**客户主题**

**店铺客户**

1.1 项目背景

在电商业务快速发展的背景下，企业对数据的实时性需求日益迫切，传统离线数仓已无法满足实时监控、动态决策等场景需求。本项目基于大数据技术栈，构建电商实时数仓，实现对用户行为、交易流程、流量变化等数据的实时采集、处理、分析，为业务提供实时决策支持，如大促实时监控、用户实时画像、动态推荐等。

1.2 项目目标

构建企业级电商实时数仓架构，实现数据从采集到分析的全链路实时处理，端到端延迟控制在分钟级以内。

规范数据分层与建模，确保数据的一致性、复用性和易用性，降低开发成本。

支持多维度业务分析需求，如流量分析、交易分析、用户分析等，提供丰富的实时统计指标。

搭建稳定、可扩展的技术架构，适配业务数据量增长与需求迭代。

二、技术架构

2.1 技术栈选型

技术组件 版本 用途

Flink 1.17.1 实时计算引擎，负责数据清洗、分流、聚合等核心处理

Kafka 3.0.0 消息队列，存储 ODS 层原始数据与 DWD 层明细数据

HBase 2.4.11 存储 DIM 层维度数据，支持高效的 K-V 查询

MySQL 8.0 存储业务数据、配置表数据（如 table\_process\_dim）

Maxwell 1.29.2 监控 MySQL Binlog，同步业务数据至 Kafka

Flume 1.11.0 采集用户行为日志，传输至 Kafka

Doris 1.5.2 OLAP 数据库，存储 DWS 层汇总指标，支持高效查询

Redis 6.0.8 旁路缓存，优化维度查询性能

Zookeeper 3.5.7 服务协调，管理 Kafka、HBase 等组件元数据

2.2 数据源：包括电商业务数据库（MySQL）、用户行为日志（APP/WEB 埋点）。

数据采集层：通过 Maxwell 同步 MySQL Binlog 数据至 Kafka，通过 Flume 采集日志数据至 Kafka。

项目首先明确店铺客户的三层核心分类，为后续数据处理和指标计算划定业务边界：

新访客户：统计周期内首次或 “超出行为有效期” 后，与店铺产生访问 / 互动 / 支付行为的客户（含新访成交、新访未成交）；

未购客户回访：历史与店铺有接触但未购买，统计周期内再次产生访问 / 互动 / 支付行为的客户（“看了又看”）；

已购客户回访：过去 365 天内有购买记录，统计周期内再次产生访问 / 互动 / 支付行为的客户（“买了又看”）。

二、项目目标与业务需求拆解

1. 核心业务目标

为电商商家提供店铺客户分层运营决策依据，通过精准统计 “新访成交、回访成交、老客复购” 三类核心场景数据，实现 “判断客户视角下店铺运营效率、拉动生意增长” 的最终目标（文档 2 任务描述）。

2. 客户分层定义（业务基础）

项目首先明确店铺客户的三层核心分类，为后续数据处理和指标计算划定业务边界：

新访客户：统计周期内首次或 “超出行为有效期” 后，与店铺产生访问 / 互动 / 支付行为的客户（含新访成交、新访未成交）；

未购客户回访：历史与店铺有接触但未购买，统计周期内再次产生访问 / 互动 / 支付行为的客户（“看了又看”）；

已购客户回访：过去 365 天内有购买记录，统计周期内再次产生访问 / 互动 / 支付行为的客户（“买了又看”）。

3. 核心业务指标（实时监控维度）

围绕 “增长通路” 设计关键指标，覆盖 “召回效率 - 转化效率 - 整体效率” 全链路，具体如下：

增长通路 核心指标 计算公式 业务意义

通路 1：客户新访 新访支付转化率 新访支付买家数 / 新访访客数 衡量新访客当日直接转化能力

新访支付金额占比 新访销售额 / 店铺总销售额 新访客户对店铺营收的贡献度

新访客单价 新访支付金额 / 新访支付买家数 新访成交客户的消费能力

通路 2：未购客户回访 未购客户召回率 未购客户回访 / 未购客户总数 未购潜在客户的召回效果

回访支付转化率 回访成交数 / 未购客户回访数 召回后未购客户的转化能力

回访成交率 未购客户召回率 × 回访支付转化率 通路 2 的整体经营效率

通路 3：已购客户回访 老客召回率 已购客户回访 / 已购客户总数 老客的二次触达效果

老客复购转化率 老客复购数 / 已购客户回访数 召回老客的复购意愿

老客复购率 老客召回率 × 老客复购转化率 通路 3 的整体经营效率

三、核心任务拆解（技术 + 业务联动）

项目采用 “业务指标驱动技术落地” 的模式，将整体任务拆解为业务指标计算和技术支撑落地两大模块，

1. 技术支撑任务（实时项目核心落地环节）

技术任务围绕 “实时数据链路构建 - 实时建模 - 湖仓存储 - 可视化展示” 全流程设计，具体任务与交付物对应如下：

技术任务模块 具体任务内容 交付物 核心能力体现

数据接入层 客户日志数据分流处理 日志处理 Java 代码 日志分流、合流处理能力（实时接收并清洗客户行为日志）

客户业务数据分流处理 业务数据处理 Java 代码 业务数据分流、同步能力（实时同步订单、会员等业务数据）

实时数仓层 实时数仓分层建模（ODS→ADS） ods 到 ads 分层的代码实现 数仓分层建模能力（按实时数仓规范落地数据分层，保障数据复用性）

复杂指标计算（跳出明细等） 指标加工代码实现 使用 COP（复杂指标计算组件）处理多维度指标的实时计算能力

存储层 Paimon 湖仓一体应用 paimon 部署文档、paimon 数仓分层落盘及 query 脚本 湖仓一体设计及部署应用能力（兼顾实时数据写入与高效查询）

可视化层 实时指标可视化展示 BI 组件部署文档、BI 看板可视化效果 复杂指标可视化能力（将实时计算的业务指标转化为商家可直观查看的看板）

基础支撑 技术选型及环境部署 需求分析文档、技术选型文档、环境部署脚本 技术选型合理性验证、生产环境快速部署能力

2. 业务指标计算任务

技术任务最终服务于业务指标落地，所有代码实现需覆盖以下业务计算逻辑：

基础数据统计：店铺客户数（访客 + 互动客户 + 支付买家去重）、各分层客户数（新访 / 未购回访 / 已购回访）；

效率指标计算：三类通路的召回率、转化率、整体成交率；

价值指标计算：各分层客户的支付金额占比、客单价、粉丝数 / 会员数（会员定义：交易额≥5000 或交易频次≥3 次）。

四、技术实现方案（实时项目技术栈与架构）

1. 核心技术栈选型（基于文档提取）

项目聚焦 “实时性” 与 “可落地性”，技术栈适配实时用户画像场景需求：

技术层 技术选型 选型依据（文档支撑）

数据处理引擎 Java（自定义代码） 文档 1 明确交付 “日志处理 Java 代码”“业务数据处理 Java 代码”，适配实时数据分流逻辑

复杂指标计算 COP 组件 文档 1 提及 “使用 cop 等处理复杂指标的能力”，用于多维度、高并发的实时指标计算

湖仓存储 Paimon 文档 1 交付 “paimon 部署文档”“paimon 数仓分层落盘脚本”，适配实时数仓的 “实时写入 + 离线查询” 混合需求

可视化工具 BI 组件（未明确具体工具） 文档 1 交付 “BI 组件部署文档”“BI 看板可视化效果”，支持业务指标实时展示

数仓架构 分层架构（ODS→ADS） 文档 1 提及 “ods 到 ads 分层的代码实现”，符合实时数仓 “轻量分层、快速赋能” 的设计原则

2. 实时数据链路设计

结合技术任务与业务需求，项目构建的实时数据链路如下：

数据接入：通过 Java 代码分别接收 “客户日志数据”（行为日志）和 “客户业务数据”（订单、会员数据），实现分流处理（避免数据混杂影响实时性）；

数据建模：基于 ODS 层（原始数据层）进行实时清洗、转换，落地至 ADS 层（应用数据层），直接支撑业务指标计算（如 ADS 层存储 “新访客户支付事实表”“老客复购事实表”）；

指标计算：通过 COP 组件对 ADS 层数据进行实时聚合，计算召回率、转化率等复杂指标；

数据存储：基于 Paimon 实现湖仓一体存储，既保障实时数据的快速落盘，也支持后续离线分析（如历史指标回溯）；

可视化展示：将实时计算的指标通过 BI 组件部署为可视化看板，商家可直观查看各分层客户的运营数据。

五、产出物与验收标准（实时项目交付管控）

项目通过明确 “交付物清单” 和 “验收标准”，保障实时功能落地质量，具体要求如下：

1. 核心产出物清单

产出物类型 具体内容 文档支撑

设计文档 店铺客户设计文档（含 ADS 层表设计、原始数据分析、关键指标实现方案、性能优化方案） 文档 2（验收标准 1）

代码资产 店铺客户代码（覆盖全部指标功能，含 “分层实现”“不分层实现” 两种方式，代码注释需包含工单编号） 文档 2（验收标准 2）、文档 1（各类 Java 代码）

测试文档 店铺客户测试文档（含测试记录、测试 SQL，且分层 / 不分层实现的数据需一致） 文档 2（验收标准 3）

上线证据 店铺客户上线截图（需包含截图时间、清晰的指标数值） 文档 2（验收标准 4）、文档 1（BI 看板可视化效果）

2. 关键验收要点（实时项目核心管控点）

功能完整性：代码需覆盖所有业务指标（如 3 类增长通路的 10 + 核心指标），且两种实现方式（分层 / 不分层）均需可用；

数据一致性：分层与不分层实现的指标数据需完全一致（避免建模逻辑漏洞）；

实时性验证：上线截图需体现 “实时性”（如截图时间与指标更新时间匹配），确保数据不是离线静态数据；

可维护性：代码注释需包含工单编号（大数据 - 用户画像 - 01 - 客户主题店铺客户），设计文档需明确性能优化方案（保障高并发场景下的实时处理能力）。

六、项目价值与亮点（实时项目视角）

1. 业务价值

精准运营支撑：首次将店铺客户拆分为 “新访 - 未购回访 - 已购回访” 三层，帮助商家定位运营薄弱环节（如未购召回率低则需优化召回策略，老客复购率低则需提升老客粘性）；

实时决策赋能：通过实时数据链路，商家可实时查看各分层客户的转化效率，快速调整运营动作（如发现新访转化率骤降，可立即优化新客引流话术）。

2. 技术价值

湖仓一体落地：采用 Paimon 实现 “实时写入 + 离线查询” 一体化，避免实时库与离线库的数据不一致问题，降低数据维护成本；

轻量化实时数仓：采用 “ODS→ADS” 简化分层（未提及 DWD/DWS 层），适配中小电商商家的实时需求，兼顾 “实时性” 与 “开发效率”（3 人日工时可落地）。