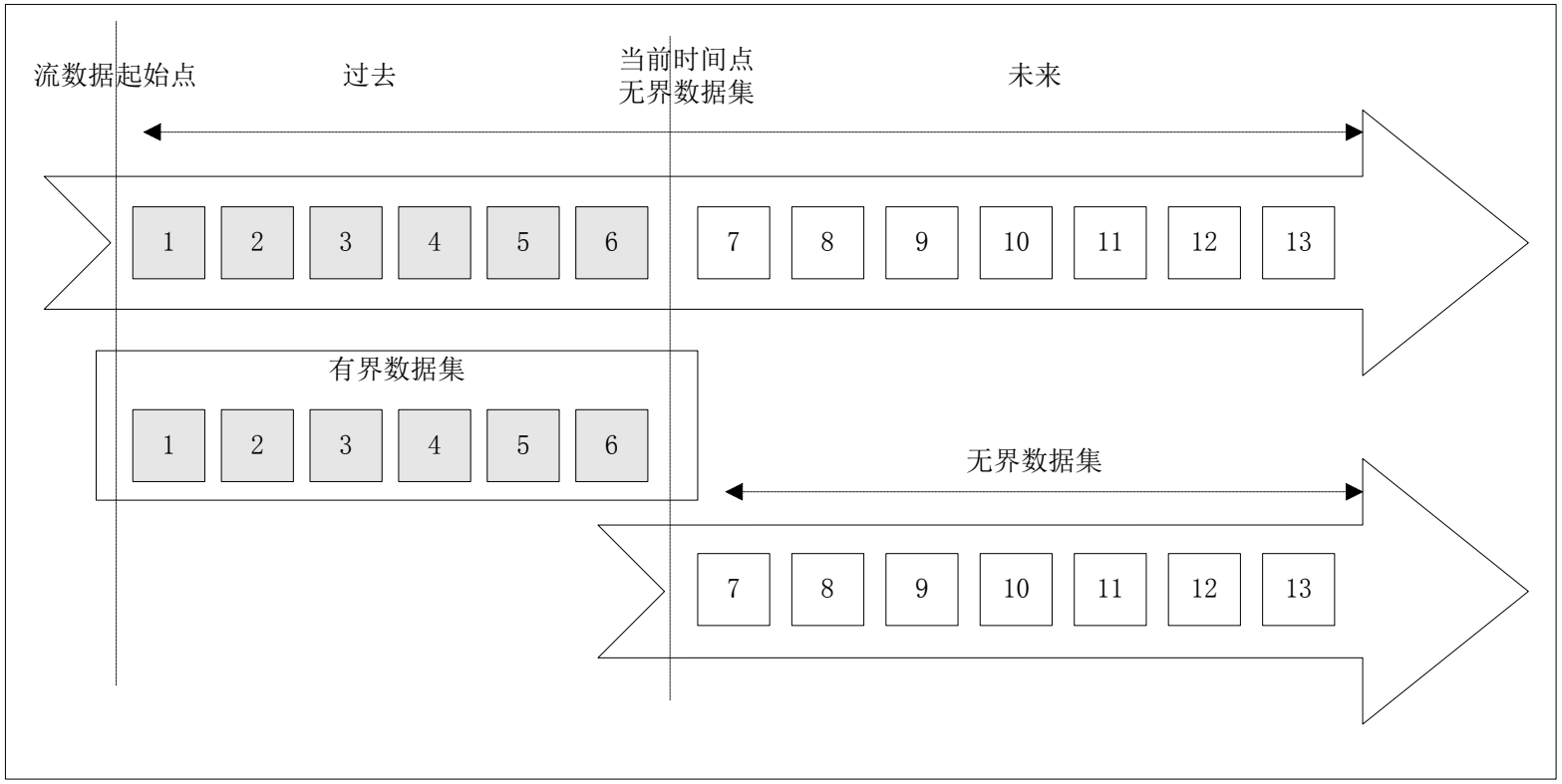
3.3.3 数据流水线应用

（3）Flink如何支持数据流水线应用

Flink的SQL接口（或者Table API）以及丰富的用户自定义函数，可以解决许多常见的数据转换问题。通过使用更具通用性的DataStream API，还可以实现具有更加强大功能的数据流水线。Flink提供了大量的连接器，可以连接到各种不同类型的数据存储系统，比如Kafka、Kinesis、Elasticsearch和JDBC数据库系统。同时，Flink供了面向文件系统的连续型数据源，可用来监控目录变化，并提供了数据槽（sink），支持以时间分区的方式写入文件。

3.4 Flink中的统一数据处理

根据数据的产生方式，我们可以把数据集分为两种类型：有界数据集和无界数据集



3.4 Flink中的统一数据处理

•有界数据集具有时间边界，在处理过程中数据一定会在某个时间范围内起始和结束，有可能是一小时，也有可能是一天内的交易数据。有界数据集的特点是，数据是静止不动的，不会存在数据的追加操作。对有界数据集的数据处理方式被称为批处理

•对于无界数据集，数据从开始生成就一直持续不断地产生新的数据，因此数据是没有边界的，例如服务器信令、网络传输流、传感器信号数据、实时日志信息等。和批量数据处理方式对应，对无界数据集的数据处理方式被称为流处理。

•有界数据集与无界数据集是一个相对模糊的概念。对于有界数据集而言，如果数据一条一条地经过处理引擎，那么也可以认为是无界的。反过来，对于无界数据集而言，如果每间隔一分钟、一小时、一天进行一次计算，那么也可以认为这一段时间内的数据又相对是有界的。

3.4 Flink中的统一数据处理

•对于Spark而言，它会使用一系列连续的微小批处理来模拟流处理，也就是说，它会在特定的时间间隔内发起一次计算，而不是每条数据都触发计算，这就相当于把无界数据集切分为多个小量的有界数据集。

•对于Flink而言，它把有界数据集看成无界数据集的一个子集，因此，将批处理与流处理混合到同一套引擎当中，用户使用Flink引擎能够同时实现批处理与流处理任务。

3.5.Flink技术栈

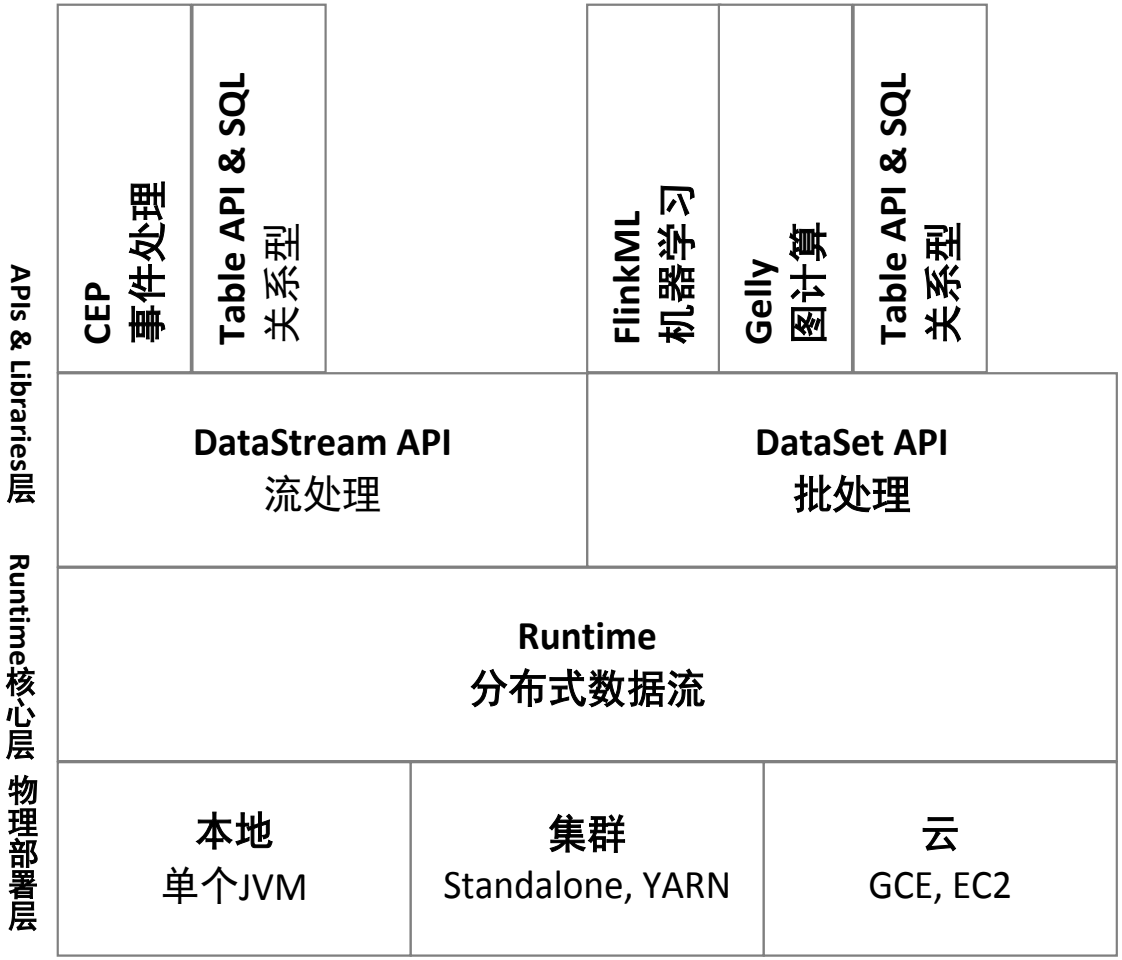
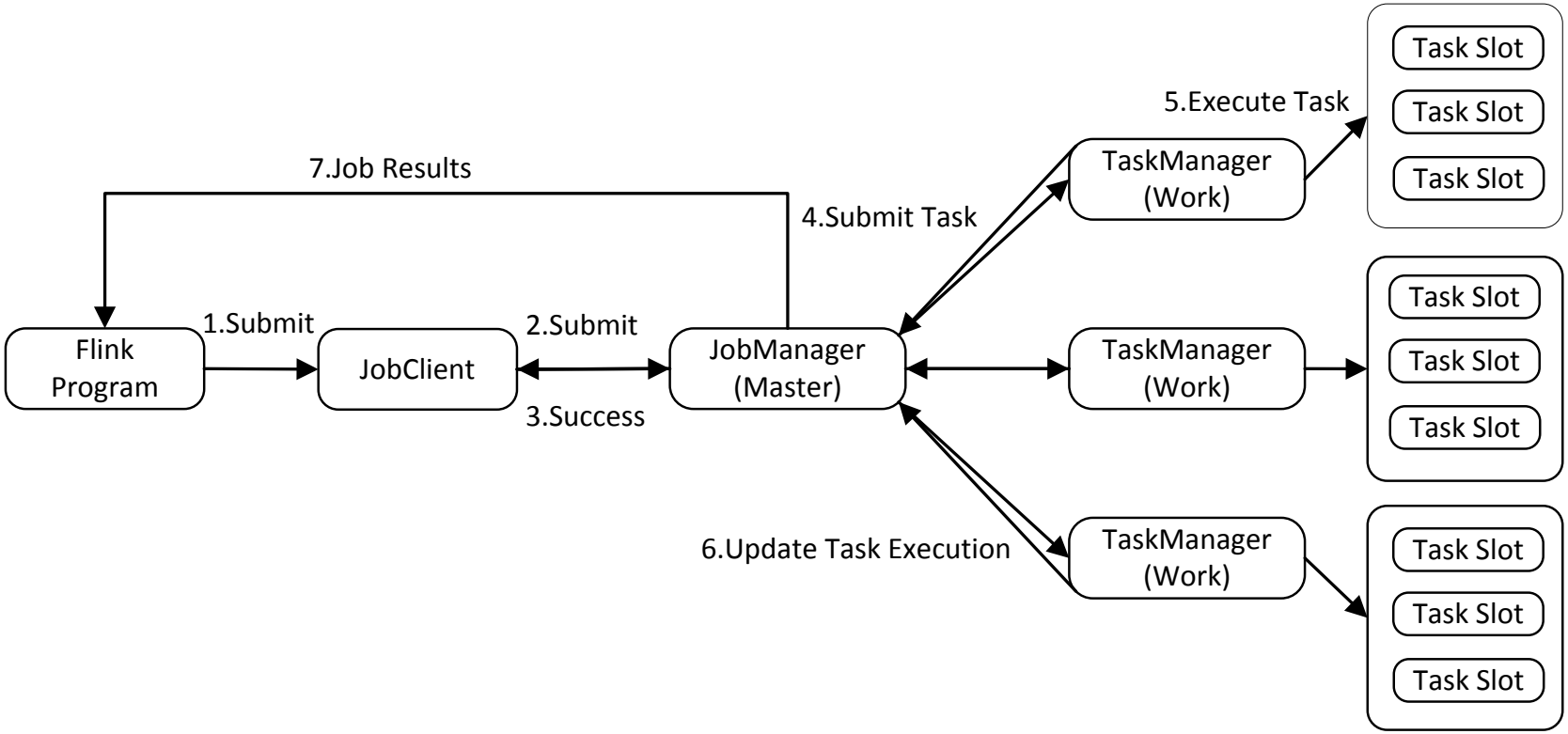


图 Flink核心组件栈

3.6.Flink工作原理

Flink系统主要由两个组件组成，分别为JobManager和TaskManager，Flink 架构也遵循Master-Slave架构设计原则，JobManager为Master节点，TaskManager为Slave节点。



3.7.Flink编程模型

Flink 提供了不同级别的抽象（如图所示），以开发流或批处理作业。

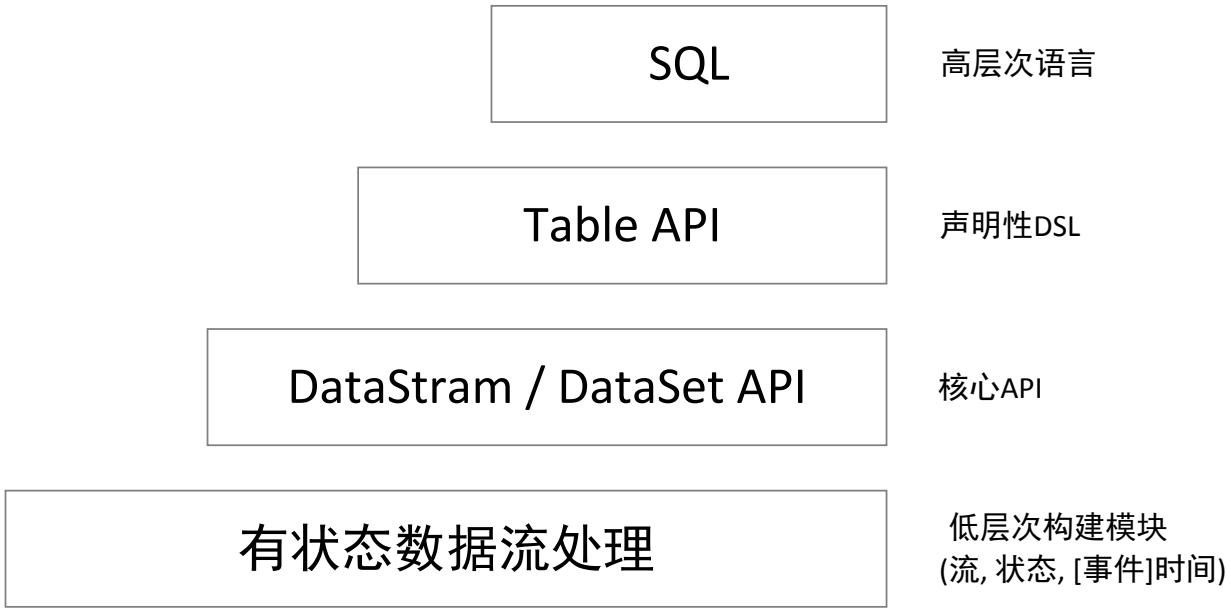


图 Flink编程模型

3.8 Flink的应用程序结构

如图所示，一个完整的Flink应用程序结构包含如下三个部分：

（1）数据源（Source）：Flink 在流处理和批处理上的数据源大概有4类：基于本地集合的数据源、基于文件的数据源、基于网络套接字的数据源、自定义的数据源。常见的自定义数据源包括Apache kafka、Amazon Kinesis Streams、RabbitMQ、Twitter Streaming API、Apache NiFi等，当然用户也可以定义自己的数据源。

（2）数据转换（Transformation）：数据转换的各种操作包括map、 flatMap、filter、keyBy、reduce、aggregation、window、windowAll、union、select等，可以将原始数据转换成满足要求的数据。

（3）数据输出（Sink）：数据输出是指Flink将转换计算后的数据发送的目的地。常见的数据输出包括写入文件、打印到屏幕、写入Socket 、自定义Sink等 。常见的自定义Sink有Apache kafka、RabbitMQ、MySQL、ElasticSearch、Apache Cassandra、Hadoop FileSystem 等。

3.8 Flink的应用程序结构

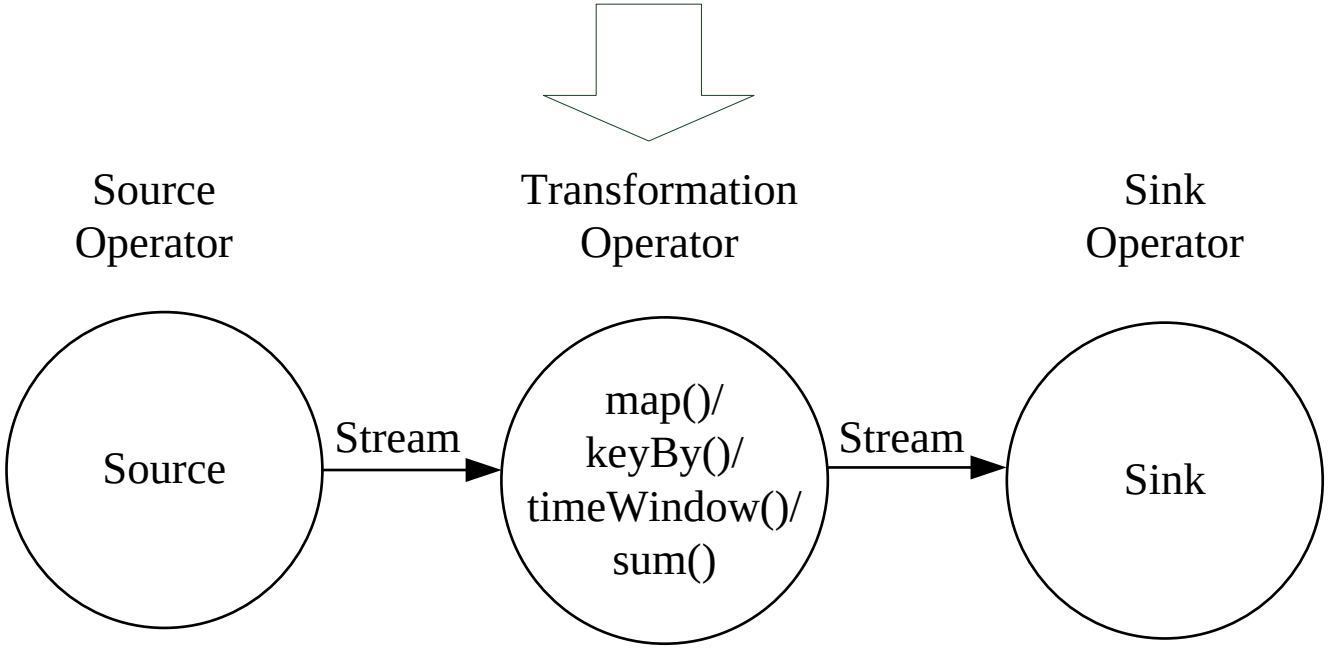
下图以一段简单代码为实例，演示了Flink的应用程序结构。

```
val source = env.socketTextStream("localhost",9999,'\n')

val dataStream = source.flatMap(_.split(" "))
    .map((_,1))
    .keyBy(0)
    .timeWindow(Time.seconds(2),Time.seconds(2))
    .sum(1)

dataStream.print()
```

Source

Transformation

Sink



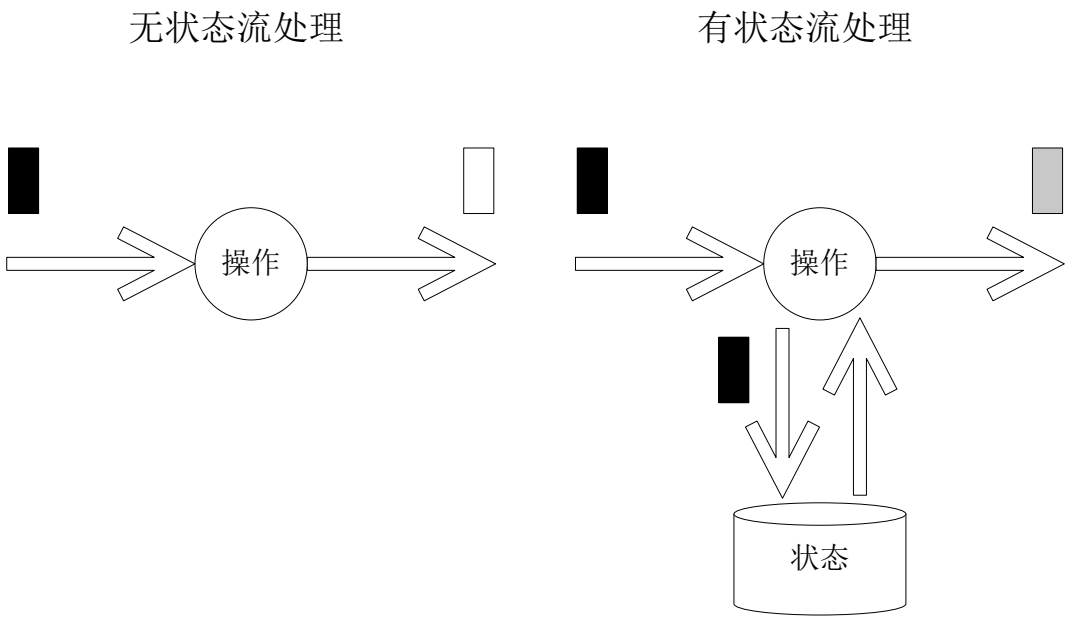
3.9 Flink中的数据一致性

•对于分布式流处理系统而言，高吞吐、低延迟往往是最主要的需求。与此同时，数据一致性在分布式系统中也很重要，对于正确性要求较高的场景，“精确一次”一致性的实现往往也非常重要。如何保证分布式系统有状态计算的一致性，是Flink作为一个分布式流计算框架必须要解决的问题。

•Flink通过异步屏障快照机制来实现“精确一次”一致性的保证，当任务中途崩溃或者取消之后，可以通过检查点或者保存点来进行恢复，实现数据流的重放，从而让任务达到一致性的效果，同时，这种机制不会牺牲系统的性能。

3.9.1 有状态计算

流计算分为无状态和有状态两种情况。无状态计算观察每个独立的事件，每一条消息来了以后和前后其他消息都没有关系，比如一个应用程序实时接收温度传感器的数据，当温度超过40度时就报警，这就是无状态的数据。有状态计算则会基于多个事件输出结果，比如，计算过去1个小时的平均温度，就属于有状态计算。



3.9.2 数据一致性

当在分布式系统中引入状态时，自然也引入了一致性问题。根据正确性级别的不同，一致性可以分为如下三种形式：

（1）最多一次（at-most-once）：尽可能正确，但不保证一定正确。也就是说，当故障发生时，什么都不做，既不恢复丢失状态，也不重播丢失的数据。这就意味着，在系统发生故障以后，聚合结果可能会出错。

（2）至少一次（at-least-once）：在系统发生故障以后，聚合计算不会漏掉故障恢复之前窗口内的事件，但可能会重复计算某些事件，这通常用于实时性较高但准确性要求不高的场合。该模式意味着系统将以一种更加简单的方式来对算子的状态进行快照处理，系统崩溃后恢复时，算子的状态中有一些记录可能会被重放多次。例如，失败后恢复时，统计值将等于或者大于流中元素的真实值。

3.9.2 数据一致性

（3）精确一次（exactly-once）：在系统发生故障后，聚合结果与假定没有发生故障情况时一致。该模式意味着系统在进行恢复时，每条记录将在算子状态中只被重播一次。例如在一段数据流中，不管该系统崩溃或者重启了多少次，该统计结果将总是跟流中的元素的真实个数一致。这种语义加大了高吞吐和低延迟的实现难度。与“至少一次”模式相比，“精确一次”模式整体的处理速度会相对比较慢，因为在开启“精确一次”模式后，为了保证一致性，就会开启数据对齐，从而会影响系统的一些性能。

3.9.3 异步屏障快照机制

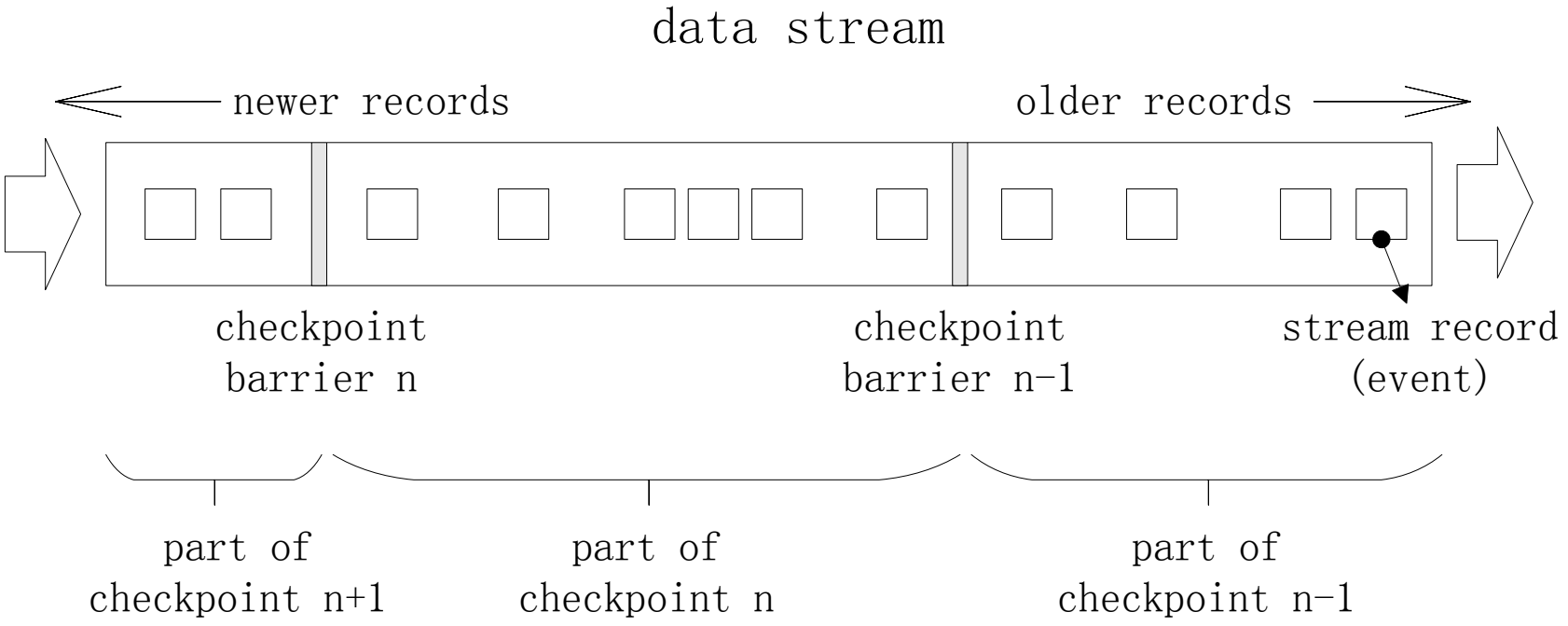
“精确一次”模式要求作业从失败恢复后的状态以及管道中的数据流要和失败时一致，通常这是通过定期对作业状态和数据流进行快照实现的。但是，传统的快照机制存在两个主要问题：

- （1）需要所有节点停止工作，即暂停整个计算过程，这个必然会影响到数据处理效率和时效性；
- （2）需要保存所有节点的操作中的状态以及所有在传输中的数据，这个会消费大量的存储空间。

为了解决上述问题，Flink采用了异步快照方式，它基于Chandy-lamport算法，制定了应对流计算“精确一次”语义的检查点机制——异步屏障快照机制（Asynchronous Barrier Snapshot）。

3.9.3 异步屏障快照机制

异步屏障快照是一种轻量级的快照技术，能以低成本备份 DAG（有向无环图）或 DCG（有向有环图）计算作业的状态，这使得计算作业可以频繁进行快照并且不会对性能产生明显影响。异步屏障快照机制的核心思想是，通过屏障消息来标记触发快照的时间点和对应的数据，从而将数据流和快照时间解耦，以实现异步快照操作，同时也大大降低了对管道数据的依赖（对 DAG 类作业甚至完全不依赖），减小了随之而来的快照大小。



3.10 本章小结

•Apache Flink是一个分布式处理引擎，用于对无界和有界数据流进行有状态计算。Flink以数据并行和流水线方式执行任意流数据程序，Flink的流水线运行时系统可以执行批处理和流处理程序。此外，Flink的运行时本身也支持迭代算法的执行。

•近年来，数据架构设计开始由传统数据处理架构、大数据Lambda架构向流处理架构演变，这种转变使得Flink可以在大数据应用场景中“大显身手”。目前，Flink支持的典型的应用场景包括事件驱动型应用、数据分析应用和数据流水线应用。

•经过多年的发展，Flink已经形成了完备的生态系统，它的技术栈可以满足企业多种应用场景的开发需求，减轻了企业的大数据应用系统的开发和维护负担。在未来，随着企业实时应用场景的不断增多，Flink在大数据市场上的地位和作用将会更加凸显，Flink的发展前景值得期待。

附录A：主讲教师林子雨简介

主讲教师：林子雨

单位：厦门大学计算机科学系
E-mail: ziyulin@xmu.edu.cn
个人网页：http://dblab.xmu.edu.cn/post/linziyu
数据库实验室网站：http://dblab.xmu.edu.cn

扫一扫访问个人主页

林子雨，男，1978年出生，博士（毕业于北京大学），全国高校知名大数据教师，现为厦门大学计算机科学系副教授，曾任厦门大学信息科学与技术学院院长助理、晋江市发展和改革局副局长。中国计算机学会数据库专业委员会委员，中国计算机学会信息系统专业委员会委员。国内高校首个“数字教师”提出者和建设者，厦门大学数据库实验室负责人，厦门大学云计算与大数据研究中心主要建设者和骨干成员，2013年度、2017年度和2020年度厦门大学教学类奖教金获得者，荣获2019年福建省精品在线开放课程、2018年厦门大学高等教育成果特等奖、2018年福建省高等教育教学成果二等奖、2018年国家精品在线开放课程。主要研究方向为数据库、数据仓库、数据挖掘、大数据、云计算和物联网，并以第一作者身份在《软件学报》《计算机学报》和《计算机研究与发展》等国家重点期刊以及国际学术会议上发表多篇学术论文。作为项目负责人主持的科研项目包括1项国家自然科学青年基金项目(No.61303004)、1项福建省自然科学青年基金项目(No.2013J05099)和1项中央高校基本科研业务费项目(No.2011121049)，主持的教改课题包括1项2016年福建省教改课题和1项2016年教育部产学协作育人项目，同时，作为课题负责人完成了国家发改委城市信息化重大课题、国家物联网重大应用示范工程区域试点泉州市工作方案、2015泉州市互联网经济调研等课题。中国高校首个“数字教师”提出者和建设者，2009年至今，“数字教师”大平台累计向网络免费发布超过1000万字高价值的研究和教学资料，累计网络访问量超过1000万次。打造了中国高校大数据教学知名品牌，编著出版了中国高校第一本系统介绍大数据知识的专业教材《大数据技术原理与应用》，并成为京东、当当网等网店畅销书籍；建设了国内高校首个大数据课程公共服务平台，为教师教学和学生学习大数据课程提供全方位、一站式服务，年访问量超过200万次，累计访问量超过1000万次。

附录B：大数据学习路线图

大数据学习路线图访问地址：http://dblab.xmu.edu.cn/post/10164/

附录C：林子雨大数据系列教材

了解全部教材信息：http://dblab.xmu.edu.cn/post/bigdatabook/

附录D：《大数据导论（通识课版）》教材

开设全校公共选修课的优质教材

本课程旨在实现以下几个培养目标：

- 引导学生步入大数据时代，积极投身大数据的变革浪潮之中
- 了解大数据概念，培养大数据思维，养成数据安全意识
- 认识大数据伦理，努力使自己的行为符合大数据伦理规范要求
- 熟悉大数据应用，探寻大数据与自己专业的应用结合点
- 激发学生基于大数据的创新创业热情

高等教育出版社 ISBN:978-7-04-053577-8 定价：32元 版次：2020年2月第1版
教材官网：http://dblab.xmu.edu.cn/post/bigdataintroduction/

附录E：《大数据导论》教材

- 林子雨 编著《大数据导论》
- 人民邮电出版社，2020年9月第1版
- ISBN:978-7-115-54446-9 定价：49.80元

教材官网：http://dblab.xmu.edu.cn/post/bigdata-introduction/

开设大数据专业导论课的优质教材

扫一扫访问教材官网

附录F：《大数据技术原理与应用（第3版）》教材

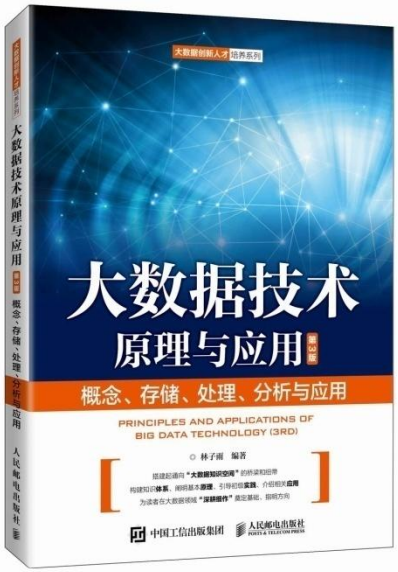
《大数据技术原理与应用——概念、存储、处理、分析与应用（第3版）》，由厦门大学计算机科学系林子雨博士编著，是国内高校第一本系统介绍大数据知识的专业教材。人民邮电出版社　ISBN:978-7-115-54405-6　定价：59.80元

全书共有17章，系统地论述了大数据的基本概念、大数据处理架构Hadoop、分布式文件系统HDFS、分布式数据 库HBase、NoSQL数据库、云数据库、分布式并行编程模型MapReduce、Spark、流计算、Flink、图计算、数据可视化以及大数据在互联网、生物医学和物流等各个领域的应用。在Hadoop、HDFS、HBase、MapReduce、Spark和Flink等重要章节，安排了入门级的实践操作，让读者更好地学习和掌握大数据关键技术。

本书可以作为高等院校计算机专业、信息管理等相关专业的大数据课程教材，也可供相关技术人员参考、学习、培训之用。

欢迎访问《大数据技术原理与应用——概念、存储、处理、分析与应用》教材官方网站：
http://dblab.xmu.edu.cn/post/bigdata3

扫一扫访问教材官网



附录G：《大数据基础编程、实验和案例教程（第2版）》

本书是与《大数据技术原理与应用（第3版）》教材配套的唯一指定实验指导书

大数据教材

1+1黄金组合

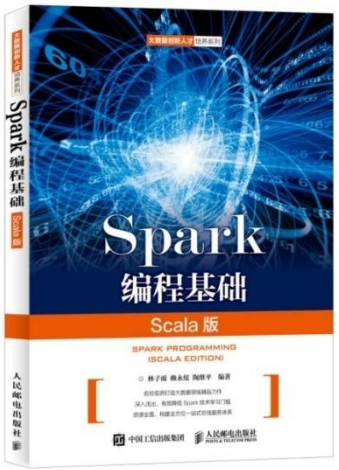
厦门大学林子雨编著

配套实验指导书

- 步步引导，循序渐进，详尽的安装指南为顺利搭建大数据实验环境铺平道路
- 深入浅出，去粗取精，丰富的代码实例帮助快速掌握大数据基础编程方法
- 精心设计，巧妙融合，八套大数据实验题目促进理论与编程知识的消化和吸收
- 结合理论，联系实际，大数据课程综合实验案例精彩呈现大数据分析全流程

林子雨编著《大数据基础编程、实验和案例教程（第2版）》
清华大学出版社 ISBN:978-7-302-55977-1 定价：69元 2020年10月第2版

附录H：《Spark编程基础（Scala版）》



《Spark编程基础（Scala版）》

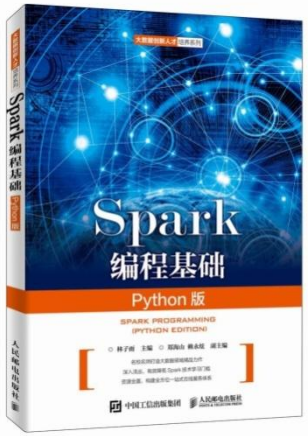
厦门大学 林子雨，赖永炫，陶继平 编著

披荆斩棘，在大数据丛林中开辟学习捷径
填沟削坎，为快速学习Spark技术铺平道路
深入浅出，有效降低Spark技术学习门槛
资源全面，构建全方位一站式在线服务体系

人民邮电出版社出版发行，ISBN:978-7-115-48816-9
教材官网：http://dblab.xmu.edu.cn/post/spark/

本书以Scala作为开发Spark应用程序的编程语言，系统介绍了Spark编程的基础知识。全书共8章，内容包括大数据技术概述、Scala语言基础、Spark的设计与运行原理、Spark环境搭建和使用方法、RDD编程、Spark SQL、Spark Streaming、Spark MLlib等。本书每个章节都安排了入门级的编程实践操作，以便读者更好地学习和掌握Spark编程方法。本书官网免费提供了全套的在线教学资源，包括讲义PPT、习题、源代码、软件、数据集、授课视频、上机实验指南等。

附录I：《Spark编程基础（Python版）》



《Spark编程基础（Python版）》

厦门大学 林子雨，郑海山，赖永炫 编著

披荆斩棘，在大数据丛林中开辟学习捷径
填沟削坎，为快速学习Spark技术铺平道路
深入浅出，有效降低Spark技术学习门槛
资源全面，构建全方位一站式在线服务体系

人民邮电出版社出版发行，ISBN:978-7-115-52439-3
教材官网： http://dblab.xmu.edu.cn/post/spark-python/

本书以Python作为开发Spark应用程序的编程语言，系统介绍了Spark编程的基础知识。全书共8章，内容包括大数据技术概述、Spark的设计与运行原理、Spark环境搭建和使用方法、RDD编程、Spark SQL、Spark Streaming、Structured Streaming、Spark MLlib等。本书每个章节都安排了入门级的编程实践操作，以便读者更好地学习和掌握Spark编程方法。本书官网免费提供了全套的在线教学资源，包括讲义PPT、习题、源代码、软件、数据集、上机实验指南等。

附录J：高校大数据课程公共服务平台

高校大数据课程

公 共 服 务 平 台

http://dblab.xmu.edu.cn/post/bigdata-teaching-platform/

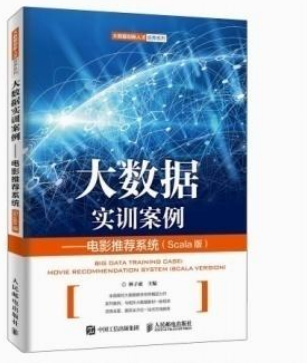
扫一扫访问平台主页

扫一扫观看3分钟FLASH动画宣传片

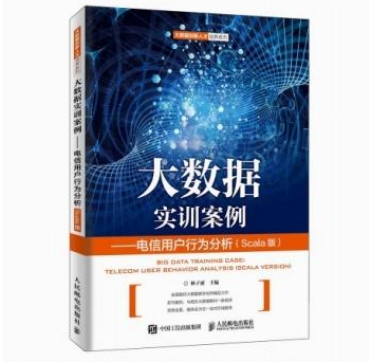
附录K：高校大数据实训课程系列案例教材

为了更好满足高校开设大数据实训课程的教材需求，厦门大学数据库实验室林子雨老师团队联合企业共同开发了《高校大数据实训课程系列案例》，目前已经完成开发的系列案例包括：

《电影推荐系统》（已经于2019年5月出版）
《电信用户行为分析》（已经于2019年5月出版）
《实时日志流处理分析》
《微博用户情感分析》
《互联网广告预测分析》
《网站日志处理分析》



系列案例教材将于2019年陆续出版发行，教材相关信息，敬请关注网页后续更新！
http://dblab.xmu.edu.cn/post/shixunkecheng/



扫一扫访问大数据实训课程系列案例教材主页

Department of Computer Science, Xiamen University, 2021