

# 【最佳实践】实时计算Flink在IoT行业的实时数仓建设实践

## 行业背景

---

- 行业现状：
  - 物联网(Internet of Things, 以下简称为 IoT)是互联网、传统电信网等资讯的承载体, 让所有具备独立功能的普通物体实现互联互通的网络。物联网广泛应用于运输和物流、健康医疗、智慧环境(家庭、办公、工厂)等领域, 具有十分广阔的市场和应用前景。物联网将智能感知、识别技术、网络通信与普适计算等技术融合起来, 被认为是继计算机、互联网、智能手机之后世界信息产业发展的下一个风口。
  - 据 IDC 估计, 到 2020 年物联网将在全球范围内产生 1.46 万亿美元的价值。据预测, 届时中国的物联网市场规模将超过 1.8 万亿人民币。得益于庞大的人口基数和低廉的芯片制造成本, 中国将成为物联网行业的主要参与者, 并在推动全球物联网市场增长上发挥重要的作用。
- 大数据在其行业中的作用：
  - 数以百万计的物联网设备连接到物联网, 产生了大量的数据, 通过大规模分析这些数据了解影响业务的上下文关系和模式, 从而做出更加实时决策, 因此, 可以说大数据和物联网密切相关。
  - 物联网大数据分析可以帮助人们更好地理解数据, 从而做出更有效、明智的决定。大数据分析使数据挖掘者和科学家能够利用传统工具分析非结构化数据。此外, 大数据分析的目的是利用数据挖掘技术, 提取知识信息, 这些信息有助于进行预测, 识别趋势, 发现隐藏的信息, 并做出决定。

## 业务场景

---

某公司开发了一套针对商场的人流量管理和分析系统。当顾客进入商场后, 固定部署的WIFI探针将实时探测并采集顾客手机或PC终端的WIFI信息。这些顾客WIFI信息经过Flink实时统计和分析后获得商场中客流量、客流量的高峰期、顾客喜欢哪些商店、新客与常客的比例、商场的成交额、顾客的实时位置等等指标。客流量指标可用于数据大屏, 顾客的实时位置指标可用于展示顾客实时密度图, 顾客喜欢的商店指标可用于个性化的广告推送应用等。

## 技术架构

---

架构解析:

数据采集: 该场景中, 数仓的数据来源于WIFI探针探测到的顾客手机和PC终端的WIFI信息, 实时采集至DataHub作为Flink的输入数据。

实时数仓架构：该场景中，整个实时数仓的ETL和BI部分的构建，全部通过Flink完成，Flink实时读取DataHub的数据进行处理，并与维表进行关联查询等操作，统计和分析结果最终输出到RDS和TableStore等存储系统，以供业务系统使用。

## 业务指标

- 场景一：客流量统计
  - 每日商场客流量
  - 每日商店客流量
  - 顾客的实时位置
  - 每日商场新增顾客人数
  - 每日商场中顾客数量前5的商店
  - 每日商场中顾客最多的时间段及顾客数量
- 场景二：顾客喜好分析
  - 单日顾客最喜欢的商店
  - 单日顾客进入超过一次的商店

## 业务代码

### 场景一：客流量统计

#### 每日商场客流量

对顾客WIFI信息按天维度进行分组，使用count distinct语法去重统计得到商场中顾客人数。

#### 输入表

```
--顾客WIFI信息
CREATE TABLE user_wifidata (
  id                varchar,
  shop_code         varchar, --商店编码
  ap_ip             varchar, --WIFI的ip地址
  occur_time        varchar, --记录时间
  user_device_mac   varchar, --终端mac地址
  rssi              varchar, --接收信号强度
  ap_mac            varchar, --WIFI的MAC地址
  pt                varchar, --pt
  ts_occur_time AS TO_TIMESTAMP(cast(cast(occur_time as bigint)*1000 as bigint)
  WATERMARK FOR ts_occur_time AS withOffset(ts_occur_time, 60000)
) WITH (
  type = 'datahub',...
);
```

## 输出表

```
--每日商场客流量
CREATE TABLE trading_area_daily_people (
    count_date          varchar, --日期
    people_sum          bigint,   --客流量
    primary key (count_date)
) WITH (
    type='rds',...
);
```

## 业务代码

```
--每日商场客流量统计
INSERT into trading_area_daily_people
select
    DATE_FORMAT(data.ts_occur_time,'yyyy-MM-dd') count_date,
    count(distinct data.user_device_mac)
from
    user_wifidata data
group by DATE_FORMAT(data.ts_occur_time,'yyyy-MM-dd')
;
```

## 每日商店的客流量

顾客WIFI信息通过关联商场中的商店维表获得商店名称，在此基础上按照商店名称和按天维度时间字段进行分组，使用count distinct语法去重统计每日商店的顾客人数。

## 输入表

--顾客WIFI信息：使用场景一“每日商场客流量”的输入表：user\_wifidata

## 输出表

```
--每日商店的客流量
CREATE TABLE shop_area_daily_people (
    count_date          varchar, --日期
    shop_name           varchar,  --商店名
    people_sum          bigint,   --客流量
    primary key (count_date)
) WITH (
    type='rds',...
);
```

## 维表

```
--商店维表
CREATE TABLE area_shop (
    id                                bigint,
    shop_code                        varchar, --商店编码
    shop_name                        varchar, --商店名
    PRIMARY KEY (shop_code),
    PERIOD FOR SYSTEM_TIME
) WITH (
    type='rds',...
);
```

## 业务代码

```
--每日商店的客流量统计
INSERT into shop_area_daily_people
select
    DATE_FORMAT(data.ts_occur_time,'yyyy-MM-dd') count_date,
    mas.shop_name,
    count(distinct data.user_device_mac)
from
user_wifidata data
left outer join area_shop FOR SYSTEM_TIME AS OF PROCTIME() mas
on data.shop_code=mas.shop_code
group by DATE_FORMAT(data.ts_occur_time,'yyyy-MM-dd'),mas.shop_name
;
```

## 顾客的实时位置

顾客WIFI信息通过关联探测顾客位置维表（place\_info），获取到顾客的实时位置信息，通过1秒钟的滚动窗口实现每秒钟上报一次位置信息，实时投放到数据大屏上，从而实现顾客实时密度图功能。

## 输入表

--顾客WIFI信息：使用场景一“每日商场客流量”的输入表：user\_wifidata

## 输出表

```
--顾客来店数据表
CREATE TABLE area_customer_info_allday (
    user_device_mac                varchar, --终端mac地址
    come_time                      varchar, --到来的时间
    ap_mac                         varchar, --WIFI的MAC地址
    occur_time                    varchar, --记录时间
    used_x                        bigint,  --顾客的经度
    used_y                        bigint,  --顾客的维度
    first_come_flag                varchar, --是否是第一次
    rssi                          varchar, --接收信号强度
```

```

        save_time          varchar, --时间
        pt                 varchar, --pt
    PRIMARY KEY (user_device_mac)
) WITH (
    type='ots'
);

```

## 维表

--探测设备可以获取到顾客的位置

```

CREATE TABLE place_info (
    user_device_mac          varchar, --终端mac地址
    used_x                   bigint,  --顾客的经度
    used_y                   bigint,  --顾客的维度
    min_rssi                 bigint,  --最小的接收信号强度
    PRIMARY KEY (user_device_mac),
    PERIOD FOR SYSTEM_TIME
) WITH (
    type='rds',...
);

```

## 业务代码

--通过1秒钟的滚动窗口实现每秒钟上报一次位置信息

```

CREATE view record_sec
(
    user_device_mac          ,
    come_time                ,
    ap_mac                   ,
    occur_time               ,
    user_loca_x              ,
    user_loca_y              ,
    first_come_flag          ,
    rssi                     ,
    save_time                ,
    shop_code                ,
    pt
) as
select data.user_device_mac ,
DATE_FORMAT(TUMBLE_START(data.ts_occur_time, INTERVAL '1' SECOND), 'yyyy-MM-dd HH:MM:ss'),
data.ap_mac,
data.occur_time,
'',
'',
'',
min(data.rssi),
FROM_UNIXTIME(unix_timestamp()),
grd.shop_code,
max(data.pt)
from

```

```

user_wifidata data
GROUP BY TUMBLE(ts_occur_time, INTERVAL '1' SECOND),ap_mac,user_device_mac,shop_c
;

CREATE view view1 as
select
a0.user_device_mac,
a0.come_time,
a0.ap_mac,
a0.occur_time,
mai.used_x ,
mai.used_y ,
a0.first_come_flag,
a0.rssi,
a0.save_time,
a0.pt
from
record_sec a0
left OUTER join place_info FOR SYSTEM_TIME AS OF PROCTIME() mai
on mai.user_device_mac=a0.user_device_mac;

INSERT into area_customer_info_allday
select *
from view1 ;

```

## 每日商场新增顾客人数

顾客WIFI信息通过关联探测顾客位置维表（place\_info），获取到顾客的实时位置信息，然后通过关联顾客来店记录维表，获取到顾客是否是新顾客信息，统计新顾客的人数。同时将新顾客信息更新到顾客来店记录表中标注成常客。此场景下，顾客来店记录表在同一个作业中即作为结果表又作为维表。

### 输入表

--顾客WIFI信息：使用场景一“每日商场客流量”的输入表：user\_wifidata

### 输出表

```

--每日商场新增顾客人数
CREATE TABLE customer_info_people_sum (
    people_sum          bigint,          --新增顾客人数
    come_time           varchar,         --日期
    PRIMARY KEY (come_time)
) WITH (
    type='rds',...
);

--顾客来店记录表
CREATE TABLE shop_user_record_update (
    user_device_mac     varchar,          --终端mac地址

```

```

first_come_flag          varchar ,          --是否是第一次来
first_coming_time        varchar,          --进店时间
last_coming_time         varchar ,         --离开时间
    PRIMARY KEY (user_device_mac),
    PERIOD FOR SYSTEM_TIME
) WITH (
    type='ots',
    tableName='shop_user_record'...
);

```

## 维表

--使用场景—“商店中顾客的实时位置”的维表: place\_info

```

CREATE TABLE place_info (
    user_device_mac          varchar, --终端mac地址
    used_x                   bigint,   --顾客的经度
    used_y                   bigint,   --顾客的维度
    min_rssi                 bigint,   --最小的接收信号强度
    PRIMARY KEY (user_device_mac),
    PERIOD FOR SYSTEM_TIME
) WITH (
    type='rds',...
);

```

--顾客来店记录

```

CREATE TABLE shop_user_record (
    user_device_mac          varchar, --终端mac地址
    first_come_flag          varchar, --是否是第一次来
    first_coming_time        varchar, --进店时间
    last_coming_time         varchar, --离开时间
    PRIMARY KEY (user_device_mac),
    PERIOD FOR SYSTEM_TIME
) WITH (
    type='ots',
    tableName='shop_user_record'...
);

```

## 业务代码

--通过1秒钟的滚动窗口实现每秒钟上报一次位置信息

```

CREATE view record_sec
(
    user_device_mac,
    come_time      ,
    ap_mac         ,
    occur_time     ,
    user_loca_x    ,
    user_loca_y    ,
    first_come_flag,
    rssi           ,

```

```

        save_time          ,
        shop_code          ,
        pt
    ) as
select data.user_device_mac ,
DATE_FORMAT(TUMBLE_START(data.ts_occur_time, INTERVAL '1' SECOND), 'yyyy-MM-dd HH:MM:ss') as ts,
data.ap_mac,
data.occur_time,
'',
'',
'',
min(data.rssi),
FROM_UNIXTIME(unix_timestamp()),
grd.shop_code,
max(data.pt)
from
user_wifidata data
GROUP BY TUMBLE(ts_occur_time, INTERVAL '1' SECOND), ap_mac, user_device_mac, shop_code;

```

```

CREATE view view1 as
select
a0.user_device_mac,
a0.come_time,
a0.ap_mac,
a0.occur_time,
mai.used_x ,
mai.used_y ,
a0.first_come_flag,
a0.rssi,
a0.save_time,
a0.pt
from
record_sec a0
left OUTER join place_infomst_buynoplace_infow_ap_info FOR SYSTEM_TIME AS OF PROCTIME() place_infomst_buynoplace_infow_ap_info
on mai.user_device_mac=a0.user_device_mac;

```

```

CREATE view view2 as
select
a1.user_device_mac,
a1.come_time,
a1.ap_mac,
a1.occur_time,
cast(a1.used_x as varchar) as used_x,
cast(a1.used_y as varchar) as used_y,
mci.first_come_flag as first_come_flag,
a1.rssi,
a1.save_time,
a1.pt,
mci.first_coming_time,
mci.last_coming_time
from
view1 a1
left OUTER join shop_user_record FOR SYSTEM_TIME AS OF PROCTIME() mci

```



```

on mci.user_device_mac = a1.user_device_mac ;

CREATE view record_out as
select
a2.user_device_mac as user_device_mac,
a2.come_time as come_time,
a2.occur_time as occur_time,
a2.ap_mac as ap_mac,
a2.used_x as used_x,
a2.used_y as used_y,
if(a2.first_come_flag is null , '1', '0') as first_come_flag, --first_come_flag,
a2.rssi as rssi,
a2.save_time as save_time,
a2.pt as pt,
a2.first_coming_time as first_coming_time,
a2.last_coming_time as last_coming_time
from
view2 a2 ;

--更新shop_user_record维表的顾客来店记录
INSERT into shop_user_record_update
select user_device_mac , first_come_flag, first_coming_time, last_coming_time
from record_out ;

--统计每日商店新顾客数
insert into customer_info_people_sum
select t1.people_sum, t1.come_time
from (select count(distinct dt1.user_device_mac) as people_sum, DATE_FORMAT(dt1.come_time, 'yyyy-MM-dd') as come_time
      from record_out dt1 where dt1.first_come_flag = '1'
      group by DATE_FORMAT(dt1.come_time, 'yyyy-MM-dd')) t1

```

## 每日商场中顾客数量前5的商店

顾客WIFI信息按照天维度和商店分组后，统计每日商店顾客数量，通过topn语句获取到每日顾客数量前5的商店信息，再通过关联商场店铺表得到商店名。

### 输入表

--顾客WIFI信息：使用场景一“每日商场客流量”的输入表：user\_wifidata

### 输出表

```

--每日商场中顾客数量前5的商店
CREATE TABLE top5_shop (
    shop_code    varchar, --商店编码
    shop_name    varchar, --商店名
    date_time    varchar, --记录顾客进入商店的时间（天维度）
    people_sum   bigint,  --顾客数量
    PRIMARY KEY (shop_code)
)

```

```

) WITH (
  type='rds'...
);

```

## 维表

--商场店铺表：使用场景—“每日商店的客流量”的维表：area\_shop

```

CREATE TABLE area_shop (
  id                                bigint,
  shop_code                        varchar, --商店编码
  shop_name                        varchar, --商店名
  PRIMARY KEY (shop_code),
  PERIOD FOR SYSTEM_TIME
) WITH (
  type='rds',...
);

```

## 业务代码

--按天、商店分组统计商店的顾客数

```

CREATE VIEW Window1 AS
SELECT shop_code,
       DATE_FORMAT(data.ts_occur_time,'yyyy-MM-dd') AS date_time,
       count(distinct user_device_mac) AS people_sum
FROM user_wifidata data
GROUP BY shop_code, DATE_FORMAT(data.ts_occur_time,'yyyy-MM-dd');

```

-- 统计每天top5客流量的商店

```

CREATE VIEW top5_view AS
SELECT shop_code, date_time, people_sum, rownum FROM
(
  SELECT
    shop_code, date_time, people_sum,
    ROW_NUMBER() OVER (PARTITION BY shop_code,date_time ORDER BY people_sum DESC)
  FROM
    Window1
)
WHERE rownum <= 5;

INSERT into top5_shop
select
  top5.shop_code ,
  mas.shop_name ,
  top5.date_time ,
  top5.people_sum
from
top5_view top5
inner join area_shop FOR SYSTEM_TIME AS OF PROCTIME() mas
on top5.shop_code=mas.shop_code

```

## 每日商场中顾客最多的时间段及顾客数量

顾客WIFI信息按照1小时滚动窗口统计商场顾客数量，使用topn语句获取到每日商场中顾客最多的时间段。

### 输入表

--顾客WIFI信息：使用场景一“每日商场客流量”的输入表：user\_wifidata

### 输出表

```
--商场中顾客最多的时间段及顾客数量
CREATE TABLE top1_time (
    start_time timestamp, --时间段
    people_sum bigint,    --顾客数量
    PRIMARY KEY (start_time)
) WITH (
    type='rds',...
);
```

### 业务代码

```
--1小时滚动窗口统计每小时商场的顾客数
CREATE VIEW Window1 AS
SELECT shop_code,
    TUMBLE_START(ts_occur_time, INTERVAL '1' hour) AS start_time,
    count(distinct user_device_mac) AS people_sum
FROM user_wifidata
GROUP BY shop_code, TUMBLE(ts_occur_time, INTERVAL '1' hour);

INSERT into top1_time
SELECT start_time, people_sum FROM
(
    SELECT
        start_time, people_sum,
        ROW_NUMBER() OVER (PARTITION BY start_time ORDER BY people_sum DESC) as rownum
    FROM
        Window1
)
WHERE rownum <= 1;
```

## 场景二：顾客喜好分析

### 每日单个顾客最喜欢的商店

顾客WIFI信息按照商店、顾客、天维度进行分组得到每日顾客进入商店的次数，使用topn语句获取到每日顾客去过最多的商店，通过关联商店名维表获取商店名，从而统计每日单个顾客最喜欢的商店。

## 输入表

--顾客WIFI信息：使用场景一“每日商场客流量”的输入表：user\_wifidata

## 输出表

--单个顾客最喜欢的商店输出表

```
CREATE TABLE favorite_shop (  
    user_device_mac varchar,          --终端mac地址  
    date_time         varchar,        --记录顾客进入商店的时间（天维度）  
    shop_code         varchar,        --商店编码  
    shop_name         varchar,        --商店名  
    shop_number       bigint ,        --进店次数  
PRIMARY KEY (shop_code)  
) WITH (  
    type='rds'...  
);
```

## 维表

--商场店铺表：使用场景一“每日商店的客流量”的维表：area\_shop

```
CREATE TABLE area_shop (  
    id                bigint,  
    shop_code         varchar, --商店编码  
    shop_name         varchar, --商店名  
    PRIMARY KEY (shop_code),  
    PERIOD FOR SYSTEM_TIME  
) WITH (  
    type='rds',...  
);
```

## 业务代码

--按天、顾客和商店分组统计单个客户当天进入单个商店的次数

```
CREATE VIEW Window1 AS  
SELECT user_device_mac,  
       shop_code,  
       DATE_FORMAT(data.ts_occur_time,'yyyy-MM-dd') AS date_time,  
       count(shop_code) AS shop_number  
FROM user_wifidata data  
GROUP BY user_device_mac,shop_code, DATE_FORMAT(data.ts_occur_time,'yyyy-MM-dd');
```

-- 统计一天中顾客进入次数最多的商店，并输出

```
CREATE VIEW top1_view AS  
SELECT user_device_mac,shop_code, date_time, shop_number, rownum FROM  
(  
    SELECT  
        user_device_mac,shop_code, date_time, shop_number,
```

```

ROW_NUMBER() OVER (PARTITION BY user_device_mac,shop_code,date_time ORDER BY s
FROM
Window1
)
WHERE rownum <= 1;

--关联商店名维表并输出
INSERT into favorite_shop
select
    top1.user_device_mac,
    top1.date_time ,
    top1.shop_code ,
    mas.shop_name ,
    top1.shop_number
from
top1_view top1
inner join area_shop FOR SYSTEM_TIME AS OF PROCTIME() mas
on top1.shop_code=mas.shop_code

```

## 单日单个顾客进入超过一次的商店

顾客WIFI信息按照商店、顾客、天维度进行分组获取到每日顾客进入商店的次数，通过where语句获取到每日顾客进入超过一次的商店，再通过关联商店名维表得到商店名，从而得到单日单个顾客进入超过一次的商店信息。

### 输入表

--顾客WIFI信息：使用场景一“每日商场客流量”的输入表：user\_wifidata

### 输出表

```

--单日单个顾客进入超过一次的商店
CREATE TABLE morethanonce_shop (
    user_device_mac varchar,           --终端mac地址
    date_time        varchar,          --记录顾客进入商店的时间（天维度）
    shop_code        varchar,          --商店编码
    shop_name        varchar,          --商店名
    shop_number      bigint,           --进店次数
    PRIMARY KEY (user_device_mac)
) WITH (
    type='rds',...
);

```

### 维表

--商场店铺表：使用场景一“每日商店的客流量”的维表：area\_shop

```

CREATE TABLE area_shop (

```

```

        id                                bigint,
        shop_code                        varchar, --商店编码
        shop_name                        varchar, --商店名
        PRIMARY KEY (shop_code),
        PERIOD FOR SYSTEM_TIME
    ) WITH (
        type='rds',...
    );

```

## 业务代码

```

--按天、顾客和商店分组统计单个客户当天进入单个商店的次数
CREATE VIEW Window1 AS
SELECT user_device_mac,
       shop_code,
       DATE_FORMAT(data.ts_occur_time,'yyyy-MM-dd')AS date_time,
       count(shop_code) AS shop_number
FROM user_wifidata data
GROUP BY user_device_mac,shop_code, DATE_FORMAT(data.ts_occur_time,'yyyy-MM-dd');

-- 1天中顾客进入超过两次的商店
CREATE VIEW exceed2_view AS
SELECT user_device_mac,shop_code, date_time, shop_number FROM
Window1
WHERE shop_number >= 2;

INSERT into morethanonce_shop
select
    exceed2.user_device_mac,
    exceed2.date_time ,
    exceed2.shop_code ,
    mas.shop_name      ,
    exceed2.shop_number
from
exceed2_view exceed2
inner join area_shop FOR SYSTEM_TIME AS OF PROCTIME() mas
on exceed2.shop_code=mas.shop_code

```