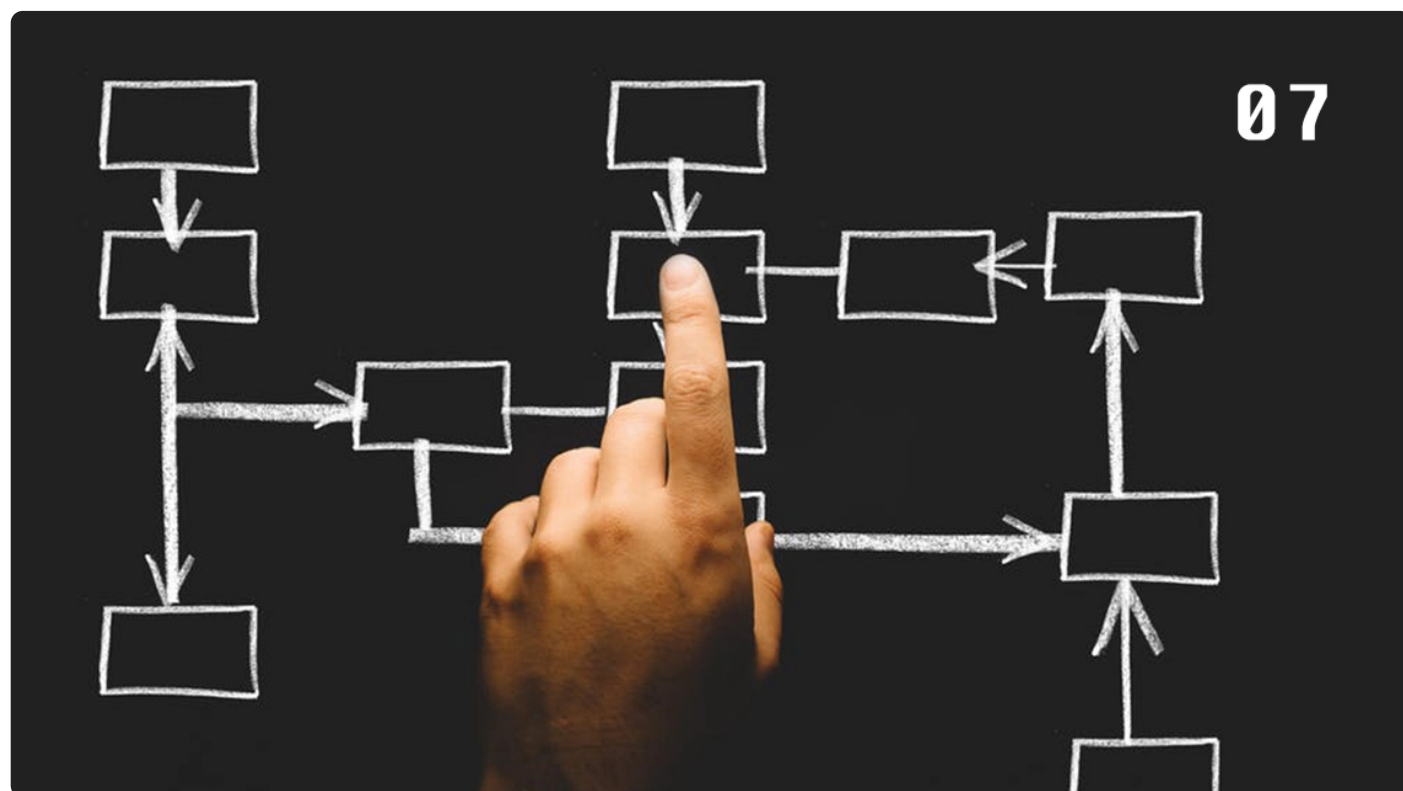


07 | Workflow设计模式：让你在大规模数据世界中君临天下

2019-05-01 蔡元楠 来自北京

《大规模数据处理实战》



你好，我是蔡元楠。

今天我要与你分享的主题是“Workflow 设计模式”。

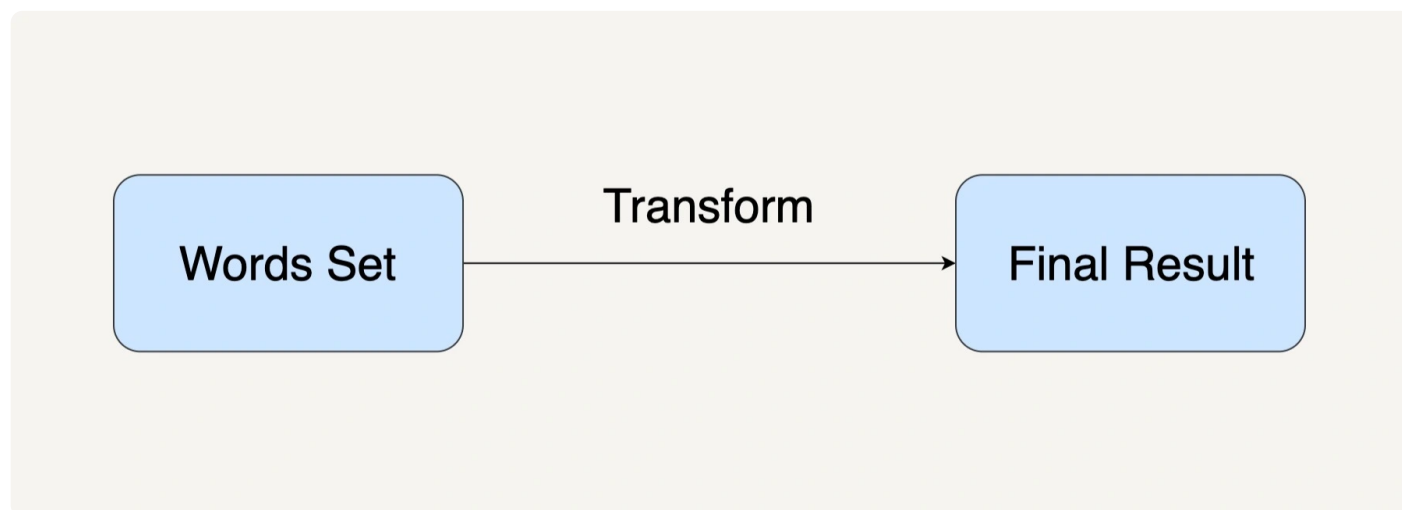
在上一讲中，我们一起学习了大规模数据处理的两种处理模式——批处理和流处理。

利用好这两种处理模式，作为架构师的你就可以运筹帷幄，根据实际需求搭建出一套符合自己应用的数据处理系统。

然而，光是掌握了这两种数据处理模式就足够自如应对大规模数据世界中的需求挑战吗？从我的实战经验中看来，其实未必。

我们每个人在最开始学习大规模数据处理的时候，可能都是以 WordCount 作为教学例子来进行学习的。

WordCount 这个例子，只需要一个单词集合作为输入，数据处理的结果是统计单词出现的次数，中间只需要经过一次数据处理的转换，就如同下图所示。



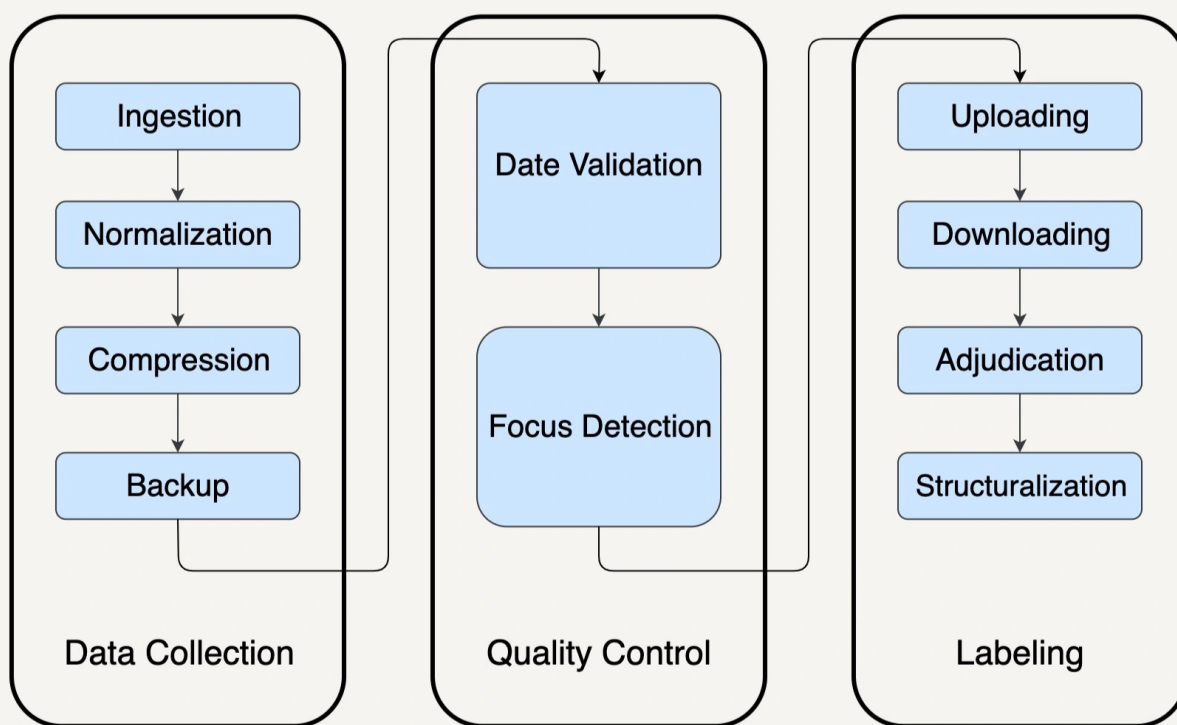
但在现实的应用场景中，各式各样的应用需求决定了大规模数据处理中的应用场景会比 WordCount 复杂很多倍。

我还是以我在第一讲中所提到过的例子来说明吧。

在根据活跃在街头的美团外卖电动车的数量来预测美团的股价这个例子中，我们的输入数据集有可能不止一个。

例如，会有自己团队在街道上拍摄到的美团外卖电动车图片，会有第三方公司提供的美团外卖电动车数据集等等。

整个数据处理流程又会需要至少 10 个处理模块，每一个处理模块的输出结果都将会成为下一个处理模块的输入数据，就如同下图所示。



像上面的图示一样，我们将这种由多个不同的处理模块连接在一起，最后得出一个自己需要结果的有向无环图（Directed Acyclic Graph/DAG），称为一个 workflow 系统（Workflow System）。

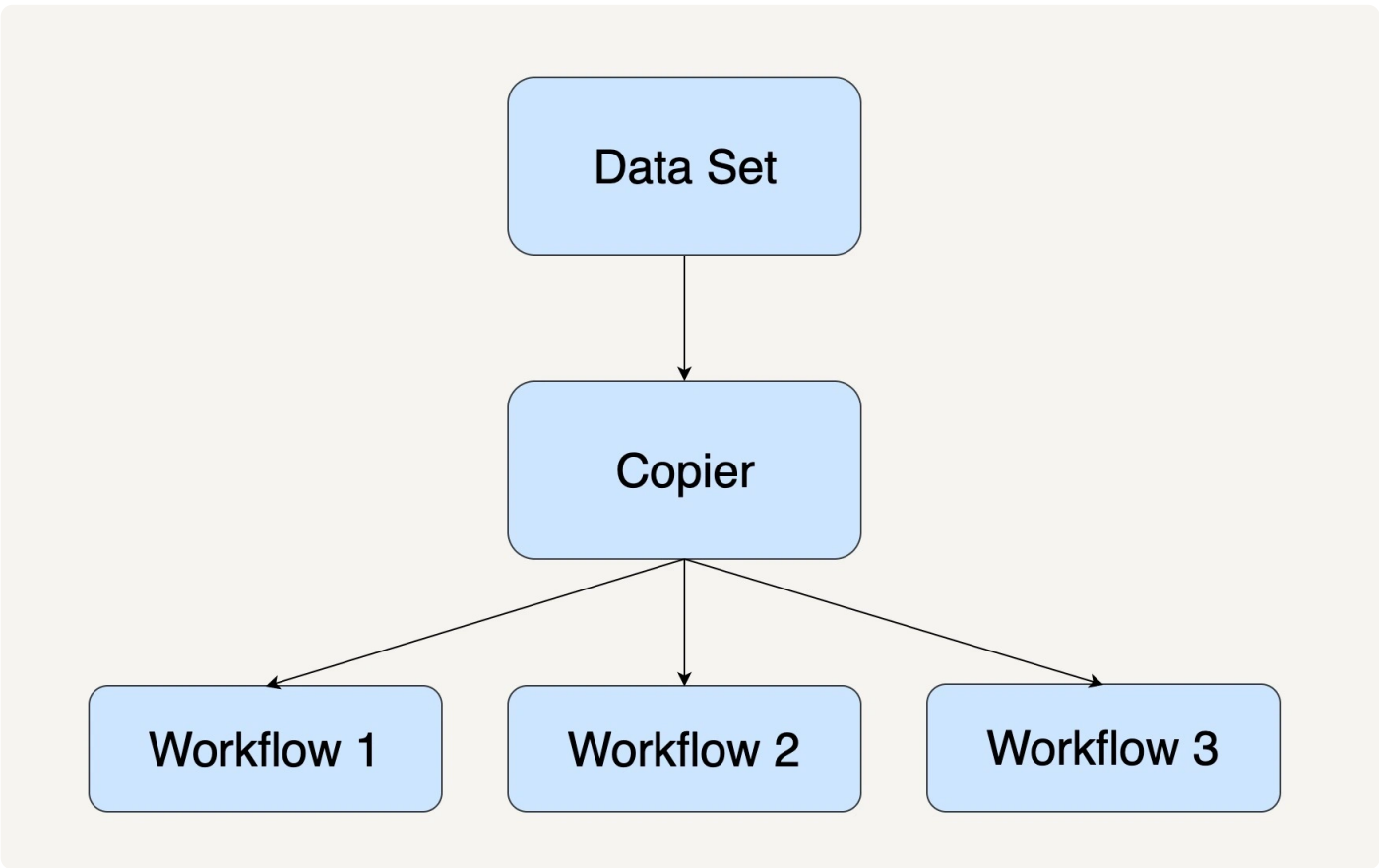
在工作流系统的每个处理模块里，系统要执行的操作有可能不是单单一个数据转换的操作这么简单。像在上面例子的 Ingestion 这个模块中，我们需要将多个不同的数据集合并在一起，也需要将不合格的一些图片过滤掉。

如果你用过 Apache Spark 1.4 以上的版本的话，Spark 平台里面的 Execution DAG 就可以为你展示一个完整的工作流图。

今天，我为你解释四种工作流系统的设计模式，希望你能够很好地理解它们，并运用在自己的数据处理系统设计中。在遇到各种复杂的应用场景的时候能够从容面对。

复制模式（Copier Pattern）

复制模式通常是将单个数据处理模块中的数据，完整地复制到两个或更多的数据处理模块中，然后再由不同的数据处理模块进行处理。 workflows 系统图通常如下图所示。



当我们在处理大规模数据时，需要对同一个数据集采取多种不同的数据处理转换，我们就可以优先考虑采用**复制模式**。

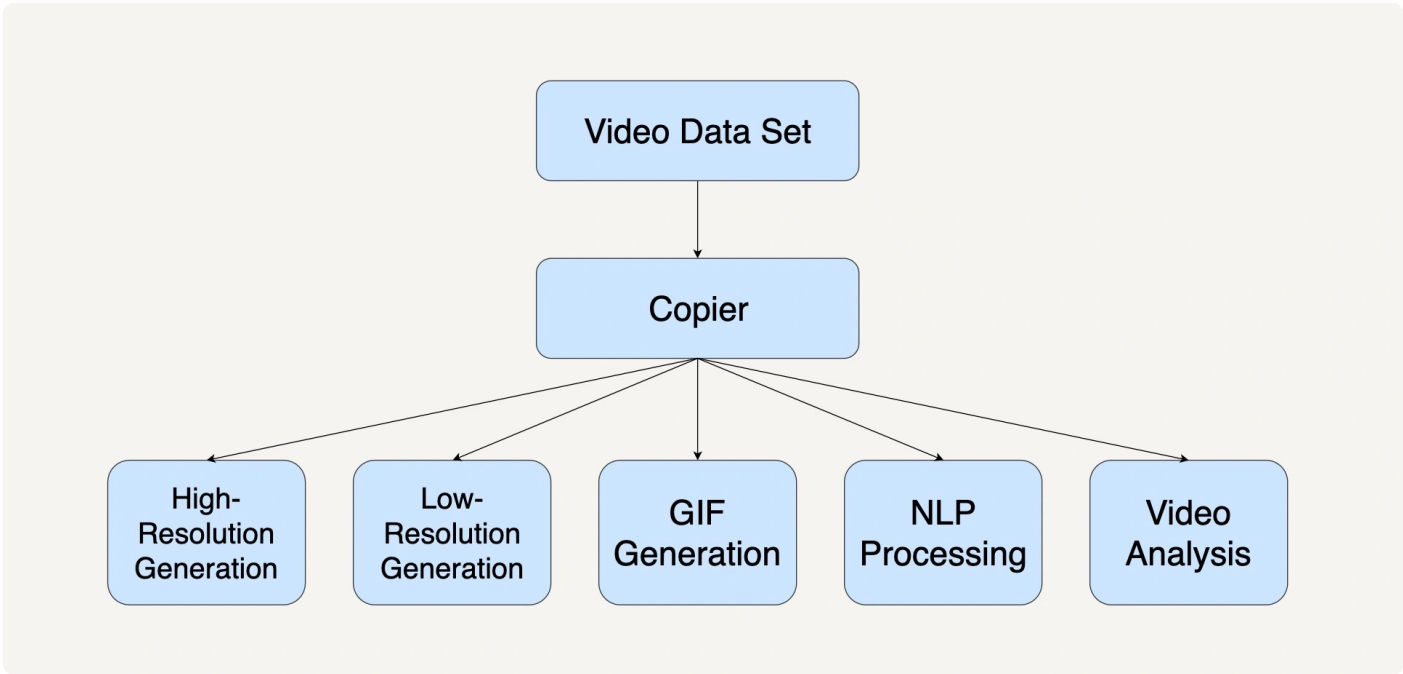
我来举个在 YouTube 视频平台中，系统处理视频数据集的一个例子吧。

我们都知道，视频平台很多时候都会提供不同分辨率的视频。4K 或 1080P 的视频可以提供给网络带宽很高的用户。而在网络很慢的情况下，视频平台系统会自动转换成低分辨率格式的视频，像 360P 这样的视频给用户。

而在 YouTube 视频平台中，如果你将鼠标放在视频缩略图上，它会自动播放一段已经生成好的动画缩略图（Animated GIF Thumbnail）。

不仅如此，在平台的背后，一个视频的数据集可能被自然语言理解（NLP）的数据处理模块分析，用以自动生成视频字幕；还有可能被视频分析的数据处理模块分析，用以产生更好的内容

推荐系统。那么，它的整个 workflow 系统就会如下图所示一样。

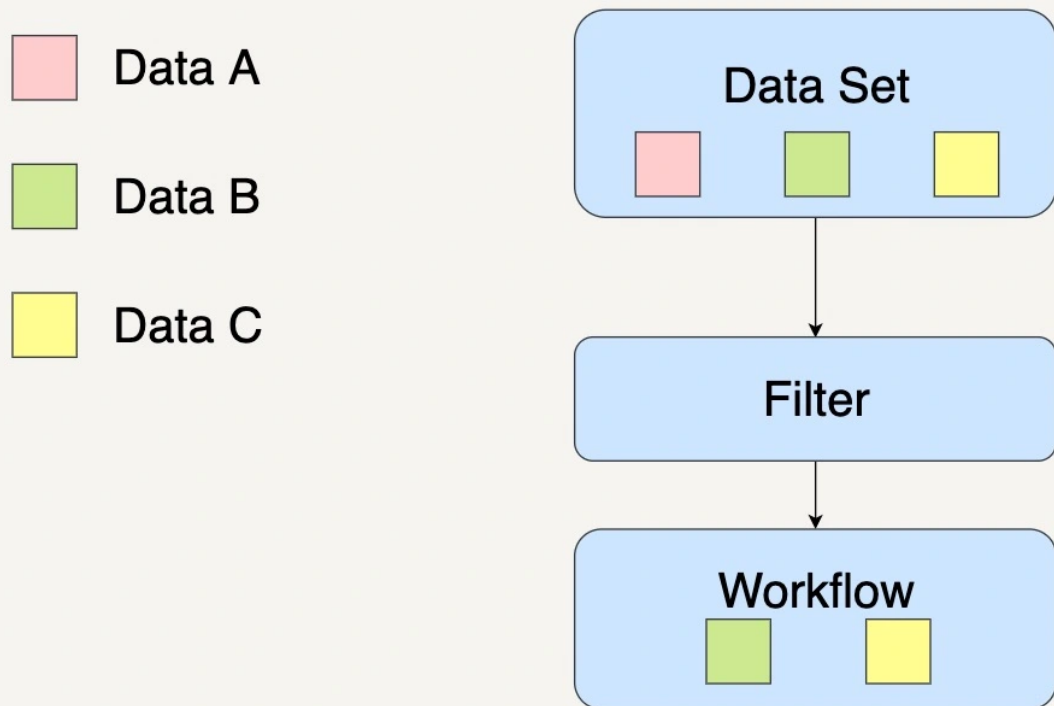


我们可以看到，在这个 workflow 系统中，每个数据处理模块的输入是相同的，而下面的 5 个数据处理模块都可以单独并且同步地运行处理。

过滤模式 (Filter Pattern)

过滤模式的作用是过滤掉不符合特定条件的数据。

在数据集通过了这个数据处理模块后，数据集会缩减到只剩下符合条件的数据。workflow 系统图通常如下图所示。



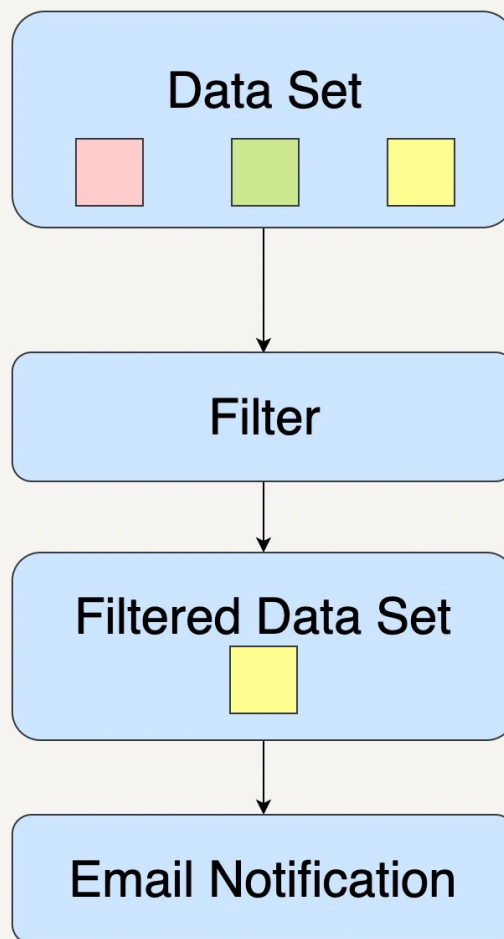
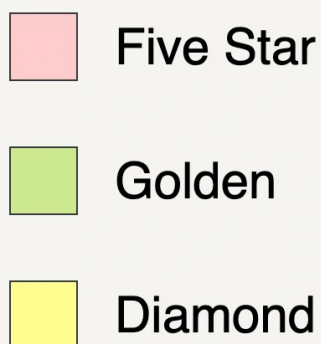
当我们在处理大规模数据时，需要针对一个数据集中某些特定的数据采取数据处理时，我们就可以优先考虑采用**过滤模式**。

我举个商城会员系统的例子来解释吧。

在商城会员系统中，系统通常会根据用户的消费次数、用户消费金额还有用户的注册时间，将用户划分成不同的等级。

假设现在商城有五星会员（Five-stars Membership）、金牌会员（Golden Membership）和钻石会员（Diamond Membership）。

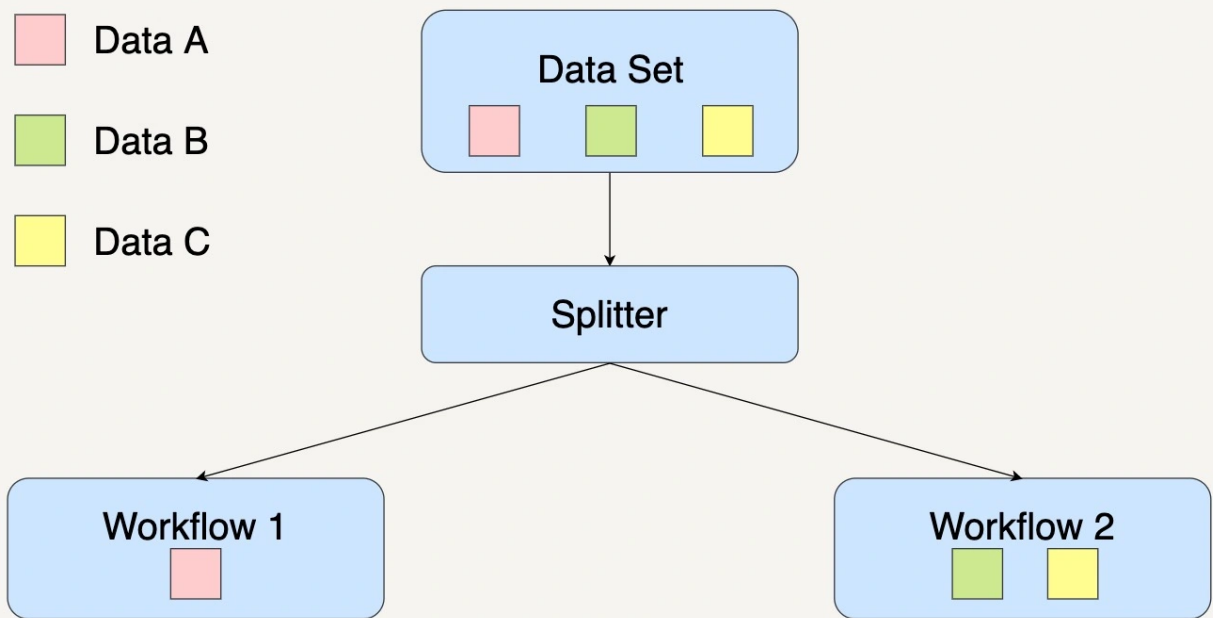
而系统现在打算通过邮件，只针对身份是钻石会员的用户发出钻石会员活动邀请。这个时候，我们就可以通过**过滤模式**，将钻石会员的用户从所有用户中筛选出来，如下图所示。



在这个工作流系统中，一个数据处理模块会将输入的数据集过滤成符合条件的数据，然后传输到下一个数据处理模块进行单独处理。

分离模式 (Splitter Pattern)

如果你在处理数据集时并不想丢弃里面的任何数据，而是想把数据分类为不同的类别来进行处理时，你就需要用到分离模式来处理数据。它的工作流系统图通常如下图所示。



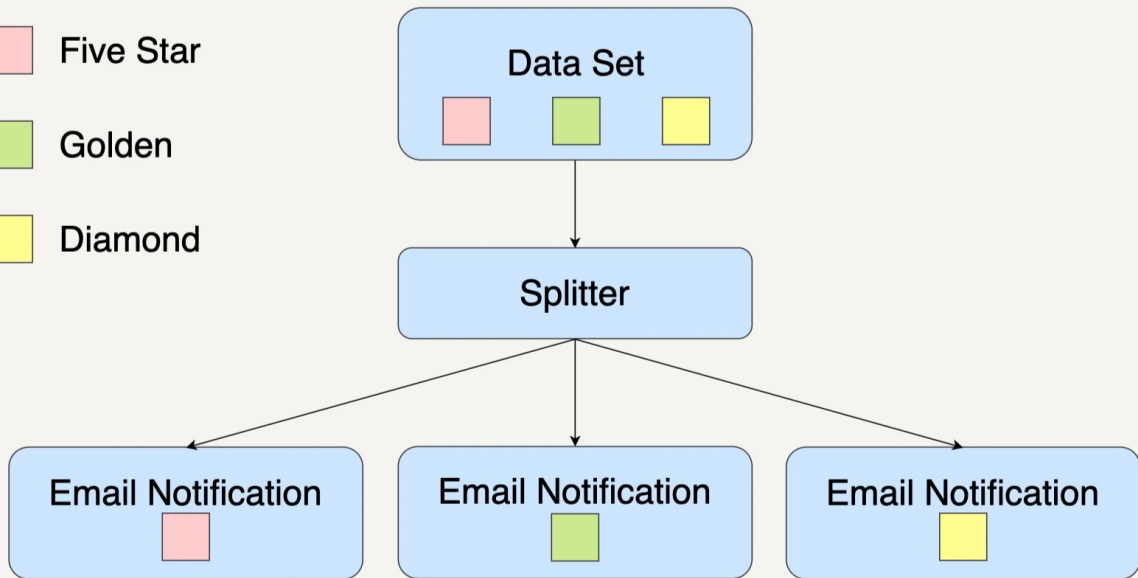
需要注意的是，分离模式并不会过滤任何数据，只是将原来的数据集分组了。

还是以刚刚商城会员系统为例。假设现在商城有五星会员、金牌会员和钻石会员。

系统现在打算通过邮件，针对**全部的会员**用户发出与他们身份相符的**不同活动**的邀请。

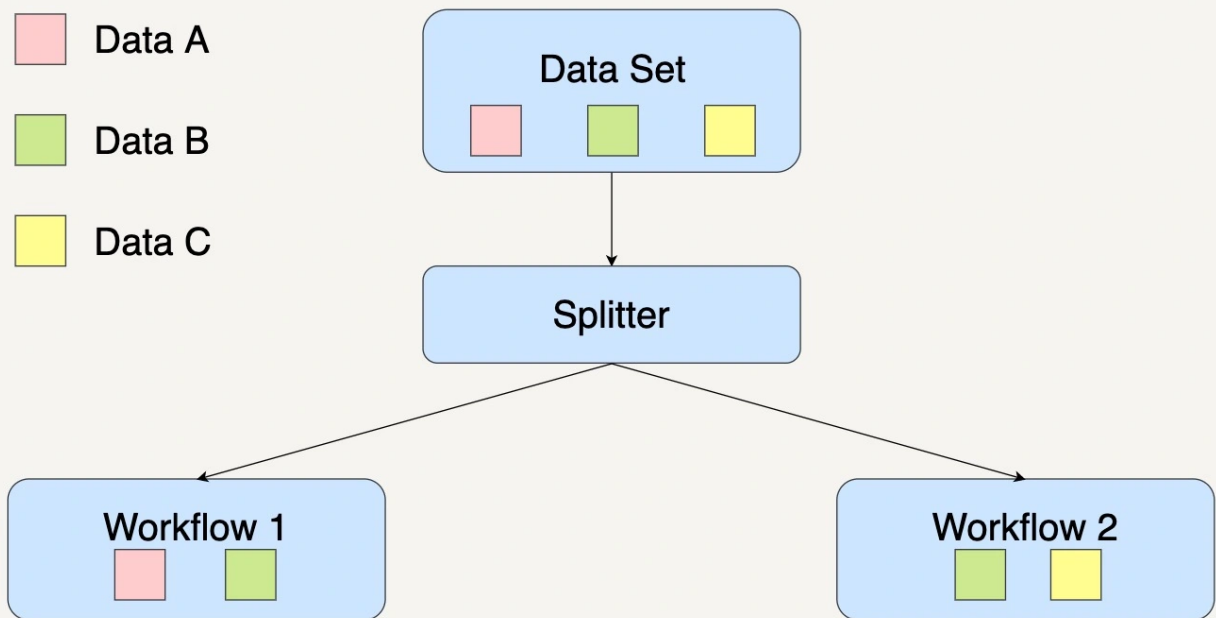
这个时候，我们就可以通过**分离模式**将用户按照会员等级分组，然后发送相应的活动内容，如下图所示。

- Five Star
- Golden
- Diamond



需要注意的是，在分离模式下，同样的数据其实是可以被划分到不同的数据处理模块的。

- Data A
- Data B
- Data C

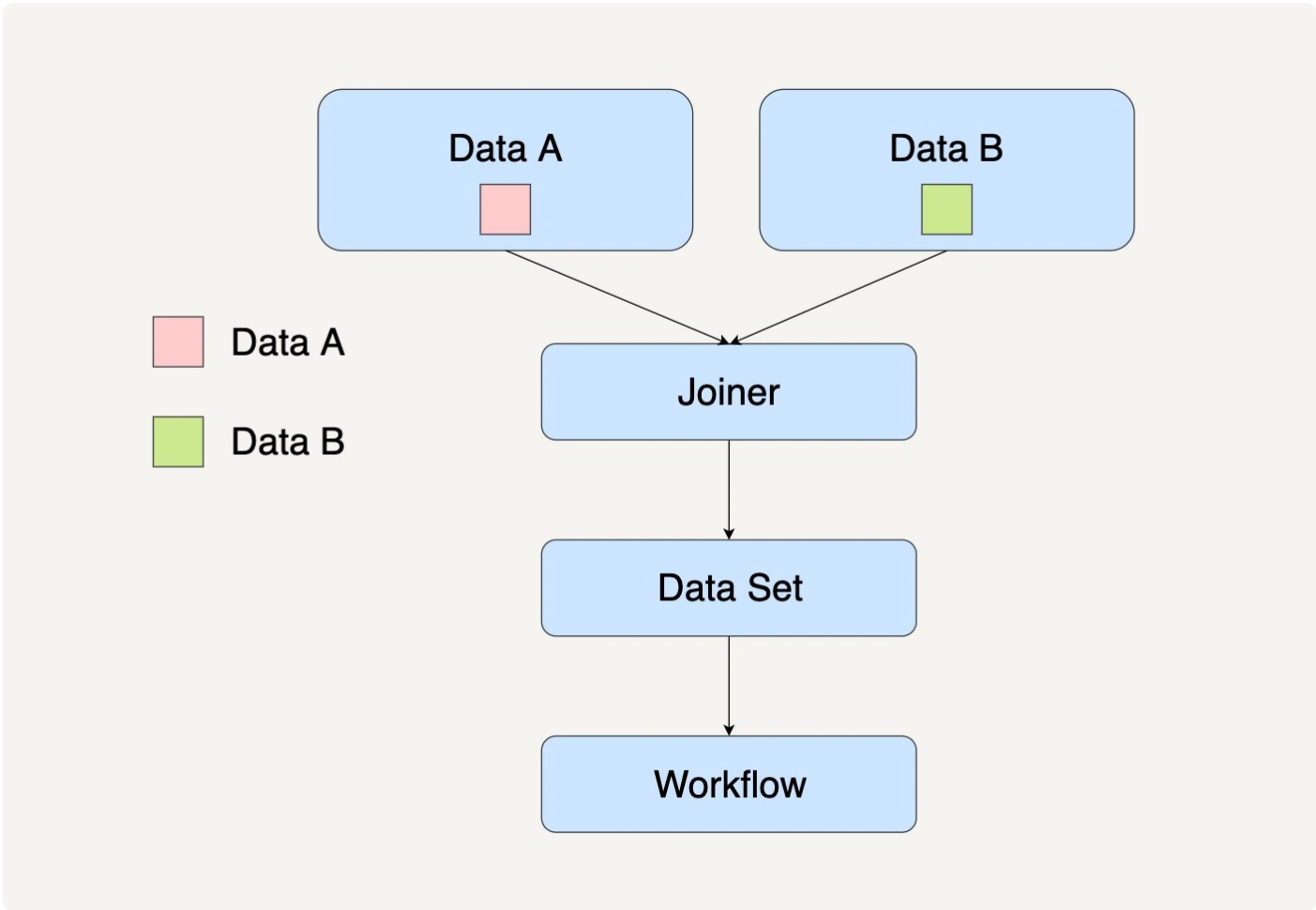


数据 B 是可以同时划分到工作流 1 和工作流 2 中。其实这种情况挺常见的，我可以给你举个例子来解释。

在银行系统上，用户可以通过勾选以短信通知或者以邮件通知的方式来提醒用户一笔交易成功。如果用户同时勾选了短信和邮件两种方式，那么属于这个用户的交易信息既会通过短信通知的数据处理模块来处理，也会通过邮件通知数据处理模块来处理。

合并模式 (Joiner Pattern)

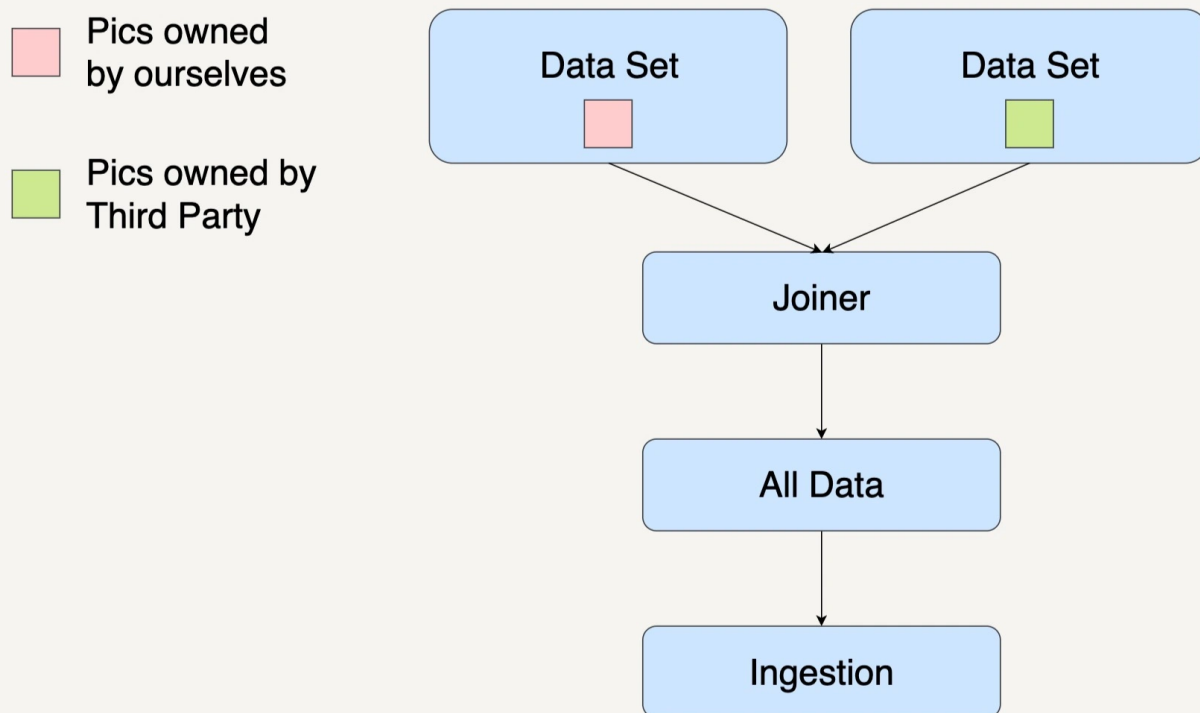
合并模式会将多个不同的数据集转换集中到一起，成为一个总数据集，然后将这个总的数据集放在一个工作流中进行处理。



还是以根据活跃在街头的美团外卖电动车的数量来预测美团的股价这个例子来说吧。

数据接入这一处理模块里，我们的输入数据有自己团队在街道上拍摄到的美团外卖电动车图片和第三方公司提供的美团外卖电动车图片。

如果我们打算先整合所有数据，然后进行其它数据处理的话，工作流系统图通常如下图所示。



小结

今天我们一起学习了在大规模数据处理中用到的四种设计模式，分别是复制模式、过滤模式、分离模式和合并模式。

在设计大规模数据处理系统的时候，我们都希望能事先设计好一个 workflow 系统图出来作为参考。

有了这样一个大规模数据处理的整体蓝图之后，对于我们理解不同的处理模块是如何相互关联或者对未来优化系统设计是有帮助的。

思考题

在一个航空预定系统中，我们需要处理用户注册、购买机票和出行前 24 小时的提醒等功能。在这整个过程中，系统的数据处理运用了哪几个设计模式呢？

欢迎你把文章分享给你的朋友。

精选留言 (27)



third

2019-05-01

用户注册，入库，合并模式

购买机票，分为查询机票和购买

查询机票，读取特定机票，过滤模式

购买机票，将所有渠道的票和合并起来，合并模式

24小时提醒，过滤出这班航班的机票，过滤模式

发送短信和电子邮箱，复制模式之后，进行分类模式发送

作者回复: 总结得很好!



👍 82



monkeyking

2019-05-05

这几个模式就是sql的几个operator吗?

复制 → subquery

过滤 → where

分离 → group by

合并 → join

作者回复: 谢谢你的留言! 感觉我又学习到了, 计算机的很多底层思想确实都是相通的。

共 3 条评论 >

👍 46



挖矿的小戈

2019-05-01

1. 注册: 合并模式 (因为注册渠道可能会有手机号注册、邮箱注册、微信注册等等不同的渠道, 所以需要合并)

2. 购买机票：过滤+合并（首先过滤出用户查找的航班机票信息、之后查找出符合条件的机票由于可能来自不同的渠道，所有需要合并后返回给用户）

3. 提醒：复制+过滤+分离

过滤：根据时间、地点等因素过滤出需要给予提醒的用户 and 机票

复制：有可能需要对同一份数据（勾选多种提醒方式的用户）进行不同的处理（邮件通知 or 电话通知 or 短信通知）

分离：将前面过滤出的用户进行分成3组，分别对应（邮件通知 + 电话通知 + 短信通知）

请大佬指教，理解有误没

作者回复：谢谢大佬的答案！理解基本无误的！



👍 30



缪斯

2019-05-01

用户注册需要用到合并模式（不同客户端），购票过程需要用到过滤模式（对时间地点进行筛选过滤选票），提醒需要用到分离模式（进行不同渠道的分发提醒通知，如短信，电话等）。

作者回复：谢谢你的答案！



👍 8



vigo

2019-09-28

秦始皇统一六国过程中都使用什么模式！



👍 5



nuclear

2019-06-05

感觉合并模式可能会有问题，如果两个流有差速怎么办？

作者回复：谢谢你的提问，这个问题问得很好！对于有边界数据来说，也就是批处理，如果是读取两个流有速度上的误差是没有关系。如果你问的是无边界数据，那种有无止境数据的流处理的话，这里需要要求你在数据完整性和结果的延时上作出取舍了。具体的内容我会在介绍Beam的时候讲到。



👍 5



潘腾

2019-07-27

就spark应用而言，过滤模式可以通过filter实现，

合并模式可以通过join实现。

至于复制模式，一般来讲一个rdd在后续处理中被多次使用到，应该就算是复制模式了吧，为了提高效率，可以使用persist持久化。

分离模式就是groupByKey吧。

这四种模式在spark中还有没有其他的实现方式呢？



👍 2



linkzhang

2019-06-02

极客星球评论功能不好用啊（汗）

请问老师，看到很多回答里面都提到，提醒功能需要用到复制模式，但我理解只需要过滤和分离，过滤出需要提醒的用户后，如果一个用户需要多种方式通知，在分离的过程中是不是已经隐含了复制数据，不然上面的例子中，一个数据无法通过分离模式输入到两个处理模块

作者回复：谢谢你的留言！其实还是需要用到复制模式的，分离的过程只是将用户划分在不同的类别中。如果同一个用户需要两种或以上的方式通知，那我们就需要把同一个用户采用复制模式复制到不同的模块中去处理了。



👍 3



linkzhang

2019-06-02

请问老师，看到很多回答里面都提到，提醒功能需要用到复制模式，但我理解只需要过滤和分离模式，过滤出需要提醒的用户后，如果一个用户勾选了多种通知方式，在分离的过程是不是已经隐含了复制数据，不然前面例子中



👍 1



坤剑

2019-05-09

workflow是否只适用于批处理？

作者回复: 谢谢你的提问! Workflow作为一种数据处理设计思想是既适用于批处理也适用于流处理的。



👍 1



james

2019-05-04

题目用mq可以搞定，没啥模式信手拈来

作者回复: 谢谢你的答案! 就问问你mq消息发丢了怎么办?

共 2 条评论 >

👍 1



明翼

2019-05-01

注册如果多个系统对新用户都处理就复制，如果按照区域注册可能是分离模式，购买不知道是否有根据会员等级提供不同服务的如果有那就分离，买不同地方这个高并发先过滤到不同机器? 至于通知，合并模式多个购买渠道信息合并一起通知所有用户

作者回复: 总结得不错!



👍 1



朱同学

2019-05-01

如果用户从注册到购买到提醒是一个工作流的话，那么注册到购买是合并模式，因为并发的购买请求可能需要进入队列排队，到提醒的话，考虑到推送实时性，我会选择分类模式，如果系统按照整分钟推送，我会将未来几天的每分钟作为一个分类，下单处理完成后，我会把新的订单集合通过复制模式分发给不同的处理分支，推送只是其中的一个分支，推送提醒处理就是把订单分到以分钟为单位分类中，到了整分推送时间直接推送对应的分类即可。

作者回复: 谢谢你的答案，分析得很不错!



👍 1



技术修行者

2020-11-09

看了大家的留言，感觉分析的好全面！

用户注册，考虑不同类型的接入端，使用合并模式。

购票，过滤模式加合并模式。

航班提醒，分类模式。

共 1 条评论 >



Siping

2020-04-04

这个系列不错



Eden2020

2020-03-15

过滤模式和分离模式，针对不同用户进行分离处理，分离前先过滤出不同用户



wang

2020-01-06

思考题：我的理解是，还是一个本质，数据集的来源，和数据集的去向去判断设计模式，用户注册的行为可以涉及到的数据集是一个实时首页的航班信息展示，那么是filter mode，从全量的航班信息数据集中找到当前时间段的满足需求的数据集，购买机票可能涉及到就是某一个条件的数据集展示，那么可以是filter mode，或者是正对一个相对全量的数据集，进行多维度的分离，比如按照时间段的处理到一个数据集，又或者按照价格处理到一个数据集，那么就是splitter mode，出行前的24小时提醒，数据集来源是所有购买了票的用户，通过时间段的处理filter 到满足某一个时间前24小时这个条件的用户集合，然后进行提醒，这是filter mode，除此之外，对于join mode 的划分还是取决于是否有多个数据源集合的条件，比如一个航空系统的航班信息的设置，是需要收集各种维度的数据决定的，比如其他航空公司的路线数据，天气等数据，这些数据的某一些相同的维度可以join 进行设置该系统的航班信息。



风中花

2019-06-26

老师这么一分析，我们这个机票预定流程。就包含了四种模式。真是仔细一想，还就是这个模式。不学习，永远不知道有这样的模式，我们一直在走着这样的模式。真是生在模式中不知模

式。



风中花

2019-06-26

看来同学们得分析，确实学到了！ 购买机票 过程是登录->验证有效性->查询 -> 选择->验证选择-验证用户有效性->购买 结束 。 登录和验证 就是一个过滤 查询 就是一个先复制 因为一份查询到多个平台拿数据。其次过滤 在合并 返回给用户 。用户选择如果不同平台数据必将涉及一个 多平台得预定 所以有一个 数据分离到不同平台预定 然后返回合并返回给用户。机票预定我想也就这么多流程了，至少我们现在是这样的，哈哈



长大的肚腩

2019-05-07

可能原始数据源的物理存储位置不同需要用合并模式，但如果针对这个注册的场景，不同渠道我们一般不用模式吧，直接一个canal同步数据库binlog日志到MQ就可以了。

作者回复: 谢谢你的留言！如果只针对注册的场景确实可以有不同的处理方式，如果用MQ的话你可能还需要log一些失败的case再重新处理。

