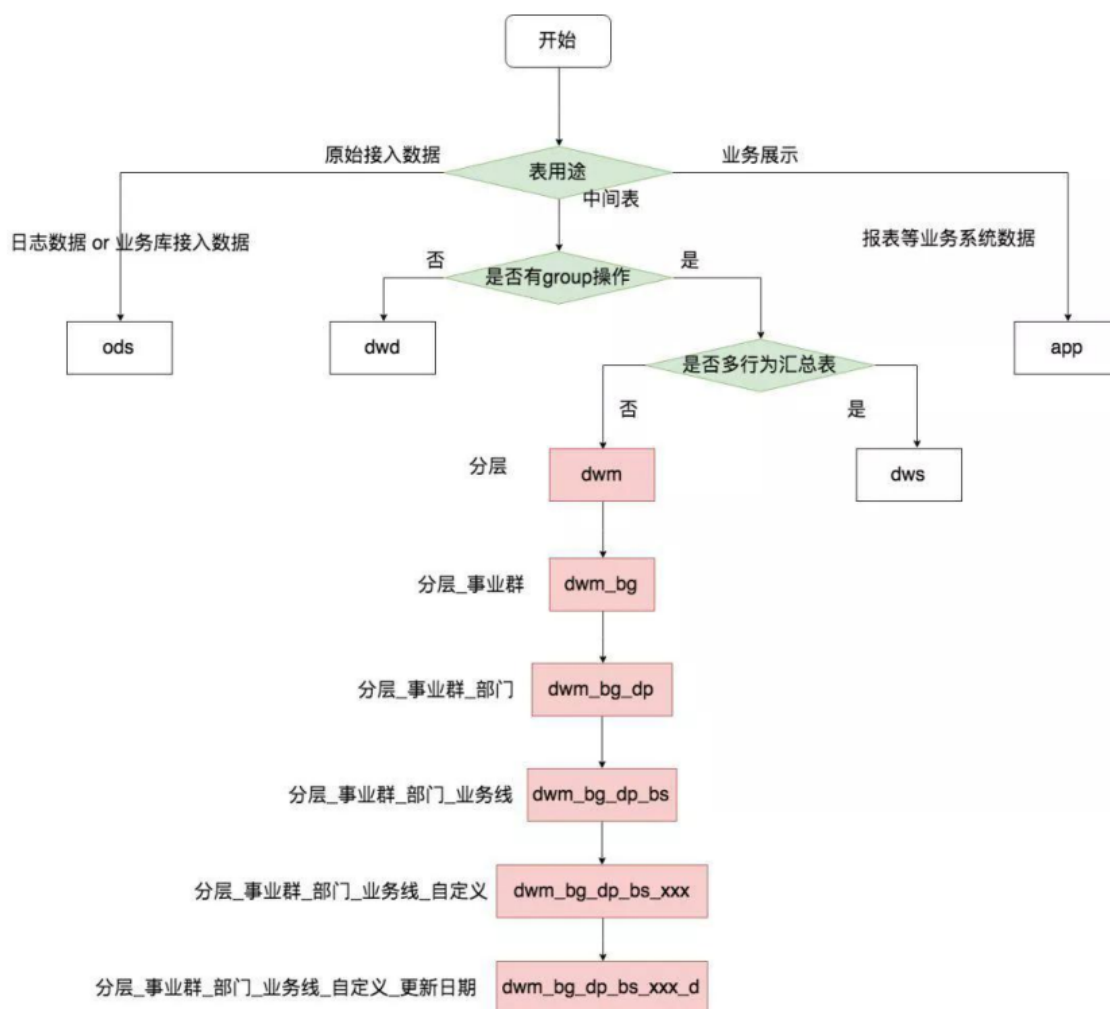


数仓命名规范

1. 词根设计规范

- 词根属于数仓建设中的规范，属于元数据管理的范畴，现在把这个划到数据治理的一部分。完整的数仓建设是包含数据治理的，只是现在谈到数仓偏向于数据建模，而谈到数据治理，更多的是关于数据规范、数据管理。
- 表命名，其实在很大程度上是对元数据描述的一种体现，表命名规范越完善，我们能从表名获取到的信息就越多。比如：一部分业务是关于货架的，英文名是：rack，rack 就是一个词根，那我们就在所有的表、字段等用到的地方都叫 rack，不要叫成别的什么。这就是词根的作用，用来统一一名，表达同一个含义。
- 指标体系中有很多“率”的指标，都可以拆解成 XXX+率，率可以叫 rate，那我们所有的指标都叫做 XXX+rate。
- 词根：可以用来统一表名、字段名、主题域名等等。

举例：以流程图的方式来展示，更加直观和易懂，本图侧重 dwm 层表的命名规范，其余命名是类似的道理：



第一个判断条件是表的用途，是中间表、原始日志还是业务展示用的表。如果该表被判断为中间表，就会走入下一个判断条件：表是否有 group 操作。通过是否有 group 操作来判断该表该划分在 dwd 层还是 dwm 和 dws 层。如果不是 dwd 层，则需要判断该表是否是多个行为的汇总表（即宽表）。最后再分别填上事业群、部门、业务线、自定义名称和更新频率等信息即可。

- 分层：表的使用范围

- 事业群和部门：生产该表或者该数据的团队
- 业务线：表明该数据是哪个产品或者业务线相关
- 主题域：分析问题的角度，对象实体
- 自定义：一般会尽可能多描述该表的信息，比如活跃表、留存表等
- 更新周期：比如说天级还是月级更新

数仓表的命名规范如下：

1) 数仓层次:

公用维度: dim

DM 层: dm

ODS 层: ods

DWD 层: dwd

DWS 层: dws

2) 周期/ 数据范围:

日快照: d

增量: i

全量: f

周: w

拉链表: l

非分区全量表: a

2. 表命名规范

1) 常规表

常规表是我们需要固化的表，是正式使用的表，是目前一段时间内需要去维护去完善的表。

规范：分层前缀[dwd|dws|ads]_部门_业务域_主题域XXX更新周期|数据范围业务域、主题域我们都可以用词根的方式枚举清楚，不断完善。更新周期主要的是时间粒度、日、月、年、周等。

2) 中间表

中间表一般出现在 Job 中，是 Job 中临时存储的中间数据的表，中间表的作用域只限于当前 Job 执行过程中，Job 一旦执行完成，该中间表的使命就完成了，是可以删除的（按照自己公司的场景自由选择，以前公司会保留几天的中间表数据，用来排查问题。

规范：mid_table_name_[0~9|dim]

table_name 是我们任务中目标表的名字，通常来说一个任务只有一个目标表。

这里加上表名，是为了防止自由发挥的时候表名冲突，而末尾大家可以选择自由发挥，起一些有意义的名字，或者简单粗暴，使用数字代替，各有优劣吧，谨慎选择。

通常会遇到需要补全维度的表，这里使用 dim 结尾。

如果要保留历史的中间表，可以加上日期或者时间戳。

3) 临时表

临时表是临时测试的表，是临时使用一次的表，就是暂时保存下数据看看，后续一般不再使用的表，是可以随时删除的表。

规范：tmp_xxx

只要加上 tmp 开头即可，其他名字随意，注意 tmp 开头的表不要用来实际使用，只是测试验证而已。

4) 维度表

维度表是基于底层数据，抽象出来的描述类的表。维度表可以自动从底层表抽象出来，也可以手工来维护。

规范：dim_xxx

维度表，统一以 dim 开头，后面加上，对该指标的描述。

5) 手工表

手工表是手工维护的表，手工初始化一次之后，一般不会自动改变，后面变更，也是手工来维护。

一般来说，手工的数据粒度是偏细的，所以暂时统一放在 dwl 层，后面如果有目标值或者其他类型手工数据，再根据实际情况分层。

规范：dwl_业务域_manual_xxx

手工表，增加特殊的主题域，manual，表示手工维护表。

3. 指标命名规范

公共规则

- 所有单词小写
- 单词之间下划线分割（反例：appName 或 AppName）
- 可读性优于长度（词根，避免出现同一个指标，命名一致性）
- 禁止使用 sql 关键字，如字段名与关键字冲突时 +col数量字段后缀 cnt 等标识...
- 金额字段后缀 price 标识
- 天分区使用字段 dt，格式统一（yyyymmdd 或 yyyy-mm-dd）
- 小时分区使用字段 hh，范围（00-23）
- 分钟分区使用字段 mi，范围（00-59）
- 布尔类型标识：is_{业务}，不允许出现空值