

YD

中华人民共和国通信行业标准

YD/T 1586.3-2007

2GHz WCDMA 数字蜂窝移动通信网

网络管理技术要求(第一阶段)

第3部分 基于CORBA技术的网络资源模型设计

Technical Specification for 2GHz WCDMA Digital Cellular Mobile
Communication Network Management (Phase I)

Part 3 CORBA Based Network Resource Model Design

2007-05-16 发布

2007-05-16 实施

中华人民共和国信息产业部 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 缩略语	1
4 配置网络资源模型设计	1
4.1 通用配置资源模型的IDL定义	1
4.2 无线接入网网络资源模型的IDL定义	17
4.3 核心网网络资源模型的IDL定义	28
5 性能网络资源模型设计	53
5.1 性能管理资源模型的IDL定义	53
5.2 数据类型的IDL定义	79
6 性能管理接口功能相关的文件	93
6.1 性能测量数据文件的Schema定义<measCollec.xsd>	93
6.2 性能测量数据文件的XML header定义	110
附录A（规范性附录） Schema文档补充说明	111
附录B（资料性附录） 性能管理功能相关XML文件示例	113
参考文献	116

前 言

本部分是《2GHz 数字蜂窝移动通信网网络管理技术要求》系列标准之一。该系列标准由 4 项标准组成，各项标准又分为多个部分。该系列标准和部分标准的名称预计如下：

1. 2GHz 数字蜂窝移动通信网网络管理通用技术要求 第 1 部分 基本原则
2. 2GHz 数字蜂窝移动通信网网络管理通用技术要求 第 2 部分 接口功能
3. 2GHz 数字蜂窝移动通信网网络管理通用技术要求 第 3 部分 接口分析
4. 2GHz 数字蜂窝移动通信网网络管理通用技术要求 第 4 部分 基于 CORBA 技术的管理接口设计
5. 2GHz WCDMA 数字蜂窝移动通信网网络管理技术要求（第一阶段） 第 1 部分 配置网络资源模型
6. 2GHz WCDMA 数字蜂窝移动通信网网络管理技术要求（第一阶段） 第 2 部分 性能网络资源模型
7. 2GHz WCDMA 数字蜂窝移动通信网网络管理技术要求（第一阶段） 第 3 部分 基于 CORBA 技术的网络资源模型设计
8. 2GHz cdma2000 数字蜂窝移动通信网网络管理技术要求（第一阶段） 第 1 部分 配置网络资源模型
9. 2GHz cdma2000 数字蜂窝移动通信网网络管理技术要求（第一阶段） 第 2 部分 性能网络资源模型
10. 2GHz cdma2000 数字蜂窝移动通信网网络管理技术要求（第一阶段） 第 3 部分 基于 CORBA 技术的网络资源模型设计
11. 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
12. 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
13. 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

本部分参考第三代移动通信伙伴项目（3GPP）中 TS 32 系列标准：

1. 3GPP TS 32.623 Telecommunication management; Configuration Management (CM); Generic network resources Integration Reference Point (IRP); Common Object Request Broker Architecture (CORBA) (通用网络资源集成参考点：基于 CORBA 接口设计)
2. 3GPP TS 32.633 Telecommunication management; Configuration Management (CM); Core Network Resources Integration Reference Point (IRP); Common Object Request Broker Architecture (CORBA) (核心网络资源集成参考点：基于 CORBA 接口设计)
3. 3GPP TS 32.643 Telecommunication management; Configuration Management (CM); UTRAN network resources Integration Reference Point (IRP); Common Object Request Broker Architecture (CORBA)

(UTRAN 网络资源集成参考点: 基于 CORBA 接口设计)

4. 3GPP TS 32.653 Telecommunication management; Configuration Management (CM); GERAN network resources Integration Reference Point (IRP): Common Object Request Broker Architecture (CORBA)

(GERAN 网络资源集成参考点: 基于 CORBA 接口设计)

本部分与上述国际标准之间的主要差异为:

- 增强 IDL 文件的注释说明;
- 增加了配置和性能网络资源模型新增部分内容的 IDL 定义;
- 增强配置和性能网络资源模型参数数据类型的定义;
- 配置资源模型的 IDL 定义增加一类文件 xxxNRMPProfile.idl, 包括 GenericNRMPProfile.idl、

IMDataProfile.idl、UtranNRMPProfile.idl 和 CoreNRMPProfile.idl。这类文件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;

- 完善配置和性能网络资源模型 Schema 文件定义;
- 对配置和性能网络资源模型的名称取值进行规范。

本部分与上述 3GPP 相关标准的一致性程度为非等效。

本部分的附录 A 为规范性附录, 附录 B 为资料性附录。

本部分由中国通信标准化协会提出并归口。

本部分起草单位: 中国移动通信集团公司、北京邮电大学

本部分主要起草人: 李治文、王 焯、芮兰兰、姚羿志、范小磊、王 峰、李云喜

广东省网络空间安全协会受控资料

2GHz WCDMA数字蜂窝移动通信网网络管理技术要求（第一阶段）

第3部分 基于CORBA技术的网络资源模型设计

1 范围

本部分规定了2GHz WCDMA数字蜂窝移动通信网（以下简称3G）（第一阶段）网络管理接口网络资源模型的IDL定义。

本部分适用于采用2GHz WCDMA数字蜂窝移动通信网的网络管理（第一阶段）。

2 规范性引用文件

下列文件中的条款通过本部分的引用而成为本部分的条款。凡是注日期的引用文件，其随后所有的修改单（不包括勘误的内容）或修订版均不适用于本部分。然而，鼓励根据本部分达成协议的各方研究是否可使用这些文件的最新版本。凡是不注日期的引用文件，其最新版本适用于本部分。

YD/T 1584.3 - 2007 2GHz数字蜂窝移动通信网网络管理通用技术要求 第3部分 接口分析

YD/T 1586.1 - 2007 2GHz WCDMA数字蜂窝移动通信网网络管理技术要求 第1部分 配置网络资源模型

YD/T 1586.2 - 2007 2GHz WCDMA数字蜂窝移动通信网网络管理技术要求 第2部分 性能网络资源模型

3 术语、定义和缩略语

3.1 缩略语

下列缩略语适用于本标准。

IDL	Interface Definition Language	接口定义语言
CORBA	Common Object Request Broker Architecture	公共对象请求代理体系

4 配置网络资源模型设计

注：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、UtranNRMDefs.idl和CoreNRMDefs.idl, 用来定义配置网络资源对象及其属性名称；

2) xxxNRMSystem.idl, 包括GenericNRMSystem.idl、UtranNRMSystem.idl和CoreNRMSystem.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、UtranNRMProfile.idl和CoreNRMProfile.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。

4.1 通用配置资源模型的IDL定义

4.1.1 GenericNRMSystem

```
//File "GenericNRMSystem.idl"
```

```
//The IRP document version number is "Generic NRM V1.0"
#ifndef GenericNRMSystem_idl
#define GenericNRMSystem_idl

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

    /**
     * The format of Distinguished Name (DN) is specified in "Name Conventions
     * for Managed Objects revision B".
     */
    typedef string DN;

    typedef sequence<DN> DNListType;

    typedef string ObjectIdType;

    typedef DN DNPrefixType;

    typedef string MobileCountryCodeType;
    typedef string ISDNAddrStringType;
    typedef sequence<MobileCountryCodeType> MobileCountryCodeSetType;

    typedef string NetworkTypeType;
    const NetworkTypeType AN = "Access Netowrk";
    const NetworkTypeType CN = "Core Netowrk";
    const NetworkTypeType AN_CN = "AN and CN";

    typedef unsigned long UserDefinedStateType;

    /**
     * A set of strings.
     */
}
```

```
*/
typedef sequence<string> StringSet;
typedef sequence <unsigned long> ULongSet;

enum NetworkIndicatorType
{
    international,
    spare,
    national,
    nationalSpare
};

enum SignallingPointLengthType
{
    bits_24,
    bits_14
};

struct SignallingInfoType
{
    SignallingPointLengthType signallingPointLength;
    unsigned long signallingPointCode;
    NetworkIndicatorType networkIndicator;
};

typedef unsigned long SignallingPointType;
const SignallingPointType SP=0;
const SignallingPointType HSTP=1;
const SignallingPointType LSTP=2;
const SignallingPointType HLSTP=3;

enum SignallingLinkTypeType
{
    N_SS7,
    W_SS7
};
```

```

typedef unsigned long linkDirectType;
const linkDirectType toHstp=1;
const linkDirectType toLstp=2;
const linkDirectType toCdmaGmsc=3;
const linkDirectType toMsc=4;
const linkDirectType toHlr=5;
const linkDirectType toMc=6;
const linkDirectType toScp=7;
const linkDirectType toBsc=8;
const linkDirectType toCncPstn=9;
const linkDirectType toCtPstn=10;
const linkDirectType toCmcc=11;
const linkDirectType toCtt=12;
const linkDirectType toVoiceMailBox=13;
const linkDirectType toColorRing=14;
const linkDirectType toGsm=15;
const linkDirectType toOthers=16;

typedef unsigned short SlcType;

typedef unsigned short SLSType;
typedef sequence<SLSType> SLSListType;

typedef unsigned short LinkStatusType;
const LinkStatusType normal_UDS = 0;
const LinkStatusType deactivated_UDS = 1;
const LinkStatusType failed_UDS = 2;
const LinkStatusType localBlocked_UDS = 3;
const LinkStatusType remoteBlocked_UDS = 4;
const LinkStatusType localInhibited_UDS = 5;
const LinkStatusType remoteInhibited_UDS = 6;

typedef unsigned long BandwidthType;
};

```

```
#endif
```

4.1.2 GenericNRMDefs

```
//File "GenericNRMDefs.idl"
```

```
//The IRP document version number is "GENERIC NRM V1.0"
```



```
#ifndef GenericNRMDefs_idl
#define GenericNRMDefs_idl

// #pragma prefix "3gppsa5.org"

//This module defines constants for each MO class name and
//the attribute names for each Generic MO class.

module GenericNRMDefs
{
    //Definitions for abstract MO class Top

    interface Top
    {
        const string ObjectClass = "ObjectClass";
        const string ObjectInstance = "ObjectInstance";
    };

    //Definitions for MO class IRPAgent

    interface IRPAgent: Top
    {
        const string CLASS = "IRPAgent";

        // Attribute Names
        //
        const string iRPAgentId = "iRPAgentId";
        const string systemDN = "systemDN";
    };

    //Definitions for abstract MO class GenericIRP

    interface GenericIRP: Top
    {
        const string CLASS = "GenericIRP";

        // Attribute Names
```

```
//  
const string iRPIId = "iRPIId";  
};  
  
//Definitions for MO class SubNetwork  
  
interface SubNetwork: Top  
{  
    const string CLASS = "SubNetwork";  
  
    // Attribute Names  
    //  
    const string subNetworkId = "subNetworkId";  
    const string dnPrefix = "dnPrefix";  
    const string userLabel = "userLabel";  
    const string setOfMcc = "setOfMcc";  
    const string userDefinedNetworkType = "userDefinedNetworkType";  
};
```

//Definitions for MO class MeContext

```
interface MeContext: Top  
{  
    const string CLASS = "MeContext";  
  
    // Attribute Names  
    //  
    const string meContextId = "meContextId";  
    const string dnPrefix = "dnPrefix";  
};
```

//Definitions for MO class ManagementNode

```
interface ManagementNode: Top  
{  
    const string CLASS = "ManagementNode";
```

```
// Attribute Names
//
const string managementNodeId = "managementNodeId";
const string managedElements = "managedElements";
const string userLabel = "userLabel";
const string userDefinedState = "userDefinedState";
const string swVersion = "swVersion";
const string locationName = "locationName";
const string vendorName = "vendorName";
};

//Definitions for MO class ManagedElement

interface ManagedElement: Top
{
    const string CLASS = "ManagedElement";

    // Attribute Names
    //
    const string managedElementId = "managedElementId";
    const string dnPrefix = "dnPrefix";
    const string userLabel = "userLabel";
    const string vendorName = "vendorName";
    const string locationName = "locationName";
    const string managedElementType = "managedElementType";
    const string managedBy = "managedBy";
    const string userDefinedState = "userDefinedState";
    const string swVersion = "swVersion";
};

//Definitions for abstract MO class ManagedFunction

interface ManagedFunction : Top
{
    const string CLASS = "ManagedFunction";

    // Attribute Names
```

```
//  
const string userLabel = "userLabel";  
};  
  
//Definitions for MO class VsDataContainer  
  
interface VsDataContainer: Top  
{  
const string CLASS = "VsDataContainer";  
  
//Attribute Names  
//  
const string vsDataContainerId = "vsDataContainerId";  
const string vsDataType = "vsDataType";  
const string vsData = "vsData";  
const string vsDataFormatVersion = "vsDataFormatVersion";  
};
```

```
//Definitions for MO class SignallingPoint
```

```
interface SignallingPoint: Top  
{  
const string CLASS = "SignallingPoint";  
  
// Attribute Names  
//  
const string signallingPointId = "signallingPointId";  
const string signallingInfo = "signallingInfo";  
const string signallingPointType = "signallingPointType";  
const string userLabel = "userLabel";  
};
```

```
interface SignallingLinkSetTP: Top
```

```
{  
const string CLASS = "SignallingLinkSetTP";  
  
// Attribute Names
```

```

//
const string signallingLinkSetTPIId = "signallingLinkSetTPIId";
const string adjacentSignallingInfo = "adjacentSignallingInfo";
const string userLabel = "userLabel";
const string signallingLinkType = "signallingLinkType";
};

```

```
interface SignallingLinkTP: Top
```

```

{
    const string CLASS = "SignallingLinkTP";

    // Attribute Names
    //
    const string signallingLinkTPIId = "signallingLinkTPIId";
    const string userLabel = "userLabel";
    const string slc = "slc";
    const string slsNormalList = "slsNormalList";
    const string slsCurrentList = "slsCurrentList";
    const string linkStatus = "linkStatus";
    const string bandwidth = "bandwidth";
};

```

```

};
#endif

```

4.1.3 GenericNRMPProfile

```

//File "GenericNRMPProfile.idl"
//The IRP document version number is "GENERIC NRM V1.0"
#ifndef GenericNRMPProfile_idl
#define GenericNRMPProfile_idl

#include "GenericNRMSystem.idl"

#pragma prefix "3gppsa5.org"

/**
 * This module defines the attribute names and
 * correspondig attribute types for all defined

```

* MO class. This module is used for reference.

*/

module GenericNRMPProfile

{

interface Top

{

readonly attribute string objectClass;

readonly attribute string objectInstance;

};

interface IRPAgent : Top

{

readonly attribute GenericNRMSystem::ObjectIdType iRPAgentId;

readonly attribute GenericNRMSystem::DN systemDN;

// The following notifications may be sent from this MO,

// notifyObjectCreation

// notifyObjectDeletion

// notifyAttributeValueChange

// notifyAckStateChanged

// notifyChangedAlarm

// notifyClearedAlarm

// notifyNewAlarm

// notifyComments

// notifyAlarmListRebuilt

// notifyPotentialFaultyAlarmList

};

interface GenericIRP : Top

{

readonly attribute string iRPId;

};

interface SubNetwork: Top

{

readonly attribute GenericNRMSystem::ObjectIdType subNetworkId;

```

readonly attribute GenericNRMSystem::DNPrefixType dnPrefix;
        attribute wstring userLabel;
readonly attribute GenericNRMSystem::MobileCountryCodeSetType setOfMcc;
readonly attribute GenericNRMSystem::NetworkTypeType userDefinedNetworkType;

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

interface MeContext: Top
{
    readonly attribute GenericNRMSystem::ObjectIdType meContextId;
    readonly attribute GenericNRMSystem::DNPrefixType dnPrefix;

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

interface ManagementNode : Top
{
    readonly attribute GenericNRMSystem::ObjectIdType managementNodeId;
    readonly attribute GenericNRMSystem::DNListType managedElements;
        attribute wstring userLabel;
    attribute GenericNRMSystem::UserDefinedStateType userDefinedState;
    readonly attribute string swVersion; // software version
    readonly attribute wstring locationName;

```

```

readonly attribute string vendorName;

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

interface ManagedElement : Top
{
    readonly attribute GenericNRMSystem::ObjectIdType managedElementId;
    readonly attribute GenericNRMSystem::DNPrefixType dnPrefix;
    attribute wstring userLabel;
    readonly attribute string vendorName;
    readonly attribute wstring locationName;
    readonly attribute GenericNRMSystem::StringSet managedElementType;
    readonly attribute GenericNRMSystem::DN managedBy;
    attribute GenericNRMSystem::UserDefinedStateType userDefinedState;
    readonly attribute string swVersion; // software version

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

```



```

        // notifyPotentialFaultyAlarmList
    };

interface ManagedFunction : Top
{
    attribute wstring userLabel;
};

interface VsDataContainer : Top
{
    readonly attribute GenericNRMSystem::ObjectIdType vsDataContainerId;
    readonly attribute string vsDataType;
        attribute any vsData;
    readonly attribute string vsDataFormatVersion;

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

interface SignallingPoint : Top
{
    readonly attribute GenericNRMSystem::ObjectIdType signallingPointId;
    readonly attribute GenericNRMSystem::SignallingInfoType signallingInfo;
    readonly attribute GenericNRMSystem::SignallingPointType signallingPointType;
        attribute wstring userLabel;

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

```

```

    // notifyObjectDeletion
    // notifyAttributeValueChange
    // notifyAckStateChanged
    // notifyChangedAlarm
    // notifyClearedAlarm
    // notifyNewAlarm
    // notifyComments
    // notifyAlarmListRebuilt
    // notifyPotentialFaultyAlarmList
};

interface SignallingLinkSetTP : Top
{
    readonly attribute GenericNRMSystem::ObjectIdType signallingLinkSetTPId;
        attribute GenericNRMSystem::SignallingInfoType adjacentSignallingInfo;
        attribute wstring userLabel;
    readonly attribute GenericNRMSystem::SignallingLinkTypeType signallingLinkType;

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

interface SignallingLinkTP : Top
{
    readonly attribute GenericNRMSystem::ObjectIdType signallingLinkTPId;
    readonly attribute wstring userLabel;
    readonly attribute GenericNRMSystem::SlcType slc;
    readonly attribute GenericNRMSystem::SLSListType slsNormalList;

```

```

readonly attribute GenericNRMSystem::SLSListType slsCurrentList;
readonly attribute GenericNRMSystem::LinkStatusType linkStatus;
readonly attribute GenericNRMSystem::BandwidthType bandwidth;

```

```

// The following notifications may be sent from this MO,

```

```

    // notifyObjectCreation
    // notifyObjectDeletion
    // notifyAttributeValueChange
    // notifyAckStateChanged
    // notifyChangedAlarm
    // notifyClearedAlarm
    // notifyNewAlarm
    // notifyComments
    // notifyAlarmListRebuilt
    // notifyPotentialFaultyAlarmList

```

```
};
```

```
};
```

```
#endif
```

4.1.4 IMDataDefs

```
//File "IMDataDefs.idl"
```

```
//The IRP document version number is "Inventory NRM V1.0"
```

```
#ifndef IMDataDefs_idl
```

```
#define IMDataDefs_idl
```

```
#include "GenericNRMDefs.idl"
```

```
/**
```

```
 * This module defines constants for each MO class name and
```

```
 * the attribute names for each defined MO class.
```

```
*/
```

```
module IMDataDefs
```

```
{
```

```
    /**
```

```
     * Definitions for MO class InventoryUnit
```

```
     */
```

```
    interface InventoryUnit: GenericNRMDefs::Top
```

```
    {
```

```

const string CLASS = "InventoryUnit";

//Attribute Names
//
const string inventoryUnitId = "inventoryUnitId";
const string inventoryUnitType = "inventoryUnitType";
const string vendorUnitFamilyType = "vendorUnitFamilyType";
const string vendorUnitTypeNumber = "vendorUnitTypeNumber";
const string vendorName = "vendorName";
const string serialNumber = "serialNumber";
const string versionNumber = "versionNumber";
const string dateOfManufacture = "dateOfManufacture";
const string dateOfLastService = "dateOfLastService";
const string unitPosition = "unitPosition";
const string manufacturerData = "manufacturerData";
};
};

#endif
4.1.5 IMDataProfile
//File "IMDataProfile.idl"
//The IRP document version number is "Inventory NRM V1.0"
#ifndef IMDataProfile_idl
#define IMDataProfile_idl

#include "GenericNRMSystem.idl"
#include "GenericNRMProfile.idl"

module IMDataProfile
{
    interface InventoryUnit : GenericNRMProfile::Top
    {
        readonly attribute GenericNRMSystem::ObjectIdType inventoryUnitId;
        readonly attribute string inventoryUnitType;
        readonly attribute string vendorUnitFamilyType;
        readonly attribute string vendorUnitTypeNumber;
        readonly attribute string vendorName;
    }
}

```

```

readonly attribute string serialNumber;
readonly attribute string versionNumber;
readonly attribute string dateOfManufacture;
readonly attribute string dateOfLastService;
readonly attribute wstring unitPosition;
readonly attribute string manufacturerData;

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

#endif

```

4.2 无线接入网网络资源模型的 IDL 定义

4.2.1 UtranNRMSystem

```

#define UtranNRMSystem_idl

#include "GenericNRMSystem.idl"

// #pragma prefix "3gppsa5.org"

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    uranCell;
    case FALSE: string cellGlobalId;
};

typedef GenericNRMSystem::ULongSet UraListType;

enum CellModeEnumType
{
    FDDMode,
    TDDMode_3_84Mcps,
    TDDMode_1_28Mcps
};
};
#endif
4.2.2 UtranNRMDefs
//File "UtranNRMDefs.idl"
//The IRP document version number is "UTRAN NRM V1.0"
#ifndef 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 mcc = "mcc";
    const string mnc = "mnc";
    const string maxCallCapability = "maxCallCapability";
    const string maxThroughOutput = "maxThroughOutput";
};

//Definitions for MO class NodeBFunction

interface NodeBFunction : GenericNRMDefs::ManagedFunction
{
    const string CLASS = "NodeBFunction";

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

//Definitions for MO class IubLink

interface IubLink : GenericNRMDefs::ManagedFunction
{
    const string CLASS = "IubLink";

    // including all Attribute Names from
```

```

// MO Class GenericNRMDefs::ManagedFunction
// additional Attribute Names is as follows.
//
const string iubLinkId = "iubLinkId";
const string relatedNodeB = "relatedNodeB";
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 relatedIubLink = "relatedIubLink";
const string cId= "cId";
const string localCellId= "localCellId";
const string uarfcnUl= "uarfcnUl";
const string uarfcnDl= "uarfcnDl";
const string primaryScramblingCode= "primaryScramblingCode";
const string primaryCpichPower= "primaryCpichPower";
const string maximumTransmissionPower= "maximumTransmissionPower";
const string primarySchPower= "primarySchPower";
const string secondarySchPower= "secondarySchPower";
const string bchPower= "bchPower";
const string lac= "lac";
const string rac= "rac";
const string sac= "sac";
const string uraList= "uraList";
const string cellMode= "cellMode";
};

//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 uarfcnUl= "uarfcnUl";
    const string uarfcnDl= "uarfcnDl";
    const string primaryScramblingCode= "primaryScramblingCode";
    const string primaryCpichPower= "primaryCpichPower";
    const string lac= "lac";
    const string userLabel = "userLabel";
    const string cellMode= "cellMode";
};

//Definitions for MO class ExternalUtranCell

interface ExternalUtranCell : GenericNRMDefs::ManagedFunction
{
    const string CLASS = "ExternalUtranCell";

    // Attribute Names
    //
    const string externalUtranCellId = "externalUtranCellId";
    const string mcc = "mcc";
    const string mnc = "mnc";
    const string cId= "cId";
    const string rncId= "rncId";
    const string uarfcnUl= "uarfcnUl";
    const string uarfcnDl= "uarfcnDl";
    const string primaryScramblingCode= "primaryScramblingCode";
    const string primaryCpichPower= "primaryCpichPower";
    const string lac= "lac";
    const string rac= "rac";
}

```

```
    const string cellMode= "cellMode";
};

//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";
    };
};
#endif

```

4.2.3 UtranNRMPProfile

```

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

#include "GenericNRMPProfile.idl"
#include "GenericNRMDefs.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 UtranNRMPProfile
{
    interface RncFunction : GenericNRMPProfile::ManagedFunction
    {
        readonly attribute GenericNRMSystem::ObjectIdType rncFunctionId;
        attribute unsigned long rncId;
        readonly attribute unsigned long mcc;
        readonly attribute unsigned long mnc;
        readonly attribute unsigned long maxCallCapability;
        readonly attribute unsigned long maxThroughOutput;

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

```

```

    // notifyAttributeValueChange
    // notifyAckStateChanged
    // notifyChangedAlarm
    // notifyClearedAlarm
    // notifyNewAlarm
    // notifyComments
    // notifyAlarmListRebuilt
    // notifyPotentialFaultyAlarmList
    // notifyStateChange
};

interface NodeBFunction : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType nodeBFunctionId;
    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 : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType iubLinkId;
    readonly attribute GenericNRMSystem::DN relatedNodeB;
    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 GenericNRMSSystem::ObjectIdType utranCellId;
    readonly attribute GenericNRMSSystem::DN relatedIubLink;
        attribute unsigned long cId;
        attribute unsigned long localCellId;
    readonly attribute unsigned long uarfcnUl;
    readonly attribute unsigned long uarfcnDl;
    readonly attribute unsigned long primaryScramblingCode; //0..511
    readonly attribute float primaryCpichPower; //-10dBm..+50dBm
    readonly attribute float maximumTransmissionPower; //0dBm..50dBm
    readonly attribute float primarySchPower; //-35dBm..+15dBm
    readonly attribute float secondarySchPower; //-35dBm..+15dBm
    readonly attribute float bchPower; //-35dBm..+15dBm
    readonly attribute unsigned long lac;
    readonly attribute unsigned long rac;
    readonly attribute unsigned long sac;
    readonly attribute UtranNRMSSystem::UraListType uraList;
    readonly attribute UtranNRMSSystem::CellModeEnumType cellMode;

    // 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 unsigned long uarfcnUl;
    readonly attribute unsigned long uarfcnDl;
    readonly attribute unsigned long primaryScramblingCode;    //0..511
    readonly attribute float primaryCpichPower;    //-10dBm..+50dBm
    readonly attribute unsigned long lac;
        attribute string userLabel;
    readonly attribute UtranNRMSystem::CellModeEnumType cellMode;

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

```

```
interface ExternalUtranCell : GenericNRMPProfile::ManagedFunction
```

```

{
    readonly attribute GenericNRMSystem::ObjectIdType externalUtranCellId;
        attribute unsigned long mcc;
        attribute unsigned long mnc;
        attribute unsigned long cId;
        attribute unsigned long rncId;
        attribute unsigned long uarfcnUl;
        attribute unsigned long uarfcnDl;
        attribute unsigned long primaryScramblingCode;    //0..511
        attribute float primaryCpichPower;    //-10dBm..+50dBm
        attribute unsigned long lac;

```

```

        attribute unsigned long rac;
readonly attribute UtranNRMSysyem::CellModeEnumType cellMode;

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

interface GsmRelation : GenericNRMPProfile::Top
{
    readonly attribute GenericNRMSysyem::ObjectIdType gsmRelationId;
        attribute GenericNRMSysyem::DN adjacentCell;
    readonly attribute unsigned long bcchFrequency;
    readonly attribute unsigned long ncc;
    readonly attribute unsigned long bcc;
    readonly attribute unsigned long lac;
        attribute string userLabel;

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

interface ExternalGSMCell : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSysyem::ObjectIdType externalGsmCellId;
        attribute unsigned long cellIdentity;
        attribute unsigned long 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
    };
};
#endif

```

4.3 核心网网络资源模型的 IDL 定义

4.3.1 CoreNRMDefs

```

//File "CoreNRMDefs.idl"
//The IRP document version number is "CN NRM V1.0"
#ifndef CoreNRMDefs_idl
#define CoreNRMDefs_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 CoreNRMDefs
{
    //Definitions for MO class MscFunction

    interface MscFunction : GenericNRMDefs::ManagedFunction
    {
        const string CLASS = "MscFunction";

        // including all Attribute Names from
        // MO Class GenericNRMDefs::ManagedFunction
        // additional Attribute Names is as follows.
        //
        const string mscFunctionId = "mscFunctionId";
        const string mscNumber = "mscNumber";
        const string mscType = "mscType";
    }
}

```



```

const string mscCapacity = "mscCapacity";
const string maxMscBHCA = "maxMscBHCA";
const string controlledRncList = "controlledRncList";
const string maxNum2MPort = "maxNum2MPort";
const string num2MCircuits = "num2MCircuits";
const string relatedIwf = "relatedIwf";
const string mccList = "mccList";
const string mncList = "mncList";
const string lacList = "lacList";
const string sacList = "sacList";
const string gcaList = "gcaList";
const string mscId = "mscId";    };

```

//Definitions for MO class GmscFunction

```

interface GmscFunction : GenericNRMDefs::ManagedFunction

```

```

{
    const string CLASS = "GmscFunction";

    // including all Attribute Names from
    // MO Class GenericNRMDefs::ManagedFunction
    // additional Attribute Names is as follows.
    //
    const string mscFunctionId = "gmscFunctionId";
    const string mscNumber = "mscNumber";
};

```

//Definitions for MO class HlrFunction

```

interface HlrFunction : GenericNRMDefs::ManagedFunction

```

```

{
    const string CLASS = "HlrFunction";

    // including all Attribute Names from
    // MO Class GenericNRMDefs::ManagedFunction
    // additional Attribute Names is as follows.
    //

```

```

const string hlrFunctionId = "hlrFunctionId";
const string hlrNumber = "hlrNumber";
const string maxNumImsi = "maxNumImsi";
const string rangeOfImsi = "rangeOfImsi";
const string maxNumMsisdn = "maxNumMsisdn";
const string rangeOfMsisdn = "rangeOfMsisdn";
const string maxNumPdpAddress = "maxNumPdpAddress";
};

```

//Definitions for MO class VlrFunction

```

interface VlrFunction : GenericNRMDefs::ManagedFunction

```

```

{
    const string CLASS = "VlrFunction";

    // including all Attribute Names from
    // MO Class GenericNRMDefs::ManagedFunction
    // additional Attribute Names is as follows.
    //
    const string vlrFunctionId = "vlrFunctionId";
    const string vlrNumber = "vlrNumber";
    const string maxNumberImsi = "maxNumImsi";
};

```

//Definitions for MO class AucFunction

```

interface AucFunction : GenericNRMDefs::ManagedFunction

```

```

{
    const string CLASS = "AucFunction";

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

```

//Definitions for MO class EirFunction

interface EirFunction : GenericNRMDefs::ManagedFunction

```
{
    const string CLASS = "EirFunction";

    // including all Attribute Names from
    // MO Class GenericNRMDefs::ManagedFunction
    // additional Attribute Names is as follows.
    //
    const string eirFunctionId = "eirFunctionId";
    const string eirNumber = "eirNumber";
    const string maxNumImei = "maxNumImei";
};
```

//Definitions for MO class SmsIwmscFunction

interface SmsIwmscFunction : GenericNRMDefs::ManagedFunction

```
{
    const string CLASS = "SmsIwmscFunction";

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

//Definitions for MO class SmsGmscFunction

interface SmsGmscFunction : GenericNRMDefs::ManagedFunction

```
{
    const string CLASS = "SmsGmscFunction";

    // including all Attribute Names from
    // MO Class GenericNRMDefs::ManagedFunction
```

```
// additional Attribute Names is as follows.
//
const string smsGmscFunctionId = "smsGmscFunctionId";
};

//Definitions for MO class IwfFunction

interface IwfFunction : GenericNRMDefs::ManagedFunction
{
    const string CLASS = "IwfFunction";

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

//Definitions for MO class CircuitEndPointSubgroup

interface CircuitEndPointSubgroup : GenericNRMDefs::Top
{
    const string CLASS = "CircuitEndPointSubgroup";

    // Attribute Names
    //
    const string circuitEndPointSubgroupId = "circuitEndPointSubgroupId";
    const string numofcircuits = "numofcircuits";
    const string circuitDirectionality = "circuitDirectionality";
    const string transmissionCharacteristics = "transmissionCharacteristics";
    const string userLabel = "userLabel";
    const string signallingInfoOfFarEnd = "signallingInfoOfFarEnd";
};

//Definitions for MO class ObservedDestination
```

```

interface ObservedDestination : GenericNRMDefs::Top
{
    const string CLASS = "ObservedDestination";

    // Attribute Names
    //
    const string observedDestinationId = "observedDestinationId";
    const string destinationCode = "destinationCode";
    const string destinationType = "destinationType";
};

```

//Definitions for MO class SgsnFunction

```

interface SgsnFunction : GenericNRMDefs::ManagedFunction
{
    const string CLASS = "SgsnFunction";

    // including all Attribute Names from
    // MO Class GenericNRMDefs::ManagedFunction
    // additional Attribute Names is as follows.
    //
    const string sgsnFunctionId = "sgsnFunctionId";
    const string sgsnNumber = "sgsnNumber";
    const string switchingCapacity = "switchingCapacity";
    const string userCapacity = "userCapacity";
    const string pdpNbrSupported = "pdpNbrSupported";
    const string mccList= "mccList";
    const string mncList = "mncList";
    const string lacList = "lacList";
    const string racList = "racList";
    const string sacList = "sacList";
    const string sgsnId = "sgsnId";
};

```

//Definitions for MO class GgsnFunction

```

interface GgsnFunction : GenericNRMDefs::ManagedFunction

```

```
{
    const string CLASS = "GgsnFunction";

    // including all Attribute Names from
    // MO Class GenericNRMDefs::ManagedFunction
    // additional Attribute Names is as follows.
    //
    const string ggsnFunctionId = "ggsnFunctionId";
    const string ggsnNumber = "ggsnNumber";
    const string switchingCapacity = "switchingCapacity";
    const string pdpNbrSupported = "pdpNbrSupported";
};

//Definitions for MO class GgsnApn

interface GgsnApn : GenericNRMDefs::Top
{
    const string CLASS = "GgsnApn";

    // Attribute Names
    // 广东省网络空间安全协会受控资料
    const string ggsnApnId = "ggsnApnId";
    const string accessPointName = "accessPointName";
    const string transAccess = "transAccess";
    const string maxPdpContextNum = "maxPdpContextNum";
    const string maxBitRate = "maxBitRate";
    const string pushSwitch = "pushSwitch";
    const string pushGreSwitch = "pushGreSwitch";
    const string dnsMode = "dnsMode";
};

//Definitions for MO class GgsnApnAddrPool

interface GgsnApnAddrPool : GenericNRMDefs::Top
{
    const string CLASS = "GgsnApnAddrPool";
```

```
// Attribute Names
//
const string ggsnApnAddrPoolId = "ggsnApnAddrPoolId";
const string addrType = "addrType";
const string addrSegList = "addrSegList";
};
```

```
//Definitions for MO class IpRouteTable
```

```
interface IpRouteTable : GenericNRMDefs::Top
{
    const string CLASS = "IpRouteTable";

    // Attribute Names
    //
    const string ipRouteTableId = "ipRouteTableId";
    const string userLabel = "userLabel";
};
```

```
//Definitions for MO class IpRouteTableItem
```

```
interface IpRouteTableItem : GenericNRMDefs::Top
{
    const string CLASS = "IpRouteTableItem";

    // Attribute Names
    //
    const string ipRouteTableItemId = "ipRouteTableItemId";
    const string ipRouteDest = "ipRouteDest";
    const string ipRouteNextHop = "ipRouteNextHop";
    const string ipRouteAge = "ipRouteAge";
    const string ipRouteMask = "ipRouteMask";
    const string ipRouteProto = "ipRouteProto";
    const string numOfHops = "numOfHops";
    const string ipRouteMetric1 = "ipRouteMetric1";
    const string ipRouteMetric2 = "ipRouteMetric2";
```

```
const string ipRouteMetric3 = "ipRouteMetric3";
const string ipRouteMetric4 = "ipRouteMetric4";
const string ipRouteMetric5 = "ipRouteMetric5";
const string ipRouteType = "ipRouteType";
};
```

```
//Definitions for MO class IpTransTable
```

```
interface IpTransTable : GenericNRMDefs::Top
```

```
{
    const string CLASS = "IpTransTable";

    // Attribute Names
    //
    const string ipTransTableId = "ipTransTableId";
    const string userLabel = "userLabel";
};
```

```
//Definitions for MO class IpTransTableItem
```

```
interface IpTransTableItem : GenericNRMDefs::Top
```

```
{
    const string CLASS = "IpTransTableItem";

    // Attribute Names
    //
    const string ipTransTableItemId = "ipTransTableItemId";
    const string phyAddress = "phyAddress";
    const string ipAddress = "ipAddress";
    const string mappingType = "mappingType";
};
```

```
//Definitions for MO class IpAddrTable
```

```
interface IpAddrTable : GenericNRMDefs::Top
```

```
{
    const string CLASS = "IpAddrTable";
```



```

// Attribute Names
//
const string ipAddrTableId = "ipAddrTableId";
const string userLabel = "userLabel";
};

//Definitions for MO class IpAddrTableItem

interface IpAddrTableItem : GenericNRMDefs::Top
{
    const string CLASS = "IpAddrTableItem";

    // Attribute Names
    //
    const string ipAddrTableItemId = "ipAddrTableItemId";
    const string ipAdEntAddr = "ipAdEntAddr";
    const string ipAdEntNetMask = "ipAdEntNetMask";
    const string ipAdEntBcastAddr = "ipAdEntBcastAddr";
    const string ipAdEntReasmMaxSize = "ipAdEntReasmMaxSize";
    const string ipAdEntSubnetworkType = "ipAdEntSubnetworkType";
};

//Definitions for MO class BgFunction

interface BgFunction : GenericNRMDefs::ManagedFunction
{
    const string CLASS = "BgFunction";

    // including all Attribute Names from
    // MO Class GenericNRMDefs::ManagedFunction
    // additional Attribute Names is as follows.
    //
    const string bgFunctionId = "bgFunctionId";
};
};
#endif

```

4.3.2 CoreNRMProfile

```

//File "CoreNRMProfile.idl"
//The IRP document version number is "CN NRM V1.0"
#ifndef CoreNRMProfile_idl
#define CoreNRMProfile_idl

#include "GenericNRMProfile.idl"
#include "GenericNRMDefs.idl"
#include "CoreNRMSystem.idl"

// #pragma prefix "3gppsa5.org"

/**
 * This module defines the attribute names and
 * correspondig attribute types for all defined
 * MO class in core network. This module is
 * used for reference.
 */
module CoreNRMProfile
{
    interface MscFunction : GenericNRMProfile::ManagedFunction
    {
        readonly attribute GenericNRMSystem::ObjectIdType mscFunctionId;
        readonly attribute GenericNRMSystem::ISDNAddrStringType mscNumber;
        readonly attribute CoreNRMSystem::MscTypeType mscType;
        readonly attribute unsigned long mscCapacity;
        readonly attribute unsigned long maxMscBHCA;
                attribute CoreNRMSystem::ControllRncListType controlledRncList;
        readonly attribute unsigned long maxNum2MPort;
        readonly attribute unsigned long num2MCircuits;
        readonly attribute string relatedIwf;
        readonly attribute CoreNRMSystem::ULongSetType mccList;
        readonly attribute CoreNRMSystem::ULongSetType mncList;
                attribute CoreNRMSystem::ULongSetType lacList;
                attribute CoreNRMSystem::ULongSetType sacList;
                attribute CoreNRMSystem::ULongSetType gcaList;
    }
}

```

```

        attribute unsigned long mscId;

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

interface GmscFunction : GenericNRMProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType mscFunctionId;
    readonly attribute GenericNRMSystem::ISDNAddrStringType mscNumber;

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

interface HlrFunction : GenericNRMProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType hlrFunctionId;
    readonly attribute GenericNRMSystem::ISDNAddrStringType hlrNumber;

```

```

    readonly attribute unsigned long maxNumImsi;
        attribute CoreNRMSystem::RangeOfImsiInHlrType rangeOfImsi;
    readonly attribute unsigned long maxNumMsisdn;
        attribute CoreNRMSystem::RangeOfMsisdnInHlrType rangeOfMsisdn;
    readonly attribute unsigned long maxNumPdpAddress;

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

interface VlrFunction : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType vlrFunctionId;
    readonly attribute GenericNRMSystem::ISDNAddrStringType vlrNumber;
    readonly attribute unsigned long maxNumImsi;

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

```

```
interface AucFunction : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType aucFunctionId;
    readonly attribute unsigned long maxNumImsi;

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

interface EirFunction : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType eirFunctionId;
    readonly attribute GenericNRMSystem::ISDNAddrStringType eirNumber;
    readonly attribute unsigned long maxNumImei;

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

```
interface SmsIwmscFunction : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType smsIwmscFunctionId;

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

```
interface SmsGmscFunction : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType smsGmscFunctionId;

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

```
interface IwfFunction : GenericNRMPProfile::ManagedFunction
{
```

readonly attribute GenericNRMSystem::ObjectIdType iwFunctionId;
 readonly attribute unsigned long iwCapacity;

// The following notifications may be sent from this MO,

// notifyObjectCreation
 // notifyObjectDeletion
 // notifyAttributeValueChange
 // notifyAckStateChanged
 // notifyChangedAlarm
 // notifyClearedAlarm
 // notifyNewAlarm
 // notifyComments
 // notifyAlarmListRebuilt
 // notifyPotentialFaultyAlarmList

};

interface CircuitEndPointSubgroup : GenericNRMPProfile::Top

{

readonly attribute GenericNRMSystem::ObjectIdType circuitEndPointSubgroupId;
 readonly attribute unsigned long numOfCircuits;
 readonly attribute CoreNRMSystem::CircuitDirectionalityType circuitDirectionality;
 readonly attribute CoreNRMSystem::TransmissionCharacteristicsType
 transmissionCharacteristics;
 attribute string userLabel;
 attribute GenericNRMSystem::SignallingInfoType signallingInfoOfFarEnd;

// The following notifications may be sent from this MO,

// notifyObjectCreation
 // notifyObjectDeletion
 // notifyAttributeValueChange
 // notifyAckStateChanged
 // notifyChangedAlarm
 // notifyClearedAlarm
 // notifyNewAlarm
 // notifyComments
 // notifyAlarmListRebuilt
 // notifyPotentialFaultyAlarmList

```

};

interface ObservedDestination : GenericNRMPProfile::Top
{
    readonly attribute GenericNRMSystem::ObjectIdType observedDestinationId;
    readonly attribute CoreNRMSystem::DestCodeType destinationCode;
    readonly attribute CoreNRMSystem::DestinationTypeType destinationType;

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

```

```

interface SgsnFunction : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType sgsnFunctionId;
    readonly attribute GenericNRMSystem::ISDNAddrStringType sgsnNumber;
    readonly attribute unsigned long switchingCapacity;
    readonly attribute unsigned long userCapacity;
    readonly attribute unsigned long pdpNbrSupported;
    readonly attribute CoreNRMSystem::StringSetType mccList;
    readonly attribute CoreNRMSystem::ULongSetType mncList;
        attribute CoreNRMSystem::ULongSetType lacList;
        attribute CoreNRMSystem::ULongSetType sacList;
        attribute CoreNRMSystem::ULongSetType racList;
        attribute unsigned long sgsnId;

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

```



```

    // notifyAlarmListRebuilt
    // notifyPotentialFaultyAlarmList
};

interface GgsnFunction : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType ggsnFunctionId;
    readonly attribute GenericNRMSystem::ISDNAddrStringType ggsnNumber;
    readonly attribute unsigned long switchingCapacity;
    readonly attribute unsigned long pdpNbrSupported;

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

interface GgsnApn : GenericNRMPProfile::Top
{
    readonly attribute GenericNRMSystem::ObjectIdType ggsnApnId;
    readonly attribute string accessPointName;
    readonly attribute CoreNRMSystem::TransAccessType transAccess;
    readonly attribute unsigned long maxPdpContextNum; // the default value is 0
    readonly attribute unsigned long maxBitRate; // the default value is 0
    readonly attribute CoreNRMSystem::PushSwitchType pushSwitch;
    readonly attribute CoreNRMSystem::PushGreSwitchType pushGreSwitch;
    readonly attribute CoreNRMSystem::DnsModeType dnsMode;

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

```

```

        // notifyObjectDeletion
        // notifyAttributeValueChange
    };

interface GgsnApnAddrPool : GenericNRMPProfile::Top
{
    readonly attribute GenericNRMSystem::ObjectIdType ggsnApnAddrPoolId;
    readonly attribute CoreNRMSystem::AddrTypeType addrType;
    readonly attribute CoreNRMSystem::AddrSegListType addrSegList;

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

interface IpRouteTable : GenericNRMPProfile::Top
{
    readonly attribute GenericNRMSystem::ObjectIdType ipRouteTableId;
        attribute string userLabel;

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

interface IpRouteTableItem : GenericNRMPProfile::Top
{
    readonly attribute GenericNRMSystem::ObjectIdType ipRouteTableItemId;
    readonly attribute string ipRouteDest;    //IP Address
    readonly attribute string ipRouteNextHop;    //IP Address
    readonly attribute unsigned long ipRouteAge;
    readonly attribute string ipRouteMask;
    readonly attribute CoreNRMSystem::IpRouteProtoType ipRouteProto;
    readonly attribute CoreNRMSystem::NumOfHopsType numOfHops;
    readonly attribute unsigned long ipRouteMetric1;

```

```

readonly attribute unsigned long ipRouteMetric2;
readonly attribute unsigned long ipRouteMetric3;
readonly attribute unsigned long ipRouteMetric4;
readonly attribute unsigned long ipRouteMetric5;
readonly attribute CoreNRMSystem::IpRouteTypeType ipRouteType;

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

interface IpTransTable : GenericNRMProfile::Top
{
    readonly attribute GenericNRMSystem::ObjectIdType ipTransTableId;
    attribute string userLabel;

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

interface IpTransTableItem : GenericNRMProfile::Top
{
    readonly attribute GenericNRMSystem::ObjectIdType ipTransTableItemId;
    attribute string phyAddress; // MAC Address
    attribute string ipAddress; // IP Address
    attribute CoreNRMSystem::MappingTypeType mappingType;

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

interface IpAddrTable : GenericNRMProfile::Top

```

```

{
    readonly attribute GenericNRMSystem::ObjectIdType ipAddrTableId;
        attribute string userLabel;

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

interface IpAddrTableItem : GenericNRMPProfile::Top
{
    readonly attribute GenericNRMSystem::ObjectIdType ipAddrTableItemId;
    readonly attribute string ipAdEntAddr; // IP Address
    readonly attribute string ipAdEntNetMask; // IP Address
    readonly attribute string ipAdEntBcastAddr; // IP Address
    readonly attribute unsigned long ipAdEntReasmMaxSize;
    readonly attribute CoreNRMSystem::SubnetworkTypeType ipAdEntSubnetworkType;

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

interface BgFunction : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType bgFunctionId;

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

```

```

        // notifyComments
        // notifyAlarmListRebuilt
        // notifyPotentialFaultyAlarmList
    };
};
#endif

```

4.3.3 CoreNRMSystem

```

//File "CoreNRMSystem.idl"
//The IRP document version number is "CN NRM V1.0"
#ifndef CoreNRMSystem_idl
#define CoreNRMSystem_idl

#include "GenericNRMSystem.idl"

// #pragma prefix "3gppsa5.org"

```

```

module CoreNRMSystem

```

```

{

```

```

    /**

```

```

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

```

```

    */

```

```

enum MscTypeType

```

```

{

```

```

    TMS1, //0

```

```

    TMS2, //1

```

```

    MSCVLR, //2

```

```

    MSC //3

```

```

};

```

```

struct RangeOfImsiType

```

```

{

```

```

    string startImsi;

```

```

    string endImsi;

```

```

};
typedef sequence<RangeOfImsiType> RangeOfImsiInHlrType;
struct RangeOfMsisdnType
{
    string startMsisdn;
    string endMsisdn;
};
typedef sequence<RangeOfMsisdnType> RangeOfMsisdnInHlrType;
enum CircuitDirectionalityType
{
    OnewayOut, //0
    OnewayIn,  //1
    Twoway     //2
};

typedef octet TransmissionCharacteristicsType;
//From right to left,
//the first bit denotes opticalFiberCable, 1 means support, 0 means not support;
//the second bit denotes coaxialCable, 1 means support, 0 means not support;
//the third bit denotes analogMicrowave, 1 means support, 0 means not support;
//the fourth bit denotes digitMicrowave, 1 means support, 0 means not support;
//the fifth bit denotes satellite, 1 means support, 0 means not support;
//the sixth bit denotes mixedGroup, 1 means support, 0 means not support;
//the seventh bit denotes transOthers, 1 means support, 0 means not support;
//the eighth bit is meaningless.

typedef char DestCodeElementType;
    // the possible value for DestCodeElementType are:
    // '0','1','2','3','4','5','6','7','8','9','0',
    // 'A','B','C','D','E','F','*','#'
typedef string DestCodeType; //a string type formed from DestCodeElementType
typedef sequence<DestCodeType> DestinationCodeList;
enum DestType
{
    International_dest,
    National_dest,
    Local_dest,

```

```

    Other_dest
};
typedef octet NatureOfAddressType;
//only 0 and 1 can occur in the string, it is of BITSTRING type.
enum DestinationTypeChoiceType
{
    NatureOfAddressChoice,
    DestTypeChoice
};
union DestinationTypeType switch(DestinationTypeChoiceType)
{
    case NatureOfAddressChoice : NatureOfAddressType natureOfAddress;
    case DestTypeChoice : DestType dest;
};
typedef GenericNRMSystem::DN ObjectType;
typedef sequence<ObjectType> ControllRncListType;
struct RelatedCircuitEPSSubgroupType
{
    unsigned long priority;
    GenericNRMSystem::DN circuitEPSSubgroup;
};
typedef sequence<RelatedCircuitEPSSubgroupType> RelatedCircuitEPSSubgroupListType;
struct RouteType
{
    string routeNo;
    RelatedCircuitEPSSubgroupListType circuitEPSSubgroupList;
};

enum TransAccessType
{
    Transparent, //0, and the default value
    NonTransparent //1
};
enum PushSwitchType
{
    UnSupport, //0
    Support //1
};

```

```

};

union PushGreSwitchType switch (PushSwitchType)
{
    case Support: boolean pushGreSwitch;
};

enum DnsModeType
{
    LocalSetPriority,
    RadiusAppointPriority
};

enum AddrTypeType
{
    Dynamic, //0
    Static //1
};

struct AddrSegType
{
    string begingIp; //IP Address
    unsigned long segLen;
};

typedef sequence<AddrSegType> AddrSegListType;
typedef unsigned short IpRouteProtoType; //1..16
const IpRouteProtoType other_Proto=1;
const IpRouteProtoType local_Proto=2;
const IpRouteProtoType netmgmt_Proto=3;
const IpRouteProtoType icmp_Proto=4;
const IpRouteProtoType egp_Proto=5;
const IpRouteProtoType ggp_Proto=6;
const IpRouteProtoType hello_Proto=7;
const IpRouteProtoType rip_Proto=8;
const IpRouteProtoType isIs_Proto=9;
const IpRouteProtoType esIs_Proto=10;
const IpRouteProtoType ciscoIgrp_Proto=11;
const IpRouteProtoType bbnSpfIgp_Proto=12;
const IpRouteProtoType ospf_Proto=13;
const IpRouteProtoType bgp_Proto=14;

```



```

const IpRouteProtoType idpr_Proto=15;
const IpRouteProtoType ciscoEigrp_Proto=16;
typedef unsigned short NumOfHopsType; //1..16
typedef unsigned short IpRouteTypeType; //1..4
const IpRouteTypeType other_RouteType=1;
const IpRouteTypeType reject_RouteType=2;
const IpRouteTypeType local_RouteType=3;
const IpRouteTypeType remote_RouteType=4;
typedef unsigned short MappingTypeType; //1..4
const MappingTypeType other_MappingType=1;
const MappingTypeType invalid_MappingType=2;
const MappingTypeType dynamic_MappingType=3;
const MappingTypeType static_MappingType=4;
typedef unsigned short SubnetworkTypeType; //1..3
const SubnetworkTypeType other_SubnetworkType=1;
const SubnetworkTypeType PDN_SubnetworkType=2;
const SubnetworkTypeType PSCoreNetwork_SubnetworkType=3;

typedef sequence<string> StringSetType;
typedef unsigned long ULongType;
typedef sequence< ULongType > ULongSetType;
};
#endif

```

5 性能网络资源模型设计

5.1 性能管理资源模型的 IDL 定义

注：下面的 IDL 文件为每个 family 定义了一个独立的 module。

- "family.measurementName.subcounter"可用于获取一个 MeasurementType 的某个 subcounter 值；
- "family.measurementName" 可用于获取一个 MeasurementType 的值。如果该 MeasurementType 有 subcounters，那么所有 subcounters 的值都应该返回；
- "family" 可用于获取该 family 下的所有 MeasurementType 的值。

其中，family 为《2GHz WCDMA 数字蜂窝移动通信网网络管理技术要求 第 2 部分 性能网络资源模型》中的英文表名；subcounter 设置时选相应的整数值（取值见下节数据类型的 IDL 定义“WCDMANRMMMeasurementSystem.idl”中的定义，对按每位置区或每路由区统计的性能测量项，其 subcounter 的取值为网络中的 LAI 或 RAI 的值，本规范未进行定义；对按用户归属的 HLR 的 HLRNumber 或漫游到的 VLR 的 VLRNumber 进行统计的性能测量项，其 subcounter 的取值为网络中的 HLRNumber 或 VLRNumber 的值，本规范未进行定义；“0”表示总和）。

例如,

(1) "mscBasicMeasurement.failImsiAttachsPerCauseGsm.5"可用于获取某个 cause (imeiNotAccepted) 的"failImsiAttachsPerCauseGsm"的值;

(2) "mscBasicMeasurement.failImsiAttachsPerCauseGsm" 可用于获取"failImsiAttachsPerCauseGsm" 的所有 subcounters 的值;

(3) "mscBasicMeasurement.attGetRoutingInfo" 可用于获取"attGetRoutingInfo"的值;

(4) "mscBasicMeasurement"可用于获取该 family 下的所有 MeasurementType 的值。

```
//File WCDMANRMMMeasurementDefs.idl
#ifdef WCDMANRMMMeasurementDefs_idl
#define WCDMANRMMMeasurementDefs_idl
```

```
// #pragma prefix "3gppsa5.org"
```

```
/**
```

```
* This module defines measurementType names constants
```

```
*/
```

```
module WCDMANRMMMeasurementDefs
```

```
{
```

```
    // msc measurement
```

```
    module mscBasicMeasurement
```

```
    {
```

```
        //get routing information from HLR
```

```
        const string attGetRoutingInfo = "attGetRoutingInfo";
```

```
        const string succGetRoutingInfo = "succGetRoutingInfo";
```

```
        //imsi attach and detach
```

```
        const string attImsiAttachs = "attImsiAttachs";
```

```
        const string succImsiAttachs = "succImsiAttachs";
```

```
        const string failImsiAttachsPerCause = "failImsiAttachsPerCause";
```

```
        const string nbrImsiDetachs="nbrImsiDetachs";
```

```
        //location update
```

```
        const string attLocationUpdatesIntraMsc= "attLocationUpdatesIntraMsc";
```

```
        const string succLocationUpdatesIntraMsc= "succLocationUpdatesIntraMsc";
```

```
        const string failLocationUpdatesIntraMscPerCause= "failLocationUpdatesIntraMscPerCause";
```

```
        const string attLocationUpdatesInterMsc= "attLocationUpdatesInterMsc";
```

```
        const string succLocationUpdatesInterMsc= "succLocationUpdatesInterMsc";
```

```
        const string failLocationUpdatesInterMscPerCause= "failLocationUpdatesInterMscPerCause";
```

```

//originating and terminating sms via MSC
const string attOrigSmsCs = "attOrigSmsCs";
const string succOrigSmsCs = "succOrigSmsCs";
const string failOrigSmsCsPerCause = "failOrigSmsCsPerCause";
const string attTermSsmsCs = "attTermSsmsCs";
const string succTermSmsCs = "succTermSmsCs";
const string failTermSmsCsPerCause = "failTermSmsCsPerCause";
//incoming handover inter MSC
const string attIncHosInterMsc = "attIncHosInterMsc";
const string succIncHosInterMsc = "succIncHosInterMsc";
//outgoing handover inter MSC
const string attOutHosInterMsc = "attOutHosInterMsc";
const string succOutHosInterMsc = "succOutHosInterMsc";
//subsequent handover to MSCa
const string attSubsequentHosToMsca = "attSubsequentHosToMsca";
const string succSubsequentHosToMsca = "succSubsequentHosToMsca";
//subsequent handover to MSCc
const string attSubsequentHosToMsc = "attSubsequentHosToMsc";
const string succSubsequentHosToMsc = "succSubsequentHosToMsc";
//external handover
const string attExternalHos = "attExternalHos";
const string attExternalHosPerCause = "attExternalHosPerCause";
const string failExternalHosWithReconn = "failExternalHosWithReconn";
const string failExternalHosWithLossOfConn = "failExternalHosWithLossOfConn";
//paging
const string attPageReqsPerLa = "attPageReqsPerLa";
const string succPageReqsPerLa = "succPageReqsPerLa";
const string attRepageReqsPerLa = "attRepageReqsPerLa";
};

module mscQos
{
    const string meanDurOfCallSetup = "meanDurOfCallSetup";
    const string meanDurOfCallAssignGsm = "meanDurOfCallAssignGsm";
    const string meanDurOfCallRabAssignUmts = "meanDurOfCallRabAssignUmts";
    const string meanDurOfLuService = "meanDurOfLuService";
    const string meanCallDur = "meanCallDur";

```

```
    const string meanDurOfTrunkSeizure = "meanDurOfTrunkSeizure";
};

//exchange measurement
module mobileTrafficFlow
{
    //GSM originating call
    const string attOrigCallsGsm = "attOrigCallsGsm";
    const string succOrigCallsGsm = "succOrigCallsGsm";
    const string ansOrigCallsGsm = "ansOrigCallsGsm";
    const string failOrigCallsGsmPerCause = "failOrigCallsGsmPerCause";
    const string attOrigCallTrafficGsm = "attOrigCallTrafficGsm";
    const string succOrigCallTrafficGsm = "succOrigCallTrafficGsm";
    const string ansOrigCallTrafficGsm = "ansOrigCallTrafficGsm";
    //WCDMA originating call
    const string attOrigCallsUmts = "attOrigCallsUmts";
    const string succOrigCallsUmts = "succOrigCallsUmts";
    const string ansOrigCallsUmts = "ansOrigCallsUmts";
    const string failOrigCallsUmtsPerCause = "failOrigCallsUmtsPerCause";
    const string attOrigCallTrafficUmts = "attOrigCallTrafficUmts";
    const string succOrigCallTrafficUmts = "succOrigCallTrafficUmts";
    const string ansOrigCallTrafficUmts = "ansOrigCallTrafficUmts";
    //internal call
    const string attInternalCalls = "attInternalCalls";
    const string succInternalCalls = "succInternalCalls";
    const string ansInternalCalls = "ansInternalCalls";
    const string failInternalCallsPerCause = "failInternalCallsPerCause";
    const string attInternalCallTraffic = "attInternalCallTraffic";
    const string succInternalCallTraffic = "succInternalCallTraffic";
    const string ansInternalCallTraffic = "ansInternalCallTraffic";
    //GSM terminating call
    const string attTermCallsGsm = "attTermCallsGsm";
    const string succTermCallsGsm = "succTermCallsGsm";
    const string ansTermCallsGsm = "ansTermCallsGsm";
    const string failTermCallsGsmPerCause = "failTermCallsGsmPerCause";
    const string attTermCallTrafficGsm = "attTermCallTrafficGsm";
    const string succTermCallTrafficGsm = "succTermCallTrafficGsm";
```

```
const string ansTermCallTrafficGsm = "ansTermCallTrafficGsm";
//WCDMA terminating call
const string attTermCallsUmts = "attTermCallsUmts";
const string succTermCallsUmts = "succTermCallsUmts";
const string ansTermCallsUmts = "ansTermCallsUmts";
const string failTermCallsUmtsPerCause = "failTermCallsUmtsPerCause";
const string attTermCallTrafficUmts = "attTermCallTrafficUmts";
const string succTermCallTrafficUmts = "succTermCallTrafficUmts";
const string ansTermCallTrafficUmts = "ansTermCallTrafficUmts";
//incoming call
const string attIncCalls = "attIncCalls";
const string succIncCalls = "succIncCalls";
const string ansIncCalls= "ansIncCalls";
const string failIncCallsPerCause = "failIncCallsPerCause";
const string attIncCallTraffic = "attIncCallTraffic";
const string succIncCallTraffic = "succIncCallTraffic";
const string ansIncCallTraffic = "ansIncCallTraffic";
//outgoing call
const string attOutCalls = "attOutCalls";
const string succOutCalls= "succOutCalls";
const string ansOutCalls= "ansOutCalls";
const string failOutCallsPerCause = "failOutCallsPerCause";
const string attOutCallTraffic = "attOutCallTraffic";
const string succOutCallTraffic = "succOutCallTraffic";
const string ansOutCallTraffic = "ansOutCallTraffic";
//transit call
const string attTransCalls = "attTransCalls";
const string succTransCalls= "succTransCalls";
const string ansTransCalls= "ansTransCalls";
const string failTransCallsPerCause = "failTransCallsPerCause";
const string attTransCallTraffic = "attTransCallTraffic";
const string succTransCallTraffic = "succTransCallTraffic";
const string ansTransCallTraffic = "ansTransCallTraffic";
//originating outgoing call
const string attOrigOutCalls = "attOrigOutCalls";
const string succOrigOutCalls= "succOrigOutCalls";
const string ansOrigOutCalls= "ansOrigOutCalls";
```

```

const string failOrigOutCallsPerCause = "failOrigOutCallsPerCause";
const string attOrigOutCallTraffic = "attOrigOutCallTraffic";
const string succOrigOutCallTraffic = "succOrigOutCallTraffic";
const string ansOrigOutCallTraffic = "ansOrigOutCallTraffic";
//terminating incoming call
const string attTermIncCalls = "attTermIncCalls";
const string succTermIncCalls= "succTermIncCalls";
const string ansTermIncCalls= "ansTermIncCalls";
const string failTermIncCallsPerCause = "failTermIncCallsPerCause";
const string attTermIncCallTraffic = "attTermIncCallTraffic";
const string succTermIncCallTraffic = "succTermIncCallTraffic";
const string ansTermIncCallTraffic = "ansTermIncCallTraffic";
//abnormal call
const string nbrCallsBlockedByLoadShedding = "nbrCallsBlockedByLoadShedding";
const string nbrCallsBlockedByInternalCongestion = "nbrCallsBlockedByInternalCongestion";
const string nbrCallsBlockedByOutCircuitBusy= "nbrCallsBlockedByOutCircuitBusy";
};

//circuit end point subgroup measurement
module circuitEndpointSubGroup
{
    //outgoing circuit end subgroup
    const string outBids= "outBids";
    const string succOutSeizures= "succOutSeizures";
    const string succOutCalls = "succOutCalls";
    const string ansOutCalls = "ansOutCalls";
    const string failOutCallsByOverflow = "failOutCallsByOverflow";
    const string failOutCallsByUserBusy = "failOutCallsByUserBusy";
    const string failOutCallsByNoAns = "failOutCallsByNoAns";
    const string failOutCallsByUnallNum = "failOutCallsByUnallNum";
    const string failOutCallsByCongestion = "failOutCallsByCongestion";
    const string succOutSeizureTraffic = "succOutSeizureTraffic";
    const string ansOutCallTraffic = "ansOutCallTraffic";
    //incoming circuit end subgroup
    const string succIncSeizures = "succIncSeizures";
    const string succIncCalls = "succIncCalls";
    const string ansIncSeizures = "ansIncSeizures";
}

```

```

const string failIncCallsByUserBusy = "failIncCallsByUserBusy";
const string failIncCallsByNoAns = "failIncCallsByNoAns";
const string failIncCallsByUnallNum = "failIncCallsByUnallNum";
const string failIncCallsByCongestion = "failIncCallsByCongestion";
const string succIncSeizureTraffic = "succIncSeizureTraffic";
const string ansIncSeizureTraffic = "ansIncSeizureTraffic";
const string nbrAvailTrunks = "nbrAvailTrunks";
};

//No.7 signalling measurement
module mtp3SignallingLinkTP
{
    const string durSigLinkOutOfService= "durSigLinkOutOfService";
    const string nbrSigLinkOutOfService= "nbrSigLinkOutOfService";
    const string nbrSentMsus = "nbrSentMsus";
    const string nbrSentSifsAndSios = "nbrSentSifsAndSios";
    const string nbrRecvedMsus = "nbrRecvedMsus";
    const string nbrRecvedSifsAndSios = "nbrRecvedSifsAndSios";
};

module mtp3SignallingLinkSetTP
{
    const string nbrSigLinkSetOutOfService= "nbrSigLinkSetOutOfService";
    const string durSigLinkSetOutOfService= "durSigLinkSetOutOfService";
    const string nbrAvailSiglinks= "nbrAvailSiglinks";
};

module mtp3bSignallingLinkTP
{
    const string durSigLinkOutOfService= "durSigLinkOutOfService";
    const string nbrSigLinkOutOfService= "nbrSigLinkOutOfService";
    const string nbrSentMsus = "nbrSentMsus";
    const string nbrSentSifsAndSios = "nbrSentSifsAndSios";
    const string nbrRecvedMsus = "nbrRecvedMsus";
    const string nbrRecvedSifsAndSios = "nbrRecvedSifsAndSios";
};

```

```

module mtp3bSignallingLinkSetTP
{
    const string nbrSigLinkSetOutOfService= "nbrSigLinkSetOutOfService";
    const string durSigLinkSetOutOfService= "durSigLinkSetOutOfService";
    const string nbrAvailSiglinks= "nbrAvailSiglinks";
};

// observed destination measurement
module observedDestination
{
    const string bids= "bids";
    const string nbrNoAvailCircuits= "nbrNoAvailCircuits";
    const string succCalls = "succCalls";
    const string ansCalls = "ansCalls";
    const string succCallTraffic = "succCallTraffic";
    const string ansCallTraffic = "ansCallTraffic";
};

//VLR measurement
module vlrBasicMeasurement
{
    //identification request to PVLR
    const string attIdentReqsToPVlr= "attIdentReqsToPVlr";
    const string succIdentReqsToPVlr= "succIdentReqsToPVlr";
    //location update
    const string attLusIntraVlr= "attLusIntraVlr";
    const string succLusIntraVlr= "succLusIntraVlr";
    const string attLusInterVlr= "attLusInterVlr";
    const string succLusInterVlr= "succLusInterVlr";
    //request for authentication set to HLR
    const string attReqsForAuthSetsSentToHlr= "attReqsForAuthSetsSentToHlr";
    const string succRecvedAuthSetsFromHlr= "succRecvedAuthSetsFromHlr";
    const string succReqAuthSetWithQuintupletsFromHlr = "succReqAuthSetWithQuintupletsFromHlr";
    const string succReqAuthSetWithTripletsFromHlr = "succReqAuthSetWithTripletsFromHlr";
    //subscriber data administration
    const string succInsertSubsData= "succInsertSubsData";
    const string succDelSubsData= "succDelSubsData";
};

```



```

//provide roaming number to HLR
const string attProvideRoamingNumber= "attProvideRoamingNumber";
const string succProvideRoamingNumber= "succProvideRoamingNumber";
};

module vlrSubscriberData
{
    const string nbrCurrentSubsInVlrPerHlr= "nbrCurrentSubsInVlrPerHlr";
    const string nbrCurrentSubsWithPowerOnInVlr = "nbrCurrentSubsWithPowerOnInVlr";
    const string nbrRoamingSubs= "nbrRoamingSubs";
    const string nbrRoamingSubsInternational= "nbrRoamingSubsInternational";
};

//HLR measurement
module hlrBasicMeasurement
{
    const string attGetRoutingInfo= "attGetRoutingInfo";
    const string succGetRoutingInfo = "succGetRoutingInfo";
    const string attProvideRoamingNumber = "attProvideRoamingNumber";
    const string succProvideRoamingNumber = "succProvideRoamingNumber";
    const string attLocationUpdates = "attLocationUpdates";
    const string succLocationUpdates = "succLocationUpdates";
    const string attCancelLocation = "attCancelLocation";
    const string succCancelLocation = "succCancelLocation";
    const string attInsertSubsData = "attInsertSubsData";
    const string succInsertSubsData = "succInsertSubsData";
    const string attDeleteSubsData = "attDeleteSubsData";
    const string succDeleteSubsData = "succDeleteSubsData";
    const string attSendAuthInfo = "attSendAuthInfo";
    const string succSendAuthInfo = "succSendAuthInfo";
    const string nbrReset = "nbrReset";
    const string attRestoreData = "attRestoreData";
    const string succRestoreData = "succRestoreData";
};

module hlrSubscriberData
{

```

```
const string nbrCurrentSubsWithPowerOnInHlrPerVlr = "nbrCurrentSubsWithPowerOnInHlrPerVlr";
const string nbrCurrentSubsInHlr = "nbrCurrentSubsInHlr";
const string nbrCurrentMsisdnInHlr = "nbrCurrentMsisdnInHlr";
};

module hlrSmServiceMeasurement
{
    const string attSendRoutingInfoForSm = "attSendRoutingInfoForSm";
    const string succSendRoutingInfoForSm = "succSendRoutingInfoForSm";
    const string nbrAlertServiceCentre= "nbrAlertServiceCentre";
    const string nbrInformServiceCenter= "nbrInformServiceCenter";
    const string nbrReadyForSm= "nbrReadyForSm";
};

module hlrSupplementServiceMeasurement
{
    const string attRegisterSs = "attRegisterSs";
    const string succRegisterSs = "succRegisterSs";
    const string attEraseSs = "attEraseSs";
    const string succEraseSs = "succEraseSs";
    const string attActSs = "attActSs";
    const string succActSs = "succActSs";
    const string attDeactSs = "attDeactSs";
    const string succDeactSs = "succDeactSs";
};

module hlrInServiceMeasurement
{
    const string attAnyTimeInterrogation = "attAnyTimeInterrogation";
    const string succAnyTimeInterrogation = "succAnyTimeInterrogation";
    const string attAnyTimeSubsInterrogation = "attAnyTimeSubsInterrogation";
    const string succAnyTimeSubsInterrogation = "succAnyTimeSubsInterrogation";
    const string attAnyTimeModification = "attAnyTimeModification";
    const string succAnyTimeModification = "succAnyTimeModification";
    const string nbrNoteSubsDataModified = "nbrNoteSubsDataModified";
};
```

```
module hlrPacketServiceMeasurement
```

```
{
    const string attSendRoutingInfoForGprs = "attSendRoutingInfoForGprs";
    const string succSendRoutingInfoForGprs = "succSendRoutingInfoForGprs";
    const string nbrFailReport= "nbrFailReport";
    const string nbrNoteMsPresentForGprs= "nbrNoteMsPresentForGprs";
    const string attUpdateGprsLocation = "attUpdateGprsLocation";
    const string succUpdateGprsLocation = "succUpdateGprsLocation";
};
```

```
module hlrLocationServiceMeasurement
```

```
{
    const string attSendRoutingInfoForLcs = "attSendRoutingInfoForLcs";
    const string succSendRoutingInfoForLcs = "succSendRoutingInfoForLcs";
};
```

```
//EIR measurement
```

```
module eirBasicMeasurement
```

```
{
    const string nbrCurrentWhiteSubsInEir= "nbrCurrentWhiteSubsInEir";
    const string nbrCurrentBlackSubsInEir= "nbrCurrentBlackSubsInEir";
    const string nbrCurrentGreySubsInEir= "nbrCurrentGreySubsInEir";
};
```

```
//SGSN measurement
```

```
module sessionManagementMeasurement
```

```
{
    //active PDP context by MS
    const string attActPdpContextMs= "attActPdpContextMs";
    const string succActPdpContextMs= "succActPdpContextMs";
    const string failActPdpContextMsPerCause = "failActPdpContextMsPerCause";

    //active PDP context by network
    const string attActPdpContextNetwork= "attActPdpContextNetwork";
    const string succActPdpContextNetwork= "succActPdpContextNetwork";
    const string failActPdpContextNetworkPerCause= "failActPdpContextNetworkPerCause";
};
```

```
//active dynamic PDP context by MS
const string attActPdpContextDynMs= "attActPdpContextDynMs";
const string succActPdpContextDynMs= "succActPdpContextDynMs";

//number of subscriber with actived PDP context
const string meanSubsWithActPdpContext= "meanSubsWithActPdpContext";
const string maxSubsWithActPdpContext= "maxSubsWithActPdpContext";

//mean number of acti ed PDP context
const string meanActPDPContext= "meanActPDPContext";
const string maxActPdpContext= "maxActPdpContext";

//deactive PDP context by SGSN
const string attDeactPdpContextSgsn= "attDeactPdpContextSgsn";
const string succDeactPdpContextSgsn= "succDeactPdpContextSgsn";

//deactive PDP context by MS
const string attDeactPdpContextMs= "attDeactPdpContextMs";
const string succDeactPdpContextMs= "succDeactPdpContextMs";

//deactive PDP context by GGSN
const string attDeactPdpContextGgsn= "attDeactPdpContextGgsn";
const string succDeactPdpContextGgsn= "succDeactPdpContextGgsn";

//active secondary PDP context
const string attActSecondPdpContext= "attActSecondPdpContext";
const string succActSecondPdpContext= "succActSecondPdpContext";

//modify PDP context by MS
const string attModPdpContextMs= "attModPdpContextMs";
const string succModPdpContextMs= "succModPdpContextMs";

//modify PDP context by SGSN
const string attModPdpContextSgsn= "attModPdpContextSgsn";
const string succModPdpContextSgsn= "succModPdpContextSgsn";
```

```

//update PDP context by GGSN
const string attUpdPdpContextGgsn= "attUpdPdpContextGgsn";
const string succUpdPdpContextGgsn= "succUpdPdpContextGgsn";

//update PDP context by SGSN
const string attUpdPdpContextSgsn= "attUpdPdpContextSgsn";
const string succUpdPdpContextSgsn= "succUpdPdpContextSgsn";
};

module subscriberManagementMeasurement
{
    //subscriber state
    const string meanStandbySubs= "meanStandbySubs";
    const string maxStandbySubs= "maxStandbySubs";
    const string meanReadySubs= "meanReadySubs";
    const string maxReadySubs= "maxReadySubs";
    const string meanPmmIdleSubs= "meanPmmIdleSubs";
    const string maxPmmIdleSubs= "maxPmmIdleSubs";
    const string meanPmmConnectedSubs= "meanPmmConnectedSubs";
    const string maxPmmConnectedSubs= "maxPmmConnectedSubs";

    //number of attached subscribers
    const string meanAttachedSubsPerRa = "meanAttachedSubsPerRa";
    const string maxAttachedSubsPerRa= "maxAttachedSubsPerRa";
};

module mobileManagementMeasurement
{
    //GPRS attach
    const string attGprsAttachGsm= "attGprsAttach";
    const string succGprsAttach= "succGprsAttach";
    const string failGprsAttachPerCause= "failGprsAttachPerCause";

    //combined GPRS/IMSI attach
    const string attCombiAttach= "attCombiAttach";
    const string succCombiAttach= "succCombiAttach";
    const string failCombiAttachPerCause= "failCombiAttachPerCause";
};

```

```
//GPRS attach with IMSI already attached
const string attGprsAttachWithImsiAttached= "attGprsAttachWithImsiAttached";
const string succGprsAttachWithImsiAttached= "succGprsAttachWithImsiAttached";
const string failGprsAttachWithImsiAttachedPerCause= "failGprsAttachWithImsiAttachedPerCause";

//GPRS detach by MS
const string attGprsDetachMs= "attGprsDetachMs";
//combined GPRS/IMSI detach by MS
const string attCombiDetachMs= "attCombiDetachMs";
//IMSI detach by MS
const string attImsiDetachMs= "attImsiDetachMs";
//GPRS detach by SGSN
const string attGprsDetachSgsn= "attGprsDetachSgsn";
const string succGprsDetachSgsn= "succGprsDetachSgsn";
//GPRS detach by HLR
const string attGprsDetachHlr= "attGprsDetachHlr";
//intra-SGSN routing area update
const string attIntraSgsnRaUpdate= "attIntraSgsnRaUpdate";
const string succIntraSgsnRaUpdate= "succIntraSgsnRaUpdate";
const string failIntraSgsnRaUpdatePerCause= "failIntraSgsnRaUpdatePerCause";

//combined RA/LA intra SGSN routing area update
const string attCombiIntraSgsnRaUpdate= "attCombiIntraSgsnRaUpdate";
const string succCombiIntraSgsnRaUpdate= "succCombiIntraSgsnRaUpdate";
const string failCombiIntraSgsnRaUpdatePerCause= "failCombiIntraSgsnRaUpdatePerCause";

//inter SGSN routing area update
const string attInterSgsnRaUpdate= "attInterSgsnRaUpdate";
const string succInterSgsnRaUpdate= "succInterSgsnRaUpdate";
const string failInterSgsnRaUpdatePerCause= "failInterSgsnRaUpdatePerCause";

//combined RA/LA inter SGSN routing area update
const string attCombiInterSgsnRaUpdate= "attCombiInterSgsnRaUpdate";
const string succCombiInterSgsnRaUpdate= "succCombiInterSgsnRaUpdate";
const string failCombiInterSgsnRaUpdatePerCause= "failCombiInterSgsnRaUpdatePerCause";
```

```

//PS paging
const string nbrPsPagingGsm= "nbrPsPagingGsm";
const string nbrPsPagingNoRspGsm= "nbrPsPagingNoRspGsm";
const string nbrPsPagingUmts= "nbrPsPagingUmts";
const string nbrPsPagingNoRspUmts= "nbrPsPagingNoRspUmts";
};

module sgsnRelocationMeasurement
{
    //inter SGSN relocation
    const string attInterSgsnReloc= "attInterSgsnReloc";
    const string succInterSgsnReloc= "succInterSgsnReloc";
    //inter SGSN combined relocation and hard handover
    const string attInterSgsnCombiReloc= "attInterSgsnCombiReloc";
    const string succInterSgsnCombiReloc= "succInterSgsnCombiReloc";
    //intra SGSN relocation
    const string attIntraSgsnReloc= "attIntraSgsnReloc";
    const string succIntraSgsnReloc= "succIntraSgsnReloc";
    //intra SGSN combined relocation and hard handover
    const string attIntraSgsnCombiReloc= "attIntraSgsnCombiReloc";
    const string succIntraSgsnCombiReloc= "succIntraSgsnCombiReloc";
};

module sgsnInterSystemHandoverMeasurement
{
    //intra SGSN handover from GSM to UMTS
    const string attIntraSgsnHoGsmToUmts= "attIntraSgsnHoGsmToUmts";
    const string succIntraSgsnHoGsmToUmts= "succIntraSgsnHoGsmToUmts";
    //intra SGSN handover from UMTS to GSM
    const string attIntraSgsnHoUmtsToGsm= "attIntraSgsnHoUmtsToGsm";
    const string succIntraSgsnHoUmtsToGsm= "succIntraSgsnHoUmtsToGsm";
    //inter SGSN handover from GSM to UMTS
    const string attInterSgsnHoGsmToUmts= "attInterSgsnHoGsmToUmts";
    const string succInterSgsnHoGsmToUmts= "succInterSgsnHoGsmToUmts";
    //inter SGSN handover from UMTS to GSM
    const string attInterSgsnHoUmtsToGsm= "attInterSgsnHoUmtsToGsm";
    const string succInterSgsnHoUmtsToGsm= "succInterSgsnHoUmtsToGsm";
};

```

```
};

module mapServiceMeasurement
{
    //request authentication set
    const string attReqAuthSetHlr= "attReqAuthSetHlr";
    const string succReqAuthSetWithQuintupletsHlr= "succReqAuthSetWithQuintupletsHlr";
    const string succReqAuthSetWithTripletsHlr= "succReqAuthSetWithTripletsHlr";
    const string succReqAuthSetWithEmptyRspHlr= "succReqAuthSetWithEmptyRspHlr";
    //GPRS update location
    const string attUpdateGprsLocationHlr= "attUpdateGprsLocationHlr";
    const string succUpdateGprsLocationHlr= "succUpdateGprsLocationHlr";
    //insert and delete subscriber data
    const string attInsertSubsDataHlr= "attInsertSubsDataHlr";
    const string attDeleteSubsDataHlr= "attDeleteSubsDataHlr";
};

module securityManagementMeasurement
{
    //P-TMSI reallocate
    const string attPtmsiRealloc= "attPtmsiRealloc";
    const string succPtmsiRealloc= "succPtmsiRealloc";

    //authentication request
    const string attAuthReq= "attAuthReq";
    const string succAuthReq= "succAuthReq";

    //identity request
    const string attIdentReq= "attIdentReq";
    const string succIdentReq= "succIdentReq";

    //security mode setup
    const string attSecMode= "attSecMode";
    const string succSecMode= "succSecMode";
};

module gtpInGnGpMeasurement
```



```

{
    //number of incoming and outgoing signalling packets
    const string nbrIncGtpCSigPkts= "nbrIncGtpCSigPkts";
    const string nbrOutGtpCSigPkts= "nbrOutGtpCSigPkts";

    //number of octets of incoming and outgoing signalling packets
    const string nbrIncGtpCSigOcts= "nbrIncGtpCSigOcts";
    const string nbrOutGtpCSigOcts= "nbrOutGtpCSigOcts";

    //number of abnormal signalling packets
    const string nbrOutGtpCSigPktsOverflow= "nbrOutGtpCSigPktsOverflow";
    const string nbrIncGtpCSigPktsError= "nbrIncGtpCSigPktsError";

    //number of incoming and outgoing data packets
    const string nbrIncGtpUDDataPkts= "nbrIncGtpUDDataPkts";
    const string nbrOutGtpUDDataPkts= "nbrOutGtpUDDataPkts";

    //number of octets of incoming and outgoing data packets
    const string nbrIncGtpUDDataOcts= "nbrIncGtpUDDataOcts";
    const string nbrOutGtpUDDataOcts= "nbrOutGtpUDDataOcts";

    //number of abnormal data packets
    const string nbrOutGtpUDDataPktsOverflow= "nbrOutGtpUDDataPktsOverflow";
    const string nbrIncGtpUDDataPktsError= "nbrIncGtpUDDataPktsError";
};

```

```

module shortMessageServiceMeasurement

```

```

{
    //originating short messages
    const string attOrigSmsPs= "attOrigSmsPs";
    const string succOrigSmsPs= "succOrigSmsPs";

    //terminating short messages
    const string attTermSmsPs= "attTermSmsPs";
    const string succTermSmsPs= "succTermSmsPs";

    //MS present for short message

```

```

const string attSmsMsPresentPs= "attSmsMsPresentPs";
const string succSmsMsPresentPs= "succSmsMsPresentPs";

//memory available for short message
const string attSmsMemoryAvaiPs= "attSmsMemoryAvaiPs";
const string succSmsMemoryAvailPs= "succSmsMemoryAvailPs";
};

//GGSN measurement
module ggsnThroughputMeasurement
{
    //number of incoming and outgoing data packets in Gn/Gp interface
    const string nbrIncDataPktsGnGp= "nbrIncDataPktsGnGp";
    const string nbrOutDataPktsGnGp= "nbrOutDataPktsGnGp";
    const string nbrIncDataOctsGnGp = "nbrIncDataOctsGnGp";
    const string nbrOutDataOctsGnGp = "nbrOutDataOctsGnGp";
    //number of incoming and outgoing signalling packets in Gn/Gp interface
    const string nbrIncSigPktsGnGp= "nbrIncSigPktsGnGp";
    const string nbrOutSigPktsGnGp= "nbrOutSigPktsGnGp";
    const string nbrIncSigOctsGnGp = "nbrIncSigOctsGnGp";
    const string nbrOutSigOctsGnGp = "nbrOutSigOctsGnGp";
    //number of incoming and outgoing data packets in Gi interface
    const string nbrOutDataPktsGi= "nbrOutDataPktsGi";
    const string nbrIncDataPktsGi= "nbrIncDataPktsGi";
    const string nbrOutDataOctsGi = "nbrOutDataOctsGi";
    const string nbrIncDataOctsGi = "nbrIncDataOctsGi";
};

//GGSN APN measurement
module apnSessionManagementMeasurement
{
    //active PDP context
    const string attActPdpContext= "attActPdpContext";
    const string succActPdpContext= "succActPdpContext";
    const string failActPdpContextUmtsPerCause = "failActPdpContextUmtsPerCause";
    //active dynamic PDP context

```

```

const string attDynActPdpContext= "attDynActPdpContext";
const string succDynActPdpContext= "succDynActPdpContext";
//active PDP context with QoS parameter
const string succActPdpContextQos= "succActPdpContextQos";
//fail to active PDP context because of no resource available
const string failActPdpContextNoResource= "failActPdpContextNoResource";
//deactive PDP context by MS
const string attDeactPdpContextMs= "attDeactPdpContextMs";
const string succDeactPdpContextMs= "succDeactPdpContextMs";
//Deactive PDP context by GGSN
const string attDeactPdpContextGgsn= "attDeactPdpContextGgsn";

const string succDeactPdpContextGgsn= "succDeactPdpContextGgsn";
//number of activated PDP context by GGSN
const string nbrActPdpContexts= "nbrActPdpContexts";
const string meanActPdpContexts= "meanActPdpContexts";
const string maxActPdpContexts= "maxActPdpContexts";
};

```

```

module apnThroughputMeasurement

```

```

{
const string nbrIncDataPktsGnGp = "nbrIncDataPktsGnGp";
const string nbrOutDataPktsGnGp = "nbrOutDataPktsGnGp";
const string nbrIncDataOctsGnGp = "nbrIncDataOctsGnGp";
const string nbrOutDataOctsGnGp = "nbrOutDataOctsGnGp";
const string nbrIncSigPktsGnGp = "nbrIncSigPktsGnGp";
const string nbrOutSigPktsGnGp = "nbrOutSigPktsGnGp";
const string nbrIncSigOctsGnGp = "nbrIncSigOctsGnGp";
const string nbrOutSigOctsGnGp = "nbrOutSigOctsGnGp";
const string nbrOutDataPktsGi = "nbrOutDataPktsGi";
const string nbrIncDataPktsGi = "nbrIncDataPktsGi";
const string nbrOutDataOctsGi = "nbrOutDataOctsGi";
const string nbrIncDataOctsGi = "nbrIncDataOctsGi";
};

```

```

//RNC measurement

```

```

module rabAssignmentMeasurement
{
    //establish 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";
    //modify RAB
    const string attRabAssignModCsPerType= "attRabAssignModCsPerType";
    const string succRabAssignModCsPerType= "succRabAssignModCsPerType";
    const string failRabAssignModCsPerCause= "failRabAssignModCsPerCause";
    const string attRabAssignModPsPerType= "attRabAssignModPsPerType";
    const string succRabAssignModPsPerType= "succRabAssignModPsPerType";
    const string failRabAssignModPsPerCause= "failRabAssignModPsPerCause";
    //release RAB
    const string attRabAssignRelCsPerType= "attRabAssignRelCsPerType";
    const string succRabAssignRelCsPerType= "succRabAssignRelCsPerType";
    const string failRabAssignRelCsPerCause= "failRabAssignRelCsPerCause";
    const string attRabAssignRelPsPerType= "attRabAssignRelPsPerType";
    const string succRabAssignRelPsPerType= "succRabAssignRelPsPerType";
    const string failRabAssignRelPsPerCause= "failRabAssignRelPsPerCause";
};

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";
    //Iu interface overload control
    const string nbrOverloadControlCsByRnc= "nbrOverloadControlCsByRnc";
    const string nbrOverloadControlPsByRnc= "nbrOverloadControlPsByRnc";
    const string nbrOverloadControlCsByCn= "nbrOverloadControlCsByCn";
    const string nbrOverloadControlPsByCn= "nbrOverloadControlPsByCn";
    //Iu interface error indication
    const string nbrErrorIndCsByRncPerCause= "nbrErrorIndCsByRncPerCause";
    const string nbrErrorIndPsByRncPerCause= "nbrErrorIndPsByRncPerCause";
    const string nbrErrorIndCsByCnPerCause= "nbrErrorIndCsByCnPerCause";
    const string nbrErrorIndPsByCnPerCause= "nbrErrorIndPsByCnPerCause";
};

module rncSoftHandoverMeasurement
{
    const string attRIAddInSho= "attRIAddInSho";
    const string failRIAddInShoPerCause= "failRIAddInShoPerCause";
    const string attRIDelInSho= "attRIDelInSho";
    const string failRIDelInShoPerCause= "failRIDelInShoPerCause";
};

```

```

};

module rncHardHandoverMeasurement
{
    const string attHho= "attHho";
    const string failHhoPerCause= "failHhoPerCause";
};

module rncRelocationMeasurement
{
    //CS relocation out with UE not involved
    const string attRelocOutPrepWithUeNotInvCsPerCause= "attRelocOutPrepWithUeNotInvCsPerCause";
    const string failRelocOutPrepWithUeNotInvCsPerCause= "failRelocOutPrepWithUeNotInvCsPerCause";
    const string attRelocOutWithUeNotInvCs= "attRelocOutWithUeNotInvCs";
    const string failRelocOutWithUeNotInvCsPerCause= "failRelocOutWithUeNotInvCsPerCause";
    //CS relocation out with UE involved
    const string attRelocOutPrepWithUeInvCsPerCause= "attRelocOutPrepWithUeInvCsPerCause";
    const string failRelocOutPrepWithUeInvCsPerCause= "failRelocOutPrepWithUeInvCsPerCause";
    const string attRelocOutWithUeInvCs= "attRelocOutWithUeInvCs";
    const string failRelocOutWithUeInvCsPerCause= "failRelocOutWithUeInvCsPerCause";
    //PS relocation out with UE not involved
    const string attRelocOutPrepWithUeNotInvPsPerCause= "attRelocOutPrepWithUeNotInvPsPerCause";
    const string failRelocOutPrepWithUeNotInvPsPerCause= "failRelocOutPrepWithUeNotInvPsPerCause";
    const string attRelocOutWithUeNotInvPs= "attRelocOutWithUeNotInvPs";
    const string failRelocOutWithUeNotInvPsPerCause= "failRelocOutWithUeNotInvPsPerCause";
    //PS relocation out with UE involved
    const string attRelocOutPrepWithUeInvPsPerCause= "attRelocOutPrepWithUeInvPsPerCause";
    const string failRelocOutPrepWithUeInvPsPerCause= "failRelocOutPrepWithUeInvPsPerCause";
    const string attRelocOutWithUeInvPs= "attRelocOutWithUeInvPs";
    const string failRelocOutWithUeInvPsPerCause= "failRelocOutWithUeInvPsPerCause";
    //CS relocation in with UE not involved
    const string attRelocInWithUeNotInvCsPerCause= "attRelocInWithUeNotInvCsPerCause";
    const string failRelocInWithUeNotInvCsPerCause= "failRelocInWithUeNotInvCsPerCause";
    //CS relocation in with UE involved
    const string attRelocInWithUeInvCsPerCause= "attRelocInWithUeInvCsPerCause";
    const string failRelocInWithUeInvCsPerCause= "failRelocInWithUeInvCsPerCause";
    //PS relocation in with UE not involved

```

```

const string attRelocInWithUeNotInvPsPerCause= "attRelocInWithUeNotInvPsPerCause";
const string failRelocInWithUeNotInvPsPerCause= "failRelocInWithUeNotInvPsPerCause";
//PS relocation in with UE involved
const string attRelocInWithUeInvPsPerCause= "attRelocInWithUeInvPsPerCause";
const string failRelocInWithUeInvPsPerCause= "failRelocInWithUeInvPsPerCause";
};

```

```

module mcInterSystemHandoverMeasurement

```

```

{
    //CS inter system handover from WCDMA to GSM
    const string attRelocOutPrepInterSysCsPerCause= "attRelocOutPrepInterSysCsPerCause";
    const string failRelocOutPrepInterSysCsPerCause= "failRelocOutPrepInterSysCsPerCause";
    const string attRelocOutInterSysCs= "attRelocOutInterSysCs";
    const string failRelocOutInterSysCsPerCause= "failRelocOutInterSysCsPerCause";
    //CS inter system handover from GSM to WCDMA
    const string attRelocInInterSysCsPerCause= "attRelocInInterSysCsPerCause";
    const string failRelocInInterSysCsPerCause= "failRelocInInterSysCsPerCause";
    //PS inter system handover from WCDMA to GPRS
    //const string attRelocOutPrepInterSysPsPerCause= "attRelocOutPrepInterSysPsPerCause";
    //const string failRelocOutPrepInterSysPsPerCause= "failRelocOutPrepInterSysPsPerCause";
    const string attRelocOutInterSysPs= "attRelocOutInterSysPs";
    const string failRelocOutInterSysPsPerCause= "failRelocOutInterSysPsPerCause";
    //PS inter system handover from GPRS to WCDMA
    const string attRelocInInterSysPs= "attRelocInInterSysPs";
    const string succRelocInInterSysPs= "succRelocInInterSysPs";
};

```

```

module iuInterfaceThroughputMeasurement

```

```

{
    const string iuUISigThroughputCs = "iuUISigThroughputCs";
    const string iuDISigThroughputCs = "iuDISigThroughputCs";
    const string iuUIDataThroughputCsPerType = "iuUIDataThroughputCsPerType";
    const string iuDIDataThroughputCsPerType = "iuDIDataThroughputCsPerType";
    const string iuUISigThroughputPs = "iuUISigThroughputPs";
    const string iuDISigThroughputPs = "iuDISigThroughputPs";
    const string iuUIDataThroughputPsPerType = "iuUIDataThroughputPsPerType";
    const string iuDIDataThroughputPsPerType = "iuDIDataThroughputPsPerType";
};

```

```

};

module iurInterfaceThroughputMeasurement
{
    const string iurUISigThroughput = "iurUISigThroughput";
    const string iurDISigThroughput = "iurDISigThroughput";
    const string iurUIDataThroughput = "iurUIDataThroughput";
    const string iurDIDataThroughput = "iurDIDataThroughput";
};

module rlcConnectionMeasurement
{
    const string nbrRlcBlockSentPerMode = "nbrRlcBlockSentPerMode";
    const string nbrRlcBlockRecvPerMode = "nbrRlcBlockRecvPerMode";
    const string nbrDiscardedRlcBlocksByRnc = "nbrDiscardedRlcBlocksByRnc";
    const string nbrRetransmittedRlcBlocksToUe = "nbrRetransmittedRlcBlocksToUe";
};

//UtranCell measurement

module cellRrcConnectionMeasurement
{
    const string attRrcConnSetupPerCause = "attRrcConnSetupPerCause";
    const string succRrcConnSetupPerCause = "succRrcConnSetupPerCause";
    const string failRrcConnSetupPerCause= "failRrcConnSetupPerCause";
    const string attRrcConnReestab= "attRrcConnReestab";
    const string failRrcConnReestabPerCause= "failRrcConnReestabPerCause";
};

module cellSoftHandoverMeasurement
{
    //soft handover
    const string attRlAddInSho= "attRlAddInSho";
    const string failRlAddInShoPerCause= "failRlAddInShoPerCause";
    const string attRlDelInSho= "attRlDelInSho";
    const string failRlDelInShoPerCause= "failRlDelInShoPerCause";
};

```



```
module hardHandoverIntraCellMeasurement
{
    const string attHhoOutIntraCell= "attHhoOutIntraCell";
    const string failHhoOutIntraCellPerCause= "failHhoOutIntraCellPerCause";
};
```

```
module iurRlManagementMeasurement
{
    const string attRlSetupIub= "attRlSetupIub";
    const string failRlSetupIubPerCause= "failRlSetupIubPerCause";
    const string attRlAddIub= "attRlAddIub";
    const string failRlAddIubPerCause= "failRlAddIubPerCause";
    const string attRlDelIub= "attRlDelIub";
    const string succRlDelIub= "succRlDelIub";
};
```

```
module iurRlManagementMeasurement
{
    const string attRlSetupIur= "attRlSetupIur";
    const string failRlSetupIurPerCause= "failRlSetupIurPerCause";
    const string attRlAddIur= "attRlAddIur";
    const string failRlAddIurPerCause= "failRlAddIurPerCause";
    const string attRlDelIur= "attRlDelIur";
    const string succRlDelIur= "succRlDelIur";
};
```

```
module cellTrafficMeasurement
{
    const string cellCcchTraffic = "cellCcchTraffic";
    const string cellCtchTraffic = "cellCtchTraffic";
    const string cellDcchTraffic = "cellDcchTraffic";
    const string cellDtchTraffic = "cellDtchTraffic";
};
```

```
module cellPagingMeasurement
{
```

```

const string attPagingType1FromUtran = "attPagingType1FromUtran";
const string succPagingType1FromUtran = "succPagingType1FromUtran";
const string attPagingType2FromUtran = "attPagingType2FromUtran";
};

//UtranRelation measurement
module hardHandoverInterCellIntraNodeBMeasurement
{
    const string attHhoOutInterCellIntraNodeB= "attHhoOutInterCellIntraNodeB";
    const string failHhoOutInterCellIntraNodeBPerCause= "failHhoOutInterCellIntraNodeBPerCause";
};

module hardHandoverInterNodeBIntraRncMeasurement
{
    const string attHhoOutInterNodeBIntraRnc= "attHhoOutInterNodeBIntraRnc";
    const string failHhoOutInterNodeBIntraRncPerCause= "failHhoOutInterNodeBIntraRncPerCause";
};

module hardHandoverInterRncViaIurMeasurement
{
    const string attHhoOutInterRncViaIur= "attHhoOutInterRncViaIur";
    const string failHhoOutInterRncViaIurPerCause= "failHhoOutInterRncViaIurPerCause";
};

module hardHandoverInterRncMeasurement
{
    const string attHhoOutInterRncCn = "attHhoOutInterRncCn";
    const string failHhoOutInterRncCnPerCause = "failHhoOutInterRncCnPerCause";
};

module hardHandoverInterSystemMeasurement
{
    //CS inter system handover from WCDMA to GSM
    const string attRelocOutPrepInterSysCsPerCause= "attRelocOutPrepInterSysCsPerCause";
    const string failRelocOutPrepInterSysCsPerCause= "failRelocOutPrepInterSysCsPerCause";
    const string attRelocOutInterSysCs= "attRelocOutInterSysCs";
    const string failRelocOutInterSysCsPerCause= "failRelocOutInterSysCsPerCause";
};

```

```

//CS inter system handover from GSM to WCDMA
const string attRelocInInterSysCsPerCause= "attRelocInInterSysCsPerCause";
const string failRelocInInterSysCsPerCause= "failRelocInInterSysCsPerCause";
//PS inter system handover from WCDMA to GPRS
//const string attRelocOutPrepInterSysPsPerCause= "attRelocOutPrepInterSysPsPerCause";
//const string failRelocOutPrepInterSysPsPerCause= "failRelocOutPrepInterSysPsPerCause";
const string attRelocOutInterSysPs= "attRelocOutInterSysPs";
const string failRelocOutInterSysPsPerCause= "failRelocOutInterSysPsPerCause";
//PS inter system handover from GPRS to WCDMA
const string attRelocInInterSysPs= "attRelocInInterSysPs";
const string succRelocInInterSysPs= "succRelocInInterSysPs";
};
};

#endif

```

5.2 数据类型的 IDL 定义

```

//File "WCDMANRMMMeasurementSystem.idl"
#ifndef WCDMANRMMMeasurementSystem_idl
#define WCDMANRMMMeasurementSystem_idl

// #pragma prefix "3gppsa5.org"

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

    // typedef unsigned long CountType;
    typedef unsigned long WCDMAMeasurementType1;
    typedef float WCDMAMeasurementType2;

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

```

// The following RANAP causes are defined in the section 9.2.1.4 of 3GPP TS 25.413 v5.5.0.
typedef CauseType 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 CauseType 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 reconfigurationCfnNotElapsed_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 transactionNotSupportedByDestinationNodeB = 26;
const RNSAPCause rlAlreadyActivatedAllocated_RNSAP = 27;
const RNSAPCause numberOfUICodesNotSupported_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 CauseType 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 rIAlreadyActivatedAllocated_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 numberOfDIcodesNotSupported_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 r1TimingAdjustmentNotSupported_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 CauseType 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 CauseType 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 CauseType 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 CauseType 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 CauseType 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 CauseType 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 CauseType 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 CauseType 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 Location Update failure causes are defined in the section 10.5.3.6 of 3GPP TS 24.008 v6.1.0.

```
typedef CauseType LocationUpdateFailureCause;
const LocationUpdateFailureCause imsiUnknownInHLR_Imsi = 2;
const LocationUpdateFailureCause illegalMS_Imsi = 3;
const LocationUpdateFailureCause imsiUnknownInVLR = 4;
const LocationUpdateFailureCause imeiNotAccepted = 5;
const LocationUpdateFailureCause illegalME_Imsi = 6;
const LocationUpdateFailureCause plmnNotAllowed_Imsi = 11;
const LocationUpdateFailureCause locationAreaNotAllowed_Imsi = 12;
const LocationUpdateFailureCause roamingNotAllowedInThisLocationArea_Imsi = 13;
const LocationUpdateFailureCause noSuitableCellsInLocationArea_Imsi = 15;
const LocationUpdateFailureCause networkFailure_Imsi = 17;
const LocationUpdateFailureCause macFailure_Imsi = 20;
const LocationUpdateFailureCause synchFailure_Imsi = 21;
const LocationUpdateFailureCause congestion_Imsi = 22;
const LocationUpdateFailureCause gsmAuthenticationUnacceptable_Imsi = 23;
const LocationUpdateFailureCause serviceOptionNotSupported_Imsi = 32;
const LocationUpdateFailureCause requestedServiceOptionNotSubscribed_Imsi = 33;
const LocationUpdateFailureCause serviceOptionTemporarilyOutOfOrder_Imsi = 34;
const LocationUpdateFailureCause callCannotBeIdentified = 38;
```

```

const LocationUpdateFailureCause failRetryUponEntryIntoANewCell_Imsi = 48;
//value range 48 - 63 is used to retry upon entry into a new cell;
const LocationUpdateFailureCause semanticallyIncorrectMessage_Imsi = 95;
const LocationUpdateFailureCause invalidMandatoryInformation_Imsi = 96;
const LocationUpdateFailureCause messageTypeNon_existentOrNotImplemented_Imsi = 97;
const LocationUpdateFailureCause messageTypeNotCompatibleWithTheProtocolState_Imsi = 98;
const LocationUpdateFailureCause informationElementNon_existentOrNotImplemented_Imsi = 99;
const LocationUpdateFailureCause conditionalIeError_Imsi = 100;
const LocationUpdateFailureCause messageNotCompatibleWithTheProtocolState_Imsi = 101;
const LocationUpdateFailureCause 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 CauseType 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 CauseType 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 requestRelatedToPossiblyDuplicatedPacketsAlready Fulfilled =
252;
const ActPdpContextUtmsFailureCause requestAlreadyFulfilled = 253;
const ActPdpContextUtmsFailureCause sequenceNumbersOfReleasedOrCancelledPacketsIs Incorrect =
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 CauseType 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 conditionalleError_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 CauseType 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 TrafficClassType;
const TrafficClassType conversational = 1;
const TrafficClassType streaming = 2;
const TrafficClassType interactive = 3;
const TrafficClassType background = 4;

typedef unsigned short RlcMode;
const RlcMode transparentMode = 1;
const RlcMode unacknowledgedMode = 2;
const RlcMode acknowledgedMode = 3;
};
#endif

```

6 性能管理接口功能相关的文件

6.1 性能测量数据文件的 Schema 定义<measCollec.xsd>

下面的Schema文件中用到的字段的说明参见附录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 measCollec.xsd -->
<schema targetNamespace="http://latest/nmc-omc/cmNrm.doc#measCollec" elementFormDefault="qualified"
xmlns="http://www.w3.org/2001/XMLSchema" xmlns:mc="http://latest/nmc-omc/cmNrm.doc#measCollec">
  <!-- 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>
            <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>
<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="(mscBasicMeasurement.failImsiAttachsPerCause.lmscBasicMeasurement.failLocationUpdatesIntraMscPerCause.lmscBasicMeasurement.failLocationUpdatesInterMscPerCause.lmscBasicMeasurement.failOrigSmsCsPerCause.lmscBasicMeasurement.failTermSmsCsPerCause.lmscBasicMeasurement.attExternalHosPerCause.lmscBasicMeasurement.attPageReqsPerLa.lmscBasicMeasurement.succPageReqsPerLa.lmscBasicMeasurement.attRepageReqsPerLa.lmobileTrafficFlow.failOrigCallsGsmPerCause.lmobileTrafficFlow.failOrigCallsUmTsPerCause.lmobileTrafficFlow.failInternalCallsPerCause.lmobileTrafficFlow.failTermCallsGsmPerCause.lmobileTrafficFlow.failTermCallsUmTsPerCause.lmobileTrafficFlow.failIncCallsPerCause.lmobileTrafficFlow.failOutCallsPerCause.lmobileTrafficFlow.failTransCallsPerCause.lmobileTrafficFlow.failOrigOutCallsPerCause.lmobileTrafficFlow.failTermIncCallsPerCause.lvlrSubscriberData.nbrCurrentSubsInVlrPerHlr.lhIrlSubscriberData.nbrCurrentSubsWithPowerOnInHlrPerVlr.lsessionManagementMeasurement.failActPdpContextMsPerCause.lsessionManagementMeasurement.failActPdpContextNetworkPerCause.lsubscriberManagementMeasurement.meanAttachedSubsPerRa.lsubscriberManagementMeasurement.maxAttachedSubsPerRa.lmobileManagementMeasurement.failGprsAttachPerCause.lmobileManagementMeasurement.failCombiAttachPerCause.lmobileManagementMeasurement.failGprsAttachWithImsiAttachedPerCause.lmobileManagementMeasurement.failIntraSgsnRaUpdatePerCause.lmobileManagementMeasurement.failCombiIntraSgsnRaUpdatePerCause.lmobileManagementMeasurement.failInterSgsnRaUpdatePerCause.lmobileManagementMeasurement.failCombiInterSgsnRaUpdatePerCause.lapnSessionManagementMeasurement.failActPdpContextUmTsPerCause.lrabAssignmentMeasurement.attRabAssignEstabCsPerType.lrabAssignmentMeasurement.succRabAssignEstabCsPerType.lrabAssignmentMeasurement.failRabAssignEstabCsPerCause.lrabAssignmentMeasurement.attRabAssignEstabPsPerType.lrabAssignmentMeasurement.succRabAssignEstabPsPerType.lrabAssignmentMeasurement.failRabAssignEstabPsPerCause.lrabAssignmentMeasurement.attRabAssignModCsPerType.lrabAssignmentMeasurement.succRabAssignModCsPerType.lrabAssignmentMeasurement.failRabAssignModCsPerCause.lrabAssignmentMeasurement.attRabAssignModPsPerType.lrabAssignmentMeasurement.succRabAssignModPsPerType.lrabAssignmentMeasurement.failRabAssignModPsPerCause.lrabAssignmentMeasurement.attRabAssignRelCsPerType.lrabAssignmentMeasurement.succRabAssignRelCsPerType.lrabAssignmentMeasurement.failRabAssignRelCsPerCause.lrabAssignmentMeasurement.attRabAssignRelPsPerType.lrabAssignmentMeasurement.succRabAssignRelPsPerType.lrabAssignmentMeasurement.failRabAssignRelPsPerCause.lrabReleaseRequestMeasurement.nbrRncRelCsRabPerCause.lrabReleaseRequestMeasurement.nbrRncRelPsRabPerCause.liuConnectionMeasurement.nbrRncRelCsIuConnPerCause.liuConnectionMeasurement.nbrRncRelPsIuConnPerCause.liuConnectionMeasurement.attRelCsIuConnPerCause.liuConnectionMeasurement.attRelPsIuConnPerCause.liuInterfaceMeasurement.nbrResetCsByRncPerCause.liuInterfaceMeasurement.nbrResetPsByRncPerCause.liuInterfaceMeasurement.nbrResetCsByCnPerCause.liuInterfaceMeasurement.nbrResetPsByCnPerCause.liuInterfaceMeasurement.nbrResetResCsByRncPerCause.liuInterfaceMeasurement.nbrResetResPsByRncPerCause.liuInterfaceMeasurement.nbrResetResCsByCnPerCause.liuInterfaceMeasurement.nbrResetResPsByCnPerCause.liuInterfaceMeasurement.nbrErrorIndCsByRncPerCause.liuInterfaceMeasurement.nbrErrorIndPsByRncPerCause.liuInterfaceMeasurement.nbrErrorIndCsByCnPerCause.liuInterfaceMeasurement.nbrErrorIndPsByCnPerCause.lrnCSoftHandoverMeasurement.failRlAddInSofterHoPerCause.lrnCSoftHandoverMeasurement.failRlDelInSofterHoPerCause.lrnCSoftHandoverM

easurement.failRIAddInShoPerCause.lrnSoftHandoverMeasurement.failRIDelInShoPerCause.lrnHardHandove
 rMeasurement.failHhoPerCause.lrnRelocationMeasurement.attRelocOutPrepWithUeNotInvCsPerCause.lrnRel
 ocationMeasurement.failRelocOutPrepWithUeNotInvCsPerCause.lrnRelocationMeasurement.failRelocOutWit
 hUeNotInvCsPerCause.lrnRelocationMeasurement.attRelocOutPrepWithUeInvCsPerCause.lrnRelocationMeas
 urement.failRelocOutPrepWithUeInvCsPerCause.lrnRelocationMeasurement.failRelocOutWithUeInvCsPerCau
 se.lrnRelocationMeasurement.attRelocOutPrepWithUeNotInvPsPerCause.lrnRelocationMeasurement.failRelo
 cOutPrepWithUeNotInvPsPerCause.lrnRelocationMeasurement.failRelocOutWithUeNotInvPsPerCause.lrnRel
 ocationMeasurement.attRelocOutPrepWithUeInvPsPerCause.lrnRelocationMeasurement.failRelocOutPrepWith
 UeInvPsPerCause.lrnRelocationMeasurement.failRelocOutWithUeInvPsPerCause.lrnRelocationMeasurement.
 attRelocInWithUeNotInvCsPerCause.lrnRelocationMeasurement.failRelocInWithUeNotInvCsPerCause.lrnRel
 ocationMeasurement.attRelocInWithUeInvCsPerCause.lrnRelocationMeasurement.failRelocInWithUeInvCsPer
 Cause.lrnRelocationMeasurement.attRelocInWithUeNotInvPsPerCause.lrnRelocationMeasurement.failRelocIn
 WithUeNotInvPsPerCause.lrnRelocationMeasurement.attRelocInWithUeInvPsPerCause.lrnRelocationMeasur
 ement.failRelocInWithUeInvPsPerCause.lrnInterSystemHandoverMeasurement.attRelocOutPrepInterSysCsPer
 Cause.lrnInterSystemHandoverMeasurement.failRelocOutPrepInterSysCsPerCause.lrnInterSystemHandoverM
 easurement.failRelocOutInterSysCsPerCause.lrnInterSystemHandoverMeasurement.attRelocInInterSysCsPerCa
 use.lrnInterSystemHandoverMeasurement.failRelocInInterSysCsPerCause.lrnInterSystemHandoverMeasurem
 ent.failRelocOutInterSysPsPerCause.liuInterfaceThroughputMeasurement.iuUIDataThroughputCsPerType.liuInt
 erfaceThroughputMeasurement.iuDIDataThroughputCsPerType.liuInterfaceThroughputMeasurement.iuUIDataT
 hroughputPsPerType.liuInterfaceThroughputMeasurement.iuDIDataThroughputPsPerType.lrlcConnectionMeasu
 rement.nbrRlcBlockSentPerMode.lrlcConnectionMeasurement.nbrRlcBlockRecvedPerMode.lcellRrcConnection
 Measurement.attRrcConnSetupPerCause.lcellRrcConnectionMeasurement.succRrcConnSetupPerCause.lcellRrc
 ConnectionMeasurement.failRrcConnSetupPerCause.lcellRrcConnectionMeasurement.failRrcConnReestabPerC
 ause.lcellSoftHandoverMeasurement.failRIAddInShoPerCause.lcellSoftHandoverMeasurement.failRIDelInShoP
 erCause.lhardHandoverIntraCellMeasurement.failHhoOutIntraCellPerCause.liubRlManagementMeasurement.fai
 lRlSetupIubPerCause.liubRlManagementMeasurement.failRIAddIubPerCause.liurRlManagementMeasurement.f
 ailRlSetupIurPerCause.liurRlManagementMeasurement.failRIAddIurPerCause.lhardHandoverInterCellIntraNode
 BMeasurement.failHhoOutInterCellIntraNodeBPerCause.lhardHandoverInterNodeBIntraRncMeasurement.failH
 hoOutInterNodeBIntraRncPerCause.lhardHandoverInterRncViaIurMeasurement.failHhoOutInterRncViaIurPerC
 ause.lhardHandoverInterRncMeasurement.failHhoOutInterRncCnPerCause.lhardHandoverInterSystemMeasure
 ment.attRelocOutPrepInterSysCsPerCause.lhardHandoverInterSystemMeasurement.failRelocOutPrepInterSysCs
 PerCause.lhardHandoverInterSystemMeasurement.failRelocOutInterSysCsPerCause.lhardHandoverInterSystem
 Measurement.attRelocInInterSysCsPerCause.lhardHandoverInterSystemMeasurement.failRelocInInterSysCsPer
 Cause.lhardHandoverInterSystemMeasurement.failRelocOutInterSysPsPerCause.)\d{1,5}"/>

</restriction>

</simpleType>

<simpleType name="measNameWithoutSubCounter">

```

<restriction base="string">
  <enumeration value="mscBasicMeasurement.attGetRoutingInfo"/>
  <enumeration value="mscBasicMeasurement.succGetRoutingInfo"/>
  <enumeration value="mscBasicMeasurement.attImsiAttachs"/>
  <enumeration value="mscBasicMeasurement.succImsiAttachs"/>
  <enumeration value="mscBasicMeasurement.nbrImsiDetachs"/>
  <enumeration value="mscBasicMeasurement.attLocationUpdatesIntraMsc"/>
  <enumeration value="mscBasicMeasurement.succLocationUpdatesIntraMsc"/>
  <enumeration value="mscBasicMeasurement.attLocationUpdatesInterMsc"/>
  <enumeration value="mscBasicMeasurement.succLocationUpdatesInterMsc"/>
  <enumeration value="mscBasicMeasurement.attOrigSmsCs"/>
  <enumeration value="mscBasicMeasurement.succOrigSmsCs"/>
  <enumeration value="mscBasicMeasurement.attTermSsmsCs"/>
  <enumeration value="mscBasicMeasurement.succTermSmsCs"/>
  <enumeration value="mscBasicMeasurement.attIncHosInterMsc"/>
  <enumeration value="mscBasicMeasurement.succIncHosInterMsc"/>
  <enumeration value="mscBasicMeasurement.attOutHosInterMsc"/>
  <enumeration value="mscBasicMeasurement.succOutHosInterMsc"/>
  <enumeration value="mscBasicMeasurement.attSubsequentHosToMsca"/>
  <enumeration value="mscBasicMeasurement.succSubsequentHosToMsca"/>
  <enumeration value="mscBasicMeasurement.attSubsequentHosToMscs"/>
  <enumeration value="mscBasicMeasurement.succSubsequentHosToMscs"/>
  <enumeration value="mscBasicMeasurement.attExternalHos"/>
  <enumeration value="mscBasicMeasurement.failExternalHosWithReconn"/>
  <enumeration value="mscBasicMeasurement.failExternalHosWithLossOfConn"/>
  <enumeration value="mscQos.meanDurOfCallSetup"/>
  <enumeration value="mscQos.meanDurOfCallAssignGsm"/>
  <enumeration value="mscQos.meanDurOfCallRabAssignUmts"/>
  <enumeration value="mscQos.meanDurOfLuService"/>
  <enumeration value="mscQos.meanCallDur"/>
  <enumeration value="mscQos.meanDurOfTrunkSeizure"/>
  <enumeration value="mobileTrafficFlow.attOrigCallsGsm"/>
  <enumeration value="mobileTrafficFlow.succOrigCallsGsm"/>
  <enumeration value="mobileTrafficFlow.ansOrigCallsGsm"/>
  <enumeration value="mobileTrafficFlow.attOrigCallTrafficGsm"/>
  <enumeration value="mobileTrafficFlow.succOrigCallTrafficGsm"/>
  <enumeration value="mobileTrafficFlow.ansOrigCallTrafficGsm"/>

```

<enumeration value="mobileTrafficFlow.attOrigCallsUmts"/>
<enumeration value="mobileTrafficFlow.succOrigCallsUmts"/>
<enumeration value="mobileTrafficFlow.ansOrigCallsUmts"/>
<enumeration value="mobileTrafficFlow.attOrigCallTrafficUmts"/>
<enumeration value="mobileTrafficFlow.succOrigCallTrafficUmts"/>
<enumeration value="mobileTrafficFlow.ansOrigCallTrafficUmts"/>
<enumeration value="mobileTrafficFlow.attInternalCalls"/>
<enumeration value="mobileTrafficFlow.succInternalCalls"/>
<enumeration value="mobileTrafficFlow.ansInternalCalls"/>
<enumeration value="mobileTrafficFlow.attInternalCallTraffic"/>
<enumeration value="mobileTrafficFlow.succInternalCallTraffic"/>
<enumeration value="mobileTrafficFlow.ansInternalCallTraffic"/>
<enumeration value="mobileTrafficFlow.attTermCallsGsm"/>
<enumeration value="mobileTrafficFlow.succTermCallsGsm"/>
<enumeration value="mobileTrafficFlow.ansTermCallsGsm"/>
<enumeration value="mobileTrafficFlow.attTermCallTrafficGsm"/>
<enumeration value="mobileTrafficFlow.succTermCallTrafficGsm"/>
<enumeration value="mobileTrafficFlow.ansTermCallTrafficGsm"/>
<enumeration value="mobileTrafficFlow.attTermCallsUmts"/>
<enumeration value="mobileTrafficFlow.succTermCallsUmts"/>
<enumeration value="mobileTrafficFlow.ansTermCallsUmts"/>
<enumeration value="mobileTrafficFlow.attTermCallTrafficUmts"/>
<enumeration value="mobileTrafficFlow.succTermCallTrafficUmts"/>
<enumeration value="mobileTrafficFlow.ansTermCallTrafficUmts"/>
<enumeration value="mobileTrafficFlow.attIncCalls"/>
<enumeration value="mobileTrafficFlow.succIncCalls"/>
<enumeration value="mobileTrafficFlow.ansIncCalls"/>
<enumeration value="mobileTrafficFlow.attIncCallTraffic"/>
<enumeration value="mobileTrafficFlow.succIncCallTraffic"/>
<enumeration value="mobileTrafficFlow.ansIncCallTraffic"/>
<enumeration value="mobileTrafficFlow.attOutCalls"/>
<enumeration value="mobileTrafficFlow.attOutCalls"/>
<enumeration value="mobileTrafficFlow.ansOutCalls"/>
<enumeration value="mobileTrafficFlow.attOutCallTraffic"/>
<enumeration value="mobileTrafficFlow.succOutCallTraffic"/>
<enumeration value="mobileTrafficFlow.ansOutCallTraffic"/>
<enumeration value="mobileTrafficFlow.attTransCalls"/>


```
<enumeration value="mobileTrafficFlow.succTransCalls"/>
<enumeration value="mobileTrafficFlow.ansTransCalls"/>
<enumeration value="mobileTrafficFlow.attTransCallTraffic"/>
<enumeration value="mobileTrafficFlow.succTransCallTraffic"/>
<enumeration value="mobileTrafficFlow.ansTransCallTraffic"/>
<enumeration value="mobileTrafficFlow.attOrigOutCalls"/>
<enumeration value="mobileTrafficFlow.succOrigOutCalls"/>
<enumeration value="mobileTrafficFlow.ansOrigOutCalls"/>
<enumeration value="mobileTrafficFlow.attOrigOutCallTraffic"/>
<enumeration value="mobileTrafficFlow.succOrigOutCallTraffic"/>
<enumeration value="mobileTrafficFlow.ansOrigOutCallTraffic"/>
<enumeration value="mobileTrafficFlow.attTermIncCalls"/>
<enumeration value="mobileTrafficFlow.succTermIncCalls"/>
<enumeration value="mobileTrafficFlow.ansTermIncCalls"/>
<enumeration value="mobileTrafficFlow.attTermIncCallTraffic"/>
<enumeration value="mobileTrafficFlow.succTermIncCallTraffic"/>
<enumeration value="mobileTrafficFlow.ansTermIncCallTraffic"/>
<enumeration value="mobileTrafficFlow.nbrCallsBlockedByLoadShedding"/>
<enumeration value="mobileTrafficFlow.nbrCallsBlockedByInternalCongestion"/>
<enumeration value="mobileTrafficFlow.nbrCallsBlockedByOutCircuitBusy"/>
<enumeration value="circuitEndpointSubGroup.outBids"/>
<enumeration value="circuitEndpointSubGroup.succOutSeizures"/>
<enumeration value="circuitEndpointSubGroup.succOutCalls"/>
<enumeration value="circuitEndpointSubGroup.ansOutCalls"/>
<enumeration value="circuitEndpointSubGroup.failOutCallsByOverflow"/>
<enumeration value="circuitEndpointSubGroup.failOutCallsByUserBusy"/>
<enumeration value="circuitEndpointSubGroup.failOutCallsByNoAns"/>
<enumeration value="circuitEndpointSubGroup.failOutCallsByUnallNum"/>
<enumeration value="circuitEndpointSubGroup.failOutCallsByCongestion"/>
<enumeration value="circuitEndpointSubGroup.succOutSeizureTraffic"/>
<enumeration value="circuitEndpointSubGroup.ansOutCallTraffic"/>
<enumeration value="circuitEndpointSubGroup.succIncSeizures"/>
<enumeration value="circuitEndpointSubGroup.succIncCalls"/>
<enumeration value="circuitEndpointSubGroup.ansIncSeizures"/>
<enumeration value="circuitEndpointSubGroup.failIncCallsByUserBusy"/>
<enumeration value="circuitEndpointSubGroup.failIncCallsByNoAns"/>
<enumeration value="circuitEndpointSubGroup.failIncCallsByUnallNum"/>
```

```

<enumeration value="circuitEndpointSubGroup.failIncCallsByCongestion"/>
<enumeration value="circuitEndpointSubGroup.succIncSeizureTraffic"/>
<enumeration value="circuitEndpointSubGroup.ansIncSeizureTraffic"/>
<enumeration value="circuitEndpointSubGroup.nbrAvailTrunks"/>
<enumeration value="mtp3SignallingLinkTP.durSigLinkOutOfService"/>
<enumeration value="mtp3SignallingLinkTP.nbrSigLinkOutOfService"/>
<enumeration value="mtp3SignallingLinkTP.nbrSentMsus"/>
<enumeration value="mtp3SignallingLinkTP.nbrSentSifsAndSios"/>
<enumeration value="mtp3SignallingLinkTP.nbrRecvedMsus"/>
<enumeration value="mtp3SignallingLinkTP.nbrRecvedSifsAndSios"/>
<enumeration value="mtp3SignallingLinkSetTP.nbrSigLinkSetOutOfService"/>
<enumeration value="mtp3SignallingLinkSetTP.durSigLinkSetOutOfService"/>
<enumeration value="mtp3SignallingLinkSetTP.nbrAvailSiglinks"/>
<enumeration value="mtp3bSignallingLinkTP.durSigLinkOutOfService"/>
<enumeration value="mtp3bSignallingLinkTP.nbrSigLinkOutOfService"/>
<enumeration value="mtp3bSignallingLinkTP.nbrSentMsus"/>
<enumeration value="mtp3bSignallingLinkTP.nbrSentSifsAndSios"/>
<enumeration value="mtp3bSignallingLinkTP.nbrRecvedMsus"/>
<enumeration value="mtp3bSignallingLinkTP.nbrRecvedSifsAndSios"/>
<enumeration value="mtp3bSignallingLinkSetTP.nbrSigLinkSetOutOfService"/>
<enumeration value="mtp3bSignallingLinkSetTP.durSigLinkSetOutOfService"/>
<enumeration value="mtp3bSignallingLinkSetTP.nbrAvailSiglinks"/>
<enumeration value="observedDestination.bids"/>
<enumeration value="observedDestination.nbrNoAvailCircuits"/>
<enumeration value="observedDestination.succCalls"/>
<enumeration value="observedDestination.ansCalls"/>
<enumeration value="observedDestination.succCallTraffic"/>
<enumeration value="observedDestination.ansCallTraffic"/>
<enumeration value="vlrBasicMeasurement.attIdentReqsToPVlr"/>
<enumeration value="vlrBasicMeasurement.succIdentReqsToPVlr"/>
<enumeration value="vlrBasicMeasurement.attLusIntraVlr"/>
<enumeration value="vlrBasicMeasurement.succLusIntraVlr"/>
<enumeration value="vlrBasicMeasurement.attLusInterVlr"/>
<enumeration value="vlrBasicMeasurement.succLusInterVlr"/>
<enumeration value="vlrBasicMeasurement.attReqsForAuthSetsSentToHlr"/>
<enumeration value="vlrBasicMeasurement.succRecvedAuthSetsFromHlr"/>
<enumeration value="vlrBasicMeasurement.succReqAuthSetWithQuintupletsFromHlr"/>

```

<enumeration value="vlrBasicMeasurement.succReqAuthSetWithTripletsFromHlr"/>
<enumeration value="vlrBasicMeasurement.succInsertSubsData"/>
<enumeration value="vlrBasicMeasurement.succDelSubsData"/>
<enumeration value="vlrBasicMeasurement.attProvideRoamingNumber"/>
<enumeration value="vlrBasicMeasurement.succProvideRoamingNumber"/>
<enumeration value="vlrSubscriberData.nbrCurrentSubsWithPowerOnInVlr"/>
<enumeration value="vlrSubscriberData.nbrRoamingSubs"/>
<enumeration value="vlrSubscriberData.nbrRoamingSubsInternational"/>
<enumeration value="hlrBasicMeasurement.attGetRoutingInfo"/>
<enumeration value="hlrBasicMeasurement.succGetRoutingInfo"/>
<enumeration value="hlrBasicMeasurement.attProvideRoamingNumber"/>
<enumeration value="hlrBasicMeasurement.succProvideRoamingNumber"/>
<enumeration value="hlrBasicMeasurement.attLocationUpdates"/>
<enumeration value="hlrBasicMeasurement.succLocationUpdates"/>
<enumeration value="hlrBasicMeasurement.attCancelLocation"/>
<enumeration value="hlrBasicMeasurement.succCancelLocation"/>
<enumeration value="hlrBasicMeasurement.attInsertSubsData"/>
<enumeration value="hlrBasicMeasurement.succInsertSubsData"/>
<enumeration value="hlrBasicMeasurement.attDeleteSubsData"/>
<enumeration value="hlrBasicMeasurement.succDeleteSubsData"/>
<enumeration value="hlrBasicMeasurement.attSendAuthInfo"/>
<enumeration value="hlrBasicMeasurement.succSendAuthInfo"/>
<enumeration value="hlrBasicMeasurement.nbrReset"/>
<enumeration value="hlrBasicMeasurement.attRestoreData"/>
<enumeration value="hlrBasicMeasurement.succRestoreData"/>
<enumeration value="hlrSubscriberData.nbrCurrentSubsInHlr"/>
<enumeration value="hlrSubscriberData.nbrCurrentMsisdnInHlr"/>
<enumeration value="hlrSmServiceMeasurement.attSendRoutingInfoForSm"/>
<enumeration value="hlrSmServiceMeasurement.succSendRoutingInfoForSm"/>
<enumeration value="hlrSmServiceMeasurement.nbrAlertServiceCentre"/>
<enumeration value="hlrSmServiceMeasurement.nbrInformServiceCenter"/>
<enumeration value="hlrSmServiceMeasurement.nbrReadyForSm"/>
<enumeration value="hlrSupplementServiceMeasurement.attRegisterSs"/>
<enumeration value="hlrSupplementServiceMeasurement.succRegisterSs"/>
<enumeration value="hlrSupplementServiceMeasurement.attEraseSs"/>
<enumeration value="hlrSupplementServiceMeasurement.succEraseSs"/>
<enumeration value="hlrSupplementServiceMeasurement.attActSs"/>

```

<enumeration value="hlrSupplementServiceMeasurement.succActSs"/>
<enumeration value="hlrSupplementServiceMeasurement.attDeactSs"/>
<enumeration value="hlrSupplementServiceMeasurement.succDeactSs"/>
<enumeration value="hlrInServiceMeasurement.attAnyTimeInterrogation"/>
<enumeration value="hlrInServiceMeasurement.succAnyTimeInterrogation"/>
<enumeration value="hlrInServiceMeasurement.attAnyTimeSubsInterrogation"/>
<enumeration value="hlrInServiceMeasurement.succAnyTimeSubsInterrogation"/>
<enumeration value="hlrInServiceMeasurement.attAnyTimeModification"/>
<enumeration value="hlrInServiceMeasurement.succAnyTimeModification"/>
<enumeration value="hlrInServiceMeasurement.nbrNoteSubsDataModified"/>
<enumeration value="hlrPacketServiceMeasurement.attSendRoutingInfoForGprs"/>
<enumeration value="hlrPacketServiceMeasurement.succSendRoutingInfoForGprs"/>
<enumeration value="hlrPacketServiceMeasurement.nbrFailReport"/>
<enumeration value="hlrPacketServiceMeasurement.nbrNoteMsPresentForGprs"/>
<enumeration value="hlrPacketServiceMeasurement.attUpdateGprsLocation"/>
<enumeration value="hlrPacketServiceMeasurement.succUpdateGprsLocation"/>
<enumeration value="hlrLocationServiceMeasurement.attSendRoutingInfoForLcs"/>
<enumeration value="hlrLocationServiceMeasurement.succSendRoutingInfoForLcs"/>
<enumeration value="eirBasicMeasurement.nbrCurrentWhiteSubsInEir"/>
<enumeration value="eirBasicMeasurement.nbrCurrentBlackSubsInEir"/>
<enumeration value="eirBasicMeasurement.nbrCurrentGreySubsInEir"/>
<enumeration value="sessionManagementMeasurement.attActPdpContextMs"/>
<enumeration value="sessionManagementMeasurement.succActPdpContextMs"/>
<enumeration value="sessionManagementMeasurement.attActPdpContextNetwork"/>
<enumeration value="sessionManagementMeasurement.succActPdpContextNetwork"/>
<enumeration value="sessionManagementMeasurement.attActPdpContextDynMs"/>
<enumeration value="sessionManagementMeasurement.succActPdpContextDynMs"/>
<enumeration value="sessionManagementMeasurement.meanSubsWithActPdpContext"/>
<enumeration value="sessionManagementMeasurement.maxSubsWithActPdpContext"/>
<enumeration value="sessionManagementMeasurement.meanActPDPContexts"/>
<enumeration value="sessionManagementMeasurement.maxActPdpContexts"/>
<enumeration value="sessionManagementMeasurement.attDeactPdpContextSgsn"/>
<enumeration value="sessionManagementMeasurement.succDeactPdpContextSgsn"/>
<enumeration value="sessionManagementMeasurement.attDeactPdpContextMs"/>
<enumeration value="sessionManagementMeasurement.succDeactPdpContextMs"/>
<enumeration value="sessionManagementMeasurement.attDeactPdpContextGgsn"/>
<enumeration value="sessionManagementMeasurement.succDeactPdpContextGgsn"/>

```

<enumeration value="sessionManagementMeasurement.attActSecondPdpContext"/>
<enumeration value="sessionManagementMeasurement.succActSecondPdpContext"/>
<enumeration value="sessionManagementMeasurement.attModPdpContextMs"/>
<enumeration value="sessionManagementMeasurement.succModPdpContextMs"/>
<enumeration value="sessionManagementMeasurement.attModPdpContextSgsn"/>
<enumeration value="sessionManagementMeasurement.succModPdpContextSgsn"/>
<enumeration value="sessionManagementMeasurement.attUpdPdpContextGgsn"/>
<enumeration value="sessionManagementMeasurement.succUpdPdpContextGgsn"/>
<enumeration value="sessionManagementMeasurement.attUpdPdpContextSgsn"/>
<enumeration value="sessionManagementMeasurement.succUpdPdpContextSgsn"/>
<enumeration value="subscriberManagementMeasurement.meanStandbySubs"/>
<enumeration value="subscriberManagementMeasurement.maxStandbySubs"/>
<enumeration value="subscriberManagementMeasurement.meanReadySubs"/>
<enumeration value="subscriberManagementMeasurement.maxReadySubs"/>
<enumeration value="subscriberManagementMeasurement.meanPmmIdleSubs"/>
<enumeration value="subscriberManagementMeasurement.maxPmmIdleSubs"/>
<enumeration value="subscriberManagementMeasurement.meanPmmConnectedSubs"/>
<enumeration value="subscriberManagementMeasurement.maxPmmConnectedSubs"/>
<enumeration value="mobileManagementMeasurement.attGprsAttach"/>
<enumeration value="mobileManagementMeasurement.succGprsAttach"/>
<enumeration value="mobileManagementMeasurement.attCombiAttach"/>
<enumeration value="mobileManagementMeasurement.succCombiAttach"/>
<enumeration value="mobileManagementMeasurement.attGprsAttachWithImsiAttached"/>
<enumeration value="mobileManagementMeasurement.succGprsAttachWithImsiAttached"/>
<enumeration value="mobileManagementMeasurement.attGprsDetachMs"/>
<enumeration value="mobileManagementMeasurement.attCombiDetachMs"/>
<enumeration value="mobileManagementMeasurement.attImsiDetachMs"/>
<enumeration value="mobileManagementMeasurement.attGprsDetachSgsn"/>
<enumeration value="mobileManagementMeasurement.succGprsDetachSgsn"/>
<enumeration value="mobileManagementMeasurement.attGprsDetachHlr"/>
<enumeration value="mobileManagementMeasurement.attIntraSgsnRaUpdate"/>
<enumeration value="mobileManagementMeasurement.succIntraSgsnRaUpdate"/>
<enumeration value="mobileManagementMeasurement.attCombiIntraSgsnRaUpdate"/>
<enumeration value="mobileManagementMeasurement.succCombiIntraSgsnRaUpdate"/>
<enumeration value="mobileManagementMeasurement.attInterSgsnRaUpdate"/>
<enumeration value="mobileManagementMeasurement.succInterSgsnRaUpdate"/>
<enumeration value="mobileManagementMeasurement.attCombiInterSgsnRaUpdate"/>

```
<enumeration value="mobileManagementMeasurement.succCombiInterSgsnRaUpdate"/>
<enumeration value="mobileManagementMeasurement.nbrPsPagingGsm"/>
<enumeration value="mobileManagementMeasurement.nbrPsPagingNoRspGsm"/>
<enumeration value="mobileManagementMeasurement.nbrPsPagingUmts"/>
<enumeration value="mobileManagementMeasurement.nbrPsPagingNoRspUmts"/>
<enumeration value="sgsnRelocationMeasurement.attInterSgsnReloc"/>
<enumeration value="sgsnRelocationMeasurement.succInterSgsnReloc"/>
<enumeration value="sgsnRelocationMeasurement.attInterSgsnCombiReloc"/>
<enumeration value="sgsnRelocationMeasurement.succInterSgsnCombiReloc"/>
<enumeration value="sgsnRelocationMeasurement.attIntraSgsnReloc"/>
<enumeration value="sgsnRelocationMeasurement.succIntraSgsnReloc"/>
<enumeration value="sgsnRelocationMeasurement.attIntraSgsnCombiReloc"/>
<enumeration value="sgsnRelocationMeasurement.succIntraSgsnCombiReloc"/>
<enumeration value="sgsnInterSystemHandoverMeasurement.attIntraSgsnHoGsmToUmts"/>
<enumeration value="sgsnInterSystemHandoverMeasurement.succIntraSgsnHoGsmToUmts"/>
<enumeration value="sgsnInterSystemHandoverMeasurement.attIntraSgsnHoUmtsToGsm"/>
<enumeration value="sgsnInterSystemHandoverMeasurement.succIntraSgsnHoUmtsToGsm"/>
<enumeration value="sgsnInterSystemHandoverMeasurement.attInterSgsnHoGsmToUmts"/>
<enumeration value="sgsnInterSystemHandoverMeasurement.succInterSgsnHoGsmToUmts"/>
<enumeration value="sgsnInterSystemHandoverMeasurement.attInterSgsnHoUmtsToGsm"/>
<enumeration value="sgsnInterSystemHandoverMeasurement.succInterSgsnHoUmtsToGsm"/>
<enumeration value="mapServiceMeasurement.attReqAuthSetHlr"/>
<enumeration value="mapServiceMeasurement.succReqAuthSetWithQuintupletsHlr"/>
<enumeration value="mapServiceMeasurement.succReqAuthSetWithTripletsHlr"/>
<enumeration value="mapServiceMeasurement.succReqAuthSetWithEmptyRspHlr"/>
<enumeration value="mapServiceMeasurement.attUpdateGprsLocationHlr"/>
<enumeration value="mapServiceMeasurement.succUpdateGprsLocationHlr"/>
<enumeration value="mapServiceMeasurement.attInsertSubsDataHlr"/>
<enumeration value="mapServiceMeasurement.attDeleteSubsDataHlr"/>
<enumeration value="securityManagementMeasurement.attPtmsiRealloc"/>
<enumeration value="securityManagementMeasurement.succPtmsiRealloc"/>
<enumeration value="securityManagementMeasurement.attAuthReq"/>
<enumeration value="securityManagementMeasurement.succAuthReq"/>
<enumeration value="securityManagementMeasurement.attIdentReq"/>
<enumeration value="securityManagementMeasurement.succIdentReq"/>
<enumeration value="securityManagementMeasurement.attSecMode"/>
<enumeration value="securityManagementMeasurement.succSecMode"/>
```

```

<enumeration value="gtpInGnGpMeasurement.nbrIncGtpCSigPkts"/>
<enumeration value="gtpInGnGpMeasurement.nbrOutGtpCSigPkts"/>
<enumeration value="gtpInGnGpMeasurement.nbrIncGtpCSigOcts"/>
<enumeration value="gtpInGnGpMeasurement.nbrOutGtpCSigOcts"/>
<enumeration value="gtpInGnGpMeasurement.nbrOutGtpCSigPktsOverflow"/>
<enumeration value="gtpInGnGpMeasurement.nbrIncGtpCSigPktsError"/>
<enumeration value="gtpInGnGpMeasurement.nbrIncGtpUDDataPkts"/>
<enumeration value="gtpInGnGpMeasurement.nbrOutGtpUDDataPkts"/>
<enumeration value="gtpInGnGpMeasurement.nbrIncGtpUDDataOcts"/>
<enumeration value="gtpInGnGpMeasurement.nbrOutGtpUDDataOcts"/>
<enumeration value="gtpInGnGpMeasurement.nbrOutGtpUDDataPktsOverflow"/>
<enumeration value="gtpInGnGpMeasurement.nbrIncGtpUDDataPktsError"/>
<enumeration value="shortMessageServiceMeasurement.attOrigSmsPs"/>
<enumeration value="shortMessageServiceMeasurement.succOrigSmsPs"/>
<enumeration value="shortMessageServiceMeasurement.attTermSmsPs"/>
<enumeration value="shortMessageServiceMeasurement.succTermSmsPs"/>
<enumeration value="shortMessageServiceMeasurement.attSmsMsPresentPs"/>
<enumeration value="shortMessageServiceMeasurement.succSmsMsPresentPs"/>
<enumeration value="shortMessageServiceMeasurement.attSmsMemoryAvailPs"/>
<enumeration value="shortMessageServiceMeasurement.succSmsMemoryAvailPs"/>
<enumeration value="ggsnThroughputMeasurement.nbrIncDataPktsGnGp"/>
<enumeration value="ggsnThroughputMeasurement.nbrOutDataPktsGnGp"/>
<enumeration value="ggsnThroughputMeasurement.nbrIncDataOctsGnGp"/>
<enumeration value="ggsnThroughputMeasurement.nbrOutDataOctsGnGp"/>
<enumeration value="ggsnThroughputMeasurement.nbrIncSigPktsGnGp"/>
<enumeration value="ggsnThroughputMeasurement.nbrOutSigPktsGnGp"/>
<enumeration value="ggsnThroughputMeasurement.nbrIncSigOctsGnGp"/>
<enumeration value="ggsnThroughputMeasurement.nbrOutSigOctsGnGp"/>
<enumeration value="ggsnThroughputMeasurement.nbrOutDataPktsGi"/>
<enumeration value="ggsnThroughputMeasurement.nbrIncDataPktsGi"/>
<enumeration value="ggsnThroughputMeasurement.nbrOutDataOctsGi"/>
<enumeration value="ggsnThroughputMeasurement.nbrIncDataOctsGi"/>
<enumeration value="apnSessionManagementMeasurement.attActPdpContext"/>
<enumeration value="apnSessionManagementMeasurement.succActPdpContext"/>
<enumeration value="apnSessionManagementMeasurement.attDynActPdpContext"/>
<enumeration value="apnSessionManagementMeasurement.succDynActPdpContext"/>
<enumeration value="apnSessionManagementMeasurement.succActPdpContextQos"/>

```

```
<enumeration value="apnSessionManagementMeasurement.attDeactPdpContextMs"/>
<enumeration value="apnSessionManagementMeasurement.succDeactPdpContextMs"/>
<enumeration value="apnSessionManagementMeasurement.attDeactPdpContextGgsn"/>
<enumeration value="apnSessionManagementMeasurement.succDeactPdpContextGgsn"/>
<enumeration value="apnSessionManagementMeasurement.nbrActPdpContexts"/>
<enumeration value="apnSessionManagementMeasurement.meanActPdpContexts"/>
<enumeration value="apnSessionManagementMeasurement.maxActPdpContexts"/>
<enumeration value="apnThroughputMeasurement.nbrIncDataPktsGnGp"/>
<enumeration value="apnThroughputMeasurement.nbrOutDataPktsGnGp"/>
<enumeration value="apnThroughputMeasurement.nbrIncDataOctsGnGp"/>
<enumeration value="apnThroughputMeasurement.nbrOutDataOctsGnGp"/>
<enumeration value="apnThroughputMeasurement.nbrIncSigPktsGnGp"/>
<enumeration value="apnThroughputMeasurement.nbrOutSigPktsGnGp"/>
<enumeration value="apnThroughputMeasurement.nbrIncSigOctsGnGp"/>
<enumeration value="apnThroughputMeasurement.nbrOutSigOctsGnGp"/>
<enumeration value="apnThroughputMeasurement.nbrOutDataPktsGi"/>
<enumeration value="apnThroughputMeasurement.nbrIncDataPktsGi"/>
<enumeration value="apnThroughputMeasurement.nbrOutDataOctsGi"/>
<enumeration value="apnThroughputMeasurement.nbrIncDataOctsGi"/>
<enumeration value="iuConnectionMeasurement.attRncEstabCsIuConn"/>
<enumeration value="iuConnectionMeasurement.attRncEstabPsIuConn"/>
<enumeration value="iuInterfaceMeasurement.nbrOverloadControlCsByRnc"/>
<enumeration value="iuInterfaceMeasurement.nbrOverloadControlPsByRnc"/>
<enumeration value="iuInterfaceMeasurement.nbrOverloadControlCsByCn"/>
<enumeration value="iuInterfaceMeasurement.nbrOverloadControlPsByCn"/>
<enumeration value="rncSoftHandoverMeasurement.attRlAddInSofterHo"/>
<enumeration value="rncSoftHandoverMeasurement.attRlDelInSofterHo"/>
<enumeration value="rncSoftHandoverMeasurement.attRlAddInSho"/>
<enumeration value="rncSoftHandoverMeasurement.attRlDelInSho"/>
<enumeration value="rncHardHandoverMeasurement.attHho"/>
<enumeration value="rncRelocationMeasurement.attRelocOutWithUeNotInvCs"/>
<enumeration value="rncRelocationMeasurement.attRelocOutWithUeInvCs"/>
<enumeration value="rncRelocationMeasurement.attRelocOutWithUeNotInvPs"/>
<enumeration value="rncRelocationMeasurement.attRelocOutWithUeInvPs"/>
<enumeration value="rncInterSystemHandoverMeasurement.attRelocOutInterSysCs"/>
<enumeration value="rncInterSystemHandoverMeasurement.attRelocOutInterSysPs"/>
<enumeration value="rncInterSystemHandoverMeasurement.attRelocInInterSysPs"/>
```



```

<enumeration value="rncInterSystemHandoverMeasurement.succRelocInInterSysPs"/>
<enumeration value="iuInterfaceThroughputMeasurement.iuUISigThroughputCs"/>
<enumeration value="iuInterfaceThroughputMeasurement.iuDISigThroughputCs"/>
<enumeration value="iuInterfaceThroughputMeasurement.iuUISigThroughputPs"/>
<enumeration value="iuInterfaceThroughputMeasurement.iuDISigThroughputPs"/>
<enumeration value="iurInterfaceThroughputMeasurement.iurUISigThroughput"/>
<enumeration value="iurInterfaceThroughputMeasurement.iurDISigThroughput"/>
<enumeration value="iurInterfaceThroughputMeasurement.iurUIDataThroughput"/>
<enumeration value="iurInterfaceThroughputMeasurement.iurDIDataThroughput"/>
<enumeration value="rlcConnectionMeasurement.nbrDiscardedRlcBlocksByRnc"/>
<enumeration value="rlcConnectionMeasurement.nbrRetransmittedRlcBlocksToUe"/>
<enumeration value="cellRrcConnectionMeasurement.attRrcConnReestab"/>
<enumeration value="cellSoftHandoverMeasurement.attRIAddInSho"/>
<enumeration value="cellSoftHandoverMeasurement.attRIDelInSho"/>
<enumeration value="hardHandoverIntraCellMeasurement.attHhoOutIntraCell"/>
<enumeration value="iubRIManagementMeasurement.attRISetupIub"/>
<enumeration value="iubRIManagementMeasurement.attRIAddIub"/>
<enumeration value="iubRIManagementMeasurement.attRIDelIub"/>
<enumeration value="iubRIManagementMeasurement.succRIDelIub"/>
<enumeration value="iurRIManagementMeasurement.attRISetupIur"/>
<enumeration value="iurRIManagementMeasurement.attRIAddIur"/>
<enumeration value="iurRIManagementMeasurement.attRIDelIur"/>
<enumeration value="iurRIManagementMeasurement.succRIDelIur"/>
<enumeration value="cellTrafficMeasurement.cellCchTraffic"/>
<enumeration value="cellTrafficMeasurement.cellCtchTraffic"/>
<enumeration value="cellTrafficMeasurement.cellDcchTraffic"/>
<enumeration value="cellTrafficMeasurement.cellDtchTraffic"/>
<enumeration value="cellPagingMeasurement.attPagingType1FromUtran"/>
<enumeration value="cellPagingMeasurement.succPagingType1FromUtran"/>
<enumeration value="cellPagingMeasurement.attPagingType2FromUtran"/>
<enumeration
value="hardHandoverInterCellIntraNodeBMeasurement.attHhoOutInterCellIntraNodeB"/>
<enumeration
value="hardHandoverInterNodeBIntraRncMeasurement.attHhoOutInterNodeBIntraRnc"/>
<enumeration value="hardHandoverInterRncViaIurMeasurement.attHhoOutInterRncViaIur"/>
<enumeration value="hardHandoverInterRncMeasurement.attHhoOutInterRncCn"/>
<enumeration value="hardHandoverInterSystemMeasurement.attRelocOutInterSysCs"/>

```

```

    <enumeration value="hardHandoverInterSystemMeasurement.attRelocOutInterSysPs"/>
    <enumeration value="hardHandoverInterSystemMeasurement.attRelocInInterSysPs"/>
    <enumeration value="hardHandoverInterSystemMeasurement.succRelocInInterSysPs"/>
  </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>

```

6.2 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：

```

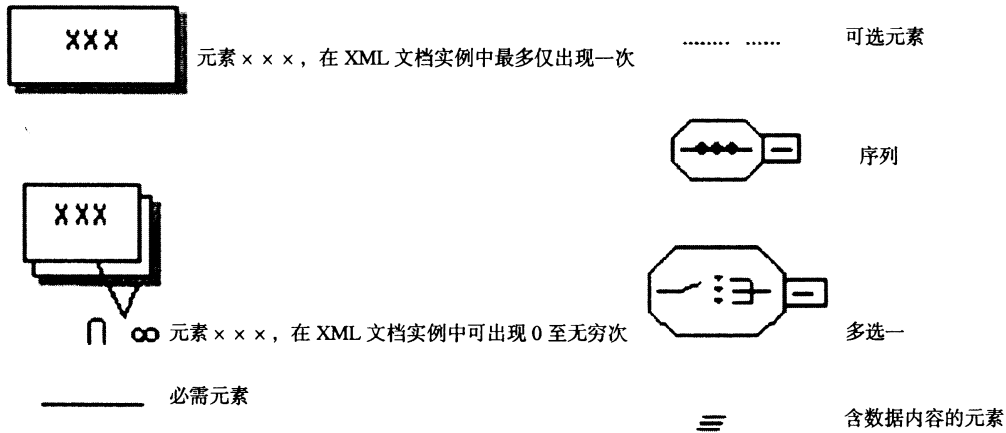
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="MeasDataCollection.xsl"?>
<measCollecFile
  xmlns=
" http://latest/nmc-omc/cmNrm.doc#measCollec "
>

```

附录 A
(规范性附录)

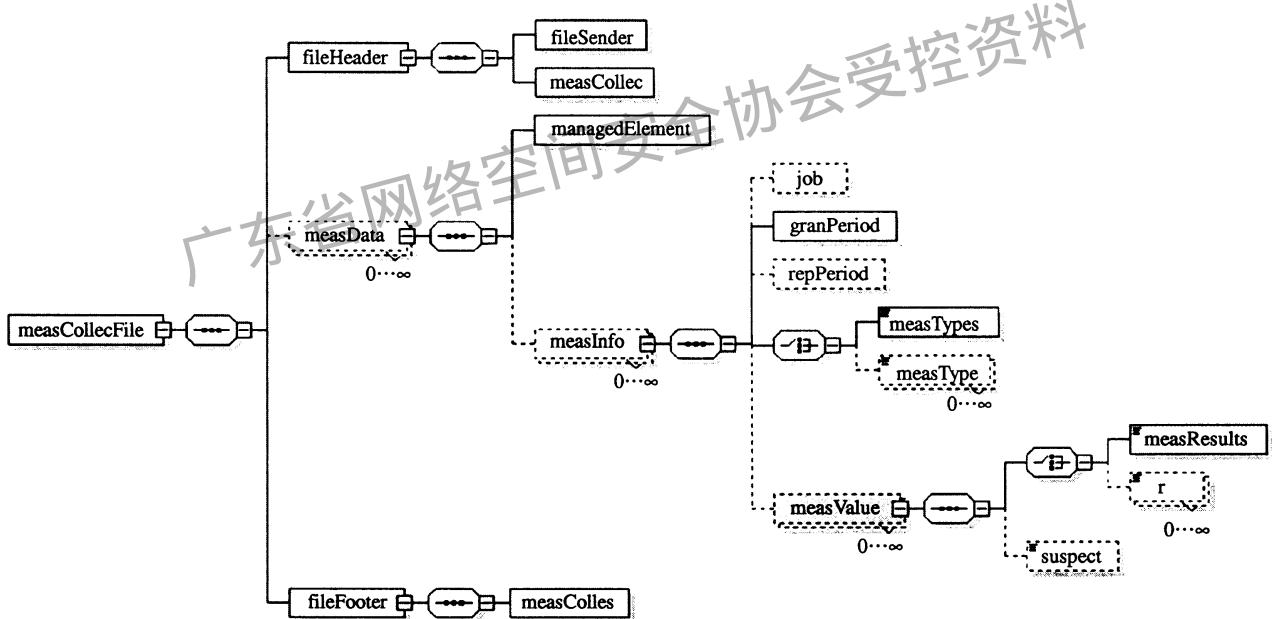
Schema 文档补充说明

A.1 XML Schema文档结构标记约定如下图所示:



A.2 性能测量数据文件的Schema定义<measCollec.xsd>

a) XML Schema 文档结构图如下:



b) XML Schema 文档元素/属性说明如下:

元素/属性 名称	元素/属性 描述
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	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）。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）
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)

表 (续)

元素/属性 名称		元素/属性 描述
fileFooter		文件尾部。包含子元素测量采集结束时间 (measCollec)。
fileSender	localDn	文件发送发。包含两个属性：本地识别名 (localDN)、网元类型 (elementType)
	elementType	
managedElement	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		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)
job		测量任务。该元素为可选元素。由其类型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SubCounter (measNameWithSubCounter) 和不含SubCounter (measNameWithoutSubCounter) 两类。从3GPP规范中扩展而来
measNameWithSubCounter		含SubCounter的数据测量项名称。表示为familyname.measurename.subcounter形式。从3GPP规范中扩展而来
measNameWithoutSubCounter		不含SubCounter的数据测量项名称。表示为familyname.measurename形式。从3GPP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

```
<?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,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"/>
```

```
      <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

```

<?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>

```

参 考 文 献

- 3GPP TS 32.623 Telecommunication management; Configuration Management (CM); Generic network resources Integration Reference Point (IRP); Common Object Request Broker Architecture (CORBA)
- 3GPP TS 32.633 Telecommunication management; Configuration Management (CM); Core Network Resources Integration Reference Point (IRP); Common Object Request Broker Architecture (CORBA)
- 3GPP TS 32.643 Telecommunication management; Configuration Management (CM); UTRAN network resources Integration Reference Point (IRP); Common Object Request Broker Architecture (CORBA)
- 3GPP TS 32.653 Telecommunication management; Configuration Management (CM); GERAN network resources Integration Reference Point (IRP); Common Object Request Broker Architecture (CORBA)

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 WCDMA 数字蜂窝移动通信网网络管理技术要求（第一阶段）
第3部分 基于CORBA技术的网络资源模型设计

YD/T 1586.3-2007

*

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

*

版权所有 不得翻印

*

本书如有印装质量问题，请与本社联系 电话：(010)67114922