**DMP 项目概况**

# 1 项目背景

网络和智能终端的普及带来了海量人群的上网行为数据，大数据技术的发展让细分人群的特性成为可能。不同的人群，网络行为习惯差异性比较大，女性、男性互联网行为不一样；25 岁大学男毕业生、

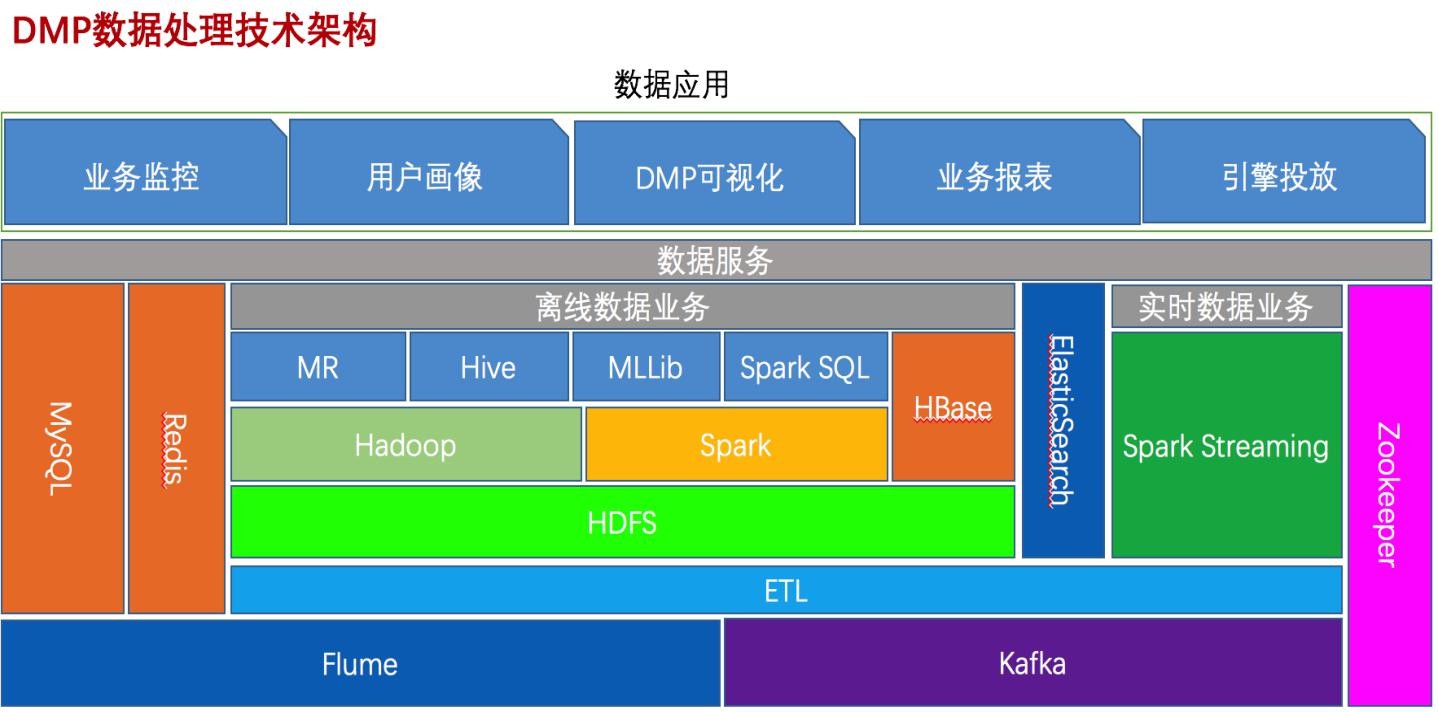
35 岁政府白富美更不一样。

服饰行业、电子类产品快速消费的互联网广告特性和房地产的广告特性的差异不言而喻。行业不同，广告投放也需要更精准的 DMP。

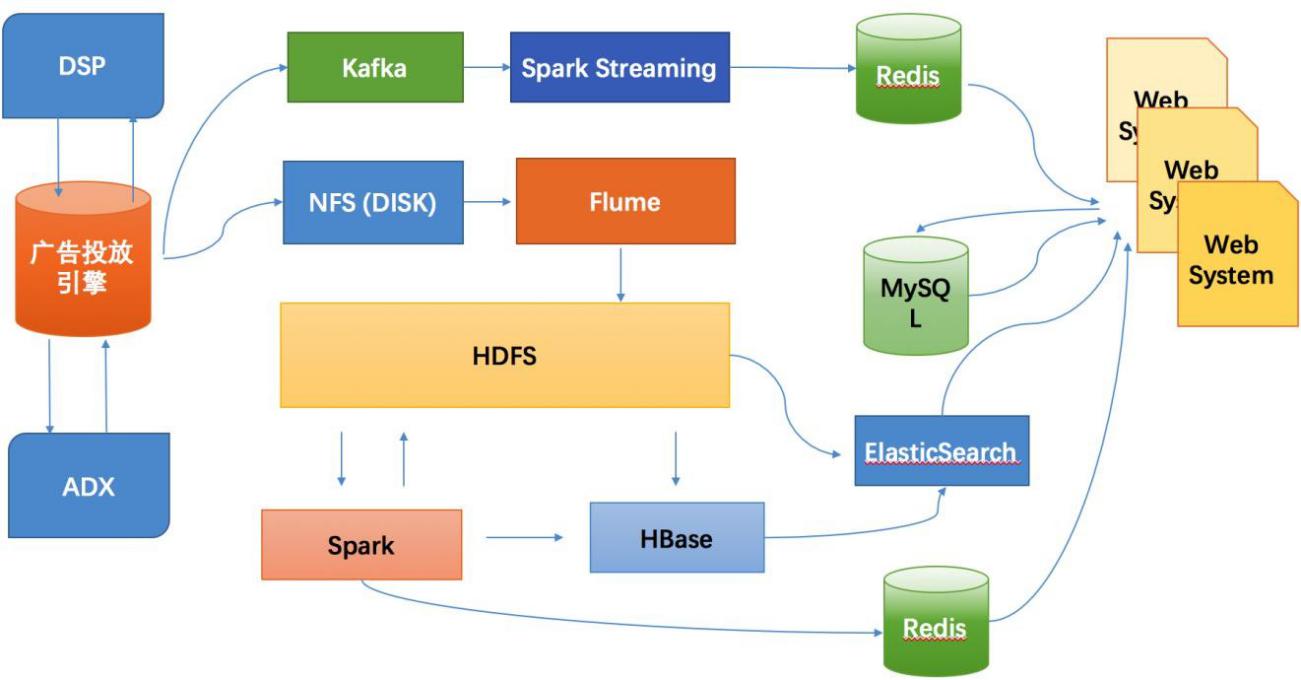
时间、地区、季节、节日、天气… …大量的因素，大量的数据交织在一起，没有细分研究，对于大数据的分析，难以为广告主带来高价值。细分研究各类因素大数据，为广告主创造更高价值。

最终节省成本，精准投放广告。

# 项目架构图（掌握）



1. 项目整体业务流程图(必须能手画出来，并且还要能说出 来)



1. 项目都分析过哪些业务模型（必须讲出来，必须熟悉业务模型）
   * 实时报表（展示量、点击量、点击率、参与竞价数、竞价成功数、消费）
   * 终端设备分析(运营商、联网方式、操作系统、客户端)
   * 渠道分析
   * 媒体分析
   * 统一用户识别
   * 数据标签化
   * 用户上下文标签合并

# 项目中知识点

1. Parquet 文件存储（列式存储）

## 可以跳过不符合条件的数据，只读取需要的数据，降低 IO 数据量。

* 1. 压缩编码可以降低磁盘存储空间，由于同一列的数据类型 是一样的，可以使用不同的压缩编码。
  2. 只读取需要的列，支持向量运算，能够获取更好的扫描性 能。
  3. Parquet适配多种计算框架，查询引擎（Hive、Impala、pig、

IBMBigSQL等等）；计算框架（MapReduce、Spark、Kite、

Cascading等等）；数据模型（Avro、Thrift、ProtocolBuffers 等）

1. Spark 算子运用

## map、flatMap、filter、case、groupBy、reduceByKey等。

1. 地理位置GEOHASH 算法

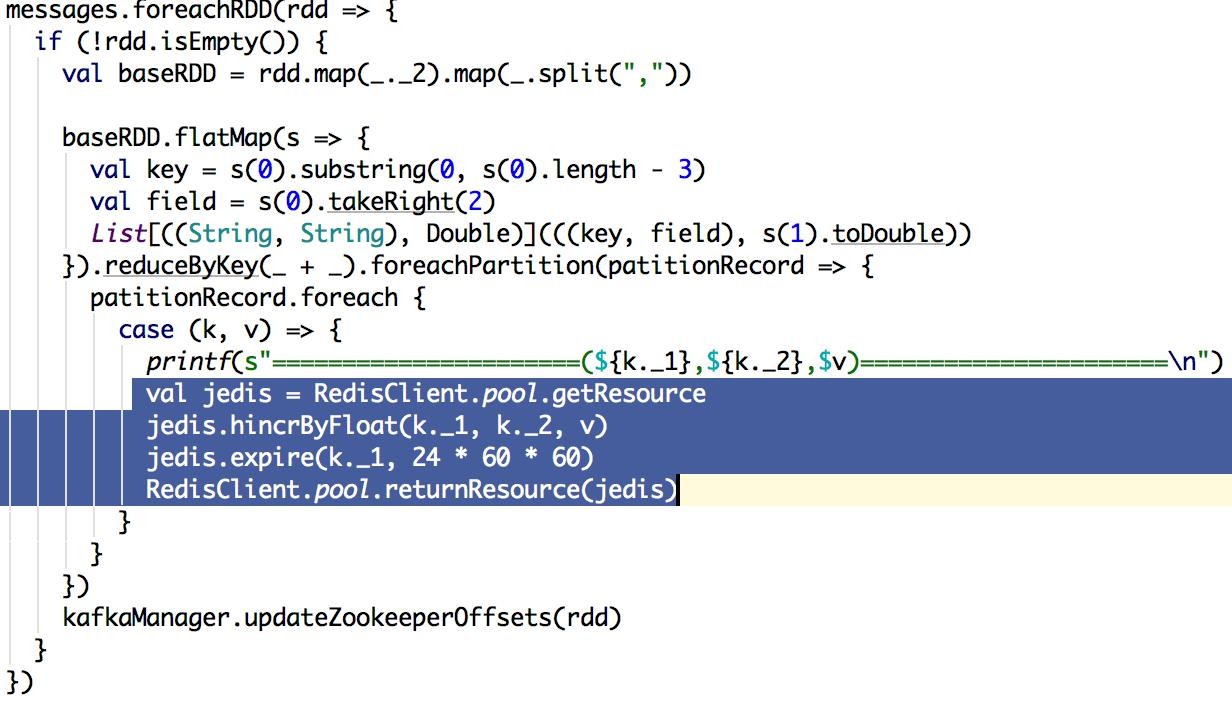




1. 多模式匹配（AC 算法）
2. 用户画像的流程

# 用了哪些技术框架，都遇到过什么问题？

1. Spark 序列化问题？
2. 类字段过多导致异常？extends Product with Serializable
3. 正确的理解分布式程序



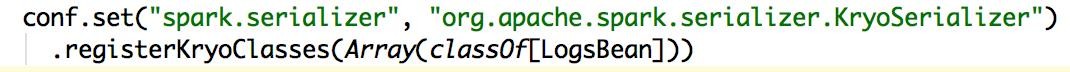
1. 数据倾斜，某个task 数据量过大。

# 软件版本

Hadoop2.7 Hive1.2 Zookeeper3.4 Spark1.6 Hbase0.98.x Scala2.10.x Flume1.5 Kafka0.9 ElasticSearch2.3

# 调优经验

* Spark 自带的任务监控，查看Stage 划分，对代码优化
* 使用 Kryo 序列化，因为 Kryo 序列化结果比 Java 标准序列化更小，更快速。

在 spark-default.conf 文 件 中 设 置 : spark.serializer：org.apache.spark.serializer.KryoSerializer

* Spark OOM ： java heap space ， OOM:GC overhead limit

exceeded 解决方法：在 spark-submit 中指定 --driver-memory

memSize 参数来设定 driver 的 jvm 内存大小。

* spark 通过合理设置 spark.default.parallelism 参数提高执行效率，一般每个 partition 对应一个task。在我的测试过程中，如果没有设置 spark.default.parallelism 参数，spark 计算出来的

partition 非常巨大，与我的cores 非常不搭。

* spark.storage.memoryFraction 默认值 0.2 ， 该参数代表了

Excutor 内存中，分配给 shuffle read task 进行聚合操作的内存比例，默认是 20 。如果内存充足，而且很少使用持久化操作， 建议调高这个比例，给 shuffle read 的聚合操作更多内存，以避免由于内存不足导致聚合过程中频繁读写磁盘，在实践中发现，合理调节该参数可以将性能提升 10 左右。

# 写了多少代码

每个scala 文件中代码行数大概在 50-100 行，项目一共有 40-50 多个scala 文件。

# 项目生产环境介绍

集群：

一共 8 台，4 台都为 40 核，120G 内存；4 台 4 核 32G 内存。数据量：

每天有效数据 200-300w，采用 gzip 压缩方式， 每天数据总量大概 150-250G。