# 高级功能逻辑

## OSPF

### 1：根据交换机品牌、型号、版本、子版本查询OSPF命令

#### FunctionalMethods.getFuzzySQL

拼接 SQL语句 实现根据 品牌、型号、版本、子版本 模糊查询。

#### FunctionalMethods.getEquivalence

##### 实现等价查询：

Huawei等价 Quidway

### 2：通过四项基本信息的精确度 筛选最精确的OSPF命令

#### ScreeningMethod.ObtainPreciseEntityClassesOspfCommand

### 3：执行OSPF命令，获取交换机返回信息

### 4：提取OSPF数据 getOSPFPojo

#### :获取配置文件中定义的关于端口号特征的关键词数组

#### :遍历关于端口号特征的关键词数组，保存交换机返回信息中存在的特征关键词到特征关键词集合。如果特征关键词集合为空，则返回空集合并告警。

#### :交换机返回信息按行分割得到交换机行信息数组，遍历 行信息数组 和 特征关键词集合,将端口号存在空格的情况去除空格。

例如： GigabitEthernet 1/0/0 替换成 GigabitEthernet1/0/0

#### :遍历交换机行信息数组，获取字符串中的IP集合。如果IP集合长度为2，且元素包含full 则为连接正常的OSPF数据。如果未获取到连接正常OSPF数据，则返回空集合并告警。

#### : \* 将 连接正常的OSPF数据 按空格分组为 列数组。遍历列数组，获取 IP特征（包括：IP 和 无效IP：0.0.0.）、端口号、FULL状态 的在列数组的下标位置，其他列未 未知列。（如果两个IP都为正常IP应该怎样取值？当时记得是随便取一个。）

**判断是否是端口号的方法逻辑是：**

**如果参数传入端口号特征关键字集合，则判断是否包含端口号特征关键字及数字，如果没有传入端口号特征关键字集合，则判断是否同时包含字母与数字。是则为端口号。**

#### :遍历交换机行信息数组，将所有元素都按空格分组为 列数组，并判断该数组长度是否与 连接正常的OSPF数据分割得到的数组长度相等。如果相等，暂时保存到 临时集合 中，

#### :遍历临时集合，判断IP特征、端口号下标对应的元素，是否为IP或端口号。判断未知列是否不为IP特征、端口号、FULL状态。如果都满足，则是OPSF数据，则按照对应元素提取值 存入OSPF集合。（不能判断元素是否为FULL，是因为FULL是正常状态，也可能不为FULL，连接异常）

#### :返回OSPF集合

### 5:遍历OSPF集合，如果状态为FULL为安全，不为FULL则异常。

## 光衰

### 1：根据交换机品牌、型号、版本、子版本查询光衰命令

#### FunctionalMethods.getFuzzySQL

拼接 SQL语句 实现根据 品牌、型号、版本、子版本 模糊查询。

#### FunctionalMethods.getEquivalence

##### 实现等价查询：

Huawei等价 Quidway

### 2：通过四项基本信息的精确度 筛选最精确的光衰命令

#### ScreeningMethod.ObtainPreciseEntityClassesLightAttenuationCommand

### 3：执行获取端口号命令，获取交换机返回信息

### 4:\*将交换机返回信息按行分割，获得行信息数组。遍历交换机返回行信息获取包含 **" UP " (状态)**且不包含铜缆 "COPPER"(写在了程序中)的元素，存储入端口号数据集合。

### 5:遍历端口号数据集合，获取端口号。

1. **:获取配置文件中定义的关于端口号特征的关键词数组。**
2. **:遍历端口号特征的关键词数组，判断端口号数据是否包含关键词**
3. **:端口号数据根据“ ”空格分割为数组**
4. **:遍历 端口号数据分割数组，判断是否以关键词开始，得到端口号。判断端口号是否包含”.” ，如果包含”.”怎说明是子端口号，结果不返回子端口号。**

### **6:**获取端口号集合，**判断**端口号**是否为"Eth" 百兆。如果为百兆则去除。**

### 7:获取数据表转义信息。如：GE 转译为 GigabitEthernet

### 8:数据表获取光衰数据命令。遍历端口号集合，替换命令中的端口号、执行命令、获取交换机返回光衰数据信息。

### 9:提取光衰数据

#### :交换机返回光衰数据信息按行分割为行数组

#### :遍历行数组，判断行数据中 是否（包含 "TX POWER"） 或(包含"RX POWER"） 或 （包含 "TX" 、"RX" 之一 且包含 "DBM"）

将"TX POWER" 替换成 "TXPOWER" 将"RX POWER" 替换成 "RXPOWER"

#### :提取TX 和 RX的值

###### 如果包含 则 判断 TX 和 RX在行信息中的位置。

* 1. 如果全部为 -1 则不包含TX和RX 则结束当前循环，遍历下一条行数据。（不存在）
  2. 如果全部不为 -1 则说明 TX 和 RX 在同一行。

**"Temp(Celsius) Voltage(V) Bias(mA) RX power(dBm) TX power(dBm)\r\n" +**

**"37(OK) 3.36(OK) 15.91(OK) -5.96(OK)[AP] -6.04(OK)";**

* + 1. 如果在同一行，则按空格分割为属性名数组
    2. 取下一行数据，获取字符串中的Double值按空格分割为属性值数组
    3. 如果属性名数组长度 与 属性值数组长度一致，则取包含RX 和 TX在属性名数组中的下标。后根据属性值数组下标取值。

**有没有 RX power(dBm) : -5.96(OK)[AP] TX power(dBm) : -6.04(OK) 情况存在**

* 1. 如果TX和 RX的位置值，有一个为-1 则说明 两个参数不在同一行

第一种情况

"Current Rx Power(dBM) :-11.87\r\n" +

"Default Rx Power High Threshold(dBM) :-2.00\r\n" +

"Default Rx Power Low Threshold(dBM) :-23.98\r\n" +

"Current Tx Power(dBM) :-2.80\r\n" +

"Default Tx Power High Threshold(dBM) :1.00\r\n" +

"Default Tx Power Low Threshold(dBM) :-6.00";

第二种情况

"Rx Power: -6.0dBm, Warning range: [-16.989, -5.999]dBm\r\n" +

"Tx Power: -6.20dBm, Warning range: [-9.500, -2.999]dBm";\*/

* + 1. 判断是否包含 "CURRENT" 当前，存在则保存。如果数量大于1 ，则只保留包含 "CURRENT" 的数据。

"Current Rx Power(dBM) :-11.87\r\n" +

"Current Tx Power(dBM) :-2.80\r\n"

* + 1. 遍历集合，因为上面逻辑已经 将"TX POWER" 替换成 "TXPOWER" 将"RX POWER" 替换成 "RXPOWER"。所以只需提取"TXPOWER" 或"RXPOWER"后的第一个参数就可以了。
  1. 返回TX RX 数据。

### 10:根据交换机IP地址，查询光衰数据记录。

### 11:判断历史记录和新查询到的端口号，如果新端口号集合中少了端口号则将少的端口号状态置为"DOWN"，并前端告警。

### 12:将数据库光衰数据记录和新查询到的光衰数据做对照，判断有无异常，插入数据库。

1. : |最新参数 - 起始值| > 额定偏差
2. : |最新参数 - 平均值| > 即时偏差

## 错误包

### 1：根据交换机品牌、型号、版本、子版本查询错误包命令

#### FunctionalMethods.getFuzzySQL

拼接 SQL语句 实现根据 品牌、型号、版本、子版本 模糊查询。

#### FunctionalMethods.getEquivalence

##### 实现等价查询：

Huawei等价 Quidway

### 2：通过四项基本信息的精确度 筛选最精确的错误包命令

#### ScreeningMethod.ObtainPreciseEntityClassesErrorRateCommand

### 3：执行获取端口号命令，获取交换机返回信息

### 4：根据交换机返回信息获取错误包端口号

逻辑大致与光衰获取端口号数据逻辑相同。差异是光衰功能不能为"Eth"百兆。错误包可以是百兆。

### 5:获取数据表转义信息。如：GE 转译为 GigabitEthernet

### 6:数据表获取错误包数据命令。遍历端口号集合，替换命令中的端口号、执行命令、获取交换机返回错误包数据信息。

### 7:提取错误包数据

#### :获取配置文件定义的关键词

* 1. :获取配置文件中关于错误包数据关键词的定义，获得品牌为key的集合。
  2. :获取定义的品牌名，当品牌为HuaWei时，则查询是否有Quidway。
  3. :根据品牌，再查询更详细的四项基本信息。
  4. :根据精确的基本信息，获取配置文件定义的关键词

#### :根据配置文件定义的关键词提取数据

* 1. 根据关键词中的占位符，获取参数值。
     1. 'Total Error: $' 与 ‘Total Error: 50’
     2. 'Input: $ input errors' 与 Input: 0 input errors
  2. 特殊情况：当关键词是Total Error 时，应该先查询上文Input 和 Output那个最近。该Total Error的值就是Input 或 Output的值。取值方式与其他一样，都是使用了占位符的方式。
  3. 描述：Description
     1. 获取配置文件关于交换机对端口线路描述的关键字。
     2. 截取行信息数据关键字的后面部分，如果以”:”开始，则去除开始”:”。
  4. 返回错误包参数和描述信息。

### 8:根据交换机IP地址，查询错误包数据记录。

### 9:判断历史记录和新查询到的端口号，如果新端口号集合中少了端口号则将少的端口号状态置为"DOWN"，并前端告警。

### 10:将数据库错误包数据记录和新查询到的错误包数据做对照。根据错误包数量是否有变化，判断有无异常，插入数据库。

**提出新逻辑，判断变化小，一般告警。 超过阈值，告警。**

## 路由聚合

### 内部路由聚合

#### 1：根据交换机品牌、型号、版本、子版本查询聚合命令

##### FunctionalMethods.getFuzzySQL

拼接 SQL语句 实现根据 品牌、型号、版本、子版本 模糊查询。

##### FunctionalMethods.getEquivalence

##### 实现等价查询：

Huawei等价 Quidway

#### 2：通过四项基本信息的精确度 筛选最精确的聚合命令

##### ScreeningMethod.ObtainPreciseEntityClassesOspfCommand

#### 3：执行内部聚合命令，获取交换机返回信息

#### 4：根据交换机返回结果，获得路由地址段

1)：根据交换机返回信息遍历交换机返回数据，提取某行数据中包含内部路由地址特征关键字的且包含两个IP特征的数据，并提取两个IP特征数据，子网掩码可能是反掩码，需要注意。

2)：根据路由地址的ip和子网掩码，获得ip地址的CIDR格式，并实现通过IP地址计算器，获得路由地址的地址段的范围功能。

3)：将根据交换机返回结果的所有IP CIDR地址，获取对应地址段集合。

4)：将地址段集合可以连续的或包含的地址段拼接起来。获得一些更长的地址段

#### 5: 根据地址段，拆分成IP CIDR地址。

1. :遍历地址段集合，对逐个地址段进行操作。
2. :获取A地址段的IP地址数，根据IP地址数，获得最小掩码数。
3. :根据A地址段的第一个IP 和 掩码数，获得对应这个IP/掩码数的B地址段。
4. :判断这个B地址段的首地址 与 A地址段首地址是否相同。如果不是，则掩码数增一，重复3)、4) 步，直到B地址段的首地址 与 A地址段首地址相同。
5. 此时就获得了新的ip CIDR格式数据。A地址段删去B地址段，查看A地址段是否还有数据。如果还有一个IP，则直接该IP与 掩码数32合并。
6. :如果A地址段的IP数量大于1，则重复2)、3)、4)、5)、6) 步骤。直到地址段所有地址都转化为IP CIDR数据。

#### 6：内部地址聚合完成，返回数据。

### 外部聚合

#### 1：执行内部聚合命令，获取交换机返回信息

#### 2：根据交换机返回信息，获取协议、CIDR、下一跳IP、优先级、端口号信息

1. :遍历交换机返回信息行数据
2. :获取 行数据满足的端口号 和 满足IP、CIDR的数据集合
3. :判断端口号是否不为null 并且 IP、CIDR的数据集合长度是否不为0
4. :IP、CIDR的数据集合，包含”/”的是CIDR 否则为IP。如果集合不包含CIDR，则使用之前获得CIDR数据。
5. :根据数据库表设置的关于协议的关键字，获取数据中的协议。
6. :优先级和Cost值获取
   1. 如果行数据中包含中括号，则获取中括号中的数据为 优先级和Cost值。
   2. 如果行数据中不包含中括号，则获取行数据中的所有纯数字，如果个数为2，则默认第一个为优先级，第二个为Cost。如果个数不为2，则提取失败。
7. :返回数据集合。

#### 3：依次按照协议、优先级、Cost值、下一跳IP分组。获取一个组新的数据集合。

#### 4：每一组数据集合，分析CIDR数据：按照内部路由聚合方式，获取对应地址段。如果地址段有连续或包含，则获取更大的地址段，然后解成