

# 京东个性化推荐技术实践



推荐搜索部 王志勇

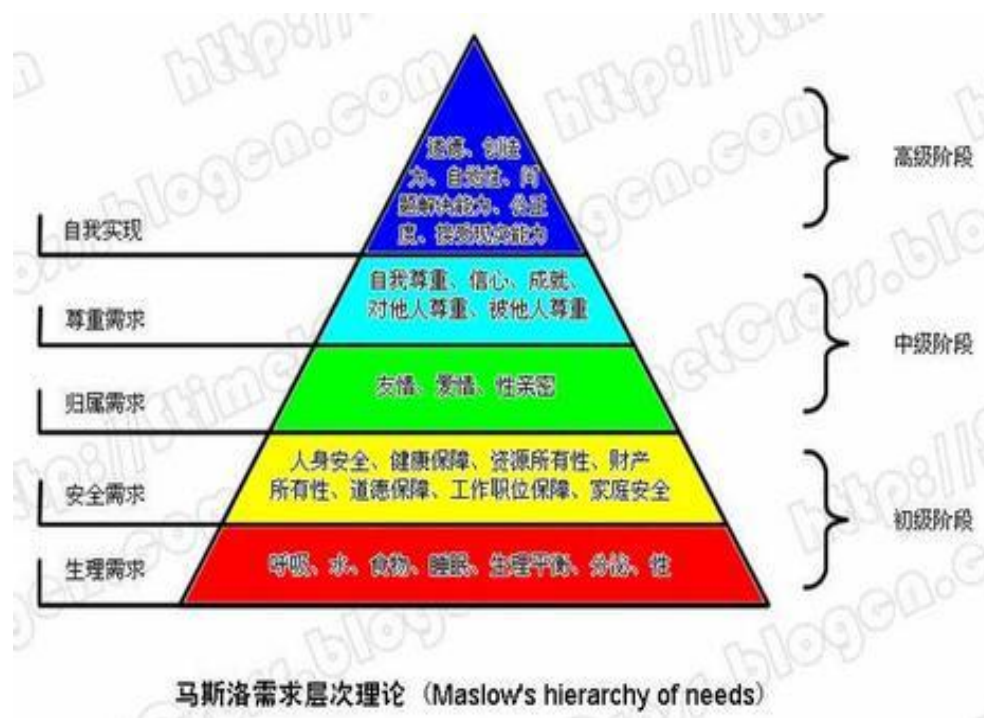
2014-09-27

- **个性化推荐简介**
  - 为什么需要个性化推荐
  - 个性化推荐为什么需要实时
- **个性化推荐系统的实践**
  - 个性化推荐面临的问题
  - 京东个性化推荐的实践方案
  - 应用与效果分析

Personalized Recommendations

# 个性化推荐简介

- 幸福，就是自己的需求被满足



# 信息过载 VS 用户注意力资源

京东有近200,000,000商品可选购，每天新增商品超过500,000



# 长尾理论 --- 个性化时代的到来



- 突破丰富的有限
- 突破传统的生产成本和流通成本





# 个性化推荐的作用



提高用户忠诚度和用户体验，提高用户购物决策的质量和效率

提高成交转化率(CTR,CVR,GMV)

提高网站交叉销售能力



# 个性化推荐需要解决的问题



合适的内容(what):

商品/产品  
店铺/品牌  
活动

合适的用户(who):

企业用户  
个人用户  
不同用户群

个性化推荐

合适的地方(where):

首页  
搜索页  
商详页  
过渡页  
订单页

合适的时机(when):

进入首页  
Query/List搜索  
点击浏览  
加入购物车  
加入关注/收藏  
完成交易

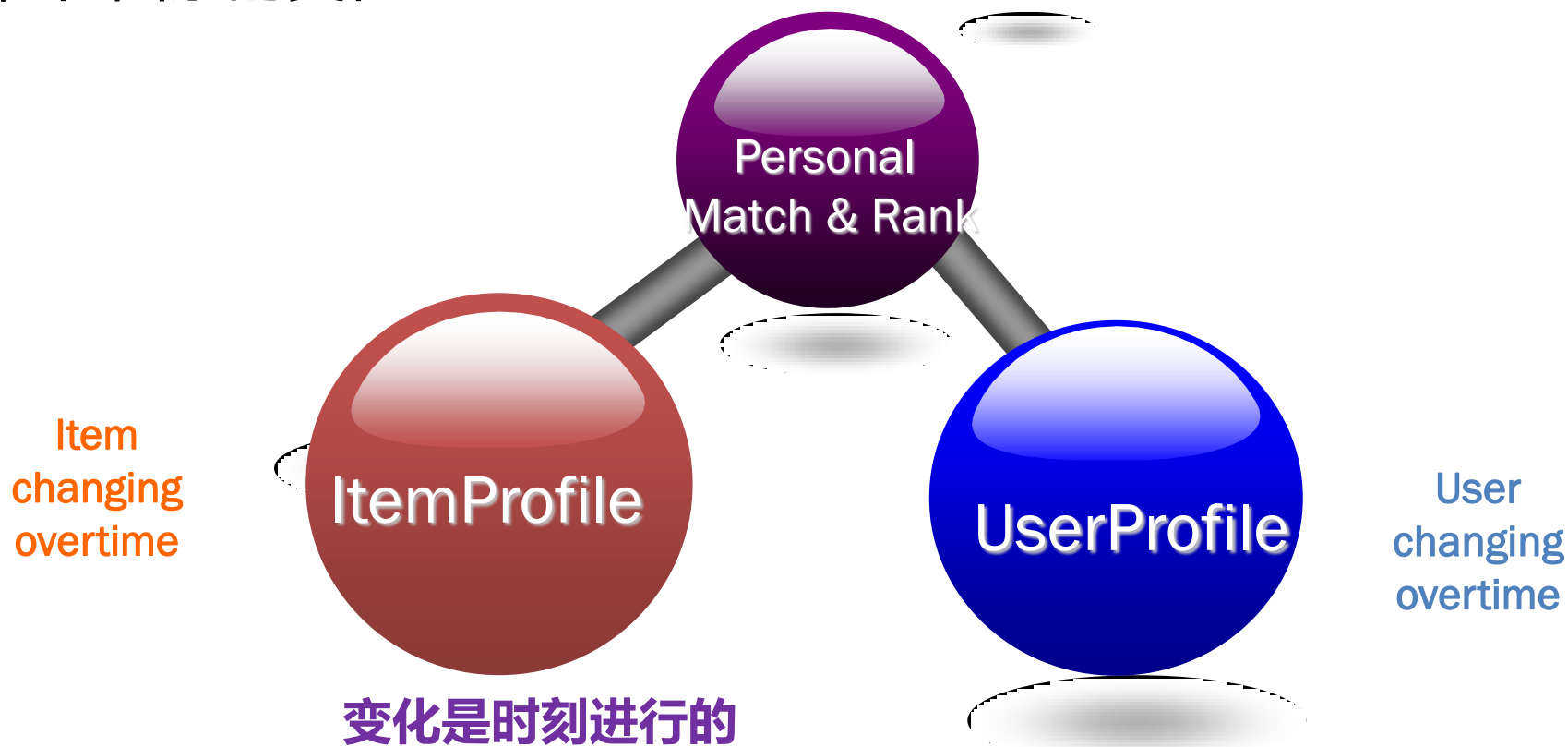
合适的渠道(how):

PC  
APP/微信/手Q  
EDM



# 个性化为什么需要实时

- 个性化因素的变化



**变化是时刻进行的**

商品在变化

用户个体在变化

群体、环境在变化

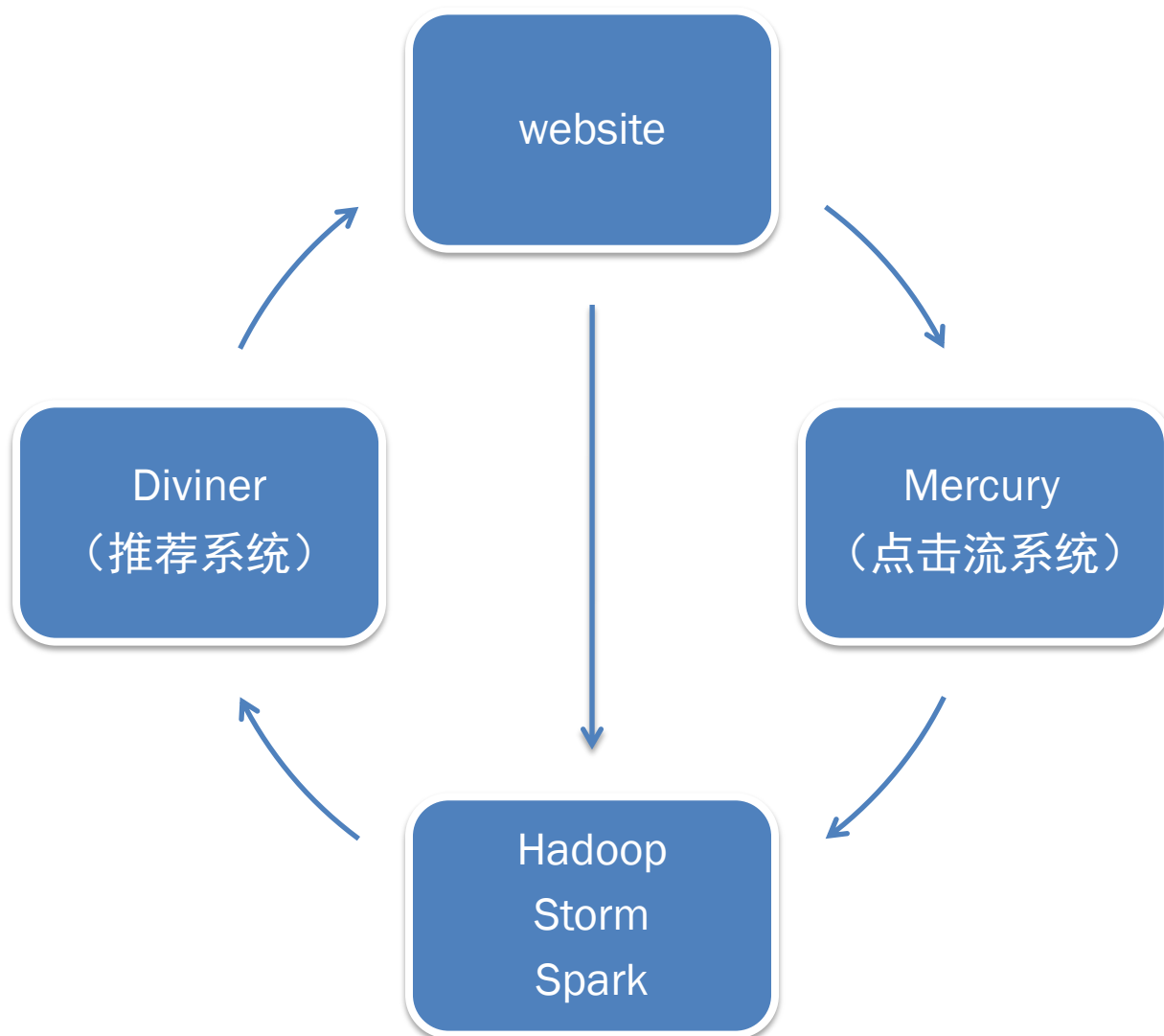
个体和群体的隶属关系也在动态变化

Personalized Recommendations

# 个性化推荐实践

- **处理持续增长的大数据能力**
- **实时分析用户购物意图能力**
- **大规模稀疏数据建模能力**

- 数据收集/存储
- 离线/在线分析
  - 数据清洗
  - 数据建模
    - 商品建模
    - 用户建模
- 在线推荐
- A/B测试平台



# 数据收集(二) 用户数据打通



★收藏京东 北京 [更换] jd\_fyw00 [请登录] [免费注册] | 我的订单



京东白条 搜索

热门搜索: 校园之星 游戏频道 空中营救 李维斯 婚博会 170靓号 童书满减 京东白条

全部商品分类

首页

服装城

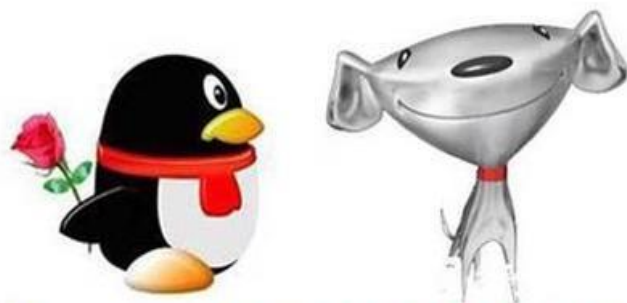
食品

团购

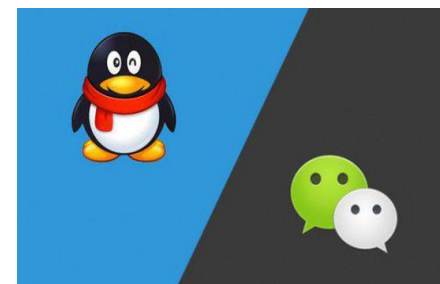
夺宝岛

闪购

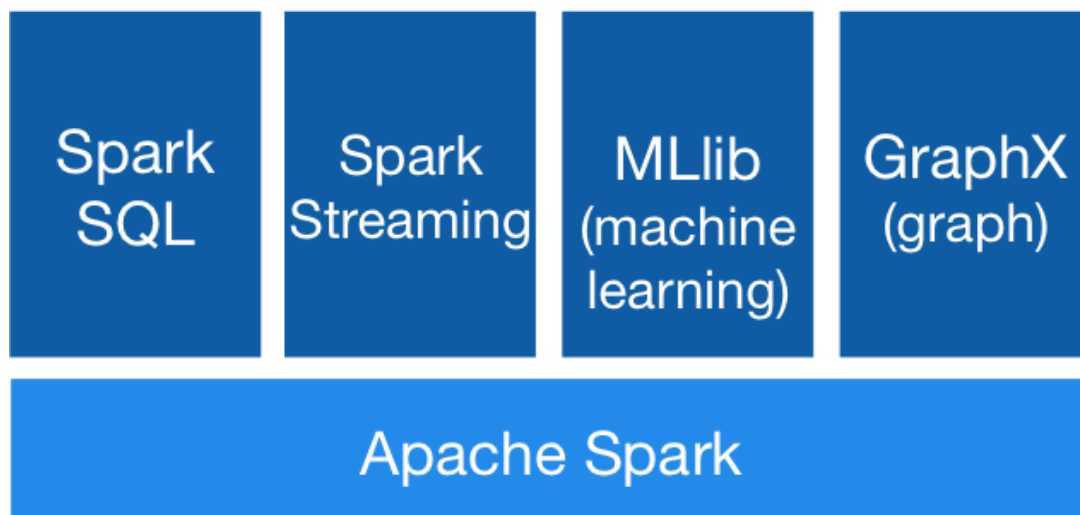
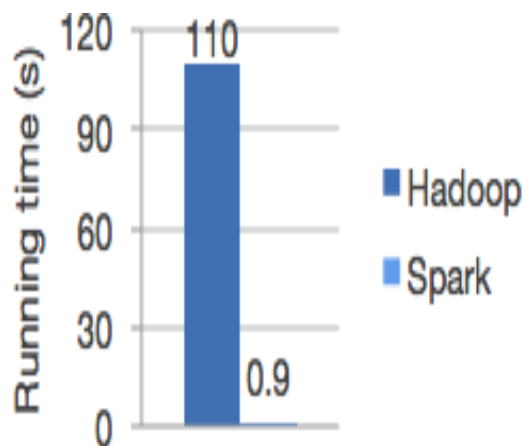
金融



Tencent 腾讯 JD.COM 京东

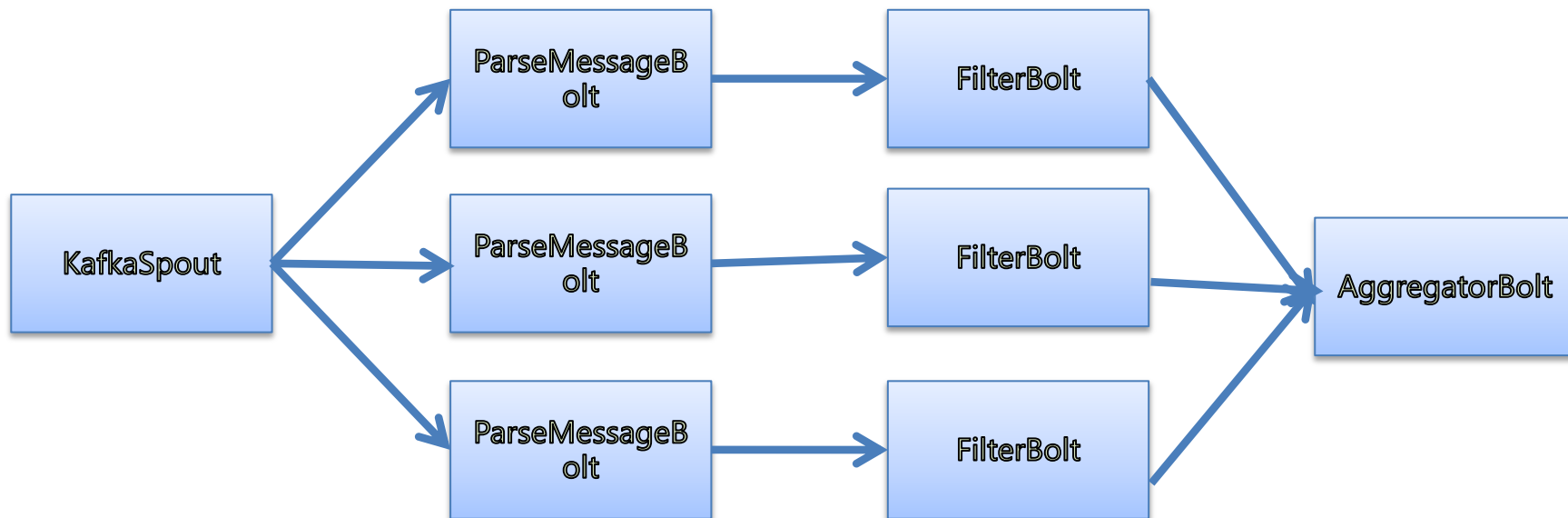
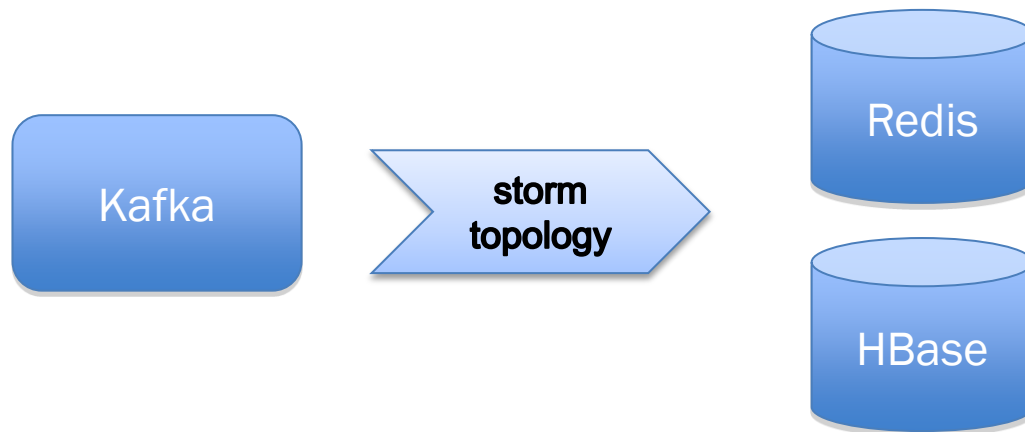


- 离线计算主要在**Hadoop**上运行**Map Reduce**
- 部分计算都使用**Mahout** , **Spark**





- 在线计算主要基于Storm，实时消息基于kafka(30亿+)。



- 数据为王，垃圾进垃圾出
- 现实数据是“肮脏”的
  - 数据残缺
  - 数据重复
  - 数据不一致
- 商业数据更“肮脏”
  - 点击作弊
  - 订单作弊
  - 交易作弊
  - 评论作弊.....



# 数据建模(一) ---- 商品画像



属性

品牌

产品

颜色/尺码

风格/材质

适用人群

图片...



流量

PV

UV

CTR

Load Time

Exit Rate...

Apple苹果 全掌气垫运动鞋 男跑步鞋情侣鞋透气休闲鞋 土豪系列AP-5S 情侣款 黑绿桔红

销售

销量/销售额

退/换货率

价格指数

物流速度

促销类型

售后服务

评论

- **相似商品挖掘：**
  - 基于内容
    - LDA
    - SimHash
  - 基于用户行为
    - Session浏览商品的CF
    - .....
- **相关商品挖掘：**
  - 基于商品的FP-Growth
  - 基于产品的FP-Growth
  - 基于图扩展
  - .....

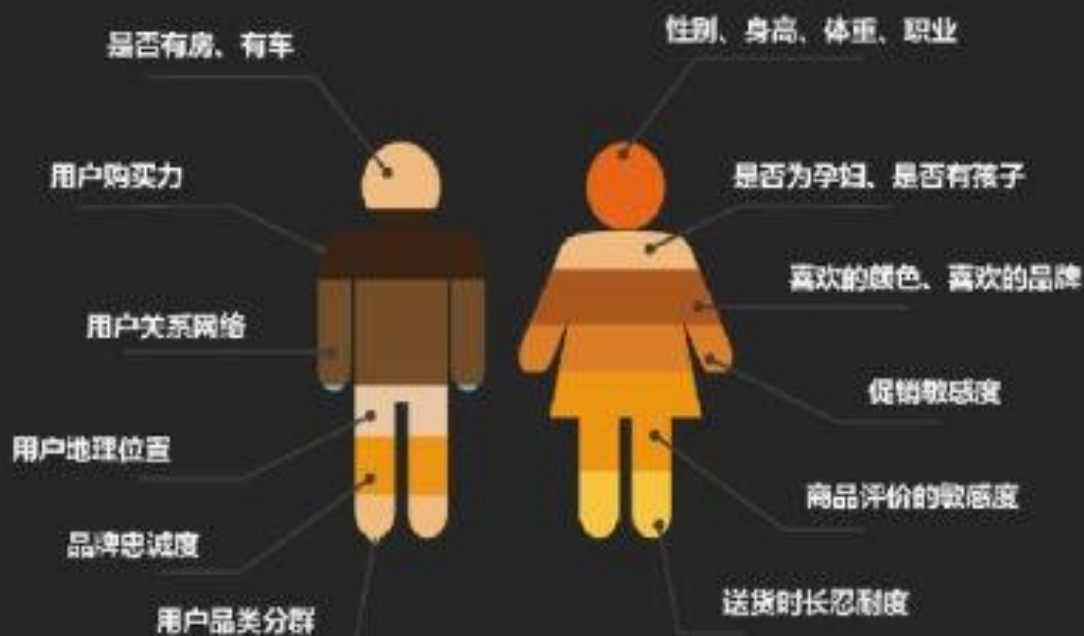
- **四大挑战**

- 商品量大 --- **2.0亿 +**
- 并发量大 --- **QPS 1.5~2万**，日请求约**10亿**
- 易变数据实时性 --- 库存，价格变化频繁
- 性能要求高 ---- **TP999 < 10ms**

- **解决方案 --- 京东缓存云**

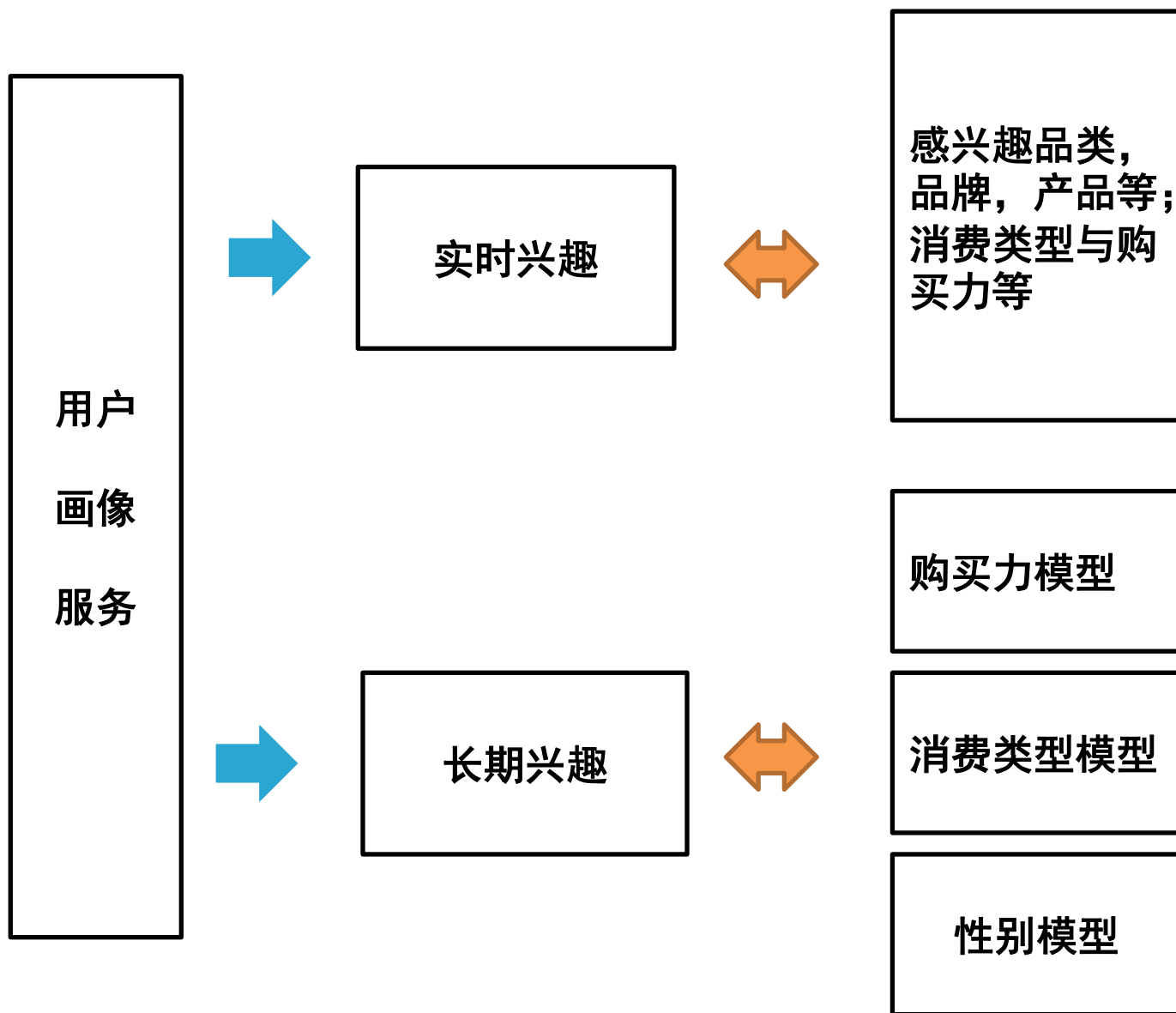
- ShardedJedis+ShardedJedisPool
  - 自动管理连接池，支持多个分片的独立连接池；
  - MasterFirst + MasterOnly

# 数据建模(二) ---- 用户画像



- **用户品牌兴趣度模型：**
  - GBDT + RankSVM
- **用户购买力模型：**
  - 基于Mahout的ALS-WR
- **用户品类兴趣度模型：**
  - 基于规则 优于 SVD++
- .....





Model Test

Tasks

用户pin

fyw004

三级分类	三级分类	权值
1396	套装	0.083
6286	立体拼插	0.083
6280	积木	0.083

品牌	品牌	权值
1396:12278	套装:曼秀雷敦 (Mentholatum)	0.083
6286:11116	立体拼插:乐高 (LEGO)	0.083
6280:11116	积木:乐高 (LEGO)	0.083

产品关键词	产品关键词	权值
1396:49406	套装:洁面膏	0.083

Display Server

Figure

Display

fyw004

查询

☒ 显示字段描述 ☐ 显示服务器反馈信息

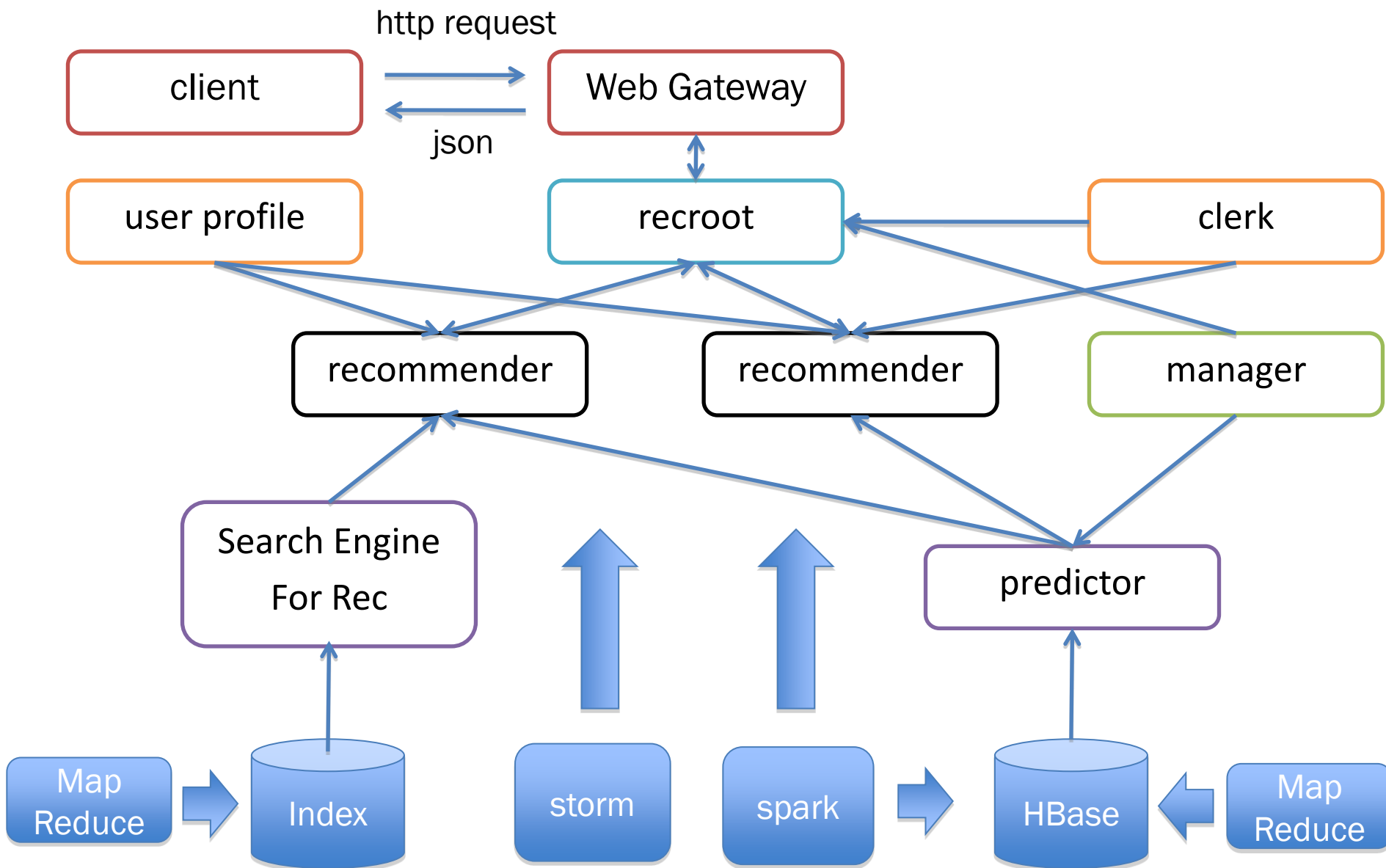
表名:portal\_pin,列名:df.dm\_user\_model

```
userKey {
  serviceName: "portal_pin"
  id: "fyw004"
}
userModelProto {
  demography {
    用户名: "fyw004"
    是否有小孩: true
    孩子性别得分: BOY_GIRL
    性别: FEMALE
    性别置信度: high, medium, low: "high"
    生命周期类型: DECLINE
    用户价值分组: VG_VERY_HIGH
    标准化价值得分: 97
    用户下单最多的省份: "北京"
  }
  shoppingFeatures {
    用户购物类型: RATIONAL
    用户促销敏感度: HIGH
    关注商品评价模型: HIGH
    颜色偏好top1: WHITE
    颜色偏好top2: BLUE
    用户购买订单里最多的一级品类名称: "服饰鞋帽"
    用户最偏好的三个品牌: "U.S.POLO ASSN., O.SA, 歌莉娅"
    用户品类分群模型: SUPERUSER
    用户忠诚度: LOYALTY
    单品促销敏感度: MIDDLE
    套装促销敏感度: LOW
    团购优惠促销敏感度: LOW
    满返满送促销敏感度: MIDDLE
    用户下单总数: 80
    局域网下单总数: 2
    网吧下单总数: 0
    学校下单总数: 0
    单位下单总数: 53
    家里下单总数: 25
    用户活跃度模型: VERY_ACTIVE
    大家电模型: true
    用户历史购买过订单量: 223
    购买力分段: 1-5从高到低: RICH
  }
}
```

表名:portal\_pin,列名:df.search\_cid3\_interest

```
userKey {
  serviceName: "portal_pin"
  id: "fyw004"
}
userCid3ProperProto {
  用户名: "fyw004"
  用户偏好的三级分类 {
    proper: "11224"裤子
    weight: 100
  }
  用户偏好的三级分类 {
    proper: "11232"凉鞋
    weight: 65
  }
  用户偏好的三级分类 {
    proper: "9249"电炖锅
    weight: 52
  }
  用户偏好的三级分类 {
    proper: "1391"护肤
    weight: 37
  }
  用户偏好的三级分类 {
    proper: "11222"套装
    weight: 34
  }
  用户偏好的三级分类 {
    proper: "9775"拖鞋/人字拖
    weight: 33
  }
  用户偏好的三级分类 {
    proper: "1662"衣物清洁
    weight: 28
  }
  用户偏好的三级分类 {
    proper: "753"电饭煲
    weight: 22
  }
  用户三级分类下偏好的品牌词 {
    proper: "11224:19048"裤子:博士蛙
    weight: 100
  }
}
```

# 在线推荐系统架构



- use **Thrift** as RPC framework
  - multi-language support: Java / C++ / PHP
- use **zookeeper** to register replicated services
  - never interrupt
- **JDNS** (JD naming service)
  - service discovery
- multi-protocol support
- SAF compatible
- high performance / robust

**Monitor**

- ☐ 集群配置
- ☐ 集群监控
- ☐ 节点监控
- ☐ 机器监控
- ☐ 报警设置

**Admin**

- ☐ 报警开关
- ☐ 系统设置

**ZooKeeper集群状态** 更新时间: 2013-09-12 20:32:52 [加入监控](#)

zkmonitor

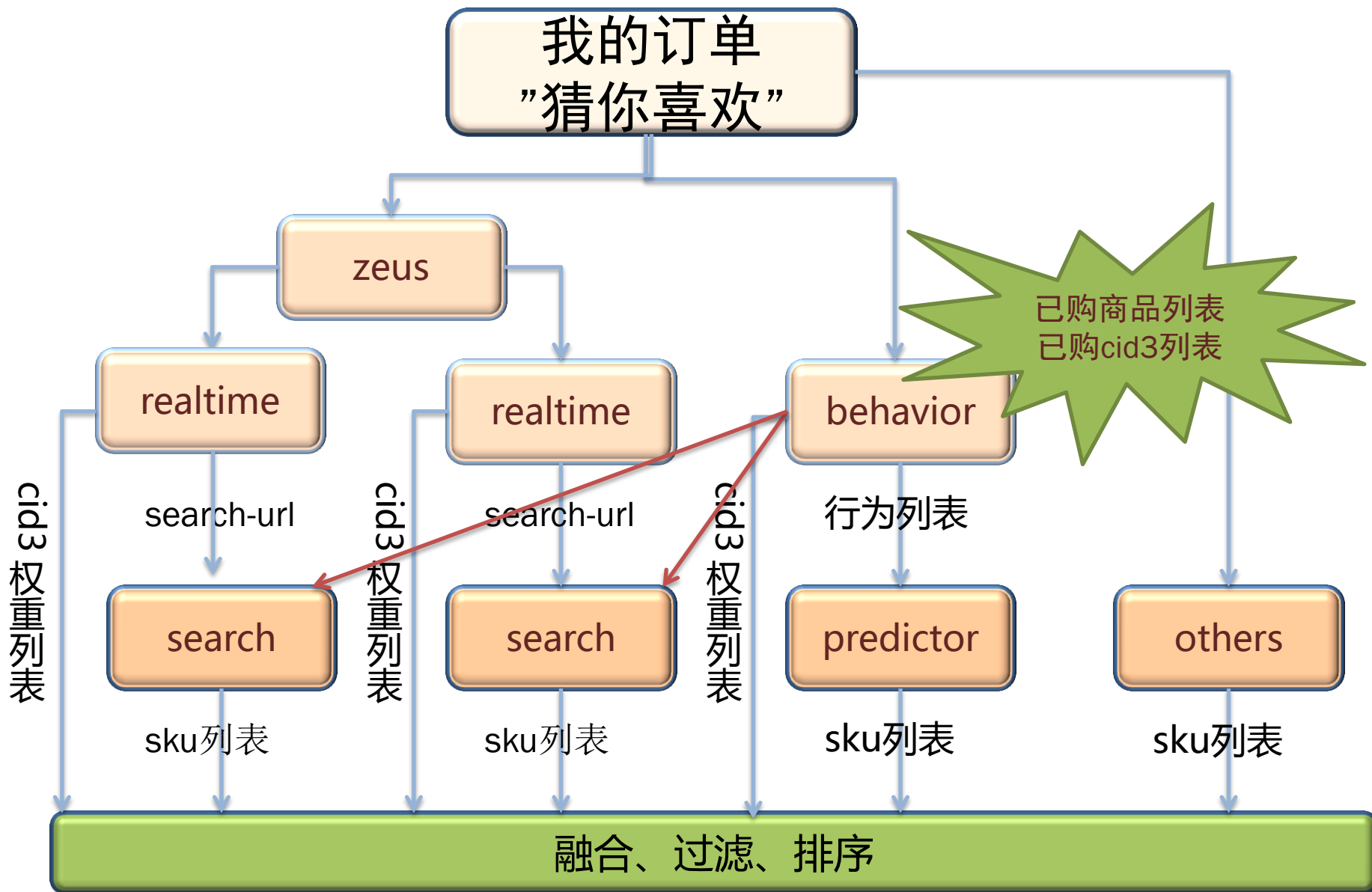
Node IP	Role	连接数	Watch数	Watched / Total Path	数据量 Sent/Received	状态	节点自检状态	查看趋势
172.17.34.145		0	0	0/0	/		OK	
172.17.34.144		0	0	0/0	/		OK	
172.17.34.143		0	0	0/0	/		OK	
172.17.34.147		0	0	0/0	/		OK	
172.17.34.146		0	0	0/0	/		OK	

YINSHI.MONITOR.ALIVE.CHECK

- brokers
- consumers
- diviner
- hbase
- jdns
  - jd
  - mercury
    - si
      - clerk
      - diviner
      - configuration
      - manager
      - predictor
      - monitors
      - rpc
        - { "ip": "172.17.34.148", "port": 18602, "protocol": "TCompactProtocol" }
        - { "ip": "172.17.34.149", "port": 18602, "protocol": "TCompactProtocol" }
        - { "ip": "172.17.34.150", "port": 18602, "protocol": "TCompactProtocol" }
        - { "ip": "172.17.34.151", "port": 18602, "protocol": "TCompactProtocol" }
        - { "ip": "172.17.34.152", "port": 18602, "protocol": "TCompactProtocol" }
        - { "ip": "172.17.34.155", "port": 18602, "protocol": "TCompactProtocol" }
        - { "ip": "172.17.34.156", "port": 18602, "protocol": "TCompactProtocol" }
        - { "ip": "172.17.34.158", "port": 18602, "protocol": "TCompactProtocol" }
        - { "ip": "172.17.34.159", "port": 18602, "protocol": "TCompactProtocol" }
        - { "ip": "172.17.34.160", "port": 18602, "protocol": "TCompactProtocol" }
        - { "ip": "172.17.34.161", "port": 18602, "protocol": "TCompactProtocol" }
        - { "ip": "172.17.34.162", "port": 18602, "protocol": "TCompactProtocol" }

- 召回(recall)
  - 从一个或者多个Predictor中读取候选推荐目标
  - 从Search中召回候选推荐目标
- 过滤(filter)
  - 去掉无货、下架、用户已购商品
- 计分(rank)
  - 多模型融合
  - 在线CTR预估
- 排序(sort)
  - 对N个结果进行排序，取最前面的n个
- 填充(fill)
  - 查询clerk服务，将商品标题、价格、图片等信息补全

## 个性化推荐案例

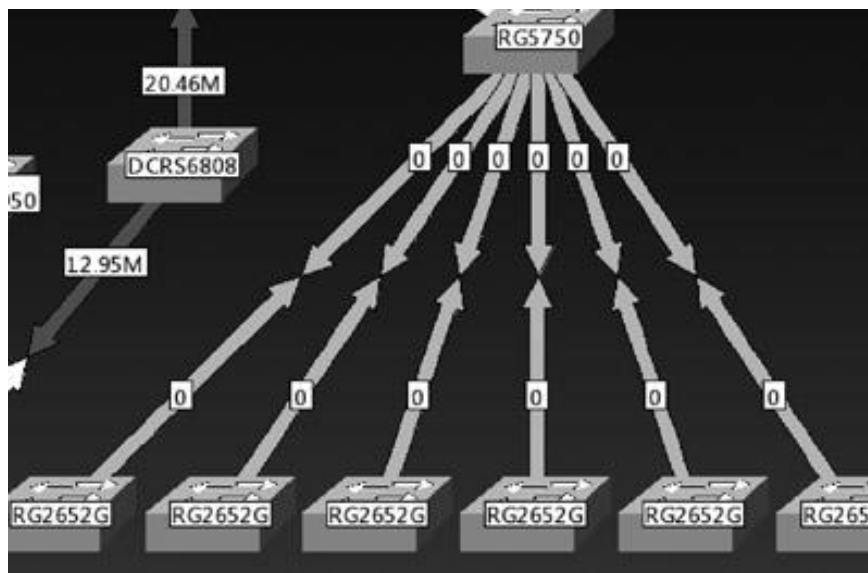




# 京东推荐A/B测试平台

- 无数据，不优化
  - 线上分流实验是进行推荐算法优化的必由之路
- 指标定义
  - CTR
  - 转化率
  - 销售额 (RPM ...)
- 数据收集
  - 点击流系统 (Mercury)
  - 数字签名，防篡改
- 数据处理：
  - Hive (standing query)
  - Shark (ad hoc query)
- 实验管理平台
- 实验报告

- Random
  - 随机分流，用于可变结果集
- Partition By User
  - 按用户切分，同一用户永远看到同样结果
- Partition By Category
  - 按分类切分，针对不同分类测试算法针对性



- 图形界面配置线上分流实验，配置即时生效，避免反复上线
- 增加实验，选择分流策略
- 调整流量，设置生效时间
- 配置追溯和回滚

▼ Diviner Configuration

Mercury Configuration

About

Help

Logged in as [diviner@d.com](#) [Logout](#)

Diviner Configuration [Edit](#)

Last Update	Least Effort	CDN Cache Time	Predictor Cache Time	Active Time	Version
3c	false	5	7200	2013-11-19 11:35:19	364

Placements

104000

Basic Configuration

Id	Name	Create Time	Update Time
104000	关联推荐:图书看了还看(图书商品页)	2013-06-28 17:30:27	2013-11-18 13:48:45

Service Configuration

Default Recommender	Traffic Strategy
/si/diviner/recommender/hub2	Partition By User

Service Parameter

#	Key	Value
1	use_cold_start	false
2	no_order_filtering	false
3	recommender_name	book

Experiments

1040000113102126

Owner:  
diviner@d.com  
Create Time:  
2013-10-21 14:19:46  
Update Time:  
2013-11-10 20:42:46

Service Configuration

Recommender	Experiment Name
/si/diviner/recommender/hub2	sku_browse_buy

Traffic Configuration

Traffic Setting
10.0

Experiment Parameter

#	Key	Value
1	predictor_type	sku_browse_buy
2	recommender_name	book

# 推荐结果持续改进

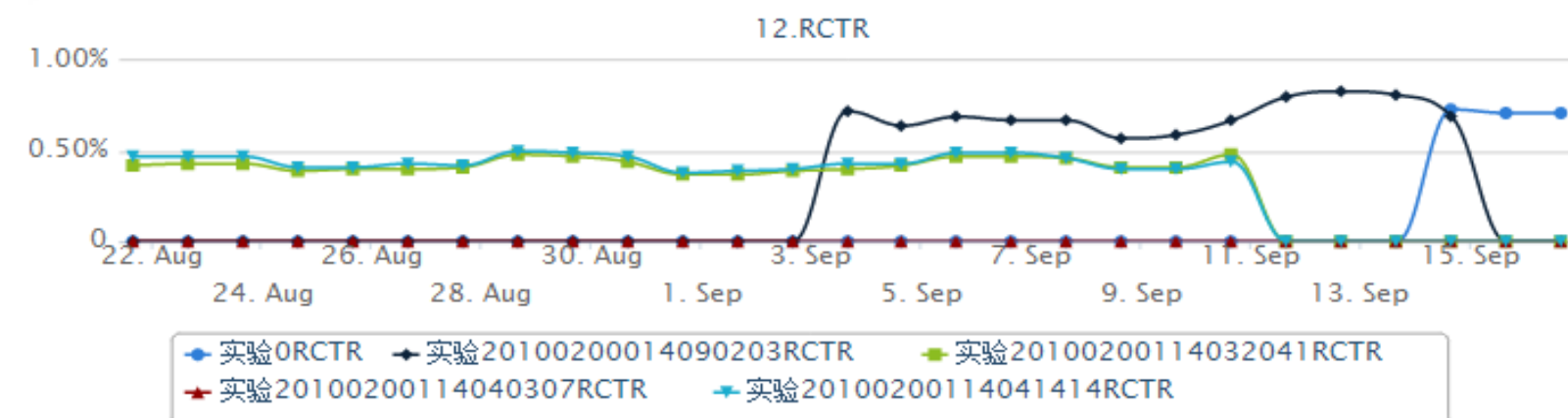
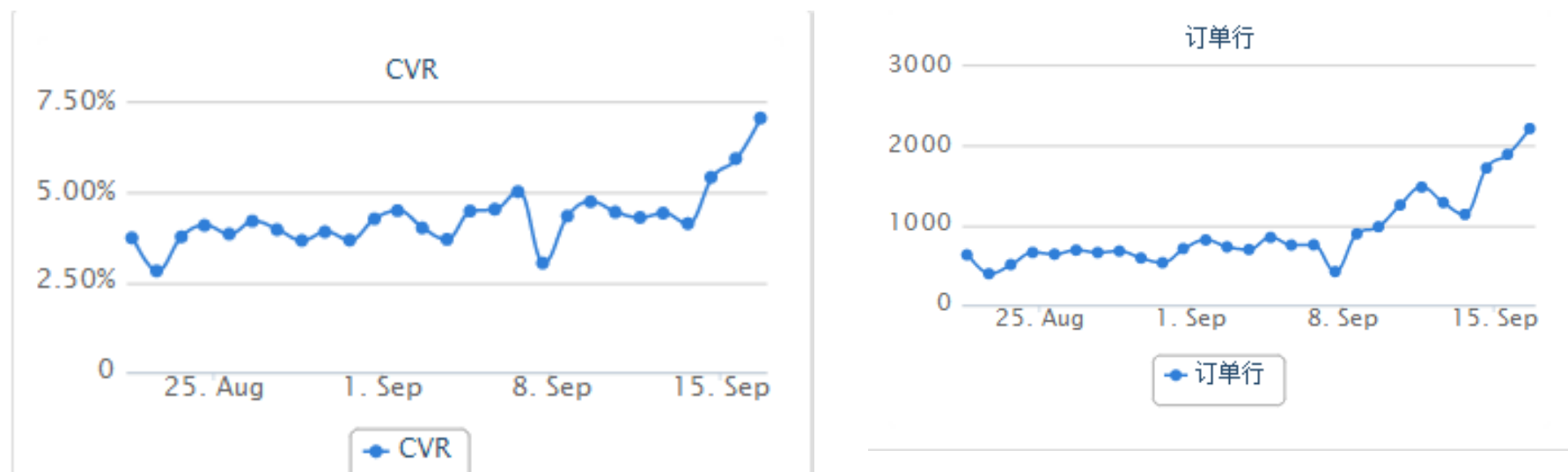


- 利用调试页面，对推荐结果进行内部review
- 将调试页面提供给产品或者业务review推荐结果，给出反馈意见
- 收到反馈意见后，与算法工程师讨论，安排算法优化
- 将优化后的结果进行线上分流实验，并将新结果调试页面发给产品或者业务，供其参照对比
- 收集实验数据，验证算法优化对指标的影响。将实验结果发送给产品或者业务，并解释算法优化的逻辑
- 优化算法正式上线
- 调试页面地址：

<http://diviner.jd.com/diviner?p=104001&uuid=1&sku=10490386&lid=1&lim=8&ec=gbk&fmt=dbq>

The screenshot displays the JD.com recommendation system interface. It features a grid of product cards with titles like '世界奇妙物语' and '金瓶梅'. A blue box labeled '推荐结果' (Recommendation Results) is overlaid on the product grid. To the left, a '参数输入' (Parameter Input) box is visible. Below the product grid, a '调试信息' (Debug Information) box shows a list of parameters and their values, such as 'Recommendation: 104001' and 'Recommendation: 104001'. The interface also includes a '调试页面' (Debug Page) section at the bottom.

# 个性化推荐案例效果





**谢谢！**