Nbi功能业务整理

-------yanb 2013.9.9

目录

*[目录 2](#_Toc25073)*

*[1概述 3](#_Toc26743)*

*[1.1 文档介绍 3](#_Toc6482)*

*[2. 功能业务介绍 5](#_Toc26990)*

*[2.1 NBI上报功能 5](#_Toc319)*

*[2.1.1 总体 5](#_Toc20686)*

*[2.1.2 时序图 5](#_Toc20175)*

*[2.1.3 流程 6](#_Toc14770)*

*[2.1.4 业务重点 7](#_Toc2439)*

*[2.2 BSA 7](#_Toc27390)*

*[2.2.1 总体图 7](#_Toc3173)*

*[2.2.2 流程 8](#_Toc27834)*

*[2.2.3 业务重点 8](#_Toc31823)*

*[2.2.4 改进 9](#_Toc327)*

*[2.3 性能数据 9](#_Toc18150)*

*[2.3.1 总体 9](#_Toc26979)*

*[2.3.2 流程 9](#_Toc5393)*

*[2.3.3 业务重点 10](#_Toc20748)*

*[2.3.4 拼装规则 13](#_Toc14009)*

*[2.4 路测数据（DT） 13](#_Toc29894)*

*[2.4.1 总体 13](#_Toc31118)*

*[2.4.2 流程 13](#_Toc18902)*

*[2.4.3 不足 14](#_Toc10960)*

*[2.5 广东故障工单 14](#_Toc2401)*

*[2.5.1 总体 14](#_Toc17510)*

*[2.5.2 NBI中实现 14](#_Toc15772)*

*[2.5.3 NBI主要实现部分流程 15](#_Toc2009)*

*[2.5.4 NBI实现业务重点 16](#_Toc10610)*

*[2.5.5 故障单整体规范 17](#_Toc7654)*

*[2.6 协调单 17](#_Toc26178)*

*[2.6.1 总体 17](#_Toc18361)*

*[2.6.2 流程 18](#_Toc12452)*

*[2.6.3 业务重点 18](#_Toc22305)*

*[2.7 透视数据 19](#_Toc25695)*

*[2.7.1 总体 19](#_Toc3382)*

*[2.7.2 流程 19](#_Toc26861)*

*[2.7.3 业务重点 20](#_Toc15170)*

*[2.8 U-CLOUD 21](#_Toc6979)*

*[2.9 广东全流程 21](#_Toc8179)*

*[2.9.1 总体 21](#_Toc26295)*

*[2.9.2 流程 22](#_Toc2848)*

*[2.9.3 业务重点 23](#_Toc3641)*

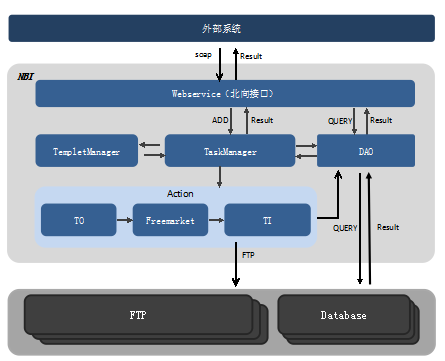
# 1概述

* 1. 文档介绍

本文档是对NBI系统中所涉及业务的一个综合介绍和总结。目的是用于指导开发，测试，以及后期的维护，重构。目前NBI中存在的文档多数是针对整个平台的，其中可能很少部分属于NBI实现，看那些文档不仅耗时，还不一定能看到重点，多数时候总是让人疑惑，故整理出这份文档。但是如果需要了解更多细节，最好还是看代码。

# 功能业务介绍

2. 1. NBI上报功能
      1. 总体



总体图

* + 1. 时序图



* + 1. 流程

1：在任务表中配置任务，分为一般任务和补采任务。当一般任务在上报过程中出现未完成的情况，会把此任务添加到补采任务列表中。补采任务不需要在任务表中配置，由程序自动添加。（补采任务是以单个数据的采集上报为单位，在一个任务模板中可能包含多个数据的采集上报）

2：nbi启动时从任务表和补采表加载任务，以循环的方式执行

3：加载模板，并进行模板解析，模板是NBI中的重点，对于模板的了解可参考如下文件。



4：解析数据源并收集数据：根据模板中配置的数据，进行相关数据的收集，数据源方式有：本地，ftp，数据库，目前基本为数据库收集。并且在收集数据时，涉及到相关的业务均由sql生成。Sql由产品提供。

5：上传数据到ftp：根据模板配置的ftp地址，进行文件上报，如果是设置文件压缩，需要进行文件压缩。上报其实也有可能是把采集的数据经过业务处理后写入到DB。

6：如果有配置后续动作，做后续动作处理。这里主要是指更新任务表，或者对相关的一些数据进行个性化处理等（比如bsa更新五项推导数据）。

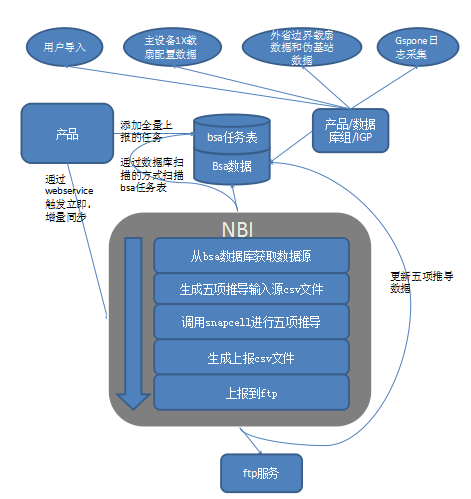
对于NBI的上报主流程来讲：不过多涉及到业务功能，可作为一个通用功能的形式存在于项目中。

* + 1. 业务重点

目前NBI数据上报主要是电信业务，也涉及到部分联通数据，对于大多数数据上报来说，利用通用上报流程，便可完成上报功能。具体的上报数据和上报文件规范见下面文件。



* 1. BSA
  2. 1. 总体图



* + 1. 流程

1：bsa分为数据上报分为两种触发方式：1：webservice触发。2：任务表触发

2：从bsa的任务表读取任务（全量同步从任务表触发，立即同步和增量同步从webservice触发）或者调用webservice触发任务

3：解析模板，获取数据源，bsa的数据收集sql是存放在和前端平台共享的一张表中。

4：执行数据源的sql生成snapcell工具的输入源文件。

5：运行snapcell工具进行五项推导

6：snapcell生成输出文件

7：nbi把输出文件按规范进行名字修改，并上传到ftp

8：更新五项推导出的五个字段

9：最后记录bsa上报的日志，主要记录上报数据方式（立即，增量，全量），上报数据的条数等。

* + 1. 业务重点

1：bsa同步数据上报分为：全量同步，增量同步，立即同步，其中全量同步为任务定时方式，由前端平台把任务写入都DB表，NBI进行扫描。增量同步和立即同步为webservice触发方式，前端进行webservice调用。

2：bsa的数据源sql，bsa的数据源和其它模板不同，不是直接写死在模板中，而是通过模板中的定位sql连接一张和前端平台共享的sql表，进行sql动态获取。前端平台中目前很多sql都是根据业务场景动态生成，如果是在模板中写死的方式，获取到的数据不准确。和前端共享表表名为：CFG\_BSA\_SYNC\_UPLOAD\_SQL，此表由前端应用研发进行维护。

3：当bsa上报完成后，不仅要更新五项推导数据，并且要向前端的一个日志表插入数据，以记录上报数据的方式（全量，增量，立即），上报数据文件条数等。此表为：log\_bsa\_upload\_file

4：五项数据进行推导的数据来源

用SnapCell根据基站天线的经纬度和高度信息以及高通公司的全球地理高程数据库，推导基站扇区中心的经纬度和高度以及地表平均高度和地表高度标准偏差等五项参数。

5：bsa数据表中各数据来源，详情参看下列文档中数据来源章节。

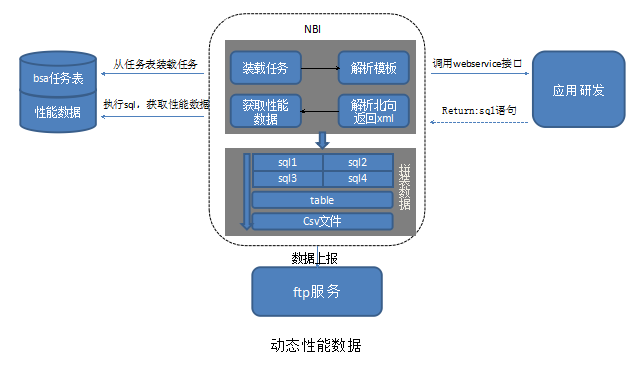


* + 1. 改进

Bsa在目前的实现中几乎是一套独立流程，利用nbi通用功能的地方也是在原有代码内部实现的基础上添加个性化判断，不利于后续维护。

* 1. 性能数据
  2. 1. 总体

性能数据分为：静态性能数据上报和动态性能数据上报。静态性能数据完全走nbi通用流程。动态性能数据上报是指数据源是动态生成的sql，利用webservice动态从前端应用研发获取sql。



* + 1. 流程

**动态性能数据上报流程**  
 1：从nbi\_cfg\_task表加载任务

2：解析任务模板，获取到生成文件名，上报的ftp地址等，但不从模板获取sql语句。

3：获取到webservice地址和接口，在任务模板中，原先配置sql语句的地方，改为配置webservice的地址,接口，参数，生成字段名等。

4：利用从步骤3中获取到的webservice地址，进行北向接口调用，获取到sql语句的xml

5：解析xml获取到具体sql，并执行sql

6：sql结果进行关联合并

7：freemarket生成文件

8：上报ftp

动态数据上报的重点在北向接口返回的是几条sql语句，nbi这边需要按照规范进行结果关联，合并。动态性能数据，目前部署地方是重庆

* + 1. 业务重点

对于静态性能数据上报，完全利用NBI通用流程进行，没什么其他业务功能；主要是动态性能数据上报中通过北向接口获取sql，并进行相关数据的组装需要重点关注。具体如下：

1）：北向接口定义

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数类型 | 参数名称 | 数据类型 | 允许为空 | 说明 |
| 接口参数 | nbisqltype | string | 否 | 获取上报SQL类型。目前有以下6种。每次传入其中一种。 NBI\_PERF\_MSC\_1X NBI\_PERF\_MSC\_DO NBI\_PERF\_CITY\_1X NBI\_PERF\_CITY\_DO NBI\_PERF\_SCENE\_1X NBI\_PERF\_SCENE\_DO |
| 接口返回 | nbisql | string | 否 | 如以下xml格式 |

接口返回xml如下：

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <nbi\_table order\_by="**${order\_by}**">  <tr id = "1" primary\_key="**${primary\_key\_1}**"><!—详细内容-->  <td left\_join="true">**${sql\_1}**</td>  <td>**${sql\_2}**</td>  </tr>  <tr id = "2" primary\_key="**${primary\_key\_2}**"><!—合计-->  <td left\_join="true">**${sql\_3}**</td>  <td>**${sql\_4}**</td>  </tr>  </nbi\_table> |

说明：

1. <nbi\_table> 包含order\_by,表示NBI对整体结果数据的排序处理。
2. <tr> ...</tr> 可能有一组或多组。例如MSC级别上报，仅有1组。即没有合计部分。如CITY性能数据上报，需要额外加一条省份数据，则有两组。
3. 每组<tr>...</tr>中的SQL，必须都要包含primary\_key，这是NBI侧组装数据的关键字。
4. 每组<tr>...</tr>有且仅有一个<td> ..</td> 带有left\_join="true"。表示以此SQL的行数作为结果本次结果的行数。如果输出文件中，在任何一个结果集中都找不到对应的列，则直接填0。
5. <td>..</td>可能有一组或多组。
6. 红色为需要程序处理。对照关系如下。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **nbisqltype** | **${order\_by}** | **${primary\_key\_1}** | **${primary\_key\_2}** | **${sql\_1}** | **${sql\_2}** | **${sql\_3}** | **${sql\_4}** |
| **NBI\_PERF\_MSC\_1X** | 数据时间点 | MSC标识,数据时间点,城市 | 无第二组<tr>..</tr> | 性能详细SQL | 资源详细SQL | 无第二组<tr>..</tr> | 无第二组<tr>..</tr> |
| **NBI\_PERF\_MSC\_DO** | 数据时间点 | MSC标识,数据时间点,城市 | 无第二组<tr>..</tr> | 性能详细SQL | 资源详细SQL | 无第二组<tr>..</tr> | 无第二组<tr>..</tr> |
| **NBI\_PERF\_CITY\_1X** | 数据时间点,对象级别 | 对象级别,数据时间点,城市 | 对象级别,数据时间点,省份 | 性能详细SQL | 资源详细SQL | 性能合计SQL | 资源合计SQL |
| **NBI\_PERF\_CITY\_DO** | 数据时间点,对象级别 | 对象级别,数据时间点,城市 | 对象级别,数据时间点,省份 | 性能详细SQL | 资源详细SQL | 性能合计SQL | 资源合计SQL |
| **NBI\_PERF\_SCENE\_1X** | 数据时间点,场景级别 | 场景级别,数据时间点,场景名称,城市 | 场景级别,数据时间点,场景名称,省份 | 性能详细SQL | 资源详细SQL | 性能合计SQL | 资源合计SQL |
| **NBI\_PERF\_SCENE\_DO** | 数据时间点,场景级别 | 场景级别,数据时间点,场景名称,城市 | 场景级别,数据时间点,场景名称,省份 | 性能详细SQL | 资源详细SQL | 性能合计SQL | 资源合计SQL |

2)：实现中需要考虑如下问题：

A：需要考虑异常、超时等问题。

B：xml中以下符号需要转义。

|  |  |  |
| --- | --- | --- |
| 符号 | 转义前 | 转义后 |
| 小于号 | < | &lt; |
| 大于号 | > | &gt; |

3)：对于性能数据的输出，拥有单表头和双表头两种格式，这个可以通过模板文件进行配置，模板文件中具体属性为：single\_table\_header

* + 1. 拼装规则

1：目前一个table下拥有两行tr，每行tr下拥有两个td，每个td里面有一条sql，总的有4条sql，sql1，sql2，sql3，sql4。Sql1和sql2属于一个tr，sql3和sql4属于一个tr。

2：tr1表示城市级别的相关性能数据。Tr2表示省级相关性能数据。

3：输出排序可按照数据时间点，场景级别，对象级别等进行。

4：生成文件时，NBI分别执行各sql语句，并根据模板中的字段（主要控制单表头输出和双表头输出），ftl中定义的字段等综合进行字段输出。

5：拼装规则：以tr1进行举例

a：根据模板中定义的字段和ftl中定义的字段拼装输出的表头。

b：执行sql1和sql2.循环sql1的数据并根据primary\_key查找sql2中对应数据拼装成一行，如果在sql2中没找到对应数据，输出字段位补为空串。

c：根据a中拼装的表头进行数据查找并输出为csv文件。

* 1. 路测数据（DT）

2. 4. 1. 总体

路测数据，比较简单的一个功能，只是提供北向接口供前端应用研发调用查询数据即可。数据从g网或w网来。并且现有文档也能支撑开发，详见文档（主要包括webservice的接口，返回值。）



* + 1. 流程

1：应用研发调用北向接口

2：在数据库中查询相关内容

3：返回数据给应用研发

由于路测数据流程简单，这里就不画图了，路测数据中涉及到的sql由产品提供。

* + 1. 不足

1：路测数据并没有涉及到上报功能，是否有放在NBI的必要？

2：路测数据中xml的生成和db查询结合在一起，需要整改。

* 1. 广东故障工单
  3. 1. 总体



总体图

* + 1. NBI中实现

在NBI中故障工单主要实现了两个功能：1：派单，2：状态变更（状态更新实现了但实际上没使用，仍然调用的.net实现的更新部分）。

派单：当mone系统根据故障模型产生故障单后，由NBI将故障单发送到外部厂商。根据故障单数据表中的状态有自动派单和手动派单两种。手动派单会在前端页面展示，由前端用户进行触发派单。自动派单由后端任务自动进行。派单每天有数量限制。

状态变更：故障单派送出去后，外部厂商接收到故障单后会进行接口回调改变该单的状态，目前外部系统仍然调用的.net提供的服务。

* + 1. NBI主要实现部分流程

NBI派单实现类中主要包括如下线程进行业务处理：

派单线程，扫描数据库线程，状态更新线程，通知心跳线程

**派单线程：**

1：每天6点唤醒扫描数据库线程，扫描故障单

2：然后唤醒db扫描线程，扫描数据库，看是否有故障单需要派送，并把故障单加入到一个队列

3：如果故障单队列未空，结束本次运行、

4：对故障单进行排序

5：开始进行派单，派单前，先把要发送的故障单从队列中删除（是否需要派送后删除？）

6：生成发送的xml

7：进入派单流程中，先判断是自动派单，还是手动派单

8：如果是自动派单，进行派单数量检查，目前是100（可配），自动派单发送前，先记录日志，如果已经到达每天派单数量限制，需发送邮件进行通知，并退出系统？。如果派单是产生异常，在原来派单的数量上减一（为什么这么做不是很清楚）

9：派单流程结束，如果有异常记录异常日志，从新初始化系统

10：最后一步解析webservice接口返回的xml，记录返回状态到数据库。

**扫描数据库线程：**主要就是进行故障工单获取，一条sql语句完成。

**状态更新线程：**主要是更新处理失败的工单。当工单派送成功后，需要记录状态到数据库，如果这个时候发生异常，便把数据记录到一个map中，这个线程便是专门处理这个hash表的。并且在线程处理中会把map用对象序列化的方式持久化到磁盘。

**通知心跳线程：**代码中实现的只是当到达200次或者5分钟的时候

唤醒数据库扫描线程。

* + 1. NBI实现业务重点

1: 工单筛选（派单规则）

通过故障筛选模型完成故障筛选后，流程进入工单筛选环节，工单筛选的具体要求是：

A：每个类型的网优故障，针对每个地市，用户可以设置派发条件：依据特定指标排序，在每个周期内，派发TOPN张工单；（注：特定指标可由程序人员根据具体要求灵活配置，N可由使用人员随时配置）

B：网优工单在指定日期每天8：00前完成自动派发，剩余工单可以由人工指定继续派发，支持最近一个月待派工单的继续派发；无法派发工单一旦判断即永久保留，不根据工单处理情况更新历史数据。

C：每一个故障筛选模型提供故障等级的设置，自动派发和人工派发时将等级信息传给电子运维，并将处理时限传给电子运维。

D：单个网元如果已有工单派发至电子运维，且尚未得到回单信息，则该网元新生成的同类型筛选故障进入待派工单统一管理，特征标识，不参与自动派单且无法操作派单。

E：单个网元工单自回单起，重新计时过滤故障，计时未达故障筛选周期时，待派工单皆特征标识，不参与自动派单且无法操作派单，特征与未回单工单而无法转派工单区分管理。

2：接口流程

1):工单派发

网优平台完成工单筛选后，通过系统接口转派工单信息至电子运维，除需要包含工单主要故障信息外，还需明确故障等级、网元等级、处理时限等信息。

网优平台派发工单至电子运维后，必须有双方确认机制，确认后，各自记录工单派发时间。

2):流程同步（状态改变）

网优工单派发至电子运维后，电子运维需要将以下状态实时同步至网优平台：处理中、处理完成（评估阶段）、回单；

电子运维进入处理完成后，即评估阶段后，网优平台启动该工单恢复认定，未恢复和恢复后则同步相应信息至电子运维，双方确认后，电子运维流转至处理中或回单状态。

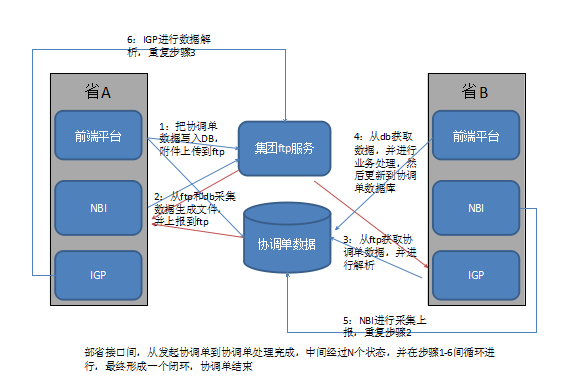
* + 1. 故障单整体规范

故障单整体规范见下列文件：



* 1. 协调单
     1. 总体

总体如下：



* + 1. 流程

1：加载任务

2：解析模板

3：判断是否是协调单

4：获取附件文件名，ftp名中的动态组成的参数，进行文件名替换

5：从数据库获取主子协调单

6：生成文件

7：从ftp下载文件

8：一起打包上传

* + 1. 业务重点

1：协调单分为主单和子单，之间为一对多的关系，主子单间按照字段code进行对应，协调单可能有各类附件，附件命名方式为：协调单主单号\_AS\_协调单主单号\*的方式。

2：前端平台产生协调单后，把数据写入到db表（MOD\_VSITE\_VERGEORDER（主单），MOD\_VSITE\_VERGEORDER\_SUB（子单）），把附件上传到ftp服务器的InterfaceFiles目录，并记录附件相关信息到DB表（MOD\_VSITE\_VERGEORDER\_FILE）

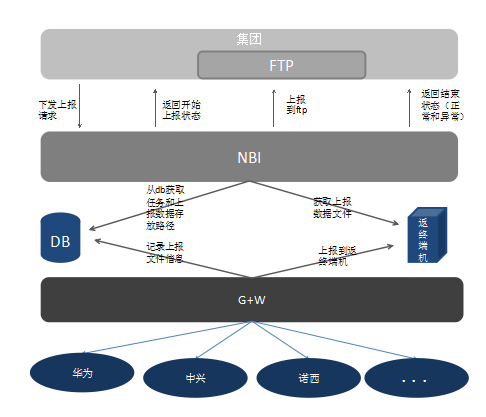
3：NBI每天凌晨3-4点进行协调单上报，根据sql拼装协调单附件名，在ftp上下载相关附件，并打包成zip格式，根据采集sql采集协调单主，子单数据，生成csv文件。

4：NBI把步骤三产生的文件上传到ftp服务器，协调单NBI部分完成。

* 1. 透视数据
     1. 总体

透视数据本应属于G+W系统中的业务功能，在G+W规范文档中定义了相关规格，在G+W系统当初实现时也想过实现相关功能，并已经实现了部分代码，最后不知道为什么没实现完，现在被写入到了NBI系统中。

透视数据上报本质上是把G+W采集上报到返终端机上的数据文件上报到集团。



总体图

* + 1. 流程

1：集团调用NBI北向接口，添加上报数据

2：NBI把需要上报的数据添加到任务表里面（非NBI通用任务表，G+W任务表noss\_upload\_report\_task）

3：NBI任务线程开始周期性扫描任务

4：从noss\_upload\_report\_task加载任务，加载任务时，会根据任务信息在G+W的上报表（noss\_file\_info）中找上报文件存放在返终端机的什么位置

5：调用集团接口，返回开始上报消息

6：开始进行数据上报

7：上报完成，调用集团接口，发送上报结果

* + 1. 业务重点

1：上报文件查找规则

A：装载任务时，首先判断数据类型，G网或则W网数据。

B：判断上报级别，是OMC还是BSC

C：因为集团下发的网元ID和网优公司可能存在区别，因此要根据集团的ID，找到对应的网优公司的消息。

D：在根据C步骤中找到的映射关系，从G+W的上报任务库中找到具体的任务ID。

E：根据任务ID找到对应的任务信息，并找到对应的文件路径。

F：如果是G网的MR数据，还需要打包成tar.gz文件

G：根据透视数据中上报时间段，找到对应的时间段文件。普通数据类型上报中文件查找完成。

H：G网的华为原始性能数据还需要报厂商相关的参数数据一起上报（代码中不是这样的，代码中是参数数据按天一起上报，这个需要在确认）。

2：文件周期上报要求

总部系统向省级系统设置数据文件周期上报的要求，包括：文件类型、对象范围、上报要求、时间间隔、过滤条件等。省级系统设置完成后向总部系统返回响应，并在定时时间点，周期上报数据文件。对同一类型的数据文件，如之前未设置数据文件周期上报要求，则启动周期上报；如该类型数据文件已周期上报，则重置数据文件上报要求，且在间隔一个周期后生效。

3：获取文件请求

总部系统向省级系统请求获取数据文件，单次生效。获取要求包括：文件类型、对象范围、上报要求、时间间隔、过滤条件等。省级系统设置完成后向总部系统返回响应。

4：文件开始上报通知

省级系统按文件周期上报要求或获取文件请求准备好需上报的数据文件，文件准备好后向总部系统上报文件开始上报通知，说明文件的信息和存储FTP位置等。总部系统可针对此通知对省分工作进行考核。

5：文件上报完成通知

省级系统按文件周期上报要求或获取文件请求准备好需上报的数据文件，文件准备好并上报完毕后向总部系统上报文件上报通知，说明文件的信息和存储FTP位置等。总部系统收到文件上报完成通知后，到指定位置获取相关文件。

6：文件异常通知

省级系统按文件周期上报要求或获取文件请求在指定时间无法准备好相关的数据文件，或者在文件上报过程中发生异常，则向总部系统发送文件异常通知，需说明异常情况和异常原因说明。

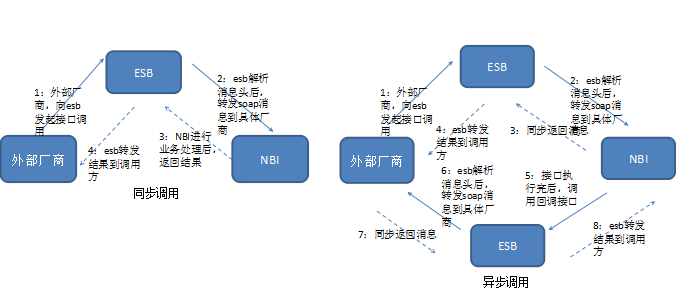
7：附录中国联通规范文档



* 1. U-CLOUD
  2. 广东全流程
     1. 总体

全流程系统为全称为广东联通网络公司全流程管控系统（以下简称泰合佳通系统）是网络公司重要的业务流程系统，涉及到网建系统、网优平台、物业系统、移动综合网管和室分综合网管等多个系统的数据对接工作。其中涉及到网优平台部分便是优网公司实现，主要由NBI提供性能数据的查询（包括G网和W网）。涉及到三个接口：基站性能数据同步查询，基站性能数据异步查询，异步回调接口。

接口间调用关系图如下：



* + 1. 流程

**基站性能数据同步查询（EMOSQuery）**

A：webservice接收到外部请求

B：进行bts校验，如果不存在，直接返回一个空ftp地址。

C：如果bts存在，进行数据查询

D：查询结果返回，生成json格式的文本文件。

E：上传到ftp

F：返回ftp路径。

**基站性能数据异步查询（EMOSAsyncQuery）**

A：webservice接收到外部请求

B：进行bts校验，如果不存在，直接返回1，表示bts不存在

C：把消息加入到消息队列，并返回结果0，表示正常。消息队列暂时未持久化，存在风险。

D：消费线程接收到消息后，EMOSQuery接口进行数据处理。

E：EMOSQuery处理完成后，调用回调接口（EMOSCallBack），通知外部厂商，ftp路径。

* + 1. 业务重点

1：接口间通过soap进行交互，接口如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 接口 | 参数 | 返回值 | 备注 |
| EMOSQuery | BTSName：基站名  NETWORK\_TYPE:网络类型 | Return：ftp路基 | 1：Btsname：泰和调用是传入参数如下:{btsName}。NBI拿到后需要去掉{}  2：生成文件指标见下面网优平台MQI指标;文件格式：{  表[{具体每行},{具体每行}]，  表[{具体每行},{具体每行}]  } |
| EMOSAsyncQuery | BTSName：基站名  NETWORK\_TYPE:网络类型 | Return：0:正常,1:数据不存在 | 同上 |
| EMOSCallBack | BTSName:基站名  FTPpath:ftp路径（ftp全路径，如：ftp://127.0.0.1:21/test/test.txt）  ErrorInfo:如果有异常这里是异常信息，没异常填空 | Return： 0:正常,1:基站不存在，2:文件存在但无数据3:FTP连接错误 | 回调接口时由NBI调用外部厂商，返回值是外部厂商返回给NBI。 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 网优平台MQI指标 | | | | | | | | | | | | | | | | | |
| WCDMA-MQI组成算法 | | | | | | | | | | | | | | | | | |
| 表名: | | | | | | | UwayWcdmaMqi | | | | | | | | | | |
| 基站名称 | 扇区名称 | 扰码 | CS域话务量 | | RLC流量 | | RLC流量 | | 寻呼成功率 | | 网络掉话率 | | 无线系统接通率 | | 系统间CS域切出成功率 | | CS域切出比例 |
| GSM-MQI组成算法 | | | | | | | | | | | | | | | | | |
| 表名 | | | | | | | | UwayGsmMqi | | | | | | | | | |
| 基站名称 | 小区名称 | CGI 话务量 | | GPRS流量 | | 寻呼成功率 | | TCH分配成功率 | | SDCCH拥塞率 | | TCH拥塞率 | | 业务信道掉话率 | | 切换成功率 | |

2：通过亿阳ESB进行调用

1：如果各个服务的接口要通过ESB进行调用，需要增加消息头，消息头主要包括以下元素：

ServiceCode：服务标示，用于标示各厂商的服务，esb通过服务标示找到具体厂商服务进行soap转发。

UserName：用户名。esb进行用户校验

AuthCode：鉴权码。目前没用，随意填写。

EventId：事件ID。目前没用，随意填写。