## 数据结构

Set rewrite=on;

1. Select Day,Game,sum(amount)

from Fact f /\*fact\*/

where f.dt between 20160101 and 20160110 and game\_type=’WEB’

1. Select td.month, gd.game,sum(amount)

from Fact f /\*fact\*/

join\_time\_dim as td on f.dt=td.dt

join game\_dim as gd on f.server=gd.server

where td.dt between 20160101 and 20160110

group by td.month,gd.game

1. FactTable:Fact

Project:

Field{name:Day,source:fact}

Field{name: Game,source:fact}

AggExp{name: amount,source:fact,aggFunc:sum}

Filter:

And:

Between:{Field{name:dt,source:fact},start:20160101,end:20160110}

Equal:{Field{name:game\_type,value=WEB}}

1. FactTable:Fact

Project:

Field{name:month,source:time\_dim}

Field{name: Game,source:game\_dim}

AggExp{name: amount,source:fact,aggFunc:sum}

Filter:

And:

RelationExist:

Fields:[

]

Source:time\_dim

Filter

Between:{Field{name:dt,source:fact},start:20160101,end:20160110}

RelationExist:

Fields:[

]

Source:time\_dim

Filter

Between:{Field{name:dt,source:fact},start:20160101,end:20160110}

## 如何判定两个逻辑表达式之间是否具有包含关系：

目标：TA->TB两个表达式，需要判定TB是否包含TA

抽象出And，Or两个逻辑运算符。And/Or决定当判定一个子语句包含关系为true时，是否还继续往下执行。

抽象出=,>,<,>=,between,exist等函数，用EXP表示

先递归左侧TA表达式，后递归右侧TB表达式：

TA1->TB1 or TB2 or TB3:

只要TB1包含TA1，继续往下执行，TB2和TB3只会将右侧继续放大，所以此时说明右侧包含左侧，后续逻辑不用继续，直接返回true；如果是未知或false，则继续执行，看后面是否有true的返回，如果有，则返回true，反之返回false

TA1->TB1 and TB2 and T3:

只要TB1不包含TA1，继续往下执行，TB2和TB3只会将右侧继续缩小，说明右侧必然不包含左侧，后续逻辑不用继续，直接返回false；如果为未知，也返回false；如果为true，则继续执行，看后续是否还有false或未知状态出现，如果有，则返回false，反之，返回true

TA1 and TA2->TB:

只要TB1包含了TA1，那么TA1 and TA2只会使左侧更小，所以最终结果包含关系为true。因此遇到TB1包含了TA1，则不用再继续，直接返回true。如果为未知和false，则继续，看后面是否有含有true的结果，如果有，则最终依然为true

TA1 or TA2->TB:

如何判定TA1->TB1:

如果是相同字段，根据EXP逻辑计算；如果不是同一个字段或是字段上的函数调用(防止函数每次执行结果不同，如now函数,故不支持函数重写)，包含关系为“未知”；

EXP对应的实现：需要根据字段类型，从而确定大小关系如何比较，如字符串，日期，数值  
=：

>=:

<=:

between:

exist:用于描述表关联，如日期维表的关联，join dim\_dt dim on f.dt=dim.dt where dim.month=201605,转换为逻辑表达式为

dt in (select dt from dim\_dt dim where dim.month=201605),因此exist的比较转换subquery之间的比较，从而递归复用上述判定逻辑

改写逻辑的设计：

针对事实表的改写

针对维表关联的改写