# 数据解析文档

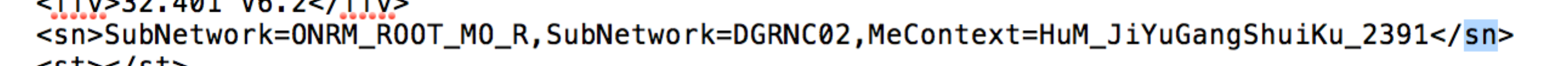
## WCDMA

### 爱立信性能

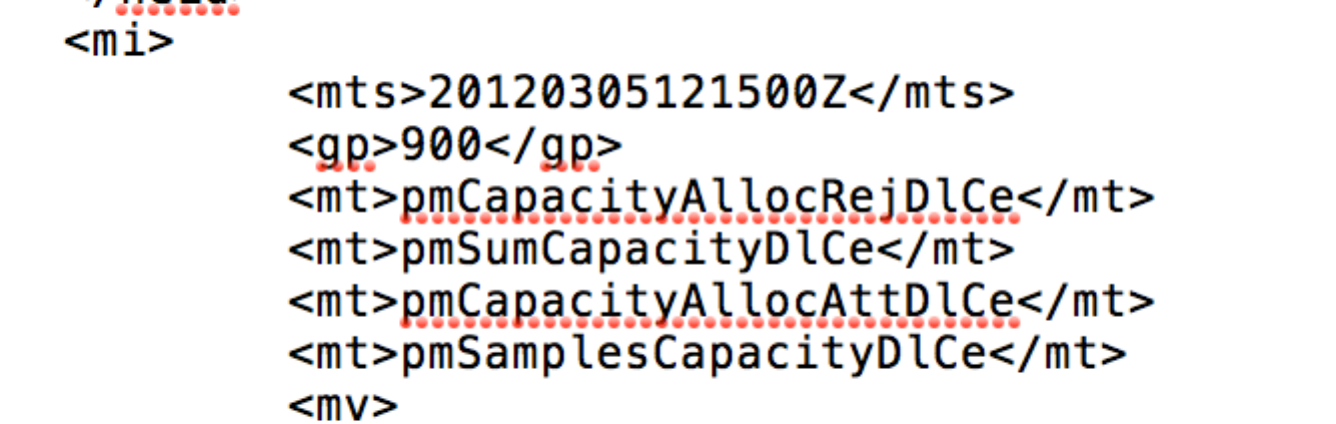
爱立信性能为XML格式的文件。文件有两种，RNC级与NODEB级，文件格式相同。

有一个配置表，表名为clt\_pm\_w\_eric\_map，用到三个列，col\_name、short\_col\_name、tab\_name，分别表示原始字段名、列名、表名，用于在解析时确定一个文件中的字段应该入库到哪个采集表的哪个列。

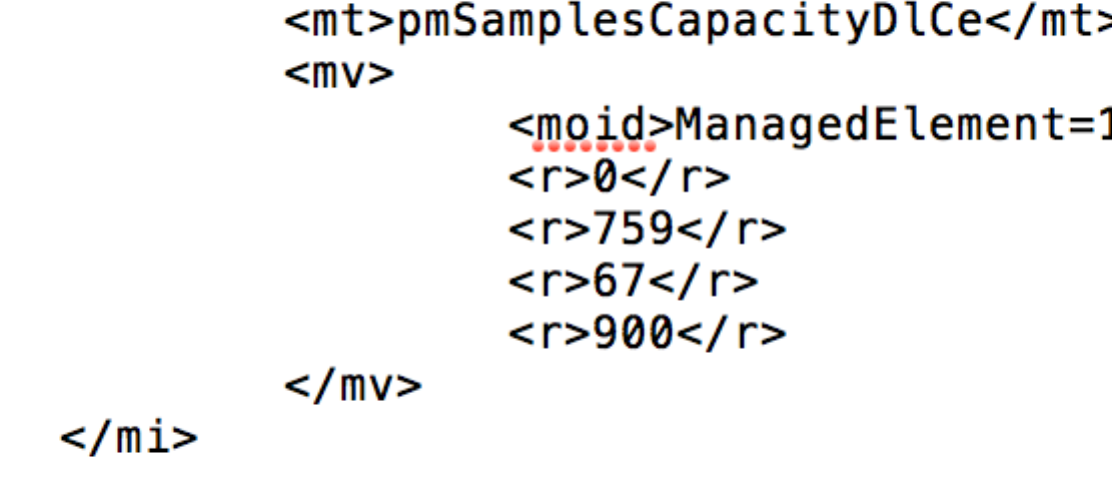
首先解析sn标签中的内容，一个文件中只有一个sn标签。它有两个SubNetwork属性和一个MeContext属性。将它们解析出来，作为一个文件的公共字段。有四个公共字段，SUBNETWORKROOT、SUBNETWORK、MECONTEXT、RNCNAME。SUBNETWORKROOT用第一个SubNetwork属性的值，SUBNETWORK和RNCNAME共用第二个SubNetwork属性的值，MECONTEXT用MeContext属性的值。



每个mi标签下的数据，为一类数据，也就是对应一个采集表，注意，有可能有多个mi标签都是同一类数据。mi下有一些mt标签，它是字段名。

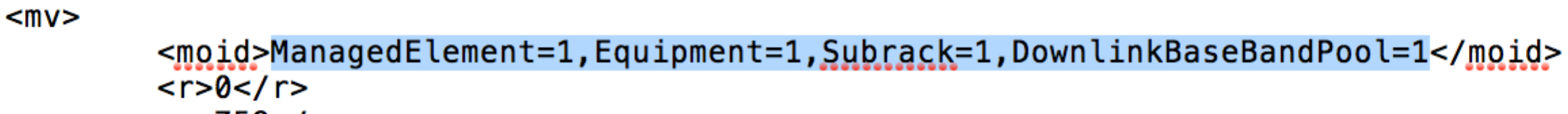


mi下有一个或多个mv标签，每个mv标签表示一条记录，mv下有一些r标签，表示字段的值，r标签的个数与mt标签个数相同，分别表示字段名和字段值。



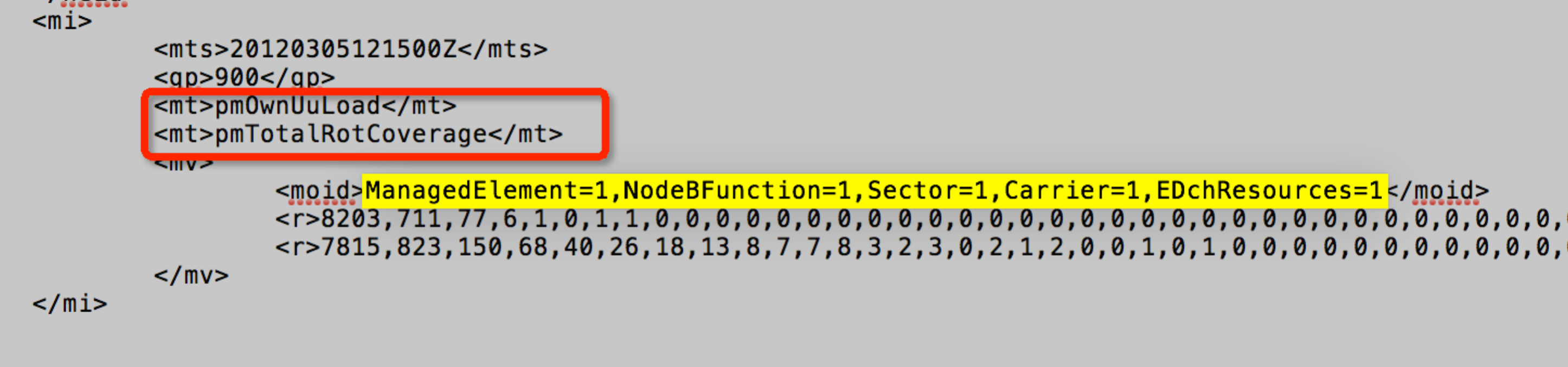
如何知道入到哪个采集表？mv标签下有一个moid标签，里面有一串键值对内容，最后一对键值对的键，表示表名。比如下图中表示是DownlinkBaseBandPool表，按“CLT\_PM\_W\_ERIC\_”加键名的方式确定表名，比图下图是CLT\_PM\_W\_ERIC\_DOWNLINKBASEBANDPOOL表。如果长度超过30，那么只保留30个字段，实际上就是CLT\_PM\_W\_ERIC\_DOWNLINKBASEBAND。另外，经过上述算法得到的表名如果是CLT\_PM\_W\_ERIC\_VCLTP，要做一个特殊处理，如果当前解析的文件是NODEB级的，那么表名改为CLT\_PM\_W\_ERIC\_NODEBVCLTP，判断NODEB级的依据是：sn标签中的第二个SubNetwork的值和MeContext的值不相同时，是NODEB级，否则，是RNC级。

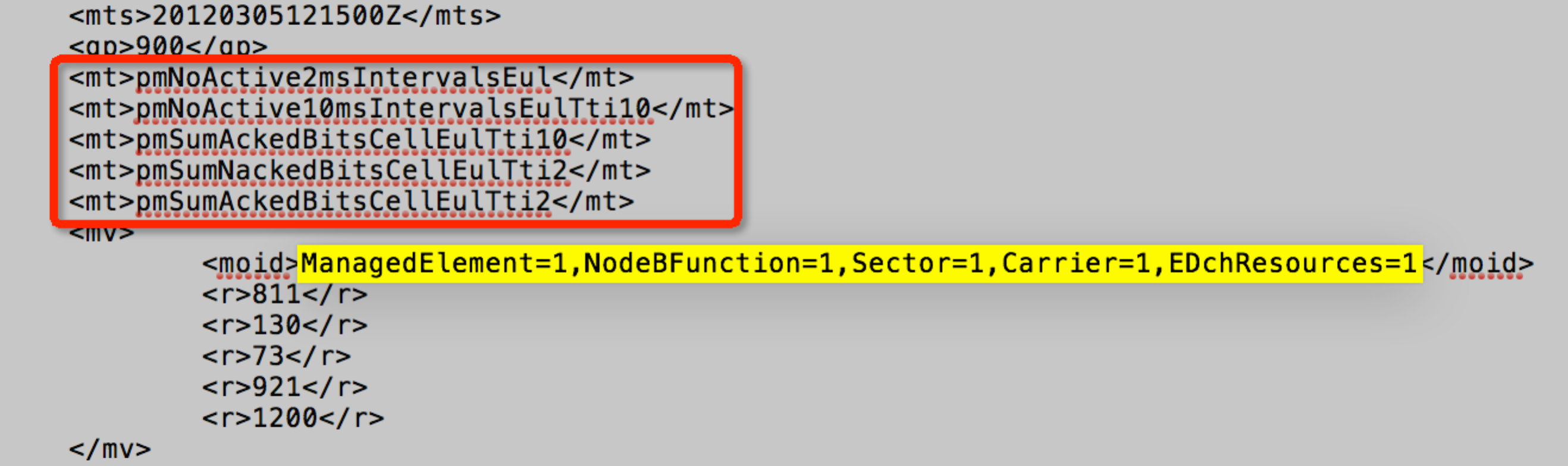
moid中这些键值对，也要作为一条记录中所有字段的一部分。也就是说，采集表中的一条记录，除了sn标签中的公共字段和mt标签定义的字段，还有moid中的几个字段。



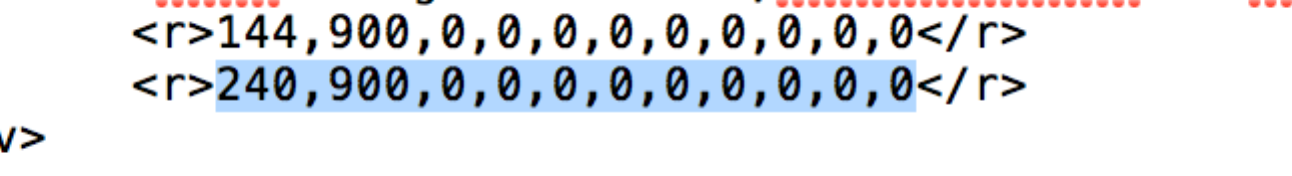
现在，已经知道有哪些字段，和表名了，但是，字段名并不能直接当作表中的列来用，有些太长了，有些有特殊符号，这时就需要在clt\_pm\_w\_eric\_map表中找对应关系，能找到的就入库，找不到的就放弃。

爱立信性能中，有一个特殊点，在整个文件中，可能存在moid完全相同的两条或多条记录，它们的mt字段名通常不一样，这时，要把这们合并为一条记录。也有moid完全相同，mt字段名，包括r标签中的值也完全一样的，这时只保留一份就行了。





在爱立信性能中，还要处理PDF计数器，所谓PDF计数器，就是一串用逗号连接在一起的数字，比如下图这个字段。



对某些PDF计数器，要进行一些公式计算，将一串数据转化为一个数字，具体算法请参考DBFunction类，和它的getMaxAvgCounter方法的注释。有几类算法，每类算法都要计算出两个值，平均值和最大值。哪些字段用什么算法，配置在clt\_pm\_w\_eric\_cal\_cfg表中，表的描述如下：

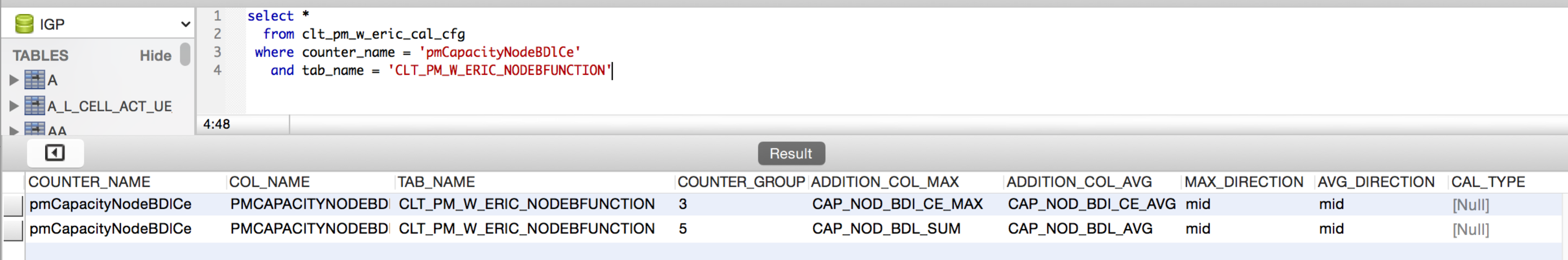
|  |  |
| --- | --- |
| 列 | 描述 |
| counter\_name | 需要计算的字段的名字（原始文件中的字段名）。 |
| col\_name | 需要计算的字段的列名（采集表中的列名）。 |
| tab\_name | 需要计算的字段，所在采集表的表名。 |
| counter\_group | 字段的计算类型，即使用哪类算法。目前有6种算法，取值分别是1到6. |
| addition\_col\_max | 算法计算出的最大值，放在哪个列（tab\_name字段所填写的表中）。 |
| addition\_col\_avg | 算法计算出的平均值，放在哪个列（tab\_name字段所填写的表中）。 |
| max\_direction | 计算最大值时，使用的计算常量，使用哪个方向的取值，有left、mid、right三个取值，即左、中、右值，对应三个常量，常量的值在clt\_pm\_w\_eric\_cal\_vals表中取。 |
| avg\_direction | 计算平均值时，使用的计算常量，使用哪个方向的取值，有left、mid、right三个取值，即左、中、右值，对应三个常量，常量的值在clt\_pm\_w\_eric\_cal\_vals表中取。 |

所谓的计算常量，就是一个浮点数。计算PDF计算器时，无论哪类算法，大概过程都是这样的：将原始内容用逗号分隔，分隔后的每一个数字，与计算常量做一些公式运算，然后每个数字的运算结果再在一起做些公式运算，最后得到最大值和平均值。但分隔后的每个数字在运算时使用的常量是不同的，在clt\_pm\_w\_eric\_cal\_vals表中有定义，它定义了分隔后每一位数使用的常量是多少，注意，每类算法要计算的PDF计数器，逗号分隔后的个数是一样的，如果不一样，就认为是文件错误。clt\_pm\_w\_eric\_cal\_vals表配置的常量又分左、中、右值，至于用哪个值，是前面描述的clt\_pm\_w\_eric\_cal\_cfg表中配置的。

下面是clt\_pm\_w\_eric\_cal\_vals表的描述（没描述的字段是没用到的，不用关注）：

|  |  |
| --- | --- |
| 列 | 描述 |
| counter\_group | 算法类型，取舍为1~6，对应clt\_pm\_w\_eric\_cal\_cfg表中的counter\_group。 |
| counter\_index | 把原始内容用逗号分隔后，每个数字的索引位置，注意，索引从1开始。 |
| left\_val | 左值。 |
| mid\_val | 中值。 |
| right\_val | 右值。 |
| raw\_val | 未指定左、中、右值时使用的默认值。 |

下面举例说明一个PDF计数器的计算过程：在解析时，遇到了一个名为pmCapacityNodeBUlCe的字段，所在采集表为CLT\_PM\_W\_ERIC\_NODEBFUNCTION，这时，在clt\_pm\_w\_eric\_cal\_cfg表中找，发现存在两条记录。



我们发现两条记录的counter\_group分别是3和5，所以说pmCapacityNodeBUlCe这个字段，要用两种算法都算一次，这里只说一条的计算过程。第一条的内容如下：

|  |  |
| --- | --- |
| 列 | 值 |
| counter\_name | pmCapacityNodeBDlCe |
| col\_name | PMCAPACITYNODEBDLCE |
| tab\_name | CLT\_PM\_W\_ERIC\_NODEBFUNCTION |
| counter\_group | 3 |
| addition\_col\_max | CAP\_NOD\_BDI\_CE\_MAX |
| addition\_col\_avg | CAP\_NOD\_BDI\_CE\_AVG |
| max\_direction | mid |
| avg\_direction | mid |

假设pmCapacityNodeBUlCe的原始内容为“240,900,0,0,0,0,0,0,0,0,0”。用逗号分隔后，为11个。算法3计算最大值的方法是从右到左找到第一个不为0的数，返回其计算常量。从右到左第一个不为0的是900，它是第2个数，并且刚才在clt\_pm\_w\_eric\_cal\_cfg表中查到计算最大值的方向是mid，那么常量取值是多少，就要到clt\_pm\_w\_eric\_cal\_vals表中取，用SQL语句表示如下：

select mid\_val From clt\_pm\_w\_eric\_cal\_vals where counter\_group=3 and counter\_index=2

条件“counter\_group=3”，是因为刚才在clt\_pm\_w\_eric\_cal\_cfg表中查到的算法类型为3.

条件“counter\_index=2”，是因为刚才从右到左找到的那个数，是第2位。

然后因为刚才在clt\_pm\_w\_eric\_cal\_cfg表中查到计算最大值的方向是mid，所以“select mid\_val”。

那么，最后得到的最大值，就是上面SQL语句查出来的那个值。然后这个值，应该放在CLT\_PM\_W\_ERIC\_NODEBFUNCTION表的 CAP\_NOD\_BDI\_CE\_MAX列，因为刚才在clt\_pm\_w\_eric\_cal\_cfg表中查到tab\_name为CLT\_PM\_W\_ERIC\_NODEBFUNCTION，addition\_col\_max为CAP\_NOD\_BDI\_CE\_MAX。

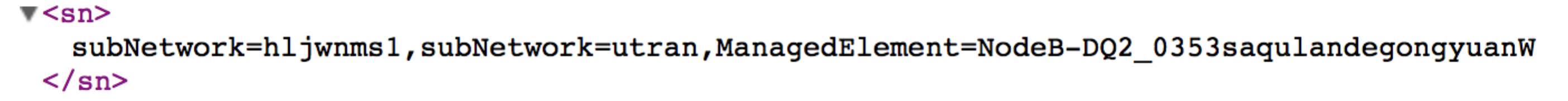
### 阿朗性能

阿朗性能为XML格式的文件。文件有三种，分别是RNC级、NODEB及、与INODE级，文件格式相同。

有一个配置表，表名为clt\_pm\_w\_al\_map，它只有三个列，分别是表名、列名、原始字段名，用来在解析入库时确定文件中的一个原始字段名对应哪个采集表的哪个列，并且，只有在clt\_pm\_w\_al\_map表中配置的采集表和字段需要入库，没有配置的表或列的在解析时跳过。

首先解析文件中的SN标签，一个文件只有一个SN标签，它有两个subNetwork和一个ManagedElement属性，它们是整个文件在入库采集表时的公共字段，也就是这个文件解析入库的表中，每条记录都有这三个字段。

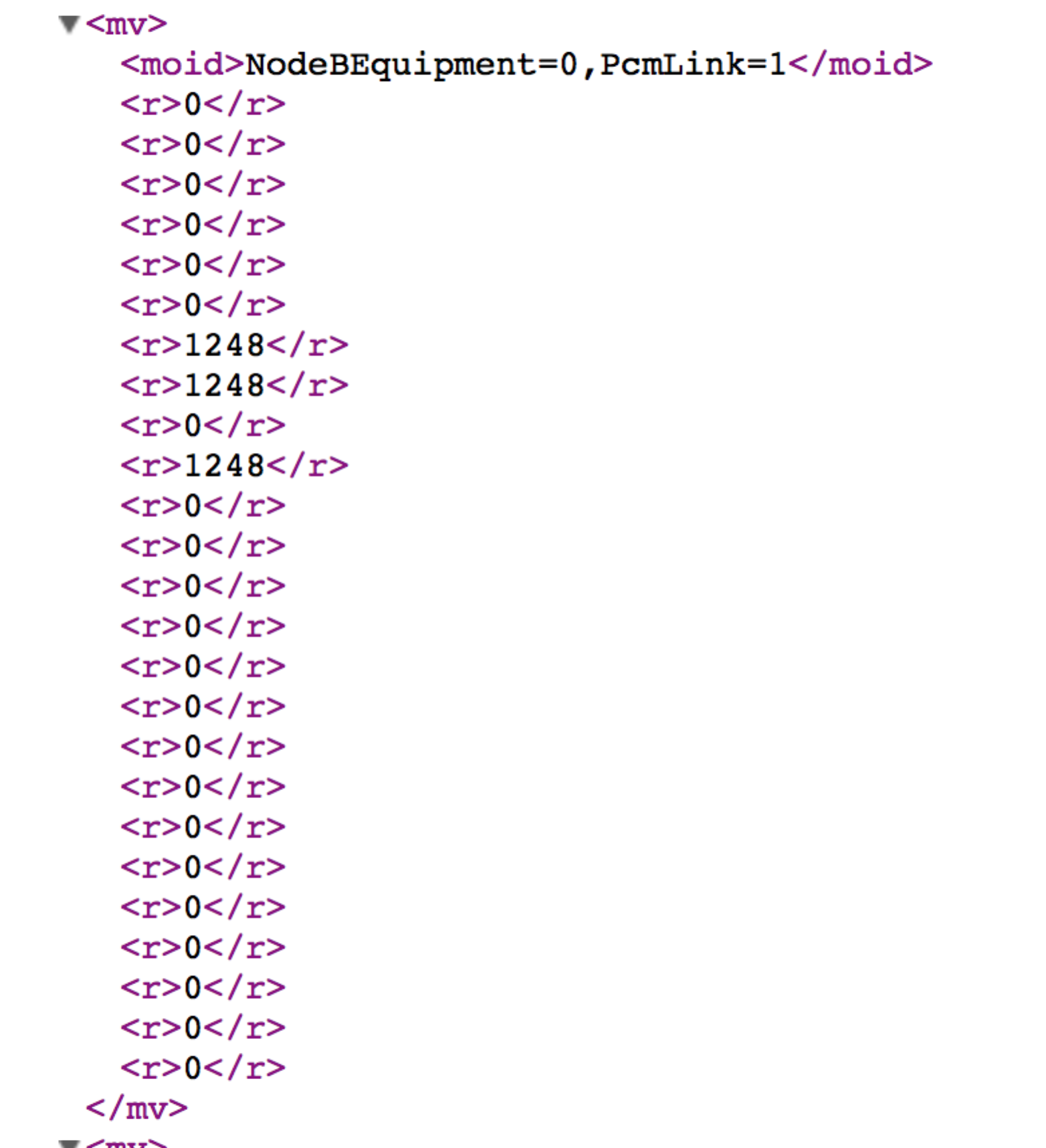
第一个subNetwork对应RNC\_NAME字段，第二个subNetwork对应SUBNETWORK字段，ManagedElement对应MANAGEDELEMENT字段。



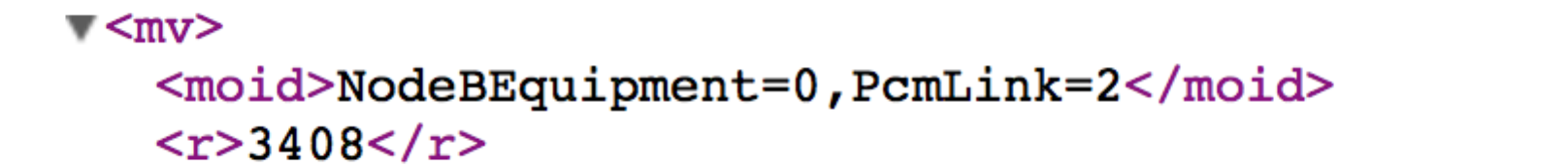
接下来是普通字段的解析。每个mi标签中，保存一类数据，也就是对应一个采集表。mi下有一些mt标签，它是字段名。



mi下有一个或多个mv标签，每个mv标签表示一条记录，mv下有一些r标签，表示字段的值，r标签的个数与mt标签个数相同，分别表示字段名和字段值。



如何知道入到哪个采集表？mv标签下有一个moid标签，里面有一串键值对内容，最后一对键值对的键，表示表名。比如下图中表示是PcmLink表，按“CLT\_PM\_W\_AL\_”加键名的方式确定表名，比图下图是CLT\_PM\_W\_AL\_PCMLINK表。moid中这些键值对，也要作为一条记录中所有字段的一部分。也就是说，采集表中的一条记录，除了sn标签中的公共字段和mt标签定义的字段，还有moid中的几个字段。



现在，已经知道有哪些字段，和表名了，但是，字段名并不能直接当作表中的列来用，有些太长了，有些有特殊符号，这时就需要在clt\_pm\_w\_al\_map表中找对应关系，能找到的就入库，找不到的就放弃。

RNC级文件中，有一个ULTRACELL表，它的字段特别多，所以拆分成了CLT\_PM\_W\_AL\_UTRANCELL1~ CLT\_PM\_W\_AL\_UTRANCELL5这五个表来入库，一条记录的SN及MOID字段作为公共字段，其它MT字段，分散到五个表中。

## CDMA

## GSM

## LTE