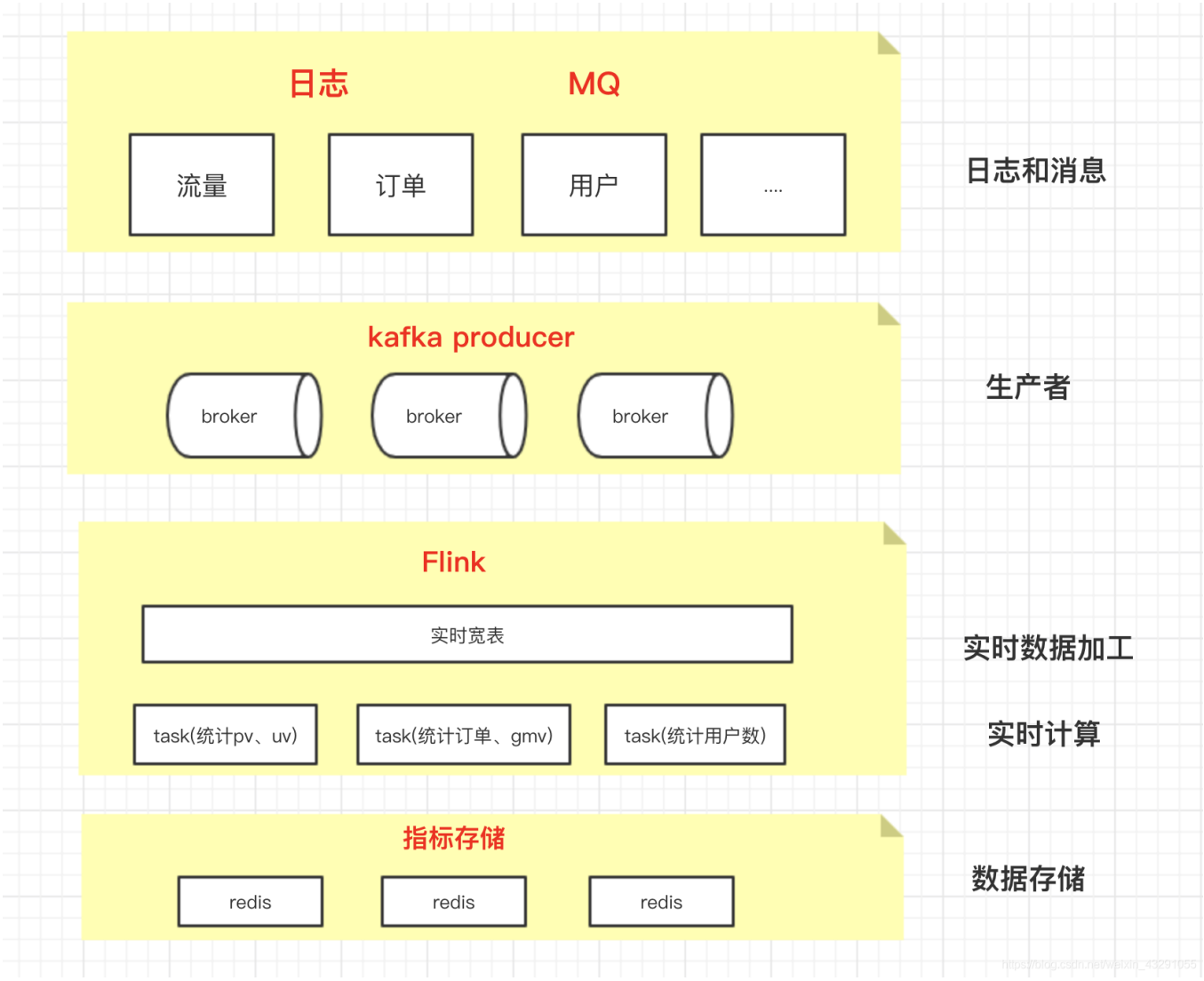
Flink实时计算指标对数方案

对于一个实时数据产品人员、或者开发人员来说，产品上展示的实时数据，pv、uv、gmv等等，怎么知道这些数据是不是正确的呢？当其他的小组开发的产品的数据(或者其他的数据提供方)又是另外一个数字，那么究竟该如何判断自己的数据还是别人的数据是正确的呢？这就需要一套实时数据对数方案，本文主要从背景、实时数据计算方案、对数方案、总结四方面来介绍，说服老板或者让其他人相信自己的数据是准确的、无误的。

一、背景：

相信做过实时数据统计的朋友，肯定会遇到一个问题，怎么知道自己算的数据是不是对的呢？比如：pv、uv、dau、gmv、订单等等统计数据。

二、实时数据统计方案



上述流程图描述了一般的实时数据计算流程，接收日志或者MQ到kafka，用Flink进行处理和计算，将最终计算结果存储在redis中，最后查询出redis中的数据给大屏、看板等展示。

但是在整个过程中，不得不思考一下，最后计算出来的存储在redis中指标数据是不是正确的呢？怎么能给用户或者老板一个信服的理由呢？相信这个问题一定是困扰所有做实时数据开发的朋友。

比如说：离线的同事说离线昨天的数据订单是1w，实时昨天的数据确实2w，存在这么大的误差，到底是实时计算出问题了，还是离线出问题了呢？

三、对数解决方案

为了方便理解，还是拿上面离线和实时的下单金额为例。

某电商双11实时数据大屏最终展示的GMV是200亿，小李当晚汇报给老板，双11GMV是200亿。第二天晨会，离线的同事小王汇报给老板，双11GMV是300亿。同时又有一个数据部门的同事小赵说，我们这边计算的是192亿。老板听到这么多数据，一瞬间就不知道该相信谁的呢？然后就说，小李、小王你们两数据差距最大，你们对一下吧，汇报我一个最终结果。

于是，小王看着自己数据告诉小李：某人在我们平台下了30个iphone x合计多少钱、某人又在我们这里买了10台联想笔记本电脑合计多少钱 .....

小李看着最终展示在大屏上的200亿GMV，瞬间就蒙了，心里想道：我这里不知道谁买了多少个iphone呀，也不知道他们花了多少钱呀?

于是小李回去请教了自己的导师，导师说你把上面的实时宽表数据存储下来，就可以和他们对了，就知道谁买了多少个iphone x了，谁有买了多少个联想电脑了。

小李想了想，按照导师的思路开发如下的宽表加工方案：

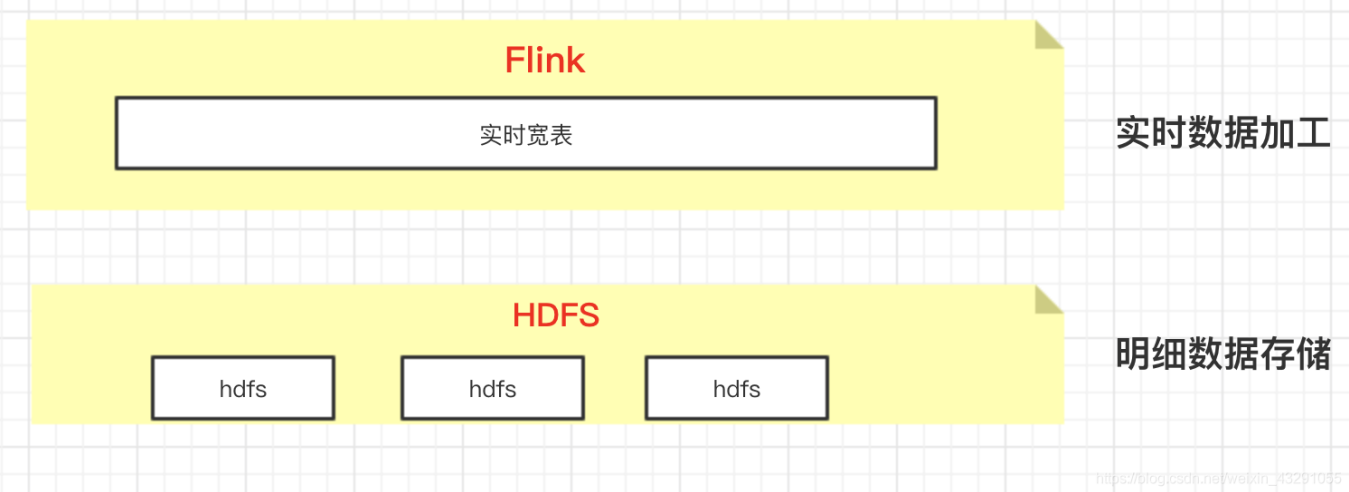
(1)用Flink将实时宽表数据存储至elasticsearch

将加工的宽表数据通过Flink写入es，这样可以得到所有数据的明细数据，拿着明细和其他数据提供方进行比对即可。

(2)用Flink实时宽表数据存储至HDFS，通过Hive进行查询

但是有一些朋友可能会说，es对应的sql count、group by语法操作，非常复杂，况且也不是用来做线上服务，而只是用与对数，所以时效性也不需要完全考虑，这样的话，就可以考虑将数据回写至HDFS了。

因此可以考虑采用下图的方案，将加工的宽表通过Flink写入到HDFS，然后新建hive表进行关联HDFS数据进行关联查询。



写HDFS与es相比，存在非常明显的优点：

a.学习成本低、会sql的基本就可以了，而不需要重新学习es负责的count、group by 等语法操作

b.可以非常方便地和离线表数据进行关联查询(大多数情况下都是和离线数据比对)，两张Hive表的关联查询，容易找出两张表的数据差异

最终小李拿着自己存储的明细数据和小王对了一下，发现是小王的口径不一样，没有排除一些预售订单，最终小李将汇报给老板，得到了老板的嘉奖。

四、总结

实时计算能提供给用户查看当前的实时统计数据，但是数据的准确性确实一个很大的问题，如何说服用户或者领导数据计算是没有问题的，就需要和其他的数据提供方进行比对了。问题的关键就在于，只要有明细数据，就可以和任意一方进行比对，毕竟有明细数据。不服？我们就对一对啊。

明细数据的存储、设计也很有讲究，可以和离线或者其他提供方的数据字段进行对齐，这样就非常方便进行比对了，而采用hive这种方式又是最简便的方式了，毕竟大多数人都是会sql的，无论开发人员还是数据人员或者BI人员。