		目	录	
平台介绍				
帮助文档指引				
平台简介				
专业术语				
版本信息				
使用流程				
流程概述				
前期准备				
埋点采集方案	设计			
数据采集与上	报			
后续使用				
功能手册				
数据空间				
书签				
数据分析				
事件分析	Ť			
漏斗分材	Ť			
间隔分材	Ť			
留存分材	Ť			
分布分材	Ť			
路径分析	Ť			
属性分析	Ť			
网页热ナ]分析			
SQL查询]			
SQL可视	化			
用户洞察				
用户标图				
用户分群	ŧ			
用户详情	上 月			
用户查询	1]			
数据中心				

数据接入

元数据管理

元事件

事件属性

用户属性

维度表属性

虚拟事件

虚拟属性

监控预警

管理中心

系统管理

协作者管理

角色管理

部门管理

岗位管理

项目管理

基本信息

成员管理

分析配置

分析主体

时区设置

消息通知

消息助手

消息中心

操作视频

功能使用

数据空间

数据分析

用户洞察

数据中心

管理中心

技术文档

基础知识

接入流程 数据格式 用户识别 预置属性与系统字段 采集上报地址获取 客户端SDK 合规说明 合规说明 Android SDK 集成文档 使用说明 APl详解 更新日志 iOS SDK

集成文档

使用说明

API详解

更新日志

Web JS SDK

集成文档

全埋点

使用说明

API详解

常见问题

更新日志

小程序SDK

微信小程序SDK

集成文档

使用说明

API详解

更新日志

跨平台框架

Flutter 插件

集成文档 使用说明 API详解 更新日志 APICIoud 集成文档 使用说明 API详解 更新日志 React Native 集成文档

使用说明

API详解

更新日志

服务端SDK

Java SDK

集成文档

使用说明

API详解

更新日志

Python SDK

集成文档

使用说明

API详解

更新日志

数据校验

数据导入

Datax

dataimporter

Logstash + Filebeat

SQL手册

DML

SELECT

SQL函数

日期函数

convert_tz

curdate,current_date

current_timestamp

curtime,current_time

date_add

date_format

date_sub

datediff

day

dayname

dayofmonth

dayofweek

dayofyear

from_days

from_unixtime

hour

makedate

minute

month

monthname

now

second

str_to_date

time_round

timediff

timestampadd

timestampdiff

to_date

to_days

unix_timestamp

utc_timestamp

week weekday weekofyear year yearweek localtime,localtimestamp quarter last_day extract date_trunc minutes_add minutes_diff minutes_sub seconds_add seconds_diff seconds_sub hours_add hours_diff hours_sub days_add days_diff days_sub weeks_add weeks_diff weeks_sub months_add months_diff months_sub years_add years_diff years_sub 字符串函数

append_trailing_char_if_absent

数据类型

BIGINT

INT

TINYINT

SMALLINT

LARGEINT

BOOLEAN

DATE

DATETIME

DECIMAL

DOUBLE

FLOAT

CHAR

VARCHAR

STRING

ARRAY

常见问题

产品问题 数据&技术问题

费用&服务问题

资源下载

平台介绍

平台介绍

帮助文档指引

帮助中心 > 平台介绍 > 帮助文档指引

▼平台介绍

帮助文档指引	平台简介	专业术语	
▼使用流程			
流程概述 集与上报	前期准备	埋点采集方案设计	数据采
▼功能手册			
数据空间 数据中心	数据分析 管理中心	用户洞察 其他	
▼技术文档			
基础知识 校验	客户端SDK 数据导入	服务端SDK SQL手册	数据
▼常见问题			
产品问题	数据8技术问题	费用8服务问题	
▼ 资源下载			
资源下载			

平台简介

帮助中心 > 平台介绍 > 平台简介

1海纳嗨数简介

海纳嗨数是一款深度用户行为分析产品,支持私有化、SaaS两种部署方式。海纳嗨数拥有规范的数据管理体系,支持多类型数 据接入和全平台的数据采集,助力企业打造数字智能化的运营管理体系。平台集成数据接入、数据分析、用户洞察、人群定 位、营销投放为一体,让用户行为数据发挥出最大的价值。

此外,海纳数科沉淀数据业务多年,具有丰富的行业经验储备,在消金、游戏、教育等多个领域具有成熟的行业解决方案,可 助力企业高效运营。在需求对接、采集方案设计、数据上报、指标设计、问题追踪等方面提供全流程定制化服务,深入挖掘业 务痛点,精准定位问题所在。无论是售前咨询、售中使用还是售后维护,都有专业的工作人员为您提供全流程指导。

2 产品能力

• 多类型数据接入

通过本地数据库、数据集、API接口、离线传输、客户端等多种方式进行内部数据打通及外部数据补充接入,扩充数据资产 池。

- 全平台数据采集 前端、服务端等全端数据采集支撑及处理,ID-Mapping精准定位行为人及关键行为信息,提高数据存储的精确度和利用率。
- 标签创建与衍生
 支持自定义、数值、初末次等多种标签创建方式及分群方式,同时提供标签二次衍生功能,最大化满足业务需求。
- 多维度数据分析
 覆盖主流事件分析、留存分析、漏斗分析等十大分析模型,辅助企业了解现状、发现问题、优化决策。
- 实时响应高效决策
 毫秒级数据更新机制,实时监测用户关键行为,实现高效化的精准化营销与投放。
- 深度下钻用户旅程
 深入了解用户旅程,探索用户偏好,刻画用户画像与建设用户分层,构建高价值的人群资产池。

3. 服务架构

海纳嗨数采用云引擎技术,集成数据空间、数据分析、用户洞察、数据中心、管理中心、书签、个人中心等功能模块,全面支撑您的数据分析需求,挖掘业务痛点并驱动业务决策。

专业术语

帮助中心 > 平台介绍 > 专业术语

名词	释义
埋点	针对特定用户行为或事件进行捕获、 处理和发送的相关技术及其实施过程。
元事件	元事件是一系列属性的集合、不具备数据类型。 代表了用户的某个或一系列有意义的行为, 一条事件主要包含两部分信息,一部分用以描述该行为如何发生, 另一部分是该行为的属性。
事件属性	在这个事件中需要采集的具体信息,是每种事件所独有的属性。 事件属性除了与该事件相关的属性外, 还能够反映用户在进行该行为时的状态。
公共事件属性	公共事件属性是通过配置后进行统一上报的, 会作用于每一个事件的属性,无需在每个事件的埋点中进行。
用户属性	用户属性表示的是用户的不变的属性以及最新状态,通过用户属性, 可以在用户行为分析时快速筛选出要分析的用户
预置属性	预置属性,是提前在 SDK 中预置好的, 这些属性的中文名以及意义都已经确定, 所有的预置属性均以"H_"开头。
维度表属性	维度表属性是基于已有属性上传维度表而新生成的属性, 可将原先上传的数据映射为另一种展示值或计算值。
分析主体	分析主体是唯一用户的标识,通常由可以识别身份的字段担任, 如设备ID、账号ID、角色ID等。若以设备ID作为分析主体, 则将不同设备ID的用户视作不同用户;若以账号ID作为分析主体, 则将不同账号ID的用户视作不同用户。

版本信息

帮助文档 > 版本信息

V1.0

- 新功能:

- 1. 新增"系统管理",用于管理系统成员信息、分配系统成员权限。
- 2. 新增"项目管理",用于管理项目成员信息、分配项目成员权限,对分析主体和时区进行管理设置。
- 3. 新增"数据接入",提供多种数据接入方式,包括服务端埋点、客户端埋点、工具导入。
- 4. 新增"元数据管理",用于用户上传埋点采集方案,并对上传的事件及属性进行管理和编辑。
- 5. 新增"用户标签",支持用户基于埋点数据自定义标签规则生成新的标签,描述用户的行为特征。
- 6. 新增"事件分析",基于事件的指标维度、属性筛选、分组查看等功能,对事件的整体趋势情况进行分析。
- 7. 新增"漏斗分析",用于分析用户在指定步骤下每一步的转化与流失情况。
- 8. 新增"属性分析",对用户各类特征进行标示,从而了解用户的属性、或者属性交叉的分布结构。
- 9. 新增"数据空间",可以将多个分析或标签同时展示,方便数据查看,同时将多组相关场景的数据维度进行比较分析,并共享给其他用户。

V1.1

- 新功能:

- 1. 新增"维度表属性",用户可基于已有属性上传维度表而新生成的属性,可将原先上传的数据映射为另一种展示值或计算值。
- 新增"监控预警",用于对重点数据或指标进行监控,可对同一指标进行分组,以不同的监控触发标准对各分组值进行监控, 更加快捷智能的洞察异常数据。
- 3. 新增"SQL查询",当已有的分析模型不满足分析需求时,可以通过SQL查询功能进行自定义查询。
- 4. 新增"用户分群",支持用户将满足特定条件的人群圈选为一个分群,深入挖掘用户行为。
- 5. 新增"消息通知",帮助用户及时接收与管理系统通知、预警通知、空间动态和资产动态等消息。
- 6. 新增"书签",帮助用户将常用分析查询保存。

- 优化点:

- 1. "元事件": 取消事件显示名限制,新增是否上报列,新增时删除事件类型,更改关联属性方式。
- 2. "事件属性": 取消属性显示名限制,属性类别新增公共属性,新增时删除属性类别,修改新增逻辑。
- 3. "用户属性": 取消属性显示名限制,属性类别展示完整,新增时删除属性类别。
- 4. "事件分析":新增事件列表下钻页面。
- 5. "漏斗分析"、"属性分析"及"用户洞察":新增用户列表下钻页面。

V1.2

- 新功能:

- 1. 新增"用户详情",用户的全景画像,展示了该用户的基础属性信息、所属标签和操作行为路径等信息,方便快速获取用户特征。
- 2. 新增"用户查询",位于导航栏的快速查询功能,方便用户在任意页面匹配满足条件的用户。
- 3. 新增"SQL可视化",可视化模块支持将查询的表数据转换为图表的形式进行展现。
- 4. 新增"间隔分析",间隔分析用于分析用户行为序列中的两个指定事件之间的转化时长。
- 5. 新增"留存分析",留存分析用于分析进行某项初始行为的用户中,会进行后续行为的人数和比例。
- 6. 新增"分布分析",分布分析主要用于分析某个事件指标的用户分布情况。
- 7. 新增"路径分析",路径分析主要用于分析用户对某些事件的行为顺序、行为偏好、关键节点以及转化效率。

使用流程

流程概述

帮助中心 > 使用流程 > 流程概述

1 流程步骤

2 流程解析

2.1 前期准备

在这一阶段,您需要提前了解以下信息。

- 1. 初次使用海纳嗨数,您需要开通我们的产品账号并创建一个新的项目,我们将会给您配置一个项目和一个账户名为admin的 超级管理员账号,您可使用该账号设定项目管理员及新增成员。
- 2. 为了更好的使用海纳嗨数,您需要对海纳嗨数分析云的用户行为分析模型(Event & User 模型)有一个基本的了解,方便 您了解数据分析的基本原理。
- 3. 了解海纳嗨数的数据结构,明确分析的目标,并梳理分析需求和埋点方案,确定产品中的账号体系、用户标识等。

2.2 数据采集方案设计

在这一阶段,您需要根据整理的需求设计埋点采集方案。

- 1. 根据梳理过的分析需求,确定需要上传的事件,并确定埋点触发条件。
- 2. 根据事件的分析维度,为事件添加属性,事件属性用于分析时的筛选条件、分析维度以及分析指标。
- 3. 根据需求设置用户属性,用户属性主要用以区分不同类型的用户。

2.3 数据采集与上报

在这一阶段,您需要确定数据上报的方式,并完成数据的上报。

- 1. 确定数据的上报方式,海纳云支持使用客户端 SDK、服务端 SDK 以及数据导入工具进行数据上报。
- 2. 根据数据采集方案以及数据上报方式进行数据上报。

2.4 后续使用

在完成上述几步后,您可以使用事件分析、漏斗分析、属性分析等分析模型查看想要分析的数据;也可以使用用户标签、用户 分群等模型查看用户行为特征,为用户打标签,圈选目标人群;同时,您可以将日常查看的指标添加为书签或保存到空间,方 便日常的运营监控。

3 流程详解

若想查看更多细节,可查看详细流程介绍

- 1. 前期准备:了解在使用海纳嗨数之前需要做些什么
- 2. 埋点采集方案设计:理解埋点采集方案并学会如何设计
- 3. 数据采集与上报:数据是如何采集好并上报到海纳嗨数
- 4. 后续使用:海纳嗨数的使用实例

前期准备

帮助中心 > 使用流程 > 前期准备

1了解接入流程

1. 开通产品账号:若想使用海纳嗨数,我们将会给您配置一个项目和一个管理员账号。

2. 创建项目:如果想要申请新的项目,请联系海纳嗨数的工作人员协助申请。我们将会为您创建新的项目,并将您的账号设置 为项目管理员,您可以自行添加项目成员。

2项目理解

2.1 什么是用户行为分析

我们关注用户的新增、留存、留失、活跃、转化等行为,也希望从不同消费水平、不同城市、不同时间段等维度挖掘用户的行为差异,这些都是用户行为分析。海纳嗨数希望通过埋点来获取用户的行为数据,解决常规分析中数据不全或数据分布杂乱的问题。

2.2 如何做用户行为分析

我们使用Event & User 模型来做用户行为分析,其包括事件(Event)和用户(User)两个部分。

Event代表的是用户发生的行为事件,简单来说,一个Event就是描述了一个用户在某个时间点,以某种方式完成了某件具体的事情。因此一个Event就包含了一下几个关键的因素:

- who:参与这个事件的用户是谁。
- when:事件发生的具体时间点。
- what:用户发生的行为,如付费。
- how: 行为的完成方式,即该行为的属性。比如付费时的付费金额、付费方式、付费内容、付费时间等。

User则用来描述每个用户的固定属性和最新状态,如用户的性别、年龄、所在城市、累计付费金额、会员等级、当前积分数 等。通过User的属性,可以在用户行为分析中快速筛选出想要分析的人群,如您想要给重点客户发放福利,就可以通过会员等 级筛选出目标人群。

事件 (Event) 和用户 (User) 通过ID相关联,将用户行为路径记录下来,再通过数据分析模型对数据进行加工就可以达到用 户行为分析的目的。

3 需求梳理

在开始做分析之前,您需要梳理分析的需求和痛点,您可以遵循下列步骤来对需求进行整理,有任何疑问请联系海纳嗨数相关人员为您进行解答。

3.1 明确分析目标

建议您先从基础的分析指标开始梳理,如新老用户数、用户活跃等,并明确这些指标计算的关键事件,如用户进行APP启动为用 户活跃,进行APP注册算做新用户。当您梳理好基础指标后,再根据自身业务需求扩展一些特殊指标。

3.2 梳理埋点事件

在梳理好分析指标后,可以将指标的关键事件进行整理,找到需要埋点的事件行为。

我们建议您将重要的行为单独设置为一个Event,如用户登录、用户注册等;对于不重要的行为,可以将多个行为设置为一个 Event,如页面浏览,具体浏览的是哪个页面可以设置属性浏览页面来进行记录,而不必将主页浏览、详情页浏览、帮助页浏览 等分别设置多个事件;若您特别关注某个环节中的某一行为,也可将该行为单独作为一个Event进行追踪,如用户日常任务中的 签到,可以将签到作为一个单独的事件。

为了方便后续的沟通,我们建议您通过文档的形式对所有需要采集的Event进行整理,也可以联系海纳嗨数相关人员协助您一起整理。

3.3 确定用户标识ID

海纳嗨数主要使用三个字段来识别用户,分别是设备 ID(distinct_id)、账号 ID(account_id)、海纳ID(user_id)。您可以结 合自身的应用场景选择对应用户身份标识ID。

若想查看海纳嗨数用户识别更多细节,可查看海纳嗨数用户识别

埋点采集方案设计

帮助中心 > 使用流程 > 埋点采集方案设计

1. 概述

在完成前期准备后,已经对事件与用户有了一定的了解,我们需要结合业务分析需求明确我们需要采集的事件、事件属性、用 户属性,弄清应该在哪些地方进行埋点,形成埋点采集方案文档。基于设计的埋点采集方案,进行埋点实施,完成数据的采集 与上报。

2. 埋点采集方案

埋点采集方案通常包括三部分:

- 1. 事件表:记录需要采集的某个事件及事件关联的事件属性
- 2. 用户表:记录需要采集的用户属性
- 3. 预置属性:此部分无需设计,是提前在 SDK 中预置好的,这些属性的中文名以及意义都已经确定,所有的预置属性均以" H_"开头

注:为了帮助理解,我们提供了海纳云数据采集模版文档供您参考:数据采集文档.xlsx

3. 设计思路

设计埋点采集方案时,事件、事件属性和用户属性的设置将直接影响到分析结果。事件通常代表需要分析的动作或行为;事件 属性除了是与事件相关的属性,还代表用户在进行该事件行为时的状态;用户属性则只表示用户最新的状态。

3.1 事件表

事件表包括要分析的事件及此事件的事件维度,即事件属性。事件属性是每个事件所独有的,需要根据分析需求及事件的触发 条件进行设置。事件表主要包括以下信息:参与事件的对象、事件发生的时间、事件发生的地点、事件发生的内容等。 事件表需填写的字段及规范如下:

- 事件名:建议只使用大小写字母、数字、下划线,且不以数字开头,请勿使用中文命名,不允许重复。
- 事件显示名:事件在使用过程中的显示名称。
- 属性名:建议只使用大小写字母、数字、下划线,且不以数字开头,请勿使用中文命名,同一事件下的属性名不允许重复,若不同事件下存在意义相同的属性,建议设置为同一属性名。
- 属性显示名:属性在使用过程中的显示名称。
- 数据类型:属性所支持的数据类型可查看数据格式部分
- 埋点平台:请按照枚举值填写,可支持Android/IOS、JS、微信小程序、百度小程序、QQ小程序、字节跳动小程序、支付宝 小程序、快应用、服务端。

3.2 用户表

用户属性主要记录用户不变的属性及最新状态,建议将以下类型的属性设置为用户属性

- 固定属性:指用户一般不会改变的属性,包括用户的个人信息、首次产生产生某行为的信息等。例:用户的姓名、性别、 注册时间、注册地点、首次付费时间等
- 2. 最新状态:指用户当前的状态,通常是用户最后产生行为的信息,比如最后上线日期,最后付费金额等
- 3. 累计结果:累计结果是最新状态的另一种形式,数据类型为数值型,例:累计付费金额、累计交易次数等

用户表需填写的字段及规范如下:

- 属性名:建议只使用大小写字母、数字、下划线,且不以数字开头,请勿使用中文命名,不支持重复。
- 属性显示名:属性在使用过程中的显示名称。
- 数据类型:属性所支持的数据类型可查看数据格式部分

设置完成后,可将属性加入埋点采集方案中

3.3 预置属性

预置属性指的是由海纳嗨数生成或取得的属性,所有的预置属性均以"H_"开头,包括预置用户属性、事件属性,无需用户进行设置,具体字段可在数据采集文档模版查看。

数据采集与上报

帮助中心 > 使用流程 > 数据采集与上报

1. 根据数据采集文档埋点

根据整理的数据采集文档进行埋点,针对不同的事件和属性,触发条件和调用的接口也有所不同。 针对用户属性,埋点时的调用方式有以下几种:

- 1. profile_set : 此方式上传的属性将会覆盖原有的属性值,如果之前不存在该用户属性,则会新建该用户属性,建议用户属性代表最新状态时使用此方式。
- 2. PROFILE_SET_ONCE : 此方式上传的用户属性只要设置一次,当该属性之前已经有值的时候,将会忽略,建议用户属性为固定值时使用此方式。
- 3. PROFILE_INCREMENT : 此方式将对该属性进行累加操作,如果该属性还未被设置,则会赋值 0 后再进行计算,可传入负值,等同于相减操作,建议用户属性代表累计结果时使用此方式。

2. 选择接入方式

在完成埋点采集方案的设计后,需要选择数据的接入方式,包括客户端 SDK、服务端 SDK 以及数据导入工具

- 服务器埋点包括java、python、php、ruby,服务器埋点可准确采集业务操作,确保数据完整上报。
- 客户端埋点包括web、ios、android,客户端埋点可以详细采集用户行为,完整采集用户本地标识。
- 数据导入工具包括Batchimporter、LogAgent、FormatImporter、HdfsImporter、SFTP,数据导入工具通常用于历史数据导入,支持对文件进行批量传输。

具体内容可查看技术文档部分

3. 三方数据接入

我们还支持外部三方数据接入,目前包括巨量引擎,用户可将巨量引擎的广告投放数据接入平台进行使用,具体接入流程可查 看数据接入部分

后续使用

帮助中心 > 使用流程 > 后续使用

若您已经完成了海纳嗨数的数据接入工作,接下来就可以自由的使用海纳嗨数进行数据分析。您可以使用事件分析、漏斗分析、属性分析等分析模型查看想要分析的数据;也可以使用用户标签、用户分群等模型查看用户行为特征,为用户打标签,圈选目标人群;同时,您可以将日常查看的指标添加为书签或保存到空间,方便日常的运营监控。 对于后续的使用若您有任何疑惑,可以联系海纳嗨数工作人员,我们会帮助您快速熟悉平台的使用流程。

1 如何使用分析模型?

不同的分析模型适用于不同的应用场景。若想对某段时间内用户产生的特定行为指标进行聚合计算,可以使用事件分析,如统 计一段时间内的新用户数、活跃用户数;若想了解某个产品在各个关键步骤环节的转化状况,可以使用漏斗分析,如分析用户 从点击链接:注册-登录-付费的操作行为。

想要创建一个分析模型,需要进行以下几个操作步骤:

第一步:选择想要计算和统计的指标,若想筛选满足条件的用户,可以使用「全局筛选」来增加筛选条件;若想按某分组进行 查看,可以使用「分组项」进行设置。如统计不同省份用户的App使用情况(注册时间在2022年9月1日前),配置如图所示:

第二步:可在结果展示区查看结果,支持更改查询的时间范围(默认过去7天)、展示内容、展示图表等。

第三步:若想保留查询配置,可将查询结果保存为书签/空间:

可至空间或书签中查看该图表:

2 如何创建用户标签?

用户标签模块支持用户基于埋点数据自定义标签规则生成新的标签,描述用户的行为特征。用户标签可以单独管理分析,也可以在分析模型中细分分析,同时可添加至空间进行日常的数据监控。

想要创建一个标签,需要进行以下几个操作步骤: 第一步:点击标签管理主页的「新增」创建标签,选择好想要使用的标签创建方式后,点击进入标签配置页面。

各种创建方式的功能说明如下:

- 数值标签:对用户的行为事件进行聚合计算,输出数值型标签。如统计用户近一年内登录的总次数。
- 自定义标签:根据若干条件筛选出满足条件的用户,自定义其标签值。如将近一年充值笔数小于10的人群定义为低价值人 群,大于等于10的定义为高价值人群。
- 初末次标签:将用户初次/末次完成某件事件的日期/距今天数/属性值作为标签值。如统计用户首次登录距今天数。
- SQL标签:使用自定义的SQL查询语句为指定的用户进行标记。
- 导入标签:使用导入的标签值为指定的用户进行标记。

第二步:进行标签规则的配置,标签的配置包括三个部分,分别是标签基础信息、标签规则配置、更新与备份。

第三步:标签基础信息、标签规则配置、更新与备份均设定好以后,点击「保存并计算」按钮即可创建标签,并同步至标签管 理主页。 第四步:若想查看标签的数据详情,可点击标签名称进入标签详情页:

3 如何配置数据空间?

在后续的使用过程中,我们首先建议您创建数据空间。数据空间可将日常运营需要关注的数据以可视化的方式进行批量展示, 并支持实时更新。创建好的空间可在项目成员内进行共享,有利于日常监控产品的运营状况,了解用户的使用习惯。

数据空间的特点有:

- 数据空间可以共享给项目其他成员查看,并支持添加协同者一起管理编辑空间。
- 数据空间支持自定义展示内容和结构排版。
- 初始化配置后,每次加载数据空间时数据会进行更新,方便日常的监控。
- 数据空间支持二次筛选、调整时间窗、数据导出等操作。

想要配置一个空间,需要进行以下几个操作步骤:

第一步:在数据空间中新建一个我的空间/公共空间:

第二步:在分析模型中配置分析指标,并将其计算结果添加到数据空间/保存为书签:

- 若将计算结果保存到数据空间,则可至数据空间直接查看该结果:
- 若将计算结果保存为书签,则可至数据空间主页添加该书签至空间:

第三步:重复步骤二,基于数据分析或用户标签模块的分析模型配置想要在数据空间模块进行统一管理和查看的指标。

第四步:可在数据空间主页对图表的大小和展示方式进行调整,并可拖拽调整图表的顺序,自定义数据空间的布局和内容,完成数据空间的创建。

第五步:设置空间的查看和协作权限,将空间共享给项目内指定成员查看和编辑。

提示1:角色为分析师和项目管理员的成员有权限编辑空间,但只有项目管理员可以设置公共空间。 提示2:在数据空间的配置上,建议预先设置基础指标模版分享给所有项目成员,各业务线再根据使用需求配置与业务相关的指 标模版,共享给相关人员使用。

功能手册

数据空间

帮助中心 > 功能手册 > 数据空间

1 概述

为了更好地满足用户个人数据分析场景的需求和企业的业务数据管理需求,数据空间分为「我的空间」和「公共空间」两种类型的空间,这两种类型的空间在操作上基本一致,区别点在于管理权限的不同。

「我的空间」由用户个人管理,支持将一个分析或者标签放进多个空间,方便日常数据的监控;「公共空间」由具有「管理公 共空间」权限的项目成员共同管理,打造企业公共的数据空间。

2 位置和使用权限

操作位置:数据空间

使用权限: 不同的项目角色操作权限如下

我的空间:

项目角色	权限	权限明细
管理员	查看、编辑	查看所有空间;对空间指标进行页面临时操作;对自建分组/ 子分组/空间进行新建、重命名、删除等操作; 对他人创建的空间进行复制、移动;管理空间; 对自建空间进行时区设置、选择分析时段、添加筛选条件、 共享设置、刷新页面、更多、添加空间内容; 对空间指标进行拖拽、设置、删除、下载
分析师	查看所有空间;对空间指标进行页面临时操作;对自建分组/ 子分组/空间进行新建、重命名、删除等操作; 对他人创建的空间进行复制、移动;管理空间; 对自建空间进行时区设置、选择分析时段、添加筛选条件、 共享设置、刷新页面、更多、添加空间内容; 对空间指标进行拖拽、设置、删除、 下载	
普通用户	查看	查看所有空间;对空间指标进行页面临时操作

公共空间:

项目角色	权限	权限明细
管理员	查看、 编辑、	查看所有空间;对空间指标进行页面临时操作; 对公共空间下的所有分组/子分组/空间进行新建、重命名、

	他人空 间- 编辑、 公共空 间管理 权限	删除等操作;管理空间;对空间进行时区设置、选择分析时段、 添加筛选条件、共享设置、刷新页面、更多、添加空间内容; 对空间指标进行拖拽、设置、删除、下载
分析师	查看、 他人空 间-编辑	查看所有空间;对空间指标进行页面临时操作; 对他人创建的公共空间进行复制、时区设置、选择分析时段、 添加筛选条件、刷新页面数据、导出PDF;对空间指标进行下载
普通用户	查看	查看所有空间;对空间指标进行页面临时操作

3页面概览

数据空间分为三大模块,分别是空间目录区、基本功能区、数据展示区

- 空间目录区:包括分组/子分组/空间的增删改查操作、管理空间
- 基本功能区:包括时区设置、选择全局时间范围、添加全局筛选条件、共享设置、刷新、性能设置、演示模式、从书签添加、添加用户标签、新建组件、新建描述等功能
- 数据展示区:用于各个数据分析模型、SQL查询、用户标签在空间的可视化展示及页面的功能操作

4 功能结构

4.1 空间目录区

1.新建分组/空间

有「管理我的空间」权限的成员可在空间目录区点击右上角的[+],选择「新建分组」或「新建空间」,然后在「我的空间」中 新建分组或空间;有「管理公共空间」权限的成员可以在「我的空间」和「公共空间」中新建分组或空间。

2.分组/空间的基本操作(1)分组操作:点击分组后面的[…],可选择重命名、删除分组

(2) 子分组操作:点击子分组后面的[…],可选择重命名、移动子分组、删除子分组

(3) 空间操作:点击空间后面的[…],可选择重命名、移动空间、复制空间、删除空间

3.管理空间 点击右上角「管理空间」的图标,进入管理空间的页面;有「管理我的空间」权限的成员可以在管理空间页面管理「我的空间」中的空间;有「管理公共空间」权限的成员可以在管理空间页面管理「我的空间」和「公共空间」中的空间。

- (1) 管理空间列表包括空间名称、空间类别、所属分组、创建人、创建时间、共享类型、操作等信息
- (2) 可通过共享空间、重命名空间、移动空间、复制空间、删除空间等操作对自建空间进行管理

4.2 基本功能区

1.空间重命名

点击「修改」图标,可对当前空间重命名。

2.时区设置

选择分析的计算时区,默认选择项目时区。可在「项目管理-分析配置-时区设置」中设置是否开启多时区功能。

3.选择全局时间范围

(1)选择的全局时间范围在计算时对空间中(除用户标签、SQL查询、属性分析之外)的每个指标组件都生效,页面上各指标 组件的时间范围将一并临时调整,并重新计算分析结果。

(2) 当切换空间或页面后,此时间范围不支持保留。

- (3) 刷新空间时,此时间范围保留。
- (4) 若取消选择全局时间范围,各指标组件的时间范围将变为原始时间并重新计算原始时间范围的分析结果。

4.添加全局筛选

(1) 添加的全局筛选条件在计算时对空间中(除用户标签、SQL查询、漏斗分析、间隔分析之外)的每个指标组件都生效,并 重新计算分析结果。

(2)全局筛选支持对事件属性、用户属性、用户标签、用户分群进行筛选,根据不同数据类型的筛选项,筛选规则也有所不同。

- 若想要添加多条筛选条件:点击「添加并列条件」可添加二级筛选条件,最多只能添加到二级,筛选条件添加条数不限。
- 添加新的条件需要设定和原条件的关联关系,点击"且/或"可切换关联关系。
- 筛选项目的内容来源如下:

事件描述	筛选项目
元事件	所选事件的所有事件属性并集、所有用户属性、所有用户标签、 所有用户分群

• 不同数据类型的筛选项目支持的条件筛选规则汇总如下:

筛选项目的数据类型	条件筛选可选项
文本型	等于、不等于、包括、不包括、有值、无值、正则匹配、 正则不匹配
数值型	等于、不等于、小于、小于等于、大于、大于等于、有值、 无值、区间
时间型	位于区间、小于、小于等于、大于、大于等于、相对当前日期 (区间)、相对当前日期(过去)、有值、无值、等于、 不等于
布尔型	为真、为假、有值、无值

- (3) 修改空间中某个指标组件的内容,只重新计算该指标组件的分析结果,不影响空间整体的筛选条件。
- (4) 新增的指标组件将按添加的筛选条件整体计算分析结果。

(5)当切换空间或页面后,此筛选条件不支持保留,如需保留,可将该筛选条件添加到「我的收藏」中或者将筛选条件设为默认的筛选条件。

(6) 刷新空间时,此筛选条件保留。

(7) 一些常用的筛选条件可以添加到个人的收藏列表,该收藏列表与个人账号绑定。

- 在「我的收藏」页面中,鼠标悬浮在收藏名称处可以查看收藏的筛选条件
- 可以将某条收藏设为默认,「设为默认」的收藏将在收藏列表中置顶,「取消默认」后将回到收藏列表原位置
- 点击「计算」,空间中所有指标组件将按该筛选条件进行重新计算,并展示筛选后的可视化结果
- 点击「重命名」,将重命名该收藏
- 点击「删除」,将删除该收藏

5.共享设置

支持将「我的空间」下的自建空间共享给其他项目成员查看或协作,实现项目成员的共同协作。共享成功后将通过「消息助 手」通知该成员,同时将该空间自动添加到对方的「我的空间」-「与我分享的」分组中。

(1) 共享设置默认为「仅自己可见」,若想为自建空间添加共享成员,可点击「指定成员-新增成员」来添加空间的共享成员, 可以共享给全部成员或者指定的角色、账号。

- 全部成员:项目内全部成员可见
- 角色:项目内该类角色成员均可见
- 账号:项目内该账号成员可见

(2) 添加的共享成员默认「仅查看」空间的内容,若希望与该成员共同协作空间的内容,可切换该成员的权限为「可协作」。

(3) 空间协作者的权限如下:

- 选择全局时间范围、添加全局筛选条件
- 添加/删除/替换空间默认的筛选条件
- 修改空间的共享设置、性能设置
- 查看空间演示模式
- 新增/删除空间中的指标组件、空间描述
- 切换空间中的指标组件的窗口尺寸大小、图表类型,更改设置
- 可下载空间各指标组件的全量数据
- 全局刷新及单个指标组件刷新

注意:普通成员仅具有可查看权限

6.刷新 可点击「刷新」图标对页面数据进行实时更新。

7.更多

```
(1)性能设置
为了优化数据的查询性能,我们提供了性能设置功能,包括三个方面:数据更新、计算精度和时区设置。
```

- 数据更新:默认「默认更新」,即空间加载后实时更新数据;也可以选择「每日02:00更新」或「超过多少分钟更新」进行 定时更新数据
- 计算精度:默认「完全精确」,即空间中所有分析结果均完全精确计算;如果对查询效率有需求,可选择「近似计算」, 即近似计算触发用户数、人均次数、人均值和去重数,其他数据精确计算
- 时区设置:开启后,默认选择项目时区,可在「项目管理-分析配置-时区设置」中设置是否开启多时区功能。选择好固定空间时区之后,将按照固定时区计算并查看空间中所有分析结果,同时被共享人也只能按照该时区查看所有分析结果
- 点击「确定」按钮后生效,该设置只对本空间生效

(2) 演示模式

如果只想查看当前空间信息,不被导航栏和菜单栏分散注意力时,点击「演示模式」进入空间演示模式,在浏览器中全屏查 看。在演示模式下,空间操作与非演示模式基本一致,有以下异同点:

- 空间创建者仍可以更改或删除空间指标组件,但不能新增空间指标组件及空间描述
- 基本功能支持:空间重命名、时区设置、选择时间范围、添加全局筛选、刷新功能
- 支持点击指标名进入分析页面或标签详情页,点击数据表中的链接可下钻至事件明细表和用户列表

点击「退出演示模式」图标,可直接退出演示模式,即回到当前空间的非演示状态。

8.添加空间内容

(1) 从书签添加

「书签」模块用以保存「数据分析」模块四大分析的查询设置和查询结果。您可以从书签列表中选择已有的书签在空间内展示,可多选,点击「确定」按钮即可在「数据展示区」展示所选的书签的可视化结果。

注意:书签列表只展示当前账号自建的书签

(2) 添加用户标签

您可以从用户标签列表中选择已有的用户标签在空间内展示,可多选,点击「确定」按钮即可在「数据展示区」展示所选的用 户标签的可视化结果。

(3)新建组件 您还可以通过新建分析模型的方式在空间添加展示内容。

STEP1:点击「新建组件」,选择需要使用的分析模型 STEP2:进入对应的分析模型页面后,进行查询设置并展示可视化结果,然后点击「保存」,选择「添加到数据空间」,点击 「确定」按钮即可完成新建,同时在「数据展示区」展示刚刚配置好的分析模型的可视化结果

(4) 新建描述

点击「新建描述」后弹出「新建描述」弹窗,描述标题必填。一个空间中可以添加多个描述,且可通过拖拽移动描述的位置。 通过描述可以将空间分成多个板块,以便于解释说明当前指标的业务含义和内容。

4.3 数据展示区

(1) 新建空间时的初始页面

对于新建空间,初始页面无任何指标组件,需通过「新建组件」、「从书签添加」、「添加用户标签」来新增空间指标组件。

(2) 添加指标组件后的页面

- 「数据展示区」用以展示多个指标组件的分析结果,目前支持事件分析、漏斗分析、间隔分析、留存分析、分布分析、路径分析、属性分析、SQL查询、用户标签的图表展示
- 支持空间指标组件的拖拽
- 支持调整指标组件的窗口尺寸大小
- 支持切换指标组件的图表展示类型
- 支持单个指标组建刷新
- 支持修改指标组件的配置选项
- 支持删除空间中的指标组件
- 支持下载指标组件的数据表全量数据

s

书签

帮助中心 > 功能手册 > 其他 > 书签

1 概述

书签是帮助您保存常用分析查询的工具,将需沉淀的分析指标保存为书签,可以直接在书签模块查看书签数据,也可以将书签添加到任意一个或多个空间,并在数据空间模块查看书签数据。通过数据空间实现书签共享,从而实现数据价值。

2 位置和使用权限

操作位置: 书签

使用权限: 不同的项目角色操作权限如下

项目角色	权限	权限明细
管理员	编辑	对书签进行查看、新建、修改、删除
分析师	编辑	对书签进行查看、新建、修改、删除
普通用户	无权限	无权限

3页面概览

书签模块主要包括书签列表的展示以及书签的增删改查等功能,同时书签支持在数据空间中展示使用。

4 功能结构

4.1 列表展示

书签列表包括书签名称、统计时段、类型、所属数据空间、创建时间、操作等信息。

- (1) 点击书签名称可进入对应的分析页面
- (2) 统计时段即保存书签时所选的时间范围
- (3) 所属数据空间即同时展示该书签的空间名称
- (4) 书签创建人可删除/批量删除书签

4.2 查看书签

当想查看某条书签具体的查询设置或可视化结果时,点击书签列表中的书签名称就可以进入到对应的分析页面查看。

4.3 新建书签

新建书签有2种方式: 1.点击「新建书签」,然后选择某个分析模型,进入分析页面并设置完查询条件后,点击「保存」,选择「保存为书签」,即可 新建书签。

2.从书签列表点击某个书签名称进入分析页面,更改查询条件后,点击「保存」,选择「保存为新书签」,即可新建书签。

4.4 编辑书签

从书签列表点击某个书签名称进入分析页面,更改查询条件后,点击「保存」,选择「更新当前书签」,即可更新当前书签, 同时支持修改当前书签的书签名称。

4.5 删除书签

当不需要某条书签或部分书签时,点击删除/批量删除即可完成删除。

数据分析

事件分析

帮助中心 > 功能手册 > 数据分析 > 事件分析

1 概述

事件分析是指基于事件的指标维度、属性筛选、分组查看等功能,对事件的整体趋势情况进行分析。通过事件分析,可以计算用户在指定时间范围内发生特定行为事件的聚合算子,从而帮助产品运营优化运营策略,实现用户的精细化运营和精准营销。

事件分析可以解决以下示例问题: 1.过去7天各渠道的支付订单的人均次数的趋势情况 2.最近半年来哪个城市的用户登录次数最高及分布情况 3.最近一个月来自上海的,关注公众号的去重用户数中按照年龄段的分布情况

2 位置和使用权限

操作位置:数据分析-事件分析

使用权限: 不同的项目角色操作权限如下

项目角色	权限	权限明细
管理员	查看、 功能使用	查看sql代码、使用分析并对分析结果进行保存, 对分析数据进行下载,查看事件列表/用户列表
分析师	功能使用	使用分析并对分析结果进行保存,对分析数据进行下载, 查看事件列表/用户列表
普通用户	无权限	无权限

3页面概览

事件分析分为三大模块,分别是查询设置模块、分析结果展示模块、基本功能模块

- 查询设置模块:包括选择分析指标、添加全局筛选条件和添加分组项
- 分析结果展示模块:包括是时间粒度、时间组件、图形展示和表格展示
- 基本功能模块:包括时区选择、近似计算、保存功能、刷新功能、下载功能和查看SQL代码

4 功能结构

4.1 查询设置

1. 添加分析指标

计算格式:选择「事件」+「事件属性」+「聚合算子」或「事件」+「聚合算子」作为分析指标,也可选择多个指标之间的基本

运算作为自定义指标。

(1)选择事件:可以选择任意一个元事件。若分析主体为用户自定义的分析主体且来源于事件属性,则下拉框中可选的元事件 只展示包含该事件属性(分析主体)的元事件。

(2) 选择聚合算子:聚合算子根据事件属性的数据类型有所不同。

(3) 当聚合算子选择「触发用户数」或「人均次数」或「数值型事件属性的人均值」时,可进行分析主体的切换。分析主体默 认为海纳ID,若以设备ID作为分析主体,则将不同设备ID的用户视作不同用户;若以账号ID作为分析主体,则将不同账号ID的用 户视作不同用户。若当前下拉框中没有想要的分析主体,可至「项目管理-分析配置-分析主体」中进行相关配置。

(4) 可对事件添加筛选条件,内容来源于事件属性、用户属性、用户标签、用户分群。

• 不同数据类型的事件属性支持的聚合计算规则汇总如下:

事件	事件属性 (数据类型)	聚合算子
元事件		总次数、触发用户数、人均次数
元事件	事件属性 (数值型)	总和、均值、人均值、中位数、最大值、最小值、 去重数
元事件	事件属性 (布尔型)	为真数、为假数、为空数、不为空数、去重数
元事件	事件属性 (其他类型)	去重数

2. 筛选条件

支持对事件属性、用户属性、用户标签、用户分群进行筛选,根据不同数据类型的筛选项,筛选规则也有所不同。

(1) 若想要添加多条筛选条件:点击「添加并列条件」可添加二级筛选条件,最多只能添加到二级,筛选条件添加条数不限。

(2) 添加新的条件需要设定和原条件的关联关系,点击"且/或"可切换关联关系。

• 筛选项目的内容来源分为以下两种情况:

情况	事件描述	筛选项目
仅一条指标	元事件	该事件下的事件属性、所有用户属性、所有用户标签、 所有用户分群
多条指标	元事件	所选事件的所有事件属性并集、所有用户属性、 所有用户标签、所有用户分群

• 不同数据类型的筛选项目支持的条件筛选规则汇总如下:

筛选项的数据类型 条件筛选可选项

文本型	等于、不等于、包括、不包括、有值、无值、正则匹配、 正则不匹配
数值型	等于、不等于、小于、小于等于、大于、大于等于、有值、 无值、区间
时间型	位于区间、小于、小于等于、大于、大于等于、相对当前日期 (区间) 、相对当前日期(过去) 、有值、无值、等于、不等于
布尔型	为真、为假、有值、无值

3. 分组项

支持将事件属性、用户属性、用户标签、用户分群设置为分组项,不同数据类型的分组项有不同的分组设定方式。

- 针对数值型分组项,分组方式包括默认区间、离散数字和自定义区间三种
- 针对时间型分组项,分组方式包括汇总与不汇总两种,其中汇总可选择按月、按周、按天、按小时、按分钟
- 针对其他类型的分组项,将直接按照离散方式展示
- 分组项目的内容来源分为以下两种情况:

	情况	事件描述	分组项目
	仅一条指标	元事件	不分组、该事件下的事件属性、所有用户用户属性、 用户标签、所有用户分群
	多条指标	元事件	不分组、所选事件的所有事件属性并集、所有用户属性、 所有用户标签、所有用户分群

4.2 分析结果展示

1. 时间组件

(1)时间粒度:时间粒度默认选择「按天」,支持的时间粒度包括:按天、按周、按月、按小时、按分钟、合计。

(2) 时间范围:默认选中动态时间「过去7天」,可点击选择其他静态/动态时间。

2. 指标选项和分组选项

(1) 指标选项:默认显示所有分析指标,最多勾选 10 项,可在复选框中重新勾选需要显示的分析指标。

(2)分组选项:默认勾选前10项分组,可在复选框中重新勾选需要显示的分组。没有设置数据排列方式时,分组选项的排序逻辑为图表中该图例的总值按分析指标之和从大到小排序;设置了数据排列方式后,分组选项的排序逻辑与当前设置的排序规则 保持一致。

3. 图形展示

在数据可视化部分,支持切换展示图形(趋势图、堆积图、累计图、分布图、饼图)、数据排列方式、显示数据等操作,可灵活设置数据的展示方式。数据最终的展示方式将按照时间粒度、时间范围、指标选项、分组选项、数据排列方式、图形样式共同决定。

趋势图

事件分析

- 堆积图
- 累计图
- 分布图
- 併图

4. 表格展示

- 数据表的计算方式支持阶段汇总、阶段均值、最大值、最小值、合计
- 数据表展示每个指标和分组排列组合后的分析指标值

5. 查看事件明细表

添加分析指标时,若选择分析事件的「总次数」,通过点击数据表中的具体数值,可链接跳转至事件明细表页面,查看更多事件属性信息:

- 支持对事件包含的事件属性进行筛选,选择需要展示和下载时需要的事件属性
- 事件明细表支持实时刷新数据、导出数据功能

6. 查看用户列表

添加分析指标时,若选择分析事件的「触发用户数」且分析主体为「海纳ID」,通过点击数据表中的具体数值,可链接跳转至用 户列表页面,查看已选人群的更多用户属性信息:

- 用户列表支持查看用户其他属性特征,了解用户更多信息
- 用户列表支持刷新数据、导出数据功能
- 用户列表支持将已选人群保存为结果分群,可至分群管理列表中查看

4.3 基本功能

1.时区设置

选择分析的计算时区,默认选择项目时区。可在「项目管理-分析配置-时区设置」中设置是否开启多时区功能。

2. 近似计算

默认开启「近似计算」,开启后将近似计算触发用户数、人均次数、人均值和去重数,极大地提高查询性能,同时将计算结果的标准误差控制在 0.3%以内。

3. 保存

事件分析的结果可以保存为书签或同步至数据空间,方便后续进行查询和修改。书签和数据空间的操作详情可至「书签」和 「数据空间」中查看。

- 保存为书签,保存后可在书签列表查看该事件分析结果。
- 同步至数据空间的操作,同步后可在数据空间处查看该事件分析结果。

4. 刷新

可点击「刷新」图标对页面数据进行实时更新。

5. 下载

当查询数据数过多时,查询结果最多只展示前1000条记录,若想查看完整的数据,可点击「下载」获取完整数据(最多下载 100万条),下载的文件格式为csv。

5 实例

5.1 查看某段时间内小程序启动的总次数在不同城市的分布情况

如下图所示,通过事件分析能够直观地分析2023-03-13至2023-03-16之间小程序启动的总次数在不同城市的分布情况。从分布 图及表格的数据可以看出,小程序启动的总次数排在前三的城市分别是上海市、重庆市和济南市。您也可以按注册渠道进行分 组,同时选择「按数据量降序」查看,有助于快速了解APP在各个渠道启动次数的分布情况,以便后期的营销策略侧重应用于 APP启动次数较高的渠道。
漏斗分析

帮助中心 > 功能手册 > 数据分析 > 漏斗分析

1 概述

漏斗分析用于分析用户在指定步骤下每一步的转化与流失情况。通过漏斗分析,不仅可以计算指定时间范围内用户在整个漏斗 过程的转化与流失情况,还可以计算用户在各个指定步骤下的转化与流失情况,从而有针对性地帮助产品人员优化产品功能 点,提升产品转化率。

漏斗分析可以解决以下示例问题: 1.分析用户首次登录到二次登录的转化情况 2.分析7天内用户从注册到成为付费用户的转化和流失情况 3.分析不同渠道的转化情况差异

2 位置和使用权限

操作位置:数据分析-漏斗分析

使用权限: 不同的项目角色操作权限如下

项目角色	权限	权限明细
管理员	查看、 功能使用	查看sql代码、使用分析并对分析结果进行保存, 对分析数据进行下载,查看事件列表/用户列表
分析师	功能使用	使用分析并对分析结果进行保存,对分析数据进行下载, 查看事件列表/用户列表
普通用户	无权限	无权限

3页面概览

漏斗分析分为三大模块,分别是查询设置模块、分析结果展示模块、基本功能模块

- 查询设置模块:包括创建漏斗步骤、添加关联属性、选择分析窗口期、添加全局筛选和添加分组项
- 分析结果展示模块:包括时间组件、步骤选框、图形展示和表格展示
- 基本功能模块:包括时区选择、分析主体选择、近似计算、保存功能、刷新功能、下载功能和查看SQL代码

4 功能结构

4.1 查询设置

1. 添加漏斗步骤

(1)选择事件:一个漏斗至少得有2个步骤,1个步骤可以选择任意一个元事件。若分析主体为用户自定义的分析主体且来源于 事件属性,则下拉框中可选的元事件只展示包含该事件属性(分析主体)的元事件。

(2)可以通过「复制」、「插入」、「添加漏斗步骤」等功能添加漏斗步骤,一个漏斗最多只能设置30个步骤。

(3) 可为漏斗步骤添加指标名称和筛选条件,筛选条件内容来源于事件属性、用户属性、用户标签、用户分群。

(4) 关联属性

在一些业务场景中,需要支持设置漏斗任意几步的属性进行关联,以确保数据的准确性。点击「关联属性」,可在当前漏斗各步骤的事件属性中选择一个事件属性作为关联属性。漏斗不同步骤关联的属性可以是相同属性,也可以是不同属性,但是要求 所选属性的数据类型必须一致,且至少设置两个漏斗步骤的关联属性。

(5) 窗口期

窗口期是指用户完成漏斗步骤的时间限制,即用户触发步骤1起,在窗口期内完成整个漏斗,才算作完成了一次成功的转化。但由于触发步骤1的时间点不同,开始漏斗和结束漏斗的绝对日期可能不同。

2. 筛选条件

支持对事件属性、用户属性、用户标签、用户分群进行筛选,根据不同数据类型的筛选项,筛选规则也有所不同。 (1)若想要添加多条筛选条件:点击「添加并列条件」可添加二级筛选条件,最多只能添加到二级,筛选条件添加条数不限。

(2) 添加新的条件需要设定和原条件的关联关系,点击"且/或"可切换关联关系。

• 筛选项目的内容来源如下:

事件描述	筛选项目
元事件	所选事件下的所有事件属性(分事件)、所有用户属性、所有用户标签、 所有用户分群

• 不同数据类型的筛选项目支持的条件筛选规则汇总如下:

筛选项的数据类型	条件筛选可选项
文本型	等于、不等于、包括、不包括、有值、无值、正则匹配、 正则不匹配
数值型	等于、不等于、小于、小于等于、大于、大于等于、有值、 无值、区间
时间型	位于区间、小于、小于等于、大于、大于等于、相对当前日期 (区间)、相对当前日期(过去)、有值、无值、等于、不等于
布尔型	为真、为假、有值、无值

3. 分组项

支持将事件属性、用户属性、用户标签、用户分群设置为分组项,不同数据类型的分组项有不同的分组设定方式。

- 针对数值型分组项,分组方式包括默认区间、离散数字和自定义区间三种
- 针对时间型分组项,分组方式包括汇总与不汇总两种,其中汇总可选择按月、按周、按天、按小时、按分钟

- 针对其他类型的分组项,将直接按照离散方式展示
- 分组项目的内容来源如下:

事件描述	分组项目
元事件	不分组、所选事件下的所有事件属性(分事件)、所有用户属性、 所有用户标签、所有用户分群

4.2 分析结果展示

1. 步骤选框

默认选择全步骤展示,也可选择从步骤 M 到步骤 N进行展示。

2. 时间范围

默认选中动态时间「过去7天」,可点击选择其他静态/动态时间。

3. 分组选项

(1) 趋势图对应的分组选项默认勾选前10项分组,可在复选框中重新勾选需要显示的分组

(2)转化图对应的分组选项默认勾选前4项分组,可在复选框中重新勾选需要显示的分组

4. 图形展示

在数据可视化部分,支持切换展示图形(趋势图、转化图)、显示数据等操作,可灵活设置数据的展示方式。数据最终的展示 方式将按照步骤选框、时间范围、分组选项、图形样式共同决定。

- 转化图
- 趋势图

5. 表格展示

- 转化图对应的表格
- (1) 数据表有转化、流失两种展示方式

(2) 有分组项时,数据表展示分组项排列组合后在每个漏斗步骤下的累计转化人数及累计转化率/单步流失人数及单步流失率;无分组项时,数据表展示总体在每个漏斗步骤下的累计转化人数及累计转化率/单步流失人数及单步流失率

- (3) 点击人数可以查看用户列表
- 趋势图对应的表格

(1) 数据表有转化、流失两种展示方式

(2) 有分组项时,数据表展示分组项排列组合后在所选时间范围下的转化均值/流失均值;无分组项时,数据表展示总体在所选时间范围下的转化均值/流失均值

6. 查看用户列表

通过点击数据表中的人数,可链接跳转至用户列表页面,查看已选人群的更多用户属性信息:

- 用户列表支持查看用户其他属性特征,了解用户更多信息
- 用户列表支持刷新数据、导出数据功能
- 用户列表支持将已选人群保存为结果分群,可至分群管理列表中查看

4.3 基本功能

1.时区设置

选择分析的计算时区,默认选择项目时区。可在「项目管理-分析配置-时区设置」中设置是否开启多时区功能。

2.选择分析主体

分析主体:默认为海纳ID,若以设备ID作为分析主体,则将不同设备ID的用户视作不同用户;若以账号ID作为分析主体,则将不同账号ID的用户视作不同用户。若当前下拉框中没有想要的分析主体,可至「项目管理-分析配置-分析主体」中进行相关配置。

3. 近似计算

默认开启「近似计算」,开启后将近似计算触发用户数,极大地提高查询性能,同时将计算结果的标准误差控制在 0.3%以内。

4. 保存

漏斗分析的结果可以保存为书签或同步至数据空间,方便后续进行查询和修改。书签和数据空间的操作详情可至「书签」和「数据空间」中查看。

- 保存为书签,保存后可在书签列表查看该漏斗分析结果。
- 同步至数据空间的操作,同步后可在数据空间处查看该漏斗分析结果。
- 5. 刷新

可点击「刷新」图标对页面数据进行实时更新。

6. 下载

当查询数据数过多时,查询结果最多只展示前1000条记录,若想查看完整的数据,可点击「下载」获取完整数据(最多下载 100万条),下载的文件格式为csv。

5 实例

5.1 查看一个完整漏斗的转化情况

如下图所示,通过"小程序启动-小程序页面浏览-点击关注"这样一个完整的漏斗流程,能够直观地了解到来自不同操作系统的用 户转化情况,以及漏斗每一步的转化情况。从转化图及表格的数据可以看出ios用户在"小程序页面浏览-点击关注"这步转化率下 降45%,根据该分析结果我们便可以对小程序页面做出优化,以此提高用户的转化率。

间隔分析

帮助中心 > 功能手册 > 数据分析 > 间隔分析

1 概述

间隔分析用于分析用户行为序列中的两个指定事件之间的转化时长。通过间隔分析,可以精准分析指定时间范围内用户在同一 行为或不同行为之间的转化时间间隔,从而有针对性地帮助运营人员调整运营策略,增强用户黏性,实现精准营销。

间隔分析可以解决以下示例问题: 1.分析用户产生两次购买行为之间的转化时长分布 2.分析1天内用户从注册到首次加购的转化时长分布 3.分析不同年龄段的用户从登录到购买的转化时长差异

2 位置和使用权限

操作位置:数据分析-间隔分析

使用权限: 不同的项目角色操作权限如下

项目角色	权限	权限明细
管理员	查看、 功能使用	查看sql代码、使用分析并对分析结果进行保存, 对分析数据进行下载,查看事件列表/用户列表
分析师	功能使用	使用分析并对分析结果进行保存,对分析数据进行下载, 查看事件列表/用户列表
普通用户	无权限	无权限

3页面概览

间隔分析分为三大模块,分别是查询设置模块、分析结果展示模块、基本功能模块

- 查询设置模块:包括选择起始事件和结束事件、添加关联属性、选择间隔上限、添加全局筛选和添加分组项
- 分析结果展示模块:包括时间粒度、时间组件、图形展示和表格展示
- 基本功能模块:包括时区选择、分析主体选择、近似计算、保存功能、刷新功能、下载功能和查看SQL代码

4 功能结构

4.1 查询设置

(1) 选择事件:起始事件和结束事件可根据不同的业务场景选择同一事件或不同事件进行分析。若分析主体为用户自定义的分

^{1.} 选择起始事件和结束事件

析主体且来源于事件属性,则下拉框中可选的元事件只展示包含该事件属性(分析主体)的元事件。

(2) 可为事件添加指标名称和筛选条件,筛选条件内容来源于所选事件的事件属性。

(3) 关联属性

在一些业务场景中,需要将间隔的起始事件和结束事件的事件属性进行关联,以确保数据的准确性。点击「关联属性」,可在 当前所选事件的事件属性中选择一个属性作为关联属性,关联的属性可以是相同属性,也可以是不同属性,但要求所选属性的 数据类型必须一致。

(4) 间隔上限

间隔上限是在计算间隔完成后进行的,相当于约定了起始事件和结束事件之间产生的间隔数据的最大值,默认为1小时,可自定 义间隔上限。

2. 筛选条件

支持对事件属性、用户属性、用户标签、用户分群进行筛选,根据不同数据类型的筛选项,筛选规则也有所不同。 (1)若想要添加多条筛选条件:点击「添加并列条件」可添加二级筛选条件,最多只能添加到二级,筛选条件添加条数不限。

(2) 添加新的条件需要设定和原条件的关联关系,点击"且/或"可切换关联关系。

• 筛选项目的内容来源如下:

事件描述	筛选项目
元事件	所选事件下的所有事件属性(分事件)、所有用户属性、所有用户标签、 所有用户分群

• 不同数据类型的筛选项目支持的条件筛选规则汇总如下:

筛选项的数据类型	条件筛选可选项
文本型	等于、不等于、包括、不包括、有值、无值、正则匹配、 正则不匹配
数值型	等于、不等于、小于、小于等于、大于、大于等于、有值、 无值、区间
时间型	位于区间、小于、小于等于、大于、大于等于、相对当前日期 (区间)、相对当前日期(过去)、有值、无值、等于、不等于
布尔型	为真、为假、有值、无值

3. 分组项

支持将事件属性、用户属性、用户标签、用户分群设置为分组项,不同数据类型的分组项有不同的分组设定方式。

- 针对数值型分组项,分组方式包括默认区间、离散数字和自定义区间三种
- 针对时间型分组项,分组方式包括汇总与不汇总两种,其中汇总可选择按月、按周、按天、按小时、按分钟
- 针对其他类型的分组项,将直接按照离散方式展示
- 分组项目的内容来源如下:

事件描述	分组项目
元事件	不分组、所选事件下的所有事件属性(分事件)、所有用户属性、 所有用户标签、所有用户分群

4.2 分析结果展示

1. 时间组件

(1)时间粒度:时间粒度默认选择「按天」,支持的时间粒度包括:按天、按周、按月、按小时、合计。

(2)时间范围:默认选中动态时间「过去7天」,可点击选择其他静态/动态时间。

3. 分组选项

(1) 箱线图对应的分组选项默认勾选前4项,可在复选框中重新勾选需要显示的分组选项。分组选项上限为4项,当时间粒度选择「合计」时,分组选项上限为50项

(2) 直方图可选择分组选项和时间选项,默认勾选「总体」,可在单选框中重新勾选需要显示的分组选项和时间选项

4. 图形展示

在数据可视化部分,支持切换展示图形(箱线图、直方图)、显示数据、切换人数/次数等操作,可灵活设置数据的展示方式。 数据最终的展示方式将按照时间粒度、时间范围、分组选项、时间选项、人数/次数、图形样式共同决定。

- 箱线图
- 直方图

5. 表格展示

• 箱线图对应的表格

(1) 有分组项时,点击「日期列」可展开子表,子表展示各分组项对应的用户数、间隔数、平均值、最小值、下四分位、中位数、上四分位、最大值

- (2) 点击用户数可以查看用户列表
- 直方图对应的表格
- (1) 有分组项时,点击「日期列」可展示子表,子表展示各分组项在各个时间间隔区间下的「人数」与「间隔数」
- (2) 点击人数可以查看用户列表

6. 查看用户列表

通过点击数据表中的用户数/人数,可链接跳转至用户列表页面,查看已选人群的更多用户属性信息:

- 用户列表支持查看用户其他属性特征,了解用户更多信息
- 用户列表支持刷新数据、导出数据功能
- 用户列表支持将已选人群保存为结果分群,可至分群管理列表中查看

4.3 基本功能

1.时区设置

选择分析的计算时区,默认选择项目时区。可在「项目管理-分析配置-时区设置」中设置是否开启多时区功能。

2.选择分析主体

分析主体:默认为海纳ID,若以设备ID作为分析主体,则将不同设备ID的用户视作不同用户;若以账号ID作为分析主体,则将不同账号ID的用户视作不同用户。若当前下拉框中没有想要的分析主体,可至「项目管理-分析配置-分析主体」中进行相关配置。

3. 近似计算

默认开启「近似计算」,开启后将近似计算触发用户数,极大地提高查询性能,同时将计算结果的标准误差控制在 0.3%以内。

4. 保存

间隔分析的结果可以保存为书签或同步至数据空间,方便后续进行查询和修改。书签和数据空间的操作详情可至「书签」和 「数据空间」中查看。

- 保存为书签,保存后可在书签列表查看该间隔分析结果。
- 同步至数据空间的操作,同步后可在数据空间处查看该间隔分析结果。

5. 刷新

可点击「刷新」图标对页面数据进行实时更新。

6. 下载

当查询数据数过多时,查询结果最多只展示前1000条记录,若想查看完整的数据,可点击「下载」获取完整数据(最多下载 100万条),下载的文件格式为csv。

5 实例

5.1 查看不同区域的用户产生两次页面浏览行为的转化时长分布

如下图所示,从箱线图及表格的数据可以看出所选时间范围内不同区域的用户进行两次小程序页面浏览行为的转化时长最大 值、上四分位、中位数、下四分位、最小值。您也可以按年龄进行分组,查看不同年龄段的用户产生同一行为或不同行为的转 化时长分布情况,有助于快速了解产品在不同年龄段的转化时长和受欢迎程度。

留存分析

帮助中心 > 功能手册 > 数据分析 > 留存分析

1 概述

留存分析用于分析进行某项初始行为的用户中,会进行后续行为的人数和比例。通过留存分析,可以计算指定时间范围内用户 在指定行为间的留存与流失情况,从而周期性地监测产品,针对性地调整产品结构、优化产品体验,助力企业及时采取措施挽 留流失用户,提升产品转化率。

留存分析可以解决以下示例问题: 1.分析用户使用产品后的3个月内,进行购买转化的留存与流失情况 2.改进某项产品功能后,验证是否有用户因优化功能点而多使用产品几个月 3.分析不同城市的用户留存与流失情况

2 位置和使用权限

操作位置:数据分析-留存分析

使用权限: 不同的项目角色操作权限如下

项目角色	权限	权限明细
管理员	查看、 功能使用	查看sql代码、使用分析并对分析结果进行保存, 对分析数据进行下载,查看事件列表/用户列表
分析师	功能使用	使用分析并对分析结果进行保存,对分析数据进行下载, 查看事件列表/用户列表
普通用户	无权限	无权限

3页面概览

留存分析分为三大模块,分别是查询设置模块、分析结果展示模块、基本功能模块

- 查询设置模块:包括选择初始行为和后续行为、添加同时展示、添加全局筛选和添加分组项
- 分析结果展示模块:包括时间组件、分析期限、分析类型、图形展示和表格展示
- 基本功能模块:包括时区选择、分析主体选择、近似计算、保存功能、刷新功能、下载功能和查看SQL代码

4 功能结构

4.1 查询设置

1. 选择初始行为和后续行为

(1)选择事件:初始行为和后续行为可根据不同的业务场景选择同一事件或不同事件进行分析。若分析主体为用户自定义的分析主体且来源于事件属性,则下拉框中可选的元事件只展示包含该事件属性(分析主体)的元事件。

(2) 可为事件添加指标名称和筛选条件,筛选条件内容来源于所选事件的事件属性。

(3) 同时展示

点击「同时展示」将支持添加同时展示的指标,用于统计留存用户的指标值。 指标的计算格式:选择「事件」+「事件属性」+「聚合算子」或「事件」+「聚合算子」作为分析指标,也可选择多个指标之间 的基本运算作为自定义指标。

2. 筛选条件

支持对用户属性、用户标签、用户分群进行筛选,根据不同数据类型的筛选项,筛选规则也有所不同。 (1)若想要添加多条筛选条件:点击「添加并列条件」可添加二级筛选条件,最多只能添加到二级,筛选条件添加条数不限。

- (2) 添加新的条件需要设定和原条件的关联关系,点击"且/或"可切换关联关系。
- 筛选项目的内容来源如下:所有用户属性、所有用户标签、所有用户分群
- 不同数据类型的筛选项目支持的条件筛选规则汇总如下:

筛选项的数据类型	条件筛选可选项
文本型	等于、不等于、包括、不包括、有值、无值、正则匹配、 正则不匹配
数值型	等于、不等于、小于、小于等于、大于、大于等于、有值、 无值、区间
时间型	位于区间、小于、小于等于、大于、大于等于、相对当前日期 (区间)、相对当前日期(过去)、有值、无值、等于、不等于
布尔型	为真、为假、有值、无值

3. 分组项

支持将事件属性、用户属性、用户标签、用户分群设置为分组项,不同数据类型的分组项有不同的分组设定方式。

- 针对数值型分组项,分组方式包括默认区间、离散数字和自定义区间三种
- 针对时间型分组项,分组方式包括汇总与不汇总两种,其中汇总可选择按月、按周、按天、按小时、按分钟
- 针对其他类型的分组项,将直接按照离散方式展示
- 分组项目的内容来源如下:

事件描述	分组项目
元事件	不分组、所选事件下的所有事件属性(分事件)、所有用户属性、 所有用户标签、所有用户分群

4.2 分析结果展示

1. 时间范围

默认选中动态时间「过去7天」,可点击选择其他静态/动态时间。

2. 分析期限

通过「分析期限」可以切换不同体量(日、周、月)下的留存与流失情况,默认选择7日,可在下拉框中进行快速选择,支持按 日、按周、按月自定义分析期限。

3. 分析类型

通过「分析类型」选定分析角度,分析类型为两种:留存与流失。分析类型选择流失时,不支持"同时展示"的指标。

4. 分组选项

分组选项默认勾选前10项分组,可在复选框中重新勾选需要显示的分组

4. 图形展示

在数据可视化部分,支持切换展示图形(趋势图、变化图)、显示数据、切换数值/百分比等操作,可灵活设置数据的展示方式。数据最终的展示方式将按照时间范围、分析期限、分析类型、分组选项、数值/百分比、图形样式共同决定。

- 趋势图
- 变化图

5. 表格展示

- 分析类型为留存时对应的表格
- (1) 有分组项时,点击「日期列」可展开子表,子表展示各分组项对应的总人数及各分析期限下的留存人数和留存率
- (2) 点击人数可以查看用户列表
- 分析类型为流失时对应的表格

(1)有分组项时,点击「日期列」可展开子表,子表展示各分组项对应的总人数及各分析期限下(不包含当日)的流失人数和 流失率

(2) 点击人数可以查看用户列表

6. 查看用户列表 通过点击数据表中的人数,可链接跳转至用户列表页面,查看已选人群的更多用户属性信息:

- 用户列表支持查看用户其他属性特征,了解用户更多信息
- 用户列表支持刷新数据、导出数据功能
- 用户列表支持将已选人群保存为结果分群,可至分群管理列表中查看

4.3 基本功能

1.时区设置

选择分析的计算时区,默认选择项目时区。可在「项目管理-分析配置-时区设置」中设置是否开启多时区功能。

2.选择分析主体

分析主体:默认为海纳ID,若以设备ID作为分析主体,则将不同设备ID的用户视作不同用户;若以账号ID作为分析主体,则将不同账号ID的用户视作不同用户。若当前下拉框中没有想要的分析主体,可至「项目管理-分析配置-分析主体」中进行相关配置。

3. 近似计算

默认开启「近似计算」,开启后将近似计算触发用户数,极大地提高查询性能,同时将计算结果的标准误差控制在 0.3%以内。

4. 保存

留存分析的结果可以保存为书签或同步至数据空间,方便后续进行查询和修改。书签和数据空间的操作详情可至「书签」和 「数据空间」中查看。

- 保存为书签,保存后可在书签列表查看该留存分析结果。
- 同步至数据空间的操作,同步后可在数据空间处查看该留存分析结果。

5. 刷新

可点击「刷新」图标对页面数据进行实时更新。

6. 下载

当查询数据数过多时,查询结果最多只展示前1000条记录,若想查看完整的数据,可点击「下载」获取完整数据(最多下载 100万条),下载的文件格式为csv。

5 实例

5.1 查看不同SDK版本下的用户从页面浏览到分享的留存情况

如下图所示,从变化图及表格的数据可以看出不同SDK版本下的用户从小程序页面浏览到小程序分享在各个分析期限下的留存人数和留存率。您也可以通过国家、城市、用户年龄等其他方面进行分组查看,或通过筛选功能仅查看中国上海的用户留存情况,有助于快速定位产品的留存人数和留存率,助力企业产品运营人员制定营销策略,快速挽回流失用户。

分布分析

帮助中心 > 功能手册 > 数据分析 > 分布分析

1 概述

分布分析主要用于分析某个事件指标的用户分布情况。可以根据每个用户聚合后的总完成次数、天数或属性值划分区间,查看 不同区间用户的数量及占比。分布分析可以通过对数据结构的分层分析,判断极端数值的占比以及对整体数据的影响,了解数 据分布的具体情况,帮助挖掘用户分布的规律,优化产品策略。

分布分析可以解决以下示例问题: 1.查看一定时间内全体用户的年龄分布情况 2.不同区间内的付费人数/次数分别有多少 3.分析用户每个月的购买频次

2 位置和使用权限

操作位置:数据分析-分布分析

使用权限: 不同的项目角色操作权限如下

项目角色	权限	权限明细
管理员	查看、 功能使用	查看sql代码、使用分析并对分析结果进行保存, 对分析数据进行下载,查看事件列表/用户列表
分析师	功能使用	使用分析并对分析结果进行保存,对分析数据进行下载, 查看事件列表/用户列表
普通用户	无权限	无权限

3页面概览

属性分析包括查询设置、分析结果展示与基本功能三部分。

- 查询设置包括用户参与事件、同时展示和添加分组项;
- 分析结果展示包括时间组件、图形展示和表格展示;
- 基本功能包括时区选择、选择分析主体、近似计算、保存功能、刷新功能、下载功能和查看SQL代码。

4 功能结构

4.1 查询设置

1. 用户参与事件

计算格式:选择「事件」+「事件属性」+「聚合算子」或「事件」+「聚合算子」作为分析指标,也可选择多个指标之间的基本

运算作为自定义指标。

(1)选择事件:可以选择任意一个元事件。若分析主体为用户自定义的分析主体且来源于事件属性,则下拉框中可选的元事件 只展示包含该事件属性(分析主体)的元事件。

(2) 选择聚合算子:聚合算子根据事件属性的数据类型有所不同。

• 不同数据类型的事件属性支持的聚合计算规则汇总如下:

事件	事件属性 (数据类型)	聚合算子
元事件		次数、天数、小时数
元事件	事件属性 (数值型)	总和、均值、中位数、最大值、最小值、去重数
元事件	事件属性 (布尔型)	为真数、为假数、为空数、不为空数、去重数
元事件	事件属性 (其他类型)	去重数

(3)可对事件添加筛选条件,内容来源于事件属性、用户属性、用户标签、用户分群,根据不同数据类型的筛选项,筛选规则 也有所不同。

1) 若想要添加多条筛选条件:点击「添加并列条件」可添加二级筛选条件,最多只能添加到二级,筛选条件添加条数不限。

2) 添加新的条件需要设定和原条件的关联关系,点击"且/或"可切换关联关系。

• 不同数据类型的筛选项支持的条件筛选规则汇总如下:

筛选项的数据类型	条件筛选可选项		
文本型	等于、不等于、包括、不包括、有值、无值、正则匹配、正则不匹配		
数值型	等于、不等于、小于、小于等于、大于、大于等于、有值、无值、 区间		
时间型	位于区间、小于、小于等于、大于、大于等于、相对当前日期 (区间) 、相对当前日期(过去) 、有值、无值、等于、不等于		
布尔型	为真、为假、有值、无值		

2. 同时展示

点击「同时展示」将支持添加同时展示的指标,用于统计留存用户的指标值。

指标的计算格式:选择「事件」+「事件属性」+「聚合算子」或「事件」+「聚合算子」作为分析指标,也可选择多个指标之间 的基本运算作为自定义指标。

3. 分组项

支持将事件属性、用户属性、用户标签、用户分群设置为分组项,不同数据类型的分组项有不同的分组设定方式。

• 针对数值型分组项,分组方式包括默认区间、离散数字和自定义区间三种

- 针对时间型分组项,分组方式包括汇总与不汇总两种,其中汇总可选择按月、按周、按天、按小时、按分钟
- 针对其他类型的分组项,将直接按照离散方式展示
- 分组项目的内容来源分为以下两种情况:

情况	事件描述	分组项目
仅一条指标	元事件	不分组、该事件下的事件属性、所有用户用户属性、 用户标签、所有用户分群
多条指标	元事件	不分组、所选事件的所有事件属性并集、所有用户属性、 所有用户标签、所有用户分群

4.2 分析结果展示

1. 时间组件

(1) 时间粒度:时间粒度默认选择「按天」,支持的时间粒度包括:按天、按周、按月、合计。

(2)时间范围:默认选中动态时间「过去7天」,可点击选择其他静态/动态时间。

2. 指标选项和分组选项

(1) 指标选项:默认显示所有分析指标,最多勾选 10 项,可在复选框中重新勾选需要显示的分析指标。

(2)分组选项:默认勾选前10项分组,可在复选框中重新勾选需要显示的分组。没有设置数据排列方式时,分组选项的排序逻辑为图表中该图例的总值按分析指标之和从大到小排序;设置了数据排列方式后,分组选项的排序逻辑与当前设置的排序规则 保持一致。

3. 图形展示

在数据可视化部分,支持切换展示图形、数据排列方式、显示数据等操作,可灵活设置数据的展示方式。数据最终的展示方式 将按照时间粒度、时间范围、是否同时展示、分组选项、数据排列方式、图形样式共同决定。

- 1) 当用户未设置同时展示时,根据用户所选的时间粒度,展示的图表类型如下:
- 2)当用户设置同时展示指标,根据用户所选的时间粒度,以及是否勾选「只看同时展示数据」,展示的图表类型如下:
 - 数据表
 - 数值分布图
 - 百分比分布图
 - 柱状图
 - 趋势图

4. 查看用户列表 通过点击数据表中的数据,可链接跳转至用户列表页面,查看已选人群的更多用户属性信息:

- 用户列表支持查看用户其他属性特征,了解用户更多信息
- 用户列表支持刷新数据、导出数据功能
- 用户列表支持将已选人群保存为结果分群,可至分群管理列表中查看

4.3 基本功能

1.时区设置

选择分析的计算时区,默认选择项目时区。可在「项目管理-分析配置-时区设置」中设置是否开启多时区功能。

2.选择分析主体

分析主体:默认为海纳ID,若以设备ID作为分析主体,则将不同设备ID的用户视作不同用户;若以账号ID作为分析主体,则将不同账号ID的用户视作不同用户。若当前下拉框中没有想要的分析主体,可至「项目管理-分析配置-分析主体」中进行相关配置。

3. 近似计算

默认开启「近似计算」,开启后将近似计算触发用户数、人均次数、人均值和去重数,极大地提高查询性能,同时将计算结果的标准误差控制在 0.3%以内。

4. 保存

分布分析的结果可以保存为书签或同步至数据空间,方便后续进行查询和修改。书签和数据空间的操作详情可至「书签」和 「数据空间」中查看。

- 保存为书签,保存后可在书签列表查看该事件分析结果。
- 同步至数据空间的操作,同步后可在数据空间处查看该事件分析结果。

5. 刷新

可点击「刷新」图标对页面数据进行实时更新。

6. 下载

当查询数据数过多时,查询结果最多只展示前1000条记录,若想查看完整的数据,可点击「下载」获取完整数据(最多下载 100万条),下载的文件格式为csv。

5 实例

5.1 查看用户浏览频次的差异

如下图所示,可以使用分布分析从行为日期查看小程序启动次数的频次差异,还可按照性别、渠道等各种分组进行查看

路径分析

帮助中心 > 功能手册 > 数据分析 > 路径分析

1 概述

路径分析主要用于分析用户对某些事件的行为顺序、行为偏好、关键节点以及转化效率。可以直观掌握用户行为扩展路线,以供优化节点内容、提升整体转化效率。利用路径分析模型,可以快速了解影响转化的主次因素,从而有目的的改进产品。

路径分析可以解决以下示例问题: 1.分析用户进入产品后的行为路径分布,或用户在退出前的行为路径分布; 2.分析特定流程(注册、登录、付款等)的转化过程,寻找并发现提升流程转化的改进点 3.分析用户在产品模块之间的流动情况

2 位置和使用权限

操作位置:数据分析-路径分析

使用权限: 不同的项目角色操作权限如下

项目角色	权限	权限明细
管理员	查看、 功能使用	查看sql代码、使用分析并对分析结果进行保存, 对分析数据进行下载,查看事件列表/用户列表
分析师	功能使用	使用分析并对分析结果进行保存,对分析数据进行下载, 查看事件列表/用户列表
普通用户	无权限	无权限

3页面概览

路径分析包括查询设置、分析结果展示与基本功能三部分。

- 查询设置包括选择参与分析的事件、设置分析路径、全局筛选以及设置会话间隔;
- 分析结果展示包括时间组件、图形展示;
- 基本功能包括时区选择、保存功能和刷新功能。

4 功能结构

4.1 查询设置

1. 参与分析的事件

(1)选择事件:点击事件选择框,进行分析事件的选择,所选项为参与路径分析的事件范围,统称为分析事件组,所选事件不 超过50。

- (2) 添加分组:可为选择的"参与分析的事件"组,添加分组
- (3) 添加筛选条件:筛选条件内容来源于所选事件的事件属性

2. 设置分析路径

选择分析事件组中的1项作为分析基础,可指定它为起始或结束事件进行分析,同时可对该事件进行筛选,筛选条件内容来源于所选事件的事件属性。

3. 全局筛选

可对事件添加筛选条件,内容来源于事件属性、用户属性、用户标签、用户分群,根据不同数据类型的筛选项,筛选规则也有所不同。

1) 若想要添加多条筛选条件:点击「添加并列条件」可添加二级筛选条件,最多只能添加到二级,筛选条件添加条数不限。

2) 添加新的条件需要设定和原条件的关联关系,点击"且/或"可切换关联关系。

• 不同数据类型的筛选项支持的条件筛选规则汇总如下:

筛选项的数据类型	条件筛选可选项
文本型	等于、不等于、包括、不包括、有值、无值、正则匹配、正则不匹配
数值型	等于、不等于、小于、小于等于、大于、大于等于、有值、无值、 区间
时间型	位于区间、小于、小于等于、大于、大于等于、相对当前日期 (区间)、相对当前日期(过去)、有值、无值、等于、不等于
布尔型	为真、为假、有值、无值

4. 设置会话间隔

可设置相邻事件的最大时间间隔,若两连续的有效事件(属于分析事件组)间的触发间隔在设置时间内时,视为两者属于同一 会话。

4.2 分析结果展示

1. 时间组件

时间范围:默认选中动态时间「过去7天」,可点击选择其他静态/动态时间。

2. 图形展示

分析结果以桑基图形式展现,以目标事件为起点/终点,详细查看后续/前置路径,可以详细查看某个节点事件的流向,同时 支持一键导出「桑基图分析结果图片」。

3. 查看用户列表

通过点击数据表中的数据,可链接跳转至用户列表页面,查看已选人群的更多用户属性信息:

- 用户列表支持查看用户其他属性特征,了解用户更多信息
- 用户列表支持刷新数据、导出数据功能
- 用户列表支持将已选人群保存为结果分群,可至分群管理列表中查看

4.3 基本功能

1.时区设置

选择分析的计算时区,默认选择项目时区。可在「项目管理-分析配置-时区设置」中设置是否开启多时区功能。

2. 保存

路径分析的结果可以保存为书签或同步至数据空间,方便后续进行查询和修改。书签和数据空间的操作详情可至「书签」和 「数据空间」中查看。

- 保存为书签,保存后可在书签列表查看该事件分析结果。
- 同步至数据空间的操作,同步后可在数据空间处查看该事件分析结果。

3. 刷新 可点击「刷新」图标对页面数据进行实时更新。

5 实例

5.1 查看用户进入产品后的行为路径分布情况

如下图所示,可以使用路径分析很直观地查看用户进入产品后的行为路径分布情况,查看用户在不同路径的流失情况,帮助我们提示核心模块的到达率,进行流失用户的去向分析。

属性分析

帮助中心 > 功能手册 > 数据分析 > 属性分析

1 概述

属性分析是通过对用户各类特征进行标示,从而了解用户的属性、或者属性交叉的分布结构,进而可以对用户进行分层标记, 以便进行后续的产品、运营动作。属性分析主要用于分析用户属性的统计与分布情况,可以同时查看不同分组值用户的统计数 与分布情况。快速掌握特定用户群的用户画像,为精细化运营提供帮助。

属性分析可以解决以下示例问题: 1.查看用户在不同省份的具体分布情况 2.对比不同渠道用户的累计付费金额人均值 3.按用户当前等级和 VIP 等级拆分查看用户数量

2 位置和使用权限

操作位置:数据分析-属性分析 使用权限:不同的项目角色操作权限如下

项目角色	权限	权限明细
管理员	编辑、保存、 下载	拥有编辑权限、保存为书签、添加到数据空间、下载数据
分析师	编辑、保存、 下载	拥有编辑权限、保存为书签、添加到数据空间、下载数据
普通用户	无权限	无权限

3页面概览

属性分析包括查询设置、分析结果展示与基本功能三部分。

- 查询设置包括选择分析属性、添加筛选条件和添加分组项/进行人群对比;
- 分析结果展示包括图形展示和表格展示;
- 基本功能包括时间显示、保存、刷新和下载。

4 功能结构

4.1 查询设置

1. 分析属性

选择「用户数」或「用户属性/标签」+「聚合算子」作为监控指标

• 不同数据类型的用户属性/标签支持的聚合算子计算规则汇总如下:

用户属性/标签 (数据类型)	聚合算子	
数值型	总和、均值、中位数、最小值、最大值、去重数	
布尔型	为真数、为假数、为空数、不为空数、去重数	
其他类型	去重数	

2. 筛选条件

支持对用户属性、用户标签、用户分群进行筛选,根据不同数据类型的筛选项,筛选规则也有所不同

1) 若想要添加多条筛选条件:点击「添加并列条件」可添加二级筛选条件,最多只能添加到二级,筛选条件添加条数不限。

2) 添加新的条件需要设定和原条件的关联关系,点击"且/或"可切换关联关系。

• 不同数据类型的筛选项支持的条件筛选规则汇总如下:

筛选项的数据类型	条件筛选可选项		
文本型	等于、不等于、包括、不包括、有值、无值、正则匹配、正则不匹配		
数值型	等于、不等于、小于、小于等于、大于、大于等于、有值、无值、 区间		
时间型	位于区间、小于、小于等于、大于、大于等于、相对当前日期 (区间) 、相对当前日期(过去) 、有值、无值、等于、不等于		
布尔型	为真、为假、有值、无值		

3. 分组项

支持将用户属性、用户标签、用户分群设置为分组项,根据不同数据类型的分组项,有不同的分组设定方式

- 针对数值型分组项,分组方式包括默认区间、离散数字和自定义区间三种
- 针对时间型分组项,分组方式包括汇总与不汇总两种,其中汇总可选择按月、按周、按天、按小时、按分钟
- 针对其他类型的分组项,将直接按照离散方式展示

4. 人群对比

点击「人群对比」,可对不同人群添加筛选条件,对比不同条件下各人群的分析属性结果

4.2 分析结果展示

1. 图形展示

- 1) 当无分组项和对比人群时, 仅展示柱状图, 只分析全体用户的情况
- 2) 当有一个分组项时,展示柱状图与饼图,且柱状图可根据分组项选择具体所展示的分组值
- 3) 当有多个分组项时,展示柱状图与饼图,且柱状图可选择横坐标展示项,并可对其他一分组项选择具体所展示的分组值
- 4) 当选择人群对比时,展示柱状图与饼图,且柱状图可选择某一个或多个人群

2. 表格展示

- 表格主要展示分组/对比人群的分析属性值。
- 1) 当无分组项和对比人群时,表格展示内容为总体的分析属性指标值
- 2) 当有一个分组项时,表格展示内容为各分组值的分析属性指标值
- 3) 当有多个分组项时,表格展示内容为各分组排列组合的分析属性指标值
- 4) 当选择人群对比时,表格展示内容为各人群的分析属性指标值

注:当分析属性为用户数时,可通过点击表格中具体的数值,链接跳转至用户列表页面,查看已选人群的更多用户属性信息:

- 用户列表支持查看用户其他属性特征,了解用户更多信息。
- 用户列表支持导出数据
- 用户列表支持将已选人群保存为结果分群,可至分群管理列表中查看。

4.3 基本功能

1. 保存

属性分析的结果可以保存为书签或同步至空间,方便后续进行查询和修改。书签或空间的操作详情可至「数据空间」和「书 签」查看。

- 保存为书签,保存后可在书签列表查看该属性分析结果。
- 同步至空间的操作,同步后可在数据空间处查看该属性分析结果。

2. 刷新

可点击「刷新」图标对页面数据进行实时更新

3. 下载

当查询数据数过多时,查询结果最多只展示前1000条记录,若想查看完整的数据,可点击「下载」获取完整数据(最多下载 100万条),下载的文件格式为csv。

5 实例

5.1 按省份查看用户数

如下图所示,可以使用属性分析很直观地查看用户在不同省份的具体分布情况。也可以按城市进行分析,有助于快速了解产品 用户具体分布在哪些城市,是否为发达地区等,以判断不同区域用户的喜好程度等。从图中数据看,用户主要分布在安徽省和 内蒙古。

网页热力分析

帮助中心 > 功能手册 > 数据分析 > 网页热力分析

1 概述

网页热力分析是应用特殊高亮的颜色显示页面、页面组区域中不同元素的点击、触达特征。网页热力分析主要用来分析用户在网页上的点击、触达深度等情况,基于用户在页面上的点击、浏览行为可以非常直观地通过热力图将大量用户的行为可视化。

网页热力分析可以解决以下示例问题:

- 1. 查看不同人群的页面浏览情况
- 2. 查看不同网页的跳转点击情况
- 3. 查看页面元素的设置和用户偏好是否一致

2 位置和使用权限

操作位置:数据分析-网页热力分析 使用权限:不同的项目角色操作权限如下

项目角色	权限	权限明细
管理员	编辑、保存、下载	拥有编辑权限、下载数据
分析师	编辑、保存、下载	拥有编辑权限、下载数据
普通用户	无权限	无权限

3页面概览

网页热力分析包括查询设置、查询结果展示与基本功能三部分。

- 查询设置包括选择显示内容,进行事件满足和用户符合筛选;
- 查询结果展示包括列表展示,点击页面/页面组名称可查看对应的点击热图/浏览热图;
- 基本功能包括时区显示、刷新、下载和查看SQL代码。

4 功能结构

4.1 查询设置

1. 显示内容

```
可选择原始页面或页面组进行查看。原始页面是用来分析单个页面的点击情况,而页面组则用来分析一系列界面结构相似的网
页整体的浏览和点击情况(例如电商行业的商品详情页,可以整体作为一个页面组来进行分析)。
```

2. 事件满足

支持对事件属性进行筛选,根据不同数据类型的筛选项,筛选规则也有所不同

1) 若想要添加多条筛选条件:点击「添加并列条件」可添加二级筛选条件,最多只能添加到二级,筛选条件添加条数不限。

- 2) 添加新的条件需要设定和原条件的关联关系,点击"且/或"可切换关联关系。
- 不同数据类型的筛选项支持的条件筛选规则汇总如下:

筛选项的数据类型	条件筛选可选项
文本型	等于、不等于、包括、不包括、有值、无值、正则匹配、正则不匹配
数值型	等于、不等于、小于、小于等于、大于、大于等于、有值、无值、 区间
时间型	位于区间、小于、小于等于、大于、大于等于、相对当前日期 (区间) 、相对当前日期(过去) 、有值、无值、等于、不等于
布尔型	为真、为假、有值、无值

3. 用户符合

支持对用户属性、用户标签、用户分群进行筛选,根据不同数据类型的筛选项,筛选规则也有所不同,筛选规则与事件满足一致

4.2 分析结果展示

1. 原始页面

- 页面名称:我方采集到的页面标题及页面地址
- 页面浏览用户数:这个页面的浏览 UV,也即事件 Web浏览页面 的用户数
- 页面浏览总次数:这个页面的浏览 PV,也即事件 Web浏览页面 的总次数
- 页面点击用户数:这个页面交互元素的点击 UV,也即事件 Web页面点击 的用户数
- 页面点击总次数:这个页面交互元素的点击 PV,也即事件 Web页面点击 的总次数

2. 页面组

- 分组名称:用户设置的页面组名称与底图地址
- 页面数:用户设置的页面组所包含的页面数
- 页面浏览用户数:这个页面组的浏览 UV,也即事件 Web浏览页面 的用户数
- 页面浏览总次数:这个页面组的浏览 PV,也即事件 Web浏览页面 的总次数
- 页面点击用户数:这个页面组交互元素的点击 UV,也即事件 Web页面点击 的用户数
- 页面点击总次数:这个页面组交互元素的点击 PV,也即事件 Web页面点击 的总次数

3. 点击热图

点击页面/页面组名称,跳转到页面详情,可查看页面的点击热图,下面以官网为例展示点击热图的相关内容

- 1. 表示被点击过的交互元素,这里显示的 3.10% 是点击率。鼠标悬浮上这些被点击过的按钮上时,会显示点击详情信息框。 而如果鼠标点击这个交互元素,则会按照原页面的逻辑进行相应的交互;
- 2. 当前元素内容:指这个交互元素里面的文本内容;
- 3. 点击次数:指这个交互元素点击了多少次;
- 4. 点击人数:指这个交互元素被多少人点击;
- 5. 点击率:点击率=这个元素的点击次数/整个页面的浏览次数(PV);

- 6. 点击占比:点击占比=这个元素点击次数/整个页面内所有可见元素的总点击次数;
- 7. 历史内容,表示取这个按钮历史最常出现的值。例如如果是新闻的话,头条位置的新闻每天是不同的;
- 8. 查看用户列表 点击「查看用户列表」,查看点击过这个按钮的具体用户是哪些人;
- 用户列表支持查看用户其他属性特征,了解用户更多信息。
- 用户列表支持导出数据
- 用户列表支持将已选人群保存为结果分群,可至分群管理列表中查看。

3. 浏览热图

点击页面/页面组名称,跳转到页面详情,可查看页面的浏览热图,浏览热图可显示用户的触达率情况,不同的颜色代表不同的 用户到该位置的用户占比,你可以通过鼠标滑动动态查看每个位置的浏览占比

4.3 基本功能

1.时区设置

选择分析的计算时区,默认选择项目时区。可在「项目管理-分析配置-时区设置」中设置是否开启多时区功能。

2. 刷新

可点击「刷新」图标对页面数据进行实时更新

3. 下载

当查询数据数过多时,查询结果最多只展示前1000条记录,若想查看完整的数据,可点击「下载」获取完整数据(最多下载 100万条),下载的文件格式为csv。

5 实例

5.1 查看页面用户的点击情况

可以查看用户对官网的体验demo元素的点击情况,看浏览的用户体验系统的意愿如何

注意: 在使用网页热力分析时,若您所需分析的网页涉及到登陆授权等情况,您需另开一个页面登陆授权后即可查看登陆 后页面的点击浏览情况;若您想分析登陆前的页面情况需退出登陆后点击页面查看。

SQL查询

帮助中心 > 功能手册 > 数据分析 > SQL查询

1 概述

当已有的分析模型不满足分析需求时,可以通过SQL查询功能进行自定义查询,SQL查询的结果可以保存为空间/书签查看。

2 位置和使用权限

操作位置:数据分析-SQL查询

使用权限: 不同的项目角色操作权限如下

项目角色	权限	权限明细
管理员	查看、 功能使 用	查看表结构、数据概览、运行SQL语句并对结果进行保存、 下载、保存为书签、保存为视图、添加图表等
分析师	查看、 功能使 用	查看表结构、数据概览、运行SQL语句并对结果进行保存、 下载、保存为书签、保存为视图、添加图表等
普通用户	无权限	无权限

3页面概览

SQL查询页面主要分为数据显示区、语句编写区和结果展示区三个部分。其中结果展示区分为表结构、数据概览、查询结果、查询历史四个功能页面。

语句编写框支持格式化操作,查询结果可以进行下载、保存为书签、保存为视图、添加图表的操作。

4 功能结构

4.1 查看数据表

SQL查询页面左侧会展示项目已有的数据表,包括事件表、属性表、标签分群表、维度表等数据。查询中主要使用到的数据表的 命名规则对应如下:

数据表	表含义	备注

dwd_event_项目名	事件表	存储元事件和事件属性数据
dwd_users_项目名	用户表	存储最新的用户属性数据
ads_feature_项目名	标签表	存储标签的最新版本数据
ads_history_feature_项目名	标签历史表	存储标签的回溯/备份版本数据
dwd_history_users_项目名	用户备份表	存储备份的用户属性数据
dim_event_事件属性名_项目名	维度表	某事件属性的维度表
view_test_项目名	视图	用户创建保存的视图

1. 对于数据表的基本操作:

- 可点击表名右侧的按钮复制表名。
- 对于用户创建的视图,用户可进行删除操作。
- 2. 点击数据表,可以查看数据表的表结构或预览前10行数据。
- 查看数据的字段名、字段类型、字段显示名、是否允许为空等基础信息。
- 预览数据前10行

4.2 语句查询

- 1. 支持查询部分语句:默认为全部执行,若只需要执行部分语句,请选中语句后点击运行。
- 2. 支持一次查询多条语句:多个查询语句使用分号间隔,结果将在结果1、结果2···处分别进行展示,鼠标悬浮至对应tab,会展示其对应执行语句。
- 3. 支持注释功能,单行注释使用-,多行注释使用/**/

4.3 添加图表

点击添加图表后,可进入SQL可视化模块,具体操作详情请至SQL可视化查看。

4.4 保存为书签

SQL查询的结果可以保存为书签或同步至空间,方便后续进行查询和修改。书签或空间的操作详情可至「数据空间」和「书签」 查看。

• 保存为书签,保存后可在书签管理列表查看该条数据。

4.5 保存为视图

SQL查询的结果可以保存为视图,在查询中可以基于视图进行其他查询。

• 如果需要删除视图,可以在数据表概览中选中视图,点击「删除视图」按钮完成删除

4.5 查询历史

查询历史将记录用户最近100条查询记录,点击按钮「复制代码」和「运行」可以快速操作,方便用户的编写和修改。

• 在查询历史页下,只能查看自己在项目中的SQL查询历史,无法查看他人的查询历史。每个用户只会记录近30天的查询记录,到期将清除查询记录。

4.6 数据下载

当查询数据数过多时,查询结果最多只展示前1000条记录,若想查看完整的数据,可点击「下载」获取完整数据(最多下载 100万条),下载的文件格式为csv。

5 实例

5.1 查看用户在某天的具体行为

SELECT * FROM dwd_event_1 WHERE user_id = 'aabbcc' AND event_time = '2022-09-22' LIMIT 1000;

5.2 查询发生过A行为但是没有发生过B行为的用户数

```
SELECT a.user_id FROM (
    SELECT DISTINCT user_id FROM dwd_event_1 WHERE event_time='2022-09-22' AND event = 'app_start'
) a
LEFT OUTER JOIN (
SELECT DISTINCT user_id FROM dwd_event_1 WHERE event_time='2022-09-22' AND event = 'app_login'
) b
ON a.user_id = b.user_id
WHERE b.user_id IS NULL
```

5.3 查询活动期间用户的付费次数分布

```
SELECT
CASE
WHEN pay_count < 5 THEN '<5'
WHEN pay_count < 10 THEN '<10'
WHEN pay_count < 15 THEN '<15'
ELSE '>=15'
END,
COUNT(*)
FROM (
SELECT user_id, COUNT(*) AS pay_count FROM dwd_event_1
```

```
WHERE event_time BETWEEN '2022-08-01' AND '202-09-01' AND event = 'pay'
GROUP BY 1
)a
GROUP BY 1
```

SQL可视化

帮助中心 > 功能手册 > 数据分析 > SQL可视化

1 概述

对于SQL查询实现的高级分析,可视化模块支持将查询的表数据转换为图表的形式进行展现。可视化模块支持以拖拽的方式配置 图表字段,并支持灵活配置图表样式,以更直观的方式展现数据变化趋势。可视化模块仅适用于 SQL 查询。

2 位置和使用权限

操作位置:数据分析-SQL查询-添加图表

使用权限: 不同的项目角色操作权限如下

项目角色	权限	权限明细
管理员	查看、功能使用	添加图表、编辑图表、将图表保存至空间/书签
分析师	查看、功能使用	添加图表、编辑图表、将图表保存至空间/书签
普通用户	无权限	无权限

3页面概览

3.1 添加数据图表

若想对查询结果添加可视化的数据图表,则点击查询结果处的「添加图表」:

点击后将进入图表设置页面:

- 若有添加图表,则保存或更新书签时将同时保存或更新结果数据和可视化图表的配置,并默认展示图;当没有添加图表时,默认展示原数据表。
- 可视化数据来源:可视化图表的数据源为 SQL 查询结果中前1000 条数据。

3.2 配置数据图表

可视化配置页面分为图表查看区、图表配置区、基础操作区三个部分。可通过拖拽的方式将字段拖动至对应展示位置,绘制图 表:

• 图表查看区:展示图表最终呈现的效果,随配置区的操作即时响应。

- 图表配置区:下设两个标签页,「字段」处展示SQL的查询结果的字段,支持对字段进行重命名。「可视化」处用于对图表 内容和样式进行设置。
- 基础操作区:返回 SQL 页面,保存/移除图表。

3.3 保存图表

点击右下角的「保存」按钮,可保存配置好的图表信息

保存后将回到SQL编辑页面,可点击以下图标切换图表展示方式:

3.4 编辑已有图表

可点击编辑按钮再次进入图表编辑页面,或点击删除按钮删除已有的图表配置:

3.5 将图表保存至空间

即可在空间查看配置的SQL可视化图表:

4 功能结构

4.1 数据配置

可视化模块提供簇状柱形图、堆积柱形图、折线图、堆积折线图、折线图和堆积柱形图(组合图)、折线图和簇状柱形图(组 合图)、饼图、环形图、一维表、二维表共10种图表展现方式,每种图表其配置的内容略有不同。

- X轴/分组:通常由非数值型字段担任,用来对数据进行分类展示。允许拖入多个字段,多个字段在图表中将合并成一个分 组项。当选择数值型字段时,将数值型字段按照离散值进行分组。如拖动省份和城市两个字段,其展示为「省份.城市」
- Y轴/指标:通常由数值型字段担任,展示指标值。允许拖入多个字段。若为数值型字段,支持的计算方式为:总和、均 值、中位数、最小值、最大值、去重数、计数、标准差;若为非数值型字段,支持的计算方式为:去重数、计数。
- 图例:通常由非数值型字段担任,用来对数据进行分类展示。只允许拖入一个字段。当选择数值型字段时,将数值型字段 按照离散值进行分组。
- 次Y轴/指标: 折线图和组合图可设置次Y轴, 其设置规则和Y轴/指标一致。
- 行: 仅二维表可设置行。允许拖入多个字段。
- 列:仅一维表和二维表可设置列。一维表允许拖入多个字段,二维表仅能拖入一个字段。

4.2 样式配置

- 可对图表进行图标题、X轴标题、Y轴标题、Y轴坐标范围、图例、网格线、显示数值等的配置。
- 一维表和二维表可设置是否对行/列进行阶段合计。

4.2 其他配置

- 在图表配置过程中若切换图表,将尽可能保留上一个图表字段和样式的配置;若无法匹配,则会清除所有已有条件,需重新配置。如簇状柱形图、堆积柱形图、折线图、堆积折线图、折线图和堆积柱形图(组合图)、折线图和簇状柱形图(组合图)可自由切换;饼图和环形图可自由切换。
- 2. 当想要排序时,可选择所使用字段中的任意一个字段进行升/降序的排序。
- 3. 对于X轴/图例这类分组字段,若勾选显示无数据的项目,则会将空值作为一类并统计其数据指标,若不勾选,则不对空值 进行聚合计算。

5 实例

5.1 复杂分析计算的可视化展示

当已有的分析模型不能满足需求时,可使用SQL查询来灵活获取想要的数据结果,但表格的展现形式不够直观,可选择合适的图表来合理展现数据。

5.2 双轴图

若您想要将两个指标放在一个图上进行展示,当数据的量级(如用户数和次数)或单位不同(如绝对值和百分比)时,若放在同一个主轴上进行比较,则会拉平数据的趋势不易于观察。因此可使用折线图或组合图的主次双轴来刻画数据,或设置不同的图形展示方式(如柱形图和折线图),以达到更直观的展示效果。

用户洞察

用户标签

帮助中心 > 功能手册 > 用户洞察 > 用户标签

1. 概述

用户标签是对某一类特定群体或对象的某项特征进行的抽象分类和概括,其值(标签值)具备可分类性。用户标签模块支持用 户基于埋点数据自定义标签规则生成新的标签,描述用户的行为特征。用户标签可以单独管理分析,也可以在分析模型中细分 分析,同时可添加至空间进行日常的数据监控。

2. 位置和使用权限

功能位置:用户洞察-用户标签

使用权限:不同的项目角色操作权限如下

项目角色	权限	权限明细
管理员	新增标签、 管理所有标签	新增标签、所有标签(修改、复制、重新计算、删除、 回溯)
分析师	新增标签、 管理自建标签	新增标签、自建标签(修改、复制、重新计算、删除、 回溯)、他人创建的标签(复制)
普通用户	无权限	无权限

3. 页面概览

用户可以使用平台提供的标签创建方式创建新的标签,并可在主页对已有标签进行管理,查看标签的数据详情,监控用户的行为变化。

3.1 创建新的标签

若用户想要创建一个新的标签,需要完成以下的几个步骤:

第一步:点击标签管理主页的「新增」创建标签,选择好想要使用的标签创建方式后,点击进入标签配置页面。

目前平台提供以下几种标签创建方式,各种创建方式的功能说明如下:

- 数值标签:对用户的行为事件进行聚合计算,输出数值型标签。如统计用户近一年内登录的总次数。
- 自定义标签:根据若干条件筛选出满足条件的用户,自定义其标签值。如将近一年充值笔数小于10的人群定义为低价值人群,大于等于10的定义为高价值人群。
- 初末次标签:将用户初次/末次完成某件事件的日期/距今天数/属性值作为标签值。如统计用户首次登录距今天数。
- SQL标签:使用自定义的SQL查询语句为指定的用户进行标记。
- 导入标签:使用导入的标签值为指定的用户进行标记。

第二步:进行标签规则的配置,标签的配置包括三个部分,分别是标签基础信息、标签规则配置、更新与备份。

1. 标签基础信息:填写标签名、标签显示名、标签备注等信息。

2. 标签规则配置

- 该标签的分析主体:即标签group by统计的对象,默认为海纳ID。若以设备ID作为分析主体,则将不同设备ID的用户视作不同用户;若以账号ID作为分析主体,则将不同账号ID的用户视作不同用户。若当前下拉框中没有想要的分析主体,可至「项目管理-分析配置-分析主体」中进行相关配置。标签创建后,分析主体不可更改。
- 时间范围:默认选中动态时间「过去7天」,可点击选择其他静态/动态时间。
- 规则定义:配置标签的计算规则。可对事件添加筛选条件,可点击「属性筛选」添加筛选条件来筛选满足条件的用户,下 拉框可以选择所选事件对应的事件属性/用户属性。

(1)若想要添加多条筛选条件:点击「属性筛选」可添加一级筛选条件,点击「添加并列条件」可添加二级筛选条件,最多只 能添加到二级,筛选条件添加条数不限。

(2) 添加新的条件需要设定和原条件的关联关系,点击"且/或"可切换关联关系。

3. 更新与备份

- 计算时区:选择标签的计算时区,默认选择项目时区。可在「项目管理-分析配置-时区设置」中设置是否开启多时区功能。
- 更新方式:分为定期更新和手动更新两种。定期更新即系统将在指定时间定期计算并更新标签数据,并以前一日作为计算基准日期;手动更新数据不会自动进行计算更新,用户可手动点击「更新数据」进行计算。若选择定期更新,需要进行「时间设置」和「备份方式」的设置
- 时间设置:设置标签更新的具体时间。
- 备份方式:若自动备份,将自动存储定期更新的标签数据,可至「回溯版本管理」中查看该标签的历史数据;否则仅保留 最新计算的标签数据,以往的标签数据将被覆盖。

第三步:标签基础信息、标签规则配置、更新与备份均设定好以后,点击「保存并计算」按钮即可创建标签,并同步至标签管 理主页。

3.2 管理已有标签

创建好的标签将同步至标签管理主页,可进行编辑、复制、重新计算、删除、查看详情等操作:

- 点击「编辑」可修改标签配置
- 点击「复制」可复制标签配置生成一个新的标签
- 点击「重新计算」会计算标签数据
- 点击「删除」可删除该标签
- 点击「标签名称」可跳转至标签详情页面

3.3 查看标签详情

标签详情页主要包括展示标签基础信息、数据可视化、回溯版本管理三个部分。

• 在页面左侧,可查看标签的配置信息、变更记录等信息。

- 提供折线图、数值堆积图、百分比堆积图、柱形图、饼图五种展示方式
- 可查看标签的数据详情,并可查看标签的用户列表。
- 对标签的回溯版本进行管理。

4. 功能结构

4.1 不同创建方式功能说明

4.1.1 数值标签

数值标签用于对用户的行为事件进行简单的聚合计算,将计算的数值结果作为标签值。如统计用户过去7天登录的总次数。

可计算内容包括:

- 某个指定事件的完成次数/天数/小时数
- 某个指定事件数值类型属性的总和/均值/中位数/最大值/最小值/去重数
- 某个指定事件布尔类型属性的为真数/为假数/为空数/不为空数/去重数
- 某个指定事件其他类型属性的去重数

计算格式:「事件」+「事件属性」+「计算算子」或「事件」+「计算算子」

- 1. 选择事件:可以选择任意一个元事件/虚拟事件。若分析主体为用户自定义的分析主体且来源于事件属性,则下拉框中可选的元事件只展示包含该事件属性(分析主体)的元事件。
- 2. 选择聚合算子:聚合算子根据事件属性的数据类型有所不同

3. 可对事件添加筛选条件。

4.1.2 自定义标签

自定义标签根据若干条件筛选出满足条件的用户,自定义其标签值。如将近一年充值笔数小于10次的人群定义为低价值人群, 大于等于10次的定义为高价值人群。定义的标签可作为分组项/筛选项用于数据分析中,如分低价值人群和高价值人群统计用户 的购买行为/统计低价值人群的购买行为。

可计算内容包括:

- 用户属性筛选:
 - (1) 某文本型用户属性等于/不等于/包括/不包括…
 - (2) 某数值型用户属性等于/不等于/包括/不包括…
 - (3) 某时间型用户属性位于区间/小于/大于/相对当前日期…
 - (4) 某布尔型用户属性为真/为假/有值…
- 用户行为筛选:
 - (1) 用户完成某个指定事件的次数/天数/小时数 等于/大于/小于…
 - (2) 用户完成某个指定事件数值类型属性的总和/均值/中位数/最大值/最小值/去重数 等于/大于/小于…
 - (3) 用户完成某个指定事件布尔类型属性的为真数/为假数/为空数/不为空数/去重数 等于/大于/小于…
 - (4) 用户完成某个指定事件其他类型属性的去重数 等于/大于/小于…
 - (5) 用户没做过某个事件
计算格式:

- 用户属性筛选:「选择用户属性」+「选择算子」+「某值」
- 用户行为筛选:「时间」+「做过」+「某事件」+「算子/事件属性.算子」+「条件判断」+「某值」
 或「时间」+「没做过」+「某事件」

可对满足筛选条件的用户定义标签值。如将过去7天签到的次数大于等于5次的用户该标签的标签值为高活跃用户,大于等于3且 小于5次的用户该标签的标签值为中活跃用户,大于等于1且小于3的用户该标签的标签值为低活跃用户,等于0次的用户该标签 的标签值为不活跃用户。

高活跃用户配置页面如下:

可增加面板配置标签值为中活跃用户、低活跃用户、不活跃用户的规则:

注1:若分析主体为用户自定义的分析主体且来源于事件属性,则用户行为筛选的元事件选择下拉框中可选的元事件只展示包含 该事件属性(分析主体)的元事件。 注2:用户(分析主体)将优先匹配至满足条件且顺序靠前的标签值中。

4.1.3 初末次标签

初末次标签将用户初次/末次完成某件事件的日期/距今天数/属性值作为标签值。如统计用户首次登录距今天数。

可计算内容包括:

- 某个指定事件初次/末次发生的具体日期/距今天数
- 某个指定事件初次/末次发生对应的某事件属性值

计算格式:「事件」+「初次发生/末次发生」+「具体时间点/距离当天日期天数/事件属性值」

- 1. 选择事件:可以选择任意一个元事件/虚拟事件。若分析主体为用户自定义的分析主体且来源于事件属性,则下拉框中可选的元事件只展示包含该事件属性(分析主体)的元事件。
- 2. 选择初次发生/末次发生。
- 3. 选择分析属性:提供具体时间点、距离当天日期天数、事件对应的事件属性供选择。
- 4. 可对事件添加筛选条件。

4.1.4 SQL标签

当已有的创建方式不能满足需求时,可以使用自定义的SQL查询语句为指定的用户进行标记。

- 1. 查询结果需为两列。
- 2. 第一列为不重复的分析主体ID,需与下拉框中选择的标签的分析主体保持一致。
- 3. 第二列为对应的标签值,需与下拉框中选择的标签数据类型保持一致。
- 4. SQL标签支持定期更新,若想统计动态时间的标签结果,请使用动态时间窗的语句。如统计最近30天,则为WHERE part_date > DATE_SUB (CURDATE (), INTERVAL 30 DAY)。
- 5. 若想预览某个表的数据结构和内容,可以点击左侧的数据表,在「表结构」和「数据概览」中进行查看,如图所示:

4.1.5 导入标签

导入标签支持上传本地文件,对用户赋予标签值,可以选择某一用户属性作为用户的关联字段。

- 1. 上传文件共两列, 第一列为身份识别信息, 第二列为标签值。
- 2. 当选择的分析主体是预置分析主体海纳ID时,需选择关联属性(选择第一列的字段名)。若您上传的文件的第一列为省份,则在关联属性处选择省份,系统会将所有省份为上海的海纳ID赋予上海对应的标签值,省份为北京的海纳ID赋予北京对应的标签值。
- 3. 上传文件第二列的数据类型需与下拉框中所选的标签数据类型保持一致。
- 4. 导入标签不支持更新与备份。

4.2 聚合和筛选算子说明

• 不同数据类型的事件属性支持的聚合计算规则汇总如下:

事件	事件属性 (数据类型)	聚合算子
任意事件		次数、天数、小时数
元事件	事件属性 (数值型)	总和、均值、中位数、最小值、最大值、去重数
元事件	事件属性 (布尔型)	为真数、为假数、为空数、不为空数、去重数
元事件	事件属性 (其他类型)	去重数

• 不同数据类型的事件属性支持的条件筛选规则汇总如下:

属性的数据类型	条件筛选可选项
文本型	等于、不等于、包括、不包括、有值、无值、正则匹配、正则不匹配
数值型	等于、不等于、小于、小于等于、大于、大于等于、有值、无值、区间
时间型	位于区间、小于、小于等于、大于、大于等于、相对当前日期 (区间) 、相对当前日期(过去) 、有值、无值、等于、不等于
布尔型	为真、为假、有值、无值

4.3 数据可视化

在数据可视化部分,支持时间筛选、自定义区间和分组、切换展示图(折线图、数值堆积图、百分比堆积图、柱形图、饼图)、数据排列方式、数据导出等操作,可灵活设置数据的展示方式。

- 折线图
- 数值堆积图
- 百分比堆积图

- 柱形图
- 併图

4.4 查看标签数据

数据表中会展示标签最新版本和回溯版本中的所有数据,最近一次手动/定期更新的数据将在图表中用红点标注,并以今天的日 期展示该条标签数据。

- 数据表支持以分组为列/以日期为列展示
- 数据表展示每个回溯日期标签值的覆盖用户数及占比
- 点击具体的数值可以查看用户列表

4.5 查看用户列表

通过点击数据详情页面数据表中的具体数值,可链接跳转至用户列表页面,查看已选人群的更多用户属性信息:

- 用户列表支持查看用户其他属性特征,了解用户更多信息。
- 用户列表支持导出数据
- 用户列表支持将已选人群保存为结果分群,可至分群管理列表中查看。

4.6 回溯版本管理

回溯版本管理用于存储定期更新的标签每日计算的数据,并支持标签的手动回溯。

回溯版本的意义:用户的行为会随着时间发生变化,当标签计算的基准时间和事件发生的时间相差较远时,标签的实效性较差,若使用此标签来做分析可能会影响分析结果。因此标签的回溯版本旨在尽可能还原事件发生当时的情况,助力精准分析。 此外,若对标签的回溯版本直接进行分析,也可以从图表上监控到用户行为在一段时间上发生的变化。

• 若标签开启了定期备份功能,则标签在定期计算时会将每日计算的标签数据存储至该标签的回溯版本管理中。

- 若想要某天标签的数据值但定期备份中没有该天的数据,可以通过手动回溯获得标签值。手动回溯主要用于补充缺失日期的标签版本、重算计算失败的标签、重算逻辑更新的标签。
- 在回溯版本管理中每天最多只有一条标签数据,即最近一次计算的数据。
- 1. 手动回溯:用户可点击「数据回溯」按钮回溯标签数据。以所选的回溯日期作为计算基准日期,若回溯日期共选择30天, 则会产生回溯标签共30个。
- 支持批量回溯,在单次操作中回溯日期范围最多可选择90天。
 说明:若某标签统计的是过去7天的用户行为,则在选择的回溯日期为8月20日时,标签计算所用到的数据的日期为8月12
 日至8月19日;若选择的回溯日期为8月21日时,则标签计算所用到的数据的日期为8月13日至8月20日,依此类推。

- 当回溯日期已有标签数据时,提供全部标签重新计算、计算失败的标签重新计算、使用历史规则计算的标签重新计算三种 方式。
- 为了尽可能还原回溯日期当天的数据情况,当标签的计算有使用到用户属性时,可以使用回溯日期当天备份的的用户属性数据做计算,若关闭则使用数据库中最新的用户属性数据做计算(默认关闭)。
 说明:若开启该功能,计算耗时相对较长,可根据实际业务需求选择是否开启该功能。
- 若使用备份的用户属性数据做计算,当计算时间的数据缺失时,提供使用最新的用户属性数据替代计算、使用相对最近的 数据计算两种方式。
- 2. 对于已经回溯(备份)过的日期,可以点击「重新计算」按钮对该日期的数据进行计算。
- 3. 如果该标签的回溯版本已经不需要了,可以点击「清理标签数据」清理回溯版本。

5 实例

5.1 描述用户行为特征

标签是描述用户历史行为的数据,相比原始数据信息,标签更有利于监控和分析。通过自定义标签,可以对用户进行分层,设置用户画像;通过数值标签,可以计算得到用户总和、均值等总体数据,方便进行日常运营监控;通过初末次标签,可以获得 用户初次/末次的行为特征,有利于营销策略的制定和调整;通过SQL标签,可以自定义对用户行为构建标签,满足个性化需求;通过导入标签,可以上传已有的用户标签,补充数据维度,丰富用户画像。

5.2 在分析模型中用作筛选条件

标签可用作分析的筛选条件。如使用自定义标签创建了标签「5A人群划分」,该标签根据用户的使用深度将用户分成了A1-A5活 跃人群。在分析中,若只想对A1用户进一步分析,就可以添加标签筛选条件,只查看A1人群的分析结果。

5.3 在分析模型中用作分组项

标签可用作分析中的分组项。在分析中,若想了解A1- A5的用户的行为差异,可以将5A人群划分用作分组项。

5.4 回溯版本在分析中的使用

为了尽可能还原事件发生时的数据情况,当在分析中使用标签时,如果标签有回溯版本,则需要在计算时选择使用标签的版本。

- 最新版本:使用最新的标签数据做计算,即最近一次手动/自动更新的标签数据。
- 动态匹配:使用事件发生当日的标签值做计算。若事件是8月3日发生,则寻找8月3日的标签数据版本做计算。
- 历史版本:使用用户指定的某个日期版本的标签值做计算,仅可选择有回溯版本的日期。

用户分群

帮助中心 > 功能手册 > 用户洞察 > 用户分群

1. 概述

用户分群通过某些特定的规则找到对应的用户群体。用户分群模块支持用户将满足特定条件的人群圈选为一个分群,深入挖掘 用户行为,助力精准营销。

2. 位置和使用权限

功能位置:用户洞察-用户分群

使用权限:不同的项目角色操作权限如下

项目角色	权限	权限明细
管理员	新增分群、 管理所有分群	新增分群、所有分群 (修改、复制、重新计算、删除)
分析师	新增分群、 管理自建分群	新增分群、自建分群(修改、复制、重新计算、 删除)、他人创建的分群(复制)
普通用户	无权限	无权限

3. 页面概览

用户可以使用平台提供的分群创建方式创建新的分群,并可在主页对已有分群进行管理,查看分群的数据详情。

3.1 创建新的分群

若用户想要创建一个新的分群,需要完成以下的几个步骤:

第一步:点击分群管理主页的「新增」创建分群,选择好想要使用的标签创建方式后,点击进入标签配置页面。

目前平台提供以下几种分群创建方式,各种创建方式的功能说明如下:

- 自定义分群:对用户属性/用户行为设定筛选条件,将满足筛选条件的用户定义为一个分群。如将近一年充值笔数大于10的 人群定义为高价值人群。
- SQL分群:使用自定义的SQL语句定义分群规则,将执行结果定位为一个分群。
- 导入分群:将导入的人群定义为一个分群。
- 结果分群:在数据分析/用户标签中,可以将满足某个共同特征的群体另存为一个分群,如将标签「用户价值分层」中标签 值为高价值的用户保存为一个分群。

第二步:进行分群规则的配置。分群的配置包括三个部分,分别是分群基础信息、分群规则配置、更新与备份。

1. 分群基础信息:填写分群名、标签显示名、分群备注等信息。

2. 分群规则配置

- 该分群的分析主体:作为不同用户的标识,默认为海纳ID。若以设备ID作为分析主体,则将不同设备ID的用户视作不同用 户;若以账号ID作为分析主体,则将不同账号ID的用户视作不同用户。若当前下拉框中没有想要的分析主体,可至「项目管 理-分析配置-分析主体」中进行相关配置。分群创建后,分析主体不可更改。
- 规则定义:配置分群的计算规则。可对事件添加筛选条件,点击「属性筛选」添加筛选条件来筛选满足条件的用户。

(1) 若想要添加多条筛选条件:点击「属性筛选」可添加一级筛选条件,点击「添加并列条件」可添加二级筛选条件,最多只能添加到二级,筛选条件添加条数不限。

(2) 添加新的条件需要设定和原条件的关联关系,点击"且/或"可切换关联关系。

3. 更新与备份

- 计算时区:选择分群的计算时区,默认选择项目时区。可在「项目管理-分析配置-时区设置」中设置是否开启多时区功能。
- 更新方式:分为定期更新和手动更新两种。定期更新即系统将在指定时间定期计算并更新分群数据,并以前一日作为计算基准日期;手动更新数据不会自动进行计算更新,用户可手动点击「更新数据」进行计算。若选择定期更新,需要进行「时间设置」和「备份方式」的设置
- 时间设置:设置分群更新的具体时间。
- 备份方式:若自动备份,将自动存储定期更新的分群数据,可至「回溯版本管理」中查看该分群的历史数据;否则仅保留 最新计算的分群数据,以往的分群数据将被覆盖。

第三步:分群基础信息、分群规则配置、更新与备份均设定好以后,点击「保存并计算」按钮即可创建分群,并同步至分群管 理主页。

3.2 管理已有分群

创建好的分群将同步至分群管理主页,可进行编辑、复制、重新计算、删除、查看详情等操作:

- 点击「编辑」可修改分群配置
- 点击「复制」可复制分群配置生成一个新的分群
- 点击「重新计算」会计算分群数据
- 点击「删除」可删除该分群
- 点击「分群名称」可跳转至分群详情页面

3.3 查看分群详情

分群详情页主要包括展示分群基础信息、数据可视化、回溯版本管理三个部分。

- 在页面左侧,可查分群的配置信息、变更记录等信息。
- 可查看分群的数据详情,并可查看分群的用户列表。
- 对分群的定期备份版本进行管理。

4. 功能结构

4.1 不同创建方式功能说明

4.1.1 自定义分群

自定义分群根据若干条件筛选出满足条件的用户,定义为一个分群。如将近一年充值笔数大于10的人群定义为高价值人群,用 于营销投放。

可筛选内容包括:

- 用户属性筛选:
 - (1) 某文本型用户属性等于/不等于/包括/不包括…
 - (2) 某数值型用户属性等于/不等于/包括/不包括…
 - (3) 某时间型用户属性位于区间/小于/大于/相对当前日期…
 - (4) 某布尔型用户属性为真/为假/有值…
- 用户行为筛选:
 - (1) 用户完成某个指定事件的次数/天数/小时数 等于/大于/小于…
 - (2) 用户完成某个指定事件数值类型属性的总和/均值/中位数/最大值/最小值/去重数 等于/大于/小于…
 - (3) 用户完成某个指定事件布尔类型属性的为真数/为假数/为空数/不为空数/去重数 等于/大于/小于…
 - (4) 用户完成某个指定事件其他类型属性的去重数 等于/大于/小于…
 - (5) 用户没做过某个事件

筛选格式:

- 用户属性筛选:「选择用户属性」+「选择算子」+「某值」
- 用户行为筛选:「时间」+「做过」+「某事件」+「算子/事件属性.算子」+「条件判断」+「某值」
 或「时间」+「没做过」+「某事件」

注1:若分析主体为用户自定义的分析主体且来源于事件属性,则用户行为筛选的元事件选择下拉框中可选的元事件只展示包含 该事件属性(分析主体)的元事件。

4.1.2 SQL分群

当已有的创建方式不能满足需求时,可以使用自定义的SQL查询语句将指定的用户圈选为一个分群。

- 查询结果仅一列,即为圈选出的满足条件的分析主体ID,其字段需与下拉框中选择的分群的分析主体保持一致。
- SQL分群支持定期更新,若想统计动态时间的分群结果,请使用动态时间窗的语句。如统计最近30天,则为WHERE part_date > DATE_SUB (CURDATE (), INTERVAL 30 DAY)。
- 若想预览某个表的数据结构和内容,可以点击左侧的数据表,在「表结构」和「数据概览」中进行查看,如图所示:

4.1.3 导入分群

导入分群支持上传本地文件,将上传的用户定义为一个人群,可以选择某一用户属性作为用户的关联字段。

- 上传文件共一列,即该分群的身份识别信息
- 当选择的分析主体是预置分析主体海纳ID时,需选择关联属性(即选择上传文件的字段名)
- 导入分群不支持更新与备份。

4.1.4 结果分群

在数据分析/用户标签中,当有需要重点分析或关注的行为时,可以通过点击表中的数据将满足条件的人群保存为一个结果分群,保存后,可以用作营销投放,也可进一步识别分析。

- 1. 新建结果分群
- 第一步:在分析/展示的数据表中点击想要了解的数据进入用户列表。
- 第二步:将指定人群保存为结果分群。
- 第三步:在分群管理列表中查看,并可在分析中操作使用。

2. 结果分群的操作

- 结果分群仅支持编辑和删除的操作,只能编辑分群的显示名和备注,不可修改规则,也不可复制分群和重新计算。
- 结果分群不支持更新备份。

4.2 聚合和筛选算子说明

• 不同数据类型的事件属性支持的聚合计算规则汇总如下:

事件	事件属性 (数据类型)	聚合算子
任意事件		次数、天数、小时数
元事件	事件属性 (数值型)	总和、均值、中位数、最小值、最大值、去重数
元事件	事件属性 (布尔型)	为真数、为假数、为空数、不为空数、去重数
元事件	事件属性 (其他类型)	去重数

• 不同数据类型的事件属性支持的条件筛选规则汇总如下:

属性的数据类型	条件筛选可选项
文本型	等于、不等于、包括、不包括、有值、无值、正则匹配、正则不匹配
数值型	等于、不等于、小于、小于等于、大于、大于等于、有值、无值、区间
时间型	位于区间、小于、小于等于、大于、大于等于、相对当前日期

	(区间) 、相对当前日期(过去)、有值、无值、等于、不等于
布尔型	为真、为假、有值、无值

4.3 数据可视化

在数据可视化部分,可查看分群的覆盖用户数、导出数据。

4.4 查看分群数据

数据表中会展示分群最新版本和回溯版本(仅例行更新且自动备份的分群有回溯版本)中的所有数据,最近一次手动/定期更新的数据将在图表中用红点标注,并以今天的日期展示该条分群数据。

• 点击具体的数值可以查看用户列表

4.5 查看用户列表

通过点击数据详情页面数据表中的具体数值,可链接跳转至用户列表页面,查看已选人群的更多用户属性信息:

- 用户列表支持查看用户其他属性特征,了解用户更多信息。
- 用户列表支持导出数据

4.6 回溯版本管理

例行更新且自动备份的分群会存在回溯版本,存储每日计算的分群数据。用户的行为会随着时间发生变化,若开启自动备份, 分群数据会每日定期更新。

• 如果该分群的回溯版本已经不需要了,可以点击「清理分群数据」清理回溯版本。

5 实例

5.1 辅助下钻分析

在分析中,可对需要重点关注的数据进行细查,将人群保存到结果分群,再作为筛选条件在分析中细查分析,深入挖掘用户行为特征。

5.2 圈选投放目标人群

自定义分群可通过用户设定的条件,筛选满足条件的人群,并将其圈选为一个人群包。将人群细分实施精细化运营,可以有效 提高转化率。如对新用户、活跃用户、首购用户、复购用户、目标用户、沉默用户可制定不同的运营策略。

• 如想对流失用户进行召回,可通过自定义分群圈选出流失的用户。如定义过去30天发生过但过去7天未发生过启动App行为的用户定义为流失用户,则可找出流失人群的信息,对其进行召回。

5.3 排除特殊人群

除了圈选正向人群外,也可使用分群圈选排除人群,在分析中可通过设定筛选条件将属于该分群的用户排除。

• 如某些用户存在恶意刷单等异常行为,可在日常运营监控中将此部分人群排除再进行分析。

用户详情

帮助中心 > 功能手册 > 用户洞察 > 用户详情

1. 概述

用户详情是某指定用户的全景画像,展示了该用户的基础属性信息、所属标签和操作行为路径等信息,方便快速获取用户特征。

2. 位置和使用权限

功能位置: 用户详情提供多个入口,方便各个分析模型对指定用户进行分析,从各分析模型的用户列表页,点击用户ID信息,即可进入用户详情页,其主要的入口有:

- 事件分析模型中的分析维度是"用户数/总次数"的"用户列表"页
- 漏斗分析模型中查看"转化人数/流失人数"的"用户列表"页
- 留存分析模型的查看"留存人数、流失人数"的"用户列表"页
- 分布分析模型中查看每一区间的人数的"用户列表"页
- 路径分析模型中"查看节点详细信息"中的"用户详情"页
- 属性分析模型中的分析维度是"用户数"的"用户列表"页
- 间隔分析模型中的查看用户数的"用户列表"页
- 用户标签、用户分群中查看"用户群"的"用户列表"页
- 用户查询中满足搜索条件的"用户列表"页或点击用户ID直接进入

这里以事件分析模型为例,其入口如下:

使用权限:不同的项目角色操作权限如下:

项目角色	权限	权限明细
管理员	查看用户详情	查看用户的基础属性信息、所属标签和操作行为路径等信息
分析师	查看用户详情	查看用户的基础属性信息、所属标签和操作行为路径等信息
普通成员	查看用户详情	查看用户的基础属性信息、所属标签和操作行为路径等信息

3. 页面概览

主页面包含用户属性区、用户标签区、事件统计区、行为序列详情区、用户切换区:

- 用户属性区:查看当前用户的用户属性信息
- 用户标签区:查看当前用户的所在标签和标签值

- 事件统计区:查看所选时间范围内,用户参与事件的总数和分布情况
- 行为序列详情区:按查看用户参与事件的行为序列和事件属性,了解用户操作行为路径
- 用户切换区:可在页面左下角切换想要查看的用户。

注:当分析主体不是预置分析主体"海纳ID"时,用户属性区只展示该分析主体对应的ID信息,不支持筛选用户属性的操作。

4. 功能结构

4.1 用户属性区

- 用户属性区默认展示该用户前10个用户属性(包括预置属性和自定义属性),显示其属性名称及其属性值。
- 用户可以自主选择想要展示的用户属性,通过点选下拉框勾选。勾选后作为个人配置进行保存。下拉框显示的数据来源为
 用户基础属性、用户维度表属性、用户虚拟属性。

4.2 用户标签区

• 用户标签区展示该用户所在的标签,按创建方式展示,可通过筛选筛选指定创建方式的标签。

4.3 事件统计区

- 用户第一次进入时,事件统计区默认展示过去7天、所有事件、按天汇总的柱形图,用户可通过点击功能按钮选择想要查看 的时间、事件和展示粒度。
- 柱形图将展示参与事件的总数
- 饼图将展示各个行为事件的分布情况。

4.4 行为序列详情区

4.4.1 初始页面

初始展示内容为「事件统计模块」所选日期范围的行为事件。

- 当时间粒度为按天、按月、按周时,"行为序列详情"按天分组显示;当时间粒度为按小时,"行为序列详情"按小时分组显示。
- 支持对日期的升降序设置。

4.4.2 展开事件详情

可点击具体事件展开查看其事件属性详情。默认展示当前事件的全部上报属性,包括预置属性和自定义属性。

4.4.3 设置属性外显

展开事件详情后,鼠标悬停某事件属性时,可将其设置为外显属性,之后无需展开事件详情,就可以在事件名后面查看属性值。

事件属性外显仅针对有该事件属性的事件:

4.4.4 定位事件

行为序列中默认展示该用户在时间范围内的全部勾选事件,如只想看某一事件的结果,可使用「定位事件」功能:

点击定位事件后,用户可定位其中一个事件,如底部tab点击:

4.4.4 查看前后行为

当定位某事件后才有该功能。可点击"查看前后行为"了解该事件的前序、后序事件。

• 查看前后行为页面

4.5 切换用户

可点击左下角的切换用户来快速切换查看的用户,切换的用户来源于上一级的"用户列表"页,最多展示1000条:

5. 实例

5.1 深度了解用户行为

用户查询从用户维度了解用户行为,通过用户的行为序列,可以更具体的分析到用户产生特定行为的原因,判断哪些行为可能会对特定行为有影响,方便后续的定位和筛查,对特定问题的产生原因进行定位。

用户查询

帮助中心 > 功能手册 > 用户洞察 > 用户查询

1. 概述

用户查询位于导航栏,方便用户在任意页面匹配满足条件的用户。

2. 位置和使用权限

功能位置:导航栏-用户查询

使用权限:不同的项目角色操作权限如下

项目角色	权限	权限明细
管理员	用户搜索、 查看用户列表	搜索满足条件的用户, 并查看用户的具体行为信息
分析师	用户搜索、 查看用户列表	搜索满足条件的用户, 并查看用户的具体行为信息
普通用户	无权限	无权限

3. 页面概览

第一步: 在弹框中输入想要查询的匹配条件:

第二步: 筛选出满足条件的用户:

- 页面上最多显示10条用户信息,展示搜索结果和对应的用户ID。
- 可点击「查看用户列表」查看所有满足条件的用户。
- 可直接点击用户ID,查看单用户详情。

第三步: 查看满足条件的用户列表:

- 支持下载完整数据。
- 支持将所有人群保存为一个结果分群。
- 支持查看用户的其他属性信息。

第四步: 若想查看某个用户基本信息和操作行为路径,可以点击用户的ID,进入用户详情页面,其功能详情请至「用户查询」 查看:

4. 功能结构

4.1 搜索规则

• 下拉框可以查询"账户ID"、"访客ID"和所有用户属性。

• 不同数据类型的用户属性支持的条件筛选规则汇总如下

属性的数据类型	条件筛选可选项
文本型	等于、不等于、包括、不包括、有值、无值、正则匹配、正则不匹配
数值型	等于、不等于、小于、小于等于、大于、大于等于、有值、无值、区间
时间型	位于区间、小于、小于等于、大于、大于等于、相对当前日期 (区间) 、相对当前日期(过去) 、有值、无值、等于、不等于
布尔型	为真、为假、有值、无值

5 实例

5.1 查看某特定用户的行为数据

当定位到某个用户的行为有异常时,想要对该用户的行为进行细查,则可以使用用户查询功能。通过搜索用户的ID(如账户ID、 访客ID),进入用户详情列表,查看用户的具体表现。

数据中心

-88-

数据接入

帮助中心 > 功能手册 > 数据中心 > 数据接入

1 概述

通用数据接入包括通用数据接入与埋点数据接入。其中,通用数据接入提供了第三方数据集成服务,可以帮助用户接入第三方 广告投放平台、大数据服务平台,让用户可使用的数据更加全面;埋点数据接入展示了各端埋点文档的入口,包括服务器埋 点、客户端埋点以及工具导入,可以根据业务需求,点击名称查看对应端的接入文档。

2 位置和使用权限

操作位置:数据中心-数据接入

使用权限:

项目角色	权限	权限明细
管理员	查看、编辑	查看、编辑
分析师	查看	查看
普通用户	无权限	无权限

3页面概览

通用数据接入主要为外部三方导入,目前已接入巨量引擎 埋点数据接入包括服务器埋点、客户端埋点以及工具导入

4 功能概述

4.1 通用数据接入

目前已提供巨量引擎接入服务,点击「配置」按钮到达巨量引擎配置页面 打开集成开关,点击新增集成方案进行相关参数的填写

参数名称	说明	获取位置	是否
方案名称	自定义名称		是
APP_ID	巨量引擎后台的项目识别 APP_ID	开发者平台 APP 应用详情页	是
Secret	巨量引擎后台的项目 Secret	开发者平台 APP 应用详情页	是

广告主ID列表	巨量引擎后台用于投放广告的广告主 (广告账户) Secret	巨量引擎商务管理平台的 「推广」页面「账户」 标签页	是
登录用户ID	用于登录巨量引擎后台的登录用户ID (非登录邮箱)	API 授权界面	是
代理名称	可不填		否
平台	iOS/Android		否
•			•

填写完成后将回调地址填入巨量引擎平台,待接入状态变为接入成功后,即表示有数据接入成功,数据将以 oe_advertising_data_collection_tags 事件的形式入库事件表。 具体对接文档如下: 巨量引擎对接文档

4.2 埋点数据接入

埋点数据接入包括服务端埋点、客户端埋点、工具导入以及数据接收地址

- 服务端埋点包括java、python、php、ruby,服务器埋点可准确采集业务操作,确保数据完整上报。
- 客户端埋点包括web、ios、android,客户端埋点可以详细采集用户行为,完整采集用户本地标识。
- 数据导入工具包括Batchimporter、LogAgent、FormatImporter、HdfsImporter、SFTP,数据导入工具通常用于历史数据导入,支持对文件进行批量传输。

元数据管理

元事件

帮助中心 > 功能手册 > 数据中心 > 元数据管理 > 元事件

1 概述

元事件是一系列属性的集合、不具备数据类型。代表了用户的某个或一系列有意义的行为,一个事件主要包含两部分信息,一 部分用以描述该行为如何发生,另一部分是该行为的属性。元事件页面主要用于对采集的事件进行统一管理,可对元事件进行 新增、修改等操作,并可对事件的系统显示状态及数据上报状态进行修改。

2 位置和使用权限

操作位置:数据中心-元数据管理-元事件 使用权限:

项目角色	权限	权限明细
管理员	元事件查看、新增、编辑	查看元事件页面、新增、 编辑元事件
分析师	元事件查看、新增、编辑	查看元事件页面
普通用户	无权限	无权限

3页面概览

元事件主要包括元事件列表展示及元事件的增改查等功能,设置完成后,此元事件可在数据空间、数据分析、用户洞察等模块 对应展示使用。

4 功能结构

4.1 列表展示

元事件管理列表包括事件名、事件显示名、事件描述、近一天事件量、是否可见、是否上报、埋点平台、更新时间及操作等信 息。

4.2 查询

可在搜索框输入事件名或事件显示名的关键字段查询所需事件,默认情况下展示所有事件。

4.3 新增

4.3.1 单条新增

点击「新增」按钮将跳转到新增页面,可对元事件进行单条新增,此入口新增的事件将默认为自定义事件

- 事件名:是事件在系统中的唯一标识,不允许重复,不能以数字开头,且只包含大小写字母、数字和下划线。
- 事件显示名:是事件在数据分析、用户洞察等模块使用过程中的显示名称。
- 事件描述:对该事件的描述信息,方便用户更好的理解业务。
- 是否可见:此开关会影响此元事件是否在数据分析、用户洞察等模块展示。「否」表示该事件字段在数据分析和用户洞察 菜单栏将同步隐藏,不可使用;「是」表示该事件字段在数据分析和用户洞察菜单栏将同步展现。
- 是否上报:此开关会影响此事件数据是否进行采集。「否」表示不再采集该事件的埋点数据,历史数据保留,增量数据不 再录入数据库;「是」表示开始接收该埋点数据。
- 埋点平台:包括ios、android、js、微信小程序、支付宝小程序等,可根据埋点需求进行选择。
- 关联属性:可直接在已有事件属性中进行选择或直接在文本框中输入新增的事件属性。若选择已有的事件属性,将自动同步显示该属性的属性显示名和数据类型;若自定义输入属性名,将把该属性名默认为新增,需输入属性显示名并选择对应的数据类型,此事件属性将默认为自定义属性。

4.3.1 批量导入

点击「批量导入」按钮跳转到批量导入页面,可对元事件进行批量新增。

第一步:文件上传,将需要采集的元事件、事件属性、用户属性按照海纳云的【数据采集文档】模版格式进行整理上传。

海纳云数据采集文档:数据采集文档.xlsx

第二步:事件表校验,系统将自动对上传的的事件表进行校验,若出现错误或相似的事件/事件属性将直接在页面进行标红且进行错误提示,用户可修改之后再次进行校验,待校验通过后即可点击第三步。

第三步:公共事件属性校验,系统将自动对上传的的公共事件属性进行校验,若出现错误或相似的公共事件属性将直接在页面 进行标红且进行错误提示,用户可修改之后再次进行校验,待校验通过后即可点击第四步。

第四步:用户表校验,系统将自动对上传的的用户表进行校验,若出现错误或相似的用户属性将直接在页面进行错误提示,用 户可修改之后再次进行校验,待校验通过后即可点击完成。

新增后的元事件、事件属性、用户属性可在数据分析、用户洞察、数据空间等模块使用。

4.4 编辑

点击操作栏中「编辑」按钮可对事件显示名、事件描述、是否可见、是否上报、埋点平台进行修改。

注:对于埋点平台的修改,如果该条数据已经存在上报数据,则不支持删除已选埋点平台,只能进行新增;如果没有数据上报,则支持删除已选埋点平台,也支持进行新增。

4.5 事件详情

点击具体某个元事件的名称,可查看该元事件关联的事件属性的关联明细,包括属性名、属性显示名、数据类型、是否可见、 属性类别等信息

4.6 删除

针对不需要的元事件,可点击操作栏中的「删除」按钮进行删除。 注1: 元事件删除后,此元事件的历史数据将同步删除无法使用

注2: 删除此元事件前,需将此元事件的是否上报按钮关闭

附件

• 数据采集文档.xlsx

事件属性

帮助中心 > 功能手册 > 数据中心 > 元数据管理 > 事件属性

1 概述

事件属性是在这个事件中需要采集的具体信息,是每种事件所独有的属性。除了是与具体事件相关的属性外,还能够反映用户 在进行该行为时的状态。事件属性页面主要用于对采集的事件属性进行统一展示与管理,可对事件属性进行新增、修改等操 作,并可对事件属性的系统显示状态进行修改。

2 位置和使用权限

操作位置:数据中心-元数据管理-事件属性

使用权限:

项目角色	权限	权限明细
管理员	查看、新增、编辑	查看事件属性页面、新增、编辑事件属性
分析师	查看、新增、编辑	查看事件属性页面
普通用户	无权限	无权限

3页面概览

事件属性主要包括事件属性列表及事件属性的增改查等功能,设置完成后,事件属性可在数据分析、用户洞察等模块对应展示。

4 功能结构

4.1 列表展示

事件属性管理列表包括属性名、属性显示名、数据类型、是否可见、属性类别、维度表、更新时间等信息。

4.2 查询

可在搜索框输入事件属性名或事件属性显示名的关键字段查询所需事件属性,默认情况下展示所有事件属性。

4.3 新增

方式1:可直接在元事件页面对元事件及其关联的事件属性进行批量上传。

方式2:当在元事件页面批量上传的事件属性不满足需求时,可点击「新增」按钮对事件属性进行单条新增。

- 属性名:是属性在系统中的唯一标识,不允许重复,不能以数字开头,且只包含大小写字母、数字和下划线。
- 属性显示名:是事件属性在数据分析、用户洞察等模块使用过程中的显示名称。
- 数据类型:此事件属性的数据类型
- 是否可见:此开关会影响此事件属性是否在数据分析、用户洞察等模块展示。「否」表示该事件属性字段在数据分析和用 户洞察菜单栏将同步隐藏,不可使用;「是」表示该事件属性字段在数据分析和用户洞察菜单栏将同步展现。
- 是否为公共属性:此开关会影响该事件属性的关联事件。
 - 「否」表示该事件属性为自定义属性,用户需自行设置其关联事件;
 - 「是」表示该事件属性为公共属性,默认关联所有元事件。
- 关联事件:若该事件属性非公共属性,用户可选择其关联的元事件。

4.4 编辑

点击「编辑」按钮可对属性显示名、是否可见等信息进行修改。

注:当此事件属性为非公共属性时,允许修改其关联事件。

4.5 属性详情

点击「事件属性名称」链接,可查看该事件属性的事件关联明细,包括事件名、事件显示名、事件类别、是否上报、是否可见 信息。

4.6 删除

针对不需要的事件属性,可点击操作栏中的「删除」按钮进行删除。 注1:事件属性删除后,此事件属性的历史数据将同步删除无法使用

注2: 删除此事件属性前,需将此事件属性的是否上报按钮关闭

用户属性

帮助中心 > 功能手册 > 数据中心 > 元数据管理 > 用户属性

1 概述

用户属性表示的是用户不变的属性以及最新状态,比如用户的 ID、注册时间或者累计付费金额等,通过用户属性,可以在用户 行为分析时快速筛选出要分析的用户。用户属性页面主要用于对采集的用户属性进行统一展示与管理,可对用户属性进行新 增、修改等操作,并可对用户属性的系统显示状态进行修改。

2 位置和使用权限

操作位置:数据中心-元数据管理-用户属性

使用权限:

项目角色	权限	权限明细
管理员	查看、新增、编辑	查看用户属性页面、新增、编辑用户属性
分析师	查看、新增、编辑	查看用户属性页面
普通用户	无权限	无权限

3页面概览

用户属性主要包括用户属性列表及用户属性的增改查等功能,设置完成后,用户属性可在分析等模块对应展示。

4 功能结构

4.1 列表展示

用户属性管理列表包括属性名、属性显示名、数据类型、是否可见、属性类别、维度表、更新时间及操作等信息。

4.2 查询

可在搜索框输入用户属性名或用户属性显示名的关键字段查询所需用户属性,默认情况下展示所有用户属性。

4.3 新增

方式1: 可直接在元事件页面对用户属性进行批量上传。 方式2: 当在元事件页面批量上传的用户属性不满足需求时,可点击「新增」按钮,对用户属性进行单条新增。

- 属性名:是属性在系统中的唯一标识,不允许重复,不能以数字开头,且只包含大小写字母、数字和下划线。
- 属性显示名:是用户属性在数据分析、用户洞察等模块使用过程中的显示名称。
- 是否可见:此开关会影响此用户属性是否在数据分析、用户洞察等模块展示。「否」表示该用户属性字段在数据分析和用 户洞察菜单栏将同步隐藏,不可使用;「是」表示该用户属性字段在数据分析和用户洞察菜单栏将同步展现。

4.4 编辑

点击操作栏中「编辑」按钮可对除属性名、数据类型之外的事件属性信息进行修改

4.5 删除

针对不需要的用户属性,可点击操作栏中的「删除」按钮进行删除。 注1: 用户属性删除后,此用户属性的历史数据将同步删除无法使用

注2: 删除此用户属性前,需将此用户属性的是否上报按钮关闭

维度表属性

帮助中心 > 功能手册 > 数据中心 > 元数据管理 > 维度表属性

1 概述

维度表属性是基于已有属性上传维度表而新生成的属性,可将原先上传的数据映射为另一种展示值或计算值。对于已经埋点的 事件属性和用户属性,我们可以通过上传维度表的方式将其进行映射为我们需要的结果,使展示值与埋点值不同,并在后续直 接或加工使用。

使用示例:对于某些属性值为英文或特殊规则展示的事件/用户属性,我们可以引入维度表属性转换成通俗易懂的表达方式进行 展示,方便用户使用。

2 位置和使用权限

操作位置:数据中心-元数据管理-事件属性/用户属性

使用权限:

项目角色	权限	权限明细
管理员	维度表查看、新增、编辑、 删除	查看维度表页面、新增、编辑、 删除维度表属性
分析师	维度表查看	查看维度表
普通用户	无权限	无权限

3 页面概览

以事件属性页面为例,可对维度表属性进行新增、替换、下载、删除等操作。

4 功能结构

4.1 新增

对于需要设置维度表属性的字段,可点击维度表列的「添加维度表」,进行维度表的新增。 第一步:上传维度表,将需要新增的维度表属性按照【维度表模版】格式进行内容填写后点击上传 维度表模版.xlsx

维度表上传文件规则:

- 1. 首行首列的内容需与原始属性字段名保持一致
- 2. 维度表属性字段名只支持字母开头且仅包含字母数字下划线

第二步:编辑属性维度,根据需求填写映射后的维度表属性显示名和对应的数据类型。 第三步:完成,系统将为您提示维度表添加情况,同时展示解析总行数、成功行数、错误丢弃行数、出现错误的列等信息。 维度表新增成功后,可在列表查看该属性对应的维度表属性,并可在数据分析及其他模块作为筛选值、分组值等进行后续使用

4.2 替换

点击「替换」可对维度表属性进行重新上传,该步骤与新建维度表步骤一致,替换成功后,之前的维度表将会被覆盖。

4.3 下载

点击下载将该维度表数据下载到本地,下载内容与上传时的内容一致。

4.4 删除

若不再需要某维度表属性数据,可点击「删除」按钮将其进行删除,删除后,分析及其他模块也将无该维度表属性。

虚拟事件

帮助中心 > 功能手册 > 数据中心 > 元数据管理 > 虚拟事件

1 概述

为了更方便灵活的分析,可以使用虚拟事件。将元事件进行拆解或者组合形成新的「虚拟事件」,从而减少每次在分析过程中反复对固定常用的事件进行筛选配置。

2 位置和使用权限

操作位置:数据中心-元数据管理-元事件

使用权限:

项目角色	权限	权限明细
管理员	虚拟事件查看、新增、编辑	查看虚拟事件页面、新增、 编辑虚拟事件
分析师	虚拟事件查看	查看虚拟事件页面
普通用户	无权限	无权限

3页面概览

虚拟事件页面即元事件页面,主要包括事件列表展示及事件的增改查等功能,设置完成后,此事件可在数据空间、数据分析、 用户洞察等模块对应展示使用。

4 功能结构

4.1 列表展示

元事件管理列表包括事件名、事件显示名、事件描述、事件类别、近一天事件量、是否可见、是否上报、埋点平台、更新时间 及操作等信息。

4.2 查询

可在搜索框输入事件名或事件显示名的关键字段查询所需事件,默认情况下展示所有事件。

4.3 新增

点击「新增虚拟事件」按钮将跳转到新增虚拟事件页面,可对元事件进行单条新增,此入口新增的事件将默认为自定义事件

- 虚拟事件名:是此虚拟事件在系统中的唯一标识,不允许重复,默认以VE_开头,且只包含大小写字母、数字和下划线。
- 虚拟事件显示名:是虚拟事件在数据分析、用户洞察等模块使用过程中的显示名称。
- 虚拟事件描述:对该事件的描述信息,方便用户更好的理解业务。
- 是否可见:此开关会影响此虚拟事件是否在数据分析、用户洞察等模块展示。「否」表示该事件字段在数据分析和用户洞 察菜单栏将同步隐藏,不可使用;「是」表示该事件字段在数据分析和用户洞察菜单栏将同步展现。
- 虚拟事件定义:可将元事件进行拆解或者组合

4.4 编辑

点击操作栏中「编辑」按钮可对虚拟事件显示名、事件描述、是否可见进行修改。

4.5 复制

点击操作栏中的「复制」按钮可对虚拟事件进行复制。

4.6 事件详情

点击具体某个虚拟事件的名称,可查看该虚拟事件关联的事件属性的关联明细,包括属性名、属性显示名、数据类型、是否可见、属性类别等信息,此虚拟事件关联的事件属性为虚拟属性新增时所涉及到的元事件所关联的事件属性的并集。

虚拟属性

帮助中心 > 功能手册 > 数据中心 > 元数据管理 > 虚拟属性

1 概述

虚拟属性,是指在数据入库之后通过 SQL 表达式对已有的事件属性和用户属性进行二次加工,产生一个新的属性值。

2 位置和使用权限

操作位置:数据中心-元数据管理-事件/用户属性

使用权限:

项目角色	权限	权限明细
管理员	虚拟属性查看、新增、编辑	查看虚拟属性页面、新增、 编辑虚拟属性
分析师	虚拟属性查看	查看虚拟属性页面
普通用户	无权限	无权限

3 如何新增虚拟属性

点击事件/用户属性列表的「新增虚拟属性」,将跳转到如下页面

- 虚拟属性名:是此虚拟属性在系统中的唯一标识,不允许重复,默认以VP_开头,且只包含大小写字母、数字和下划线。
- 虚拟属性显示名:是虚拟属性在数据分析、用户洞察等模块使用过程中的显示名称。
- 虚拟属性描述:对该属性的描述信息,方便用户更好的理解业务。
- 是否可见:此开关会影响此虚拟属性是否在数据分析、用户洞察等模块展示。「否」表示该事件字段在数据分析和用户洞 察菜单栏将同步隐藏,不可使用;「是」表示该事件字段在数据分析和用户洞察菜单栏将同步展现。
- 数据类型:可自行选择此虚拟属性所属的数据类型
- 关联事件:选择此虚拟事件属性所关联的事件,若为虚拟用户属性则无需选择
- SQL表达式:虚拟属性根据表达式中引用到的属性类型分为用户虚拟属性和事件虚拟属性,需要注意的是确定了虚拟用户属性的类型之后只能引用用户表内的属性;确定了虚拟事件属性的类型之后只能引用事件表内的属性。

4 使用场景

应用场景一:时间差计算

可使用函数datediff计算用户属性"时间1"和用户属性"时间2"的间隔天数,生成虚拟用户属性"生命周期天数",如: datediff(CAST(users. time1 AS DATETIME), CAST(users. time2 AS DATETIME)) 应用场景二:属性计算

可直接对属性进行数值计算,使用事件属性income和事件属性expend,生成虚拟事件属性:"利润",如:event. H_income - event. expend

监控预警

帮助中心 > 功能手册 > 数据中心 > 监控预警

1 概述

监控预警用于对重点数据或指标进行监控,可对同一指标进行分组,以不同的监控触发标准对各分组值进行监控,更加快捷智能的洞察异常数据。对于监控指标异常情况,可第一时间进行多端提醒,便于快速发现问题,缩短响应时间,提高对于产品的把握度。

2 位置和使用权限

操作位置:数据中心-监控预警 使用权限:

项目角色	权限	权限明细
管理员	新增监控预警、 管理所有监控预警	新增监控预警、修改、 删除所有监控预警
分析师	新增监控预警、 管理自建监控预警	新增监控预警、修改、 删除自建监控预警
普通用户	无权限	无权限

3页面概览

监控预警包括监控列表展示、监控的新增、编辑、删除以及查看监控详情等操作。

4 功能结构

4.1 列表展示

监控预警列表包括监控名称、累计监控次数、指标值、差异率、监控粒度、监控情况、最近监控时间、创建时间、创建人、状态等信息。

4.2 新增

拥有监控预警操作权限的用户可点击「新增监控」按钮跳转到新增页面对监控预警进行新增。

新增监控需填写三部分内容:基本信息、监控设置、通知方式

1. 基本信息:基本信息包括监控名称、分析主体、监控指标、分组、监控状态。

- 分析主体:可根据需求选择该指标的分析主体,默认为海纳ID,可选项需在「项目管理-分析配置-分析主体」提前配置
- 监控指标:选择「事件」+「事件属性」+「聚合算子」或「事件」+「聚合算子」作为监控指标,此外可对监控指标进行条件筛选

1)选择事件:可以选择任意一个元事件/虚拟事件。若分析主体为用户自定义的分析主体且来源于事件属性,则下拉框中可选的元事件只展示包含该事件属性(分析主体)的元事件。

2) 选择聚合算子:聚合算子根据事件属性的数据类型有所不同

• 不同数据类型的事件属性支持的聚合计算规则汇总如下:

事件	事件属性 (数据类型)	聚合算子
任意事件		次数、天数、小时数
元事件	事件属性 (数值型)	总和、均值、中位数、最小值、最大值、去重数
元事件	事件属性 (布尔型)	为真数、为假数、为空数、不为空数、去重数
元事件	事件属性 (其他类型)	去重数

3) 设置筛选条件:支持对所选事件关联的事件属性、所有用户属性、用户标签、用户分群进行筛选,根据不同数据类型的筛选 项,筛选规则也有所不同

• 不同数据类型的筛选项支持的条件筛选规则汇总如下:

筛选项的数据类型	条件筛选可选项
文本型	等于、不等于、包括、不包括、有值、无值、正则匹配、正则不匹配
数值型	等于、不等于、小于、小于等于、大于、大于等于、有值、无值、 区间
时间型	位于区间、小于、小于等于、大于、大于等于、相对当前日期 (区间) 、相对当前日期(过去) 、有值、无值、等于、不等于
布尔型	为真、为假、有值、无值

2. 监控设置:包括监控粒度、监控时段、监控规则

- 监控粒度:包括按天和按小时两种
- 监控时段:与监控粒度对应,对应关系如下:
 当监控粒度为按天时,可选择的监控时段为周一到周日的一天或多天
 当监控粒度为按小时时,可选择的监控时段为周一到周日的一天或多天的一个或多个时段

• 监控规则:包括分组值和触发规则,可点击「+」按钮对监控规则进行新增

监控粒度	比较方式	和谁比	比什么
按天	高/低	上一天	数值/百分比
按天	高/低	上周同时刻	数值/百分比
按天	高/低	7日均值	数值/百分比
按天	高/低	固定值	数值
按小时	高/低	上一小时	数值/百分比
按小时	高/低	昨天同时刻	数值/百分比
按小时	高/低	24小时均值	数值/百分比
按小时	高/低	固定值	数值

分组值:与基本信息中所选择分组对应,默认为空(即总体);若未选择分组,则分组值只能为总体; 触发规则:根据不同的监控粒度,基于「比较方式」、「和谁比」、「比什么」的规则如下:

3. 通知方式

可选择系统通知或邮件通知,及时关注监控异常情况。

- 若选择系统通知,则默认在消息通知处通知此监控预警的创建人;
- 若选择邮件通知,可对通知对象进行选择。

4.3 监控详情

点击监控名称,可查看该监控预警的监控详情,包括左侧信息展示栏,右侧图表展示区,上方全局设置 1. 左侧:监控信息展示栏。展示监控名称、监控指标、分组、时间粒度、监控规则信息,置灰展示监控配置时信息,不允许修 改,默认展开,可收起,可拖拉,拖动范围为屏幕1/4到1/2。 2. 右侧:图表展示区,包括选择时间窗、切换规则、趋势图展示与表格展示 1)时间窗 点击时间窗,弹出时间选择框(与分析部分保持一致) 时间选择范围: 按小时预警:记录近一年数据,选取时最多只可选择展示一个月数据,当天可选择。 按天预警:记录近一年数据,选取时最多可选择展示近一年数据,当天不可选择。 2)趋势图展示:趋势图与所设置的规则对应,可切换规则展示不同的趋势图 3)表格展示:包括任务执行时间、监控情况、监控规则、分组项、指标值、对比值、较对比项差异值、较对比项差异率

- 任务执行时间:监控任务运行时间
- 监控情况:包括正常与异常,若出现非监控时段,则表明该任务执行时间不在所设置的监控时间段
- 监控规则:左侧监控任务设定区的规则
- 分组值:不同规则下对应的分组值
- 指标值:执行该监控任务时的指标值
- 对比值:左侧监控任务设定的对比项的值

- 较对比项差异值:即指标值-对比值结果
- 较对比项差异率:即(指标值-对比值)/对比值*100%
 注:当数据量过多时,表格默认展示前1000条数据
- 3. 上方:全局设置:包括时区展示和数据下载
- 1) 时区展示:为用户在时区设置配置好的默认项目时区
- 2) 若想下载全量数据,可点击下载按钮,将全量数据下载到本地

4.4 修改

点击「修改」按钮可对监控信息进行修改。

- 监控名称、监控状态、监控时段及通知方式修改不清空之前的数据
- 若进行分析主体、监控指标、分组、监控粒度、监控规则的修改,将清空历史数据

4.5 删除

点击「删除」按钮可将该监控预警删除。


管理中心

系统管理

协作者管理

帮助中心 > 功能手册 > 管理中心 > 系统管理 > 协作者管理

1 概述

协作者管理主要用于管理并展示租户下的所有成员的信息,可对此租户下的用户进行新增、修改、删除等操作,租户下成员包括所有项目内的成员,在对每个项目添加项目成员之前,需在此菜单先把成员录入加入租户。

2 位置和使用权限

操作位置:管理中心-系统管理-协作者管理

使用权限:此菜单为系统菜单,仅租户管理员拥有此菜单权限

3页面概览

协作者管理主要包括系统用户的列表展示及增删改查等功能

4 功能结构

4.1 列表展示

用户管理列表包括帐号、姓名、所属租户、系统角色、所属岗位、所属部门等信息。

4.2 新增

4.2.1 邀请协作者

可点击「邀请协作者」按钮将系统用户邀请进此租户下

4.2.2 创建协作者

可点击「创建协作者」按钮对用户进行单个新增

4.2.3 批量新增

当用户较多时,可点击「导入」按钮对用户进行批量导入,点击「点击下载」按钮下载我们所提供的模版,根据模版进行内容 填写,文件格式支持xls、xlsx、csv。

4.3 导出

可点击「导出」按钮将列表所有成员信息导出。

4.4 查看

可点击「查看」按钮查看该成员的详细信息。

4.5 编辑

可点击「编辑」按钮对用户的姓名、手机号码、邮箱、帐号状态、系统角色等信息进行修改。

4.6 删除

可点击「删除」按钮对该成员信息的删除,删除后数据将无法恢复。

角色管理

帮助中心 > 功能手册 > 管理中心 > 系统管理 > 角色管理

1 概述

角色管理主要用于管理并展示系统中的所有角色的信息,可对角色进行新增、修改、删除等操作,此角色主要控制系统菜单,即「系统管理」部分菜单的操作权限

2 位置和使用权限

操作位置:

管理中心-系统管理-角色管理

使用权限:此菜单为系统菜单,仅租户管理员拥有此菜单权限 注:租户管理员默认拥有全部项目的系统与项目权限,其他人系统与项目权限分开。

3 页面概览

角色管理主要包括系统角色的列表展示及增删改查等功能

4 功能结构

4.1 列表展示

角色管理列表包括系统角色名称、系统角色编码、所属租户、更新时间、更新人等信息。

4.2 新增

可点击「新增」按钮对角色进行新增,输入系统角色名称、系统角色编码、选择对应的系统菜单功能权限即可完成。

4.3 查看

可点击「查看」按钮查看该角色的详细信息。

4.4 编辑

可点击「编辑」按钮对该系统角色的名称、编码、权限进行修改。

4.5 删除

可点击「删除」按钮删除该角色信息。

部门管理

帮助中心 > 功能手册 > 管理中心 > 系统管理 > 部门管理

1 概述

部门管理主要用于管理并展示系统中的所有部门的信息,可对部门进行新增、修改、删除等操作

2 位置和使用权限

操作位置:

管理中心-系统管理-部门管理

使用权限:此菜单为系统菜单,仅租户管理员拥有此菜单权限

3页面概览

部门管理主要包括部门的列表展示及增删改查等功能。

4 功能结构

4.1 列表展示

部门管理列表包括部门名称、所属租户、部门全称、排序、备注、更新时间、更新人等信息。

4.2 新增

点击「新增」按钮可对部门进行新增,输入部门名称、部门全称、排序后点击确定即可完成新增。

4.3 查看

点击「查看」按钮可查看该部门的详细信息。

4.4 编辑

点击「编辑」按钮可对该部门的名称、全称、排序等信息进行修改

4.5 删除

点击「删除」按钮可对该部门信息进行删除。

4.6 添加子项

点击「添加子项」按钮可以该部门新增下级部门。

岗位管理

帮助中心 > 功能手册 > 管理中心 > 系统管理 > 岗位管理

1 概述

岗位管理主要用于管理并展示系统中的所有岗位的信息,可对岗位进行新增、修改、删除等操作

2 位置和使用权限

操作位置:管理中心-系统管理-岗位管理

使用权限:此菜单为系统菜单,仅超级管理员拥有此菜单权限

3 页面概览

岗位管理主要包括岗位的列表展示及增删改查等功能。

4 功能结构

4.1 列表展示

岗位管理列表包括岗位名称、所属租户、岗位类型、岗位编号、岗位排序、备注、更新时间、更新人等信息。

4.2 新增

点击「新增」按钮可对岗位进行新增,选择岗位类型、输入岗位编号、岗位名称、岗位排序后点击确定即可完成新增。

4.3 查看

点击「查看」按钮可查看该岗位的详细信息。

4.4 编辑

点击「编辑」按钮可对该岗位的岗位类型、输入岗位编号、岗位名称、岗位排序等信息进行修改。

4.5 删除

点击「删除」按钮可对该岗位信息进行删除。

项目管理

基本信息

帮助中心 > 功能手册 > 管理中心 > 项目管理 > 基本信息

1. 概述

基本信息主要用于描述项目的整体信息,包括租户ID、项目标识、成员数量等信息。

2. 位置和使用权限

操作位置:管理中心-项目管理-基本信息

使用权限:

项目角色	权限	权限明细
管理员	查看	查看项目基本信息
分析师	无权限	无权限
普通用户	无权限	无权限

成员管理

帮助中心 > 功能手册 > 管理中心 > 项目管理 > 成员管理

1 概述

成员管理主要用于管理项目内的成员,项目管理员可将系统内的用户直接加入项目,并为其分配角色,配置功能权限,各个项 目内的成员相互隔离。

2 位置和使用权限

操作位置:管理中心-项目管理-成员管理

使用权限:

项目角色	权限	权限明细
管理员	查看、新增、编辑	查看成员管理页面、新增、 编辑成员信息
分析师	无权限	无权限
普通用户	无权限	无权限

3页面概览

成员管理包括列表展示以及成员的新增、删除等功能。

4 功能结构

4.1列表展示

用户管理列表包括帐号、姓名、项目角色等信息。

4.2 新增

点击添加成员到项目,选择需要邀请的成员,并为每个成员分配角色,目前项目角色默认管理员、分析师、普通成员三种,当 项目角色与系统角色均分配时,以最高权限为准。

4.3 修改

可直接在项目角色栏对项目成员角色进行修改。

4.4 删除

点击「删除」可对该项目成员进行删除。

分析配置

分析主体

帮助中心 > 功能手册 > 管理中心 > 项目管理> 分析配置 > 分析主体

1 概述

分析主体是唯一用户的标识,通常由可以识别身份的字段担任,如设备ID、账号ID、角色ID等。若以设备ID作为分析主体,则将不同设备ID的用户视作不同用户;若以账号ID作为分析主体,则将不同账号ID的用户视作不同用户。

- 海纳ID (即user_id)是海纳嗨数的唯一用户标识ID,是默认的分析主体(推荐使用)。在分析应用中,可将事件属性或用户 属性作为分析主体以满足不同的业务需求。
- 分析主体可在漏斗分析、用户标签、用户分群等场景中使用。

2 位置和使用权限

功能位置:管理中心-项目管理-分析配置-分析主体

使用权限:项目管理员具有分析主体的管理权限,其他角色具有使用权限

项目角色	权限	权限明细
管理员	分析主体管理	新增、删除、使用分析主体
分析师	使用分析主体	在分析中使用已有的分析主体
普通用户	使用分析主体	在分析中使用已有的分析主体

3页面概览

4 功能结构

4.1 分析主体的配置

有分析主体管理权限的成员,点击分析主体管理页面的「新增」按钮,可以新增分析主体:

- 海纳ID是HINA云平台的默认分析主体,不可编辑和删除。
- 分析主体可以选择数据类型为数值型或字符型的属性。

4.2 分析主体的功能差异

使用海纳ID作为分析主体时,可以在「用户列表」和「用户详情」中查看更多信息,如查看用户的其他属性值辅助构建用户画像。当没有特殊使用场景时,推荐使用海纳ID作为分析主体。

- 使用默认分析主体(海纳ID):
- 使用自定义分析主体:

5 实例

分析主体的应用价值主要为:

- 使统计结果更准确。在分析和标签的计算逻辑中,需要提供身份识别字段做关联计算,不同的字段计算出来的结果可能会 不同,自定义分析主体可以使统计更灵活。
- 使统计维度更多样。用户的行为可能隐藏在不同的事件中,如想要了解不同科目老师的授课情况/不同店铺的经营情况,此时的分析主体切换为科目ID/店铺ID就可以得到基于不同科目/店铺的统计结果。

5.1 在分析模型中使用

在实际场景中,我们会需要在部分的分析场景里面切换分析主体。在漏斗分析中,如果我们关心的不是「用户」的行为转化漏 斗,而是某个商品,从入库、存库再到销售出去的漏斗模型,就可以将商品ID作为分析主体,分析各商品的销售流转路径。

5.2 在用户标签/分群中使用

在创建标签/分群时默认将海纳ID作为分析主体,也可选择其他分析主体。选择不同的分析主体其计算结果可能会不同,如使用 手机号或设备ID作为用户唯一标识,由于设备号和手机号一对多或多对一的情况,因此其圈选的人数和计算结果可能会存在差 异。

时区设置

帮助中心 > 功能手册 > 管理中心 > 项目管理> 分析配置 > 时区设置

1 概述

对于多时区的项目可开启多时区设置切换分析计算的时区,适用于有海外业务的项目。

首先介绍下几种时间的概念:

- 客户端时间:以用户当时客户端的时区作为依据,将物理时间(时间戳)转化成显示时间。如用户A在北京时间2月1日下午 6点做过事件1,用户B在纽约时间2月1日下午6点做过事件1,尽管它们发生的时间在物理时间上不一致,但是在分析时会 将两个时间当作同时发生。
- 服务端时间:以服务端人工配置一个固定的时区为依据,将物理时间(时间戳)转化成显示时间。如用户A在北京时间2月1 日下午6点做过事件1,用户B在纽约时间2月1日下午6点做过事件1,如果服务器时间为北京时间,则在上报时,会把用户B 的时间转换为北京时间上报,即用户B在北京时间2月1日下午18点做过事件1,用户A早于用户B发生该事件。

2 位置和使用权限

功能位置:管理中心-项目管理-分析配置-时区设置

使用权限:项目管理员具有时区的管理权限,其他角色具有使用权限

项目角色	权限	权限明细
管理员	设置、使用时区	开启/关闭时区,进行时区配置
分析师	使用时区	在分析中使用已有的时区
普通用户	使用时区	在分析中使用已有的时区

3页面概览

4 功能结构

4.1 时区的配置

1. 时区配置默认关闭,将数据上报的原始时间(即客户端时间)作为数据计算和展示的时间。

2. 若启用多时区,数据将以选择的时区计算并展示,选择不同的时区计算结果可能会不同。

- 3. 设置默认项目时区:可选择一个时区作为项目的默认时区,所选时区将作为计算和展示时的默认选择时区,时区可选择范 围为当地时间、UTC-12~UTC+14当中的任一个时区。
- 4. 设置可展示时区:选择项目成员在计算和分析时可配置的时区,选中的时区将在分析中选择时区的下拉框中展示。

注1: 若要开启多时区配置,请检查系统属性「时区@timezone_offset」是否上报且有数据,否则不建议开启该功能。

注2: 单时区项目不建议开启多时区配置;多时区项目若已开启多时区配置不建议关闭,关闭后,已有的分析和报表展示可能会受到影响。

注3: 开启多时区后,若选择当地时间作为计算时区,则没有时区转换的概念(相当于关闭多时区配置),即以客户端时间作为 计算和展示的时间。

4.2 时区的计算规则

时区转换的计算公式为:展示时间 = 事件时间+ (展示时区 - 事件时区)

如某用户在1月1日中午12点发生了事件A,该事件的时区为UTC+8,如果在分析时设置计算和展示的时区为UTC+10,则该事件的展示时间为12+(10-8)即14点;如果设置计算和展示的时区为UTC-1,则该事件的展示时间为12+(-1-8)即3点。

4.3 切换时区的局限性

- 1. 由于只有事件带有系统属性「时区@timezone_offset」,因此时区转换只针对事件,用户属性不进行时区偏移转换。
- 2. SQL查询的计算不设置时区,可在SQL语句中自行转换。

5 实例

多时区配置的应用价值主要为:

- 统一不同时区的数据,减小计算误差。多时区配置可将不同时区上报的事件时间进行统一转换,将所有事件时间都偏移到 相同时区,避免数据跨天汇总统计不准确等问题。
- 符合项目成员的查看习惯。若某公司在北京、伦敦、东京等地区均有分部,中国的用户可以将时区设置为UTC+8,美国的用 户可将时区设置为UTC-4,日本的用户可将时区设置为UTC+9,可以保留不同地区成员的分析习惯。
- 1. 分析和计算标签等场景中涉及到的时间条件,均是基于时区转换后的时间进行统计汇总,保证数据的准确性。
- 分析和计算标签等场景均需要设定数据的时间窗(如过去7天)和展示的时间粒度(如按天/月/周汇总)。开启多时区功能后,会先对事件发生的时间进行时区转换,再进行统计汇总,统计汇总的结果会在分析或看板中展示。

消息通知

消息助手

帮助中心 > 功能手册 > 管理中心 > 消息通知 > 消息助手

1 概述

为了帮助您能及时接收系统通知、预警通知、空间动态和资产动态等消息,我们提供了「通知助手」的功能,仅支持查看近30 天前1000条消息。

2 位置和使用权限

操作位置: 消息助手

使用权限: 不同的项目角色操作权限如下

项目角色	权限	权限明细
管理员	查看	查看
分析师	查看	查看
普通用户	查看	查看

3页面概览

消息助手主要包括展示全部消息、展示未读消息、全部标为已读、查看所有消息等功能。

4 功能结构

4.1 消息类型

消息类型	消息内容
系统通知	产品迭代、通知公告
预警通知	监控预警
空间动态	共享空间、停止共享空间、添加空间协作者、取消添加空间协作者
资产动态	自建的用户标签/用户分群被他人修改/删除、自建的用户标签/ 用户分群的回溯版本管理中的标签/分群数据被清理

4.2 页面操作

1.点击「未读」直接展示未读消息,可以全部标为已读消息 2.点击「查看所有消息」后进入消息中心

消息中心

帮助中心 > 功能手册 > 管理中心 > 消息通知 > 消息中心

1 概述

消息中心旨在帮助您查看所有消息,包括系统通知、预警通知、空间动态、资产动态等消息。

2 位置和使用权限

操作位置:管理中心-消息通知-消息中心

使用权限: 不同的项目角色操作权限如下

项目角色	权限	权限明细
管理员	查看	查看
分析师	查看	查看
普通用户	查看	查看

3页面概览

4 功能结构

4.1 列表展示

(1) 切换不同消息类型可以查看不同类型的消息

(2) 点击「全部消息」展示该消息类型下的全部消息,点击「未读消息」展示该消息类型下的未读消息,可以批量/全部标记 为已读消息

操作视频

功能使用

数据空间

数据空间



数据分析

事件分析



漏斗分析



间隔分析



留存分析



属性分析



网页热力分析



SQL查询与可视化



用户洞察

用户标签



用户分群



数据中心

元数据管理



监控预警



管理中心

用户与权限



技术文档

基础知识

海纳系统提供了全端数据接入方案:

- 客户端 SDK
- 服务端 SDK
- 数据导入

总体接入流程

您需要数据接入时,并实现数据采集并分析。全流程一般为三个步骤: 第一步:首先租户对接我们的数据分析师,根据实际业务需求一起整理并梳理出精准的数据采集方案; 第二步:根据采集方案以及SDK或者数据导入接入文档,由租户的研发人员根据数据采集方案完成数据接入工作; 第三步:通过海纳分析平台,验证数据接入正确性。

全端接入方案

客户端 SDK:可以采集到设备信息,不给租户服务器增加压力,接入简单易用。 服务端 SDK:适合采集核心业务数据,采集内容更加精准,方便租户升级迭代。 数据导入工具:通常用于历史数据导入。

标识用户

标识用户,是指选择一个合适的标识符(例如设备 ID 或者注册 ID)作为 distinct_id 来发送数据到海纳营销云。选择合适的 distinct_id 对数据分析的准确性会有很大影响,因此,在进行任何数据接入之前,都应当先确认您的用户标识方式。海纳分析提供了灵活、强大的用户标识能力,您可以根据自己的需求来选择合适的方案,具体请阅读文档标识用户。如果您依然不确定如何进行用户标识,请联系及时联系海纳的数据分析师。

客户端埋点

方案一:直接使用海纳客户端 SDK。这个方案相对简单、易用,并且海纳 SDK 提供了可靠性保证(例如网络不好的情况下延迟 发送)。一般情况下,我们建议采用此方案。

方案二:使用已有的业务 API,把埋点需要的数据同步传到业务服务器,然后在服务端再使用神策的服务端 SDK (Java SDK) 进行接入。这个方案本质上其实是服务端埋点,优点是对于业务统计可能会更加准确(因为和业务调用是同步的),安全性比 较高(可以进行一定的客户端加密来增大伪造数据的难度),缺点是实施难度较大。我们一般建议对于关键的业务事件(例如 购买、支付等)采用这种方案。

服务端埋点

不管是客户端的埋点数据通过 API 发送给服务端之后,还是直接在服务端的已有业务逻辑里直接埋点,都属于服务端接入。服务端接入可以使用 服务端 SDK,且在一般情况下,我们都建议在服务的入口处(例如 MVC 的 Controller 层)进行埋点,这样既能获取到大部分埋点所需要的数据,又方便统一管理,每个 SDK 均有不同类型的发送方案(即 Consumer)可以选择,有两大 类方案:

方案一:使用直接发送数据的 Consumer (例如 Java 的 AsyncBatchConsumer) ,实时的发数据给神策的服务。优点是方便简 单,缺点是在机器故障或者超大流量的极端情况下可能会丢失一小部分数据,并且可能对业务造成一定影响。如果数据量不大 可以使用此方案。

方案二:使用写入本地日志的 Consumer(例如 Java 的 LoggingConsumer),配合 LogAgent 进行导入。优点是因为有本地持久 化,可靠性会更高,缺点是代码会稍微复杂,同时还需要自己负责本地日志的存储和删除等运维操作。建议在大数据量或者对 数据准确性有高要求的时候使用此方案。

接入流程

帮助中心 > 技术手册 > 基础知识 > 接入流程

海纳嗨数平台提供了全端数据接入方案:

- 客户端 SDK
- 服务端 SDK
- 数据导入

1. 总体接入流程

当您需要数据接入,实现数据采集及分析时,一般需要以下三个步骤: 第一步:首先租户对接我们的数据分析师,根据实际业务需求一起整理并梳理出精准的数据采集方案; 第二步:根据数据采集方案和相关需求,按需要进行具体的接入工作,可使用SDK埋点或者数据导入方式,由租户的研发人员根 据数据采集方案完成数据接入; 第三步:通过海纳嗨数平台,进行数据测试和验证,完成验证后即可上线到正式环境。

2. 全端接入方案

客户端 SDK:可以采集到设备信息,不给租户服务器增加压力,接入简单易用。 服务端 SDK:适合采集核心业务数据,采集内容更加精准,方便租户升级迭代。 数据导入工具:通常用于历史数据导入。

2.1 标识用户

标识用户,是指选择一个合适的标识符(例如设备 ID 或者注册 ID)作为 distinct_id 来发送数据到海纳嗨数云。选择合适的 distinct_id 对数据分析的准确性会有很大影响,因此,在进行任何数据接入之前,都应当先确认您的用户识别方式。海纳嗨数提 供了灵活、强大的用户识别能力,您可以根据自己的需求来选择合适的方案,具体请阅读文档 用户识别。如果您依然不确定如 何进行用户识别,请联系及时联系海纳嗨数的数据分析师。

2.2 客户端埋点

方案一:直接使用海纳嗨数客户端 SDK。这个方案相对简单、易用,并且海纳嗨数 SDK 提供了可靠性保证(例如网络不好的情况下延迟发送)。一般情况下,我们建议采用此方案。

方案二:使用已有的业务 API,把埋点需要的数据同步传到业务服务器,然后在服务端再使用海纳嗨数的服务端 SDK (Java SDK) 进行接入。这个方案本质上其实是服务端埋点,优点是对于业务统计可能会更加准确(因为和业务调用是同步的),安 全性比较高(可以进行一定的客户端加密来增大伪造数据的难度),缺点是实施难度较大。我们一般建议对于关键的业务事件 (例如购买、支付等)采用这种方案。

2.3 服务端埋点

不管是客户端的埋点数据通过 API 发送给服务端之后,还是直接在服务端的已有业务逻辑里直接埋点,都属于服务端接入。服 务端接入可以使用 服务端 SDK,且在一般情况下,我们都建议在服务的入口处(例如 MVC 的 Controller 层)进行埋点,这样既 能获取到大部分埋点所需要的数据,又方便统一管理,每个 SDK 均有不同类型的发送方案(即 Consumer)可以选择,有两大 类方案:

方案一:使用直接发送数据的 Consumer(例如 Java 的 AsyncBatchConsumer),实时的发数据给海纳嗨数的服务。优点是方 便简单,缺点是在机器故障或者超大流量的极端情况下可能会丢失一小部分数据,并且可能对业务造成一定影响。如果数据量 不大可以使用此方案。

方案二:使用写入本地日志的 Consumer(例如 Java 的 LoggingConsumer),配合 LogAgent 进行导入。优点是因为有本地持久

化,可靠性会更高,缺点是代码会稍微复杂,同时还需要自己负责本地日志的存储和删除等运维操作。建议在大数据量或者对 数据准确性有高要求的时候使用此方案。

2.4 数据导入

数据格式

帮助中心 > 技术手册 > 基础知识 > 数据格式

本章节将会详细介绍海纳嗨数支持的数据结构、数据类型以及数据限制。通过本章节,您将了解如何构建符合规则的数据,排查数据传输问题。

如果您使用的是 数据导入 或 RESTful API 上传数据,需要按照本章节中的数据规则对数据进行格式处理。

1. 数据结构

海纳嗨数接受的是符合规则的 JSON 数据:如果使用的是 SDK 接入,则数据将会被转化成 JSON 数据进行传输。如果使用导入工具方式上传数据,则数据需要是符合规则的 JSON 数据。

JSON 数据以行为单位:即一行一条 JSON 数据,对应物理意义上的一条数据,数据意义上对应的是用户产生一次行为,或者是 设置一次用户属性。

1.1 事件结构

事件记录: 记录一个 Event 及关联的 Properties。

```
{
    "distinct_id": "F77C4B19-6E05-4ED3-B415-348A3E34BD97",
    "event_time": "2022-08-09 15:12:33",
    "type": "track",
    "event": "app_start",
    "properties": {
       "H_app_version":"1.3",
        "H_wifi":true,
       "H_ip":"252.79.22.65",
       "H_province":"上海市",
       "H_city":<mark>"上海市"</mark>,
       "H_lib_method": "code",
       "H_os_version": "13.7",
       "H_lib": "iOS",
       "H_manufacturer": "Apple",
       "H_lib_version": "4.4.0",
       "H_os": "iOS"
        "event_duration": 0,
        "product_name":"苹果",
        "resume_from_background": true,
 }
}
```

对于上述字段的说明如下:

- distinct_id:类型是字符串,对用户的标识,对未登录用户,可以填充设备标识、CookieID等,对于登录用户,则应该填充 注册账号;这里的例子,我们假设是一个登录用户,所以填充的是一个用户账号;
- event_time:类型是时间,事件发生的实际时间精确到秒;
- type:track 表明是记录一个 Event ,这里我们假设是APP启动行为;
- event:事件名,需是合法的变量名,不能以特殊字符或数字开头,且只包含:大小写字母、数字、下划线,其中以 H_ 开 头的表明是系统的预置事件,自定义事件名请不要以 H_ 开头;

- properties:这个 Event 的具体属性,以 dict 的形式存在。其中以H_开头的表明是系统的预置属性,它的类型和中文名已经 预先定义好了。自定义属性名需要是合法的变量名,不能以特殊字符或数字开头,且只包含:大小写字母、数字、下划 线,自定义属性不能以 H_开头;同一个名称的 property,在不同 event 中,必须保持一致的定义和类型;字段名不区分大 小写。
 - H_app_version:用户所使用的 App 的版本;
- ٠
- H_wifi:这条事件发生时,用户是否在使用 wifi;
- •

.

- H_ip:用户使用设备的 IP。若数据中出现H_ip,且数据中没有 H_province 或 H_city 字段,将使用该 IP 解析出省市信 息填入缺失字段;
- H_province、H_city:省、市,在没有填充这两个字段的时候,会根据 IP 进行解析;

event_duration、product_name、resume_from_background:设置一些自定义事件属性,前提必须先在元事件管理中配置,否则则会抛弃上送字段,不入库处理。

登录事件记录:

这个接口是一个登录功能(将设备ID与登录账户绑定),可在业务登录时或者打开APP时判断是否登录调用该事件。



这条JSON数据表示,一个匿名 ID 为 F77C4B19-6E05-4ED3-B415-348A3E34BD97 的用户,成功完成了注册,注册后的注册 ID 是 zhangsan。并且系统后台,会将 original_id 为 F77C4B19-6E05-4ED3-B415-348A3E34BD97 的用户和 distinct_id 为 zhangsan 的用户,代表匿名ID与注册ID进行绑定。

1.2、用户结构

Profile 相关操作,主要是用来设置用户属性的,提供了如下一系列接口:用户属性记录:

{
 "distinct_id": "zhangsan",
 "type": "profile_set",
 "event_time": "2022-08-09 16:09:09",
 "properties": {
 "H_province":"湖南",
 "name": "小红",
 "age":33,
 }
}

```
"Gender":"男",
"H_signup_time": "2015-06-26 11:43:15.610"
}
}
```

直接设置一个用户的profile_set_once 与 profile_set 接口不同的是,如果用户表对应该字段值已经存在,上送的值不会覆盖原来 的值,如果 不存在,则会自动创建。因此,profile_set_once 比较适用于为用户设置首次激活时间、首次注册时间等只在首次设 置时有效的属性。

属性值加减(profile_increment)

```
{
    "distinct_id": "zhangsan",
    "type": "profile_increment",
    "time": "2022-09-09 16:33:21",
    "properties": {
        "level": 1
    }
}
```

增加或减少一个用户的某个 数值 类型的属性值,比如给用户属性 level 的值加 1 或者减 1 。如果用户表(users)中不存在这个用 户,则会在用户表中自动创建该用户的 id 记录,并给该用户设置相应的属性值,会在默认值 0 的基础上增加上传的值。

1.3 属性数据类型

分析属性数据类型定义

数据类型	类型名称	限制	备注
DECIMAL	数值	小数点后最多保留3位	如:12,12.34
VARCHAR	文本	使用 UTF-8 编码后最大长度 1024 字节, 如需调整最大长度可以联系分析师	如:"zhangsan","city"
DATETIME	日期时间	yyyy-MM-dd HH:mm:ss.SSS 或 yyyy- MM-dd HH:mm:ss 或 yyyy-mm-dd (时分秒按00:00:00处理)	
BOOLEAN	布尔		如: true,false, 对应数据库0,1

预置属性

为了帮助客户能够更方便地使用我们的产品,我们目前分别为 事件 和 用户 提供了一些预置字段,详细查看预置属性与系统字段。

预置属性的作用:

- 预置属性会在SDK中自动采集,也可以手动设置关闭。
- 海纳嗨数系统会基于部分预置属性做业务逻辑处理,比如预置属性H_IP,如果前端没上送,则海纳嗨数系统会取请求中的IP 填充。
• 部分预置属性用于全埋点过程中自动采集,可以手动设置相关预置属性采集关闭。

保留字段

为了保证自定义属性不与系统变量名冲突,设置属性时清规避系统字段,系统字段如下:

user_id distinct_id account_id update_time part_date event event_time server_time timezone_offset

用户识别

帮助中心 > 技术手册 > 基础知识 > 用户识别

1. 用户识别

用户识别是分析中一个重要的组成部分,在分析用户之前我们必须首先能够识别每个用户,分辨出哪些是新用户,哪些是留存 用户。这样不但能够更加清晰地了解到底有多少用户访问了你的网站,分辨他们是谁(用户ID、邮箱、性别年龄等);同时也能 够帮助你更好地跟踪你的用户,发现它们的行为特征,兴趣爱好及个性化的设置等,以便于更好地把握用户需求,提升用户体 验。在实际业务中用户可能含有多个ID体系,通过ID-Mapping(用户融合和识别)可以在同一个主体上建立统一的标签体系,然 后基于此进行人群圈选或洞察,还原用户真实画像,以便于能够在分析中数智化洞察用户的的行为及业务特征,总结规律,然 后再实际运营中为不同分层用户设计针对性的服务与营销策略。那么海纳是怎么设计识别用户的呢?

1.1 用户识别 ID

海纳嗨数主要使用三个字段进行用户识别

- 设备 ID (distinct_id):不一定为设备的唯一标识,一个设备ID只会指向一个设备,但一个设备会有多个设备ID,我们通过设备注册服务获取到的设备信息会通过客户端sdk在设备本地进行存储。
- 账号 ID (account_id) :客户业务系统中用户唯一标识,一般情况直接使用产品业务中使用的用户标识,比如登录账号。
- 用户 ID(user_id):海纳嗨数分析平台为用户/设备自动分配的唯一标识,并与各个平台的账号ID相关联,以实现对匿名和 实名状态用户的统一标识。

1.1.1 设备 ID

设备 ID 即匿名ID,用户未登录情况下,SDK会自动生成一个唯一标识来,识别用户,当用户登录后SDK会自动将设备ID 设置为 账户ID。

1.1.2账号 ID

账号 ID 指的是用户在客户平台上注册生成的唯一编号,后续的客户相关行为操作都通过该唯一编号进行关联用户相关信息。

1.1.3用户 ID

用户 ID是海纳嗨数内部用来识别用户的唯一标识 ID。所有入库数据,系统都会根据"账号 ID"与"设备 ID"生成该条数据的"用户 ID",以标识用户属性和行为数据归属。

1.2 用户识别方案详解

在不同的业务或平台导致,用户标识使用不一样,用户场景主要归类为三种情况:

- A、只使用设备 ID
- B、关联设备 ID 和账号 ID (一对一)
- C、关联设备 ID 和账号 ID (多对一)

1.2.1 方案一:只使用设备 ID

1.2.1.1 适用场景

平台没有用户注册体系,或者极少数用户会进行多设备登录的产品,如工具类产品、搜索引擎等;用户跨设备使用或者多用户 共用设备不是产品的常见场景的。 1.2.1.2 局限性

- 同一用户在不同设备使用会被认为不同的用户(设备ID不同,对应user_id也不一样),导致出现真实客户为一个,但是系统识别出多个用户出来,影响后续的用户分析统计(即用户换新手机,新手机的操作行为无法关联之前手机行为数据)。
- 不同用户在相同设备使用会被认为是一个用户(设备ID相同,对应user_id也一样),导致出现多个真实用户,被系统识别成一个用户,影响后续的用户分析统计有影响(即用户把手机送给朋友,但朋友的操作行为仍为该用户下)。

1.2.1.3 案例分析

流程说明:

- 1. 安装APP:张三用户在华为手机上新安装了 App,并进入APP进行操作,集成到APP中的SDK会基于设备信息生成设备 ID 为 A,上送的 distinct_id 为 A,海纳嗨数会生成user_id为u01,若客户端SDK 调用接口 profile_set方法 ,则将设备 ID A和 user_id 存入 users 表的 user_id, distinct_id 字段中。
- 2. 登录APP:张三在华为手机上注册并登录,由于是原来的手机设备,发送的 distinct_id 仍为 A,所以user_id 仍为 u01。
- 3. 操作APP:张三登录之后继续进行一系列操作,发送的 distinct_id 仍 为 A,所以user_id 仍为 u01。
- 4. 退出APP:张三退出登录并进行了一系列操作,发送的 distinct_id 仍为 A, 所以user_id 仍为 uO1。
- 5. 登录APP: 张三把华为手机送给朋友李四了,李四用自己的账号登录,由于手机设备没变,所以发送的 distinct_id 仍为 A, user_id 仍为 u01。
- 6. 操作APP: 李四使用账号 B 在该华为手机上操作,设备 A 上进行了一系列操作,由于设备未变,所以发送的 distinct_id 仍为 A, user_id 仍为 u01。
- 7. 安装APP:张三更换了一个苹果手机,并在APP上进行操作,由于设备 ID 变为 B,发送的 distinct_id 为 B,分配的user_id 为 u02,若调用接口 profile_set方法,则将设备 ID为B和 user_id为u02 存入 users 表的 user_id, distinct_id 字段中。
- 8. 登录APP:张三在苹果手机上使用账号 A 进行登录,由于还是苹果手机设备,发送的 distinct_id 为 B,所以对应user_id 仍 为 u02。
- 9. 操作APP:张三该登录之后的后续操作,由于还是苹果手机设备,发送的 distinct_id 为 B,所以对应user_id 仍为 u02,只 要不更换设备。

1.2.2 方案二:设备 ID 和登录 ID (一对一)

使用设备 ID 标识用户虽然简单,但是对于某些应用场景这种方式不够准确,因此海纳嗨数SDK提供了"设备ID"与"登录ID"的方案,从而解决方案一的局限性,从而更准确的进行用户追踪分析。

1.2.2.1 适用场景

- "设备ID"和"登录ID"绑定后,用户在该设备ID上或该登录ID下的行为就会进行聚合,被认为是同一个用户行为数据。在进行 海纳嗨数用户分析是,更加精准。
- "设备ID" 和"登录ID"绑定实现了更准确的用户追踪,需要增加埋点接入的复杂度。所以需要根据匹配的场景选择适当的方案。

1.2.2.2 局限性

- 一个设备 ID 只能和一个登录 ID 绑定,而事实上一台设备可能有多个用户使用(比如把手机送朋友)。
- 一个登录 ID 只能和一个设备 ID 绑定,而事实上一个用户可能用一个登录 ID 在多台设备上登录(比如用户新买手机)。

1.2.2.3 案例分析

流程说明:

- 1. 安装APP:张三用户在华为手机上新安装了 App,并进入APP进行操作,集成到APP中的SDK会基于设备信息生成设备 ID 为 A,上送的 distinct_id 为 A,海纳嗨数会生成user_id为u01,若客户端SDK 调用接口 profile_set方法 ,则将设备 ID A和 user_id 存入 users 表的 user_id, distinct_id 字段中。
- 2. 登录APP:张三在华为手机上注册并登录,由于是原来的手机设备,这里会调用 SDK 的 login(客户端)或 track_signup 接口(服务端),设备ID(A)和登录ID[zhangsan] 关联成功,那user表中account_id 为zhangsan,distinct_id 为 A,user_id依然

为步骤1内部生成的海纳ID。

- 3. 使用APP:张三登录之后继续进行一系列操作,发送的 distinct_id 仍为 A,所以user_id 仍为 u01。
- 4. 退出APP:张三退出登录并进行了一系列操作,发送的 distinct_id 仍为 A,所以user_id 仍为 u01。
- 5. 登录APP: 张三把华为手机送给朋友李四了,李四用自己的账号登录,由于手机设备没变,在登录过程中设备ID为A尝试与 李四进行关联绑定,由于设备ID已经绑定了用户张三,所以李四绑定失败,系统重新创建一个用户lisi,那user表中 account_id 为lisi, distinct_id 为 lisi, user_id内部生成的海纳ID[u02]。
- 6. 使用APP: 李四使用账号 B 在该华为手机上操作,设备 A 上进行了一系列操作,由于该设备已经标记为lisi,所以发送的 distinct_id 为 lisi, user_id 仍为 u02。
- 7. 使用APP:张三更换了一个苹果手机,并在APP上进行操作,由于设备 ID 变为 B,发送的 distinct_id 为 B,分配的user_id 为 u03,若调用接口 profile_set方法 ,则将设备 ID为B和 user_id为u03 存入 users 表的 user_id, distinct_id 字段中。
- 8. 登录APP:张三在苹果手机上使用账号zhangsan进行登录,由于账户zhangsan已经在用户表中,发送的 distinct_id 为 B,但 是已存在用户,所以对应user_id 为 u01。
- 9. 使用APP:张三该登录之后的后续操作,由于用户名没变,所以对应user_id 仍为 u01。

1.2.3 方案三:设备 ID 和登录 ID (多对一)

1.2.3.1 适用场景

客户系统平台有多个客户端(PC端、iOS端、安卓端等),就会出现一个用户可能会在多个客户端设备上进行一系列操作,海纳 嗨数分析系统支持一个登录ID绑定多个设备,多个设备的一系列操作会归纳为一个海纳ID(user_id)。

1.2.3.2 局限性

一个设备ID一旦跟登录ID绑定,就不能进行解绑,在其他用户未登录前的一些列操作都会归类为绑定的登录ID,所以在海纳嗨数 分析会出现微量的误差。

1.2.3.3 案例分析

流程说明:

- 1. 安装APP:张三用户在华为手机上新安装了 App,并进入APP进行操作,集成到APP中的SDK会基于设备信息生成设备 ID 为 A,上送的 distinct_id 为 A,海纳嗨数会生成user_id为u01,若客户端SDK 调用接口 profile_set方法,则将设备 ID A和 user_id 存入 users 表的 user_id, distinct_id 字段中。
- 2. 登录APP:张三在华为手机上注册并登录,由于是原来的手机设备,这里会调用 SDK 的 login (客户端)或 track_signup 接口(服务端),设备ID(A)和登录ID[zhangsan]关联成功,那user表中account_id为zhangsan,distinct_id为A,user_id依然为步骤1内部生成的海纳ID。
- 3. 使用APP:张三登录之后继续进行一系列操作,发送的 distinct_id 仍为 A, 所以user_id 仍为 u01。
- 4. 退出APP:张三退出登录并进行了一系列操作,发送的 distinct_id 仍为 A,所以user_id 仍为 u01。
- 5. 登录APP: 张三把华为手机送给朋友李四了,李四用自己的账号登录,由于手机设备没变,在登录过程中设备ID为A尝试与 李四进行关联绑定,由于设备ID已经绑定了用户张三,所以李四绑定失败,系统重新创建一个用户lisi,那user表中 account_id 为lisi, distinct_id 为 lisi, user_id内部生成的海纳ID[u02]。
- 6. 使用APP: 李四使用账号 B 在该华为手机上操作,设备 A 上进行了一系列操作,由于该设备已经标记为lisi,所以发送的 distinct_id 为 lisi, user_id 仍为 u02。
- 7. 使用APP:张三更换了一个苹果手机,并在APP上进行操作,由于设备 ID 变为 B,发送的 distinct_id 为 B,分配的user_id 为 u03,若调用接口 profile_set方法 ,则将设备 ID为B和 user_id为u03 存入 users 表的 user_id, distinct_id 字段中。
- 8. 登录APP:张三在苹果手机上使用账号zhangsan进行登录,由于账户zhangsan已经在用户表中,发送的 distinct_id为B,且 张三已是存在的用户,所以对应user_id为 u01,但设备B为绑定用户,所以该设备B关联在zhangsan用户下,导致设备B对 应的user_id为u03,设置在关联设备用户ID集合(associated_user_id_list)字段下,且将用户为u03的用户记录删除,在数据 分析时会将关联设备用户行为数据归类为绑定用户上。
- 9. 使用APP:张三该登录之后的后续操作,由于用户名没变,所以对应user_id 仍为 u01。

1.2.4 总结

三种方案并没有那个更好,建议租户结合自身应用场景选择对应方案。

预置属性与系统字段

帮助中心 > 技术手册 > 基础知识 > 预置属性与系统字段

预置属性与系统字段

• A、预置属性指的是由海纳嗨数生成或取得的属性,所有的预置属性以"H_"开头,总共分为两类:用户属性、事件属性,这些属性已定义其说明和含义。

预置属性与事件详细查看:预置事件与属性总表.xlsx

• B、系统字段指的是数据库中具有特殊用途的字段,所以自定义属性时不能出现与预置属性和系统字段重名的情况,否则视为非法字段,不能入库。

 	-	125		

系统字段:

属性英文	变量名	属性值类型
part_date	日期	DATE
event	事件名	VARCHAR
event_time	日期时间	DATETIME
server_time	服务器时间	DATETIME
timezone_offset	时区	INT
user_id	用户ID	VARCHAR
distinct_id	设备ID	VARCHAR
account_id	账号ID	VARCHAR
update_time	更新时间	DATETIME

注:

- 1、H_is_first_time 字段只有 App 端的 H_AppStart 、JS 端的 H_pageView 和小程序端的H_MPLaunch事件有该属性,其他 事件中默认没有该属性,但是允许客户自己上传,客户自己填值。
- 2、H_is_first_day 字段 App 端和 JS 端所有事件都会自动采集。

采集上报地址获取

帮助中心 > 技术手册 > 基础知识 > 采集上报地址获取

1. 登录海纳嗨数平台

平台地址:https://hicloud.hinadt.com

2. 按照下图步骤获取



客户端SDK

合规说明

隐私声明

帮助中心 > 技术文档 > 客户端SDK > 合规说明 > 隐私声明

海纳嗨数 SDK 隐私声明 版本号:V1.0 发布时间:2023年【02】月【28】日 生效时间:2023年【02】月【28】日 联系电话:400-667-9660 联系邮箱:service@hinadt.com 注册地址:上海市徐汇区桂林路656号5层5205室

海纳致远数字科技(上海)有限公司及其关联公司(以下简称"海纳数科"、"我们")作为海纳嗨数 SDK 的开发和运营者,深知用 户信息的重要性,尊重并保护用户个人信息安全。为帮助集成海纳嗨数 SDK 的产品和/或服务的开发者和运营者(以下简称"开 发者")在符合相关法律法规、政策及标准的规定下开展第三方 SDK 集成业务,更好的落实用户个人信息保护相关要求,同时帮 助使用集成海纳嗨数 SDK 的产品和/或服务的终端用户(以下称"终端用户")充分了解开发者使用海纳嗨数 SDK "收集、使用、 保存、共享、转让"终端用户个人信息的目的、方式、范围和信息安全保护措施,特制定《海纳嗨数 SDK 隐私声明》(以下简称 "本声明")。

本声明适用于我们通过海纳嗨数 SDK Android 版、海纳嗨数 SDK iOS 版以及我们不时提供或更新的其他版本向您提供的产品和 服务。如果某一版本有特殊功能或者个人信息处理活动与本声明不一致的情况,我们会予以说明。如果我们的产品和/或服务单 独设置了隐私政策/隐私协议的,单独的隐私政策/隐私协议优先于本声明适用;单独的隐私政策/隐私协议未提及的内容,适用 本声明。

开发者和终端用户(简称"您")接入、使用海纳嗨数 SDK 产品和/或服务前,请务必仔细阅读本声明,并确认您已经阅读并且充 分理解本声明的所有内容,特别应重点阅读我们以"黑体/斜体/下划线"标识的涉及个人信息权益保护的重要条款,确保您充分理 解和同意之后再开始接入、使用海纳嗨数 SDK 提供的产品和/或服务。如果您不理解本声明的部分或全部内容,建议您通过拨 打客服电话 400-667-9660 与我们联系,进行相关咨询。在您充分理解本声明的基础上,仍然不同意本声明条款的,建议您立 即停止接入和使用海纳嗨数 SDK 提供的产品和/或服务。

请您确保,您是18周岁(含)以上人士。如您是18周岁以下的未成年人,应由您的监护人陪同阅读本声明,并经由监护人同意 本声明后,您才能使用海纳嗨数 SDK 提供的产品和/或服务并向我们提供您的个人信息。

特别提示,如果您是开发者,您应当:

1. 您应确保在App首次运行时通过明显方式(弹窗)提示终端用户阅读您的《隐私政策》,并取得终端用户的合法授权后,再初始化SDK进行信息收集与处理。如果终端用户不同意您的隐私政策,则不能初始化海纳嗨数的各项SDK,也无法使用相应SDK对应功能。

如果您的App在终端用户首次运行时,需要注册用户账号才能使用,则可以在账号注册环节提示终端用户同意您的《隐私政 策》,之后完成注册;如果您的App并不一定需要终端用户注册用户账号才能使用,那么如果终端用户不同意您的《隐私政 策》,按照最新合规政策,您不应停止让终端用户使用您的App,仍需保留用户的基本使用权利。若终端用户将使用到需收集相 关个人信息才能使用的功能,可以再次提醒终端用户需要同意您的《隐私政策》才能正常使用相关功能,若终端用户仍不同 意,则无法提供对应功能,可在终端用户下次需要使用时再提示同意您的《隐私政策》。 2.在使用海纳嗨数各项SDK产品时,开发者需在App《隐私政策》的"与授权合作伙伴共享"条款中,将《海纳嗨数SDK隐私声 明》加入其中,并向终端用户逐一明示您嵌入的SDK收集使用个人信息的目的、方式和范围。 开发者在APP自身隐私政策中添加海纳嗨数各项SDK的隐私政策文本模版如下: SDK名称:海纳嗨数SDK【按照开发者使用的海纳嗨数具体SDK名称填写】 第三方名称:海纳致远数字科技(上海)有限公司 使用目的:【请开发者根据具体使用目的填写】 收集个人信息:设备信息(包括IMEI、Android ID、IDFA、IDFV、OAID、UUID、Mac地址、IMSI信息等)、日志信息(包括IP地 址、访问服务的URL、浏览器类型和使用语言等)、位置信息、唯一应用程序编号。(此处为示例:基于不同的设备和系统(安 卓/iOS)及系统版本,以及开发者在集成、使用海纳嗨数 SDK 产品时决定的权限,收集的设备信息会有所不同,因此开发者应 对实际收集的个人信息向用户进行说明。) 隐私政策:【嵌入本隐私声明超链接】 3.遵守本声明的相关要求。 特别提示,如果您是终端用户,您应当: 鉴于您不是与海纳数科存在直接合同关系的开发者,而是安装了经有效注册海纳嗨数账号并集成了海纳嗨数SDK的开发者所开发 或者运营的应用的设备持有者,我们会要求开发者通过隐私政策或服务协议等适当的方式使您知晓其应用集成了海纳嗨数服务 功能模块,并确保您知悉且同意开发者获取和使用您的用户相关信息的权利延展至海纳嗨数,使我们可以获取和使用实现相关

服务功能所必要的合理信息。

本声明将帮助您了解以下内容:

- 一、我们如何收集和使用终端用户的个人信息
- 二、 我们如何使用 Cookie 和同类技术
- 三、 我们如何共享、转让、公开披露终端用户的个人信息
- 四、我们如何保存终端用户的个人信息
- 五、我们如何确保终端用户的个人信息安全
- 六、 我们如何使终端用户拥有管理个人信息的权利
- 七、我们如何处理未成年人的个人信息
- 八、我们如何更新与修订本声明
- 九、其他
- 十、如何联系我们
- 十一、名词解释

1. 我们如何收集和使用终端用户的个人信息

在终端用户使用集成海纳嗨数 SDK 的产品和/或服务过程中,我们严格遵守法律、法规的规定,按照本声明如下所述方式收 集、使用终端用户主动提供的或因为使用服务而产生的信息,用以向您提供服务。无论通过上述何种方式收集和使用终端用户 的个人信息,我们通常会要求开发者征得终端用户同意的情况下进行个人信息收集,除非在某些情况下,基于法律义务或者可 能需要保护终端用户或其他人的重要利益收集个人信息。

1.1. 为实现 SDK 产品功能所需要收集的终端用户个人信息

为了满足法律法规及提供产品和/或服务的基本要求,以及给终端用户提供更好的服务体验,我们会收集终端用户在使用集成海 纳嗨数 SDK 的产品和/或服务时产生的相关信息,并将这些信息传输至开发者购买的海纳数科集群,这些信息和权限包括如 下:

SDK 名称	个人信息类型	个人信息内容	个人信息处
嗨数云 SDK	设备属性信息(IMEI/Andriod ID/OAID/IDFA/IDFV/UUID/IMSI/Mac 地址)、设备连接信息 (浏览器类型、电信运营商、 设备品牌)	设备连接信息 (浏览器类型、 电信运营商、设备品牌	用户身份识
	日志信息	终端用户对集成海纳嗨数 SDK 的产品和/ 或服务的使用情况、IP 地址、所访问服务的 URL、 浏览器类型和使用的语言 、访问服务的时间	用户行为分
	位置信息	IP 地址解析的位置信息、 GPS 位置信息	用户行为分
	唯一应用程序编号	应用唯一标识、 应用名称和应用版本号	用户行为分
•			•

请注意,基于不同的设备和系统(如:Android/iOS)及系统版本,以及开发者在集成海纳嗨数 SDK 产品时决定的权限的不同, 相关产品和/或服务收集的信息会有所不同,因此开发者应对实际收集的终端用户个人信息向终端用户进行说明。嗨数云 SDK 提供的预置事件和预置属性信息收集能力可参考:【预置属性与系统字段】

单独的设备信息、日志信息、位置信息等是无法识别特定自然人身份的。如果我们将前述信息与其他个人信息结合用于识别特定自然人身份,或将其与能够识别自然人身份的信息结合使用,则在结合使用期间,除取得终端用户授权或法律法规另有规定 外,开发者应将该类个人信息作匿名化、去标识化处理。

1.2. 设备权限调用说明

如果终端用户需要关闭为使用上述某些特定功能而开启的功能权限(如:设备信息权限、地理位置权限等),大多数移动设备 都会支持终端用户的这项需求,具体方法请参考或联系移动设备的服务商或生产商。请终端用户注意,用户开启任一权限即代 表授权我们可以收集和使用相关信息来提供对应服务,终端用户一旦关闭任一权限即代表用户取消了授权,我们将不再基于对 应权限继续收集和使用相关信息,也无法提供该权限对应的服务。

Android SDK 产品权限说明

为保证支持客户数据收集的正常展开,Android SDK 需要以下系统权限:

权限	用途	是否必须
INTERNET	允许应用发送统计数据	必须权限,SDK 发送埋点数据需要此权限
ACCESS_NETWORK_STATE	允许应用检测网络状态	必须权限,SDK 会根据网络状态选择是否发送数据
READ_PHONE_STATE	允许应用获取设备 IMEI、MEID	可选权限,采用 App 内推广和收集 \$carrier 属性时会用到此权限
ACCESS_WIFI_STATE	允许应用获取 MAC 地址	可选权限,采用 App 内推广时会用到此权限
4		▶

iOS SDK 产品权限说明

为保证支持客户数据收集的正常展开,iOS SDK 需要以下系统权限:

权限	用途	是否必须
网络 (国服专门)	允许应用发数据	必须权限,SDK 发送埋点数据需要此权限 定位
定位	允许应用获取 GPS 数据	可选权限,SDK 收集 GPS 数据时需要此权限
IDFA	允许应用获取 IDFA	可选权限,采用 App 内推广时会用到此权限

1.3. 基于征得同意的例外

根据相关法律法规的规定,在以下情形中,我们可以在不征得您的授权同意的情况下收集、使用一些必要的用户个人信息:与 履行法律法规规定的义务相关的; 1.与国家安全、国防安全直接相关的; 2.与公共安全、公共卫生、重大公共利益直接相关的; 3.与犯罪侦查、起诉、审判和判决执行等直接相关的; 4.出于维护您或其他个人的生命、财产等重大合法权益但又很难得到本人同意的; 5.所涉及的个人信息全国主体中在公公公公开的;

6.根据您的要求签订和履行合同所必需的;

7.从合法公开披露的信息中收集到您的个人信息,如从合法的新闻报道、政府信息公开等渠道;

8.法律法规规定的其他情形。

2. 我们如何使用 Cookie 和同类技术

当终端用户通过互联网包括但不限于移动终端各类设备使用集成海纳嗨数 SDK 的产品和/或服务时,为确保系统正常运转,我 们会在终端用户的设备上存储名为 Cookie 的小数据文件。Cookie 通常包含标识符、站点名称以及一些号码和字符。Cookie 主 要的功能是便于为终端用户供更加周到的个性化服务,并允许用户设定特定的服务选项。

当终端用户使用集成海纳嗨数 SDK 的产品和/或服务,我们会向终端用户的设备发送 Cookie。Cookie(或者其他匿名标识符) 并将上述 Cookie 直接发送给开发者购买的海纳数科集群。

我们不会将 Cookie 用于本声明所述目的之外的任何用途。终端用户可根据自己的偏好管理或删除 Cookie。终端用户可以清除计 算机上保存的所有 Cookie,大部分网络浏览器都设有阻止 Cookie 的功能。但如果终端用户这么做,则需要在每一次访问集成海 纳嗨数 SDK 的产品和服务时亲自更改终端用户的设置,但终端用户可能因为该等修改,无法登录或使用开发者提供的依赖于 Cookie 的服务或功能。

3. 我们如何共享、转让、公开披露终端用户的个人信息

终端用户使用集成海纳嗨数 SDK 的产品和/或服务时,通过海纳嗨数 SDK 收集的信息均存储在开发者购买的海纳数科集群,我 们无权接触用户个人数据,不涉及对终端用户个人信息的共享、转让、公开批露,但以下情况除外:

3.1在获取明确同意的情况下共享:获得您的明确同意后,我们会与其他方共享。

3.2我们可能会根据法律法规规定,或按司法机构、政府主管部门的强制性要求,对外共享。

3.3与我们的关联公司共享:为向您实现本声明的目的,您的用户信息可能会与海纳数科的关联公司共享,我们只会在必要情况 下向关联公司共享必要的信息,关联公司与海纳数科受本隐私政策中所声明目的的同等约束。关联公司如要改变信息的处理目的,将再次征求您的授权同意。

3.4与授权合作伙伴共享:仅为实现本声明的目的,我们可能会与受信赖的第三方供应商、顾问或其他服务商(简称"合作伙伴")共享您的某些用户信息,以提供更好的客户服务和用户体验。目前未涉及与合作伙伴共享,随着我们产品功能的升级迭代,如涉及该共享行为我们将提前获取您的明确授权。我们仅会出于合法、正当、必要、特定、明确的目的共享信息,并且只会共享提供服务所必要的信息。

对我们与之共享信息的公司、组织,我们会与其签署严格的保密协议,要求他们按照我们的说明、本隐私政策列明的目的或用途以及保密和安全措施来处理共享信息。

4. 我们如何保存终端用户的个人信息

我们严格遵守相关法律法规,采取业内认可的合理可行的措施,为开发者提供了终端用户个人信息保护能力,防止信息遭到未 经授权的访问、披露、使用、修改,避免信息损坏或丢失。

4.1保存期限

终端用户在使用集成海纳嗨数 SDK 的产品和/或服务时,由开发者购买的海纳数科集群及其关联的服务器负责存储终端用户个 人信息。开发者应按照法律法规的要求进行信息保存,并请确保在终端用户的个人信息超出保留期间后,对信息进行删除或匿 名化处理。

4.2 保存地域

开发者收集的终端用户个人信息存储在其购买的海纳数科集群及其关联的服务器上。开发者应严格按照法律法规要求进行个人 信息存储(如:将中国境内收集的个人信息存储于中国境内)。

5. 我们如何确保终端用户的个人信息安全

5.1产品安全措施

我们在产品开发过程中严格遵循安全开发生命周期流程管理,将安全和隐私要求嵌入产品开发的每一个阶段,通过实施各项安 全控制手段,有效地减少产品安全漏洞的数量将实际安全风险尽可能降低,从而构建更安全的产品。

5.2漏洞管理措施

我们具有完善的漏洞管理策略,通过多种手段对内/外部安全漏洞与威胁情报进行监控,对涉及的产品漏洞和威胁情报进行研判 和处置,严格按照"责任到人、时效催促、过程可视化"的流程机制进行漏洞全生命周期管理,确保所有漏洞能够及时解决。

5.3个人信息安全

围绕个人信息处理的收集、传输、存储、使用、删除等生命周期环节,海纳数科产品提供了收集延迟初始化、数据加密、数据 水印、数据脱敏等一系列的技术手段,确保个人信息的安全及合规,开发者可在集成嗨数云 SDK 时,选择恰当的保护手段保障 终端客户个人信息的安全。

5.4安全事件处置

我们将尽力确保海纳数科产品的安全性,确保使用海纳嗨数 SDK 收集的终端用户个人信息的安全性,但请您理解,由于技术的 限制以及在互联网行业可能存在的各种恶意手段,不可能始终保证信息百分之百的安全。为防止安全事故的发生,我们会制定 网络安全事件应急预案,及时协助开发者处置系统漏洞、计算机病毒、网络攻击、网络侵入等安全风险,采取相应的补救措 施,力求将损失最小化。

在不幸发生个人信息安全事件后,我们建议开发者按照法律法规的要求,及时向终端用户告知安全事件的基本情况和可能的影响、开发者已采取或将要采取的处理措施、终端用户可自主防范和降低的风险的建议和对终端用户的补救措施等。开发者应及时将事件相关情况以站内通知、短信通知、电话、邮件等终端用户预留的联系方式告知用户,难以逐一告知时开发者应采取合理、有效的方式发布公告。

同时,开发者应按照监管部门要求,主动上报个人信息安全事件的处置情况,紧密配合政府机关的工作。

6. 我们如何使终端用户拥有管理个人信息的权利

我们非常重视开发者和终端用户对个人信息的关注,在此:

6.1针对开发者

鉴于开发者直接负有响应用户个人信息请求的义务,开发者应:根据其集成海纳嗨数 SDK 的实际情况, 为用户提供并明确其查阅、复制、修改、删除个人信息、撤回同意、转移个人信息、限制个人信息处理、获取个人信息副本和注销账号的功能和途径。

6.2 针对终端用户

由于终端用户不是我们的直接用户,与海纳嗨数 SDK 之间没有直接的交互功能界面,为保障您的权利实现,我们已要求第三方 开发者承诺提供便于操作的用户权利实现功能和途径。如您需要查阅、复制、修改、删除其相关的个人信息、撤回同意、限制 个人信息处理、获取个人信息副本和注销账号,可通过第三方开发者提供的上述功能实现个人信息的需求。 请您留意,我们难以控制第三方开发者的行为,如开发者未按照承诺进行提供,您可通过本声明第十条中的方式与我们取得联 系,我们将尽力协调、支持并保障终端用户的权利的实现。若您依然选择直接向海纳数科提出相关请求的,我们为了保护海纳 数科的客户以及他人的合法权益,我们会要求您进行身份的验证,请您及时提供相关的权利要求证明文件。在向您的应用提供 方进行核实后,我们会根据法律法规以及本隐私政策的规定处理您的相关请求。

7. 我们如何处理未成年人的个人信息

我们非常重视对未成年用户人信息的保护。海纳嗨数 SDK 产品和/或服务主要面向企业和成人。我们已提请开发者确保产品和/ 或服务的使用对象为 18 周岁(含)以上人士。如您是 18 周岁以下的未成年人,您应确保您的监护人陪同阅读并同意本声明 后,您再使用集成海纳嗨数 SDK 的产品和/或服务并提供您的个人信息。

对于经父母或法定监护人同意而收集未成年人个人信息的情况,我们只会在受到法律允许、父母或监护人明确同意或者保护未 成年人所必要的情况下使用或公开披露此信息。

若您是未成年人的监护人,当您对您所监护的未成年人使用我们的产品和/或服务或其向我们提供的用户信息有任何疑问时,请 您及时与我们联系。我们将根据国家相关法律法规及本声明的规定保护未成年人用户信息的保密性及安全性。如果我们发现自 己在未事先获得可证实的父母或法定监护人同意的情况下收集了未成年人的个人信息,则会尽快删除相关数据。

8. 我们如何更新与修订本声明

我们保留适时更新本声明的权利。未经您明确同意,我们不会改变您按照本声明所应享有的权利。如因海纳嗨数 SDK 功能等原 因导致的本声明发生更新,我们会通过网站公告等方式通知您,以便您能及时了解本声明所做的任何变更。

如果您是第三方开发者,当更新后的本声明对处理终端用户个人信息情况有重大变化时,您应当适时更新隐私政策,并通过弹 窗等方式告知 App 等终端用户。如果终端用户不同意接受修改后的隐私政策,请终端用户停止使用集成海纳嗨数 SDK 的产品 和/或服务。如果终端用户继续使用集成海纳嗨数 SDK 的产品和/或服务,则视为接受我们对本声明相关条款所做的修改。

9. 其他

9.1《海纳嗨数隐私协议》是我们产品及服务统一适用的一般性隐私条款,其中所规定的用户权利及信息安全保护措施均适用嗨数云 SDK 用户,在您使用海纳嗨数 SDK 提供的产品和/或服务时,如《海纳嗨数隐私协议》与本声明存在不一致或矛盾之处,请以本声明为准。

9.2本文档内容为海纳数科产品使用和技术细节说明文档,不包含适销类条款;具体企业采购产品和技术服务内容,以商业采购 合同为准。

10. 如何联系我们

如对本隐私声明内容有任何疑问、意见或建议,您可通过登录海纳嗨数"官网-联系我们"或选择拨打热线 400-667-9660与我们联系。一般情况下,我们将在15日予以回复("回复期"),但您理解并同意,结合您请求的复杂度,前述回复期可能延迟至30日,但我们会在回复期届满前向您书面告知延迟原因。

为保障我们高效处理您的问题并及时向您反馈,您需要向我们提交有效身份证明、有效联系方式和书面请求及相关证据,我们 会在验证您的身份后处理您的请求。

如果您对我们的回复或者处理意见不满意的,在此您同意向海纳数科注册地(上海市徐汇区)人民法院提交诉讼解决。

11. 名词解释

本声明中使用的特定词语,具有如下含义:

1.关联公司:海纳致远数字科技(上海)有限公司合并报表范围内的任何公司主体,上述公司中任何一方现在或将来控制、受控 制或与其处于共同控制下。

2.控制:是指直接或间接地拥有影响所提及公司管理的能力,无论是通过所有权、有投票权的股份、合同或其他被依法认定的方式。

3.您:指使用我们的产品和/或服务的注册用户,包括集成海纳嗨数 SDK 的产品和/或服务的开发者和运营者、使用集成海纳嗨数 SDK 的产品和/或服务的终端用户。

4.个人信息:指以电子或者其他方式记录的能够单独或者与其他信息结合识别特定自然人身份或者反映特定自然人活动情况的各种信息。个人信息包括个人基本信息、个人身份信息、个人生物识别信息、网络身份标识信息、个人健康生理信息、个人教育工作信息、个人财产信息、个人通信信息、联系人信息、个人上网记录、个人常用设备信息、个人位置信息等。为免疑义,个人信息包括但不限于个人敏感信息。

5.个人信息主体:指个人信息所标识的自然人。

6.个人敏感信息:指一旦泄露、非法提供或滥用可能危害人身和财产安全,极易导致个人名誉、身心健康受到损害或歧视性待遇 等的个人信息。个人敏感信息包括个人财产信息、个人健康生理信息、个人生物识别信息、个人身份信息、网络身份标识信息 等。

7.去标识化:指通过对个人信息的技术处理,使其在不借助额外信息的情况下,无法识别个人信息主体的过程。
8.匿名化:指通过对个人信息的技术处理,使得个人信息主体无法被识别,且处理后的信息不能被复原的过程。
9.中国或中国境内:指中华人民共和国大陆地区,仅为本声明之目的,不包含香港特别行政区、澳门特别行政区和台湾地区。

Android SDK

SDK 功能清单

一级功能	二级功能	具体功能 / 描述	
标识用户	ᅌᅻᆤᄁᄛᄼᄑᅌ	自动使用设备 ID 标识匿名用户	
	日列称识匿名用广	支持替换为自定义的匿名 ID	
	标识登录用户	使用登录 ID 标识登录用户	
	设置用户属性		
	自动采集设备信息	自动采集默认的设备信息,屏幕宽高、系统版本号等	
		支持自动采集屏幕方向	
		支持自动采集经纬度信息	
	全埋点	App 启动	
		App 退出	
		App 页面浏览	
采集数据		App 元素点击	
	公共属性	静态公共属性	
		动态公共属性	
		清除公共属性	
		采集激活(安装)事件	
	自定义埋点	统计事件时长	
		自定义代码埋点	
存储数据	设置上限	设置本地缓存上限值	
	删除本地缓存	删除缓存在本地的所有事件	
上报数据	自动上报数据	SDK 在满足一定条件后自动发送事件数据	

	手动上报数据	SDK 支持手动上报事件数据
调试功能	调试模式	开启 / 关闭调试模式
	调试日志	显示 / 关闭调试日志/td>
高级功能	数据加密	数据加密之后再进行存储和发送
	App 与 H5 打通	打通后 H5 的事件数据由 App SDK 来存储和发送
	合规	支持合规要求
	采集推送点击事件	SDK 自动采集推送点击事件
	SDK 采集控制	控制 SDK 开启或关闭

集成文档

帮助中心 > 技术文档 > 客户端SDK > Android SDK > 集成文档

注意:

- 1. 在线maven集成、离线包集成均支持;
- 2. Android SDK 要求最低系统版本为 API 14 (Android 4.0) ;
- 3. 目前Android SDK (aar 格式) 大小约为 600 KB。

1. 在线maven集成

1、在项目根目录下的build.gradle文件,配置maven地址,并配置插件依赖;

```
buildscript {
  repositories {
 mavenCentral()
//配置maven地址
maven { url 'https://repo1.maven.org/maven2/' }
...
}
dependencies {
     . . .
//添加SDK插件依赖
classpath 'com.hinadt.hicloud.android:auto-plugin:1.0.0'
...
}
}
allprojects {
  repositories {
mavenCentral()
//配置maven地址
maven { url 'https://repo1.maven.org/maven2/' }
...
}
}
```

2、在项目主module目录下的build.gradle文件,配置SDK依赖,并配置应用插件;

```
apply plugin: 'com.android.application'

//配置应用插件

apply plugin: 'com.hinadt.hicloud.android'

android {

...

}

dependencies {

...

//配置SDK依赖

implementation 'com.hinadt.hicloud.android:HinaCloudSDK:1.0.0'

...

}
```

2. 离线包集成

请先下载HinaCloudSDK-Android-1.0.0.zip包,里面包含SDK包(aar包)和插件包(jar包),下载详见【资源下载】;

2.1 配置SDK

- 1、解压HinaCloudSDK-Android-1.0.0.zip包,获取到SDK离线包(aar包);
- 2、在项目的app module,或者library module目录下,新建libs目录,并把SDK离线包拷贝到libs目录下;
- 3、SDK离线包拷贝后,所在的module的build.gradle文件,进行配置依赖,具体配置如下:

```
. . .
android {
. . .
}
//配置依赖目录
repositories {
  flatDir {
      dirs 'libs'
 }
}
dependencies {
   ...
   //配置依赖
   implementation(name: 'xxx', ext: 'aar') //只允许在依赖aar文件的module下调用该aar文件
或
   api (name:'xxx',ext:'aar') //在其他依赖该library module的module中, 也可调用该aar文件
```

注意:如果是组件module,除了需要改用api配置依赖,还需要额外配置,方式有两种:

方式一: 在其它所有直接或间接依赖该library module的module中配置build.gradle文件如下:

```
repositories {
    flatDir {
        dirs '../xxx/libs','libs' //将xxx替换为引入aar文件的module名
    }
}
```

```
方式二:
在project的build.gradle文件中统一配置如下::
```

```
allprojects {
    repositories
    flatDir {
        dirs project(':xxx').file('libs') //将xxx替换为引入aar文件的module名
        }
    }
}
```

2.2 配置插件

- 1、解压HinaCloudSDK-Android-1.0.0.zip包,获取到离线包(jar包);
- 2、在项目根目录下,新建repo(自定义)目录,并把插件离线包拷贝到repo目录下;
- 3、在项目根目录下的build.gradle文件,配置使用本地gradle插件包,配置如下:

```
buildscript {
 repositories {
    . . .
 //配置本地仓库
 maven {
  url './repo'
  }
}
  dependencies {
     //注意:如果项目的gradle插件版本是4.0以前的,建议升级到4.0以上版本再集成;
     classpath 'com.android.tools.build:gradle:4.1.3'
...
//添加 android-gradle-plugin 依赖
classpath files('./repo/插件离线包文件名.jar')
}
}
allprojects {
  repositories {
     ...
   //配置本地仓库
 maven {
     url './repo'
}
}
}
```

注意:如果项目gradle插件版本低于4.0的,且不考虑升级,可以通过平台提供的插件仓库(zip)文件,使用如下:

- (1) 下载plugin_repo.zip文件,详见【资源下载】,解压后拷贝到上述repo目录;
- (2) 上述配置 classpath files('./repo/插件离线包文件名.jar')
- 更改为: classpath 'com.hinadt.hicloud.android:auto-plugin:1.0.0'

4、在主module的build.gradle文件,应用 com.hinadt.hicloud.android 插件,配置如下:

```
apply plugin: 'com.android.application'
//配置应用插件
apply plugin: 'com.hinadt.hicloud.android'
android {
....
}
```

3. 配置权限

SDK 在 AndroidManifest.xml 中注册了以下四个权限,权限及解释说明如下:

```
<!-- 必须权限,允许应用访问网络,SDK 发送埋点数据需要此权限-->
<uses-permission android:name="android.permission.INTERNET" />
<!-- 必须权限,允许应用检测网络状态,SDK 会根据网络状态选择是否发送数据-->
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

<!-- 可选权限, 允许应用获取设备 IMEI, 采用 App 内推广和采集运营商属性时会用到此权限-->
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<!-- 可选权限, 允许应用获取 MAC 地址, 采用 App 内推广时会用到此权限-->
<uses-permission android:name="android.permission.READ_PHONE_STATE" /></uses-permission android:name="android.permission.READ_PHONE_STATE" /></uses-permission.accestanter" /></uses-permission.accestanter</pre>

注意:如果想要去除 SDK 注册的权限,可以使用 tools:node="remove" 配置。关于 tools:node="remove" 的详细说明可参考谷歌 官方文档,配置代码参考:

<uses-permission android:name="android.permission.READ_PHONE_STATE" tools:node="remove" />

使用说明

帮助中心 > 技术文档 > 客户端SDK > Android SDK > 使用说明

1. 初始化SDK

在 Application 的 onCreate() 方法中,按如下方式进行初始化; (注意需要在主线程初始化)

```
public class MyApplication extends Application {
@Override
public void onCreate() {
   super.onCreate();
 //应在最早的时机,进行SDK初始化;
 //延迟初始化SDK, 会导致全埋点采集不准确;
initHinaCloudSDK();
...
}
   private void initHinaCloudSDK() {
      HinaCloudSDK.Builder builder = new HinaCloudSDK.Builder();
      builder.enableLog(BuildConfig.DEBUG)//配置日志开关
      .setServerUrl("https://xxx?token=123456")//配置采集上报地址(在线申请)
          .setAutoTrackEventType(IAutoTrackEventType.APP_START
          | IAutoTrackEventType.APP_END
                  | IAutoTrackEventType.APP_VIEW_SCREEN
                  | IAutoTrackEventType.APP_CLICK)//配置全埋点开启,不配置则为关闭
     .build(this);//SDK初始化构建
}
}
```

注意: 1、采集上报地址获取方式,请参考【采集上报地址获取】; 2、更多初始化配置,请参考【API详解】;

2. 代码埋点 (手动采集)

说明:使用track()方法进行直接埋点,埋点事件支持添加自定义属性。

```
//直接埋点,不含自定义属性
HinaCloudSDK.getInstance().track("BuyProduct");
//直接埋点,添加自定义属性
try {
    JSONObject properties = new JSONObject();
    properties.put("ProductID", 123456); // 设置商品 ID
    properties.put("ProductCatalog", "Laptop Computer"); // 设置商品类别
    HinaCloudSDK.getInstance().track("BuyProduct", properties);
} catch (JSONException e) {
    e.printStackTrace();
}
```

注意:更多功能使用,请参考【API详解】;

API详解

帮助中心 > 技术文档 > 客户端SDK > Android SDK > API详解

1. 代码埋点 (手动采集)

用于记录埋点事件。

1.1 直接埋点

说明:使用track()方法进行直接埋点,埋点事件支持添加自定义属性。

```
//直接埋点,不含自定义属性
HinaCloudSDK.getInstance().track("BuyProduct");
//直接埋点,添加自定义属性
try {
    JSONObject properties = new JSONObject();
    properties.put("ProductID", 123456); // 设置商品 ID
    properties.put("ProductCatalog", "Laptop Computer"); // 设置商品类别
    HinaCloudSDK.getInstance().track("BuyProduct", properties);
} catch (JSONException e) {
    e.printStackTrace();
}
```

1.2 时长埋点

```
说明:需要成对调用计时器的开始和结束方法,以此来实现对统计时长的事件采集;
*在事件开始时,调用 trackTimerStart("ViewProduct") ,该方法并不会真正发送事件;
*在事件结束时,调用 trackTimerEnd("ViewProduct", properties) ,SDK 会触发 "Event" 事件,并自动将事件持续时间记
录在事件属性 "$event_duration" 中。
```

```
// 进入商品页面
// 调用 trackTimerStart("ViewProduct") 标记事件启动时间
HinaCloudSDK.getInstance().trackTimerStart("ViewProduct");
// ... 用户浏览商品
// 离开商品页
try {
    // 在属性中记录商品 ID
    JSONObject properties = new JSONObject();
    properties.put("product_id", PRODUCT_ID);
    // 调用 track, 记录 ViewProduct 事件, 并在属性 event_duration 中记录用户浏览商品的时间
    HinaCloudSDK.getInstance().trackTimerEnd("ViewProduct", properties);
    } catch (JSONException e) {
    e.printStackTrace();
    }
```

2. 全埋点 (自动采集)

SDK 可以自动采集一些用户行为,如 App 启动、退出、浏览页面、控件点击,共计四种。

2.1 开启全埋点

说明:在初始化时,通过setAutoTrackEventType进行配置开启,不配置则视为关闭全埋点。

//在初始化时,配置开启 builder.setAutoTrackEventType(IAutoTrackEventType.APP_CLICK | IAutoTrackEventType.APP_START | IAutoTrackEventType.APP_END | IAutoTrackEventType.APP_VIEW_SCREEN);

//忽略指定类型的全埋点事件(Integer对应四种自动采集的事件常量)
HinaCloudSDK.getInstance().disableAutoTrackEventType(List<Integer> list)

//重启被忽略的全埋点事件(Integer对应四种自动采集的事件常量) HinaCloudSDK.getInstance().enableAutoTrackEventType(List<Integer> list)

注意:fragment页面采集需要单独开启和配置,具体详见下一小节【2.3 fragment全埋点控制】;

2.2 activity全埋点控制

说明:通过 ignoreAutoTrackActivity()方法可以忽略某个或某些 Activity 的 App 点击事件与 App 页面浏览事件(包含所属 Fragment)。同时也可以通过 resumeAutoTrackActivities()方法恢复被忽略的页面。

```
//忽略单个、多个 Activity
HinaCloudSDK.getInstance().ignoreAutoTrackActivities(List<Class<?>> activitiesList);
//恢复被忽略的 Activity
HinaCloudSDK.getInstance().resumeAutoTrackActivities(List<Class<?>> activitiesList);
```

2.3 fragment全埋点控制

说明:通过 trackFragmentAppViewScreen()方法可以开启全部 Fragment 页面浏览事件的自动采集功能。

//初始化 SDK 之后, 开启自动采集 Fragment 页面浏览事件 HinaCloudSDK.getInstance().trackFragmentAppViewScreen();

//开启后,指定 Fragment采集 HinaCloudSDK.getInstance().enableAutoTrackFragment(Class<?> fragment);

//开启后,忽略单个、多个Fragment采集 HinaCloudSDK.getInstance().ignoreAutoTrackFragments(List<Class<?>> fragmentList);

//开启后,对被忽略的 Fragment恢复采集 HinaCloudSDK.getInstance().resumeIgnoredAutoTrackFragments(List<Class<?>> fragmentList);

注意:enableAutoTrackFragment() 方法用于采集指定的 Fragment 页面,ignoreAutoTrackFragment() 方法用于忽略指定的 Fragment 页面。当二者设置同时存在时,以 enableAutoTrackFragment() 方法设置的为准,ignoreAutoTrackFragment() 方法设置 的信息不在生效。

2.4 view全埋点控制

说明:view全埋点采集控制,支持某个view或某类view忽略;

//通过 ignoreView() 方法可以忽略某个 View 对象的 App 点击事件 HinaCloudSDK.getInstance().ignoreView(View view);

//通过 ignoreViewType() 方法可以忽略某种控件类型及其子类型的 App 点击事件 HinaCloudSDK.getInstance().ignoreViewType(Class viewType);

3. 事件属性

在进行埋点事件追踪时,您可以根据需求对埋点事件进行属性的定义。目前 SDK 中提供了公共属性用于给每个埋点事件添加属 性。

3.1 获取预置属性

说明:如需了解和使用预置属性,可以通过此方法获取预置属性。

```
JSONObject jsonObject = HinaCloudSDK.getInstance().getPresetProperties()
```

3.2 设置公共属性

说明:公共属性是指对于所有事件都需要添加的属性,设置之后 SDK 会在每次触发埋点时,自动获取并添加到触发的事件中。

```
HinaCloudSDK.getInstance().registerCommonProperties(new ICommonProperties() {
    @Override
    public JSONObject getCommonProperties() {
        try {
            JSONObject jsonObject = new JSONObject();
            jsonObject.put("appName", "Test");
            jsonObject.put("isLogin", xxxUtils.isLogin());
            return jsonObject;
            } catch (Exception e) {
               e.printStackTrace();
            }
            return null;
            }
            });
```

4. 用户属性

4.1 设置用户属性

说明:同一个 key 多次设置时, value 值会进行覆盖替换。

```
HinaCloudSDK.getInstance().userSet(String key, Object value);
或
HinaCloudSDK.getInstance().userSet(JSONObject properties);
```

4.2 固定初始值的属性

说明:同一个 key 多次设置时,value 值只会记录初次设定的值;适用于为用户设置首次激活时间、首次注册时间等属性。

```
HinaCloudSDK.getInstance().userSetOnce(String key, Object value);
或
HinaCloudSDK.getInstance().userSetOnce(JSONObject properties);
```

4.3 数值类型的属性

说明:同一个 key 多次设置时, value 值(数值类型)进行求和计算;常用于记录用户付费次数、付费额度、积分等属性。

```
HinaCloudSDK.getInstance().userAdd(String key, Number value)
或
HinaCloudSDK.getInstance().userAdd(Map<String, ? extends Number> properties)
```

4.4 集合类型的属性

说明:同一个 key 多次设置时,value 值(字符串类型)都会被添加到集合里;适用于用户喜爱的电影、用户点评过的餐厅等属 性。

```
Set<String> movies = new HashSet<String>();
movies.add("Sicario");
movies.add("Love Letter");
// 设定用户观影列表属性,设定后属性 "Movies" 为: ["Sicario", "Love Letter"]
HinaCloudSDK.getInstance().userAppend("Movies", movies);
```

4.5 属性取消

说明:取消已设置的某个用户属性。

HinaCloudSDK.getInstance().userUnset(String key);

4.6 清空用户属性

说明:清空当前用户的用户属性。

```
HinaCloudSDK.getInstance().userDelete();
```

5. 用户关联

5.1 用户登录

当用户注册成功或登录成功时,需要调用 SDK 的 setUserUld() 方法。

```
HinaCloudSDK.getInstance().setUserUId("业务userid");
```

注意:为了准确记录登录用户的行为信息,建议在以下时机各调用一次用户登录方法:

·用户在注册成功时

·用户登录成功时

·已登录用户每次启动 App 时

5.2 获取设备唯一ID

说明:如需获取设备唯一id,请在初始化SDK后调用getDeviceUld()方法进行获取。

HinaCloudSDK.getInstance().getDeviceUId();

5.3 自定义设备唯一ID

说明:默认情况下,SDK 会生成设备唯一ID 并可以保证该ID的唯一性,如需替换SDK默认分配的,配置如下;

//初始化时配置 builder.setDeviceUId(String deviceUId)

//或者在初始化后, 通过API配置 HinaCloudSDK.getInstance().setDeviceUId(xxx);

注意: SDK 默认使用 Androidld 作为设备 ID, 如果 Androidld 获取不到则获取随机的 UUID。

6. 数据存储与发送

在每次调用 track()、setUserUid()、userSet() 等方法时,SDK 会将埋点事件保存在数据库中,并会检查如下条件,以判断是否向服务器上传数据:

```
    1.是否是 WIFI/26/36/46/56 网络条件
    2.是否满足发送条件之一:

            a.与上次发送的时间间隔是否大于 flushInterval
            b.本地缓存日志数目是否大于 flushPendSize
            c.事件类型为 setUserUid() 方法触发的 $SignUp 事件
```

6.1 上报条件

6.1.1 缓存条数

说明:设置本地缓存日志的最大条目数,默认本地埋点数据缓存为100条。

//初始化时配置 builder.setFlushPendSize(int size);

//或者在初始化后, 通过API配置 HinaCloudSDK.getInstance().setFlushPendSize(xxx);

6.1.2 发送间隔

说明:设置埋点数据发送的间隔,默认为15s。

//设置两次数据发送的最小时间间隔,最小值 5 秒;时间间隔,单位毫秒。 //初始化时配置 builder.setFlushInterval(int interval); //或者在初始化后,通过API配置

HinaCloudSDK.getInstance().setFlushInterval(xxx);

6.1.3 设置数据的网络上传策略

说明:默认情况下,在 WIFI/3G/4G/5G 网络条件下,SDK 都会尝试去同步数据。可以自由组合来指定发送数据的网络策略。

HinaCloudSDK.getInstance().setFlushNetworkPolicy(xxx);

6.2 立即上报 (强制上报)

说明:如果追求数据采集的时效性,调用flush(),即可立即执行上报。

HinaCloudSDK.getInstance().flush();

注意:在 App 进入后台状态或监听到网络切换有网络时,SDK 会调用 flush() 方法,将缓存的数据发送。

6.3 本地数据缓存上限值

说明:SDK本地数据库默认缓存数据的上限值为 32MB。

```
//设置本地缓存上限值, 单位 byte, 默认为 32MB:32 * 1024 * 1024, 最小 16MB:16 * 1024 * 1024,
//若小于 16MB, 则按 16MB 处理。
//初始化时配置
builder.setMaxCacheSize(long maxCacheSize)
```

```
//或者在初始化后, 通过API配置;
HinaCloudSDK.getInstance().setMaxCacheSize(xxx);
```

注意:当存储数量达到上限值,会依次丢弃老数据,保留最新的数据。

6.4 清空本地缓存事件

说明:删除 App 本地存储的所有事件。

HinaCloudSDK.getInstance().clear();

注意:如果不是特殊要求,请不要调用此方法。

6.5 使用自签证书

说明:初始化 SDK 时,通过 setSSLSocketFactory() 方法设置自签证书,设置自签证书后,SDK 内所有的 HTTPS 请求会进行此 接口设置的证书校验。

builder.setSSLSocketFactory(SSLSocketFactory sslSocketFactory)

7. 其他功能

7.1 开启屏幕方向的自动采集

说明:初始化 SDK 时,进行如下配置,即可开启屏幕方向属性的收集,屏幕方向信息记录在事件的 \$screen_orientation 属性中。

builder.enableTrackScreenOrientation(boolean enable);

注意:该接口开启后会读取手机传感器信息,请根据合规需要进行说明。

7.2 设置和清除位置信息

说明:通过 setGPSLocation() 方法,把第三方定位获取到的经纬度信息设置给 SDK,设置之后经纬度信息记录在事件的 \$longitude、\$latitude 属性中。清除位置信息调用 clearGPSLocation() 方法进行清除。

//设置经纬度 HinaCloudSDK.getInstance().setGPSLocation(double latitude, double longitude)

//清除位置信息 HinaCloudSDK.getInstance().clearGPSLocation()

7.3 开启Crash信息的自动采集

说明:初始化 SDK 时,通过如下配置,即可开启App Crash信息的采集。 SDK 默认关闭 Crash 信息采集,开启采集后,App Crash 时,会触发 AppCrashed 事件,堆栈信息记录在 app_crashed_reason 属性中。

builder.enableTrackAppCrash();

7.4 开启推送点击事件的自动采集

说明:初始化 SDK 时,进行如下配置。

//传入 true 代表开启推送点击事件自动采集
builder.enablePushAutoTrack(boolean enable)

7.5 开启App与H5打通功能

说明:初始化 SDK 时,进行如下配置。

builder.enableJSBridge(boolean enable);

7.6 日志开关

说明:初始化 SDK 时,进行如下配置。

builder.enableLog(boolean enable)

7.7 禁用SDK

说明:初始化 SDK 时,进行如下配置。

builder.disableSDK(boolean disableSDK);

更新日志

帮助中心 > 技术文档 > 客户端SDK > Android SDK > 更新日志

版本:1.0.0

【发布日期】: 2023-02-23

【更新内容】:

1. 发布SDK初版,支持代码埋点和自动埋点;

【注意事项】: 暂无

版本:1.0.1

【发布日期】: 2023-04-04

【更新内容】:

1. 增加初始化失败异常兜底;

【注意事项】: 暂无

iOS SDK

SDK 功能清单

一级功能	二级功能	具体功能 / 描述	
标识用户	自动标识匿名用户	自动使用设备 ID 标识匿名用户	
		支持替换为自定义的匿名 ID	
	标识登录用户	使用登录 ID 标识登录用户	
	设置用户属性		
	自动采集设备信息	自动采集默认的设备信息,屏幕宽高、系统版本号等	
		支持自动采集屏幕方向	
		支持自动采集经纬度信息	
	全埋点	App 启动	
		App 退出	
		App 页面浏览	
采集数据		App 元素点击	
	公共属性	静态公共属性	
		动态公共属性	
		清除公共属性	
		采集激活(安装)事件	
	自定义埋点	统计事件时长	
		自定义代码埋点	
存储数据	设置上限	设置本地缓存上限值	
	删除本地缓存	删除缓存在本地的所有事件	
上报数据	自动上报数据	SDK 在满足一定条件后自动发送事件数据	

	手动上报数据	SDK 支持手动上报事件数据
调试功能	调试模式	开启 / 关闭调试模式
	调试日志	显示 / 关闭调试日志/td>
高级功能	数据加密	数据加密之后再进行存储和发送
	App 与 H5 打通	打通后 H5 的事件数据由 App SDK 来存储和发送
	合规	支持合规要求
	采集推送点击事件	SDK 自动采集推送点击事件
	SDK 采集控制	控制 SDK 开启或关闭

集成文档

帮助中心 > 技术文档 > 客户端SDK > iOS SDK > 集成文档

1. CocoaPods方式

在 Podfile 文件中添加 pod 'HinaCloudSDK',在根目录执行pod install或pod update命令。

pod 'HinaCloudSDK'

2. 源码方式

1. 下载HinaCloudSDK

前往【资源下载】下载 HinaCloudSDK.framework.zip 并解压;

2. 将HinaCloudSDK引入项目

将HinaCloudSDK.framework文件夹导入App项目,选中Copy items if needed;

3. 添加依赖库

项目设置 "Build Phase" -> "Link Binary With Libraries" 中添加依赖库: libicucore、libsqlite3 和 libz;

*SDK兼容iOS 8.0及以上版本

使用说明

帮助中心 > 技术文档 > 客户端SDK > iOS SDK > 使用说明

1. 初始化SDK

在AppDelegate.m的- (BOOL)application:didFinishLaunchingWithOptions: 方法中添加初始化代码:

```
// 引入SDK
#import <HinaCloudSDK/HinaCloudSDK.h>
- (BOOL)application:(VIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
  // 初始化配置
   HNBuildOptions *options = [[HNBuildOptions alloc] initWithServerURL:<数据接收地址> launchOptions:launchOp
tions];
/**
* 可选项:开启全埋点,选择开启的全埋点类型,可根据需求进行组合
* 支持四种全埋点类型: App启动、退出、元素点击、页面浏览
*/
options.autoTrackEventType = HNAutoTrackAppStart |
                              HNAutoTrackAppEnd |
                             HNAutoTrackAppClick |
                            HNAutoTrackAppScreen;
#ifdef DEBUG
// 开启 Log
options.enableLog = YES;
#endif
// 初始化 sdk
[HinaCloudSDK startWithConfigOptions:options];
return YES;
}
```

注意:

- 1、采集上报地址需在海纳嗨数平台,进行在线申请,请参考【采集上报地址获取】;
- 2、关于 HNBuildOptions 的更多初始化配置请参考 API详解;

2. 代码埋点(手动采集)

调用 track 接口,追踪一个带有属性的 event

```
/**
 * @param event event的名称
 * @param propertyDict event的属性
 */
 [[HinaCloudSDK sharedInstance] track:@"Shopping" withProperties:@{
    @"GoodsPrice":@"100",
    @"GoodsId":[NSNumber numberWithUnsignedLong:123456789],
    @"GoodsName":@"football"
 }];
```

API详解

帮助中心 > 技术文档 > 客户端SDK > iOS SDK > API详解

1. 代码埋点 (手动采集)

1.1 直接埋点

1.2 统计埋点事件时长

需要成对调用计时器的开始、结束等方法,以此来实现对统计时长的事件采集:

```
// 开始播放电影
[[HinaCloudSDK sharedInstance] trackTimerStart:@"WatchMovie"];
// 暂停播放
[[HinaCloudSDK sharedInstance] trackTimerPause:@"WatchMovie"];
// 恢复播放
[[HinaCloudSDK sharedInstance] trackTimerResume:@"WatchMovie"];
// 结束播放
[[HinaCloudSDK sharedInstance] trackTimerEnd:@"WatchMovie" withProperties:@{@"MovieType":@"classical"}];
注意:
- 只有调用 - trackTimerEnd: 时, SDK 才会真正记录事件
- 多次调用 - trackTimerStart:, 以最后一次调用作为计时起点
- 默认情况下, 统计的时长不包括用户 App 在后台的时长
```

2. 全埋点 (自动采集)

全埋点也称自动埋点、无痕埋点,目前SDK支持的全埋点有 App 启动、App 退出、浏览页面和控件点击 四种事件。

2.1 开启全埋点

SDK初始化前,可通过配置HNBuildOptions实例的autoTrackEventType属性按需开启全埋点类型:

2.2 开启子页面浏览事件自动采集

在 SDK初始化 前,通过 HNBuildOptions 实例的 enableAutoTrackChildViewScreen 属性开启自动采集子页面的浏览事件:

options.enableAutoTrackChildViewScreen = YES;

2.3 自动采集事件的忽略补充

SDK开启全埋点后,SDK 支持通过配置忽略部分页面或控件的采集。

2.3.1 忽略某个控件的点击事件

button.hinaDataIgnoreView = YES;

2.3.2 忽略某类控件的点击事件

```
// 支持多次调用,对合集进行忽略
[[HinaCloudSDK sharedInstance] disableViewType:[HNButton class]];
[[HinaCloudSDK sharedInstance] disableViewType:[HNView class]];
```

2.3.3 忽略页面的浏览事件

```
// 可多次调用,对合集进行忽略
[[HinaCloudSDK sharedInstance] disableAutoTrackViewControllers:@[@"DemoViewController",@"HNBaseViewContro
ller"]];
```

2.3.4 手动触发页面的点击事件

```
// 可在 viewDidAppear 中触发此操作
[[HinaCloudSDK sharedInstance] trackViewScreen:[UIViewController new]];
```

3. 事件属性

在进行埋点事件追踪时,您可以根据需求对埋点事件进行属性的定义。目前 SDK 中提供了公共属性用于给每个埋点事件添加属 性。

3.1 获取预置属性

如需了解和使用预置属性,可以通过此方法获取预置属性。

```
// 获取sdk预置的属性
[[HinaCloudSDK sharedInstance] getPresetProperties];
```

3.2 设置事件公共属性

公共属性是指对于所有事件都需要添加的属性,设置之后 SDK 会在每次触发埋点时,自动获取并添加到触发的事件中。
4. 用户属性

4.1 设置用户属性

用于直接设置用户的一个或者几个 userDict, 建立用户模型。同一个 key 多次设置时, value 值会进行覆盖替换。

[[HinaCloudSDK sharedInstance] set:@{

}];

4.2 固定初始值的属性

对于需要保证只有首次设置时有效的属性,如用户首次注册地点、首次设置的用户昵称等,可以使用setOnce接口进行记录。后续设置已经存在的属性不会覆盖已有数据,如果设置不存在的属性则会自动创建:

```
// 设定用户 AdSource 渠道为为 "App Store"
[[HinaCloudSDK sharedInstance] setOnce:@"AdSource" to:@"App Store"];
```

4.3 数值类型的属性

同一个 key 多次设置时, value 值(数值类型)进行累加;常用于记录用户付费次数、付费额度、积分等属性。

```
// 对单条属性累加
[[HinaCloudSDK sharedInstance] add:@"my_level" by:[NSNumber numberWithInt:1]];
// 对多条属性累加
[[HinaCloudSDK sharedInstance] add:@{
    @"my_level":[NSNumber numberWithInt:1],
    @"my_golds":[NSNumber numberWithInt:23]
}];
```

4.4 集合类型的属性

同一个 key 多次设置时, value 值 (字符串类型)进行追加;适用于用户喜爱的电影、用户点评过的餐厅等属性。

```
// 追加水果列表的类型
[[HinaCloudSDK sharedInstance] append:@"Fruits" by:[NSSet setWithArray:@[@"apple",@"pitaya"]]];
```

4.5 属性取消

取消已设置的某个用户属性。

// 取消Level属性
[[HinaCloudSDK sharedInstance] unset:@"Level"];

4.6 清空用户属性

[[HinaCloudSDK sharedInstance] deleteUser];

5 用户关联

用户关联是为了对用户进行唯一标识,提高用户行为分析的准确性。

5.1 用户登录

当用户注册成功或者进行登录时,需要调用SDK 的用户登录接口:

[[HinaCloudSDK sharedInstance] setUserUId:@"<你的登录id>"];

// 为了准确记录登录用户的行为信息,建议在以下时机各调用一次用户登录接口:
 // · 用户在注册成功时
 // · 用户登录成功时

//·已登录用户每次启动 App 时

5.2 获取设备唯一ID

// 如需获取设备唯一id, 请在初始化SDK后调用 [[HinaCloudSDK sharedInstance] deviceUId];

注意:

如果 App 引入了 AdSupport 库, SDK 默认使用 IDFA 作为设备唯一 ID, 使用 IDFA 能避免用户在重装 App 后设备 ID 发 生变化的情况。

若没有IDFA, SDK 会使用 IDFV, 如果 IDFV 获取失败, 则使用随机的 UUID, 一般情况下都能够获取到 IDFV。如果使用 IDF V 或 UUID, 当用户卸载重装 App 时设备 ID 会变。

5.3 自定义设备唯一ID



6. 数据存储与发送

6.1 上报条件

iOS SDK 每次触发事件时会检查如下条件,以判断是否向服务器上传数据:

1. 当前网络是否符合 flushNetworkPolicy (默认 3G、4G、5G、WiFi)

- 2. 与上次发送的时间间隔是否大于 flushInterval (默认 15 秒)
- 3. 本地缓存的事件条目数是否大于 flushPendSize (默认 100 条)

只有1、2 或1、3 满足时,SDK 才会进行发送数据。以上参数支持自定义,可以通过修改相应参数值来达到控制事件上报的频率:

HNBuildOptions *ops = [[HNBuildOptions alloc] initWithServerURL:@"<数据接收地址>" launchOptions:launchOpt ions];

```
// 设置触发间隔, 默认 15 * 1000 毫秒
ops.flushInterval = 10 * 1000;
```

```
// 设置触发条数, 默认 100 条
ops.flushPendSize = 50;
```

// 设置上报网络策略, 默认 3G、4G、5G、WiFi ops.flushNetworkPolicy = HNNetworkTypeALL;

[HinaCloudSDK startWithConfigOptions:ops];

6.2 立即上报 (强制上报)

特殊场景需要立即上报的,可以在事件触发后调用 - flush:

[[HinaCloudSDK sharedInstance] flush];

6.3 本地数据缓存上限值

如果一直不满足上报条件,本地缓存的事件量达到 maxCacheSize 时,每次再触发新的事件,SDK 会依次丢弃老数据,保留最 新的数据。maxCacheSize支持自定义,默认10000条:

ops.maxCacheSize = 15000;

6.4 清空本地缓存事件

为适应 GDPR 要求,SDK 可调用 - clear 方法用于清空本地缓存的全部事件:

// 删除本地缓存的全部事件,请慎用! [[HinaCloudSDK sharedInstance] clear];

6.5 本地证书校验

SDK 发送数据时支持本地证书校验,可按照如下步骤进行配置: 服务器生成 SSL 证书,如果不是 .cer 格式,需要使用 OpenSSL 进行格式转换:

openssl x509 -in 你的证书.crt -out 你的证书.cer -outform der

```
/*
   设置证书校验方式,默认为 SASSLPinningModeNone
   */
   SASecurityPolicy *sp = [SASecurityPolicy policyWithPinningMode:SASSLPinningModePublicKey];
   /*
   是否允许无效证书(自建证书),默认为 NO ,
   如果是需要验证自建证书,需要设置为 YES。
   */
   sp.allowInvalidCertificates = YES;
   /*
   是否需要验证域名,默认为 YES,
   如证书的域名与请求的域名不一致,需把该项设置为 NO 。
   */
   sp.validatesDomainName = NO;
   /*
   配置本地证书路径
   */
   sp.pinnedCertificates = [SASecurityPolicy certificatesInBundle:[NSBundle mainBundle]];
   // SDK 初始化配置
   HNBuildOptions *ops = [[HNBuildOptions alloc] initWithServerURL:@"<数据接收地址>" launchOptions:launchOpt
ions];
   /*
   设置证书策略
   */
 ops.securityPolicy = sp;
   [HinaCloudSDK startWithConfigOptions:ops];
```

SSLPinningMode 的三种模式:

- SASSLPinningModeNone:默认模式,客户端无条件信任服务端返回证书,不做 App 本地证书校验
- SASSLPinningModePublicKey:只验证证书中公钥,不验证证书的有效期等信息
- SASSLPinningModeCertificate:校验本地证书的所有信息

SASSLPinningModeCertificate 模式会校验证书有效期等信息,如果不能保证用户 App 证书始终处于未过期状态,建议使用 SASSLPinningModePublicKey 模式。

7. 其他功能

7.1 开启屏幕方向的自动采集

可以在初始化 SDK 后调用 - enableTrackScreenOrientation: 进行开启,开启后的屏幕方向信息记录在事件的 \$screen_orientation 属性中:

[[HinaCloudSDK sharedInstance] enableTrackScreenOrientation:YES];

7.2 开启位置信息的自动采集

SDK支持自动采集GPS信息,可以在SDK初始化后调用 enableTrackGPSLocation 方法进行开启,采集后的经纬度信息会乘以10⁶ 后记录在 \$longitude、\$latitude 属性中:

```
[[HinaCloudSDK sharedInstance] enableTrackGPSLocation:YES];
```

注意:如果app业务中未开启定位权限,针对审核问题,开启此项功能后需在plist文件中添加相关用途说明。

7.3 开启Crash信息的自动采集

SDK支持自动采集Crash信息,可以在初始化SDK后配置 HNBuildOptions 实例化的 enableTrackAppCrash 属性进行开启,此功能 默认是关闭的:

HNBuildOptions *options = [[HNBuildOptions alloc] initWithServerURL:kAppServerURL launchOptions:launchOpt
ions];

```
options.enableTrackAppCrash = YES;
[HinaCloudSDK startWithConfigOptions:options];
```

7.4 开启推送点击事件的自动采集

在SDK初始化 前,通过 HNBuildOptions 实例的 enablePushAutoTrack 属性开启推送点击事件的自动采集:

```
options.enablePushAutoTrack = YES;
// 如需设置推送唯一id, 请在初始化SDK后调用
[[HinaCloudSDK sharedInstance] userPushKey:@"###" pushId:@"###"];
```

7.5 开启 App 与 H5打通功能

初始化 SDK 时,进行如下配置,即可开启 App 打通 H5 功能:

options.enableJSBridge = YES;

7.6 日志开关

```
#ifdef DEBUG
    // 开启 Log
    options.enableLog = YES;
#endif
```

7.7 禁用SDK

```
// 禁用 SDK, 默认为 NO
// 禁用后, SDK 将不会触发事件, 也不会发送网络请求
options.disableSDK = NO;
```

更新日志

帮助中心 > 技术文档 > 客户端SDK > iOS SDK > 更新日志

版本:1.0.0

【发布日期】: 2023-02-23

【更新内容】:

1. 发布SDK初版,支持代码埋点和自动埋点。

Web JS SDK

SDK 功能清单

一级功能	二级功能	具体功能 / 描述	
	自动标识匿名用户	自动使用设备 ID 标识匿名用户	
		支持替换为自定义的匿名 ID	
你识用尸	标识登录用户	使用登录 ID 标识登录用户	
	设置用户属性		
	自动采集设备信息	自动采集默认的设备制造商,操作系统、浏览器版本等	
		Web 浏览页面	
	全埋点	Web 元素点击	
☆隹粉掘		Web 视区停留	
不朱奴加	公共属性	静态公共属性	
		动态公共属性	
		清除公共属性	
	自定义埋点	自定义代码埋点	
上报数据	自动上报数据	SDK 在满足一定条件后自动发送事件数据	
シ囲シᅻ᠇┧台ヒ	调试模式	开启 / 关闭调试模式	
明以为旧日	调试日志	显示 / 关闭调试日志	
ㅎ 47-11-44	数据加密	数据加密之后再进行存储和发送	
	APP与 H5 打通	打通后 H5 的事件数据由 App SDK 来存储和发送	
同纵划形	合规	支持合规要求	
	SDK 采集控制	控制 SDK 开启或关闭	

集成文档

帮助中心 > 技术文档 > 客户端SDK > Web Js SDK > 集成文档

1. 集成SDK

步骤一:下载请前往【Web SDK资源下载】

压缩包中提供了两种规范的脚本,您可以根据需要选择所需的脚本。下文描述的同步载入,需要使用 hina.min.js 文件, ES6模块载入需要使用 hina.esm.min.js

步骤二:载入 JavaScript SDK

您可以任意选择同步载入或者ES6模块载入的方式去使用 SDK. 在初始化 SDK 的时候需要传入一些配置参数: serverUrl :上传数据的 URL,需要进行配置

1.1 同步载入

对于 同步载入 或者 服务端渲染 方式,请使用 hina.min.js ,将以下代码置入 html 的 <head> ,在页面头部通过 script src 的方式引入,将以下代码置入初始化代码中,并配置对应参数:

<!--Hina SDK BEGIN--> <script src="./hina.min.js"></script> <script> // 创建 SDK 全局对象 const hina = window['hinaDataStatistic']; // 用配置对象初始化 SDK hina.init({ serverUrl: 'xxx', showLog: true, //单页面配置,默认开启,若页面中有锚点设计,需要将该配置删除,否则触发锚点会多触发 H_pageview 事件 isSinglePage: true, autoTrackConfig: { //是否开启自动点击采集, true表示开启, 自动采集 H_WebClick 事件 clickAutoTrack: true, //是否开启页面停留采集, true表示开启, 自动采集 H_WebStay 事件 stayAutoTrack: true, }, }); hina.quick('autoTrack'); // 将 SDK 实例赋给全局变量 hina, 或者其他您指定的变量 window["hina"] = hina; </script> <!--Hina SDK END-->

1.2 ES6模块化引入

对于 ES6 模块化引入,直接放入主文件中使用

```
import hina from './hina.es6.min.js'
hina.init({
    serverUrl: 'xxx',
    showLog: true,
```

集成文档

//单页面配置,默认开启,若页面中有锚点设计,需要将该配置删除,否则触发锚点会多触发 H_pageview 事件 isSinglePage: true, autoTrackConfig: { //是否开启自动点击采集, true表示开启,自动采集 H_WebClick 事件 clickAutoTrack: true, //是否开启页面停留采集, true表示开启,自动采集 H_WebStay 事件 stayAutoTrack: true, }, }); //用于采集 H_pageview 事件 hina.quick('autoTrack');

全埋点

帮助中心 > 技术文档 > 客户端SDK > Web Js SDK > 全埋点

1. 开启全埋点

1.1 代码示例

```
<script src="./hina.min.js"></script>
<script>
// 创建 SDK 全局对象
const hina = window['hinaDataStatistic'];
// 用配置对象初始化 SDK
hina.init({
  serverUrl: 'xxx',
 showLog : true,
 //单页面配置,默认开启,若页面中有锚点设计,需要将该配置删除,否则触发锚点会多触发 H_pageview 事件
 isSinglePage: true,
 autoTrackConfig: {
 //是否开启自动点击采集,true表示开启,自动采集 H_WebClick 事件
      clickAutoTrack: true,
 //是否开启页面停留采集,true表示开启,自动采集 H_WebStay 事件
      stayAutoTrack: true,
  },
});
hina.quick('autoTrack');
</script>
```

Web SDK全埋点包括三种事件:Web页面浏览、Web元素点击、Web视区停留。如果需要针对全埋点类型进行单独配置,可以参考下面单个全埋点类型使用的文档

1.2 Web页面浏览(H_pageview)

1.2.1 自动模式

配置参数 isSinglePage,默认值为 false ,表示是否开启自动采集 web 页面浏览事件 H_pageview 的功能

```
//如果进首页不会自动 redirect 时, hn.quick('autoTrack') 是需要的, 否则不需要
isSinglePage: true
```

1.2.2 手动模式

在页面切换的时候,手动调用 hn.quick('autoTrackSinglePage') 来采集 web 页面浏览事件 H_pageview ,这个方法在页面 url 切 换后调用

```
//在react中的全局onUpdate里调用
onUpdate(){
    hn.quick('autoTrackSinglePage');
}
//在vue路由切换时调用
router.afterEach(() => {
```

```
hn.quick("autoTrackSinglePage");
});
```

1.3 Web元素点击(H_WebClick)

```
// SDK初始化参数配置
autoTrackConfig: {
    //是否开启触达图, true表示开启, 自动采集H_WebClick事件
    //默认只有点击 a input button textarea 四种元素时, 才会触发该事件
    clickAutoTrack: true
}
```

1.4 视区停留(H_WebStay)

```
// SDK初始化参数配置
autoTrackConfig: {
    //是否开启触达图, true表示开启, 自动采集H_WebStay事件
    stayAutoTrack: true
}
```

2. 其他元素类型的元素点击事件采集

2.1 支持div类型元素的自动采集

在原来的全埋点(采集a、button、input、textarea标签) 基础上新增对div标签的采集,采集规则为:

```
1. div为叶子结点(无子元素)时采集div的点击
```

2. div中有且只有样式标签 (['mark','strong','b','em','i','u','abbr','ins','del','s','sup']) 时,点击div或者样式标签都采集div的点击

```
通过 collectTags 配置是否开启div的全埋点采集,默认不采集
```

```
autoTrackConfig: {
    collectTags: {
        div: true
    }
}
```

2.2 支持任意类型元素的自动采集

通过 collectTags 配置是否开启其他任意元素的全埋点采集(默认不采集), div通过配置最多采集3层嵌套情况

```
autoTrackConfig: {
    clickAutoTrack: true,
    collectTags: {
        div: {
            maxLevel: 1, //默认是1, 可配置范围是[1,3]
        },
        img: true,
        li: true,
        //...其他标签
```

}

}

2.3 配置特殊属性支持其他类型元素的自动采集

2.3.1 配置特殊属性: data-hinadata-click

```
开启全埋点时,需要自动采集点击事件的元素增加属性: data-hinadata-click
```

```
<div name='test' data-hinadata-click> 测试点击元素 </div>
 测试元素
```

2.3.2 配置页面元素点击的自定义属性

开启全埋点时,支持配置带有指定属性的页面元素点击,自动采集点击事件

```
autoTrackConfig: {
    trackAttribute: ['auto-click-track'],
    clickAutoTrack: true,
    stayAutoTrack: true,
}
```

3. 全埋点相关参数配置

3.1 Web元素点击

autoTrackConfig相关参数,提供对于H_WebClick事件的自定义设置处理

参数	默认值	说明	
clickAutoTrack	true	是否开启自动点击采集,默认true为开启,设置false可算	
loadTimeout 1000(ms)		设置多少毫秒后开始渲染点击	
isCollectUrl	_	返回true则采集当前页面的元素点击事件, 返回false则不采集	
collectElement	_	点击页面元素触发这个函数来判断是否需要采集这个元量 返回true表示采集,返回false则不采集	
addCustomProperty	—	在当前H_WebClick事件中添加自定义属性	
isCollectInput –		返回true会采集input内容,返回false则不会采集	

```
elemer發發elector 默认值 SDK默认以元素ID为选择器采集事件,
说明 如果不想用ID作为选择器,可以设置参数为false
```

```
autoTrackConfig: {
clickAutoTrack: true,
loadTimeout: 3000,
isCollectUrl: function(){
//只采集首页点击事件
if(location.href == 'xxx.com/index'){
return true;
}
return false;
},
collectElement: function(element){
//如果这个元素有属性hn-disabled=true,则不采集该元素
if(element.getAttribute('hn-disabled') == 'true' ){
       return false;
}else{
return true;
}
},
addCustomProperty: function(element){
//如果点击元素有name=test属性,点击事件中会增加自定义属性price:120
if(element.getAttribute('name') == 'test'){
return {
price: 120
}
}
},
isCollectInput: function(element){
//如果元素id是 hina, 则采集input内容
if(element.id == 'hina'){
return true;
}
},
element_selector: false
}
```

3.2 Web视区停留

stayAutoTrackConfig相关参数

参数	默认值	说明
isCollectUrl	_	返回true则采集当前页面的视区停留事件,返回false则不采集

autoTrackConfig相关参数,提供对于H_WebStay事件的自定义设置处理

参数	默认值	说明
----	-----	----

参数	默认值	说明
stayAutoTrack	true	是否开启页面停留采集,默认true为开启, 设置false可关闭
stayDelayTime	4000(ms)	有效停留时长超过多少毫秒过开始采集
maxStayPageDuration	18000(ms)	预置属性停留时长 event_duration 的最大值, 默认 18000 秒,5 小时

```
stayAutoTrackConfig: {
    isCollectUrl: function(){
        //只采集首页停留事件
        if(location.href == 'xxx.com/index'){
            return true;
        }
        return false;
    }
},
autoTrackConfig: {
    stayAutoTrack: true,
    stayDelayTime: 4000,
    maxStayPageDuration: 18000
}
```

使用说明

帮助中心 > 技术文档 > 客户端SDK > Web Js SDK > 使用说明

1. 配置项目数据接收地址

在web中引入SDK时,需要手动将 server_url 参数设置为相应项目的数据接收地址。 说明:采集上报地址需在海纳云平台,进行在线申请,请参考【采集上报地址获取】;

2. 常用功能

在使用常用功能之前,建议你先了解用户识别规则;SDK默认会使用随机生成的字符串作为访客ID,并持久化存储访客ID在本地;用户未登录之前,会以访客ID作为身份识别ID。注意:访客 ID 在本地缓存清理时会改变。

2.1 设置用户UID

在用户产生登录行为时,可调用 setUserUId 来设置用户的账号 ID. SDK优先以账号 ID 作为身份标识,设置后的账号 ID 将会 被保存,多次调用 setUserUId 将会覆盖先前的账号 ID :

```
// 用户的登录唯一标识,此数据对应上报数据里的 distinct_id,此时 distinct_id的值为 ABC 
hina.setUserUId("ABC");
```

2.2 发送事件

您可以直接调用 track 上传自定义事件,并添加自定义属性,此处以用户购买某商品作为范例:

```
hina.track('BuyProduct', {
    ProductName: "Books",
    ProductPrice: 23.5,
});
```

2.3 设置公共事件属性

公共事件属性指的就是每个事件都会带有的属性,您可以调用 registerCommonProperties 来设置公共事件属性,我们推荐 您在发送事件前,先设置公共事件属性.

```
hina.registerCommonProperties({
    name: 'AAA'
    age: 30,
})
```

公共事件属性将会被保存到缓存中,无需每次打开网页时调用。如果调用 registerCommonProperties 上传了先前已设置过的公共事件属性,则会覆盖之前的属性。

2.4 设置用户属性

对于一般的用户属性,您可以调用 userSet 来进行设置,使用该接口上传的属性将会覆盖原有的属性值,如果之前不存在 该用户属性,则会新建该用户属性

hina.userSet({ phone:'137xxxx1234' });

3. 调试查看事件信息

测试阶段在初始化代码中设置	showLog 为 true ,页面上有事件触发时,浏览器开发者工具会打印采集的事件信息	1
hina.init({		
serverUrl: 'xxx',		
})		

事件数据发送成功时,在浏览器开发者工具的Network(网络)模块,可以看到网络请求,如果状态码为200,则代表事件数据发送 成功

API详解

帮助中心 > 技术文档 > 客户端SDK > Web Js SDK > API详解

1. SDK初始化参数

参数	默认值	说明
serverUrl	无	数据接收地址
autoTrackConfig	无	默认配置表示不自动采集元素点击事件和页面停留事件, 配置成 {} 表示开启 H_WebClick 采集 和 H_WebStay 自动采集,默认 H_WebClick 只采集 a,button,input , textarea 四个 dom 元素的点击事件。
showLog	false	设置 true 后会在网页控制台打 logger, 会显示发送的数据,设置 false 表示不显示。
max_string_length	500	通用字符串最大长度,超过部分会被截取丢弃(由于超过 7000 的字符串会导致 url 超长发不出去,所以限制长度
sendType	image	表示使用image请求方式发数据, 默认选用'image'图片get请求方式发送数据, 支持'ajax'和'beacon'方式发送, 这两种默认都是post请求方式,beacon方式兼容性差
callback_timeout	200, 单位毫秒	表示函数回调超时时间,如果数据发送超过 callback_timeout还未返回结果,会强制执行回调函数
queue_timeout	300, 单位毫秒	表示队列发送超时时间, 如果数据发送超过queue_timeout还未返回结果, 会强制发送下一条数据
datasend_timeout	3000 , 单位毫秒	表示数据发送超时时间,如果数据发送超过 datasend_timeout 还未返回结果,会强制取消该请求
preset_properties	{}	是否开启 H_latest 相关的最近一次相关事件属性采集以及配置 H_url 作为公共属性,默认值为一个对象
isSinglePage	false	表示是否开启单页面自动采集 H_pageview 功能,SDK 会在 url 改变之后自动采集web页面浏览事件 H_pageview

2. 设置用户ID

SDK 实例默认会使用随机数作为每个用户的默认访客 ID,该 ID 将会作为用户在未登录状态下身份识别 ID。需要注意的是,访客 ID 在用户清理缓存以及更换设备时将会变更

2.1 设置访客ID

一般情况下,您不需要自定义访客 ID,系统会自动生成自定义ID

hina.setDeviceUId('ABC')

如果需要获得访客 ID,可以调用 getDeviceUId 获取

//返回访客ID

const distinctId = hina.getDeviceUId();

2.2 设置账号ID

```
在用户进行登录时,可调用 setUserUId 来设置用户的账号 ID, SDK将会以账号 ID 作为身份识别 ID,并且设置的账号 ID 将 A在调用 removeUserUId 之前一直保留,多次调用 setUserUId 将覆盖先前的账号 ID
```

```
// 用户的登录唯一标识
hina.setUserUId('xxx')
```

3. 用户属性

3.1 设置用户属性

直接设置用户的属性,如果存在则覆盖

hina.userSet({ name:'xx' });

3.2 固定初始值的属性

如果不存在则设置,存在就不设置

hina.userSetOnce({ name:'xx' });

3.3 数值类型的属性

对当前用户的属性做递增或者递减

```
// 表示num递增2
hina.userAdd({ 'num': 2 })
//表示num递减2
hina.userAdd({ 'num': -2 })
```

3.4 集合类型的属性

给数组属性添加值,通过userSet只能改变属性的值,如果这个属性是数组类型的,你不想完全改变这个值,只想做添加操作可以使用此方法

```
hina.userAppend({ fruit:['apple'] });
//给fruit增加一个值
hina.userAppend({ fruit:['lemon'] });
//或者可以这样写
hina.userAppend({ fruit: 'lemon' });
```

4. 发送事件

在 SDK 初始化完成之后,您就可以进行数据埋点,收集用户的的行为信息

4.1 普通事件

```
可以调用 track 来上传事件
hina.track(
//事件名称
'BuyProduct',
//需要上传的事件属性
{
ProductName: "Books",
ProductPrice: 23.5,
}
);
```

4.2 设置公共事件属性

4.2.1 设置事件公共属性

```
对于所有事件都需要添加的属性,可在初始化 SDK 后,调用 registerCommonProperties 将属性注册为公共属性
hina.registerCommonProperties({
title: document.title
})
当设置动态公共属性的时候,需要使用函数类型作为属性值
函数返回值只能是string、bool、number、date、array类型
```

```
hina.registerCommonProperties({
    afterTwoHour: function(){
        return new Date().addHours(2)
    }
})
```

4.2.2 获取预置属性

此方法可获取,页面地址,页面标题,前向地址,SDK 类型及版本,屏幕宽高,最近一次的相关属性

```
hina.quick('isReady',function(){
    const presetProperties = hina.getPresetProperties();
})
```

4.3 数据批量发送

```
// 默认不开启批量发送
batch_send:false,
// 开启批量发送
batch_send:true,
//或者
batch_send:{
    datasend_timeout: 6000, //一次请求超过多少毫秒的话自动取消, 防止请求无响应
    send_interval: 6000, //间隔多少毫秒发一次数据
    storage_length: 200, // 存储 localStorage 条数最大值, 默认: 200 。如 localStorage 条数超过该值, 则使
用 image 方式立即发送数据
    },
```

注意:

- 1. 批量发送功能和回调函数功能不可以同时使用,比如track加了callback,使用批量发送后callback不会执行
- 2. 批量发送默认使用 ajax 的方式发送数据,如果不支持跨域ajax发送数据,会使用img且实时发送数据
- 3. 如果localStorage 里已经存了超过 200 条数据,会导致批量发送功能失效,localStorage 中只保存这 200 条数据,新产生的数据使用 img 且实时发送数据的方式。当进入同域名的新页面时,会自动检查缓存中是否有数据,如果有会继续发送缓存的数据
- 4. useAppBridge和batch_send只能选择一个,开启了打通App就不能再用批量发送

常见问题

帮助中心 > 技术文档 > 客户端SDK > Web Js SDK > 常见问题

1. 单页面的页面标题H_title问题

对于单页面应用项目,SDK全埋点预置事件采集的页面标题默认为document.title属性,可能存在异常 具体问题:

1. title如果没有更新赋值,获取title一直是首页的title不会发生变化更新

```
// 切换页面之前完成title值的更新
router.beforeEach(()=>{
    document.title = 'new title'
})
```

2.title是页面动态生成赋值的,切换页面发送的H_pageview事件的H_title值为上一个页面的title,解决方案可以参考【全埋 点1.2.2手动模式】

更新日志

版本:1.0.0

【发布日期】: 2023-02-23

【更新内容】:

1. 发布SDK初版,支持全埋点和自定义埋点。

小程序SDK

SDK 功能清单

一级功能	二级功能	具体功能 / 描述	
た,0日	自动标识匿名用户	自动使用设备 ID 标识匿名用户	
		支持替换为自定义的匿名 ID	
柳朳用广	标识登录用户	使用登录 ID 标识登录用户	
	设置用户属性		
	白动变焦边久/户自	自动采集默认的设备信息,屏幕宽高、系统版本号等	
	日列木朱仅笛信忌	支持自动采集经纬度信息	
		小程序启动	
	全埋点	小程序显示	
		小程序隐藏	
爫住 쐐セ		小程序页面浏览	
木朱奴伍		小程序页面点击	
		小程序页面离开	
		小程序页面分享	
	公共属性	静态公共属性	
		动态公共属性	
		清除公共属性	
存储数据	设置发送时间	设置多少秒发一次数据	
┝╆╔╫╓┼┲	自动上报数据	SDK 在满足一定条件后自动发送事件数据	
1区贸灯店	手动上报数据	SDK 支持手动上报事件数据	
调试功能	调试日志	显示 / 关闭调试日志	

高级功能	数据加密	数据加密之后再进行存储和发送
	与 H5 打通	小程序用户ID传给H5
	合规	支持合规要求
	SDK 采集控制	控制 SDK 开启或关闭

微信小程序SDK

微信小程序 SDK

集成文档 使用说明 API详解 更新日志

集成文档

帮助中心 > 技术文档 > 客户端SDK > 小程序 SDK > 微信小程序 SDK >集成文档

1. 集成SDK

下载压缩包

说明:请根据自身开发框架,选择正确的初始化流程.

原生 uni-app编译为小程序 Taro mpvue WePY

原生小程序

app.js

```
var hn = require('/dist/wechat/hinaData.cjs.js');
// 配置初始化参数
hn.setPara({
  name: 'hn',
   serverUrl: '您的数据接收地址',
  // 全埋点控制开关
   autoTrack:{
      appLaunch: true, // 默认为 true, false 则关闭 H_MPLaunch 事件采集
      appShow: true, // 默认为 true, false 则关闭 H_MPShow 事件采集
      appHide: true, // 默认为 true, false 则关闭 H_MPHide 事件采集
      pageShow: true, // 默认为 true, false 则关闭 H_MPViewScreen 事件采集
      pageShare: true, // 默认为 true, false 则关闭 H_MPShare 事件采集
      mpClick: false, // 默认为 false, true 则开启 H_MPClick 事件采集
      mpFavorite: true, // 默认为 true, false 则关闭 H_MPAddFavorites 事件采集
      pageLeave: false // 默认为 false, true 则开启 H_MPPageLeave事件采集
   },
   // 自定义渠道追踪参数, 如 source_channel: ["custom_param"]
   source_channel: [],
   // 是否允许控制台打印查看埋点数据(建议开启查看)
  show_log: true,
   // 是否允许修改 onShareAppMessage 里 return 的 path, 用来增加(登录 ID, 分享层级, 当前的 path), 在 app onS
how 中自动获取这些参数来查看具体分享来源、层级等
   allow_amend_share_path: true
});
// 初始化 SDK
if(不使用 openid 作为匿名 ID){
   hn.init();
}
// 如果需要使用 openid 作为匿名 ID, 请单独获取 openid 之后调用 hn.setOpenid() 方法
wx.request({
   url: '后端获取 OpenID 的请求',
   success: function(res){
      if(res.OpenID){
        hn.identify(res.OpenID, true);
```

```
},
complete: function(){
    // 如果获取 openid 失败, SDK 会以 UUID 作为匿名 ID 发数据
    hn.init();
}
});
App({
    onLaunch : function( options ){},
    onShow : function( options ){},
    onHide : function(){}
});
```

uni-app

main.js

```
import Vue from 'vue';
import hn from '/dist/wechat/hinaData.esm.js'
// 配置初始化参数
hn.setPara({
name: 'hn',
serverUrl: '您的数据接收地址',
// 全埋点控制开关
autoTrack:{
      appLaunch: true, // 默认为 true, false 则关闭 H_MPLaunch 事件采集
      appShow: true, // 默认为 true, false 则关闭 H_MPShow 事件采集
      appHide: true, // 默认为 true, false 则关闭 H_MPHide 事件采集
      pageShow: true, // 默认为 true, false 则关闭 H_MPViewScreen 事件采集
      pageShare: true, // 默认为 true, false 则关闭 H_MPShare 事件采集
      mpClick: false, // 默认为 false, true 则开启 H_MPClick 事件采集
      mpFavorite: true, // 默认为 true, false 则关闭 H_MPAddFavorites 事件采集
      pageLeave: false // 默认为 false, true 则开启 H_MPPageLeave事件采集
},
// 自定义渠道追踪参数, 如 source_channel: ["custom_param"]
source_channel: [],
// 是否允许控制台打印查看埋点数据(建议开启查看)
show_log: true,
// 是否允许修改 onShareAppMessage 里 return 的 path, 用来增加(登录 ID, 分享层级, 当前的 path), 在 app onS
how 中自动获取这些参数来查看具体分享来源、层级等
  allow_amend_share_path: true
});
// 初始化 SDK
if(不使用 openid 作为匿名 ID){
   hn.init();
}
// 如果需要使用 openid 作为匿名 ID, 请单独获取 openid 之后调用 hn.setOpenid() 方法
wx.request({
  url: '后端获取 OpenID 的请求',
   success: function(res){
      if(res.OpenID){
       hn.identify(res.OpenID, true);
      }
},
  complete: function(){
      // 如果获取 openid 失败, SDK 会以 UUID 作为匿名 ID 发数据
      hn.init();
```

集成文档

}

```
});
import App from './App';
Vue.config.productionTip = false;
App.mpType = 'app';
const app = new Vue({
    ...App
});
app.$mount();
```

Taro

app.jsx

```
import Taro, { Component } from '@tarojs/taro'
import Index from './pages/index'
import hn from '/dist/wechat/hinaData.esm.js'
// 配置初始化参数
hn.setPara({
name: 'hn',
serverUrl: '您的数据接收地址',
// 全埋点控制开关
autoTrack:{
      appLaunch: true, // 默认为 true, false 则关闭 H_MPLaunch 事件采集
      appShow: true, // 默认为 true, false 则关闭 H_MPShow 事件采集
      appHide: true, // 默认为 true, false 则关闭 H_MPHide 事件采集
      pageShow: true, // 默认为 true, false 则关闭 H_MPViewScreen 事件采集
      pageShare: true, // 默认为 true, false 则关闭 H_MPShare 事件采集
      mpClick: false, // 默认为 false, true 则开启 H_MPClick 事件采集
 mpFavorite: true, // 默认为 true, false 则关闭 H_MPAddFavorites 事件采集
pageLeave: false // 默认为 false, true 则开启 H_MPPageLeave事件采集
},
// 自定义渠道追踪参数, 如 source_channel: ["custom_param"]
source_channel: [],
// 是否允许控制台打印查看埋点数据(建议开启查看)
show_log: true,
// 是否允许修改 onShareAppMessage 里 return 的 path, 用来增加(登录 ID, 分享层级, 当前的 path), 在 app onS
how 中自动获取这些参数来查看具体分享来源、层级等
  allow_amend_share_path: true
});
// 初始化 SDK
if(不使用 openid 作为匿名 ID){
   hn.init();
}
// 如果需要使用 openid 作为匿名 ID, 请单独获取 openid 之后调用 hn.setOpenid() 方法
wx.request({
 url: '后端获取 OpenID 的请求',
  success: function(res){
if(res.OpenID){
       hn.identify(res.OpenID,true);
}
},
```

```
complete: function(){
       // 如果获取 openid 失败, SDK 会以 UUID 作为匿名 ID 发数据
       hn.init();
 }
});
class App extends Component {
   config = {
       pages: [
           'pages/index/index'
       ],
       window: {
           backgroundTextStyle: 'light',
           navigationBarBackgroundColor: '#fff',
           navigationBarTitleText: 'WeChat',
           navigationBarTextStyle: 'black'
    }
   }
   constructor(){
       super();
   }
}
```

mpvue

main.js

```
import hn from '/dist/wechat/hinaData.esm.js'
import Vue from 'vue';
import App from './App';
// 配置初始化参数
hn.setPara({
name: 'hn',
serverUrl: '您的数据接收地址',
// 全埋点控制开关
autoTrack:{
 appLaunch: true, // 默认为 true, false 则关闭 H_MPLaunch 事件采集
      appShow: true, // 默认为 true, false 则关闭 H_MPShow 事件采集
      appHide: true, // 默认为 true, false 则关闭 H_MPHide 事件采集
      pageShow: true, // 默认为 true, false 则关闭 H_MPViewScreen 事件采集
      pageShare: true, // 默认为 true, false 则关闭 H_MPShare 事件采集
      mpClick: false, // 默认为 false, true 则开启 H_MPClick 事件采集
 mpFavorite: true, // 默认为 true, false 则关闭 H_MPAddFavorites 事件采集
pageLeave: false // 默认为 false, true 则开启 H_MPPageLeave事件采集
},
// 自定义渠道追踪参数, 如 source_channel: ["custom_param"]
source_channel: [],
// 是否允许控制台打印查看埋点数据(建议开启查看)
show_log: true,
// 是否允许修改 onShareAppMessage 里 return 的 path, 用来增加(登录 ID, 分享层级, 当前的 path), 在 app onS
how 中自动获取这些参数来查看具体分享来源、层级等
 allow_amend_share_path: true
});
// 初始化 SDK
if(不使用 openid 作为匿名 ID){
  hn.init();
}
```

集成文档

```
// 如果需要使用 openid 作为匿名 ID, 请单独获取 openid 之后调用 hn.setOpenid() 方法
wx.request({
    url: '后端获取 OpenID 的请求',
    success: function(res){
        if(res.OpenID){
            hn.identify(res.OpenID,true);
        }
    },
    complete: function(){
            // 如果获取 openid 失败, SDK 会以 UUID 作为匿名 ID 发数据
            hn.init();
    }
});
Vue.config.productionTip = false;
```

```
App.mpType = 'app';
```

const app = new Vue(App);
app.\$mount();

WePY

app.wpy

```
import hn from '/dist/wechat/hinaData.esm.js'
import wepy from 'wepy';
// 配置初始化参数
hn.setPara({
name: 'hn',
serverUrl: '您的数据接收地址',
// 全埋点控制开关
autoTrack:{
appLaunch: true, // 默认为 true, false 则关闭 H_MPLaunch 事件采集
      appShow: true, // 默认为 true, false 则关闭 H_MPShow 事件采集
      appHide: true, // 默认为 true, false 则关闭 H_MPHide 事件采集
      pageShow: true, // 默认为 true, false 则关闭 H_MPViewScreen 事件采集
      pageShare: true, // 默认为 true, false 则关闭 H_MPShare 事件采集
      mpClick: false, // 默认为 false, true 则开启 H_MPClick 事件采集
      mpFavorite: true, // 默认为 true, false 则关闭 H_MPAddFavorites 事件采集
      pageLeave: false // 默认为 false, true 则开启 H_MPPageLeave事件采集
},
// 自定义渠道追踪参数, 如 source_channel: ["custom_param"]
source_channel: [],
// 是否允许控制台打印查看埋点数据(建议开启查看)
show_log: true,
// 是否允许修改 onShareAppMessage 里 return 的 path, 用来增加(登录 ID, 分享层级, 当前的 path), 在 app onS
how 中自动获取这些参数来查看具体分享来源、层级等
  allow_amend_share_path: true
});
// 初始化 SDK
if(不使用 openid 作为匿名 ID){
   hn.init();
}
// 如果需要使用 openid 作为匿名 ID, 请单独获取 openid 之后调用 hn.setOpenid() 方法
wx.request({
url: '后端获取 OpenID 的请求',
```

```
success: function(res){
    if(res.OpenID){
        hn.identify(res.OpenID,true);
    }
    },
    complete: function(){
        // 如果获取 openid 失败, SDK 会以 UUID 作为匿名 ID 发数据
        hn.init();
    }
});
export default class extends wepy.app {
    onLaunch(option) {}
```

}

附件

hiCloudSDK.zip

使用说明

帮助中心 > 技术文档 > 客户端SDK > 小程序 SDK > 微信小程序 SDK >使用说明

1. 配置项目数据接收地址

在web中引入SDK时,需要手动将 serverUrl 参数设置为相应项目的数据接收地址。 说明:采集上报地址需在海纳云平台,进行在线申请,请参考【采集上报地址获取】;

2. 常用功能

在使用常用功能之前,建议你先了解用户识别规则;SDK默认会使用随机生成的字符串作为访客ID,并持久化存储访客ID在本地;用户未登录之前,会以访客ID作为身份识别ID。注意:访客 ID 在本地缓存清理时会改变。

2.1 设置用户UID

在用户产生登录行为时,可调用 setUserUId 来设置用户的账号 ID. SDK优先以账号 ID 作为身份标识,设置后的账号 ID 将会 被保存,多次调用 setUserUId 将会覆盖先前的账号 ID。

2.2 发送事件

您可以直接调用 track 上传自定义事件,并添加自定义属性,此处以用户购买某商品作为范例:

```
hina.track('BuyProduct', {
    ProductName: "Books",
    ProductPrice: 23.5,
});
```

2.3 设置公共事件属性

公共事件属性指的就是每个事件都会带有的属性,您可以调用 registerCommonProperties 来设置公共事件属性,我们推荐 您在发送事件前,先设置公共事件属性.

```
hina.registerCommonProperties({
    name: 'AAA'
    age: 30,
})
```

公共事件属性将会被保存到缓存中,无需每次打开网页时调用。如果调用 registerCommonProperties 上传了先前已设置过的公共事件属性,则会覆盖之前的属性。

2.4 设置用户属性

对于一般的用户属性,您可以调用 userSet 来进行设置,使用该接口上传的属性将会覆盖原有的属性值,如果之前不存在 该用户属性,则会新建该用户属性

```
hina.userSet({ phone: '137xxxx1234' });
```

3. 调试查看事件信息

测试阶段在初始化代码中设置 show_log 为 true ,页面上有事件触发时,浏览器开发者工具会打印采集的事件信息 hina.init({ ... serverUrl: 'xxx', show_log: true ... })

事件数据发送成功时,在浏览器开发者工具的Network(网络)模块,可以看到网络请求,如果状态码为200,则代表事件数据发送 成功

API详解

帮助中心 > 技术文档 > 客户端SDK > 小程序 SDK > 微信小程序 SDK >API详解

1.SDK初始化参数

参数	类型	默认值	含义
name	String	hn	SDK 使用的一个默认的全局变量, 会注册在 App 全局函数内,在 Pag 中可以通过 app[name].track 来使序
server_url	String	数据接收地址	
autoTrack	Object	无	是否开启自动采集
show_log	Boolean	true	是否打印 log
send_timeout	Number	1000	请求发送超时时间 (如果一个请求发送后, 超过规定时间没响应, 则继续发送下一条数据)
use_client_time	Boolean	false	是否使用客户端时间
batch_send	Boolean	true	小程序中是否使用批量发送数据功
datasend_timeout	Number	3000	请求发送取消时间

参数	类型	默认值	含义
source_channel	Array	无	需要解析的渠道参数
is_persistent_save	Object	无	是否需要将最近一次渠道信息保存 wxStorage 中
preset_properties	Object	无	配置采集指定预置属性
autotrack_exclude_page	Object	无	配置指定页面不采集 H_MPViewScreen 页面浏览事件
framework	Object	无	如果使用 Taro 框架开发小程序, 元素点击事件会触发多次,配置 Taro 参数之后可以只采集一次
preset_events	Object	无	预置事件的自定义控制

2.全埋点采集逻辑

事件名称	生命周期	采集时机
H_MPLaunch (小程序启动)	App.onLaunch	小程序进程被杀死, 重新打开时会触发
H_MPShow (小程序显示)	App.onShow	小程序启动,或从后台进入前台显示
H_MPHide (小程序进入后台)	App.onHide	点击小程序右上角退出按钮、 微信进入后台、手机锁屏、 小程序进程被杀死时
H_MPViewScreen		小程序启动打开页面、

(小程序页面浏览 事件名称)	Page on Show 生命周期	尘程序内打开页面、 采集时机 从后台进入前台打开页面时触发
H_MPShare (小程序分享)	page.onShareAppMessage	设置这个函数后,点击分享按钮触发
H_MPClick (小程序元素点击)		当 Page 中定义的事件处理函数被触发时采集
H_MPAddFavorites (小程序收藏)	page.onAddToFavorites	点击小程序收藏按钮时触发
H_MPPageLeave (小程序页面离开)	page.onHide 或 page.onUnload	小程序页面隐藏或卸载时触发
< >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>		
更新日志

帮助中心 > 技术文档 > 客户端SDK > 小程序 SDK > 微信小程序 SDK >更新日志

版本:1.0.0

【发布日期】: 2023-02-23

【更新内容】:

1. 发布SDK初版,支持代码埋点和自动埋点;

【注意事项】: 暂无

跨平台框架

Flutter 插件

集成文档

1. 配置依赖

在 Flutter 项目的 pubspec.yaml 文件, dependencies里添加插件配置:

dependencies:
 flutter:
 sdk: flutter
 # 添加插件配置
 hina_flutter_plugin:
 git:
 url: git@github.com:liujxgitlab/hina-flutter-plugin.git

2. 安装插件

执行 flutter packages get 命令安装插件:

flutter pub get

使用说明

1. 初始化SDK

说明:使用init方法,按如下方式进行初始化。

参数key (String类型)	参数value类型	参数说明
serverUrl	String	【必填】配置数据上报地址 (<u>在线获取</u>)
flushInterval	int	【可选】自定义上报时间间隔
flushPendSize	int	【可选】自定义上报缓存条数
enableLog	bool	【可选】开启日志
maxCacheSizeForAndroid	long	【可选】自定义缓存上限 (Android用)
maxCacheSizeForIOS	int	【可选】自定义缓存上限 (iOS用)
autoTrackTypePolicy	Set <haautotracktype>?</haautotracktype>	【可选】开启全埋点
networkTypePolicy	Set <hanetworktype>?</hanetworktype>	【可选】自定义上报网络条件
enableJSBridge	bool	【可选】App打通H5

HinaFlutterPlugin.init(
 serverUrl: "数据接收地址",
 autoTrackTypeList: {
 HAAutoTrackType.APP_START,
 HAAutoTrackType.APP_END
 });

注意:

- 1、通常原生端和 Flutter 端只要初始化一次即可。若选择在 Flutter 端初始化,就不需要再在原生端初始化了。
- 2、采集上报地址(serverUrl)获取方式,请参考【采集上报地址获取】;
- 3、更多初始化配置,请参考【API详解】;

2. 代码埋点 (手动采集)

说明:使用track()方法进行直接埋点,埋点事件支持添加自定义属性。

var eventName = 'increment'; var properties = {'ProductID': '123456', 'ProductCatalog': 'increment counter'}; HinaFlutterPlugin.track(eventName, properties);

API详解

1. 代码埋点 (手动采集)

用于记录埋点事件。

1.1 直接埋点

说明:使用track()方法进行直接埋点,埋点事件支持添加自定义属性。

```
var eventName = 'increment';
var properties = {'ProductID': '123456', 'ProductCatalog': 'increment counter'};
HinaFlutterPlugin.track(eventName, properties);
```

1.2 时长埋点

```
说明:需要成对调用计时器的开始和结束方法,以此来实现对统计时长的事件采集;
*在事件开始时,调用 trackTimerStart("ViewProduct"),该方法并不会真正发送事件;
*在事件结束时,调用 trackTimerEnd("ViewProduct", properties),SDK 会触发 "Event" 事件,并自动将事件持续时间记录在事件属性 "$event_duration" 中。
```

```
// 调用 trackTimerStart("ViewProduct") 标记(浏览商品)事件启动时间
HinaFlutterPlugin.trackTimerStart("ViewProduct");
```

// 在属性中记录商品 ID
var properties = <String, dynamic>{};
properties["product_id"] = PRODUCT_ID;
// 内部会调用 track, 记录 ViewProduct 事件, 并在属性 event_duration 中记录用户浏览商品的时间
HinaFlutterPlugin.trackTimerEnd("ViewProduct", properties);

2. 全埋点 (自动采集)

SDK 可以自动采集一些用户行为,如 App 启动、退出、浏览页面、控件点击,共计四种。

说明:在初始化时,通过autoTrackTypeList参数进行配置开启,不配置则视为关闭全埋点。

```
//在初始化时,配置开启
HinaFlutterPlugin.init(
...
autoTrackTypeList: {
HAAutoTrackType.APP_START,
HAAutoTrackType.APP_END
});
```

注意:浏览页面、控件点击仅对原生工程支持,全Flutter工程仅支持采集APP启动和退出;

3. 事件属性

在进行埋点事件追踪时,您可以根据需求对埋点事件进行属性的定义。目前 SDK 中提供了公共属性用于给每个埋点事件添加属 性。

3.1 获取预置属性

说明:如需了解和使用预置属性,可以通过此方法获取预置属性。

```
// 返回Future<Map<String, dynamic>?>
var result = await HinaFlutterPlugin.getPresetProperties();
```

3.2 设置公共属性

说明:公共属性是指对于所有事件都需要添加的属性,设置之后 SDK 会在每次触发埋点时,自动获取并添加到触发的事件中。

```
var properties = {'app_name': '张三啦啦啦'};
HinaFlutterPlugin.registerCommonProperties(properties);
```

4. 用户属性

4.1 设置用户属性

说明:同一个 key 多次设置时, value 值会进行覆盖替换。

```
var properties = <String, dynamic>{};
properties["user_name"] = '张三';
HinaFlutterPlugin.userSet(properties);
```

4.2 固定初始值的属性

说明:同一个 key 多次设置时, value 值只会记录初次设定的值;适用于为用户设置首次激活时间、首次注册时间等属性。

```
var properties = <String, dynamic>{};
properties["user_register_time"] = '10101010';
HinaFlutterPlugin.userSetOnce(properties);
```

4.3 数值类型的属性

说明:同一个 key 多次设置时, value 值(数值类型)进行求和计算;常用于记录用户付费次数、付费额度、积分等属性。

```
var properties = <String, num>{};
properties['score'] = 8;
HinaFlutterPlugin.userAdd(properties);
```

4.4 集合类型的属性

说明:同一个 key 多次设置时,value 值(字符串类型)都会被添加到集合里;适用于用户喜爱的电影、用户点评过的餐厅等属 性。

```
List<String> values = ['电影1', '电影2'];
// 设定用户观影列表属性, 设定后属性 "Movies" 为: ["电影1", "电影2"]
HinaFlutterPlugin.userAppend("movie", values);
```

4.5 属性取消

说明:取消已设置的某个用户属性。

```
var key = 'user_name';
HinaFlutterPlugin.userUnset(key);
```

4.6 清空用户属性

说明:清空当前用户的用户属性。

HinaFlutterPlugin.userDelete();

5. 用户关联

5.1 用户登录

说明:当用户注册成功或登录成功时,需要调用 SDK 的 setUserUld() 方法。

HinaFlutterPlugin.setUserUId("业务userid");

注意:为了准确记录登录用户的行为信息,建议在以下时机各调用一次用户登录方法:

- ·用户在注册成功时
- ·用户登录成功时
- ·已登录用户每次启动 App 时

5.2 获取设备唯一ID

说明:如需获取设备唯一id,请在初始化SDK后调用getDeviceUld()方法进行获取。

//返回Future<String?>
var deviceUId = await HinaFlutterPlugin.getDeviceUId();

5.3 自定义设备唯一ID

说明:默认情况下,SDK 会生成设备唯一ID 并可以保证该ID的唯一性,如需替换SDK默认分配的,配置如下:

```
//通过API配置
```

HinaFlutterPlugin.setDeviceUId(String deviceUId);

注意: SDK 默认使用 AndroidId 作为设备 ID , 如果 AndroidId 获取不到则获取随机的 UUID。

6. 数据存储与发送

在每次调用 track()、setUserUid()、userSet() 等方法时,SDK 会将埋点事件保存在数据库中,并会检查如下条件,以判断是否向服务器上传数据:

```
    1.是否是 WIFI/26/36/46/56 网络条件
    2.是否满足发送条件之一:

            a.与上次发送的时间间隔是否大于 flushInterval
            b.本地缓存日志数目是否大于 flushPendSize
            c.事件类型为 setUserUid() 方法触发的 $SignUp 事件
```

6.1 上报条件

6.1.1 缓存条数

```
说明:设置本地缓存日志的最大条目数,默认本地埋点数据缓存为100条。
```

```
//初始化时配置
HinaFlutterPlugin.init(
...
flushPendSize: 1,
...);
```

//或者在初始化后, 通过API配置 HinaFlutterPlugin.setFlushPendSize(int size);

6.1.2 发送间隔

说明:设置埋点数据发送的间隔,默认为15s。

6.1.3 设置数据的网络上传策略

说明:默认情况下,在 WIFI/3G/4G/5G 网络条件下,SDK 都会尝试去同步数据。可以自由组合来指定发送数据的网络策略。

//或者在初始化后,通过API配置; HinaFlutterPlugin.setFlushNetworkPolicy(Set<HANetworkType>? networkTypeList);

6.2 立即上报 (强制上报)

说明:如果追求数据采集的时效性,调用flush(),即可立即执行上报。

HinaFlutterPlugin.flush();

注意:在 App 进入后台状态或监听到网络切换有网络时,SDK 会调用 flush() 方法,将缓存的数据发送。

6.3 本地数据缓存上限值

说明:Android和iOS设置维度不一样,如有需要,请分开设置;

HinaFlutterPlugin.init(

```
...
//android配置空间大小, SDK 本地数据库默认缓存数据的上限值为 32MB
//设置本地缓存上限值,单位 byte,默认为 32MB: 32 * 1024 * 1024,最小 16MB: 16 * 1024 * 1024,
//若小于 16MB,则按 16MB 处理。
maxCacheSizeForAndroid: 32 * 1024 * 1024,
//iOS配置条数,默认10000条
maxCacheSizeForIOS: 10000,
...);
```

注意:当存储数量达到上限值,会依次丢弃老数据,保留最新的数据。

6.4 清空本地缓存事件

说明:删除 App 本地存储的所有事件。

HinaFlutterPlugin.clear();

注意:如果不是特殊要求,请不要调用此方法。

7. 其他功能

7.1 开启App与H5打通功能

说明:初始化 SDK 时,进行如下配置。

HinaFlutterPlugin.init(...

> enableJSBridge: true, ...);

7.2 日志开关

说明:初始化 SDK 时,进行如下配置。

HinaFlutterPlugin.init(... enableLog: true, ...);

更新日志

插件版本:1.0.0

【发布日期】: 2023-03-23

【更新内容】:

1. 发布插件初版,支持代码埋点和自动埋点;

APICloud

集成文档

集成方式

方式一:使用离线模块 (自定义模块)

1、下载离线模块文件(moduleHinaCloud.zip文件)到本地;

2、打开"模块"->"自定义模块",上传本地模块文件,并点击"保存";

3、保存成功后,会在下方显示,如下图,点击右上角的"+"号,即完成添加;

方式二:使用在线模块 (模块库)

1、打开"模块"->"模块库",在搜索框输入"moduleHinaCloud"后,进行搜索;

2、搜索到目标模块,会在下方显示,如下图,点击右上角的"+"号,即完成添加;

使用说明

1. 导入模块

说明:在项目的 js 文件中导入 APICloud 模块。

var moduleHC = api.require('moduleHinaCloud');

2. 初始化SDK

说明:调用模块的init方法进行初始化。

方法名: init()

请求参数:

参数key (String类型)	参数value类型	参数说明
serverUrl	String	【必填】配置采集上报URL(在线获取)
flushInterval	int	【可选】自定义上报时间间隔
flushPendSize	int	【可选】自定义上报缓存条数
enableLog	bool	【可选】开启日志
maxCacheSizeForAndroid	long	【可选】自定义缓存上限(Android用)
maxCacheSizeForIOS	int	【可选】自定义缓存上限(iOS用)
autoTrackTypePolicy	int	【可选】开启全埋点
networkTypePolicy	int	【可选】自定义上报网络条件

请求示例:

```
var param = {
   serverUrl:"https://xxx",
   flushInterval:1000,
   flushPendSize:1,
   enableLog:true,
   autoTrackTypePolicy:3
};
moduleHC.init(param, function(ret, err){
   alert(JSON.stringify(ret));
});
```

返回参数:无

3. 代码埋点 (手动采集)

说明:使用track()方法进行直接埋点,埋点事件支持添加自定义属性。

参数key (String类型)	参数value类型	参数说明
eventName	String	事件名
properties	json-object	自定义事件属性

请求示例:

```
var param = {eventName:"clickTab1",properties:{who:"william"}};
moduleHC.track(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

返回参数:无

API详解

1. 代码埋点 (手动采集)

用于记录埋点事件。

1.1 直接埋点

说明:使用track()方法进行直接埋点,埋点事件支持添加自定义属性。

方法名: track() 请求参数:

参数key (String类型)	参数value类型	参数说明
eventName	String	事件名
properties	json-object	自定义事件属性

请求示例:

```
var param = {eventName:"clickTab1",properties:{who:"william"}};
moduleHC.track(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

返回参数:无

1.2 时长埋点

```
说明:需要成对调用计时器的开始和结束方法,以此来实现对统计时长的事件采集;
*在事件开始时,调用 trackTimerStart("ViewProduct"),该方法并不会真正发送事件;
*在事件结束时,调用 trackTimerEnd("ViewProduct", properties),SDK 会触发 "Event" 事件,并自动将事件持续时间记
录在事件属性 "$event_duration" 中。
```

方法名: trackTimerStart() 请求参数:

参数key (String类型)	参数value类型	参数说明
eventName	String	事件名

请求示例:

```
var param = {eventName:"view_product"};
moduleHC.trackTimerStart(param, function(ret, err){
```

```
alert(JSON.stringify(ret));
});
```

返回参数:无

方法名: trackTimerEnd() 请求参数:

参数key (String类型)	参数value类型	参数说明
eventName	String	事件名
properties	json-object	自定义事件属性

请求示例:

```
var param = {eventName:"view_product",properties:{product:"商品1"}};
moduleHC.trackTimerEnd(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

返回参数:无

2. 全埋点 (自动采集)

SDK 可以自动采集一些用户行为,如 App 启动、退出、浏览页面、控件点击,共计四种。

说明:在初始化时,通过 autoTrackTypePolicy 参数进行配置开启,不配置则视为关闭全埋点。

```
var param = {
...,
/**
* 设置全埋点采集策略,默认关闭
* TYPE_NONE = 0;//关闭
* APP_START = 1;//启动 1
* APP_END = 1 << 1;//退出 2
* APP_CLICK = 1 << 2;//控件点击 4
* APP_VIEW_SCREEN = 1 << 3;//浏览页面 8
*例:若需要开启 App 启动、退出,则需要设置 1 + 2 = 3
*/
//全埋点开启 App 启动、退出自动采集。
autoTrackTypePolicy:3,
. . .
};
moduleHC.init(param, function(ret, err){
   alert(JSON.stringify(ret));
});
```

注意:浏览页面、控件点击仅对原生工程支持,APICloud 工程仅支持采集APP启动和退出;

3. 事件属性

在进行埋点事件追踪时,您可以根据需求对埋点事件进行属性的定义。目前 SDK 中提供了公共属性用于给每个埋点事件添加属 性。

3.1 获取预置属性

说明:如需了解和使用预置属性,可以通过此方法获取预置属性。

方法名: getPresetProperties()

请求参数:无

请求示例:

```
var param = {};
moduleHC.getPresetProperties(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

返回参数:

参数key (String类型)	参数value类型	参数说明
?	json-object	SDK内部预置参数

返回示例:

```
{
    "app_name":"xxx",
    "app_version":"1.0.0",
    ...
}
```

3.2 设置公共属性

说明:公共属性是指对于所有事件都需要添加的属性,设置之后 SDK 会在每次触发埋点时,自动获取并添加到触发的事件中。

```
方法名: registerCommonProperties()
请求参数:
```

参数key(String类型)	参数value类型	参数说明
properties	json-object	自定义公共属性

请求示例:

```
var param = {properties:{app_name_alias:"张三啦啦啦"}};
moduleHC.registerCommonProperties(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

返回参数:无

4. 用户属性

4.1 设置用户属性

说明:同一个 key 多次设置时, value 值会进行覆盖替换。

方法名: userSet() 请求参数:

参数key(String类型)	参数value类型	参数说明
properties	json-object	自定义用户属性

请求示例:

```
var param = {properties:{gender:"男"}};
moduleHC.userSet(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

返回参数:无

4.2 固定初始值的属性

说明:同一个 key 多次设置时, value 值只会记录初次设定的值;适用于为用户设置首次激活时间、首次注册时间等属性。

方法名: userSetOnce() 请求参数:

参数key(String类型)	参数value类型	参数说明
properties	json-object	自定义用户属性(固定初始值)

请求示例:

```
var param = {properties:{register_date:"2022-01-01"}};
uzmoduledemo.userSetOnce(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

返回参数:无

4.3 数值类型的属性

说明:同一个 key 多次设置时, value 值(数值类型)进行求和计算;常用于记录用户付费次数、付费额度、积分等属性。

方法名: userAdd() 请求参数:

参数key (String类型)	参数value类型	参数说明
properties	json-object	自定义用户属性 (注意json里面的value值是数值类型)

请求示例:

```
var param = {properties:{score:80}};
uzmoduledemo.userAdd(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

返回参数:无

4.4 集合类型的属性

说明:同一个 key 多次设置时,value 值(字符串类型)都会被添加到集合里;适用于用户喜爱的电影、用户点评过的餐厅等属 性。

方法名: userAppend() 请求参数:

参数key (String类型)	参数value类型	参数说明
propertyName	String	属性name
appendList	String集合	要追加的属性值集合

请求示例:

```
var param = {propertyName:"movies",appendList:["电影1","电影2"]};
uzmoduledemo.userAppend(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

返回参数:无

4.5 属性取消

说明:取消已设置的某个用户属性。

方法名: userUnset() 请求参数:

参数key(String类型)	参数value类型	参数说明
propertyName	String	属性name

请求示例:

```
var param = {propertyName:"gender"};
uzmoduledemo.userUnset(param, function(ret, err){
        alert(JSON.stringify(ret));
});
```

返回参数:无

4.6 清空用户属性

说明:清空当前用户的用户属性。

方法名: userDelete() 请求参数:无

请求示例:

```
var param = {};
uzmoduledemo.userDelete(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

返回参数:无

5.1 用户登录

说明:当用户注册成功或登录成功时,需要调用 SDK 的 setUserUld() 方法。

方法名: setUserUld() 请求参数:

参数key(String类型)	参数value类型	参数说明
userUld	String	业务方-用户id

请求示例:

```
var param = {userUId:"1234567890"};
uzmoduledemo.setUserUId(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

返回参数:无

5.2 获取设备唯一ID

说明:如需获取设备唯一id,请在初始化SDK后调用getDeviceUld()方法进行获取。

方法名: getDeviceUld() 请求参数:无

请求示例:

```
var param = {};
uzmoduledemo.getDeviceUId(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

返回参数:

参数key(String类型)	参数value类型	参数说明
deviceUld	String	设备唯一id

返回示例:

```
{
    "deviceUId":"abcdefghijklmnopqrstuvwxyz",
}
```

5.3 自定义设备唯一ID

说明:默认情况下,SDK 会生成设备唯一ID 并可以保证该ID的唯一性,如需替换SDK默认分配的,配置如下:

方法名: setDeviceUld() 请求参数:

参数key(String类型)	参数value类型	参数说明
deviceUld	String	业务方-自定义设备唯一id

请求示例:

```
var param = {deviceUId:"dededededededede"};
uzmoduledemo.setDeviceUId(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

返回参数:无

注意: SDK 默认使用 AndroidId 作为设备 ID, 如果 AndroidId 获取不到则获取随机的 UUID。

6. 数据存储与发送

在每次调用 track()、setUserUid()、userSet() 等方法时,SDK 会将埋点事件保存在数据库中,并会检查如下条件,以判断是否向服务器上传数据:

```
    1.是否是 WIFI/26/36/46/56 网络条件
    2.是否满足发送条件之一:

            a.与上次发送的时间间隔是否大于 flushInterval
            b.本地缓存日志数目是否大于 flushPendSize
            c.事件类型为 setUserUid() 方法触发的 $SignUp 事件
```

6.1 上报条件

6.1.1 缓存条数

说明:设置本地缓存日志的最大条目数,默认本地埋点数据缓存为100条。

方法名: setFlushPendSize 方法: 请求参数:

参数key(String类型)	参数value类型	参数说明
flushPendSize	int	

请求示例:

```
var param = {flushPendSize:100};
uzmoduledemo.setFlushPendSize(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

返回参数:无

6.1.2 发送间隔

说明:设置埋点数据发送的间隔,默认为15s。

```
方法名: setFlushInterval 方法:
请求参数:
```

参数key (String类型)	参数value类型	参数说明
flushInterval	int	

请求示例:

```
var param = {flushInterval:15000};
uzmoduledemo.setFlushInterval(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

返回参数:无

6.1.3 设置数据的网络上传策略

说明:默认情况下,在WIFI/3G/4G/5G网络条件下,SDK都会尝试去同步数据。

方法名: setFlushNetworkPolicy 方法: 请求参数:

参数key (String类型)	参数value类型	参数说明
networkTypePolicy	int	

请求示例:

```
/**
* 设置 flush 时网络发送策略,默认 36、46、WI-FI 环境下都会尝试 flush
* TYPE_NONE = 0;//NULL
* TYPE_2G = 1;//2G
* TYPE_3G = 1 << 1;//3G 2
* TYPE_4G = 1 << 2;//4G 4
* TYPE_WIFI = 1 << 3;//WIFI 8
* TYPE_5G = 1 << 4;//5G 16
* TYPE_ALL = 0xFF;//ALL 255
 * 例:若需要开启 46 56 发送数据, 则需要设置 4 + 16 = 20
*/
//指定只在 3G/4G/WIFI 条件下发送数据。
var param = {networkTypePolicy:14};
uzmoduledemo.setFlushNetworkPolicy(param, function(ret, err){
   alert(JSON.stringify(ret));
});
```

返回参数:无

6.2 立即上报 (强制上报)

说明:如果追求数据采集的时效性,调用flush(),即可立即执行上报。

方法名: flush() 请求参数:无

请求示例:

```
var param = {};
uzmoduledemo.flush(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

返回参数:无

6.3 本地数据缓存上限值

moduleHC.init(param, function(ret, err){
 alert(JSON.stringify(ret));
});

6.4 清空本地缓存事件

说明:删除 App 本地存储的所有事件,如果不是特殊要求,请不要调用此方法。

```
var param = {};
uzmoduledemo.clear(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

返回参数:无

7. 其他功能

7.1 日志开关

说明:初始化 SDK 时,进行如下配置。

```
var param = {
...,
//日志开关
enableLog:true,
...
};
moduleHC.init(param, function(ret, err){
    alert(JSON.stringify(ret));
});
```

更新日志

插件版本:1.0.0

【发布日期】: 2023-03-31

【更新内容】:

1. 发布 APICloud 模块初版,支持代码埋点和自动埋点;

React Native

集成文档

1. 集成RN模块

对于 React Native 开发的应用,可以使用 npm 或 yarn 方式,集成RN模块。

npm install hina-react-native-lib

或者

yarn add hina-react-native-lib

2. 集成原生SDK

2.1 Android 端

Android SDK集成文档

2.1.1 手动注册Package类

如果需要把 React Native 组件集成到 Android 应用中,需要在加载 ReactRootView 的 Activity 中添加 RNHinaReactNativeLibPackage 并实现相关回调。

```
public class RNPageActivity extends AppCompatActivity implements DefaultHardwareBackBtnHandler {
   private ReactRootView mReactRootView;
   public static ReactInstanceManager mReactInstanceManager;
   private final static String TAG = "RNPageActivity";
   @Override
   protected void onCreate(Bundle savedInstanceState) {
       super.onCreate(savedInstanceState);
       mReactRootView = new ReactRootView(this);
       mReactInstanceManager = ReactInstanceManager.builder()
               .setApplication(getApplication())
               .setCurrentActivity(this)
               .setBundleAssetName("index.android.bundle")
               .setJSMainModulePath("index")
               .addPackage(new MainReactPackage())
               .addPackage(new RNHinaReactNativeLibPackage())//添加RNHinaReactNativeLibPackage类
               .setUseDeveloperSupport(BuildConfig.DEBUG)
               .setInitialLifecycleState(LifecycleState.RESUMED)
               .build();
       // 这个"MyApp"名字一定要和我们在index.js中注册的名字保持一致AppRegistry.registerComponent()
       mReactRootView.startReactApplication(mReactInstanceManager, "MyApp", null);
       setContentView(mReactRootView);
 }
  @Override
   public void invokeDefaultOnBackPressed() {
       super.onBackPressed();
```

```
}
@Override
   protected void onResume() {
     super.onResume();
      if (mReactInstanceManager != null) {
          mReactInstanceManager.onHostResume(this, this);
          Log.d(TAG, "onResume");
 }
}
@Override
   protected void onPause() {
     super.onPause();
     if (mReactInstanceManager != null) {
        mReactInstanceManager.onHostPause(this);
         Log.d(TAG, "onPause");
  }
}
@Override
   public void onBackPressed() {
     if (mReactInstanceManager != null) {
          mReactInstanceManager.onBackPressed();
          Log.d(TAG, "onBackPressed");
    } else {
         super.onBackPressed();
      }
}
@Override
   protected void onDestroy() {
super.onDestroy();
if (mReactInstanceManager != null) {
          mReactInstanceManager.onHostDestroy(this);
          Log.d(TAG, "onDestroy");
}
if (mReactRootView != null) {
          mReactRootView.unmountReactApplication();
}
}
public boolean onKeyUp(int keyCode, KeyEvent event) {
if (BuildConfig.DEBUG) {
if (keyCode == KeyEvent.KEYCODE_MENU) {//Ctrl + M 打开RN开发者菜单
             mReactInstanceManager.showDevOptionsDialog();
             return true;
}
}
return super.onKeyUp(keyCode, event);
}
}
```

注意:

需要在 onResume()、onPause()、onDestroy() 回调方法中实现 mReactInstanceManager.onHostResume(this, this)、mReactInstanceManager.onHostDestroy(this) 方法,否则可能出现点击事件全埋点无法触发情况。

2.2 iOS 端

iOS SDK集成文档

2.2.1 在项目目录下执行:

```
cd ios && pod install && cd ..
```

注意:

如果pod集成出现问题,也可以参考原生手动集成方式。

```
2.2.2 在iOS原生端集成SDK
```

在 AppDelegate 的 -application:didFinishLaunchingWithOptions: 中添加初始化代码:

```
// 初始化配置
HNBuildOptions *options = [[HNBuildOptions alloc] initWithServerURL:0"#<数据接收地址>" launchOptions:launch
Options];
 // 开启全埋点, 可根据需求进行组合
 options.autoTrackEventType = HNAutoTrackAppStart |
 HNAutoTrackAppEnd |
 HNAutoTrackAppClick
 HNAutoTrackAppScreen;
#ifdef DEBUG
// SDK 开启 Log
 options.enableLog = YES;
#endif
/**
 * 其他配置
 */
// 初始化 SDK
[HinaCloudSDK startWithConfigOptions:options];
```

使用说明

1. 导入模块

import hinaCloud from 'hina-react-native-lib'

2. 初始化SDK

说明:使用init方法,按如下方式进行初始化。

参数key (String类型)	参数value类型	参数说明
serverUrl	String	【必填】配置数据上报地址(在线获取)
flushInterval	int	【可选】上报时间间隔(毫秒)
flushPendSize	int	【可选】自定义上报缓存条数
enableLog	bool	【可选】开启日志
maxCacheSizeForAndroid	long	【可选】自定义缓存上限(Android用)
maxCacheSizeForIOS	int	【可选】自定义缓存上限(iOS用)
autoTrackTypePolicy	int	【可选】开启全埋点
networkTypePolicy	int	【可选】自定义上报网络条件
enableJSBridge	bool	【可选】App打通H5

```
var param = {
    serverUrl:"数据上报地址",
    enableLog:true,
    autoTrackTypePolicy:AutoTrackType.START|AutoTrackType.END
};
```

hinaCloud.init(param)

3. 代码埋点 (手动采集)

说明:使用track()方法进行直接埋点,埋点事件支持添加自定义属性。

```
var eventName = "click_one";
var properties = {who:"william"};
```

hinaCloud.track(eventName, properties);

API详解

1. 代码埋点 (手动采集)

用于记录埋点事件。

1.1 直接埋点

说明:使用track()方法进行直接埋点,埋点事件支持添加自定义属性。

```
var eventName = "click_one";
var properties = {who:"william"};
hinaCloud.track(eventName, properties);
```

1.2 时长埋点

```
说明:需要成对调用计时器的开始和结束方法,以此来实现对统计时长的事件采集;
*在事件开始时,调用 trackTimerStart("ViewProduct"),该方法并不会真正发送事件;
*在事件结束时,调用 trackTimerEnd("ViewProduct", properties),SDK 会触发 "Event" 事件,并自动将事件持续时间记
录在事件属性 "$event_duration" 中。
```

```
// 调用 trackTimerStart("ViewProduct") 标记(浏览商品)事件启动时间
hinaCloud.trackTimerStart("ViewProduct");
```

// 在属性中记录商品 ID
// 内部会调用 track, 记录 ViewProduct 事件, 并在属性 event_duration 中记录用户浏览商品的时间
hinaCloud.trackTimerEnd("ViewProduct", {product:"商品1"});

2. 全埋点 (自动采集)

SDK 可以自动采集一些用户行为,如 App 启动、退出、浏览页面、控件点击,共计四种。

说明:在初始化时,通过autoTrackTypePolicy参数进行配置开启,不配置则视为关闭全埋点。

```
var param = {
    ...
    server_url:'<#数据接收地址>',
    autoTrackTypePolicy:AutoTrackType.START|AutoTrackType.END
};
hinaCloud.init(param)
```

.

注意:浏览页面、控件点击仅对原生工程支持,RN工程仅支持采集APP启动和退出;

3. 事件属性

在进行埋点事件追踪时,您可以根据需求对埋点事件进行属性的定义。目前 SDK 中提供了公共属性用于给每个埋点事件添加属 性。

3.1 获取预置属性

```
说明:如需了解和使用预置属性,可以通过此方法获取预置属性。
```

```
// 返回promise类型
hinaCloud.getPresetProperties().then(result => {})
```

3.2 设置公共属性

说明:公共属性是指对于所有事件都需要添加的属性,设置之后 SDK 会在每次触发埋点时,自动获取并添加到触发的事件中。

```
var properties = {app_name: 'yourAppName'};
hinaCloud.registerCommonProperties(properties);
```

4. 用户属性

4.1 设置用户属性

说明:同一个 key 多次设置时, value 值会进行覆盖替换。

```
var properties = {age:"18"};
hinaCloud.userSet(properties);
```

4.2 固定初始值的属性

说明:同一个 key 多次设置时, value 值只会记录初次设定的值;适用于为用户设置首次激活时间、首次注册时间等属性。

```
var properties = {register_date:"2022-01-01"};
hinaCloud.userSetOnce(properties);
```

4.3 数值类型的属性

说明:同一个 key 多次设置时, value 值(数值类型)进行求和计算;常用于记录用户付费次数、付费额度、积分等属性。

```
var properties = {score:80};
hinaCloud.userAdd(properties);
```

4.4 集合类型的属性

说明:同一个 key 多次设置时,value 值(字符串类型)都会被添加到集合里;适用于用户喜爱的电影、用户点评过的餐厅等属 性。

```
// 设定用户观影列表属性,设定后属性 "movies" 为: ["movie1", "movie2"]
hinaCloud.userAppend("movies", ["movie1", "movie2"]);
```

4.5 属性取消
说明:取消已设置的某个用户属性。

hinaCloud.userUnset("gender");

4.6 清空用户属性

说明:清空当前用户的用户属性。

hinaCloud.userDelete();

5. 用户关联

5.1 用户登录

说明:当用户注册成功或登录成功时,需要调用 SDK 的 setUserUld() 方法。

hinaCloud.setUserUId("业务userId");

注意:为了准确记录登录用户的行为信息,建议在以下时机各调用一次用户登录方法:

·用户在注册成功时

·用户登录成功时

·已登录用户每次启动 App 时

5.2 获取设备唯一ID

说明:如需获取设备唯一id,请在初始化SDK后调用getDeviceUld()方法进行获取。

```
// 返回promise类型
hinaCloud.getDeviceUId().then(result => {})
```

5.3 自定义设备唯一ID

说明:默认情况下,SDK 会生成设备唯一ID 并可以保证该ID的唯一性,如需替换SDK默认分配的,配置如下:

hinaCloud.setDeviceUId("自定义设备Id");

注意: SDK 默认使用 AndroidId 作为设备 ID, 如果 AndroidId 获取不到则获取随机的 UUID。

6. 数据存储与发送

在每次调用 track()、setUserUld()、userSet() 等方法时,SDK 会将埋点事件保存在数据库中,并会检查如下条件,以判断是否 向服务器上传数据:

```
    1.是否是 WIFI/2G/3G/4G/5G 网络条件
    2.是否满足发送条件之一:

            a.与上次发送的时间间隔是否大于 flushInterval
```

```
b.本地缓存日志数目是否大于 flushPendSize
c.事件类型为 setUserUid() 方法触发的 $SignUp 事件
```

6.1 上报条件

6.1.1 缓存条数

说明:设置本地缓存日志的最大条目数,默认本地埋点数据缓存为100条。

```
var param = {
    ...
    flushPendSize:50,
    ...
};
hinaCloud.init(param)
```

6.1.2 发送间隔

说明:设置埋点数据发送的间隔,默认为15s。

```
var param = {
    ...
    flushInterval:30000,//设置两次数据发送的最小时间间隔,最小值 5 秒;时间间隔,单位毫秒。
    ...
};
```

hinaCloud.init(param)

6.1.3 设置数据的网络上传策略

说明:默认情况下,在 WIFI/3G/4G/5G 网络条件下,SDK 都会尝试去同步数据。可以自由组合来指定发送数据的网络策略。

```
/**
 * 设置 flush 时网络发送策略, 默认 36、46、WI-FI 环境下都会尝试 flush
 * TYPE_NONE = 0;//NULL
 * TYPE_26 = 1;//26
 * TYPE_36 = 1 << 1;//36 2
 * TYPE_46 = 1 << 2;//46 4
 * TYPE_WIFI = 1 << 3;//WIFI 8
 * TYPE_56 = 1 << 4;//56 16
 * TYPE_ALL = 0xFF;//ALL 255
 * 例:若需要开启 46 56 发送数据,则需要设置 4 + 16 = 20
 */
//指定只在 36/46/WIFI 条件下发送数据。
hinaCloud.setFlushNetworkPolicy(14);</pre>
```

6.2 立即上报 (强制上报)

说明:如果追求数据采集的时效性,调用flush(),即可立即执行上报。

hinaCloud.flush();

注意:在 App 进入后台状态或监听到网络切换有网络时,SDK 会调用 flush() 方法,将缓存的数据发送。

6.3 本地数据缓存上限值

说明:Android和iOS设置维度不一样,如有需要,请分开设置;

```
var param = {
    ...
    //android配置空间大小, SDK 本地数据库默认缓存数据的上限值为 32MB
    //设置本地缓存上限值, 单位 byte, 默认为 32MB: 32 * 1024 * 1024, 最小 16MB: 16 * 1024 * 1024,
    //若小于 16MB, 则按 16MB 处理。
    maxCacheSizeForAndroid: 32 * 1024 * 1024,
    //iOS配置条数, 默认20000条
    maxCacheSizeForIOS: 10000,
    ...
};
```

hinaCloud.init(param)

注意:当存储数量达到上限值,会依次丢弃老数据,保留最新的数据。

6.4 清空本地缓存事件

说明:删除 App 本地存储的所有事件。

hinaCloud.clear();

注意:如果不是特殊要求,请不要调用此方法。

7. 其他功能

7.1 开启App与H5打通功能

说明:初始化 SDK 时,进行如下配置。

```
var param = {
    ...
    enableJSBridge:true,
    ...
};
hinaCloud.init(param)
```

7.2 日志开关

说明:初始化 SDK 时,进行如下配置。

```
var param = {
    ...
enableLog:true,
    ...
};
```

hinaCloud.init(param)

更新日志

服务端SDK

Java SDK

SDK 功能清单

一级功能	二级功能	具体功能 / 描述	
	专动医女田古	自动使用设备 ID 标识匿名用户	
		支持替换为自定义的匿名 ID	
柳枫用广	标识登录用户	使用登录 ID 标识登录用户	
	设置用户属性		
		静态公共属性	
	公共属性	动态公共属性	
		清除公共属性	
	占守之王方	统计事件时长	
	日正义垟总	自定义代码埋点	
采集数据	存储数据	设置上限	设置本地缓存上限值
		删除本地缓存	删除缓存在本地的所有事1
	上报数据	自动上报数据	SDK 在满足一定条件后自动发注
		手动上报数据	SDK 支持手动上报事件数
	调试功能	调试日志	显示 / 关闭调试日志/td>
高级功能	数据加密	数据加密之后再进行存储和发送	
	合规	支持合规要求	
	SDK 采集控制	控制 SDK 开启或关闭	

集成文档

帮助中心 > 技术文档 > 服务端SDK > Java SDK > 集成文档

注意:

- 1. 目前仅支持离线包集成, maven库支持尚未开通
- 2. Java SDK 要求JDK版本为1.8
- 3. 目前Java SDK (jar 格式) 大小约为 8 KB

1. 配置依赖

- 1、下载SDK离线包(aar包),详见【资源下载】;
- 2、下载zip包后, 解压, 使用maven命令安装在本地仓库

```
mvn install:install-file -Dfile=D:\hina-java-sdk-1.0.jar -DgroupId=com.hina.sdk -DartifactId=hina-java-sdk -D
version=1.0 -Dpackaging=jar
```

3、项目中使用maven依赖

```
<dependency>
    <groupId>com.hina.sdk</groupId>
        <artifactId>hina-java-sdk</artifactId>
        <version>1.0</version>
</dependency>
```

2. 获取数据接收地址

如果使用cloud服务,数据接收地址如:

参考文档:数据上报地址获取

3. 初始化

由于 SDK 初始化操作,会创建一个与之对应的内存缓存队列,所以请谨慎创建 SDK 实例对象,推荐在应用程序启动时全局初 始化一次,然后进行全局调用。避免在方法内部进行初始化操作,程序运行过程中创建多个实例对象,导致 00M 现象

3.1 在普通 Java 程序中初始化 SDK

```
// SERVER_URL需要在网站上获取
String SERVER_URL="";
HinaSdk hinaSdk=new HinaSdk();
hinaSdk.initWithNet(SERVER_URL,6);
// 设置全局属性
hinaSdk.registerSuperProperties("macos","ventura","192.168.1.102");
```

```
// 用户的 Distinct ID
String distinctId = <mark>"ABCDEF123456789"</mark>;
```

集成文档

```
// 记录用户登录事件
Map<String, Object> map = new HashMap<>();
map.put("H_tinyint_1_AUT", 6);
map.put("H_smallint_1_AUT", 60);
map.put("H_int_1_AUT", 600);
map.put("H_bigint_1_AUT", 6000);
map.put("H_largeint_1_AUT", 60000);
map.put("H_char_1_AUT", "char6588");
hinaSdk.logInEvent("java_sdk_tz_205","H_pre_event_AUT", map);
```

3.2 在Spring项目中初始化SDK

如果您的项目使用 Spring 框架,推荐将初始化操作装配成 Bean,交给 Spring 容器来管理。在使用的类中注入即可使用。

```
@Configuration
public class HinaConfig {
    @Bean(destroyMethod = "shutdown")
    public HinaSdk init() throws IOException {
        HinaSdk hinaSdk=new HinaSdk();
        hinaSdk.initWithNet(SERVER_URL,6,true,50);
        return hinaSdk;
    }
}
```

使用说明

帮助中心 > 技术文档 > 服务端SDK > Java SDK > 使用说明

1. 配置项目数据接收地址

在web中引入SDK时,需要手动将 server_url 参数设置为相应项目的数据接收地址。 说明:采集上报地址需在海纳云平台,进行在线申请,请参考【采集上报地址获取】;

2. 常用功能

SDK采集事件分登录前事件,登录后事件

2.1 绑定匿名ID和登录后ID

用户登录后,可以将登录ID和匿名ID绑定

hina.bindId(<mark>"匿名ID","登录ID</mark>")

2.2 登录前发送事件

可以调用 notLogInEvent 来上传事件

hinaSdk.notLogInEvent("匿名ID","事件名称",事件数据);

2.3 登录后发送事件

可以调用 logInEvent 来上传事件

hinaSdk.logInEvent("登录ID","事件名称",事件数据);

2.4 设置公共事件属性

对于所有事件都需要添加的属性,可在初始化 SDK 后,调用 registerCommonProperties 将属性注册为公共属性

hinaSdk.registerCommonProperties("macos","ventura","192.168.1.102");

如果调用 registerCommonProperties 上传了先前已设置过的公共事件属性,则会覆盖之前的属性。

2.5 设置用户属性

对于一般的用户属性,您可以调用 setUser 来进行设置,使用该接口上传的属性将会覆盖原有的属性值,如果之前不存在 该用户属性,则会新建该用户属性

hina.setUser("用户ID",用户属性)

使用说明

API详解

帮助中心 > 技术文档 > 服务端SDK > Java SDK > API详解

1. SDK初始化参数

参数	默认值	说明
serverPath	String	数据接收地址
builSize	int	条数,当缓存的数据量达到50条时,批量发送数据
throwException	boolean	false 表示数据同步失败不抛出异常
maxCacheSize	int	内存中数据最大缓存条数,如此值大于0, 代表缓存的数据会有条数限制,最小 3000 条,最大 6000 条。否则无条数限制

2. 设置用户ID

SDK 实例登录前使用匿名ID发送事件,登录后使用登录后ID发送事件,匿名ID和登录后ID

用户登录后,可以将登录ID和匿名ID绑定

hina.bindId("匿名ID","登录ID")

3. 设置用户属性

3.1 设置用户属性

直接设置用户的属性,如果存在则覆盖

hina.setUser("用户ID",用户属性)

3.2 固定初始值的属性

如果不存在则设置,存在就不设置

hina.setUserOnce(<mark>"用户ID</mark>",用户属性)

3.3 数值类型的属性

对当前用户的属性做递增或者递减

hina.incrementUser("用户ID",key,value)

3.4 集合类型的属性

给数组属性添加值, 通过userSet只能改变属性的值, 如果这个属性是数组类型的,你不想完全改变这个值,只想做添加操作可以 使用此方法

hina.appendUser("用户ID",key,value)

3.5 属性取消

删除当前用户的一些属性

hina.unsetUser("用户ID",用户属性)

3.6 清空用户属性

删除当前用户及他的所有属性

hina.deleteUser("用户ID")

4 设置公共事件属性

对于所有事件都需要添加的属性,可在初始化 SDK 后,调用 registerCommonProperties 将属性注册为公共属性

hinaSdk.registerCommonProperties("macos","ventura","192.168.1.102");

5. 发送事件

在 SDK 初始化完成之后,您就可以进行数据埋点,收集用户的的行为信息

5.1 登录前事件

可以调用 notLogInEvent 来上传事件

hinaSdk.notLogInEvent("匿名ID","事件名称",事件数据);

5.2 登录后事件

可以调用 logInEvent 来上传事件

hinaSdk.logInEvent("登录ID","事件名称",事件数据);

注意:

- 1. 批量发送功能和回调函数功能不可以同时使用,比如track加了callback,使用批量发送后callback不会执行
- 2. 批量发送默认使用 ajax 的方式发送数据,如果不支持跨域ajax发送数据,会使用img且实时发送数据
- 3. 如果localStorage 里已经存了超过 200 条数据,会导致批量发送功能失效,localStorage 中只保存这 200 条数据,新产生 的数据使用 img 且实时发送数据的方式。当进入同域名的新页面时,会自动检查缓存中是否有数据,如果有会继续发送缓 存的数据
- 4. useAppBridge和batch_send只能选择一个,开启了打通App就不能再用批量发送

更新日志

帮助中心 > 技术文档 > 服务端SDK > Java SDK > 更新日志

版本:1.0.0

【发布日期】: 2023-02-23

【更新内容】:

1. 发布SDK初版,支持自定义埋点。

Python SDK

SDK 功能清单

一级功能	二级功能	具体功能 / 描述	
标识用户	专动医女田立	自动使用设备 ID 标识匿名用户	
		支持替换为自定义的匿名 ID	
	标识登录用户	使用登录 ID 标识登录用户	
	设置用户属性		
		静态公共属性	
	公共属性	动态公共属性	
		清除公共属性	
	百分之重方	统计事件时长	
	日正义连点	自定义代码埋点	
采集数据	存储数据	设置上限	设置本地缓存上限值
		删除本地缓存	删除缓存在本地的所有事件
	上报数据	自动上报数据	SDK 在满足一定条件后自动发注
		手动上报数据	SDK 支持手动上报事件数
	调试功能	调试日志	显示 / 关闭调试日志/td>
	数据加密	数据加密之后再进行存储和发送	
高级功能	合规	支持合规要求	
	SDK 采集控制	控制 SDK 开启或关闭	
•			

集成文档

帮助中心 > 技术文档 > 服务端SDK > Java SDK > 集成文档

注意:

- 1. 目前仅支持离线包集成
- 2. python版本要求为python3以上

1. 集成hina SDK

1、在 Python 代码中集成海纳分析 SDK,使用海纳分析采集并分析用户数据。 我们推荐使用源码依赖的方式,集成SDK,源码下载地址: HinaCloudSDK-Python-1.0.0.zip,然后在代码中引用

import hina_sdk

2. 获取配置信息

首先从海纳云分析的主页中,获取数据接收的 URL 和 Token (Cloud 版)

参考文档:数据上报地址获取

3. 在程序中初始化 SDK

由于 SDK 初始化操作,会创建一个与之对应的内存缓存队列,所以请谨慎创建 SDK 实例对象,推荐在应用程序启动时全局初 始化一次,然后进行全局调用。避免在方法内部进行初始化操作,程序运行过程中创建多个实例对象,导致 OOM 现象

import hina_sdk as hina

init_server 是将数据实时发送至网络端, server需要在海纳云分析中获取 hina.init_server(server,timeout) # 记录用户登录事件 distinct_id = 'ABCDEF123456789' hina.track(distinct_id, 'UserLogin', True)

使用说明

帮助中心 > 技术文档 > 服务端SDK > Java SDK > 使用说明

1. 配置项目数据接收地址

在web中引入SDK时,需要手动将 server_url 参数设置为相应项目的数据接收地址。 说明:采集上报地址需在海纳云平台,进行在线申请,请参考【采集上报地址获取】;

2. 常用功能

SDK采集事件分登录前事件,登录后事件

2.1 绑定匿名ID和登录后ID

用户登录后,可以将登录ID和匿名ID绑定

hina.track_signup(<mark>"匿名ID","登录ID</mark>")

2.2 登录前发送事件

可以调用 not_login_track 来上传事件

hina.not_login_track(<mark>"匿名ID","事件名称"</mark>, 事件数据);

2.3 登录后发送事件

可以调用 login_track 来上传事件

hina.login_track("登录ID","事件名称",事件数据);

2.4 设置公共事件属性

对于所有事件都需要添加的属性,可在初始化 SDK 后,调用 register_common_properties 将属性注册为公共属性

hina.register_common_properties({'ServerVersion': '1.2', 'Location': 'Beijing'})

如果调用 register_common_properties 上传了先前已设置过的公共事件属性,则会覆盖之前的属性。

2.5 设置用户属性

对于一般的用户属性,您可以调用 set_user 来进行设置,使用该接口上传的属性将会覆盖原有的属性值,如果之前不存在 该用户属性,则会新建该用户属性

hina.set_user('ABCDEF123456789', {'Sex': 'Male','UserLevel': 'Elite VIP',}, True)

使用说明

API详解

帮助中心 > 技术文档 > 服务端SDK > Java SDK > API详解

1. SDK初始化参数

参数	默认值	说明
serverPath	String	数据接收地址
timeout	int	超时时间

2. 设置用户ID

SDK 实例登录前使用匿名ID发送事件,登录后使用登录后ID发送事件,匿名ID和登录后ID

用户登录后,可以将登录ID和匿名ID绑定

hina.track_signup("匿名ID","登录ID")

3. 设置用户属性

3.1 设置用户属性

直接设置用户的属性,如果存在则覆盖

hina.set_user('ABCDEF123456789', {'Sex': 'Male','UserLevel': 'Elite VIP',}, True)

3.2 固定初始值的属性

如果不存在则设置,存在就不设置

hina.set_user_once('ABCDEF123456789', {'Sex': 'Male','UserLevel': 'Elite VIP',})

3.3 数值类型的属性

对当前用户的属性做递增或者递减

hina.incr_user(distinct_id, {'GamePlayed': 1}, True)

3.4 集合类型的属性

给数组属性添加值,通过userSet只能改变属性的值,如果这个属性是数组类型的,你不想完全改变这个值,只想做添加操作可以 使用此方法

hina.append_user(distinct_id, {'Movies': ['Dead Poets Society']}, True)

3.5 属性取消

删除当前用户的一些属性

hina.unset_user(distinct_id,key,True)

3.6 清空用户属性

删除当前用户及他的所有属性

hina.delete_user(distinct_id,True)

4 设置公共事件属性

对于所有事件都需要添加的属性,可在初始化 SDK 后,调用 register_common_properties 将属性注册为公共属性

hina.register_common_properties({'ServerVersion': '1.2', 'Location': 'Beijing'})

5. 发送事件

在 SDK 初始化完成之后,您就可以进行数据埋点,收集用户的的行为信息

5.1 发送登录前事件

可以调用 not_login_track 来上传事件

hina.not_login_track(distinct_id, 'ViewProduct', properties, True)

5.2 发送登录后事件

可以调用 login_track 来上传事件

hina.login_track(distinct_id, 'ViewProduct', properties, True)

更新日志

帮助中心 > 技术文档 > 服务端SDK > Java SDK > 更新日志

版本:1.0.0

【发布日期】: 2023-02-23

【更新内容】:

1. 发布SDK初版,支持自定义埋点。

数据校验

数据导入

数据导入

Datax

一、概述

DataX 是阿里巴巴集团内被广泛使用的离线数据同步工具/平台,实现包括 MySQL、SQL Server、Oracle、PostgreSQL、HDFS、 Hive、HBase、OTS、ODPS 等各种异构数据源之间高效的数据同步功能。 如需了解更多DataX,可以访问DataX主页

二、环境要求

1、python环境

2、jdk1.8及以上

三、安装

3.1 下载

- DataX主页
- wget命令下载,如下所示

wget http://datax-opensource.oss-cn-hangzhou.aliyuncs.com/datax.tar.gz

下载后得到datax.tar.gz包

3.2 解压

tar -zxvf datax.tar.gz

3.3 插件安装

hinawriter是一个写入插件,主要是根据配置对读入数据进行格式整理,之后再打包数据,压缩并经过 base64 之后发送至海纳 的数据接收地址。 下载插件:hinawriter插件 把 hinawriter.tar.gz 解压到 datax/plugin/writer 目录下 解压插件包

tar -zxvf hinawriter.tar.gz
rm -rf hinawriter.tar.gz

解压完成之后记得删除

3.4 配置

配置文件是格式为json,配置样例如下图所示

```
{
 "job": {
   "setting": {
     "speed": {
       "channel": 1
     }
   },
   "content": [
    {
       "reader": {
         "name": "mysqlreader",
          "parameter": {
           "username": "root",
            "password": "pssword",
            "column": [
             "user_id",
             "event",
             "project",
             "item_id",
              "item_name",
              "item_cate"
           ],
            "connection": [
             {
                "jdbcUrl": [
                  "jdbc:mysql://127.0.0.1:3306/table_name"
               ],
                "table": [
                 "event1",
                 "event2"
               ]
             }
           ]
        }
       },
        "writer": {
         "name": "hinawriter",
          "parameter": {
            "thread": 3,
            "type": "track",
            "pushUrl": "https://loanetc.mandao.com/hina-cloud-engine/hn?token=yt888",
            "appid": "test",
            "column": [
             {
                "index": "0",
                "colTargetName": "user_id"
             },
              {
                "index": "1",
                "colTargetName": "event"
             },
              {
                "index": "2",
                "colTargetName": "time",
                "type": "date",
                "dateFormat": "yyyy-MM-dd HH:mm:ss.SSS"
             },
              {
                "index": "3",
               "colTargetName": "distinct_id"
             },
```



reader配置参数说明请查看DataX主页 hinawriter配置参数说明

配置	必填	默认	说明
thread	否	3	线程数,为每个 channel 内部并发使用,与 DataX 的 channel 数量无关。
type	是	无	写入的数据格式 track、profile_set等
pushUrl	是	无	海纳接入点地址
appid	是	无	对应项目的 appid
compress	否	无	压缩方式,gzip:支持gzip压缩
column	是	无	详见column参数说明

column参数说明

配置	必填	默认	说明
index	index/value 必须选择其一	无	从reader到datax传输的数据列号, 从0开始
value	index/value 必须选择其一	无	数据常量,不从reader中获取
colTargetName	是	无	输出字段名
type	否	String	数据类型,见下表
dateFormat	否	无	type为date 类型时,可设置日期格式

数据类型

DataX数据类型	hinawriter数据类型
Int	Number
Long	Number
Double	Number
String	String
bytes	String
Boolean	Boolean
Date	Date

3.5 执行

在datax安装目录下/datax/bin目录下 执行命令如下:python datax.py 配置.json

四、常见问题

1、datax运行自检命令报错 datax/plugin/reader/._xxxx/plugin.json]不存在

删除.临时文件即可 在datax/plugin目录下运行

find ./* -type f -name ".*er" | xargs rm -rf

2、datax运行自检命令报错

datax/plugin/writer/hinawriter.tar.gz/plugin.json]不存在

删除hinawriter的插件包 在datax/plugin/writer目录下运行

rm -rf hinawriter.tar.gz

附件

• hinawriter.tar.gz

dataimporter

一、概述

dataimporter用于将一些符合数据格式的外部数据导入到海纳云分析系统中,目前支持的格式有CSV表格,MySQL数据库, ORACLE数据库,nginx日志,json格式数据。

二、环境要求

- 1、python环境3.4及以上
- 2、如需导入数据库数据,需要预先install相应的依赖包

三、下载安装

• 下载 dataimporter工具包: dataimporter下载地址,压缩即可使用

四、参数配置

4.1 公共参数

参数名	别名	是否必填	描述
—url	-1	和 output_file 选一个必填	数据接收地址
-output_file	-0	和 url 选一个必填	输出的文件名,输出每行是一个符合格式的 JSON。
-project	-j	否	指定的 project 名,默认是 default
–skip_cnt	-C	否	第一次运行请忽略,如果运行失败, 可以使用此配置指定跳过开头多少行
–debug	-D	否	如果指定了就是使用 debug 模式, 不会导入数据,只在
_ quit_on_error	-Q	否	如果选中,则出现一条错误日志就会退出
- log_level	-lv	否	日志输出最小等级,默认为 DEBUG

4.2 event公共参数

参数名	别名	是否必填	描述
-distinct_id_from	-df	是	指定列作为 distinct_id
–is_login		否	distinct_id 是否是登录 ID, 默认为否
-event_from	-ef	和 event_default 选一个必填	指定列作为事件名
-event_default	-ed	和 event_from 选一个必填	指定固定字符串作为事件名
-timestamp_from	-tf	和 timestamp_default 选一个必填	指定列作为 time
_ timestamp_default	-td	和 timestamp_from 选一个必填	指定固定时间字符串作为 time
_ timestamp_format	-tf	否	和 timestamp_from 一起使用, 通过此配置指定时间格式。 默认是%Y-%m-%d %H:%M:%S

4.3 profile公共参数

参数名	别名	是否必填	描述
-distinct_id_from	-df	是	指定列作为 distinct_id
-is_login		否	distinct_id 是否是登录 ID,默认为否

4.4 item公共参数

参数名	别名	是否必填	描述
-item_type		是	指定列作为 item_type
–item_id		是	指定列作为 item_id

4.5 signup公共参数

参数名	别名	是否必填	描述
-login_id_from		是	指定列作为登录 ID
-anonymous_id_from		是	指定列作为匿名 ID

4.6 csv参数

参数名	别名	是否必填	描述
-filename	-f	是	CSV 文件路径。
-property_list	-pl	否	用逗号分割选取的 property,举例 -p name,time 将会将 name 和 time 两列作为 property 导入。如果不填写则表示全部作为 property 导入。
-skip_identify	-i	否	对应的列将不会做自动类型判断,举例配置– skip_identify name,id 后将会对 name 和 id 不做类型判断,完全作为 string 导入。 如果不填写则表示全部的选中列都会自动做类 型判断。
-ignore_value		否	指定某些值为空,比如指定 –ignore_value null 则所有的 null 都被认为是空值。
-csv_delimiter		否	CSV 文件的列分隔符,默认为 ',', 只接受单字符参数,也可以传 \ + ascii的数字,比如 \9 表示是 \t 。
–csv_quotechar		否	CSV 文件的引用字符, 用于指定CSV字符串的开始和结尾,默认为 "",只接受单字符参数,也可以传 \ + ascii的数字,比如 \9 表示是 \t 。
– csv_prefetch_lines		否	CSV 文件预读行数, 预读用于判断列的类型,默认为 -1, 即预读整个文件。注意如果数据分布不均 (比如前几行某个字段没有但是后面有) 不要加这个参数。
-file_encoding		否	设置 CSV 文件编码格式,默认为 utf-8。

参数名	别名	是否必填	描述
-list_type		否	指定属性为 list 用逗号分割选取的属性, 举例–list_type list_a, list_b 将会将 list_a 和 list_b 两列作为 list 格式导入。 list_type:list_a, list_b

4.7 mysql参数

参数名	别名	是否必填	描述
–user	-u	是	连接 MySQL 数据库所需的用户名。
-password	-p	是	连接 MySQL 数据库所需的密码。
-host	-i	是	连接 MySQL 数据库所需的地址。
–port	-P	是	连接 MySQL 数据库所需的端口。
–db	-d	是	MySQL 对应的数据库名, 一次只能指定一个。
—sql	-q	和 filename 选一个必填	查询语句,建议加 order by 等方式保证多次查询结果顺序一致。
–filename	-f	和 sql 选一个必填	查询语句所在的文件路径,建议加 order by 等方式保证多次查询结果顺序一致。
_ bool_property_list	-bp	否	逗号分割的 bool 类型属性列表, 会将对应的属性值为 1 的转化为 true,0 转化为 false。
-case_sensitive	-CS	否	导入的属性名是否是大小写敏感, 注意如果大小写不敏感会全部转化为大写, 默认为 true。

4.8 oracle参数

参数名	别名	是否必填	描述
-user	-u	是	连接 Oracle 数据库所需的用户名。

参数名	别名	是否必填	描述
-password	-p	是	连接 Oracle 数据库所需的密码。
–dsn	-dsn	是	连接 Oracle 数据库所需的 dsn。
—sql	-q	和 filename 选一个必填	查询语句,建议加 order by 等方式保证多次查询结果顺序一致。
-filename	-f	和 sql 选一个必填	查询语句所在的文件路径,建议加 order by 等方式保证多次查询结果顺序一致。
_ bool_property_list	-bp	否	逗号分割的 bool 类型属性列表, 会将对应的属性值为 1 的转化为 true,0 转化为 false。
-case_sensitive	-CS	否	导入的属性名是否是大小写敏感, 注意如果大小写不敏感会全部转化为大写, 默认为 false。

4.9 nginx日志参数

参数名	别名	是否必填	描述
-filename	-f	是	Nginx 日志文件路径。
-log_format	-F	是	Nginx 日志配置,类似 "'\$remote_addr"
-property_list	-pl	是	用逗号分割选取的 property 举例
-skip_identify	-i	否	对应的列将不会做自动类型判断,举例
-url_fields	-uf	否	对应的列将作为 url 解析,用逗号分割。
-filter_path	-fp	否	过滤对应的 path,可多选。
-ignore_value		否	指定某些值为空,比如指定 –ignore_value null 则所有的null都被认为是空值。
<pre>_ property_list_cnames</pre>		否	用逗号分割 property 的对应名称, 需要和– property_list 一一对应。

4.10 json参数

参数名	别名	是否必填	描述
–path	-p	是	json文件目录路径

五、命令类型

命令名称	描述
csv_profile	将 CSV 格式文件转化成用户属性导入
csv_event	将 CSV 格式文件转化成事件导入
csv_item	将 CSV 格式文件转化成物品数据导入
csv_signup	将 CSV 格式文件转化成用户关联关系导入
mysql_profile	将 MySQL 数据库中数据转化成用户属性导入
mysql_event	将 MySQL 数据库中数据转化成事件导入
mysql_item	将 MySQL 数据库中数据转化成物品数据导入
mysql_signup	将 MySQL 数据库中数据转化成用户关联关系导入
oracle_profile	将 Oracle 数据库中数据转化成用户属性导入
oracle_event	将 Oracle 数据库中数据转化成事件导入
oracle_item	将 Oracle 数据库中数据转化成物品数据导入
oracle_signup	将 Oracle 数据库中数据转化成用户关联关系导入
nginx_profile	将 Nginx 日志转化成用户属性导入
nginx_event	将 Nginx 日志转化成事件导入
nginx_item	将 Nginx 日志转化成物品数据导入
nginx_signup	将 Nginx 日志转化成用户关联关系导入
json	将 JSON 数据导入

六、导入示例

假设有以下 csv 文件描述了上面的示例用户事件 (参考代码包下 examples/events.csv)

user_id,action,time,item_id,item_name,item_type 111,action1,2022-07-12 13:11:22,12345,蛤蟆夯么啊哈啊哈就好,type1 222,action2,2022-07-13 13:22:33,12346,呵呵呵呵呵哈哈哈哈哈,type2 333,action3,2022-07-14 17:33:54,12347,哈哈哈好久没牛鬼蛇神,type3 444,action4,2022-07-15 05:42:53,12348,呵呵呵呵呵二二二嗯多,type4

将这些数据导入到本地私有部署的环境,以 user_id 作为用户的 id,以 time 作为事件发生的时间,以 action 作为事件名称,只 导入 item_id 和 item_name 作为事件属性。 配置属性如下所示

```
--url 'http://localhost:8088/hina-cloud-engine/hn' \
--distinct_id_from 'user_id' \
--timestamp_from 'time' \
--event_from 'action' \
--filename './examples/events.csv' \
--property_list 'item_id,item_name' \
```

七、导入执行

• 填写好对应的配置文件信息csv_event.conf,可参考工具包中默认配置,可以修改后直接使用

```
url: http://localhost:8088/hina-cloud-engine/hn
distinct_id_from: user_id
is_login
timestamp_from: time
event_from: action
filename: ./examples/events.csv
property_list: item_id,item_name
```

• 执行命令如下:其中csv_event是命令类型,csv_event.conf是配置信息

python3 data_import.py csv_event @./conf/csv_event.conf

• 如果需要配置导入mysql或者oracle数据库数据,执行前需要先安装依赖,如下所示:

mysql依赖安装

python3 -m pip install PyMySQL --upgrade

oracle依赖安装

python3 -m pip install cx_Oracle --*upgrade*

附件

- data-import.zip
- dataimporter.zip

Logstash + Filebeat

一、概述

Logstash 是由 Elastic 公司推出的一款开源的服务器端数据处理管道,能够同时从多个来源采集数据,转换数据,然后将数据发送指定的存储库中。Logstash 官方介绍。

Filebeat 是 Elastic 公司为解决 Logstash "太重" 的问题推出的一款轻量级日志采集器,在处理数量众多的服务器、虚拟机和容器 生成的日志时可使用 Logstash + Filebeat 的日志采集方式。Filebeat 官方介绍。

基于 Logstash + Filbeat 的数据采集流程为:后端 SDK 生成数据文件 => Filebeat 读取文件 => Logstash Beat input => Logstash hina_analytics output => 海纳云。



· 2.1、下载安装包

wget https://artifacts.elastic.co/downloads/logstash/logstash-7.2.0.tar.gz

• 2.2、解压安装包

tar -zxvf logstash-7.2.0.tar.gz

• 2.3、下载hina_analytics插件

下载地址:hina_analytics插件

- 2.4、安装hina_analytics插件
- 安装之前先修改Gemfile中的source,可以提高安装速度

vim Gemfile

修改为source为 https://gems.ruby-china.com

• 把下载的插件移动到安装包下,执行以下命令,执行时间可能会较长,请耐心等待~~

./bin/logstash-plugin install --no-verify --local logstash-output-hina_analytics-1.0.3.gem

• 安装完成后可以查看

./bin/logstash-plugin list

• 看见新安装的插件 logstash-output-hina_analytics 证明安装成功。
• 2.5、hina_analytics插件参数配置

参数名	类型	必填	说明
url	list	是	海纳云的数据接收地址, 可同时配置多个数据接收地址,例如:url => [" http://IP1:8088/hina-cloud- engine/hn "," http://IP1:8088/hina-cloud- engine/hn "]。
project	string	否	项目名,不写默认为 default ,配置后会覆盖事件中和 url 中指定的 project。优先级为:project 参数配置 > 事件中指定 > url 中指定。
json_check	boolean	否	是否校验数据格式为JSON,默认否,配置为true, 则读取的每条数据必须为JSON,非JSON不处理
compress	boolean	否	是否压缩发送数据,默认否,支持gzip压缩
thread	number	否	请求发送并发线程数,默认3

• 2.6、Logstash 配置

配置中主要包含 input、filter 和 output 三部分, Logstash 处理日志数据只需配置 input 和 output 即可

hina_output.conf 参考示例:

```
#使用 beats 作为输入
input {
    beats {
        port => "5566"
    }
}
# 使用 hina_analytics 作为输出
output{
    hina_analytics {
        url => "https://xxx.xxx.com/hina-cloud-engine/hn?token=xxx"
    }
}
```

• 2.7、Logstash 启动

指定 config/hina_output.conf 为输入输出配置文件启动

nohup bin/logstash -f config/hina_output.conf >/dev/null 2>&1 &

三、Filebeat使用

•3.1、下载安装包

wget https://artifacts.elastic.co/downloads/beats/filebeat/filebeat-7.2.0-linux-x86_64.tar.gz

• 3.2、解压安装包

tar -zxvf filebeat-7.2.0-linux-x86_64.tar.gz

• 3.3、Filebeat 配置

进入解压目录

vim filebeat.yml

filebeat.yml 参考示例:

```
# Filebeat 收集 /data/log 目录下所有以log结尾的日志文件
filebeat.inputs:
    type: log
paths:
        - /data/log/*.log
# 将数据发送至Logstash
```

output.logstash: hosts: [<mark>"ip:5566"</mark>]

• 3.4、启动Filebeat

指定 filebeat.yml 为输入输出配置文件启动

```
nohup ./filebeat -c filebeat.yml >/dev/null 2>&1 &
```

附件

logstash-output-hina_analytics-1.0.3.gem

SQL手册

SQL手册介绍

SQL手册主要介绍HINA云对数仓的DML语句、SQL函数、数据类型说明,使用事件属性、用户属性、SQL查询功能时可参考该文档。

HINA云数仓选用的是Apache Doris,是一个基于 MPP 架构的高性能、实时的分析型数据库,以极速易用的特点被人们所熟知, 仅需亚秒级响应时间即可返回海量数据下的查询结果,不仅可以支持高并发的点查询场景,也能支持高吞吐的复杂分析场景。 DML

SELECT

SELECT Name SELECT description 主要介绍Select语法使用 语法: SELECT [ALL | DISTINCT | DISTINCTROW] select_expr [, select_expr ...] [FROM table_references [PARTITION partition_list] [WHERE where_condition] [GROUP BY {col_name | expr | position} [ASC | DESC], ... [WITH ROLLUP]] [HAVING where_condition] [ORDER BY {col_name | expr | position} [ASC | DESC], ...] [LIMIT {[offset,] row_count | row_count OFFSET offset}] [INTO OUTFILE 'file_name'] 语法说明:

- 1. select_expr, ··· 检索并在结果中显示的列,使用别名时,as为自选。
- 2. select_expr, ··· 检索的目标表 (一个或者多个表 (包括子查询产生的临时表)
- 3. where_definition 检索条件(表达式),如果存在WHERE子句,其中的条件对行数据进行筛选。where_condition是一个表达 式,对于要选择的每一行,其计算结果为true。如果没有WHERE子句,该语句将选择所有行。在WHERE表达式中,您可以 使用除聚合函数之外的任何MySQL支持的函数和运算符
- 4. ALL | DISTINCT : 对结果集进行刷选, all 为全部, distinct/distinctrow 将刷选出重复列,默认为all
- 5. INTO OUTFILE 'file_name' :保存结果至新文件 (之前不存在)中,区别在于保存的格式。
- 6. Group [asc/desc]by having : 对结果集进行分组, having 出现则对 group by 的结果进行刷选,
- 7. Order by : 对最后的结果进行排序, Order by 通过比较一列或者多列的大小来对结果集进行排序。

Order by 是比较耗时耗资源的操作,因为所有数据都需要发送到 1 个节点后才能排序,排序操作相比不排序操作需要更多的内存。

如果需要返回前 N 个排序结果,需要使用 LIMIT 从句;为了限制内存的使用,如果用户没有指定 LIMIT 从句,则默认返回 前 65535 个排序结果。

8. Limit n :限制输出结果中的行数, limit m,n 表示从第m行开始输出n条记录。

9. Having 从句不是过滤表中的行数据,而是过滤聚合函数产出的结果。

通常来说 having 要和聚合函数 (例如: COUNT(), SUM(), AVG(), MIN(), MAX())以及 group by 从句一起使 用。

10. SELECT 支持使用 PARTITION 显式分区选择,其中包含 table_reference 中表的名称后面的分区或子分区(或两者) 列表

语法约束:

- 1. SELECT也可用于检索计算的行而不引用任何表。
- 2. 所有的字句必须严格地按照上面格式排序,一个 HAVING 子句必须位于 GROUP BY 子句之后,并位于 ORDER BY 子句之前。
- 3. 别名关键词 AS 自选。别名可用于 group by, order by 和 having
- 4. Where 子句:执行 WHERE 语句以确定哪些行应被包含在 GROUP BY 部分中,而 HAVING 用于确定应使用结果集中的哪些 行。
- 5. HAVING 子句可以引用总计函数,而WHERE子句不能引用,如 count,sum,max,min,avg,同时,where 子句可以引用除总计函数外的其他函数。Where 子句中不能使用列别名来定义条件。
- 6. Group by 后跟 with rollup 可以对结果进行一次或者多次统计。

联接查询:

支持一下JOIN语法

JION
table_references:
table_reference [, table_reference]
table_reference:
table_factor
join_table
table_factor:
tbl_name [[AS] alias]
[{USE IGNORE FORCE} INDEX (key_list)]
<pre>(table_references)</pre>
<pre> { OJ table_reference LEFT OUTER JOIN table_reference</pre>
ON conditional_expr }
join_table:
table_reference [INNER CROSS] JOIN table_factor [join_condition]
<pre>table_reference STRAIGHT_JOIN table_factor</pre>
<pre> table_reference STRAIGHT_JOIN table_factor ON condition</pre>
table_reference LEFT [OUTER] JOIN table_reference join_condition
<pre>table_reference NATURAL [LEFT [OUTER]] JOIN table_factor</pre>
table_reference RIGHT [OUTER] JOIN table_reference join_condition
table_reference NATURAL [RIGHT [OUTER]] JOIN table_factor
join_condition:
ON conditional expr

-

UNION语法:

```
SELECT ...
UNION [ALL| DISTINCT] SELECT .....
[UNION [ALL| DISTINCT] SELECT ...]
```

UNION 用于将多个 SELECT 语句 的结果组合 到单个结果集中。

第一个 SELECT 语句中的列名称用作返回结果的列名称。在每个 SELECT 语句的 相应位置列出的选定列 应具有相同的数据类型。(例如,第一个语句选择的第一列应该与其他语句选择的第一列具有相同的类型。)

默认行为 UNION 是从结果中删除重复的行。可选 DISTINCT 关键字除了默认值之外没有任何效果,因为它还指定了重 复行删除。使用可选 ALL 关键字,不会发生重复行删除,结果包括所有 SELECT 语句中的 所有匹配行

WITH语句:

要指定公用表表达式,请使用 WITH 具有一个或多个逗号分隔子句的子句。每个子条款都提供一个子查询,用于生成结果 集,并将名称与子查询相关联。下面的示例定义名为的CTE cte1 和 cte2 中 WITH 子句,并且是指在它们的顶层 SELECT 下面的 WITH 子句:

WITH

```
cte1 AS (SELECT a, b FROM table1) ,
  cte2 AS (SELECT c, d FROM table2)
SELECT b, d FROM cte1 JOIN cte2
WHERE cte1.a = cte2.c;
```

在包含该 WITH 子句的语句中,可以引用每个 CTE 名称以访问相应的 CTE 结果集。

CTE 名称可以在其他 CTE 中引用,从而可以基于其他 CTE 定义 CTE。

CTE 可以引用自身来定义递归 CTE 。 递归 CTE 的常见应用包括分层或树状结构数据的序列生成和遍历。

example

1. 查询年龄分别是 18,20,25 的学生姓名

select Name from student where age in (18,20,25);

2. GROUP BY 示例

```
--查询tb_book表, 按照type分组, 求每类图书的平均价格, select type,avg(price) from tb_book group by type;
```

3. DISTINCT 使用

```
--查询tb_book表, 除去重复的type数据
select distinct type from tb_book;
```

4. ORDER BY 示例

对查询结果进行升序(默认)或降序(DESC)排列。升序NULL在最前面,降序NULL在最后面

```
--查询tb_book表中的所有记录,按照id降序排列,显示三条记录
select * from tb_book order by id desc limit 3;
```

5. LIKE模糊查询

可实现模糊查询,它有两种通配符:	8 和	_ ,	%	可以匹配一个或多个字符,	_	可以匹配一个字符
查找所有第二个字符是h的图书						
<pre>select * from tb_book where nam</pre>	e like('	'_ <mark>h%'</mark>);				

6. LIMIT限定结果行数

--1.降序显示3条记录 select * from tb_book order by price desc limit 3;

--2.从id=1显示4条记录 select * from tb_book where id limit 1,4;

7. CONCAT联合多列

-*-把name和price合并成一个新的字符串输出* select id,concat(name,<mark>":"</mark>,price) as info,type from tb_book;

8. 使用函数和表达式

--*计算tb_book表中各类图书的总价格* select sum(price) as total,type from tb_book group by type; --*price打八折* select *,(price * 0.8) as "八折" from tb_book;

9. UNION 示例

SELECT a FROM t1 WHERE a = 10 AND B = 1 ORDER by LIMIT 10 UNION SELECT a FROM t2 WHERE a = 11 AND B = 2 ORDER by LIMIT 10;

10. WITH 子句示例

```
WITH cte AS
(
SELECT 1 AS col1, 2 AS col2
UNION ALL
SELECT 3, 4
)
SELECT col1, col2 FROM cte;
```

11. JOIN 示例

```
SELECT * FROM t1 LEFT JOIN (t2, t3, t4)
ON (t2.a = t1.a AND t3.b = t1.b AND t4.c = t1.c)
```

等同于

```
SELECT * FROM t1 LEFT JOIN (t2 CROSS JOIN t3 CROSS JOIN t4)
ON (t2.a = t1.a AND t3.b = t1.b AND t4.c = t1.c)
```

12. INNER JOIN

```
SELECT t1.name, t2.salary
FROM employee AS t1 INNER JOIN info AS t2 ON t1.name = t2.name;
SELECT t1.name, t2.salary
FROM employee t1 INNER JOIN info t2 ON t1.name = t2.name;
```

13. LEFT JOIN

```
SELECT left_tbl.*
FROM left_tbl LEFT JOIN right_tbl ON left_tbl.id = right_tbl.id
WHERE right_tbl.id IS NULL;
```

14. RIGHT JOIN

keywords

SELECT

Best Practice

- 1. 关于SELECT子句的一些附加知识
 - 可以使用AS alias_name为select_expr指定别名。别名用作表达式的列名,可用于GROUP BY, ORDER BY或HAVING子句。AS关键字是在指定列的别名时养成使用AS是一种好习惯。
 - FROM后的table_references指示参与查询的一个或者多个表。如果列出了多个表,就会执行JOIN操作。而对于每一个指 定表,都可以为其定义别名
 - 。 SELECT后被选择的列,可以在ORDER IN和GROUP BY中,通过列名、列别名或者代表列位置的整数(从1开始)来引用

```
SELECT college, region, seed FROM tournament
    ORDER BY region, seed;
SELECT college, region AS r, seed AS s FROM tournament
    ORDER BY r, s;
SELECT college, region, seed FROM tournament
    ORDER BY 2, 3;
```

- 如果ORDER BY出现在子查询中,并且也应用于外部查询,则最外层的ORDER BY优先。
- 如果使用了GROUP BY,被分组的列会自动按升序排列(就好像有一个ORDER BY语句后面跟了同样的列)。如果要避免GROUP BY因为自动排序生成的开销,添加ORDER BY NULL可以解决:

SELECT a, COUNT(b) FROM test_table GROUP BY a ORDER BY NULL;

• 当使用ORDER BY或GROUP BY对SELECT中的列进行排序时,服务器仅使用max_sort_length系统变量指示的初始字节数对值进行排序。

• Having子句一般应用在最后,恰好在结果集被返回给MySQL客户端前,且没有进行优化。(而LIMIT应用在HAVING后)

SQL标准要求:HAVING必须引用在GROUP BY列表中或者聚合函数使用的列。然而,MySQL对此进行了扩展,它允许 HAVING引用Select子句列表中的列,还有外部子查询的列。

如果HAVING引用的列具有歧义,会有警告产生。下面的语句中,col2具有歧义:

SELECT COUNT(col1) AS col2 FROM t GROUP BY col2 HAVING col2 = 2;

- 切记不要在该使用WHERE的地方使用HAVING。HAVING是和GROUP BY搭配的。
- HAVING子句可以引用聚合函数,而WHERE不能。

SELECT user, MAX(salary) FROM users GROUP BY user HAVING MAX(salary) > 10;

• LIMIT子句可用于约束SELECT语句返回的行数。LIMIT可以有一个或者两个参数,都必须为非负整数。

/*取回结果集中的6~15行*/ SELECT * FROM tbl LIMIT 5,10; /*那如果要取回一个设定某个偏移量之后的所有行,可以为第二参数设定一个非常大的常量。以下查询取回从第96行起 的所有数据*/ SELECT * FROM tbl LIMIT 95,18446744073709551615; /*若LIMIT只有一个参数,则参数指定应该取回的行数,偏移量默认为0,即从第一行起*/

- SELECT…INTO可以让查询结果写入到文件中
- 2. SELECT关键字的修饰符
 - **。** 去重

ALL和DISTINCT修饰符指定是否对结果集中的行(应该不是某个列)去重。

ALL是默认修饰符,即满足要求的所有行都要被取回来。

DISTINCT删除重复的行。

- 3. 子查询的主要优势
 - 子查询允许结构化的查询,这样就可以把一个语句的每个部分隔离开。
 - 有些操作需要复杂的联合和关联。子查询提供了其它的方法来执行这些操作

4. 加速查询

- 尽可能利用分区分桶作为数据过滤条件,减少数据扫描范围
- 充分利用前缀索引字段作为数据过滤条件加速查询速度

5. UNION

• 只使用 union 关键词和使用 union disitnct 的效果是相同的。由于去重工作是比较耗费内存的,因此使用 union all 操作 查询速度会快些,耗费内存会少些。如果用户想对返回结果集进行 order by 和 limit 操作,需要将 union 操作放在子查 询中,然后 select from subquery,最后把 subquery 和 order by 放在子查询外面。

select * from (select age from student_01 union all select age from student_02) as t1
order by age limit 4;

+ -				- +	
	ā	age			
+ -				- +	
		18		1	
		19		1	
		20		1	
		21		1	
+ -				- +	
4	rows	in	set	(0.01	sec

6. JOIN

- 在 inner join 条件里除了支持等值 join , 还支持不等值 join , 为了性能考虑 , 推荐使用等值 join。
- 其它 join 只支持等值 join

SQL函数

日期函数

convert_tz

convert_tz

description

Syntax

DATETIME CONVERT_TZ(DATETIME dt, VARCHAR from_tz, VARCHAR to_tz)

转换datetime值,从 from_tz 给定时区转到 to_tz 给定时区,并返回结果值。 如果参数无效该函数返回NULL。

Example

mysql> select convert_tz('2019-08-01 13:21:03', 'Asia/Shanghai', 'America/Los_Angeles'); +convert_tz('2019-08-01 13:21:03', 'Asia/Shanghai', 'America/Los_Angeles') | ------+--+ 2019-07-31 22:21:03 -----+mysql> select convert_tz('2019-08-01 13:21:03', '+08:00', 'America/Los_Angeles'); ---------+ +-convert_tz('2019-08-01 13:21:03', '+08:00', 'America/Los_Angeles') | -----+---+ 2019-07-31 22:21:03 1 -----+--

keywords

CONVERT_TZ

curdate,current_date

curdate,current_date

description

Syntax

DATE CURDATE()

获取当前的日期,以DATE类型返回

Examples

<pre>mysql> SELECT CURDATE();</pre>
++
CURDATE()
++
2019-12-20
++
<pre>mysql> SELECT CURDATE() + 0;</pre>
++
CURDATE() + 0
++
20191220
++

keywords

CURDATE, CURRENT_DATE

current_timestamp

current_timestamp

description

Syntax

DATETIME CURRENT_TIMESTAMP()

获得当前的时间,以Datetime类型返回

example

```
mysql> select current_timestamp();
+-----+
| current_timestamp() |
+-----+
| 2019-05-27 15:59:33 |
+-----+
```

```
DATETIMEV2 CURRENT_TIMESTAMP(INT precision)
```

获得当前的时间,以DatetimeV2类型返回 precision代表了用户想要的秒精度,当前精度最多支持到微秒,即precision取值范围为[0, 6]。

example

```
mysql> select current_timestamp(3);
+----+
| current_timestamp(3) |
+---+
| 2022-09-06 16:18:00.922 |
+---++
```

注意:

- 1. 当前只有DatetimeV2数据类型可支持秒精度
- 2. 受限于JDK实现,如果用户使用JDK8构建FE,则精度最多支持到毫秒(小数点后三位),更大的精度位将全部填充0。如果 用户有更高精度需求,请使用JDK11。

keywords

CURRENT_TIMESTAMP, CURRENT, TIMESTAMP

curtime,current_time

curtime,current_time

Syntax

TIME CURTIME()

Description

获得当前的时间,以TIME类型返回

Examples

mysql>	select	curr	<pre>ent_time();</pre>
+		+	
curre	ent_time	e()	
+		+	
15:25	5:47		
+		+	

keywords

CURTIME, CURRENT_TIME

date_add

date_add

description

Syntax

INT DATE_ADD(DATETIME date,INTERVAL expr type)

向日期添加指定的时间间隔。

date 参数是合法的日期表达式。

expr 参数是您希望添加的时间间隔。

type 参数可以是下列值: YEAR, MONTH, DAY, HOUR, MINUTE, SECOND

example

```
mysql> select date_add('2010-11-30 23:59:59', INTERVAL 2 DAY);
+-----+
| date_add('2010-11-30 23:59:59', INTERVAL 2 DAY) |
+----+
| 2010-12-02 23:59:59 |
+----+
```

keywords

DATE_ADD, DATE, ADD

date_format

date_format

description

Syntax

VARCHAR DATE_FORMAT(DATETIME date, VARCHAR format)

将日期类型按照format的类型转化为字符串, 当前支持最大128字节的字符串,如果返回值长度超过128,则返回NULL

date 参数是合法的日期。format 规定日期/时间的输出格式。

可以使用的格式有:

%a | 缩写星期名

- %b | 缩写月名
- %c | 月,数值
- %D|带有英文前缀的月中的天
- %d | 月的天,数值(00-31)
- %e | 月的天,数值(0-31)

%f|微秒

- %H | 小时 (00-23)
- %h | 小时 (01-12)
- %I | 小时 (01-12)
- %i | 分钟,数值(00-59)
- %j | 年的天 (001-366)
- %k | 小时 (0-23)
- %||小时(1-12)
- %M | 月名
- %m | 月,数值(00-12)
- %p | AM 或 PM
- %r | 时间,12-小时(hh:mm:ss AM 或 PM)

%S | 秒(00-59)

%s | 秒(00-59)

- %T | 时间, 24-小时 (hh:mm:ss)
- %U | 周 (00-53) 星期日是一周的第一天
- %u | 周 (00-53) 星期一是一周的第一天
- %V | 周 (01-53) 星期日是一周的第一天, 与 %X 使用
- %v | 周 (01-53) 星期一是一周的第一天, 与 %x 使用
- %W | 星期名
- %w | 周的天 (0=星期日, 6=星期六)
- %X | 年,其中的星期日是周的第一天,4位,与%V 使用
- %x | 年,其中的星期一是周的第一天,4位,与%v 使用
- %Y | 年,4位
- %y | 年,2位
- %% | 用于表示 %
- 还可以使用三种特殊格式:
- yyyyMMdd
- yyyy-MM-dd

yyyy-MM-dd HH:mm:ss

example

```
mysql> select date_format('2009-10-04 22:23:00', '%W %M %Y');
+----+
| date_format('2009-10-04 22:23:00', '%W %M %Y') |
+----+
| Sunday October 2009 |
+----+
mysql> select date_format('2007-10-04 22:23:00', '%H:%i:%s');
+-----+
| date_format('2007-10-04 22:23:00', '%H:%i:%s') |
+----+
22:23:00
                      +-----+
mysql> select date_format('1900-10-04 22:23:00', '%D %y %a %d %m %b %j');
   ----------+
+---
| date_format('1900-10-04 22:23:00', '%D %y %a %d %m %b %j') |
+-----+
| 4th 00 Thu 04 10 Oct 277
+------
mysql> select date_format('1997-10-04 22:23:00', '%H %k %I %r %T %S %w');
   ------
| date_format('1997-10-04 22:23:00', '%H %k %I %r %T %S %w') |
+------
22 22 10 10:23:00 PM 22:23:00 00 6
```

+-----+ mysql> select date_format('1999-01-01 00:00:00', '%X %V'); +----+ | date_format('1999-01-01 00:00:00', '%X %V') | +----+ 1998 52 +----+ mysql> select date_format('2006-06-01', '%d'); +-----+ | date_format('2006-06-01 00:00:00', '%d') | +----+ 01 +-----+ mysql> select date_format('2006-06-01', '%%%d'); +-----+ | date_format('2006-06-01 00:00:00', '%%%d') | +----+ 801 +----+

keywords

DATE_FORMAT, DATE, FORMAT

Footer © 2022 GitHub, Inc. Footer navigation Terms

date_sub

date_sub

description

Syntax

INT DATE_SUB(DATETIME date,INTERVAL expr type)

从日期减去指定的时间间隔

date 参数是合法的日期表达式。

expr 参数是您希望添加的时间间隔。

type 参数可以是下列值: YEAR, MONTH, DAY, HOUR, MINUTE, SECOND

example

```
mysql> select date_sub('2010-11-30 23:59:59', INTERVAL 2 DAY);
+-----+
| date_sub('2010-11-30 23:59:59', INTERVAL 2 DAY) |
+----+
| 2010-11-28 23:59:59 |
+----++
```

keywords

DATE_SUB, DATE, SUB

datediff

datediff

description

Syntax

DATETIME DATEDIFF(DATETIME expr1,DATETIME expr2)

计算expr1 - expr2,结果精确到天。

expr1 和 expr2 参数是合法的日期或日期/时间表达式。

注释:只有值的日期部分参与计算。

example

keywords

DATEDIFF

day

day

description

Syntax

INT DAY(DATETIME date)

获得日期中的天信息,返回值范围从1-31。

参数为Date或者Datetime类型

example

<pre>mysql> select day('1987-01-31');</pre>
++
day('1987-01-31 00:00:00')
++
31
++

keywords

DAY

dayname

dayname

description

Syntax

VARCHAR DAYNAME(DATE)

返回日期对应的日期名字

参数为Date或者Datetime类型

example

mysql> select dayname('2007-02-03 00:00:00');
+-----+
| dayname('2007-02-03 00:00:00') |
+----+
| Saturday |
+----+

keywords

DAYNAME

dayofmonth

dayofmonth

description

Syntax

INT DAYOFMONTH(DATETIME date)

获得日期中的天信息,返回值范围从1-31。

参数为Date或者Datetime类型

example

```
mysql> select dayofmonth('1987-01-31');
+----+
| dayofmonth('1987-01-31 00:00:00') |
+----+
| 31 |
+----+
```

keywords

DAYOFMONTH

dayofweek

dayofweek

description

Syntax

INT DAYOFWEEK(DATETIME date)

DAYOFWEEK函数返回日期的工作日索引值,即星期日为1,星期一为2,星期六为7

参数为Date或者Datetime类型或者可以cast为Date或者Datetime类型的数字

example

keywords

DAYOFWEEK

dayofyear

dayofyear

description

Syntax

INT DAYOFYEAR(DATETIME date)

获得日期中对应当年中的哪一天。

参数为Date或者Datetime类型

example

keywords

DAYOFYEAR

Footer © 2022 GitHub, Inc. Footer navigation Terms Privacy

from_days

from_days

description

Syntax

DATE FROM_DAYS(INT N)

通过距离0000-01-01日的天数计算出哪一天

example

mysql	> select	from_da	ys(730669);
+		+	
fro	m_days(73	30669)	
+		+	
200	0-07-03		
+		+	

keywords

FROM_DAYS, FROM, DAYS

from_unixtime

from_unixtime

description

Syntax

DATETIME FROM_UNIXTIME(INT unix_timestamp[, VARCHAR string_format])

将 unix 时间戳转化为对应的 time 格式, 返回的格式由 string_format 指定

支持date_format中的format格式,默认为%Y-%m-%d%H:%i:%s

传入的是整形,返回的是字符串类型

其余 string_format 格式是非法的,返回NULL

如果给定的时间戳小于 0 或大于 253402271999,则返回 NULL。即时间戳范围是:

1970-01-01 00:00:00 ~ 9999-12-31 23:59:59

example



keywords

FROM_UNIXTIME, FROM, UNIXTIME

hour

hour

description

Syntax

INT HOUR(DATETIME date)

获得日期中的小时的信息,返回值范围从0-23。

参数为Date或者Datetime类型

example

```
mysql> select hour('2018-12-31 23:59:59');
+-----+
| hour('2018-12-31 23:59:59') |
+----+
| 23 |
+----+
```

keywords

HOUR

makedate

makedate

description

Syntax

DATE MAKEDATE(INT year, INT dayofyear) 返回指定年份和dayofyear构建的日期。dayofyear必须大于0,否则结果为空。

example

mysql> select mak	edate(2021,1), maked	date(<mark>2021,100</mark>), makeda	te(2021,400)
+	+	+	+
makedate(2021,	1) makedate(2021,	100) makedate(2021,	400)
+	+	+	+
2021-01-01	2021-04-10	2022-02-04	
+	+	+	+

keywords

YEARWEEK

minute

minute

description

Syntax

INT MINUTE(DATETIME date)

获得日期中的分钟的信息,返回值范围从0-59。

参数为Date或者Datetime类型

example

```
mysql> select minute('2018-12-31 23:59:59');
+----+
| minute('2018-12-31 23:59:59') |
+----+
| 59 |
+----+
```

keywords

MINUTE

month

month

description

Syntax

INT MONTH(DATETIME date)

返回时间类型中的月份信息,范围是1,12

参数为Date或者Datetime类型

example

<pre>mysql> select month('1987-01-01');</pre>
++
month('1987-01-01 00:00:00')
++
1
++

keywords

MONTH

monthname

monthname

description

Syntax

VARCHAR MONTHNAME(DATE)

返回日期对应的月份名字

参数为Date或者Datetime类型

example

mysql> select monthname('2008-02-03 00:00:00');
+----+
| monthname('2008-02-03 00:00:00') |
+----+
| February |
+----+

keywords

MONTHNAME

now

now

description

Syntax

DATETIME NOW()

获得当前的时间,以Datetime类型返回

example

my	sql>	select	t now();	
+-					+
	now())			
+-					•+
	2019-	-05-27	15:58	:25	
+-					+

DATETIMEV2 NOW(INT precision)

获得当前的时间,以DatetimeV2类型返回 precision代表了用户想要的秒精度,当前精度最多支持到微秒,即precision取值范围为[0, 6]。

example

```
mysql> select now(3);
+----+
| now(3) |
+----+
| 2022-09-06 16:13:30.078 |
+----+
```

注意:

1. 当前只有DatetimeV2数据类型可支持秒精度

2. 受限于JDK实现,如果用户使用JDK8构建FE,则精度最多支持到毫秒(小数点后三位),更大的精度位将全部填充0。如果 用户有更高精度需求,请使用JDK11。

keywords

NOW
second

second

description

Syntax

INT SECOND(DATETIME date)

获得日期中的秒的信息,返回值范围从0-59。

参数为Date或者Datetime类型

example

```
mysql> select second('2018-12-31 23:59:59');
+----+
| second('2018-12-31 23:59:59') |
+----+
| 59 |
+----+
```

keywords

SECOND

str_to_date

str_to_date

description

Syntax

DATETIME STR_TO_DATE(VARCHAR str, VARCHAR format)

通过format指定的方式将str转化为DATE类型,如果转化结果不对返回NULL

支持的format格式与date_format一致

example

<pre>mysql> select str_to_date('2014-12-21 12:34:56', '%Y-%m-%d %H:%i:%s');</pre>
str_to_date('2014-12-21 12:34:56', '%Y-‰m-%d %H:%i:%s')
++ 2014-12-21 12:34:56
++
$m_{x,x,y} = m_{x,y,y} + m_{x$
++
str_to_date('2014-12-21 12:34%3A56', '%Y-%m-%d %H:%i%%3A%s')
++
++
murgly collect str to dota (100442 Mandau) (8764/ 64/)
++
str_to_date('200442 Monday', '%X%V %W')
++
++
<pre>mysql> select str_to_date("2020-09-01", "%Y-%m-%d %H:%1:%S"); ++</pre>
str_to_date('2020-09-01', '%Y-%m-%d %H:%i:%s')
+
1 row in set (0.01 sec)

keywords

STR_TO_DATE, STR, TO, DATE

time_round

time_round description Syntax DATETIME TIME_ROUND(DATETIME expr) DATETIME TIME_ROUND(DATETIME expr, INT period) DATETIME TIME_ROUND(DATETIME expr, DATETIME origin) DATETIME TIME_ROUND(DATETIME expr, INT period, DATETIME origin) TIME_ROUND 由两部分组成,每部分由以下可选值组成 函数名 TIME : SECOND , MINUTE , HOUR , DAY , WEEK , MONTH , YEAR . ROUND : FLOOR , CEIL . 返回 expr 的上/下界。 • period 指定每个周期有多少个 TIME 单位组成,默认为 1。 origin 指定周期的开始时间,默认为 1970-01-01T00:00:00 , WEEK 的默认开始时间为 1970-01-04T00:00:00 , • 即周日。可以比 expr 大。 • 请尽量选择常见 period ,如 3 MONTH ,90 MINUTE 等,如设置了非常用 period ,请同时指定 origin 。 example MySQL> SELECT YEAR_FLOOR('20200202000000'); +----+ | year_floor('20200202000000') | +----+ 2020-01-01 00:00:00 +------MySQL> SELECT MONTH_CEIL(CAST('2020-02-02 13:09:20' AS DATETIME), 3); --quarter +-----month_ceil(CAST('2020-02-02 13:09:20' AS DATETIME), 3) | 2020-04-01 00:00:00 +------MySQL> SELECT WEEK_CEIL('2020-02-02 13:09:20', '2020-01-06'); --monday

++
| week_ceil('2020-02-02 13:09:20', '2020-01-06 00:00:00') |
+----+
| 2020-02-03 00:00:00 |
+----+

keywords

TIME_ROUND

timediff

timediff

description

Syntax

TIME TIMEDIFF(DATETIME expr1, DATETIME expr2)

TIMEDIFF返回两个DATETIME之间的差值

TIMEDIFF函数返回表示为时间值的expr1 - expr2的结果,返回值为TIME类型

example

mysql> SELECT TIMEDIFF(now(),utc_timestamp()); +----+ | timediff(now(), utc_timestamp()) | +----+ 08:00:00 +----+ mysql> SELECT TIMEDIFF('2019-07-11 16:59:30','2019-07-11 16:59:21'); +-----+ timediff('2019-07-11 16:59:30', '2019-07-11 16:59:21') | +----+ 00:00:09 +-----+ mysql> SELECT TIMEDIFF('2019-01-01 00:00:00', NULL); +-----+ timediff('2019-01-01 00:00:00', NULL) | +----+ | NULL +-----+

keywords

TIMEDIFF

timestampadd

timestampadd

description

Syntax

DATETIME TIMESTAMPADD(unit, interval, DATETIME datetime_expr)

将整数表达式间隔添加到日期或日期时间表达式datetime_expr中。

interval的单位由unit参数给出,它应该是下列值之一:

SECOND, MINUTE, HOUR, DAY, WEEK, MONTH, or YEAR.

example

<pre>mysql> SELECT TIMESTAMPADD(MINUTE,1,'2019-01-02');</pre>
++
timestampadd(MINUTE, 1, '2019-01-02 00:00:00')
++
2019-01-02 00:01:00
++
<pre>mysql> SELECT TIMESTAMPADD(WEEK,1,'2019-01-02');</pre>
++
timestampadd(WEEK, 1, '2019-01-02 00:00:00')
++
2019-01-09 00:00:00
++

keywords

TIMESTAMPADD

timestampdiff

timestampdiff

description

Syntax

INT TIMESTAMPDIFF(unit,DATETIME datetime_expr1, DATETIME datetime_expr2)

返回datetime_expr2-datetime_expr1,其中datetime_expr1和datetime_expr2是日期或日期时间表达式。

结果(整数)的单位由unit参数给出。interval的单位由unit参数给出,它应该是下列值之一:

SECOND, MINUTE, HOUR, DAY, WEEK, MONTH, or YEAR.

example

MySQL> SELECT TIMESTAMPDIFF(MONTH, '2003-02-01', '2003-05-01'); +-----+ timestampdiff(MONTH, '2003-02-01 00:00:00', '2003-05-01 00:00:00') | +------3 | +-----+ MySQL> SELECT TIMESTAMPDIFF(YEAR, '2002-05-01', '2001-01-01'); +-----+ timestampdiff(YEAR, '2002-05-01 00:00:00', '2001-01-01 00:00:00') | +-----+ -1 | +-----+ MySQL> SELECT TIMESTAMPDIFF(MINUTE, '2003-02-01', '2003-05-01 12:05:55'); +-----+ | timestampdiff(MINUTE, '2003-02-01 00:00:00', '2003-05-01 12:05:55') | +----+ 128885 +-----+

keywords

TIMESTAMPDIFF

to_date

to_date

description

Syntax

DATE TO_DATE(DATETIME)

返回 DATETIME 类型中的日期部分。

example

```
mysql> select to_date("2020-02-02 00:00:00");
+-----+
| to_date('2020-02-02 00:00:00') |
+----+
| 2020-02-02 |
+----+
```

keywords

TO_DATE

to_days

to_days

description

Syntax

INT TO_DAYS(DATETIME date)

返回date距离0000-01-01的天数

参数为Date或者Datetime类型

example

```
mysql> select to_days('2007-10-07');
+-----+
| to_days('2007-10-07') |
+----+
| 733321 |
+----+
```

keywords

TO_DAYS, TO, DAYS

unix_timestamp

unix_timestamp

description

Syntax

INT UNIX_TIMESTAMP(), UNIX_TIMESTAMP(DATETIME date), UNIX_TIMESTAMP(DATETIME date, STRING fmt),

将 Date 或者 Datetime 类型转化为 unix 时间戳。

如果没有参数,则是将当前的时间转化为时间戳。

参数需要是 Date 或者 Datetime 类型。

对于在 1970-01-01 00:00:00 之前或 2038-01-19 03:14:07 之后的时间, 该函数将返回 0。

Format 的格式请参阅 date_format 函数的格式说明。

该函数受时区影响。

example

```
mysql> select unix_timestamp();
+----+
| unix_timestamp() |
+----+
1558589570
+----+
mysql> select unix_timestamp('2007-11-30 10:30:19');
+-----+
| unix_timestamp('2007-11-30 10:30:19') |
+-----+
           1196389819
+-----+
mysql> select unix_timestamp('2007-11-30 10:30-19', '%Y-%m-%d %H:%i-%s');
+-----+
| unix_timestamp('2007-11-30 10:30-19') |
+-----+
          1196389819
+-----+
mysql> select unix_timestamp('2007-11-30 10:30%3A19', '%Y-%m-%d %H:%i%%3A%s');
+-----+
| unix_timestamp('2007-11-30 10:30%3A19') |
+-----+
            1196389819
+-----+
mysql> select unix_timestamp('1969-01-01 00:00:00');
+----+
| unix_timestamp('1969-01-01 00:00:00') |
+----+
0
+-----+
```

keywords

UNIX_TIMESTAMP, UNIX, TIMESTAMP

utc_timestamp

utc_timestamp

description

Syntax

DATETIME UTC_TIMESTAMP()

返回当前UTC日期和时间在 "YYYY-MM-DD HH:MM:SS" 或

"YYYYMMDDHHMMSS"格式的一个值

根据该函数是否用在字符串或数字语境中

example

```
mysql> select utc_timestamp(),utc_timestamp() + 1;
+-----+
| utc_timestamp() | utc_timestamp() + 1 |
+-----+
| 2019-07-10 12:31:18 | 20190710123119 |
+-----+
```

keywords

UTC_TIMESTAMP, UTC, TIMESTAMP

week

week

description

Syntax

INT WEEK(DATE date) INT WEEK(DATE date, INT mode)

参数为Date或者Datetime类型

example

mysql> select week('2020-1-1');
+----+
| week('2020-1-1') |
+----+
| 0 |
+----+

mysql> select week('2020-7-1',1);
+-----+
| week('2020-7-1', 1) |
+----+
| 27 |
+----+

keywords

WEEK

weekday

weekday

Description

Syntax

INT WEEKDAY (DATETIME date)

WEEKDAY函数返回日期的工作日索引值,即星期一为0,星期二为1,星期日为6

参数为Date或者Datetime类型或者可以cast为Date或者Datetime类型的数字

注意WEEKDAY和DAYOFWEEK的区别:

	+-		+ -		-+-		+-		+ -		+-		+-		- +
		Sun		Mon		Tues	s	Wed		Thu	r	Fri		Sat	
	+-		+-		-+-		+-		+-		-+-		+-		- +
weekday		6		0		1		2		3		4		5	
	+-		+-		-+-		+-		+-		-+-		+-		- +
dayofweek		1		2		3		4		5		6		7	
	+-		+ -		-+-		-+-		+ -		-+-		+-		+-+

example

<pre>mysql> select weekday('2019-06-25');</pre>
++
weekday('2019-06-25 00:00:00')
++
1
++
<pre>mysql> select weekday(cast(20190625 as date));</pre>
++
weekday(CAST(20190625 AS DATE))
++
1
++

keywords

WEEKDAY

weekofyear

weekofyear

description

Syntax

INT WEEKOFYEAR(DATETIME date)

获得一年中的第几周

参数为Date或者Datetime类型

example

keywords

WEEKOFYEAR

year

year

description

Syntax

INT YEAR(DATETIME date)

返回date类型的year部分,范围从1000-9999

参数为Date或者Datetime类型

example

<pre>mysql> select year('1987-01-01');</pre>
++
year('1987-01-01 00:00:00')
++
1987
++

keywords

YEAR

yearweek

yearweek

description

Syntax

```
INT YEARWEEK(DATE date)
INT YEARWEEK(DATE date, INT mode)
```

参数为Date或者Datetime类型

example

```
mysql> select yearweek('2021-1-1');
+-----+
| yearweek('2021-1-1') |
+----+
| 202052 |
+----+
```

```
mysql> select yearweek('2020-7-1');
+-----+
| yearweek('2020-7-1') |
+-----+
| 202026 |
+-----+
```

mysql> select yearweek('2024-12-30',1);
+-----+
| yearweek('2024-12-30 00:00:00', 1) |
+----+
| 202501 |
+----+

keywords

YEARWEEK

localtime,localtimestamp

localtime,localtimestamp

Syntax

DATETIME localtime() DATETIME localtimestamp()

description

获得当前的时间,以Datetime类型返回

Example

<pre>mysql> select localtime();</pre>
++
localtime()
++
2022-09-22 17:30:23
++
<pre>mysql> select localtimestamp();</pre>
++
<pre> localtimestamp() </pre>
++
2022-09-22 17:30:29

keywords

localtime , localtimestamp

quarter

quarter

Syntax

INT quarter(DATETIME date)

description

返回指定的日期所属季度,以INT类型返回

Example

keywords

quarter

last_day

last_day

Syntax

DATE last_day(DATETIME date)

Description

返回输入日期中月份的最后一天;所以返回的日期中,年和月不变,日可能是如下情况: '28'(非闰年的二月份), '29'(闰年的二 月份), '30'(四月,六月,九月,十一月), '31'(一月,三月,五月,七月,八月,十月,十二月)

example

```
mysql > select last_day('2000-02-03');
+----+
| last_day('2000-02-03 00:00:00') |
+----+
| 2000-02-29 |
+----+
```

keywords

LAST_DAY,DAYS

extract

extract

Syntax

INT extract(unit FROM DATETIME)

description

提取DATETIME某个指定单位的值。单位可以为year, month, day, hour, minute或者second

Example

<pre>mysql> select extract(year from '2022-09-22 17:01:30') as year,</pre>
-> extract(month from '2022-09-22 17:01:30') as month,
-> extract(day from '2022-09-22 17:01:30') as day,
-> extract(hour from '2022-09-22 17:01:30') as hour,
-> extract(minute from '2022-09-22 17:01:30') as minute,
-> extract(second from '2022-09-22 17:01:30') as second;
++
year month day hour minute second
++
2022 9 22 17 1 30
++

keywords

extract

date_trunc

date_trunc

Syntax

DATETIME DATE_TRUNC(DATETIME datetime,VARCHAR unit)

description

将datetime按照指定的时间单位截断。

datetime 参数是合法的日期表达式。

unit 参数是您希望截断的时间间隔,可选的值如下:[second,minute,hour,day,month,quarter,year]。 如果unit 不符合上述可选 值,结果将返回NULL。

example





keywords

DATE_TRUNC, DATE, TRUNC

minutes_add

minutes_add

Syntax

DATETIME MINUTES_ADD(DATETIME date, INT minutes)

description

从日期时间或日期加上指定分钟数

参数 date 可以是 DATETIME 或者 DATE 类型,返回类型为 DATETIME。

example

```
mysql> select minutes_add("2020-02-02", 1);
+-----+
| minutes_add('2020-02-02 00:00:00', 1) |
+-----+
| 2020-02-02 00:01:00 |
+----+
```

keywords

MINUTES_ADD

minutes_diff

minutes_diff

Syntax

INT minutes_diff(DATETIME enddate, DATETIME startdate)

description

开始时间到结束时间相差几分钟

example

```
mysql> select minutes_diff('2020-12-25 22:00:00','2020-12-25 21:00:00');
+-----+
| minutes_diff('2020-12-25 22:00:00', '2020-12-25 21:00:00') |
+-----+
| 60 |
+-----+
```

keywords

minutes_diff

minutes_sub

minutes_sub

Syntax

DATETIME MINUTES_SUB(DATETIME date, INT minutes)

description

从日期时间或日期减去指定分钟数

参数 date 可以是 DATETIME 或者 DATE 类型,返回类型为 DATETIME。

example

```
mysql> select minutes_sub("2020-02-02 02:02:02", 1);
+-----+
| minutes_sub('2020-02-02 02:02:02', 1) |
+----+
| 2020-02-02 02:01:02 |
+----+
```

keywords

MINUTES_SUB

seconds_add

seconds_add

Syntax

DATETIME SECONDS_ADD(DATETIME date, INT seconds)

description

从日期时间或日期加上指定秒数

参数 date 可以是 DATETIME 或者 DATE 类型,返回类型为 DATETIME。

example

```
mysql> select seconds_add("2020-02-02 02:02:02", 1);
+-----+
| seconds_add('2020-02-02 02:02:02', 1) |
+----+
| 2020-02-02 02:02:03 |
+----+
```

keywords

SECONDS_ADD

seconds_diff

Syntax

INT seconds_diff(DATETIME enddate, DATETIME startdate)

description

开始时间到结束时间相差几秒

example

mysql> select seconds_diff('2020-12-25 22:00:00','2020-12-25 21:00:00');
+-----+
| seconds_diff('2020-12-25 22:00:00', '2020-12-25 21:00:00') |
+-----+
| 3600 |
+----+

keywords

seconds_diff

seconds_sub

seconds_sub

Syntax

DATETIME SECONDS_SUB(DATETIME date, INT seconds)

description

从日期时间或日期减去指定秒数

参数 date 可以是 DATETIME 或者 DATE 类型,返回类型为 DATETIME。

example

```
mysql> select seconds_sub("2020-01-01 00:00:00", 1);
+-----+
| seconds_sub('2020-01-01 00:00:00', 1) |
+----+
| 2019-12-31 23:59:59 |
+----+
```

keywords

SECONDS_SUB

hours_add

hours_add

Syntax

DATETIME HOURS_ADD(DATETIME date, INT hours)

description

从日期时间或日期加上指定小时数

参数 date 可以是 DATETIME 或者 DATE 类型,返回类型为 DATETIME。

example

```
mysql> select hours_add("2020-02-02 02:02:02", 1);
+-----+
| hours_add('2020-02-02 02:02:02', 1) |
+-----+
| 2020-02-02 03:02:02 |
+----+
```

keywords

HOURS_ADD

hours_diff

hours_diff

###Syntax
INT hours_diff(DATETIME enddate, DATETIME startdate)

description

开始时间到结束时间相差几小时

example

mysql> select hours_diff('2020-12-25 22:00:00','2020-12-25 21:00:00');
+------+
| hours_diff('2020-12-25 22:00:00', '2020-12-25 21:00:00') |
+-----+
| 1 |
1 |
+----++

keywords

hours_diff

hours_sub

hours_sub

Syntax

DATETIME HOURS_SUB(DATETIME date, INT hours)

description

从日期时间或日期减去指定小时数

参数 date 可以是 DATETIME 或者 DATE 类型,返回类型为 DATETIME。

example

```
mysql> select hours_sub("2020-02-02 02:02:02", 1);
+-----+
| hours_sub('2020-02-02 02:02:02', 1) |
+-----+
| 2020-02-02 01:02:02 |
+----+
```

keywords

HOURS_SUB

days_add

days_add

Syntax

DATETIME DAYS_ADD(DATETIME date, INT days)

description

从日期时间或日期加上指定天数

参数 date 可以是 DATETIME 或者 DATE 类型,返回类型与参数 date 的类型一致。

example

mysql> select days_add(to_date("2020-02-02 02:02:02"), 1); +-----+ | days_add(to_date('2020-02-02 02:02:02'), 1) | +-----+ | 2020-02-03 | +----+

keywords

DAYS_ADD

days_diff

days_diff

Syntax

INT days_diff(DATETIME enddate, DATETIME startdate)

description

开始时间到结束时间相差几天

example

mysql> select days_diff('2020-12-25 22:00:00','2020-12-24 22:00:00'); +------+ | days_diff('2020-12-25 22:00:00', '2020-12-24 22:00:00') | +-----+ | 1 | +-----+

keywords

days_diff

days_sub

days_sub

Syntax

DATETIME DAYS_SUB(DATETIME date, INT days)

example

```
mysql> select days_sub("2020-02-02 02:02:02", 1);
+-----+
| days_sub('2020-02-02 02:02:02', 1) |
+----+
| 2020-02-01 02:02:02 |
+----+
```

keywords

DAYS_SUB
weeks_add

weeks_add

Syntax

DATETIME WEEKS_ADD(DATETIME date, INT weeks)

description

从日期加上指定星期数

参数 date 可以是 DATETIME 或者 DATE 类型,返回类型与参数 date 的类型一致。

example

```
mysql> select weeks_add("2020-02-02 02:02:02", 1);
+-----+
| weeks_add('2020-02-02 02:02:02', 1) |
+----+
| 2020-02-09 02:02:02 |
+----+
```

keywords

WEEKS_ADD

weeks_diff

weeks_diff

Syntax

INT weeks_diff(DATETIME enddate, DATETIME startdate)

description

开始时间到结束时间相差几星期

example

keywords

weeks_diff

weeks_sub

Syntax

DATETIME WEEKS_SUB(DATETIME date, INT weeks)

description

从日期时间或日期减去指定星期数

参数 date 可以是 DATETIME 或者 DATE 类型,返回类型与参数 date 的类型一致。

example

```
mysql> select weeks_sub("2020-02-02 02:02:02", 1);
+-----+
| weeks_sub('2020-02-02 02:02:02', 1) |
+----+
| 2020-01-26 02:02:02 |
+----+
```

keywords

WEEKS_SUB

months_add

months_add

Syntax

DATETIME MONTHS_ADD(DATETIME date, INT months)

description

从日期加上指定月份

参数 date 可以是 DATETIME 或者 DATE 类型,返回类型与参数 date 的类型一致。

example

```
mysql> select months_add("2020-01-31 02:02:02", 1);
+-----+
| months_add('2020-01-31 02:02:02', 1) |
+----+
| 2020-02-29 02:02:02 |
+----+
```

keywords

MONTHS_ADD

months_diff

months_diff

Syntax

INT months_diff(DATETIME enddate, DATETIME startdate)

description

开始时间到结束时间相差几个月

example

<pre>mysql> select months_diff('2020-12-25','2020-10-25');</pre>	
+	+
months_diff('2020-12-25 00:00:00', '2020-10-25 00:00:00	')
+	+
	2
+	+

keywords

months_diff

months_sub

months_sub

Syntax

DATETIME MONTHS_SUB(DATETIME date, INT months)

description

从日期时间或日期减去指定月份数

参数 date 可以是 DATETIME 或者 DATE 类型,返回类型与参数 date 的类型一致。

example

```
mysql> select months_sub("2020-02-02 02:02:02", 1);
+-----+
| months_sub('2020-02-02 02:02:02', 1) |
+----+
| 2020-01-02 02:02:02 |
+----+
```

keywords

MONTHS_SUB

years_add

years_add

syntax

DATETIME YEARS_ADD(DATETIME date, INT years)

description

从日期加上指定年数

参数 date 可以是 DATETIME 或者 DATE 类型,返回类型与参数 date 的类型一致。

example

```
mysql> select years_add("2020-01-31 02:02:02", 1);
+-----+
| years_add('2020-01-31 02:02:02', 1) |
+-----+
| 2021-01-31 02:02:02 |
+-----+
```

keywords

YEARS_ADD

years_diff

years_diff

Syntax

INT years_diff(DATETIME enddate, DATETIME startdate)

description

开始时间到结束时间相差几年

example

mysql> select years_diff('2020-12-25','2019-10-25');
+-----+
| years_diff('2020-12-25 00:00:00', '2019-10-25 00:00:00') |
+-----+
| 1 |
+----++
| 1 |

keywords

years_diff

years_sub

years_sub

Syntax

DATETIME YEARS_SUB(DATETIME date, INT years)

description

从日期时间或日期减去指定年数

参数 date 可以是 DATETIME 或者 DATE 类型,返回类型与参数 date 的类型一致。

example

```
mysql> select years_sub("2020-02-02 02:02:02", 1);
+-----+
| years_sub('2020-02-02 02:02:02', 1) |
+-----+
| 2019-02-02 02:02:02 |
+-----+
```

keywords

YEARS_SUB

字符串函数

append_trailing_char_if_absent

append_trailing_char_if_absent

description

Syntax

VARCHAR append_trailing_char_if_absent(VARCHAR str, VARCHAR trailing_char)

如果 str 字符串非空并且末尾不包含 trailing_char 字符,则将 trailing_char 字符附加到末尾。 trailing_char 只能包含一个字符,如果包含多个字符,将返回NULL

example

MySQL [test]> select append_trailing_char_if_absent('a','c'); -----+ | append_trailing_char_if_absent('a', 'c') | -----+ +--| ac +-----+ 1 row in set (0.02 sec) MySQL [test]> select append_trailing_char_if_absent('ac','c'); +---------| append_trailing_char_if_absent('ac', 'c') | +-------+ ac +-----+ 1 row in set (0.00 sec)

keywords

APPEND_TRAILING_CHAR_IF_ABSENT

数据类型

数据类型

BIGINT

BIGINT

description

BIGINT 8字节有符号整数,范围[-9223372036854775808,9223372036854775807]

keywords

BIGINT

INT

INT

description

INT 4字节有符号整数,范围[-2147483648, 2147483647]

keywords

INT

Footer

TINYINT

TINYINT

description

TINYINT 1字节有符号整数,范围[-128,127]

keywords

TINYINT

SMALLINT

SMALLINT

description

SMALLINT 2字节有符号整数,范围[-32768, 32767]

keywords

SMALLINT

LARGEINT

LARGEINT

description

LARGEINT 16字节有符号整数,范围[-2^127 + 1 ~ 2^127 - 1]

keyword

LARGEINT

BOOLEAN

BOOLEAN

description

BOOL, BOOLEAN 与TINYINT一样, 0代表false, 1代表true

keywords

BOOLEAN

DATE

DATE

description

```
DATE函数
Syntax:
DATE(expr)
将输入的类型转化为DATE类型
DATE类型
日期类型,目前的取值范围是['0000-01-01', '9999-12-31'],默认的打印形式是'YYYY-MM-DD'
```

example

```
mysql> SELECT DATE('2003-12-31 01:02:03');
-> '2003-12-31'
```

keywords

DATE

DATETIME

DATETIME

description

DATETIME

日期时间类型,取值范围是['0000-01-01 00:00:00', '9999-12-31 23:59:59']. 打印的形式是'YYYY-MM-DD HH:MM:SS'

keywords

DATETIME

DECIMAL

DECIMAL

description

DECIMAL(M[,D]) 高精度定点数, M 代表一共有多少个有效数字(precision), D 代表小数位有多少数字(scale), 有效数字 M 的范围是 [1, 27], 小数位数字数量 D 的范围是 [0, 9], 整数位数字数量的范围是 [1, 18], 另外, M 必须要大于等于 D 的取值。

默认值为 DECIMAL(9, 0)。

keywords

DECIMAL

DOUBLE

DOUBLE

description

DOUBLE 8字节浮点数

keywords

DOUBLE

FLOAT

FLOAT

description

FLOAT 4字节浮点数

keywords

FLOAT

CHAR

CHAR

description

CHAR(M) 定长字符串,M代表的是定长字符串的长度。M的范围是1-255

keywords

CHAR

VARCHAR

VARCHAR

description

VARCHAR(M)

变长字符串,M代表的是变长字符串的字节长度。M的范围是1-65533。

注意:变长字符串是以UTF-8编码存储的,因此通常英文字符占1个字节,中文字符占3个字节。

keywords

VARCHAR

STRING

STRING

description

STRING

变长字符串,最大(默认)支持1048576字节(1MB)。String类型的长度还受 be 配置 `string_type_length_soft_limit _bytes`,实际能存储的最大长度 取两者最小值,String类型只能用在value 列,不能用在 key 列和分区 分桶列

注意:变长字符串是以UTF-8编码存储的,因此通常英文字符占1个字节,中文字符占3个字节。

keywords

STRING

ARRAY

ARRAY

description

ARRAY<T> T支持的类型有: BOOLEAN, TINYINT, SMALLINT, INT, BIGINT, LARGEINT, FLOAT, DOUBLE, DECIMAL, DATE, DATETIME, CHAR, VARCHAR, STRING

example

ARRAY<INT> ARRAY<VARCHAR(1024)> ARRAY<STRING> ARRAY<DATE> ARRAY<DATETIME>

常见问题

产品问题

1数据中心

1.1 数据接入

Q1:外部三方导入只有列举的几个平台可选吗?

A:目前已支持巨量、热云等平台的接入,若您有其他平台的接入需求,可联系我们海纳嗨数的技术人员,我们将在后续为您进 行所需平台的对接。

1.2 元数据管理

Q1:目前不需要的事件暂时不想显示在日常分析的可选元事件中需要如何操作?

A:可让有「元数据管理权限」的用户直接在元事件列表将对应元事件的「是否可见」按钮关闭,该按钮关闭后不影响数据的正常入库。如果此事件被应用在已有空间中时,关闭后空间中对对应的分析将不可用。

Q2:事件已经创建好,为什么在分析模型的选择事件中找不到?

A:首先,请确认此事件是否有数据上报,没有数据上报的事件无法在分析中查看到。如已数据上报,请检查此事件是否可见, 如果不可见,请设置为可见状态。如果事件的是否可见状态为可见,但依旧无法在分析中使用,那么请在「分析」的选择事件 的下拉菜单的「事件分组设置」中,查看此事件在分组中是否被设置为可见。

Q3:谁可以将需要采集的事件与属性更新到项目中?

A:具有「元数据管理权限」的成员,可前往「元数据管理」的「元事件」、「事件属性」和「用户属性」中分别创建好需要采 集的信息。

1.3 监控预警

Q1:其他人创建的监控有异常我如何得到通知?

A:可让该监控创建人设置监控通知方式时选择邮件、短信等通知方式,并在通知成员中勾选需通知人。

Q2:新增监控时需把所有监控时段都勾选吗?

A:您只需勾选需要监控的时段,我们将会对所有时段的数据进行记录,但只对监控时段的指标进行异常判断。

2 数据空间

Q1:不同角色的用户在数据空间的权限有何不同?

A:普通用户仅可以查看公共空间的指标;分析师角色的用户可以配置我的空间的指标,同时可以查看、复制公共空间的指标; 管理员角色的用户不仅可以配置我的空间的指标,也可以配置公共空间的指标。

Q2:空间协作者有哪些空间权限?

A:若其他用户将您添加为TA创建的空间的协作者,那您将和TA共同协作TA创建的空间,包括空间指标的添加、删除和修改、选择全局时间范围、添加全局筛选条件、修改空间性能设置、查看共享设置、下载数据、导出PDF等权限。

Q3:如何复制公共空间中的指标?

A:若您想复制公共空间中的单个指标,您可以点击该指标进入分析模型中,点击保存即可保存到自己的书签中,在自己创建的 空间中添加该书签即可;若您想复制空间中所有指标,您可以直接复制空间到自己创建的分组下即可。

3数据分析

3.1 事件分析

Q1:分组过多时,表格数据显示不完整?

A:分组数量过多时,表格数据最多只展示前1000条数据,可通过页面的下载功能来获取完整数据。

Q2:按周查看时数据的时间范围是如何定义的?

A:以时间范围为2022-09-21至2022-09-29为例,按周查看时,图表展示2022-09-19当周和2022-09-26当周的数据,2022-09-19当周计算的数据范围为2022-09-21至2022-09-25的数据,2022-09-26当周计算的数据范围为2022-09-26至2022-09-29的数据。

3.2 漏斗分析

Q1:关联属性和分组项的区别?

A:关联属性是在漏斗内设置的属性,是根据设置的属性生成漏斗,而分组项是漏斗外设置的,基于已生成的漏斗,得到根据分 组项进行分组后的结果。

Q2:限制后续漏斗事件在时间区间内的作用?勾选/不勾选有何区别?

A:设置的时间区间默认为漏斗第一个步骤事件发生的时间范围,如果勾选"限制后续漏斗事件在时间区间内",那么漏斗各个步骤事件的发生时间在满足窗口期的同时,均被限定在所选时间区间内。比如,时间范围选择的是2022-09-19至2022-09-24,窗口期选择的是2天,用户在9月24日发生了漏斗的第一个步骤事件,在9月25日完成了漏斗的最后一个步骤事件,没有勾选该选择时,漏斗会把9月25日完成的最后一个步骤事件也算作一次该漏斗的转化,勾选了该选择后,则不会被计算在内。

3.3 属性分析

Q1:可否按照用户接入的时间顺序筛选用户?

A:用户属性中,默认情况下不会有时间属性,如果需要按接入时间查看,需要在埋点时,记录下时间类属性,然后依据该属性进行分析。

Q2:在筛选条件中选择标签时为什么没有标签版本的选择?

A:属性分析主要分析的是用户属性、分群以及标签,这些都记录的是当前的状态,因此在使用标签作为筛选条件时,将默认使用最新版本的标签,无需用户进行选择。

4 用户洞察

4.1 用户标签

Q1:创建好的标签数据,如何在分析中使用?

A: (1) 创建好的标签,我们可以在分析中用作筛选条件,如使用数值标签创建了标签「过去7天用户登录次数」,该标签统计 了过去7天每个用户的登录次数。在分析中,若只想对登录次数较高的用户进一步分析,就可以添加标签筛选条件,筛选登录次 数大于某数的用户。

(2) 创建好的标签,我们可用作分析中的分组项。如使用自定义标签创建了标签「用户价值分层」,该标签将用户划分为高价 值用户、中价值用户和低价值用户。在分析中,可将该标签设置为分组项,查看不同价值层级用户的行为。

Q2:自定义标签同一个用户会不会同时在多个分层中?

A:同一个用户(分析主体)不会同时存在多个分层中,用户(分析主体)将优先匹配至满足条件且顺序靠前的标签值中。如我们 设定分层1、分层2、分层3 三个分层,用户对于三个分层的规则都是满足的,那么该用户会按照分层规则的制定顺序,优先匹 配至靠前的分层中。

Q3:如何查看一个标签的历史数据?

A:标签若存在历史数据,会储存在标签详情页面的回溯版本管理中,可至此处进行查看/手动回溯历史数据。

4.2 用户分群

Q1:如何创建结果分群?

A:在数据分析/用户标签中,当有需要重点分析或关注的行为时,均可以通过点击数据表中的数据将满足条件的人群保存为一 个结果分群,保存后,结果分群的数据可在分群管理页面查看,结果分群不支持修改创建规则。

Q2:分群如何使用?

A:分群可在数据分析中用作筛选条件或分组项,并可以用来圈选目标投放人群或排除特殊人群,作为营销投放的人群包使用, 提高投放的转化率。

5 整体问题

Q1:怎么查看页面停留时长?

A:海纳云预置的 Web 页面浏览/App 页面浏览/小程序页面浏览事件,默认是不采集每个页面的浏览时长。如果想分析每个页面的浏览时长,客户需要自定义埋点采集每个页面浏览时长属性,这样计算的结果更精确,但是埋点比较耗时。

Q2:如何查看 App 使用时长?

A:海纳云采集事件 App 退出默认会采集本次访问 App 的使用时长(从 App 启动到 App 退出的访问时间间隔),可以通过查看 App 退出的事件时长均值查看访问 App 的使用时长均值。

数据&技术问题

1.1 数据分析/SQL查询/导出

Q1:导出CSV文件,用Excle打开,出现行列乱序问题?

A:导出的数据里面如果包含"\t"、"\n"、","等特殊转义字符,可在该字段上增加双引号("")或者使用JSON函数(json_quote())跳 过其中包含的特殊转义字符。

费用&服务问题

资源下载

1、Android SDK 资源

SDK类型	资源地址	版本号	发布日期
Android埋点SDK	HinaCloudSDK-Android-1.0.0.zip	1.0.0	2023-3-21

注意:针对gradle低版本(低于4.0)离线集成,插件仓库包下载地址:plugin_repo.zip

2、iOS SDK 资源

SDK类型	资源地址	版本号	发布日期
iOS埋点SDK	HinaCloudSDK-iOS-1.0.0.zip	1.0.0	2023-2-23

3、Web SDK 资源

SDK类型	资源地址	版本号	发布日期
Web埋点SDK	HinaCloudSDK-Web-1.0.0.zip	1.0.0	2023-2-23

4、Java SDK 资源

SDK类型	资源地址	版本号	发布日期
Java埋点SDK	HinaCloudSDK-Java-1.0.0.zip	1.0.0	2023-2-23

5、Python SDK 资源

SDK类型	资源地址	版本号	发布日期
Python埋点SDK	HinaCloudSDK-Python-1.0.0.zip	1.0.0	2023-2-23