

ICS 33.040

L 04

**YD**

# 中华人民共和国通信行业标准

YD/T 2427-2012

---

## 2GHz TD-SCDMA 数字蜂窝移动通信网 高速上行链路分组接入 (HSUPA) 网络管理技术要求

2GHz TD-SCDMA digital cell mobile communications network  
HSUPA management technical specification

2012-12-28 发布

2013-03-01 实施

---

中华人民共和国工业和信息化部 发布

## 目 次

前 言	II
1 范围	1
2 规范性引用文件	1
3 术语、定义和缩略语	1
3.1 术语和定义	1
3.2 缩略语	1
4 配置网络资源模型	2
4.1 通用配置网络资源模型	2
4.2 无线接入网配置网络资源模型	2
5 性能网络资源模型	5
5.1 性能参数的命名规则	5
5.2 接入网性能数据	5
6 基于 CORBA 技术的网络资源模型设计	23
6.1 配置网络资源模型设计	23
6.2 性能网络资源模型设计	33
6.3 性能管理接口功能相关的文件	51
附录 A (规范性附录) Schema 文档补充说明	59
附录 B (资料性附录) 性能管理功能相关 XML 文件示例	61
参考文献	64

## 前 言

本标准按照GB/T 1.1-2009给出的规则起草。

本标准由中国通信标准化协会提出并归口。

本标准起草单位：中兴通讯股份有限公司、中国移动通信集团设计院有限公司、鼎桥通信技术有限公司、北京邮电大学。

本标准主要起草人：徐 昕、陈丽萍、黄树强、梁双春、李冶文。

广东省网络空间安全协会受控资料

# 2GHz TD-SCDMA数字蜂窝移动通信网

## 高速上行链路分组接入（HSUPA）网络管理技术要求

### 1 范围

本标准规定了2GHz TD-SCDMA数字蜂窝移动通信网高速上行链路分组接入（HSUPA）网络管理接口的功能、配置网络资源模型、性能网络资源模型，以及基于CORBA技术的网络资源模型设计。

本标准适用于2GHz TD-SCDMA数字蜂窝移动通信网高速上行链路分组接入（HSUPA）的网络管理。

### 2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件，仅所注日期的版本适用于本文件。凡是不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

YD/T 1585.1-2007 2GHz TD-SCDMA数字蜂窝移动通信网网络管理技术要求（第二阶段）第1部分：配置网络资源模型

YD/T 1585.2-2007 2GHz TD-SCDMA数字蜂窝移动通信网网络管理技术要求（第二阶段）第2部分：性能网络资源模型

YD/T 1585.3-2007 2GHz TD-SCDMA数字蜂窝移动通信网网络管理技术要求（第二阶段）第3部分：基于CORBA技术的网络资源模型设计

### 3 术语、定义和缩略语

#### 3.1 术语和定义

YD/T 1585.1-2007和YD/T 1585.2-2007界定的术语和定义适用于本文件。

#### 3.2 缩略语

下列缩略语适用于本文件。

3GPP	3rd Generation Partnership Project	第三代移动通信伙伴计划
AMC	Adaptive Modulation and Coding	自适应调制与编码
ARQ	Automatic Repeat Request	自动重发请求
CCTrCH	Code Composite Transport Channel	编码组合传输信道
CDMA	Code Division Multiple Access	码分多址
CORBA	Common Object Request Broker Architecture	通用请求代理体系结构
DCCH	Dedicated Control Channel	专用控制信道(逻辑信道)
DSCH	Downlink Shared Channel	下行链路共享信道
GPRS	General Packet Radio Service	通用分组无线业务
HARQ	Hybrid Automatic Repeat Request	混合自动重发请求
HSDPA	High Speed Downlink Packet Access	高速下行链路分组接入
HSUPA	High Speed Uplink Packet Access	高速上行链路分组接入
MAC	Medium Access Control	媒体接入控制
MM	Mobility Management	移动性管理

RAB	Radio Access Bearer	无线接入承载
RACH	Random Access Channel	随机接入信道
RB	Radio Bearer	无线承载
RLC	Radio Link Control	无线链路控制
RNC	Radio Network Controller	无线网络控制器
RRC	Radio Resource Control	无线资源控制
RRM	Radio Resource Management	无线资源管理
SRNC	Serving RNC	服务 RNC
TCH	Traffic Channel	业务信道
TDM	Time Division Multiplex	时分复用
TD-SCDMA	Time Division Synchronous CDMA	时分同步码分多址
TFC	Transport Format Combination	传输格式组合
TTI	Transmission Time Interval	传输时间间隔
UE	User Equipment	用户设备
UL	Up-Link	上行链路
UTRAN	UMTS Terrestrial Radio Access Network	UMTS 地面无线接入网
WCDMA	Wideband Code Division Multiple Access	宽带码分多址

#### 4 配置网络资源模型

##### 4.1 通用配置网络资源模型

见YD/T 1585.1-2007中4.1。

##### 4.2 无线接入网配置网络资源模型

###### 4.2.1 无线接入网配置网络资源对象关系图

见YD/T 1585.1-2007中4.2.1。

###### 4.2.2 无线接入网配置网络资源对象

###### 4.2.2.1 RNC 配置对象-RncFunction

###### 4.2.2.1.1 被管对象类描述

见YD/T 1585.1-2007中4.2.2.1.1。

###### 4.2.2.1.2 属性描述

见YD/T 1585.1-2007中4.2.2.1.2。

###### 4.2.2.1.3 可发送的通知描述

见YD/T 1585.1-2007中4.2.2.1.3。

###### 4.2.2.2 Node B 配置对象-Node BFunction

###### 4.2.2.2.1 被管对象类描述

见YD/T 1585.1-2007中4.2.2.2.1。

###### 4.2.2.2.2 属性描述

见YD/T 1585.1-2007中4.2.2.2.2。

###### 4.2.2.2.3 可发送的通知描述

见YD/T 1585.1-2007中4.2.2.2.3。

#### 4.2.2.3 lubLink 配置对象类-lubLink

##### 4.2.2.3.1 被管对象类描述

见YD/T 1585.1-2007中4.2.2.3.1。

##### 4.2.2.3.2 属性描述

见YD/T 1585.1-2007中4.2.2.3.2。

##### 4.2.2.3.3 可发送的通知描述

见YD/T 1585.1-2007中4.2.2.3.3。

#### 4.2.2.4 UtranCell 配置对象类-UtranCell

##### 4.2.2.4.1 被管对象类描述

见YD/T 1585.1-2007中4.2.2.4.1。

##### 4.2.2.4.2 属性描述

见YD/T 1585.1-2007中4.2.2.4.2。

##### 4.2.2.4.3 可发送的通知描述

见YD/T 1585.1-2007中4.2.2.4.3。

#### 4.2.2.5 UtranRelation 配置对象类-UtranRelation

##### 4.2.2.5.1 被管对象类描述

见YD/T 1585.1-2007中4.2.2.5.1。

##### 4.2.2.5.2 属性描述

见YD/T 1585.1-2007中4.2.2.5.2。

##### 4.2.2.5.3 可发送的通知描述

见YD/T 1585.1-2007中4.2.2.5.3。

#### 4.2.2.6 ExternalUtranCell 配置对象类-ExternalUtranCell

##### 4.2.2.6.1 被管对象类描述

见YD/T 1585.1-2007中4.2.2.6.1。

##### 4.2.2.6.2 属性描述

见YD/T 1585.1-2007中4.2.2.6.2。

##### 4.2.2.6.3 可发送的通知描述

见YD/T 1585.1-2007中4.2.2.6.3。

#### 4.2.2.7 GsmRelation 配置对象类 - GsmRelation

##### 4.2.2.7.1 被管对象类描述

见YD/T 1585.1-2007中4.2.2.7.1。

##### 4.2.2.7.2 属性描述

见YD/T 1585.1-2007中4.2.2.7.2。

##### 4.2.2.7.3 可发送的通知描述

见YD/T 1585.1-2007中4.2.2.7.3。

#### 4.2.2.8 ExternalGSMCell 配置对象类-ExternalGSMCell

## 4.2.2.8.1 被管对象类描述

见YD/T 1585.1-2007中4.2.2.8.1。

## 4.2.2.8.2 属性描述

见YD/T 1585.1-2007中4.2.2.8.2。

## 4.2.2.8.3 可发送的通知描述

见YD/T 1585.1-2007中4.2.2.8.3。

## 4.2.2.9 Carrier 配置对象-Carrier

## 4.2.2.9.1 被管对象类描述

Carrier实现载频的所有逻辑功能。该对象实例化在UtranCell之下，有主载频和辅载频之分。UtranCell中依然保留主载频信息。该对象从ManagedFunction对象继承而来。

## 4.2.2.9.2 属性描述（见表1）

表1 Carrier 属性

属性名	中文名称	说明	类型和取值说明	限定
carrierId	载频标识	该对象的RDN命名属性	字符串（命名属性类型）	M, R
userLabel	用户友好名	由OMC厂商设定初始值，作为其内部的标识	字符串	M, R/W
uarfcnType	载频类型	表明主载频还是辅载频。	枚举（主载波(0)，辅载波(1)）	M, R
uarfcn	频点	UTRA绝对无线频率（UARFCN，见3GPP TS 25.331，仅用于TDD）	整型(0..16383)	M, R
timeSlotList	时隙列表	TDD模式小区的时隙配置信息。对于1.28 Mcchip/s小区，列表包含7个元素。每个元素包含三个部分：时隙标识、时隙方向和时隙状态。（见3GPP TS 25.331，仅用于TDD）	时隙信息配置列表。每个时隙信息包括：timeSlotId（整型，1.28 Mcps小区时取值0~6），timeSlotDirection（枚举：UI(0)，DI(1)），timeSlotStatus（枚举：激活(0)，非激活(1)）	M, R/W
operationalState	运行状态	运行状态	枚举	M, R
HSUPA扩展部分				
HsupaFlag	是否支持HSUPA	是否支持HSUPA	枚举（支持，不支持）	M, R
HsupaState	HSUPA是否处于激活状态	HSUPA是否处于激活状态	枚举（激活，非激活）	M, R/W
nbrHsupaEAGCH	授权信道数	每个载频上，配置的E-AGCH信道数	整型(0~4)	M, R
nbrHsupaEHICH	反馈信道数	每个载频上，配置的E-HICH信道数	整型(0~4)	M, R

## 4.2.2.9.3 可发送的通知描述（见表2）

表2 Carrier 可发送通知

中文名称	英文名称	限定
对象创建通知	notifyObjectCreation	M
对象删除通知	notifyObjectDeletion	M
对象属性值改变通知	notifyAttributeValueChange	M
确认状态改变通知	notifyAckStateChanged	M
变化的告警通知	notifyChangedAlarm	C
清除的告警通知	notifyClearedAlarm	M
新的告警通知	notifyNewAlarm	M
增加说明通知	notifyComments	O

## 5 性能网络资源模型

### 5.1 性能参数的命名规则

见YD/T 1585.2-2007中4.1。

### 5.2 接入网性能数据

#### 5.2.1 概述

见YD/T 1585.2-2007中4.5。

#### 5.2.2 RNC 性能数据

##### 5.2.2.1 概述

见YD/T 1585.2-2007中4.6.1。

##### 5.2.2.2 RNC 基本话务数据

见YD/T 1585.2-2007中4.6.2。

##### 5.2.2.3 RNC 切换重定位统计数据

见YD/T 1585.2-2007中4.6.3。

##### 5.2.2.4 No.7 信令性能数据

见YD/T 1585.2-2007中4.6.4。

#### 5.2.3 Cell 性能数据

##### 5.2.3.1 概述

Cell性能数据为与小区相关的性能项数据，包括：

- 小区RRC连接管理统计数据；
- 针对HSUPA服务小区的HSUPA特性相关性能统计数据，包括：
  - MAC-d Flow建立统计（尝试/成功/失败）；
  - HSUPA RB建立统计（尝试/成功/失败）；
  - 小区粒度上E-DCH 服务小区更改统计（尝试/成功）；
  - 小区内E-DCH切换统计（尝试/成功）；
  - 小区间E-DCH切换统计（尝试/成功）；
  - HSUPA小区系统资源统计（吞吐量、用户数、码资源使用）；
  - 完整性测量统计；
  - 正常HSUPA释放统计；
  - E-DCH无线承载异常释放统计。

##### 5.2.3.2 小区 RRC 连接管理统计数据

见YD/T 1585.2-2007中4.7.2。

##### 5.2.3.3 HSUPA 服务小区 HSUPA 特性相关性能统计数据

###### 5.2.3.3.1 MAC-d Flow 建立统计

###### 5.2.3.3.1.1 HSUPA MAC-d Flow 建立尝试次数

- a) 统计RNC MAC-d Flow建立尝试次数。
- b) CC。
- c)



1) RNC向Node B发送“无线链路建立请求”(RADIO LINK SETUP REQUEST), 其中包含IE为“E-DCH MAC-d Flows Information TDD ”;

2) RNC向Node B发送“无线链路重配置准备”(RADIO LINK RECONFIGURATION PREPARE), 其中包含IE为“E-DCH MAC-d Flows To Add ”;

3) RNC向Node B发送“无线链路重配请求”(RADIO LINK RECONFIGURATION REQUEST), 其中包含IE为“E-DCH MAC-d Flows To Add ”;

4) RNC向Node B发送“无线链路增加请求”(RADIO LINK ADDITION REQUEST), 其中包含IE为“E-DCH MAC-d Flows Information TDD ” (3GPP TS 25.433)。

d) 整型。

e) HSUPA.AttMacdSetup。

f) UtranCell。

g) 分组交换。

h) UMTS。

#### 5.2.3.3.1.2 HSUPA MAC-d Flow 建立成功次数

a) 统计RNC MAC-d Flow建立成功次数。

b) CC。

c)

1) RNC向Node B发送含IE为“E-DCH MAC-d Flows Information TDD”的RADIO LINK SETUP REQUEST消息后, 收到从Node B发送的RADIO LINK SETUP RESPONSE消息;

2) RNC向Node B发送包含IE为“E-DCH MAC-d Flows To Add”的RADIO LINK RECONFIGURATION PREPARE消息后, 收到从Node B发送RADIO LINK RECONFIGURATION READY消息;

3) RNC向Node B发送含IE为“E-DCH MAC-d Flows To Add”的RADIO LINK RECONFIGURATION REQUEST消息后, 收到从Node B发送的RADIO LINK RECONFIGURATION RESPONSE消息;

4) RNC向Node B发送含IE为“E-DCH MAC-d Flows Information TDD”的RADIO LINK ADDITION REQUEST消息后, 收到从Node B发送的RADIO LINK ADDITION RESPONSE消息 (3GPP TS 25.433)。

d) 整型。

e) HSUPA.SuccMacdSetup。

f) UtranCell。

g) 分组交换。

h) UMTS。

#### 5.2.3.3.1.3 HSUPA MAC-d Flow 建立失败次数

a) 统计RNC MAC-d Flow建立失败次数。

b) CC。

c)

1) RNC向Node B发送含IE为“E-DCH MAC-d Flows Information TDD”的RADIO LINK SETUP REQUEST消息后, 收到从Node B发送的RADIO LINK SETUP FAILURE消息;

2) RNC向Node B发送包含IE为“E-DCH MAC-d Flows To Add”的RADIO LINK RECONFIGURATION PREPARE消息后,收到从Node B发送RADIO LINK RECONFIGURATION FAILURE消息;

3) RNC向Node B发送含IE为“E-DCH MAC-d Flows To Add”的RADIO LINK RECONFIGURATION REQUEST消息后,收到从Node B发送的RADIO LINK RECONFIGURATION FAILURE消息;

4) RNC向Node B发送含IE为“E-DCH MAC-d Flows Information TDD”的RADIO LINK ADDITION REQUEST消息后,收到从Node B发送的RADIO LINK ADDITION FAILURE消息;

5) RNC向Node B发送含IE为“E-DCH MAC-d Flows Information TDD”的RADIO LINK SETUP REQUEST消息后,收不到从Node B发送的RADIO LINK SETUP RESPONSE消息,失败原因值为‘No Reply’;

6) RNC向Node B发送包含IE为“E-DCH MAC-d Flows To Add”的RADIO LINK RECONFIGURATION PREPARE消息后,收不到从Node B发送RADIO LINK RECONFIGURATION READY消息,失败原因值为‘No Reply’;

7) RNC向Node B发送含IE为“E-DCH MAC-d Flows To Add”的RADIO LINK RECONFIGURATION REQUEST消息后,收不到从Node B发送的RADIO LINK RECONFIGURATION RESPONSE消息,失败原因值为‘No Reply’;

8) RNC向Node B发送含IE为“E-DCH MAC-d Flows Information TDD”的RADIO LINK ADDITION REQUEST消息后,收不到从Node B发送的RADIO LINK ADDITION RESPONSE消息,失败原因值为‘No Reply’(3GPP TS 25.331中未规定)。

统计失败总次数的,使用.sum后缀。每次HSUPA MAC-d Flow建立失败都根据失败原因来增加对应测量项的值。可能的失败原因参见3GPP TS 25.433中的定义。所有失败原因的对应测量项的值相加的总和应该等于.sum测量项的值。如果仅能支持部分失败原因,则首先需要保证提供.sum的测量项。

d) 整型。该类性能测量项的个数等于失败原因个数+1(其中的“1”对应.sum的)。

e) HSUPA.FailMacdSetup.Cause。

后缀Cause表示失败原因。

失败原因是“No Reply”的,使用.NoReply后缀。

f) UtranCell。

g) 分组交换。

h) UMTS。

### 5.2.3.3.2 HSUPA RB 建立统计

#### 5.2.3.3.2.1 HSUPA RB 建立尝试次数

a) 统计PS数据业务指配到HSUPA RB的尝试次数。

b) CC。

c)

1) RNC向UE发送将“RB information to setup”信元中的“RB mapping info”IE中的“Uplink transport channel type”设置为“E-DCH”的“RB建立”(RADIO BEARER SETUP)消息;

2)如果该RB之前的“Uplink transport channel type”不是“E-DCH”,RNC向UE发送将“RB information to reconfigure”信元中的“RB mapping info”IE中的“Uplink transport channel type”设置为“E-DCH”的“RB重配置”(RADIO BEARER RECONFIGURATION)消息(3GPP TS 25.331)。

- d) 整型。
- e) HSUPA.AttRBSetup。
- f) UtranCell。
- g) 分组交换。
- h) UMTS。

#### 5.2.3.3.2.2 HSUPA RB 建立成功次数

- a) 统计PS数据业务指配到HSUPA RB的成功次数。
- b) CC。
- c)

1) RNC向UE发送将“RB information to setup”信元中的“RB mapping info”IE中的“Uplink transport channel type”设置为“E-DCH”的“RB建立”(RADIO BEARER SETUP)消息之后,RNC收到UE发送的建立在HSUPA上的“RB建立完成”(RADIO BEARER SETUP COMPLETE)消息;

2) 如果该RB之前的“Uplink transport channel type”不是“E-DCH”,RNC向UE发送将“RB information to reconfigure”信元中的“RB mapping info”IE中的“Uplink transport channel type”设置为“E-DCH”的“RB重配置”(RADIO BEARER RECONFIGURATION)消息后,RNC收到UE发送的建立在HSUPA上的“RB重配置完成”(RADIO BEARER RECONFIGURATION COMPLETE)消息(3GPP TS 25.331)。

- d) 整型。
- e) HSUPA.SuccRBSetup。
- f) UtranCell。
- g) 分组交换。
- h) UMTS。

#### 5.2.3.3.2.3 HSUPA RB 建立失败次数

- a) 统计PS数据业务指配到HSUPA RB的失败次数。
- b) CC。
- c)

1) RNC向UE发送将“RB information to setup”信元中的“RB mapping info”IE中的“Uplink transport channel type”设置为“E-DCH”的“RB建立”(RADIO BEARER SETUP)消息之后,RNC收到UE发送的建立在HSUPA上的“RB建立失败”(RADIO BEARER SETUP FAILURE)消息;

2) 如果该RB之前的“Uplink transport channel type”不是“E-DCH”,RNC向UE发送将“RB information to reconfigure”信元中的“RB mapping info”IE中的“Uplink transport channel type”设置为“E-DCH”的“RB重配置”(RADIO BEARER RECONFIGURATION)消息后,RNC收到UE发送的建立在HSUPA上的“RB重配置失败”(RADIO BEARER RECONFIGURATION FAILURE)消息;

3) RNC向UE发送将“RB information to setup”信元中的“RB mapping info”IE中的“Uplink transport channel type”设置为“E-DCH”的“RB建立”(RADIO BEARER SETUP)消息之后,RNC收不到UE发送的建立在HSUPA上的“RB建立完成”(RADIO BEARER SETUP COMPLETE)消息。失败原因值为‘No Reply’;

4) 如果该RB之前的“Uplink transport channel type”不是“E-DCH”,RNC向UE发送将“RB information to reconfigure”信元中的“RB mapping info”IE中的“Uplink transport channel type”设置为“E-DCH”的“RB重配置”(RADIO BEARER RECONFIGURATION)消息后,RNC收不到UE发送的建立在HSUPA上的“RB重配置完成”(RADIO BEARER RECONFIGURATION COMPLETE)消息。失败原因值为‘No Reply’(3GPP TS 25.331中未规定)。

统计失败总次数的,使用.sum后缀。每次HSUPA RB建立失败都根据失败原因来增加对应测量项的值。可能的失败原因参见3GPP TS 25.331中的定义。所有失败原因的对应测量项的值相加的总和应该等于.sum测量项的值。如果仅能支持部分失败原因,则首先需要保证提供.sum的测量项。

d) 整型。该类性能测量项的个数等于失败原因个数+1(其中的“1”对应.sum的)。

e) HSUPA.FailRBSetup.Cause。

后缀Cause表示失败原因。

失败原因是“No Reply”的,使用.NoReply后缀。

f) UtranCell。

g) 分组交换。

h) UMTS。

### 5.2.3.3.3 小区粒度上 E-DCH 服务小区更改统计

#### 5.2.3.3.3.1 切出 E-DCH 服务小区更改尝试次数

a) 统计E-DCH信道进行切出服务小区更改的尝试的次数。

b) CC。

c)

1) RNC向UE发送“物理信道重配置”(PHYSICAL CHANNEL RECONFIGURATION)消息次数,且该消息指示UE进行E-DCH服务小区更改出尝试;

2) RNC向UE发送“无线承载建立”(RADIO BEARER SETUP)消息次数,且该消息指示UE进行E-DCH服务小区更改出尝试;

3) RNC向UE发送“无线承载释放”(RADIO BEARER RELEASE)消息次数,且该消息指示UE进行E-DCH服务小区更改出尝试;

4) RNC向UE发送“无线承载重配置”(RADIO BEARER RECONFIGURATION)消息次数,且该消息指示UE进行E-DCH服务小区更改出尝试;

5) RNC向UE发送“传输信道重配置”(TRANSPORT CHANNEL RECONFIGURATION)消息次数,且该消息指示UE进行E-DCH服务小区更改出尝试(3GPP TS 25.331)。

d) 整型。

e) HSUPA.AttOutCellChange。

f) UtranCell。

g) 分组交换。

h) UMTS。

#### 5.2.3.3.3.2 切出 E-DCH 服务小区更改成功次数

a) 统计E-DCH服务小区切出更改成功次数。

b) CC。

c)

1) RNC收到UE发送的“物理信道重配置完成”(PHYSICAL CHANNEL RECONFIGURATION COMPLETE)消息次数,且该消息指示UE的E-DCH服务小区更改出成功完成;

2) RNC收到UE发送的“无线承载建立完成”(RADIO BEARER SETUP COMPLETE)消息次数,且该消息指示UE的E-DCH服务小区更改出成功完成;

3) RNC收到UE发送的“无线承载重配置完成”(RADIO BEARER RECONFIGURATION COMPLETE)消息次数,且该消息指示UE的E-DCH服务小区更改出成功完成;

4) RNC收到UE发送的“无线承载释放完成”(RADIO BEARER RELEASE COMPLETE)消息次数,且该消息指示UE的E-DCH服务小区更改出成功完成;

5) RNC收到UE发送的“传输信道重配置完成”(TRANSPORT CHANNEL RECONFIGURATION COMPLETE)消息次数,且该消息指示UE的E-DCH服务小区更改出成功完成(3GPP TS 25.331)。

d) 整型。

e) HSUPA.SuccOutCellChange。

f) UtranCell。

g) 分组交换。

h) UMTS。

#### 5.2.3.3.4 小区内 E-DCH 切换统计

##### 5.2.3.3.4.1 小区内 RACH 到 E-DCH 信道转换尝试次数

a) 统计小区内 RACH 到 E-DCH 信道转换尝试次数。

b) CC。

c)

1) 信道转换仅仅是小区内从 RACH 到 E-DCH 时, RNC 向 UE 发送“无线承载重配置”(RADIO BEARER RECONFIGURATION)消息次数;

2) 信道转换仅仅是小区内从 RACH 到 E-DCH 时, RNC 向 UE 发送“无线承载建立”(RADIO BEARER SETUP)消息次数;

3) 信道转换仅仅是小区内从 RACH 到 E-DCH 时, RNC 向 UE 发送“无线承载释放”(RADIO BEARER RELEASE)消息次数;

4) 信道转换仅仅是小区内从 RACH 到 E-DCH 时, RNC 向 UE 发送“小区更新确认”(CELL UPDATE CONFIRM)消息次数;

5) 信道转换仅仅是小区内从 RACH 到 E-DCH 时, RNC 向 UE 发送“传输信道重配置”(TRANSPORT CHANNEL RECONFIGURATION)消息次数(3GPP TS 25.331)。

d) 整型。

- e) HSUPA.AttRachToEdchIntraCell。
- f) UtranCell。
- g) 分组交换。
- h) UMTS。

#### 5.2.3.3.4.2 小区内 RACH 到 E-DCH 信道转换成功次数

- a) 统计小区内 RACH 到 E-DCH 信道转换成功次数。
- b) CC。
- c)

1) 信道转换仅仅是小区内从 RACH 到 E-DCH 时, RNC 收到 UE 发送的“无线承载重配置完成”(RADIO BEARER RECONFIGURATION COMPLETE) 消息次数;

2) 信道转换仅仅是小区内从 RACH 到 E-DCH 时, RNC 收到 UE 发送的“无线承载建立完成”(RADIO BEARER SETUP COMPLETE) 消息次数;

3) 信道转换仅仅是小区内从 RACH 到 E-DCH 时, RNC 收到 UE 发送的“无线承载释放完成”(RADIO BEARER RELEASE COMPLETE) 消息次数;

4) 信道转换仅仅是小区内从 RACH 到 E-DCH 时, RNC 收到 UE 发送的“传输信道重配置完成”(TRANSPORT CHANNEL RECONFIGURATION COMPLETE) 消息次数 (3GPP TS 25.331)。

d) 整型。

e) HSUPA.SuccRachToEdchIntraCell。

f) UtranCell。

g) 分组交换。

h) UMTS。

#### 5.2.3.3.4.3 小区内 DCH 到 E-DCH 信道转换尝试次数

- a) 统计小区内 DCH 到 E-DCH 信道转换尝试次数。
- b) CC。
- c)

1) 信道转换仅仅是小区内从 DCH 到 E-DCH 时, RNC 向 UE 发送“无线承载重配置”(RADIO BEARER RECONFIGURATION) 消息次数;

2) 信道转换仅仅是小区内从 DCH 到 E-DCH 时, RNC 向 UE 发送“无线承载建立”(RADIO BEARER SETUP) 消息次数;

3) 信道转换仅仅是小区内从 DCH 到 E-DCH 时, RNC 向 UE 发送“无线承载释放”(RADIO BEARER RELEASE) 消息次数;

4) 信道转换仅仅是小区内从 DCH 到 E-DCH 时, RNC 向 UE 发送“小区更新确认”(CELL UPDATE CONFIRM) 消息次数;

5) 信道转换仅仅是小区内从 DCH 到 E-DCH 时, RNC 向 UE 发送“传输信道重配置”(TRANSPORT CHANNEL RECONFIGURATION) 消息次数 (3GPP TS 25.331)。

d) 整型。

e) HSUPA.AttDchToEdchIntraCell。

- f) UtranCell。
- g) 分组交换。
- h) UMTS。

#### 5.2.3.3.4.4 小区内 DCH 到 E-DCH 信道转换成功次数

- a) 统计小区内 DCH 到 E-DCH 信道转换成功次数。
- b) CC。
- c)

1) 信道转换仅仅是小区内从 DCH 到 E-DCH 时, RNC 收到 UE 发送的“无线承载重配置完成”(RADIO BEARER RECONFIGURATION COMPLETE) 消息次数;

2) 信道转换仅仅是小区内从 DCH 到 E-DCH 时, RNC 收到 UE 发送的“无线承载建立完成”(RADIO BEARER SETUP COMPLETE) 消息次数;

3) 信道转换仅仅是小区内从 DCH 到 E-DCH 时, RNC 收到 UE 发送的“无线承载释放完成”(RADIO BEARER RELEASE COMPLETE) 消息次数;

4) 信道转换仅仅是小区内从 DCH 到 E-DCH 时, RNC 收到 UE 发送的“传输信道重配置完成”(TRANSPORT CHANNEL RECONFIGURATION COMPLETE) 消息次数 (3GPP TS 25.331)。

- d) 整型。
- e) HSUPA.SuccDchToEdchIntraCell。
- f) UtranCell。
- g) 分组交换。
- h) UMTS。

#### 5.2.3.3.4.5 小区内 E-DCH 到 RACH 信道转换尝试次数

- a) 统计小区内 E-DCH 到 RACH 信道转换尝试次数。
- b) CC。
- c)

1) 信道转换仅仅是小区内从 E-DCH 到 RACH 时, RNC 向 UE 发送“无线承载重配置”(RADIO BEARER RECONFIGURATION) 消息次数;

2) 信道转换仅仅是小区内从 E-DCH 到 RACH 时, RNC 向 UE 发送“无线承载建立”(RADIO BEARER SETUP) 消息次数;

3) 信道转换仅仅是小区内从 E-DCH 到 RACH 时, RNC 向 UE 发送“无线承载释放”(RADIO BEARER RELEASE) 消息次数;

4) 信道转换仅仅是小区内从 E-DCH 到 RACH 时, RNC 向 UE 发送“小区更新确认”(CELL UPDATE CONFIRM) 消息次数;

5) 信道转换仅仅是小区内从 E-DCH 到 RACH 时, RNC 向 UE 发送“传输信道重配置”(TRANSPORT CHANNEL RECONFIGURATION) 消息次数 (3GPP TS 25.331)。

- d) 整型。
- e) HSUPA.AttEdchToRachIntraCell。
- f) UtranCell。

- g) 分组交换。
- h) UMTS。

#### 5.2.3.3.4.6 小区内 E-DCH 到 RACH 信道转换成功次数

- a) 统计小区内E-DCH 到RACH信道转换成功次数。
- b) CC。
- c)

1) 信道转换仅仅是小区内从E-DCH 到RACH时，RNC收到UE发送的“无线承载重配置完成”（RADIO BEARER RECONFIGURATION COMPLETE）消息次数；

2) 信道转换仅仅是小区内从E-DCH 到RACH时，RNC收到UE发送的“无线承载建立完成”（RADIO BEARER SETUP COMPLETE）消息次数；

3) 信道转换仅仅是小区内从E-DCH 到RACH时，RNC收到UE发送的“无线承载释放完成”（RADIO BEARER RELEASE COMPLETE）消息次数；

4) 信道转换仅仅是小区内从E-DCH 到RACH时，RNC收到UE发送的“传输信道重配置完成”（TRANSPORT CHANNEL RECONFIGURATION COMPLETE）消息次数（3GPP TS 25.331）。

- d) 整型。
- e) HSUPA.SuccEdchToRachIntraCell。
- f) UtranCell。
- g) 分组交换。
- h) UMTS。

#### 5.2.3.3.4.7 小区内 E-DCH 到 DCH 信道转换尝试次数

- a) 统计小区内E-DCH到DCH信道转换尝试次数。
- b) CC。
- c)

1) 信道转换仅仅是小区内从E-DCH到DCH时，RNC向UE发送“无线承载重配置”（RADIO BEARER RECONFIGURATION）消息次数；

2) 信道转换仅仅是小区内从E-DCH到DCH时，RNC向UE发送“无线承载建立”（RADIO BEARER SETUP）消息次数；

3) 信道转换仅仅是小区内从E-DCH到DCH时，RNC向UE发送“无线承载释放”（RADIO BEARER RELEASE）消息次数；

4) 信道转换仅仅是小区内从E-DCH到DCH时，RNC向UE发送“小区更新确认”（CELL UPDATE CONFIRM）消息次数；

5) 信道转换仅仅是小区内从DCH到E-DCH时，RNC向UE发送“传输信道重配置”（TRANSPORT CHANNEL RECONFIGURATION）消息次数（3GPP TS 25.331）。

- d) 整型。
- e) HSUPA.AttEdchToDchIntraCell。
- f) UtranCell。
- g) 分组交换。



h) UMTS。

#### 5.2.3.3.4.8 小区内 E-DCH 到 DCH 信道转换成功次数

a) 统计小区内E-DCH到DCH信道转换成功次数。

b) CC。

c)

1) 信道转换仅仅是小区内从E-DCH到DCH时, RNC收到UE发送的“无线承载重配置完成”(RADIO BEARER RECONFIGURATION COMPLETE) 消息次数;

2) 信道转换仅仅是小区内从E-DCH到DCH时, RNC收到UE发送的“无线承载建立完成”(RADIO BEARER SETUP COMPLETE) 消息次数;

3) 信道转换仅仅是小区内从E-DCH到DCH时, RNC收到UE发送的“无线承载释放完成”(RADIO BEARER RELEASE COMPLETE) 消息次数;

4) ANSPORT CHANNEL RECONFIGURATION COMPLETE) 消息次数 (3GPP TS 25.331)。

d) 整型。

e) HSUPA.SuccEdchToDchIntraCell。

f) UtranCell。

g) 分组交换。

h) UMTS。

#### 5.2.3.3.5 小区间 E-DCH 切换统计

##### 5.2.3.3.5.1 小区间 RACH 到 E-DCH 信道转换尝试次数

a) 统计小区间 RACH 到 E-DCH 信道转换尝试次数。

b) CC。

c)

1) 信道转换仅仅是小区间从 RACH 到 E-DCH 时, RNC 向 UE 发送“无线承载重配置”(RADIO BEARER RECONFIGURATION) 消息次数;

2) 信道转换仅仅是小区间从 RACH 到 E-DCH 时, RNC 向 UE 发送“无线承载建立”(RADIO BEARER SETUP) 消息次数;

3) 信道转换仅仅是小区间从 RACH 到 E-DCH 时, RNC 向 UE 发送“无线承载释放”(RADIO BEARER RELEASE) 消息次数;

4) 信道转换仅仅是小区间从 RACH 到 E-DCH 时, RNC 向 UE 发送“小区更新确认”(CELL UPDATE CONFIRM) 消息次数;

5) 信道转换仅仅是小区间从 RACH 到 E-DCH 时, RNC 向 UE 发送“传输信道重配置”(TRANSPORT CHANNEL RECONFIGURATION) 消息次数 (3GPP TS 25.331) 。

d) 整型。

e) HSUPA.AttRachToEdchInterCell。

f) UtranCell。

g) 分组交换。

h) UMTS。

#### 5.2.3.3.5.2 小区间 RACH 到 E-DCH 信道转换成功次数

a) 统计小区间 RACH 到 E-DCH 信道转换成功次数。

b) CC。

c)

1) 信道转换仅仅是小区间从 RACH 到 E-DCH 时, RNC 收到 UE 发送的“无线承载重配置完成”(RADIO BEARER RECONFIGURATION COMPLETE) 消息次数;

2) 信道转换仅仅是小区间从 RACH 到 E-DCH 时, RNC 收到 UE 发送的“无线承载建立完成”(RADIO BEARER SETUP COMPLETE) 消息次数;

3) 信道转换仅仅是小区间从 RACH 到 E-DCH 时, RNC 收到 UE 发送的“无线承载释放完成”(RADIO BEARER RELEASE COMPLETE) 消息次数;

4) 信道转换仅仅是小区间从 RACH 到 E-DCH 时, RNC 收到 UE 发送的“传输信道重配置完成”(TRANSPORT CHANNEL RECONFIGURATION COMPLETE) 消息次数 (3GPP TS 25.331)。

d) 整型。

e) HSUPA.SuccRachToEdchInterCell。

f) UtranCell。

g) 分组交换。

h) UMTS。

#### 5.2.3.3.5.3 小区间 DCH 到 E-DCH 信道转换尝试次数

a) 统计小区间 DCH 到 E-DCH 信道转换尝试次数。

b) CC。

c)

1) 信道转换仅仅是小区间从 DCH 到 E-DCH 时, RNC 向 UE 发送“无线承载重配置”(RADIO BEARER RECONFIGURATION) 消息次数;

2) 信道转换仅仅是小区间从 DCH 到 E-DCH 时, RNC 向 UE 发送“无线承载建立”(RADIO BEARER SETUP) 消息次数;

3) 信道转换仅仅是小区间从 DCH 到 E-DCH 时, RNC 向 UE 发送“无线承载释放”(RADIO BEARER RELEASE) 消息次数;

4) 信道转换仅仅是小区间从 DCH 到 E-DCH 时, RNC 向 UE 发送“小区更新确认”(CELL UPDATE CONFIRM) 消息次数;

5) 信道转换仅仅是小区间从 DCH 到 E-DCH 时, RNC 向 UE 发送“传输信道重配置”(TRANSPORT CHANNEL RECONFIGURATION) 消息次数 (3GPP TS 25.331)。

d) 整型。

e) HSUPA.AttDchToEdchInterCell。

f) UtranCell。

g) 分组交换。

h) UMTS。

#### 5.2.3.3.5.4 小区间 DCH 到 E-DCH 信道转换成功次数

- a) 统计小区间 DCH 到 E-DCH 信道转换成功次数。
- b) CC。
- c)

1) 信道转换仅仅是小区间从 DCH 到 E-DCH 时, RNC 收到 UE 发送的“无线承载重配置完成”(RADIO BEARER RECONFIGURATION COMPLETE) 消息次数;

2) 信道转换仅仅是小区间从 DCH 到 E-DCH 时, RNC 收到 UE 发送的“无线承载建立完成”(RADIO BEARER SETUP COMPLETE) 消息次数;

3) 信道转换仅仅是小区间从 DCH 到 E-DCH 时, RNC 收到 UE 发送的“无线承载释放完成”(RADIO BEARER RELEASE COMPLETE) 消息次数;

4) 信道转换仅仅是小区间从 DCH 到 E-DCH 时, RNC 收到 UE 发送的“传输信道重配置完成”(TRANSPORT CHANNEL RECONFIGURATION COMPLETE) 消息次数 (3GPP TS 25.331)。

- d) 整型。
- e) HSUPA.SuccDchToEdchInterCell。
- f) UtranCell。
- g) 分组交换。
- h) UMTS。

#### 5.2.3.3.5.5 小区间 E-DCH 到 RACH 信道转换尝试次数

- a) 统计小区间 E-DCH 到 RACH 信道转换尝试次数。
- b) CC。
- c)

1) 信道转换仅仅是小区间从 E-DCH 到 RACH 时, RNC 向 UE 发送“无线承载重配置”(RADIO BEARER RECONFIGURATION) 消息次数;

2) 信道转换仅仅是小区间从 E-DCH 到 RACH 时, RNC 向 UE 发送“无线承载建立”(RADIO BEARER SETUP) 消息次数;

3) 信道转换仅仅是小区间从 E-DCH 到 RACH 时, RNC 向 UE 发送“无线承载释放”(RADIO BEARER RELEASE) 消息次数;

4) 信道转换仅仅是小区间从 E-DCH 到 RACH 时, RNC 向 UE 发送“小区更新确认”(CELL UPDATE CONFIRM) 消息次数;

5) 信道转换仅仅是小区间从 E-DCH 到 RACH 时, RNC 向 UE 发送“传输信道重配置”(TRANSPORT CHANNEL RECONFIGURATION) 消息次数 (3GPP TS 25.331)。

- d) 整型。
- e) HSUPA.AttEdchToRachInterCell。
- f) UtranCell。
- g) 分组交换。
- h) UMTS。

#### 5.2.3.3.5.6 小区间 E-DCH 到 RACH 信道转换成功次数

- a) 统计小区间 E-DCH 到 RACH 信道转换成功次数。

b) CC。

c)

1) 信道转换仅仅是小区间从E-DCH 到RACH时, RNC收到UE发送的“无线承载重配置完成”(RADIO BEARER RECONFIGURATION COMPLETE) 消息次数;

2) 信道转换仅仅是小区间从E-DCH 到RACH时, RNC收到UE发送的“无线承载建立完成”(RADIO BEARER SETUP COMPLETE) 消息次数;

3) 信道转换仅仅是小区间从E-DCH 到RACH时, RNC收到UE发送的“无线承载释放完成”(RADIO BEARER RELEASE COMPLETE) 消息次数;

4) 信道转换仅仅是小区间从E-DCH 到RACH时, RNC收到UE发送的“传输信道重配置完成”(TRANSPORT CHANNEL RECONFIGURATION COMPLETE) 消息次数(3GPP TS 25.331)。

d) 整型。

e) HSUPA.SuccEdchToRachInterCell。

f) UtranCell。

g) 分组交换。

h) UMTS。

#### 5.2.3.3.5.7 小区间 E-DCH 到 DCH 信道转换尝试次数

a) 统计小区间E-DCH到DCH信道转换尝试次数。

b) CC。

c)

1) 信道转换仅仅是小区间从E-DCH到DCH时, RNC向UE发送“无线承载重配置”(RADIO BEARER RECONFIGURATION) 消息次数;

2) 信道转换仅仅是小区间从E-DCH到DCH时, RNC向UE发送“无线承载建立”(RADIO BEARER SETUP) 消息次数;

3) 信道转换仅仅是小区间从E-DCH到DCH时, RNC向UE发送“无线承载释放”(RADIO BEARER RELEASE) 消息次数;

4) 信道转换仅仅是小区间从E-DCH到DCH时, RNC向UE发送“小区更新确认”(CELL UPDATE CONFIRM) 消息次数;

5) 信道转换仅仅是小区间从DCH到E-DCH时, RNC向UE发送“传输信道重配置”(TRANSPORT CHANNEL RECONFIGURATION) 消息次数(3GPP TS 25.331)。

d) 整型。

e) HSUPA.AttEdchToDchInterCell。

f) UtranCell。

g) 分组交换。

h) UMTS。

#### 5.2.3.3.5.8 小区间 E-DCH 到 DCH 信道转换成功次数

a) 统计小区间E-DCH到DCH信道转换成功次数。

b) CC。

c)

1) 信道转换仅仅是小区间从E-DCH到DCH时, RNC收到UE发送的“无线承载重配置完成”(RADIO BEARER RECONFIGURATION COMPLETE) 消息次数;

2) 信道转换仅仅是小区间从E-DCH到DCH时, RNC收到UE发送的“无线承载建立完成”(RADIO BEARER SETUP COMPLETE) 消息次数;

3) 信道转换仅仅是小区间从E-DCH到DCH时, RNC收到UE发送的“无线承载释放完成”(RADIO BEARER RELEASE COMPLETE) 消息次数;

4) 信道转换仅仅是小区间从E-DCH到DCH时, RNC收到UE发送的“传输信道重配置完成”(TRANSPORT CHANNEL RECONFIGURATION COMPLETE) 消息次数(3GPP TS 25.331)。

d) 整型。

e) HSUPA.SuccEdchToDchInterCell。

f) UtranCell。

g) 分组交换。

h) UMTS。

#### 5.2.3.3.6 HSUPA 小区系统资源统计 (吞吐量、用户数、码资源使用)

##### 5.2.3.3.6.1 E-DCH 平均用户数

a) 在测试周期中测试小区内平均的 E-DCH 用户数。

b) SI。

c) 小区内平均的 E-DCH 用户数。

d) 实型。

e) HSUPA.MeanNbrUser。

f) UtranCell。

g) 分组交换。

h) UMTS。

##### 5.2.3.3.7 完整性测量统计

###### 5.2.3.3.7.1 收到的 MAC-E 分组数

a) UE 收到的 MAC-E 分组的次数。

b) CC。

c) UE 收到的 MAC-E 分组的次数。

d) 整型。

e) HSUPA.NbrMacePdu。

f) UtranCell。

g) 分组交换。

h) UMTS。

###### 5.2.3.3.7.2 确认的 MAC-E 分组数

a) 小区正确接收到 MAC-E 分组的次数。

b) CC。

- c) 小区正确接收到 MAC-E 分组的次数。
- d) 整型。
- e) HSUPA.NbrAckdMacePdu。
- f) UtranCell。
- g) 分组交换。
- h) UMTS。

#### 5.2.3.3.7.3 确认的 MAC-E 分组字节数

- a) 在测试的采样周期中从所有手机用户中成功接收的 MAC-E分组的总的字节数。
- b) CC。
- c) 从所有手机用户中成功接收的 MAC-E分组的总的字节数。
- d) 整型。
- e) HSUPA.NbrAckdMaceOcts。
- f) UtranCell。
- g) 分组交换。
- h) UMTS。

#### 5.2.3.3.8 正常 HSUPA 释放统计

##### 5.2.3.3.8.1 RNC 因用户不激活而触发用户 E-DCH 信道释放的次数

- a) 统计在服务 E-DCH 小区，由于用户不活而导致释放 E-DCH 信道的次数。
- b) CC。
- c)
  - 1) RNC 由于检测到在预定义的间隔内，UE 没有数据发送，认为用户不激活，RNC 向 UE 发送“无线承载重配置”（RADIO BEARER RECONFIGURATION）消息次数；
  - 2) RNC 由于检测到在预定义的间隔内，UE 没有数据发送，认为用户不激活，RNC 向 UE 发送“无线承载释放”（RADIO BEARER RELEASE）消息次数；
  - 3) RNC 由于检测到在预定义的间隔内，UE 没有数据发送，认为用户不激活，RNC 向 UE 发送“无线资源控制连接释放”（RRC CONNECTION RELEASE）消息次数。
- d) 整型。
- e) HSUPA.SuccEdchReleaseUserInact。
- f) UtranCell。
- g) 分组交换。
- h) UMTS。

##### 5.2.3.3.8.2 RNC 因 E-DCH 转换为 DCH/RACH 信道而触发用户 E-DCH 信道释放的次数

- a) 统计小区内或小区间，由于E-DCH 到DCH或RACH信道转换而导致释放E-DCH信道的次数。
- b) CC。
- c)
  - 1) 由于小区内或小区间E-DCH 信道转换为DCH或RACH，RNC向UE发送“无线承载重配置”（RADIO BEARER RECONFIGURATION）消息次数；

2) 由于小区内或小区间E-DCH 信道转换为DCH或RACH, RNC向UE发送“无线承载建立”(RADIO BEARER SETUP) 消息次数;

3) 由于小区内或小区间E-DCH 信道转换为DCH或RACH, RNC向UE发送“无线承载释放”(RADIO BEARER RELEASE) 消息次数;

4) 由于小区内或小区间E-DCH 信道转换为DCH或RACH, RNC向UE发送“小区更新确认”(CELL UPDATE CONFIRM) 消息次数;

5) 由于小区内或小区间E-DCH 信道转换为DCH或RACH, RNC向UE发送“传输信道重配置”(TRANSPORT CHANNEL RECONFIGURATION) 消息次数(3GPP TS 25.331)。

d) HSUPA.SuccEdchReleaseEdchToDchOrRach.

e) UtranCell.

f) 分组交换。

g) UMTS.

#### 5.2.3.3.8.3 RNC 因 E-DCH 服务小区改变而触发用户 E-DCH 信道释放的次数

a) 统计由于E-DCH信道进行切出服务小区更改而导致释放E-DCH信道的次数。

b) CC.

c)

1) RNC向UE发送“物理信道重配置”(PHYSICAL CHANNEL RECONFIGURATION) 消息次数, 且该消息指示UE进行E-DCH服务小区更改出尝试;

2) RNC向UE发送“无线承载建立”(RADIO BEARER SETUP) 消息次数, 且该消息指示UE进行E-DCH服务小区更改出尝试;

3) RNC向UE发送“无线承载释放”(RADIO BEARER RELEASE) 消息次数, 且该消息指示UE进行E-DCH服务小区更改出尝试;

4) RNC向UE发送“无线承载重配置”(RADIO BEARER RECONFIGURATION) 消息次数, 且该消息指示UE进行E-DCH服务小区更改出尝试;

5) RNC向UE发送“传输信道重配置”(TRANSPORT CHANNEL RECONFIGURATION) 消息次数, 且该消息指示UE进行E-DCH服务小区更改出尝试(3GPP TS 25.331)。

d) 整型。

e) HSUPA. SuccEdchReleaseOutCellChange.

f) UtranCell.

g) 分组交换。

h) UMTS.

#### 5.2.3.3.8.4 核心网触发用户 E-DCH 信道释放的次数

a) 统计在服务 E-DCH 小区, 由于核心网发起释放, 而导致用户 E-DCH 信道释放的次数。

b) CC.

c)

1) RNC 收到 CN 下发的“无线接入承载指派请求”(RAB ASSIGNMENT REQUEST)或“IU 释放命令”(IU RELEASE COMMAND)消息, RNC 向 UE 发送“无线承载重配置”(RADIO BEARER RECONFIGURATION)消息次数;

2) RNC 收到 CN 下发的“无线接入承载指派请求”(RAB ASSIGNMENT REQUEST)或“IU 释放命令”(IU RELEASE COMMAND)消息, RNC 向 UE 发送“无线承载释放”(RADIO BEARER RELEASE)消息次数;

3) RNC 收到 CN 下发的“无线接入承载指派请求”(RAB ASSIGNMENT REQUEST)或“IU 释放命令”(IU RELEASE COMMAND)消息, RNC 向 UE 发送“无线资源控制连接释放”(RRC CONNECTION RELEASE)消息次数。

d) 整型。

e) HSUPA.SuccEdchReleaseCnInit。

f) UtranCell。

g) 分组交换。

h) UMTS。

### 5.2.3.3.9 E-DCH 无线承载异常释放统计

#### 5.2.3.3.9.1 E-DCH 无线承载异常释放次数

a) 统计 E-DCH 无线承载异常释放的次数。

b) CC。

c)

1) RNC 由于其它异常原因, 而非用户不活、信道迁移、服务 E-DCH 小区改变或核心网释放原因, 释放用户 E-DCH 信道, RNC 收到“无线承载重配完成”(RADIO BEARER RECONFIGURATION COMPLETE)消息次数;

2) RNC 由于其它异常原因, 而非用户不活、信道迁移、服务 E-DCH 小区改变或核心网释放原因, 释放用户 E-DCH 信道, UE 未响应“无线承载重配”(RADIO BEARER RECONFIGURATION)消息次数;

3) RNC 由于其它异常原因, 而非用户不活、信道迁移、服务 E-DCH 小区改变或核心网释放原因, 释放用户 E-DCH 信道, RNC 收到“无线承载释放完成”(RADIO BEARER RELEASE COMPLETE)消息次数;

4) RNC 由于其它异常原因, 而非用户不活、信道迁移、服务 E-DCH 小区改变或核心网释放原因, 释放用户 E-DCH 信道, UE 未响应“无线承载释放”(RADIO BEARER RELEASE)消息次数。

d) 整型。

e) HSUPA.AbnormalEdchRelease。

f) UtranCell。

g) 分组交换。

h) UMTS。

## 5.2.4 UtranRelation 性能数据

### 5.2.4.1 概述



UtranRelation性能数据为小区之间相关联的性能项数据，包括：

- 针对小区关系的HSUPA特性相关性能统计数据；
- 小区关系粒度上E-DCH 服务小区更改统计（尝试/成功）；
- Node B内小区间硬切换统计数据；
- RNC内Node B间小区间硬切换统计数据；
- 通过Iur接口的RNC间小区间硬切换统计数据；
- 核心网控制的RNC间小区间硬切换统计数据；
- 处于不同系统（不同无线接入技术）的小区间切换统计数据。

#### 5.2.4.2 小区关系粒度上 E-DCH 服务小区更改统计

##### 5.2.4.2.1 按照邻接小区关系的切出 E-DCH 服务小区更改尝试次数

- a) 按照邻接小区关系，统计E-DCH信道进行切出服务小区更改的尝试的次数。
- b) CC。
- c)
  - 1) RNC向UE发送“物理信道重配置”（PHYSICAL CHANNEL RECONFIGURATION）消息次数，且该消息指示UE进行E-DCH服务小区更改出尝试；
  - 2) RNC向UE发送“无线承载建立”（RADIO BEARER SETUP）消息次数，且该消息指示UE进行E-DCH服务小区更改出尝试；
  - 3) RNC向UE发送“无线承载释放”（RADIO BEARER RELEASE）消息次数，且该消息指示UE进行E-DCH服务小区更改出尝试；
  - 4) RNC向UE发送“无线承载重配置”（RADIO BEARER RECONFIGURATION）消息次数，且该消息指示UE进行E-DCH服务小区更改出尝试；
  - 5) RNC向UE发送“传输信道重配置”（TRANSPORT CHANNEL RECONFIGURATION）消息次数，且该消息指示UE进行E-DCH服务小区更改出尝试（3GPP TS 25.331）。
- d) 整型。
- e) HSUPA.AttOut。
- f) UtranRelation。
- g) 分组交换。
- h) UMTS。

##### 5.2.4.2.2 按照邻接小区关系的切出 E-DCH 服务小区更改成功次数

- a) 按照邻接小区关系，统计E-DCH服务小区切出更改成功次数。
- b) CC。
- c)
  - 1) RNC收到UE发送的“物理信道重配置完成”（PHYSICAL CHANNEL RECONFIGURATION COMPLETE）消息次数，且该消息指示UE的E-DCH服务小区更改出成功完成；
  - 2) RNC收到UE发送的“无线承载建立完成”（RADIO BEARER SETUP COMPLETE）消息次数，且该消息指示UE的E-DCH服务小区更改成功完成；

3) RNC收到UE发送的“无线承载重配置完成”(RADIO BEARER RECONFIGURATION COMPLETE)消息次数,且该消息指示UE的E-DCH服务小区更改出成功完成;

4) RNC收到UE发送的“无线承载释放完成”(RADIO BEARER RELEASE COMPLETE)消息次数,且该消息指示UE的E-DCH服务小区更改出成功完成;

5) RNC收到UE发送的“传输信道重配置完成”(TRANSPORT CHANNEL RECONFIGURATION COMPLETE)消息次数,且该消息指示UE的E-DCH服务小区更改出成功完成(3GPP TS 25.331)。

d) 整型。

e) HSUPA.SuccOut。

f) UtranRelation。

g) 分组交换。

h) UMTS。

#### 5.2.4.3 Node B 内小区间硬切换统计数据

见YD/T 1585.2-2007中4.8.2。

#### 5.2.4.4 RNC 内 Node B 间小区间硬切换统计数据

见YD/T 1585.2-2007中4.8.3。

#### 5.2.4.5 通过 Iur 接口的 RNC 间小区间硬切换统计数据

见YD/T 1585.2-2007中4.8.4。

#### 5.2.4.6 核心网控制的 RNC 间小区间硬切换统计数据

见YD/T 1585.2-2007中4.8.5。

#### 5.2.4.7 处于不同系统(不同无线接入技术)的小区间切换统计数据

见YD/T 1585.2-2007中4.8.6。

### 6 基于 CORBA 技术的网络资源模型设计

#### 6.1 配置网络资源模型设计

配置网络资源模型设计中有3类idl文件,这3类文档及其用途如下:

1) xxxNRMDefs.idl包括GenericNRMDefs.idl、IMDataDefs.idl和UtranNRMDefs.idl,用来定义配置网络对象及其属性名称;

2) xxxNRMSystem.idl包括GenericNRMSystem.idl和UtranNRMSystem.idl,用来定义配置网络资源对象的属性使用的数据类型;

3) xxxNRMProfile.idl包括GenericNRMProfile.idl、IMDataProfile.idl和UtranNRMProfile.idl,只是用来描述配置网络资源对象的属性名称及其数据类型的对应关系,实现时并不使用此类IDL文件。

##### 6.1.1 通用配置资源模型的 IDL 定义

见YD/T 1585.3-2007中4.1给出的IDL定义。

##### 6.1.2 无线接入网网络资源模型的 IDL 定义

###### 6.1.2.1.1 UtranNRMDefs

```
//File "UtranNRMDefs.idl"
//The IRP document version number is "UTRAN NRM V1.0"
#ifdef UtranNRMDefs_idl
```

```
#define UtranNRMDefs_idl

#include "GenericNRMDefs.idl"

#pragma prefix "3gppsa5.org"

/**
 * This module defines constants for each MO class name and
 * the attribute names for each defined MO class.
 */
module UtranNRMDefs
{
    //Definitions for MO class RncFunction
    interface RncFunction : GenericNRMDefs::ManagedFunction
    {
        const string CLASS = "RncFunction";

        // including all Attribute Names from
        // MO Class GenericNRMDefs::ManagedFunction
        // additional Attribute Names is as follows.
        //
        const string rncFunctionId = "rncFunctionId";
        const string rncId = "rncId";
        const string mnc = "mnc";
        const string mcc = "mcc";
    };

    //Definitions for MO class Node BFunction
    interface Node BFunction : GenericNRMDefs::ManagedFunction
    {
        const string CLASS = "Node BFunction";

        // including all Attribute Names from
        // MO Class GenericNRMDefs::ManagedFunction
        // additional Attribute Names is as follows.
        //
        const string Node BFunctionId = "Node BFunctionId";
        const string relatedIubLink = "relatedIubLink";
    };

    //Definitions for MO class IubLink
    interface IubLink : GenericNRMDefs::ManagedFunction
    {
        const string CLASS = "IubLink";
    };
};
```

```

// Attribute Names
//
const string iubLinkId = "iubLinkId";
const string relatedNode B = "relatedNode B";
const string relatedUtranCells = "relatedUtranCells";
};

//Definitions for MO class UtranCell
interface UtranCell : GenericNRMDefs::ManagedFunction
{
    const string CLASS = " UtranCell ";

    // Attribute Names
    //
const string utranCellId = "utranCellId";
const string cId= "cId";
const string localCellId= "localCellId";
const string cellMode ="cellMode";
const string maximumTransmissionPower = "maximumTransmissionPower";
const string uarfcn = "uarfcn";
const string cellParameterId = "cellParameterId";
const string primaryCpchPower = "primaryCpchPower";
const string dwPchPower = "dwPchPower";
const string timeSlotList = "timeSlotList";
const string lac = "lac";
const string rac = "rac";
const string sac = "sac";
const string uraList = "uraList";
const string relatedIubLink = "relatedIubLink";
};

//Definitions for MO class UtranRelation
interface UtranRelation : GenericNRMDefs::Top
{
    const string CLASS = "UtranRelation";

    // Attribute Names
    //
const string utranRelationId = "utranRelationId";
const string adjacentCell = "adjacentCell";
const string cellMode ="cellMode";
const string uarfcn = "uarfcn";
const string cellParameterId = "cellParameterId";
const string primaryCpchPower = "primaryCpchPower";

```

```
const string lac = "lac";
const string userLabel = "userLabel" ;
};

//Definitions for MO class ExternalUtranCell
interface ExternalUtranCell : GenericNRMDefs::ManagedFunction
{
    const string CLASS = "ExternalUtranCell";

    // Attribute Names
    //
    const string externalUtranCellId = "externalUtranCellId";
    const string cId = "cId";
    const string mnc = "mnc";
    const string mcc = "mcc";
    const string rncId = "rncId";
    const string cellMode = "cellMode";
    const string uarfcn = "uarfcn";
    const string cellParameterId = "cellParameterId";
    const string primaryCpchPower = "primaryCpchPower";
    const string lac = "lac";
    const string rac = "rac";
};

//Defination for MO class GsmRelation
interface GsmRelation: GenericNRMDefs::Top
{
    const string CLASS = "GsmRelation";

    //Attribute Names
    //
    const string gsmRelationId = "gsmRelationId";
    const string adjacentCell = "adjacentCell";
    const string bcchFrequency = "bcchFrequency";
    const string ncc = "ncc";
    const string bcc = "bcc";
    const string lac = "lac";
    const string userLabel = "userLabel";
};

//Defination for MO ExternalGSMCell
interface ExternalGSMCell: GenericNRMDefs::ManagedFunction
{
    const string CLASS = "ExternalGSMCell";
```

```

//Attribute Names
//
const string externalGsmCellId = "externalGsmCellId";
const string cellIdentity = "cellIdentity";
const string bcchFrequency = "bcchFrequency";
const string ncc = "ncc";
const string bcc = "bcc";
const string lac = "lac";
const string mcc = "mcc";
const string mnc = "mnc";
const string rac = "rac";
const string racc = "racc";
};

//Definitions for MO class Carrier
interface Carrier : GenericNRMDefs::ManagedFunction
{
    const string CLASS = "Carrier";

    // Attribute Names
    //
    const string carrierId = "carrierId";
    const string uarfcnType = "uarfcnType";
    const string uarfcn = "uarfcn";
    const string timeSlotList = "timeSlotList";
    const string operationalState = "operationalState";
    // HSUPA specific attribute(s)
    const string HsupaFlag = "HsupaFlag";
    const string HsupaState = "HsupaState";
    const string nbrHsupaEAGCH = "nbrHsupaEAGCH";
    const string nbrHsupaEHICH = "nbrHsupaEHICH";
};
#endif

```

#### 6.1.2.1.2 UtranNRMPProfile

```

//File "UtranNRMPProfile.idl"
//The IRP document version number is "UTRAN NRM V1.0"
#ifndef UtranNRMPProfile_idl
#define UtranNRMPProfile_idl

#include "GenericNRMSystem.idl"
#include "GenericNRMPProfile.idl"
#include "UtranNRMSystem.idl"

```

```

##pragma prefix "3gppsa5.org"

/**
 * This module defines the attribute names and
 * correspondig attribute types for all defined
 * MO class in Utran network. This module is
 * used for reference.
 */
module UtranNRMProfile
{
  interface RncFunction : GenericNRMProfile::ManagedFunction
  {
    readonly attribute GenericNRMSystem::ObjectIdType mcFunctionId;
    attribute unsigned long mcId;
    readonly attribute unsigned long mnc;
    readonly attribute unsigned long mcc;

    // The following notifications may be sent from this MO,
    // notifyObjectCreation
    // notifyObjectDeletion
    // notifyAttributeValueChange
    // notifyAckStateChanged
    // notifyChangedAlarm
    // notifyClearedAlarm
    // notifyNewAlarm
    // notifyComments
    // notifyAlarmListRebuilt
    // notifyPotentialFaultyAlarmList
  };

  interface Node BFunction : GenericNRMProfile::ManagedFunction
  {
    readonly attribute GenericNRMSystem::ObjectIdType Node BFunctionId;
    readonly attribute GenericNRMSystem::DN relatedIubLink;

    // The following notifications may be sent from this MO,
    // notifyObjectCreation
    // notifyObjectDeletion
    // notifyAttributeValueChange
    // notifyAckStateChanged
    // notifyChangedAlarm
    // notifyClearedAlarm
    // notifyNewAlarm
    // notifyComments
    // notifyAlarmListRebuilt
  }
}

```

```

    // notifyPotentialFaultyAlarmList
};

interface IubLink : GenericNRMSProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType iubLinkId;
        attribute GenericNRMSystem::DN relatedNode B;
        attribute GenericNRMSystem::DNListType relatedUtranCells;

    // The following notifications may be sent from this MO,
    // notifyObjectCreation
    // notifyObjectDeletion
    // notifyAttributeValueChange
    // notifyAckStateChanged
    // notifyChangedAlarm
    // notifyClearedAlarm
    // notifyNewAlarm
    // notifyComments
    // notifyAlarmListRebuilt
    // notifyPotentialFaultyAlarmList
};

interface UtranCell : GenericNRMSProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType utranCellId;
        attribute unsigned long cId;
        attribute unsigned long localCellId;

    readonly attribute UtranNRMSystem::CellModeEnumType cellMode;
    readonly attribute unsigned short maximumTransmissionPower; //0..50Dbm
    readonly attribute unsigned long uarfcn;
        attribute unsigned short cellParameterId;

    readonly attribute short primaryCpchPower;
    readonly attribute short dwPchPower;
        attribute UtranNRMSystem::TimeSlotListConfigStructType timeSlotList;

    readonly attribute unsigned long lac;
    readonly attribute unsigned long rac;
    readonly attribute unsigned long sac;

    readonly attribute UtranNRMSystem::UraListType uraList;
    readonly attribute GenericNRMSystem::DN relatedIubLink;

    // The following notifications may be sent from this MO,
    // notifyObjectCreation
    // notifyObjectDeletion
    // notifyAttributeValueChange
    // notifyAckStateChanged

```



```

    // notifyChangedAlarm
    // notifyClearedAlarm
    // notifyNewAlarm
    // notifyComments
    // notifyAlarmListRebuilt
    // notifyPotentialFaultyAlarmList
};

interface UtranRelation : GenericNRMPProfile::Top
{
    readonly attribute GenericNRMSystem::ObjectIdType utranRelationId;
        attribute UtranNRMSystem::AdjacentCellType adjacentCell;
    readonly attribute UtranNRMSystem::CellModeEnumType cellMode;
    readonly attribute unsigned long uarfcn;    //conditional
    readonly attribute unsigned long cellParameterId;    //conditional
    readonly attribute unsigned short primaryCpchPower;    //conditional
    readonly attribute unsigned long lac;    //conditional
        attribute wstring userLabel;

    // The following notifications may be sent from this MO,
    // notifyObjectCreation
    // notifyObjectDeletion
    // notifyAttributeValueChange
};

interface ExternalUtranCell : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType externalUtranCellId;
        attribute unsigned long cId;
        attribute unsigned long mcc;
        attribute unsigned long mnc;
        attribute unsigned long mcId;
    readonly attribute UtranNRMSystem::CellModeEnumType cellMode;
        attribute unsigned long uarfcn;
        attribute unsigned long cellParameterId;
        attribute unsigned short primaryCpchPower;
        attribute unsigned long lac;
        attribute unsigned long rac;

    // The following notifications may be sent from this MO,
    // notifyObjectCreation
    // notifyObjectDeletion
    // notifyAttributeValueChange
};

```

```

interface GsmRelation: GenericNRMProfile::Top
{
    readonly attribute GenericNRMSystem::ObjectIdType gsmRelationId;
        attribute GenericNRMSystem::DN adjacentCell;
    readonly attribute UtranNRMSystem::AdjacentCellType bcchFrequency;    //conditional
    readonly attribute unsigned long ncc;    //conditional
    readonly attribute unsigned long bcc;    //conditional
    readonly attribute unsigned long lac;    //conditional
        attribute wstring userLabel;

    // The following notifications may be sent from this MO,
    // notifyObjectCreation
    // notifyObjectDeletion
    // notifyAttributeValueChange
};

interface ExternalGSMCell: GenericNRMProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType externalGsmCellId;
        attribute unsigned long cellIdentity;
        attribute short bcchFrequency;
        attribute unsigned long ncc;
        attribute unsigned long bcc;
        attribute unsigned long lac;
        attribute unsigned long mcc;
        attribute unsigned long mnc;
        attribute unsigned long rac;
        attribute unsigned long racc;

    // The following notifications may be sent from this MO,
    // notifyObjectCreation
    // notifyObjectDeletion
    // notifyAttributeValueChange
};

interface Carrier : GenericNRMProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType carrierId;
    readonly attribute UtranNRMSystem::uarfcnType uarfcnType;
    readonly attribute unsigned long uarfcn;
        attribute UtranNRMSystem::TimeSlotListConfigStructType timeSlotList;
    readonly attribute GenericNRMSystem::OperationalStateType operationalState;
    // HSUPA specific attribute(s)
    readonly attribute UtranNRMSystem::HsupaFlagType HsupaFlag;
    readonly attribute UtranNRMSystem::HsupaStateType HsupaState;
}

```

```

    readonly attribute unsigned long nbrHsupaEAGCH;
    readonly attribute unsigned long nbrHsupaEHICH;

    // The following notifications may be sent from this MO,
    // notifyObjectCreation
    // notifyObjectDeletion
    // notifyAttributeValueChange
    // notifyAckStateChanged
    // notifyChangedAlarm
    // notifyClearedAlarm
    // notifyNewAlarm
    // notifyComments
    // notifyStateChange
};
};
#endif

```

### 6.1.2.1.3 UtranNRMSystem

```

//File "UtranNRMSystem.idl"
//The IRP document version number is "UTRAN NRM V1.0"
#ifndef UtranNRMSystem_idl
#define UtranNRMSystem_idl

#include "GenericNRMSystem.idl"

module UtranNRMSystem
{
    /**
     * This module adds datatype definitions for types
     * used in the Utran NRM which are not basic datatypes defined
     * already in CORBA and datatypes defined already in
     * GenericNRMSystem.
     */

    union AdjacentCellType switch(boolean)
    {
        case TRUE: GenericNRMSystem::DN utranCell;
        case FALSE: string cellGloableId;
    };
    enum CellModeEnumType
    {
        FDD_mode,
        TDD_mode_1_28Mcps,
        TDD_mode_3_84Mcps
    }

```

```

};
enum uarfcnType
{
    MainCarrier, //0 主载波
    SubCarrier //1 辅载波
};
enum TimeSlotDirectionType
{
    UL,
    DL
};
enum TimeSlotStatusType
{
    Activated,
    Deactivated
};
struct TimeSlotConfigStructType
{
    unsigned short timeSlotId;
    TimeSlotDirectionType timeSlotDirection;
    TimeSlotStatusType timeSlotStatus;
};
enum HsupaFlagType
{
    Supported,
    Unsupported
};
enum HsupaStateType
{
    Activated,
    Deactivated
};
typedef sequence<TimeSlotConfigStructType> TimeSlotListConfigStructType;
typedef sequence<unsigned long> UraListType;
};
#endif

```

## 6.2 性能网络资源模型设计

### 6.2.1 性能管理资源模型的 IDL 定义

#### 6.2.1.1 TDHsupaMeasurementDefs

```

//File "TDHsupaMeasurementDefs.idl"
#ifndef TDHsupaMeasurementDefs_idl
#define TDHsupaMeasurementDefs_idl

```

```

// #pragma prefix "3gppsa5.org"

/**
 * This module defines measurementType names constants
 */
module TDHsupaMeasurementDefs
{
    //for RNC measurement
    module rabAssignmentMeasurement
    {
        //assignment RAB
        const string attRabAssignEstabCsPerType= "attRabAssignEstabCsPerType";
        const string succRabAssignEstabCsPerType= "succRabAssignEstabCsPerType";
        const string failRabAssignEstabCsPerCause= "failRabAssignEstabCsPerCause";
        const string attRabAssignEstabPsPerType= "attRabAssignEstabPsPerType";
        const string succRabAssignEstabPsPerType= "succRabAssignEstabPsPerType";
        const string failRabAssignEstabPsPerCause= "failRabAssignEstabPsPerCause";
    };
    module rabReleaseRequestMeasurement
    {
        const string nbrRncRelCsRabPerCause= "nbrRncRelCsRabPerCause";
        const string nbrRncRelPsRabPerCause= "nbrRncRelPsRabPerCause";
    };
    module iuConnectionMeasurement
    {
        //establish Iu connection
        const string attRncEstabCsIuConn= "attRncEstabCsIuConn";
        const string attRncEstabPsIuConn= "attRncEstabPsIuConn";
        //request to release Iu connection
        const string nbrRncRelCsIuConnPerCause= "nbrRncRelCsIuConnPerCause";
        const string nbrRncRelPsIuConnPerCause= "nbrRncRelPsIuConnPerCause";
        //release Iu connection
        const string attRelCsIuConnPerCause= "attRelCsIuConnPerCause";
        const string attRelPsIuConnPerCause= "attRelPsIuConnPerCause";
    };
    module iuInterfaceMeasurement
    {
        //Iu interface reset
        const string nbrResetCsByRncPerCause= "nbrResetCsByRncPerCause";
        const string nbrResetPsByRncPerCause= "nbrResetPsByRncPerCause";
        const string nbrResetCsByCnPerCause= "nbrResetCsByCnPerCause";
        const string nbrResetPsByCnPerCause= "nbrResetPsByCnPerCause";
        //Iu interface reset resource
        const string nbrResetResCsByRncPerCause= "nbrResetResCsByRncPerCause";
        const string nbrResetResPsByRncPerCause= "nbrResetResPsByRncPerCause";
    };
}

```

```

const string nbrResetResCsByCnPerCause= "nbrResetResCsByCnPerCause";
const string nbrResetResPsByCnPerCause= "nbrResetResPsByCnPerCause";
//lu interface overload control
const string nbrOverloadControlCsByRnc= "nbrOverloadControlCsByRnc";
const string nbrOverloadControlPsByRnc= "nbrOverloadControlPsByRnc";
const string nbrOverloadControlCsByCn= "nbrOverloadControlCsByCn";
const string nbrOverloadControlPsByCn= "nbrOverloadControlPsByCn";
//lu interface error indication
const string nbrErrorIndCsByRncPerCause= "nbrErrorIndCsByRncPerCause";
const string nbrErrorIndPsByRncPerCause= "nbrErrorIndPsByRncPerCause";
const string nbrErrorIndCsByCnPerCause= "nbrErrorIndCsByCnPerCause";
const string nbrErrorIndPsByCnPerCause= "nbrErrorIndPsByCnPerCause";
};
module mcHardHandoverMeasurement
{
    const string attHho= "attHho";
    const string failHhoPerCause= "failHhoPerCause";
};
module mcRelocationMeasurement
{
    const string relocAttPrep= "relocAttPrep";
    const string relocSuccPrep= "relocSuccPrep";
    const string relocFailPrepCause= "relocFailPrepCause";
    const string relocSucc= "relocSucc";
};
module mcInterSystemHandoverMeasurement
{
    //relocation in RAT
    const string attRelocPrepOutRATHOCSPerCause= "attRelocPrepOutRATHOCSPerCause";
    const string failRelocPrepOutRATHOCSPerCause= "failRelocPrepOutRATHOCSPerCause";
    const string succRelocPrepOutRATHOCS= "succRelocPrepOutRATHOCS";
    //CS inter system handover from 3G to 2G
    const string iRATHOAttOutCS= "iRATHOAttOutCS";
    const string iRATHOFailOutCSCause= "iRATHOFailOutCSCause";
    const string iRATHOSuccOutCS= "iRATHOSuccOutCS";
    //CS inter system handover from 2G to 3G
    const string iRATHOAttIncCS= "iRATHOAttIncCS";
    const string iRATHOFailIncCSCause= "iRATHOFailIncCSCause";
    const string iRATHOSuccIncCS= "iRATHOSuccIncCS";
    //PS inter system handover from 3G to 2G
    const string iRATHOAttOutPSUTRAN= "iRATHOAttOutPSUTRAN";
    const string iRATHOFailOutPSUTRANCause= "iRATHOFailOutPSUTRANCause";
    const string iRATHOSuccOutPSUTRAN= "iRATHOSuccOutPSUTRAN";
    const string iRATHOSuccOutPSUE= "iRATHOSuccOutPSUE";
    //PS inter system handover from 2G to 3G

```

```

const string iRATHOAttIncPS= "iRATHOAttIncPS";
const string iRATHOSuccIncPS= "iRATHOSuccIncPS";
};
//No.7 signalling measurement
module signallingPointTP
{
const string nbrUsrUnavailRx= "nbrUsrUnavailRx";
const string nbrUsrUnavailTx= "nbrUsrUnavailTx";
const string nbrTraTx= "nbrTraTx";
const string nbrTraRx= "nbrTraRx";
const string nbrMsuDropRteErr= "nbrMsuDropRteErr";
};
module signallingLinkTP
{
const string nbrChangeOverTx= "nbrChangeOverTx";
const string nbrChangeOverRx= "nbrChangeOverRx";
const string nbrChangeOverAckTx= "nbrChangeOverAckTx";
const string nbrChangeOverAckRx= "nbrChangeOverAckRx";
const string nbrChangeBackTx= "nbrChangeBackTx";
const string nbrChangeBackRx= "nbrChangeBackRx";
const string nbrChangeBackAckTx= "nbrChangeBackAckTx";
const string nbrChangeBackAckRx= "nbrChangeBackAckRx";
const string nbrLnkInhDenTx= "nbrLnkInhDenTx";
const string nbrLnkInhDenRx= "nbrLnkInhDenRx";
const string nbrLnkForceUninhTx= "nbrLnkForceUninhTx";
const string nbrLnkForceUninhRx= "nbrLnkForceUninhRx";
const string nbrLnkLocInhTstTx= "nbrLnkLocInhTstTx";
const string nbrLnkLocInhTstRx= "nbrLnkLocInhTstRx";
const string nbrLnkRmtInhTstTx= "nbrLnkRmtInhTstTx";
const string nbrLnkRmtInhTstRx= "nbrLnkRmtInhTstRx";
const string nbrLnkConOrdTx= "nbrLnkConOrdTx";
const string nbrLnkConOrdRx= "nbrLnkConOrdRx";
const string nbrLnkConAckTx= "nbrLnkConAckTx";
const string nbrLnkConAckRx= "nbrLnkConAckRx";
const string nbrLnkTstRx= "nbrLnkTstRx";
const string nbrLnkTstTx= "nbrLnkTstTx";
const string nbrLnkTstAckRx= "nbrLnkTstAckRx";
const string nbrLnkTstAckTx= "nbrLnkTstAckTx";
const string nbrTxDrop= "nbrTxDrop";
const string nbrTxCongDrop= "nbrTxCongDrop";
const string nbrSifOctTx= "nbrSifOctTx";
const string nbrSifOctRx= "nbrSifOctRx";
const string nbrSioOctTx= "nbrSioOctTx";
const string nbrSioOctRx= "nbrSioOctRx";
const string nbrMsuTx= "nbrMsuTx";

```

```

const string nbrMsuRx= "nbrMsuRx";
const string nbrCong1= "nbrCong1";
const string nbrCong2= "nbrCong2";
const string nbrCong3= "nbrCong3";
const string durSigLinkOutOfService= "durSigLinkOutOfService";
const string durLnkCong= "durLnkCong";
const string nbrLnkErrPduRx= "nbrLnkErrPduRx";
};
module signallingLinkSetTP
{
    const string durLnkSetUnav= "durLnkSetUnav";
};

//for UtranCell measurement
module cellRrcConnectionMeasurement
{
    const string rrcAttConnEstabCause= "rrcAttConnEstabCause";
    const string rrcFailConnEstabCause= "rrcFailConnEstabCause";
    const string rrcSuccConnEstabCause= "rrcSuccConnEstabCause";
    const string rrcAttConnReEstab= "rrcAttConnReEstab";
    const string rrcFailConnReEstabCause= "rrcFailConnReEstabCause";
    const string rrcSuccConnReEstab= "rrcSuccConnReEstab";
    const string rrcAttConnRelDCCHCause= "rrcAttConnRelDCCHCause";
    const string rrcAttConnRelCCCHCause= "rrcAttConnRelCCCHCause";
};
module cellHsupaMeasurement
{
    //MAC-d Flow establish
    const string HsupaAttMacdSetup= "HsupaAttMacdSetup";
    const string HsupaSuccMacdSetup= "HsupaSuccMacdSetup";
    const string HsupaFailMacdSetupCause= "HsupaFailMacdSetupCause";
    //HSUPA RB establish
    const string HsupaAttRBSetup= "HsupaAttRBSetup";
    const string HsupaSuccRBSetup= "HsupaSuccRBSetup";
    const string HsupaFailRBSetupCause= "HsupaFailRBSetupCause";
    //E-DCH service cell change on cell
    const string HsupaAttOutCellChange= "HsupaAttOutCellChange";
    const string HsupaSuccOutCellChange= "HsupaSuccOutCellChange";
    //E-DCH handover intracell
    const string HsupaAttRachToEdchIntraCell= "HsupaAttRachToEdchIntraCell";
    const string HsupaSuccRachToEdchIntraCell= "HsupaSuccRachToEdchIntraCell";
    const string HsupaAttDchToEdchIntraCell = "HsupaAttDchToEdchIntraCell ";
    const string HsupaSuccDchToEdchIntraCell= "HsupaSuccDchToEdchIntraCell";
    const string HsupaAttEdchToRachIntraCell= "HsupaAttEdchToRachIntraCell";
    const string HsupaSuccEdchToRachIntraCell= "HsupaSuccEdchToRachIntraCell";
};

```



```

const string HsupaAttEdchToDchIntraCell= "HsupaAttEdchToDchIntraCell";
const string HsupaSuccEdchToDchIntraCell= "HsupaSuccEdchToDchIntraCell";
//E-DCH handover intercell
const string HsupaAttRachToEdchInterCell= "HsupaAttRachToEdchInterCell";
const string HsupaSuccRachToEdchInterCell= "HsupaSuccRachToEdchInterCell";
const string HsupaAttDchToEdchInterCell = "HsupaAttDchToEdchInterCell ";
const string HsupaSuccDchToEdchInterCell= "HsupaSuccDchToEdchInterCell";
const string HsupaAttEdchToRachInterCell= "HsupaAttEdchToRachInterCell";
const string HsupaSuccEdchToRachInterCell= "HsupaSuccEdchToRachInterCell";
const string HsupaAttEdchToDchInterCell= "HsupaAttEdchToDchInterCell";
const string HsupaSuccEdchToDchInterCell= "HsupaSuccEdchToDchInterCell";
//HSUPA cell resource
const string HsupaMeanNbrUser= "HsupaMeanNbrUser";
//integrality
const string HsupaNbrMacePdu= "HsupaNbrMacePdu";
const string HsupaNbrAckdMacePdu= "HsupaNbrAckdMacePdu";
const string HsupaNbrAckdMaceOcts= "HsupaNbrAckdMaceOcts";
//abnormal E-DCH release
const string HsupaSuccEdchReleaseUserInact= "HsupaSuccEdchReleaseUserInact";
const string HsupaSuccCnInitEdchRelease= "HsupaSuccCnInitEdchRelease";
//abnormal E-DCH release
const string HsupaAbnormalEdchRelease= "HsupaAbnormalEdchRelease";
};

//for UtranRelation measurement
module cellRelationHsupaMeasurement
{
    //E-DCH service cell change on cellrelation
    const string HsupaAttOut= "HsupaAttOut";
    const string HsupaSuccOut= "HsupaSuccOut";
};

module hardHandoverInterCellIntraNode BMeasurement
{
    const string hHOAttOutIntraNode B= "hHOAttOutIntraNode B";
    const string hHOFailOutIntraNode BCause= "hHOFailOutIntraNode BCause";
    const string hHOSuccOutIntraNode B= "hHOSuccOutIntraNode B";
};

module hardHandoverInterNode BIntraRncMeasurement
{
    const string hHOAttOutInterNode BIntraRNC= "hHOAttOutInterNode BIntraRNC";
    const string hHOFailOutInterNode BIntraRNCCause= "hHOFailOutInterNode BIntraRNCCause";
    const string hHOSuccOutInterNode BIntraRNC= "hHOSuccOutInterNode BIntraRNC";
};

module hardHandoverInterRncViaIurMeasurement
{

```

```

const string hHOAttOutInterRNCIur= "hHOAttOutInterRNCIur";
const string hHOSuccOutInterRNCIur= "hHOSuccOutInterRNCIur";
const string hHOFailOutInterRNCIurCause= "hHOFailOutInterRNCIurCause";
};
module hardHandoverInterRncMeasurement
{
const string attHhoOutInterRncCn= "attHhoOutInterRncCn";
const string failHhoOutInterRncCnPerCause= "failHhoOutInterRncCnPerCause";
};
module hardHandoverInterSystemMeasurement
{
//relocation in RAT
const string attRelocPrepOutRATHOCSPerCause= "attRelocPrepOutRATHOCSPerCause";
const string failRelocPrepOutRATHOCSPerCause= "failRelocPrepOutRATHOCSPerCause";
const string succRelocPrepOutRATHOCS = "succRelocPrepOutRATHOCS";
//CS inter system handover from 3G to 2G
const string iRATHOAttOutCS= "iRATHOAttOutCS";
const string iRATHOFailOutCSCause= "iRATHOFailOutCSCause";
const string iRATHOSuccOutCS= "iRATHOSuccOutCS";
//CS inter system handover from 2G to 3G
const string iRATHOAttIncCS= "iRATHOAttIncCS";
const string iRATHOFailIncCSCause= "iRATHOFailIncCSCause";
const string iRATHOSuccIncCS= "iRATHOSuccIncCS";
//PS inter system handover from 3G to 2G
const string iRATHOAttOutPSUTRAN= "iRATHOAttOutPSUTRAN";
const string iRATHOFailOutPSUTRANCause= "iRATHOFailOutPSUTRANCause";
const string iRATHOSuccOutPSUTRAN= "iRATHOSuccOutPSUTRAN";
const string iRATHOSuccOutPSUE= "iRATHOSuccOutPSUE";
//PS inter system handover from 2G to 3G
const string iRATHOAttIncPS= "iRATHOAttIncPS";
const string iRATHOSuccIncPS= "iRATHOSuccIncPS";
};
};
#endif

```

## 6.2.2 数据类型的IDL定义

### 6.2.2.1 TDSCDMANRMMeasurementSystem

```

//File "TDSCDMANRMMeasurementSystem.idl"
#ifndef TDSCDMANRMMeasurementSystem_idl
#define TDSCDMANRMMeasurementSystem_idl

// #pragma prefix "3gppsa5.org"

/**
* This module defines type definitions for performance measurements

```

```
*/
module TDSCDMANRMMMeasurementSystem
{

typedef unsigned long TDSCDMAMeasurementType1;
typedef float TDSCDMAMeasurementType2;

// The following RANAP causes are defined in the section 9.2.1.4 of 3GPP TS 25.413 v5.5.0.
typedef unsigned short RANAPCause;

//Radio Network Layer Cause. Value range is 1 - 64.
const RANAPCause rabPreempted = 1;
const RANAPCause trelocoverallExpiry = 2;
const RANAPCause trelocprepExpiry = 3;
const RANAPCause treloccompleteExpiry = 4;
const RANAPCause tqueingExpiry = 5;
const RANAPCause relocationTriggered = 6;
const RANAPCause trelocallocExpiry = 7;
const RANAPCause unableToEstablishDuringRelocation = 8;
const RANAPCause unknownTargetRnc = 9;
const RANAPCause relocationCancelled = 10;
const RANAPCause successfulRelocation = 11;
const RANAPCause requestedCipheringAndOrIntegrityProtectionAlgorithmsNotSupported = 12;
const RANAPCause conflictWithAlreadyExistingIntegrityProtectionAndOrCipheringInformation = 13;
const RANAPCause failureInTheRadioInterfaceProcedure = 14;
const RANAPCause releaseDueToUtranGeneratedReason = 15;
const RANAPCause userInactivity_RANAP = 16;
const RANAPCause timeCriticalRelocation = 17;
const RANAPCause requestedTrafficClassNotAvailable = 18;
const RANAPCause invalidRABParametersValue = 19;
const RANAPCause requestedMaximumBitRateNotAvailable = 20;
const RANAPCause requestedGuaranteedBitRateNotAvailable = 21;
const RANAPCause requestedTransferDelayNotAchievable = 22;
const RANAPCause invalidRabParametersCombination = 23;
const RANAPCause conditionViolationForSduParameters = 24;
const RANAPCause conditionViolationForTrafficHandlingPriority = 25;
const RANAPCause conditionViolationForGuaranteedBitRate = 26;
const RANAPCause userPlaneVersionsNotSupported = 27;
const RANAPCause iuUpFailure = 28;
const RANAPCause relocationFailureInTargetCnRncOrTargetSystem = 29;
const RANAPCause invalidRabId = 30;
const RANAPCause noRemainingRab = 31;
const RANAPCause interactionWithOtherProcedure = 32;
const RANAPCause requestedMaximumBitRateForDINotAvailable = 33;
const RANAPCause requestedMaximumBitRateForUINotAvailable = 34;
```

```
const RANAPCause requestedGuaranteedBitRateForDINotAvailable = 35;
const RANAPCause requestedGuaranteedBitRateForUINotAvailable = 36;
const RANAPCause repeatedIntegrityCheckingFailure = 37;
const RANAPCause requestedRequestTypeNotSupported = 38;
const RANAPCause requestSuperseded = 39;
const RANAPCause releaseDueToUeGeneratedSignallingConnectionRelease = 40;
const RANAPCause resourceOptimisationRelocation = 41;
const RANAPCause requestedInformationNotAvailable = 42;
const RANAPCause relocationDesirableForRadioReasons = 43;
const RANAPCause relocationNotSupportedInTargetRncOrTargetSystem = 44;
const RANAPCause directedRetry = 45;
const RANAPCause radioConnectionWithUeLost = 46;
const RANAPCause rncUnableToEstablishAllRfcs = 47;
const RANAPCause decipheringKeysNotAvailable = 48;
const RANAPCause dedicatedAssistanceDataNotAvailable = 49;
const RANAPCause relocationTargetNotAllowed = 50;
const RANAPCause locationReportingCongestion = 51;
const RANAPCause reduceLoadInServingCell = 52;
const RANAPCause noRadioResourcesAvailableInTargetCell = 53;
const RANAPCause geranIuModeFailure = 54;
const RANAPCause accessRestrictedDueToSharedNetworks = 55;
const RANAPCause incomingRelocationNotSupportedDueToPuesbineFeature = 56;
//Transport Layer Cause. Value range is 65 - 80.
const RANAPCause signallingTransportResourceFailure = 65;
const RANAPCause iuTransportConnectionFailedToEstablish = 66;

//NAS Cause. Value range is 81 - 96.
const RANAPCause userRestrictionStartIndication = 81;
const RANAPCause userRestrictionEndIndication = 82;
const RANAPCause normalRelease = 83;

//Protocol Cause. Value range is 97 - 112.
const RANAPCause transferSyntaxError_RANAP = 97;
const RANAPCause semanticError_RANAP = 98;
const RANAPCause messageNotCompatibleWithReceiverState_RANAP = 99;
const RANAPCause abstractSyntaxErrorReject_RANAP = 100;
const RANAPCause abstractSyntaxErrorIgnoreAndNotify_RANAP = 101;
const RANAPCause abstractSyntaxErrorFalselyConstructedMessage_RANAP = 102;

//Miscellaneous Cause. Value range is 113 - 128.
const RANAPCause operationAndMaintenanceIntervention_RANAP = 113;
const RANAPCause noResourceAvailable = 114;
const RANAPCause unspecifiedFailure = 115;
const RANAPCause networkOptimisation = 116;
```

```
//Non-standard Cause. Value range is 129 - 256. Cause value 256 shall not be used.

// The following RNSAP causes are defined in the section 9.2.1.5 of 3GPP TS 25.423 v5.6.0.
typedef unsigned short RNSAPCause;

//Radio Network Layer Cause.
const RNSAPCause unknownCid_RNSAP = 1;
const RNSAPCause cellNotAvailable_RNSAP = 2;
const RNSAPCause powerLevelNotSupported_RNSAP = 3;
const RNSAPCause ulScramblingCodeAlreadyInUse = 4;
const RNSAPCause dlRadioResourcesNotAvailable_RNSAP = 5;
const RNSAPCause ulRadioResourcesNotAvailable_RNSAP = 6;
const RNSAPCause measurementNotSupportedForTheObject_RNSAP = 7;
const RNSAPCause combiningResourcesNotAvailable_RNSAP = 8;
const RNSAPCause combiningNotSupported_RNSAP = 9;
const RNSAPCause reconfigurationNotAllowed = 10;
const RNSAPCause requestedConfigurationNotSupported_RNSAP = 11;
const RNSAPCause synchronisationFailure = 12;
const RNSAPCause requestedTxDiversityModeNotSupported_RNSAP = 13;
const RNSAPCause measurementTemporarilyNotAvailable_RNSAP = 14;
const RNSAPCause unspecified_RNL_RNSAP = 15;
const RNSAPCause invalidCmSettings = 16;
const RNSAPCause reconfigurationCfmNotElapsed_RNSAP = 17;
const RNSAPCause numberOfDLCodesNotSupported_RNSAP = 18;
const RNSAPCause dedicatedTransportChannelTypeNotSupported_RNSAP = 19;
const RNSAPCause dlSharedChannelTypeNotSupported = 20;
const RNSAPCause ulSharedChannelTypeNotSupported = 21;
const RNSAPCause commonTransportChannelTypeNotSupported_RNSAP = 22;
const RNSAPCause ulSpreadingFactorNotSupported = 23;
const RNSAPCause dlSpreadingFactorNotSupported = 24;
const RNSAPCause cmNotSupported_RNSAP = 25;
const RNSAPCause transactionNotSupportedByDestinationNode B = 26;
const RNSAPCause rlAlreadyActivatedAllocated_RNSAP = 27;
const RNSAPCause numberOfULCodesNotSupported_RNSAP = 28;
const RNSAPCause cellReservedForOperatorUse = 29;
const RNSAPCause dpcModeChangeNotSupported_RNSAP = 30;
const RNSAPCause informationTemporarilyNotAvailable_RNSAP = 31;
const RNSAPCause informationProvisionNotSupportedForTheObject_RNSAP = 32;
const RNSAPCause powerBalancingStatusNotCompatible_RNSAP = 33;
const RNSAPCause delayedActivationNotSupported_RNSAP = 34;
const RNSAPCause rlTimingAdjustmentNotSupported_RNSAP = 35;
const RNSAPCause unknownRnti = 36;

//Transport Layer Cause.
```

```
const RNSAPCause transportResourceUnavailable_RNSAP = 37;
const RNSAPCause unspecified_TL_RNSAP = 38;

//Protocol Cause.
const RNSAPCause transferSyntaxError_RNSAP = 39;
const RNSAPCause abstractSyntaxErrorReject_RNSAP = 40;
const RNSAPCause abstractSyntaxErrorIgnoreAndNotify_RNSAP = 41;
const RNSAPCause messageNotCompatibleWithReceiverState_RNSAP = 42;
const RNSAPCause semanticError_RNSAP = 43;
const RNSAPCause unspecified_Protocol_RNSAP = 44;
const RNSAPCause abstractSyntaxErrorFalselyConstructedMessage_RNSAP = 45;

//Miscellaneous Cause.
const RNSAPCause controlProcessingOverload_RNSAP = 46;
const RNSAPCause hardwareFailure_RNSAP = 47;
const RNSAPCause operationAndMaintenanceIntervention_RNSAP = 48;
const RNSAPCause notEnoughUserPlaneProcessingResources_RNSAP = 49;
const RNSAPCause Unspecified_Misc_RNSAP = 50;

// The following NBAP causes are defined in the section 9.2.1.6 of 3GPP TS 25.433 v5.5.0.
typedef unsigned short NBAPCause;

//Radio Network Layer Cause.
const NBAPCause unknownCid_NBAP = 1;
const NBAPCause cellNotAvailable_NBAP = 2;
const NBAPCause powerLevelNotSupported_NBAP = 3;
const NBAPCause dlRadioResourcesNotAvailable_NBAP = 4;
const NBAPCause ulRadioResourcesNotAvailable_NBAP = 5;
const NBAPCause rlAlreadyActivatedAllocated_NBAP = 6;
const NBAPCause NodeBResourcesUnavailable = 7;
const NBAPCause measurementNotSupportedForTheObject_NBAP = 8;
const NBAPCause combiningResourcesNotAvailable_NBAP = 9;
const NBAPCause requestedConfigurationNotSupported_NBAP = 10;
const NBAPCause synchronizationFailure = 11;
const NBAPCause priorityTransportChannelEstablished = 12;
const NBAPCause sibOriginationInNodeBNotSupported = 13;
const NBAPCause requestedTxDiversityModeNotSupported_NBAP = 14;
const NBAPCause unspecified_RNL_NBAP = 15;
const NBAPCause bcchSchedulingError = 16;
const NBAPCause measurementTemporarilyNotAvailable_NBAP = 17;
const NBAPCause invalidCmSetting = 18;
const NBAPCause reconfigurationCfnNotElapsed_NBAP = 19;
const NBAPCause numberOfDlCodesNotSupported_NBAP = 20;
const NBAPCause scpichNotSupported = 21;
const NBAPCause combiningNotSupported_NBAP = 22;
```

```
const NBAPCause ulSfNotSupported = 23;
const NBAPCause dlSfNotSupported = 24;
const NBAPCause commonTransportChannelTypeNotSupported_NBAP = 25;
const NBAPCause dedicatedTransportChannelTypeNotSupported_NBAP = 26;
const NBAPCause downlinkSharedChannelTypeNotSupported = 27;
const NBAPCause uplinkSharedChannelTypeNotSupported = 28;
const NBAPCause cmNotSupported_NBAP = 29;
const NBAPCause txDiversityNoLongerSupported = 30;
const NBAPCause unknownLocalCellId = 31;
const NBAPCause numberOfUICodesNotSupported_NBAP = 32;
const NBAPCause informationTemporarilyNotAvailable_NBAP = 33;
const NBAPCause informationProvisionNotSupportedForTheObject_NBAP = 34;
const NBAPCause cellSynchronisationNotSupported = 35;
const NBAPCause cellSynchronisationAdjustmentNotSupported = 36;
const NBAPCause dpcModeChangeNotSupported_NBAP = 37;
const NBAPCause ipdlAlreadyActivated = 38;
const NBAPCause ipdlNotSupported = 39;
const NBAPCause ipdlParametersNotAvailable = 40;
const NBAPCause frequencyAcquisitionNotSupported = 41;
const NBAPCause powerBalancingStatusNotCompatible_NBAP = 42;
const NBAPCause requestedTypeOfBearerRearrangementNotSupported = 43;
const NBAPCause signallingBearerRearrangementNotSupported = 44;
const NBAPCause bearerRearrangementNeeded = 45;
const NBAPCause delayedActivationNotSupported_NBAP = 46;
const NBAPCause riTimingAdjustmentNotSupported_NBAP = 47;

//Transport Layer Cause.
const NBAPCause transportResourceUnavailable_NBAP = 48;
const NBAPCause unspecified_TL_NBAP = 49;

//Protocol Cause.
const NBAPCause transferSyntaxError_NBAP = 50;
const NBAPCause abstractSyntaxErrorReject_NBAP = 51;
const NBAPCause abstractSyntaxErrorIgnoreAndNotify_NBAP = 52;
const NBAPCause messageNotCompatibleWithReceiverState_NBAP = 53;
const NBAPCause semanticError_NBAP = 54;
const NBAPCause Unspecified_Protocol_NBAP = 55;
const NBAPCause abstractSyntaxErrorFalselyConstructedMessage_NBAP = 56;

//Miscellaneous Cause.
const NBAPCause controlProcessingOverload_NBAP = 57;
const NBAPCause hardwareFailure_NBAP = 58;
const NBAPCause operationAndMaintenanceIntervention_NBAP = 59;
const NBAPCause notEnoughUserPlaneProcessingResources_NBAP = 60;
const NBAPCause unspecified_Misc_NBAP = 61;
```

```
// The following cell update causes are defined in the section 10.3.3.3 of 3GPP TS 25.331 v5.5.0.
```

```
typedef unsigned short CellUpdateCause;  
const CellUpdateCause cellReselection = 1;  
const CellUpdateCause periodicalCellUpdate = 2;  
const CellUpdateCause uplinkDataTransmission = 3;  
const CellUpdateCause pagingResponse = 4;  
const CellUpdateCause reenteredServiceArea = 5;  
const CellUpdateCause radioLinkFailure = 6;  
const CellUpdateCause rlcUnrecoverableError = 7;
```

```
// The following establishment causes are defined in the section 10.3.3.11 of 3GPP TS 25.331 v5.5.0.
```

```
typedef unsigned short EstablishmentCause;  
const EstablishmentCause originatingConversationalCall = 1;  
const EstablishmentCause originatingStreamingCall = 2;  
const EstablishmentCause originatingInteractiveCall = 3;  
const EstablishmentCause originatingBackgroundCall = 4;  
const EstablishmentCause originatingSubscribedTrafficCall = 5;  
const EstablishmentCause terminatingConversationalCall = 6;  
const EstablishmentCause terminatingStreamingCall = 7;  
const EstablishmentCause terminatingInteractiveCall = 8;  
const EstablishmentCause terminatingBackgroundCall = 9;  
const EstablishmentCause emergencyCall = 10;  
const EstablishmentCause interRatCellReselection = 11;  
const EstablishmentCause interRatCellChangeOrder = 12;  
const EstablishmentCause registration = 13;  
const EstablishmentCause detach = 14;  
const EstablishmentCause originatingHighPrioritySignalling = 15;  
const EstablishmentCause originatingLowPrioritySignalling = 16;  
const EstablishmentCause callReestablishment = 17;  
const EstablishmentCause terminatingHighPrioritySignalling = 18;  
const EstablishmentCause terminatingLowPrioritySignalling = 19;  
const EstablishmentCause terminatingCauseUnknown = 20;
```

```
// The following failure causes are defined in the section 10.3.3.13 of 3GPP TS 25.331 v5.5.0.
```

```
typedef unsigned short FailureCause;  
const FailureCause configurationUnsupported = 1;  
const FailureCause physicalChannelFailure_Failure = 2;  
const FailureCause incompatibleSimultaneousReconfiguration = 3;  
const FailureCause protocolError_Failure = 4;  
const FailureCause compressedModeRuntimeError = 5;  
const FailureCause cellUpdateOccurred = 6;  
const FailureCause invalidConfiguration = 7;  
const FailureCause configurationIncomplete = 8;  
const FailureCause unsupportedMeasurement = 9;
```



```
// The following rejection causes are defined in the section 10.3.3.31 of 3GPP TS 25.331 v5.5.0.
typedef unsigned short RejectionCause;
const RejectionCause congestion_Reject = 1;
const RejectionCause unspecified_Reject = 2;

// The following release causes are defined in the section 10.3.3.32 of 3GPP TS 25.331 v5.5.0.
typedef unsigned short ReleaseCause;
const ReleaseCause normalEvent = 1;
const ReleaseCause preemptiveRelease = 2;
const ReleaseCause congestion_Release = 3;
const ReleaseCause reestablishmentReject = 4;
const ReleaseCause userInactivity_Release = 5;
const ReleaseCause directedSignallingConnectionReestablishment = 6;
const ReleaseCause unspecified_Release = 7;

// The following inter-RAT change failure causes are defined in the section 10.3.8.5 of 3GPP TS 25.331 v5.5.0.
typedef unsigned short InterRatChangeFailureCause;
const InterRatChangeFailureCause configurationUnacceptable_IRATChange = 1;
const InterRatChangeFailureCause physicalChannelFailure_IRATChange = 2;
const InterRatChangeFailureCause protocolError_IRATChange = 3;
const InterRatChangeFailureCause unspecified_IRATChange = 4;

// The following inter-RAT handover failure causes are defined in the section 10.3.8.6 of 3GPP TS 25.331 v5.5.0.
typedef unsigned short InterRatHandoverFailureCause;
const InterRatHandoverFailureCause configurationUnacceptable_IRATHo = 1;
const InterRatHandoverFailureCause physicalChannelFailure_IRATHo = 2;
const InterRatHandoverFailureCause protocolError_IRATHo = 3;
const InterRatHandoverFailureCause interRatProtocolError = 4;
const InterRatHandoverFailureCause unspecified_IRATHo = 5;

//The following call failure causes are used in the category "mobileTrafficFlow".
typedef unsigned short CallFailureCause;
const CallFailureCause callingPartAuthFail = 1;
const CallFailureCause callingPartCipherModeFail = 2;
const CallFailureCause interfaceABusy = 3;
const CallFailureCause callingPartAssignFail = 4;
const CallFailureCause exchangeCongestion = 5;
const CallFailureCause userEarlyRelease = 6;
const CallFailureCause calledPartAssignFail = 7;
const CallFailureCause calledPartDetermineBusy = 8;
const CallFailureCause userUnreachable = 9;
const CallFailureCause alertingEarlyRelease = 10;
const CallFailureCause outCircuitOverflow = 11;
```

```

const CallFailureCause calledPartBusy = 12;
const CallFailureCause noAnswer = 13;

//The following Imsi attach failure causes are defined in the section 10.5.3.6 of 3GPP TS 24.008 v6.1.0.
typedef unsigned short ImsiAttachFailureCause;
const ImsiAttachFailureCause imsiUnknownInHLR_Imsi = 2;
const ImsiAttachFailureCause illegalMS_Imsi = 3;
const ImsiAttachFailureCause imsiUnknownInVLR = 4;
const ImsiAttachFailureCause imeiNotAccepted = 5;
const ImsiAttachFailureCause illegalME_Imsi = 6;
const ImsiAttachFailureCause plmnNotAllowed_Imsi = 11;
const ImsiAttachFailureCause locationAreaNotAllowed_Imsi = 12;
const ImsiAttachFailureCause roamingNotAllowedInThisLocationArea_Imsi = 13;
const ImsiAttachFailureCause noSuitableCellsInLocationArea_Imsi = 15;
const ImsiAttachFailureCause networkFailure_Imsi = 17;
const ImsiAttachFailureCause macFailure_Imsi = 20;
const ImsiAttachFailureCause synchFailure_Imsi = 21;
const ImsiAttachFailureCause congestion_Imsi = 22;
const ImsiAttachFailureCause gsmAuthenticationUnacceptable_Imsi = 23;
const ImsiAttachFailureCause serviceOptionNotSupported_Imsi = 32;
const ImsiAttachFailureCause requestedServiceOptionNotSubscribed_Imsi = 33;
const ImsiAttachFailureCause serviceOptionTemporarilyOutOfOrder_Imsi = 34;
const ImsiAttachFailureCause callCannotBeIdentified = 38;
const ImsiAttachFailureCause failRetryUponEntryIntoANewCell_Imsi = 48;
//value range 48 - 63 is used to retry upon entry into a new cell;
const ImsiAttachFailureCause semanticallyIncorrectMessage_Imsi = 95;
const ImsiAttachFailureCause invalidMandatoryInformation_Imsi = 96;
const ImsiAttachFailureCause messageTypeNon_existentOrNotImplemented_Imsi = 97;
const ImsiAttachFailureCause messageTypeNotCompatibleWithTheProtocolState_Imsi = 98;
const ImsiAttachFailureCause informationElementNon_existentOrNotImplemented_Imsi = 99;
const ImsiAttachFailureCause conditionalIeError_Imsi = 100;
const ImsiAttachFailureCause messageNotCompatibleWithTheProtocolState_Imsi = 101;
const ImsiAttachFailureCause protocolError_Imsi = 111; // unspecified
//The following activate PDP context MS failure causes are defined in the section 10.5.6.6 of 3GPP TS 24.008 v6.1.0.
typedef unsigned short ActPdpContextMsFailureCause;
const ActPdpContextMsFailureCause operatorDeterminedBarring_Ms = 8;
const ActPdpContextMsFailureCause llcOrSndcpFailure = 25;
const ActPdpContextMsFailureCause insufficientResources = 26;
const ActPdpContextMsFailureCause unknownOrMissingAccessPointName = 27;
const ActPdpContextMsFailureCause unknownPdpAddressOrPdpType_Ms = 28;
const ActPdpContextMsFailureCause userAuthenticationFailed_Ms = 29;
const ActPdpContextMsFailureCause activationRejectedByGgsn = 30;
const ActPdpContextMsFailureCause activationRejected = 31; //unspecified
const ActPdpContextMsFailureCause serviceOptionNotSupported_Ms = 32;
const ActPdpContextMsFailureCause requestedServiceOptionNotSubscribed_Ms = 33; //redefined

```

```

const ActPdpContextMsFailureCause serviceOptionTemporarilyOutOfOrder_Ms = 34;//redefined
const ActPdpContextMsFailureCause nsapiAlreadyUsed = 35;
const ActPdpContextMsFailureCause regularPdpContextDeactivation = 36;
const ActPdpContextMsFailureCause qosNotAccepted = 37;
const ActPdpContextMsFailureCause networkFailure_Ms = 38;
const ActPdpContextMsFailureCause reactivationRequested = 39;
const ActPdpContextMsFailureCause featureNotSupported = 40;
const ActPdpContextMsFailureCause semanticErrorInTheTftOperation_Ms = 41;
const ActPdpContextMsFailureCause syntacticalErrorInTheTftOperation = 42;
const ActPdpContextMsFailureCause unknownPdpContext = 43;
const ActPdpContextMsFailureCause semanticErrorsInPacketFilters_Ms= 44;
const ActPdpContextMsFailureCause syntacticalErrorInPacketFilters= 45;
const ActPdpContextMsFailureCause PdpContextWithoutTftAlreadyActivated_Ms = 46;
const ActPdpContextMsFailureCause InvalidTransactionIdentifierValue = 81;
const ActPdpContextMsFailureCause semanticallyIncorrectMessage_Ms = 95;
//const ActPdpContextMsFailureCause invalidMandatoryInformation_Ms = 96;//redefined
const ActPdpContextMsFailureCause messageTypeNon_existentOrNotImplemented_Ms = 97;
const ActPdpContextMsFailureCause messageTypeNotCompatibleWithTheProtocolState_Ms = 98;
const ActPdpContextMsFailureCause informationElementNon_existentOrNotImplemented_Ms = 99;
const ActPdpContextMsFailureCause conditionalIEError_Ms = 100;
const ActPdpContextMsFailureCause messageNotCompatibleWithTheProtocolState_Ms = 101;
const ActPdpContextMsFailureCause protocolError_Ms = 111; // unspecified

```

//The following activate PDP context UMTS failure causes are defined in the section 7.7.1 of 3GPP TS 29.060 v6.1.0. and 3GPP TS 32.215 v5.4.0.

```

typedef unsigned short ActPdpContextUtmsFailureCause;
const ActPdpContextUtmsFailureCause non_existent = 192;
const ActPdpContextUtmsFailureCause invalidMessageFormat = 193;
const ActPdpContextUtmsFailureCause imsiNotKnown = 194;
const ActPdpContextUtmsFailureCause msIsGprsDetached = 195;
const ActPdpContextUtmsFailureCause msIsNotGprsResponding = 196;
const ActPdpContextUtmsFailureCause msRefuses = 197;
const ActPdpContextUtmsFailureCause versionNotSupported = 198;
const ActPdpContextUtmsFailureCause noResourcesAvailable = 199;
const ActPdpContextUtmsFailureCause serviceNotSupported = 200;
const ActPdpContextUtmsFailureCause mandatoryIEIncorrect = 201;
const ActPdpContextUtmsFailureCause mandatoryIEMissing = 202;
const ActPdpContextUtmsFailureCause optionalIEIncorrect = 203;
const ActPdpContextUtmsFailureCause systemFailure = 204;
const ActPdpContextUtmsFailureCause roamingRestriction = 205;
const ActPdpContextUtmsFailureCause p_tmsiSignatureMismatch = 206;
const ActPdpContextUtmsFailureCause gprsConnectionSuspended = 207;
const ActPdpContextUtmsFailureCause authenticationFailure = 208;
const ActPdpContextUtmsFailureCause userAuthenticationFailed_Utms = 209;
const ActPdpContextUtmsFailureCause contextNotFound = 210;

```

```

const ActPdpContextUtmsFailureCause allDynamicPdpAddressesAreOccupied = 211;
const ActPdpContextUtmsFailureCause noMemoryIsAvailable = 212;
const ActPdpContextUtmsFailureCause relocationFailure = 213;
const ActPdpContextUtmsFailureCause unknownMandatoryExtensionHeader = 214;
const ActPdpContextUtmsFailureCause semanticErrorInTheTftOperation_Utms = 215;
const ActPdpContextUtmsFailureCause syntacticErrorInTheTftOperation = 216;
const ActPdpContextUtmsFailureCause semanticErrorsInPacketFilters_Utms= 217;
const ActPdpContextUtmsFailureCause syntacticErrorsInPacketFilters= 218 ;
const ActPdpContextUtmsFailureCause missingOrUnknownApn = 219;
const ActPdpContextUtmsFailureCause unknownPdpAddressOrPdpType_Utms = 220;
const ActPdpContextUtmsFailureCause pdpContextWithoutTftAlreadyActivated_Utms = 221;
const ActPdpContextUtmsFailureCause apnAccessDenied_noSubscription = 222;
//value range 223-240 is for future use;
//value range 241-255 is reserved for GPRS charging protocol use;
const ActPdpContextUtmsFailureCause requestRelatedToPossiblyDuplicatedPacketsAlreadyFulfilled = 252;
const ActPdpContextUtmsFailureCause requestAlreadyFulfilled = 253;
const ActPdpContextUtmsFailureCause sequenceNumbersOfReleasedOrCancelledPacketsIsIncorrect = 254;
const ActPdpContextUtmsFailureCause requestNotFulfilled = 255;

//The following GPRS attach failure causes are defined in the section 10.5.5.14 of 3GPP TS 24.008 v6.1.0.
typedef unsigned short gprsAttathFailureCause;
const gprsAttathFailureCause imsiUnknownInHLR_Gprs = 2;
const gprsAttathFailureCause illegalMS_Gprs = 3;
const gprsAttathFailureCause illegalME_Gprs = 6;
const gprsAttathFailureCause gprsServicesNotAllowed = 7;
const gprsAttathFailureCause gprsServicesAndNon_GprsServicesNotAllowed = 8;
const gprsAttathFailureCause msIdentityCannotBeDerivedByTheNetwork = 9;
const gprsAttathFailureCause implicitlyDetached = 10;
const gprsAttathFailureCause plmnNotAllowed_Gprs = 11;
const gprsAttathFailureCause locationAreaNotAllowed_Gprs =12;
const gprsAttathFailureCause roamingNotAllowedInThisLocationArea_Gprs = 13;
const gprsAttathFailureCause noSuitableCellsInLocationArea_Gprs = 15;
const gprsAttathFailureCause networkFailure_Gprs = 17;
const gprsAttathFailureCause macFailure_Gprs = 20;
const gprsAttathFailureCause synchFailure_Gprs = 21;
const gprsAttathFailureCause congestion_Gprs = 22;
const gprsAttathFailureCause gsmAuthenticationUnacceptable_Gprs = 23;
const gprsAttathFailureCause NoPdpContextActivated = 40;
const gprsAttathFailureCause failRetryUponEntryIntoANewCell_Gprs = 48;
//value range 48 - 63 is used to retry upon entry into a new cell;
const gprsAttathFailureCause semanticallyIncorrectMessage_Gprs = 95;
const gprsAttathFailureCause invalidMandatoryInformation_Gprs = 96;
const gprsAttathFailureCause messageTypeNon_existentOrNotImplemented_Gprs = 97;
const gprsAttathFailureCause messageTypeNotCompatibleWithTheProtocolState_Gprs = 98;
const gprsAttathFailureCause informationElementNon_existentOrNotImplemented_Gprs = 99;

```

```

const gprsAttathFailureCause conditionalIeError_Gprs = 100;
const gprsAttathFailureCause messageNotCompatibleWithTheProtocolState_Gprs = 101;
const gprsAttathFailureCause protocolError_Gprs = 111; // unspecified

// The following originating and terminating SMS failure causes are defined in the section 8.2.5.4 of 3GPP TS 24.011
v5.2.0.
typedef unsigned short smsFailureCause;
// Cause values in a mobile originating SM_transfer attempt failure
const smsFailureCause unassignedOrUnallocatedNumber = 1;
const smsFailureCause operatorDeterminedBarring_Sms = 8;
const smsFailureCause callBarred = 10;
const smsFailureCause reserved = 11;
const smsFailureCause shortMessageTransferRejected = 21;
const smsFailureCause destinationOutOfOrder = 27;
const smsFailureCause unidentifiedSubscriber = 28;
const smsFailureCause facilityRejected = 29;
const smsFailureCause unknownSubscriber = 30;
const smsFailureCause networkOutOfOrder = 38;
const smsFailureCause temporaryFailure = 41;
const smsFailureCause congestion_Sms = 42;
const smsFailureCause resourcesUnavailable = 47; //unspecified
const smsFailureCause requestedFacilityNotSubscribed = 50;
const smsFailureCause requestedFacilityNotImplemented = 69;
const smsFailureCause invalidShortMessageTransferReferenceValue = 81;
const smsFailureCause semanticallyIncorrectMessage_Sms = 95;
const smsFailureCause invalidMandatoryInformation_Sms = 96;
const smsFailureCause messageTypeNon_existentOrNotImplemented_Sms = 97;
const smsFailureCause messageNotCompatibleWithShortMessageProtocolState = 98;
const smsFailureCause informationElementNon_existentOrNotImplemented_Sms = 99;
const smsFailureCause protocolError_Sms = 111; //unspecified
const smsFailureCause interworking = 127; //unspecified
// Cause values in a mobile terminating SM_transfer attempt failure
const smsFailureCause memoryCapacityExceeded = 22;
//const smsFailureCause invalidShortMessageTransferReferenceValue = 81; //redefined
//const smsFailureCause semanticallyIncorrectMessage_Sms = 95; //redefined
//const smsFailureCause invalidMandatoryInformation_Sms = 96; //redefined
//const smsFailureCause messageTypeNon_existentOrNotImplemented_Sms = 97; //redefined
//const smsFailureCause messageNotCompatibleWithShortMessageProtocolState = 98; //redefined
//const smsFailureCause informationElementNon_existentOrNotImplemented_Sms = 99; //redefined
//const smsFailureCause protocolError_Sms = 111; //unspecified //redefined

typedef unsigned short CauseType;
const CauseType sum = 0;
const CauseType other = 65535;
const CauseType noResponse = 65534;

```

```

struct CausePairType
{
    CauseType cause;
    unsigned long value;
};
typedef sequence<CausePairType> TDSCDMAMeasurementType3;

typedef unsigned short TrafficClass;
const TrafficClass conversational = 1;
const TrafficClass streaming = 2;
const TrafficClass interactive = 3;
const TrafficClass background = 4;
struct ClassPairType
{
    TrafficClass class;
    unsigned long value;
};
typedef sequence<ClassPairType> TDSCDMAMeasurementType4;
typedef string LocationAreaIdentificationType;
//LocationAreaIdentificationType is composed of MCC, MNC and LAC;
struct LocationAreaMeasurementType
{
    LocationAreaIdentificationType LocationAreaIdentification;
    unsigned long value;
};
typedef sequence<LocationAreaMeasurementType> TDSCDMAMeasurementType5;

typedef string RoutingAreaIdentificationType;
// RoutingAreaIdentificationType is composed of LAI and RAC;
struct RoutingAreaMeasurementType
{
    RoutingAreaIdentificationType RoutingAreaIdentification;
    unsigned long value;
};
typedef sequence<RoutingAreaMeasurementType> TDSCDMAMeasurementType6;
};

#endif

```

### 6.3 性能管理接口功能相关的文件

#### 6.3.1 性能测量数据文件的 Schema 定义<TDHsupaMeasCollec.xsd>

下面的Schema文件中用到的字段的说明见附录A，示例参见附录B。

版本号：PM FILE V1.0

<?xml version="1.0" encoding="UTF-8"?>

```

<!-- Measurement collection data file XML schema TDHsupaMeasCollec.xsd -->
<schema targetNamespace="http://latest/nmc-omc/cmNrm.doc#TDHsupaMeasCollec" elementFormDefault="qualified"
xmlns="http://www.w3.org/2001/XMLSchema" xmlns:mc="http://latest/nmc-omc/cmNrm.doc#TDHsupaMeasCollec">
  <!-- Measurement collection data file root XML element -->
  <element name="measCollecFile">
    <complexType>
      <sequence>
        <element name="fileHeader">
          <complexType>
            <sequence>
              <element name="fileSender">
                <complexType>
                  <attribute name="localDn" type="string" use="optional"/>
                  <attribute name="elementType" type="string" use="optional"/>
                </complexType>
              </element>
              <element name="measCollec">
                <complexType>
                  <attribute name="beginTime" type="dateTime" use="required"/>
                </complexType>
              </element>
            </sequence>
            <attribute name="fileFormatVersion" type="string" use="required"/>
            <attribute name="vendorName" type="string" use="optional"/>
            <attribute name="dnPrefix" type="string" use="optional"/>
          </complexType>
        </element>
        <element name="measData" minOccurs="0" maxOccurs="unbounded">
          <complexType>
            <sequence>
              <element name="managedElement">
                <complexType>
                  <attribute name="localDn" type="string" use="optional"/>
                  <attribute name="userLabel" type="string" use="optional"/>
                  <attribute name="swVersion" type="string" use="optional"/>
                </complexType>
              </element>
              <element name="measInfo" minOccurs="0" maxOccurs="unbounded">
                <complexType>
                  <sequence>
                    <element name="job" minOccurs="0">
                      <complexType>
                        <attribute name="jobId" type="string" use="required"/>
                      </complexType>
                    </element>
                  </sequence>
                </complexType>
              </element>
            </sequence>
          </complexType>
        </element>
      </sequence>
    </complexType>
  </element>
</schema>

```

```

<element name="granPeriod">
  <complexType>
    <attribute name="duration" type="duration" use="required"/>
    <attribute name="endTime" type="dateTime" use="required"/>
  </complexType>
</element>
<element name="repPeriod" minOccurs="0">
  <complexType>
    <attribute name="duration" type="duration" use="required"/>
  </complexType>
</element>
<choice>
  <element name="measTypes">
    <simpleType>
      <list itemType="mc:measName"/>
    </simpleType>
  </element>
  <element name="measType" minOccurs="0" maxOccurs="unbounded">
    <complexType>
      <simpleContent>
        <extension base="mc:measName">
          <attribute name="p" type="positiveInteger" use="required"/>
        </extension>
      </simpleContent>
    </complexType>
  </element>
</choice>
<element name="measValue" minOccurs="0" maxOccurs="unbounded">
  <complexType>
    <sequence>
      <choice>
        <element name="measResults">
          <simpleType>
            <list itemType="mc:measResultType"/>
          </simpleType>
        </element>
        <element name="r" minOccurs="0" maxOccurs="unbounded">
          <complexType>
            <simpleContent>
              <extension base="mc:measResultType">
                <attribute name="p" type="positiveInteger"
                  use="required"/>
              </extension>
            </simpleContent>
          </complexType>
        </element>
      </choice>
    </sequence>
  </complexType>
</element>

```



```

    </element>
  </choice>
  <element name="suspect" type="boolean" minOccurs="0"/>
</sequence>
  <attribute name="measObjLdn" type="string" use="required"/>
</complexType>
</element>
</sequence>
</complexType>
</element>
</sequence>
</complexType>
</element>
<element name="fileFooter">
  <complexType>
    <sequence>
      <element name="measCollec">
        <complexType>
          <attribute name="endTime" type="dateTime" use="required"/>
        </complexType>
      </element>
    </sequence>
  </complexType>
</element>
</sequence>
</complexType>
</element>
<simpleType name="measNameWithSubCounter">
  <restriction base="string">
    <pattern

```

value="(rabAssignmentMeasurement.failRabAssignEstabCsPerCause.|rabAssignmentMeasurement.failRabAssignEstabPsPerCause|rabReleaseRequestMeasurement.nbrRncRelCsRabPerCause.|rabReleaseRequestMeasurement.nbrRncRelPsRabPerCause.|iuConnectionMeasurement.nbrRncRelCsIuConnPerCause.|iuConnectionMeasurement.nbrRncRelPsIuConnPerCause.|iuConnectionMeasurement.attRelPsIuConnPerCause.|iuInterfaceMeasurement.nbrResetCsByRncPerCause.|iuInterfaceMeasurement.nbrResetPsByRncPerCause.|iuInterfaceMeasurement.nbrResetCsByCnPerCause.|iuInterfaceMeasurement.nbrResetPsByCnPerCause.|iuInterfaceMeasurement.nbrResetResCsByRncPerCause.|iuInterfaceMeasurement.nbrResetResPsByRncPerCause.|iuInterfaceMeasurement.nbrResetResCsByCnPerCause.|iuInterfaceMeasurement.nbrResetResPsByCnPerCause.|iuInterfaceMeasurement.nbrErrorIndCsByRncPerCause.|iuInterfaceMeasurement.nbrErrorIndPsByRncPerCause.|iuInterfaceMeasurement.nbrErrorIndCsByCnPerCause.|iuInterfaceMeasurement.nbrErrorIndPsByCnPerCause.|mncHardHandoverMeasurement.failHhoPerCause.|mncRelocationMeasurement.relocFailPrepCause.|mncInterSystemHandoverMeasurement.attRelocPrepOutRATHOCSPerCause.|mncInterSystemHandoverMeasurement.failRelocPrepOutRATHOCSPerCause.|mncInterSystemHandoverMeasurement.iRATHOFailOutCSCause.|mncInterSystemHandoverMeasurement.iRATHOFailIncCSCause.|mncInterSystemHandoverMeasurement.iRATHOFailOutPSUTRANCause.|cellRrcConnectionMeasurement.rrcAttConnEstabCause.|cellRrcConnectionMeasurement.rrcFailConnEstabCause.|cellRrcConnectionMeasurement.rrcSuccConnEstabCause.|cellRrcConnectionMeasurement.rrcFailConnReEstabCause.|cellRrcConnectionMeasurement.rrcAttConnRelDCCHCause.|cellRrcConnectionMeasurement.rrcAttConnRelCCCHCause.|cellHsupaMeasurement.HsupaFailMacdSetupCause.|cell

```

HsupaMeasurement.HsupaFailRBSetupCause,|hardHandoverInterCellIntraNode BMeasurement.hHOFailOutIntraNode BCause,|
hard HandoverInterNode BIntraRncMeasurement.hHOFailOutInterNode BIntraRNCCause,|hardHandoverInterRncViaIur
Measurement.h HOFailOutInterRNCIurCause,|hardHandoverInterRncMeasurement.failHhoOutInterRncCnPerCause,|hard
HandoverInterSystemMeasurement.attRelocPrepOutRATHOCSPerCause,|hardHandoverInterSystemMeasurement.failRelocPrepO
utRATHOCSPerCause,|hardHandoverInterSystemMeasurement.iRATHOFailOutCSCause,|hardHandoverInterSystemMeasurement
.iRATHOFailIncCSCause,|hardHandoverInterSystemMeasurement.iRATHOFailOutPSUTRANCause,|iuConnectionMeasurement.
attRelCsIuConnPerCause.)d{1,5}">

```

```

<!--HSUPA measurements on UtranCell beginning with family name "cellHsupa"-->

```

```

</restriction>

```

```

</simpleType>

```

```

<simpleType name="measNameWithoutSubCounter">

```

```

<restriction base="string">

```

```

<enumeration value="rncHardHandoverMeasurement.attHho"/>

```

```

<enumeration value="hardHandoverInterRncMeasurement.attHhoOutInterRncCn"/>

```

```

<enumeration value="rabAssignmentMeasurement.attRabAssignEstabCsPerType"/>

```

```

<enumeration value="rabAssignmentMeasurement.attRabAssignEstabPsPerType"/>

```

```

<enumeration value="iuConnectionMeasurement.attRncEstabCsIuConn"/>

```

```

<enumeration value="iuConnectionMeasurement.attRncEstabPsIuConn"/>

```

```

<enumeration value="signallingLinkTP.durLnkCong"/>

```

```

<enumeration value="signallingLinkSetTP.durLnkSetUnav"/>

```

```

<enumeration value="signallingLinkTP.durSigLinkOutOfService"/>

```

```

<enumeration value="hardHandoverInterNode BIntraRncMeasurement.hHOAttOutInterNode BIntraRNC"/>

```

```

<enumeration value="hardHandoverInterRnc ViaIurMeasurement.hHOAttOutInterRNCIur"/>

```

```

<enumeration value="hardHandoverInterCellIntraNode BMeasurement.hHOAttOutIntraNode B"/>

```

```

<enumeration value="hardHandoverInterNode BIntraRncMeasurement.hHOSuccOutInterNode BIntraRNC"/>

```

```

<enumeration value="hardHandoverInterRnc ViaIurMeasurement.hHOSuccOutInterRNCIur"/>

```

```

<enumeration value="hardHandoverInterCellIntraNode BMeasurement.hHOSuccOutIntraNode B"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaAbnormalEdchRelease"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaAttDchToEdchInterCell"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaAttDchToEdchIntraCell"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaAttEdchToDchInterCell"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaAttEdchToDchIntraCell"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaAttEdchToRachInterCell"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaAttEdchToRachIntraCell"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaAttMacdSetup"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaAttOutCellChange"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaAttRachToEdchInterCell"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaAttRachToEdchIntraCell"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaAttRBSetup"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaMeanNbrUser"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaNbrAckdMaceOcts"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaNbrAckdMacePdu"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaNbrMacePdu"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaSuccCnInitEdchRelease"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaSuccDchToEdchInterCell"/>

```

```

<enumeration value="cellHsupaMeasurement.HsupaSuccDchToEdchIntraCell"/>
<enumeration value="cellHsupaMeasurement.HsupaSuccEdchReleaseUserInact"/>
<enumeration value="cellHsupaMeasurement.HsupaSuccEdchToDchInterCell"/>
<enumeration value="cellHsupaMeasurement.HsupaSuccEdchToDchIntraCell"/>
<enumeration value="cellHsupaMeasurement.HsupaSuccEdchToRachInterCell"/>
<enumeration value="cellHsupaMeasurement.HsupaSuccEdchToRachIntraCell"/>
<enumeration value="cellHsupaMeasurement.HsupaSuccMacdSetup"/>
<enumeration value="cellHsupaMeasurement.HsupaSuccOutCellChange"/>
<enumeration value="cellHsupaMeasurement.HsupaSuccRachToEdchInterCell"/>
<enumeration value="cellHsupaMeasurement.HsupaSuccRachToEdchIntraCell"/>
<enumeration value="cellHsupaMeasurement.HsupaSuccRBSetup"/>
<enumeration value="cellRelationHsupaMeasurement.HsupaAttOut"/>
<enumeration value="cellRelationHsupaMeasurement.HsupaSuccOut"/>
<enumeration value="mcInterSystemHandoverMeasurement.iRATHOAttIncCS"/>
<enumeration value="hardHandoverInterSystemMeasurement.iRATHOAttIncCS"/>
<enumeration value="mcInterSystemHandoverMeasurement.iRATHOAttIncPS"/>
<enumeration value="hardHandoverInterSystemMeasurement.iRATHOAttIncPS"/>
<enumeration value="mcInterSystemHandoverMeasurement.iRATHOAttOutCS"/>
<enumeration value="hardHandoverInterSystemMeasurement.iRATHOAttOutCS"/>
<enumeration value="mcInterSystemHandoverMeasurement.iRATHOAttOutPSUTRAN"/>
<enumeration value="hardHandoverInterSystemMeasurement.iRATHOAttOutPSUTRAN"/>
<enumeration value="mcInterSystemHandoverMeasurement.iRATHOSuccIncCS"/>
<enumeration value="hardHandoverInterSystemMeasurement.iRATHOSuccIncCS"/>
<enumeration value="mcInterSystemHandoverMeasurement.iRATHOSuccIncPS"/>
<enumeration value="hardHandoverInterSystemMeasurement.iRATHOSuccIncPS"/>
<enumeration value="mcInterSystemHandoverMeasurement.iRATHOSuccOutCS"/>
<enumeration value="hardHandoverInterSystemMeasurement.iRATHOSuccOutCS"/>
<enumeration value="mcInterSystemHandoverMeasurement.iRATHOSuccOutPSUE"/>
<enumeration value="hardHandoverInterSystemMeasurement.iRATHOSuccOutPSUE"/>
<enumeration value="mcInterSystemHandoverMeasurement.iRATHOSuccOutPSUTRAN"/>
<enumeration value="hardHandoverInterSystemMeasurement.iRATHOSuccOutPSUTRAN"/>
<enumeration value="signallingLinkTP.nbrChangeBackAckRx"/>
<enumeration value="signallingLinkTP.nbrChangeBackAckTx"/>
<enumeration value="signallingLinkTP.nbrChangeBackRx"/>
<enumeration value="signallingLinkTP.nbrChangeBackTx"/>
<enumeration value="signallingLinkTP.nbrChangeOverAckRx"/>
<enumeration value="signallingLinkTP.nbrChangeOverAckTx"/>
<enumeration value="signallingLinkTP.nbrChangeOverRx"/>
<enumeration value="signallingLinkTP.nbrChangeOverTx"/>
<enumeration value="signallingLinkTP.nbrCong1"/>
<enumeration value="signallingLinkTP.nbrCong2"/>
<enumeration value="signallingLinkTP.nbrCong3"/>
<enumeration value="signallingLinkTP.nbrLnkConAckRx"/>
<enumeration value="signallingLinkTP.nbrLnkConAckTx"/>
<enumeration value="signallingLinkTP.nbrLnkConOrdRx"/>

```

```

<enumeration value="signallingLinkTP.nbrLnkConOrdTx"/>
<enumeration value="signallingLinkTP.nbrLnkErrPduRx"/>
<enumeration value="signallingLinkTP.nbrLnkForceUninhRx"/>
<enumeration value="signallingLinkTP.nbrLnkForceUninhTx"/>
<enumeration value="signallingLinkTP.nbrLnkInhDenRx"/>
<enumeration value="signallingLinkTP.nbrLnkInhDenTx"/>
<enumeration value="signallingLinkTP.nbrLnkLocInhTstRx"/>
<enumeration value="signallingLinkTP.nbrLnkLocInhTstTx"/>
<enumeration value="signallingLinkTP.nbrLnkRmtInhTstRx"/>
<enumeration value="signallingLinkTP.nbrLnkRmtInhTstTx"/>
<enumeration value="signallingLinkTP.nbrLnkTstAckRx"/>
<enumeration value="signallingLinkTP.nbrLnkTstAckTx"/>
<enumeration value="signallingLinkTP.nbrLnkTstRx"/>
<enumeration value="signallingLinkTP.nbrLnkTstTx"/>
<enumeration value="signallingPointTP.nbrMsuDropRteErr"/>
<enumeration value="signallingLinkTP.nbrMsuRx"/>
<enumeration value="signallingLinkTP.nbrMsuTx"/>
<enumeration value="iuInterfaceMeasurement.nbrOverloadControlCsByCn"/>
<enumeration value="iuInterfaceMeasurement.nbrOverloadControlCsByRnc"/>
<enumeration value="iuInterfaceMeasurement.nbrOverloadControlPsByCn"/>
<enumeration value="iuInterfaceMeasurement.nbrOverloadControlPsByRnc"/>
<enumeration value="signallingLinkTP.nbrSifOctRx"/>
<enumeration value="signallingLinkTP.nbrSifOctTx"/>
<enumeration value="signallingLinkTP.nbrSioOctRx"/>
<enumeration value="signallingLinkTP.nbrSioOctTx"/>
<enumeration value="signallingPointTP.nbrTraRx"/>
<enumeration value="signallingPointTP.nbrTraTx"/>
<enumeration value="signallingLinkTP.nbrTxCongDrop"/>
<enumeration value="signallingLinkTP.nbrTxDrop"/>
<enumeration value="signallingPointTP.nbrUsrUnavailRx"/>
<enumeration value="signallingPointTP.nbrUsrUnavailTx"/>
<enumeration value="mcRelocationMeasurement.relocAttPrep"/>
<enumeration value="mcRelocationMeasurement.relocSucc"/>
<enumeration value="mcRelocationMeasurement.relocSuccPrep"/>
<enumeration value="cellRrcConnectionMeasurement.rrcAttConnReEstab"/>
<enumeration value="cellRrcConnectionMeasurement.rrcSuccConnReEstab"/>
<enumeration value="rabAssignmentMeasurement.succRabAssignEstabCsPerType"/>
<enumeration value="rabAssignmentMeasurement.succRabAssignEstabPsPerType"/>
<enumeration value="mcInterSystemHandoverMeasurement.succRelocPrepOutRATHOCS"/>
<enumeration value="hardHandoverInterSystemMeasurement.succRelocPrepOutRATHOCS"/>
<!--HSUPA measurements on UtranCell beginning with family name "cellHsupa"-->
</restriction>
</simpleType>
<simpleType name="measName">
<union memberTypes="mc:measNameWithSubCounter mc:measNameWithoutSubCounter"/>

```

```
</simpleType>
<simpleType name="measResultType">
  <union memberTypes="decimal">
    <simpleType>
      <restriction base="string">
        <enumeration value="NIL"/>
      </restriction>
    </simpleType>
  </union>
</simpleType>
</schema>
```

### 6.3.2 性能测量数据文件的 XML header 定义

在实际性能测量数据文件中应该使用下面的XML header定义:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="TDHsupaMeasDataCollection.xsl"?>
<measCollecFile xmlns="http://latest/nmc-omc/cmNrm.doc#TDHsupaMeasCollec">
```

广东省网络空间安全协会受控资料

附录 A  
(规范性附录)  
Schema 文档补充说明

A.1 XML Schema文档结构标记

XML Schema文档结构标记约定如图A.1所示。

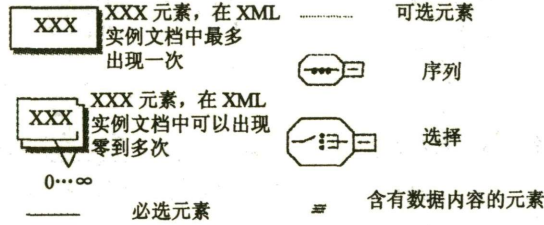


图 A.1 XML Schema 文档结构标记

A.2 性能测量数据文件的Schema定义

a) XML Schema 文档<TDHsupaMeasCollec.xsd>结构如图 A.2 所示。

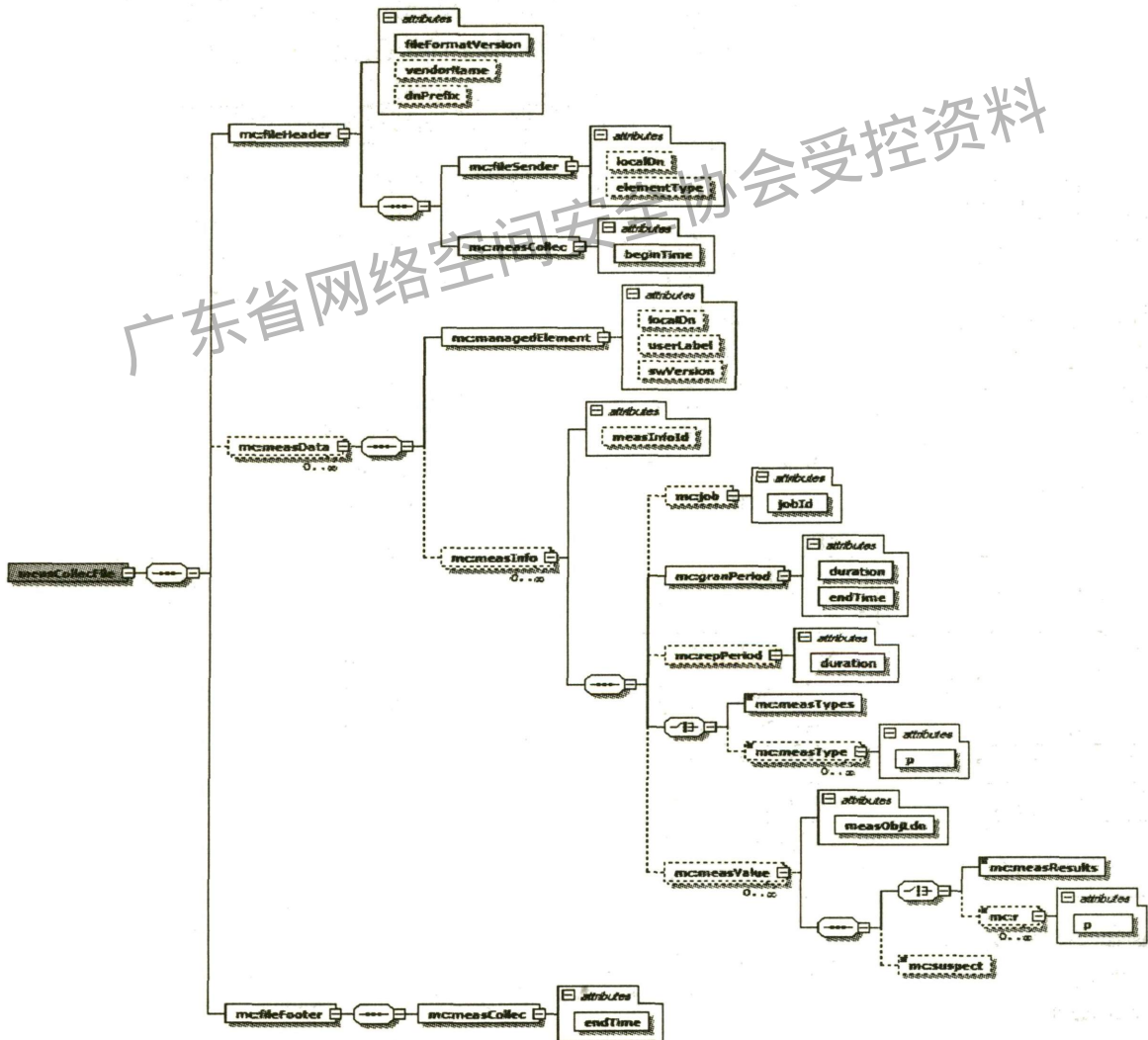


图 A.2 TDHsupaMeasCollec.xsd 文档结构

b) XML Schema 文档元素/属性说明如表 A.1 所示。

表 A.1 TDHsupaMeasCollec.xsd 文档元素/属性说明

元素/属性名称		元素/属性描述
元素名称	包含属性	
measCollecFile		性能数据采集文件。是该Schema的根元素。由三个子元素组成：文件头部（fileHeader）、采集数据(measData)以及文件尾部(FileFooter)
fileHeader	fileFormatVersion	文件头部。由两个子元素组成：文件发送方（fileSender）、测量采集开始时间（measCollec）。
	vendorName	自身包含三个属性：文件格式版本（fileFormatVersion）、制造商名称（vendorName）和识别名前缀（dnPrefix）
	dnPrefix	
measData		性能测量数据。在一份采集上报文件中可出现零（未采集到数据）至多次。由两个子元素组成：管理网元（managedElement）及其性能采集结果（measInfo）
fileFooter		文件尾部。包含子元素测量采集结束时间（measCollec）
fileSender	localDn	文件发送方。包含两个属性：本地识别名（localDN）、网元类型（elementType）。
	elementType	
Managed Element	localDn	被管网元。包括三个属性本地识别名（localDn）、用户友好名（userLabel）、软件版本（swVersion）
	userLabel	
	swVersion	
measInfo	measInfoId	测量信息。由四个子元素组成：测量任务（job）、测量粒度周期（granPeriod）、测量上报周期（repPeriod）、测量类型（measType/measTypes）和测量值（measValue）。包含一个可选属性：测量信息标识符（measInfoId）
job	jobId	测量任务。该元素为可选元素。由其属性jobId唯一标识
granPeriod	duration	测量粒度周期。包含两个属性，持续时间（duration）、结束时间（endTime）
	endTime	
repPeriod	duration	测量上报周期。该元素为可选元素。包含唯一属性：持续时间（duration）
measTypes/measType		采集类型。均由measName扩展而来。在XML文件实例中，两个元素择一使用。不同的是measTypes是以列表方式呈现，且只出现一次；measType可出现多次，由属性值为非负数的p加以区分
measType	p	p为属性限定。属性用于区分不同的measType
measResults/r		采集结果。均由measResultType扩展而来。在XML文件实例中，两个元素择一使用。值为空表示该采集项的取值无法获得。不同的是measResults是以列表方式呈现，且只出现一次；r可出现多次，由属性值为非负数的p加以区分。r的p属性应与measType的p属性一一对应
r	p	p为属性限定。表示对<measType p>的一个采集结果应答。<r p>需和<measType p>一一对应
measValue	measObjLdn	采集值。由两个子元素组成：采集结果列表（measResults/r）和一个标记采集数据是否可信的标志位（suspect）。本身还包含一个属性：测量对象本地识别名（measObjLdn）
suspect		用于标记采集值是否可信。默认值为False（即可信）
measCollec	beginTime	性能采集开始时间
	endTime	性能采集结束时间
measName		性能测量项名称。分为包含子测量项（measNameWithSubCounter）和不含子测量项（measNameWithoutSubCounter）两类。从3GPP TS 32.435规范中扩展而来
measNameWithSubCounter		含子测量项的数据测量项名称。表示为familyname.measurename.subcounter形式。从3GPP TS 32.435规范中扩展而来
measNameWithoutSubCounter		不含子测量项的数据测量项名称。表示为familyname.measurename形式。从3GPP TS 32.435规范中扩展而来

## 附录 B

(资料性附录)

## 性能管理功能相关 XML 文件示例

## B.1 性能管理功能相关XML文件示例一：不使用可选的p属性

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="MeasDataCollection.xsl"?>
```

<!-- The following is an example of a XML schema based XML measurement report file without use of optional positioning attributes on measurement types and results -->

```
<measCollecFile xmlns="http://latest/nmc-omc/cmNrm.doc#measCollec" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://latest/nmc-omc/cmNrm.doc#measCollec
D:\Downloads\GB\WCDMAM-2.XSD">
```

```
<fileHeader fileFormatVersion="PM FILE V1.0" vendorName="Company NN" dnPrefix="DC=a1.company NN.com,Sub
Network=1,IRPAgent=1">
```

```
<fileSender localDn="SubNetwork=CountryNN,MeContext=MEC-Gbg-1,ManagedElement=RNC-Gbg-1" elementType
="RNC"/>
```

```
<measCollec beginTime="2000-03-01T14:00:00+02:00"/>
```

```
</fileHeader>
```

```
<measData>
```

```
<managedElement localDn="SubNetwork=CountryNN,MeContext=MEC-Gbg-1,ManagedElement=RNC-Gbg-1"
userLabel="RNC Telecomville"/>
```

```
<measInfo>
```

```
<job jobId="1231"/>
```

```
<granPeriod duration="PT900S" endTime="2000-03-01T14:14:30+02:00"/>
```

```
<repPeriod duration="PT1800S"/>
```

```
<measTypes>mscBasicMeasurement.failImsiAttachsPerCause.50000
```

```
hardHandoverInterSystemMeasurement.failRelocOutInterSysPsPerCause.0
```

```
mobileManagementMeasurement.failIntraSgsnRaUpdatePerCause.22222
```

```
mobileManagementMeasurement.failIntraSgsnRaUpdatePerCause.1 </measTypes>
```

```
<measValue measObjLdn="RncFunction=RF-1,UtranCell=Gbg-997">
```

```
<measResults>234 345 567 789</measResults>
```

```
</measValue>
```

```
<measValue measObjLdn="RncFunction=RF-1,UtranCell=Gbg-998">
```

```
<measResults>890 901 123 234</measResults>
```

```
</measValue>
```

```
<measValue measObjLdn="RncFunction=RF-1,UtranCell=Gbg-999">
```

```
<measResults>456 567 678 789</measResults>
```

```
<suspect>true</suspect>
```

```
</measValue>
```

```
</measInfo>
```



```

</measData>
<fileFooter>
  <measCollec endTime="2000-03-01T14:15:00+02:00"/>
</fileFooter>
</measCollecFile>

```

## B.2 性能管理功能相关XML文件示例二：使用可选的p属性

```

<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="MeasDataCollection.xsl"?>

```

<!-- The following is an example of a XML schema based XML measurement report file with use of optional positioning attributes on measurement types and results -->

```

<measCollecFile xmlns="http://latest/nmc-omc/cmNrm.doc#measCollec" xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance" xsi:schemaLocation="http://latest/nmc-omc/cmNrm.doc#measCollec
D:\Downloads\GB\WCDMAM-2.XSD">

  <fileHeader fileFormatVersion="PM FILE V1.0" vendorName="Company NN" dnPrefix="DC=a1.company
NN.com,SubNetwork=1,IRPAgent=1">
    <fileSender localDn="SubNetwork=CountryNN,MeContext=MEC-Gbg-1,ManagedElement=RNC-Gbg-1" elementType
="RNC"/>
    <measCollec beginTime="2000-03-01T14:00:00+02:00"/>
  </fileHeader>
  <measData>
    <managedElement localDn="SubNetwork=CountryNN,MeContext=MEC-Gbg-1,ManagedElement=RNC-Gbg-1"
userLabel="RNC Telecomville"/>
    <measInfo>
      <job jobId="1231"/>
      <granPeriod duration="PT900S" endTime="2000-03-01T14:14:30+02:00"/>
      <repPeriod duration="PT1800S"/>
      <measType p="1">mscBasicMeasurement.attGetRoutingInfo</measType>
      <measType p="2">hardHandoverInterSystemMeasurement.failRelocOutInterSysPsPerCause.0</measType>
      <measType p="3">cellRrcConnectionMeasurement.failRrcConnReestabPerCause.50000</measType>
      <measType p="5">hardHandoverInterSystemMeasurement.failRelocOutInterSysPsPerCause.5</measType>
      <measValue measObjLdn="RncFunction=RF-1,UtranCell=Gbg-997">
        <r p="1">234</r>
        <r p="2">345</r>
        <r p="3">567</r>
        <r p="4">789</r>
      </measValue>
      <measValue measObjLdn="RncFunction=RF-1,UtranCell=Gbg-998">
        <r p="1">890</r>
        <r p="2">901</r>
        <r p="3">123</r>

```

```
<r p="4">234</r>
</measValue>
<measValue measObjLdn="RncFunction=RF-1,UtranCell=Gbg-999">
  <r p="1">456</r>
  <r p="2">567</r>
  <r p="3">678</r>
  <r p="4">789</r>
  <suspect>true</suspect>
</measValue>
</measInfo>
</measData>
<fileFooter>
  <measCollec endTime="2000-03-01T14:15:00+02:00"/>
</fileFooter>
</measCollecFile>
```

广东省网络空间安全协会受控资料

## 参 考 文 献

- [1] 3GPP TS 25.319 V8.5.0 Enhanced Uplink; Overall Description; Stage 2
  - [2] 3GPP TS 25.827 V7.1.0 1.28 Mcps TDD Enhanced Uplink; Physical Layer Aspects
  - [3] 3GPP TR 25.804 V6.1.0 Feasibility study on Uplink Enhancements for UTRA TDD
  - [4] 3GPP TS 25.331 V8.5.0 Radio Resource Control (RRC); Protocol Specification
  - [5] 3GPP TS 25.433 V8.3.0 UTRAN Iub Interface Node B Application Part (NBAP) Signalling
  - [6] 3GPP TS 32.405 V8.4.0 Performance Measurements Universal Terrestrial Radio Access Network (UTRAN)
- 

广东省网络空间安全协会受控资料

广东省网络空间安全协会受控资料

中华人民共和国  
通信行业标准  
2GHz TD-SCDMA 数字蜂窝移动通信网  
高速上行链路分组接入 (HSUPA) 网络管理技术要求  
YD/T 2427-2012

\*

人民邮电出版社出版发行  
北京市崇文区夕照寺街 14 号 A 座  
邮政编码: 100061  
宝隆元 (北京) 印刷技术有限公司印刷

\*

开本: 880×1230 1/16                      2013 年 3 月第 1 版  
印张: 4.5                                      2013 年 3 月北京第 1 次印刷  
字数: 121 千字

15115·44  
定价: 55 元