

ICS 33.040

M 15



中华人民共和国通信行业标准

YD/T 2333.2-2011

2GHz TD-SCDMA/WCDMA 数字蜂窝移动通信网 演进版高速分组接入（HSPA+）网络管理技术要求 第 2 部分：基于 CORBA 技术的信息模型设计

2GHz TD-SCDMA/WCDMA digital cell mobile communications
network HSPA+ management technical specification
part 2: CORBA based information model design

2011-12-20 发布

2012-02-01 实施

中华人民共和国工业和信息化部 发布

目次

前 言.....II

1 范围.....1

2 规范性引用文件.....1

3 缩略语.....1

4 WCDMA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.....2

4.2 无线接入网网络资源模型的IDL定义.....16

5 WCDMA性能网络资源模型设计.....26

5.1 性能参数的IDL定义.....26

5.2 数据类型的IDL定义.....34

6 WCDMA性能管理接口功能相关的文件.....46

6.1 性能测量数据文件的Schema定义<HspaeMeasCollec.xsd>.....46

6.2 性能测量数据文件的XML header定义.....53

7 TD-SCDMA网络资源模型设计.....54

7.1 通用网络资源模型的IDL定义.....54

7.2 无线接入网网络资源模型的IDL定义.....54

8 TD-SCDMA性能测量数据模型设计.....70

8.1 性能测量数据模型的IDL定义.....70

8.2 数据类型的IDL定义.....80

9 TD-SCDMA性能测量数据接口功能相关的文件.....93

9.1 性能测量数据文件的Schema定义<measCollec.xsd>.....93

9.2 性能测量数据文件的XML header定义.....108

附录A（规范性附录） XML Schema文档补充说明.....109

附录B（资料性附录） 性能管理功能相关XML文件示例.....111

参考文献.....115

前 言

本标准是《2GHz TD-SCDMA/WCDMA数字蜂窝移动通信网演进版高速分组接入（HSPA+）网络管理技术要求》部分标准中的第二部分。该部分标准的结构及名称预计如下：

1. 2GHz TD-SCDMA/WCDMA数字蜂窝移动通信网演进版高速分组接入（HSPA+）网络管理技术要求 第1部分：信息模型

2. 2GHz TD-SCDMA/WCDMA数字蜂窝移动通信网演进版高速分组接入（HSPA+）网络管理技术要求 第2部分：基于CORBA技术的网络资源模型设计

本部分依据GB/T 1.1-2009的规则起草。

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：北京邮电大学、中国联合网络通信集团有限公司、北京市天元网络技术股份有限公司。

本部分主要起草人：芮兰兰、王智立、高 娴、柯小婉、徐克航、陈 冰。

2GHz TD-SCDMA/WCDMA数字蜂窝移动通信网
演进版高速分组接入（HSPA+）网络管理技术要求
第2部分：基于CORBA技术的信息模型设计

1 范围

本部分规定了采用TD-SCDMA/WCDMA Release 7/8 HSPA+接入技术的2GHz数字蜂窝移动通信网的网络管理接口的CORBA/IDL定义。

本部分适用于对采用TD-SCDMA/WCDMA Release 7/8 HSPA+接入技术的2GHz数字蜂窝移动通信网的网络管理。

2 规范性引用文件

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

YD/T 1586.1-2007	2GHz WCDMA数字蜂窝移动通信网网络管理技术要求(第一阶段) 第1部分：配置网络资源模型
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/T1585.1-2007	2GHz TD-SCDMA数字蜂窝移动通信网网络管理技术要求（第二阶段） 第1部分：网络资源模型
YD/T1585.2-2007	2GHz TD-SCDMA数字蜂窝移动通信网网络管理技术要求（第二阶段） 第2部分：性能测量数据
YD/T1586.3-2006	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

3 缩略语

下列缩略语适用于本文件。

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
HSPA+	High Speed Packet Access Plus	演进版高速分组接入

4 WCDMA 配置网络资源模型设计

配置网络资源模型设计中有3类idl文件，这3类文档及其用途如下。

- a) xxxNRMDefs.idl包括GenericNRMDefs.idl、IMDataDefs.idl和UtranNRMDefs.idl，用来定义配置网络对象及其属性名称；
- b) xxxNRMSystem.idl包括GenericNRMSystem.idl和UtranNRMSystem.idl，用来定义配置网络资源对象的属性使用的数据类型；

c) xxxNRMPProfile.idl包括GenericNRMPProfile.idl、IMDataProfile.idl和UtranNRMPProfile.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 GenericNRMDefs

```
//File "GenericNRMDefs.idl"
//The IRP document version number is "Generic NRM V1.0"
#ifndef GenericNRMDefs_idl
#define GenericNRMDefs_idl

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

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

    // Attribute Names
    //
    const string signallingLinkTPId = "signallingLinkTPId";
    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
```

4.1.2 GenericNRMPProfile

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

#include "GenericNRMSystem.idl"

/**
 * 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
    };
};
#endif
```

```

    // notifyAlarmListRebuilt
    // notifyPotentialFaultyAlarmList
};

interface GenericIRP : Top
{
    readonly attribute string iRPIId;
};

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

```

4.1.3 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

```

4.1.4 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 MDataProfile

```

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

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

module IMDataProfile
{
    interface InventoryUnit : GenericNRMPProfile::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 UtranNRMDefs

```

//File "UtranNRMDefs.idl"
//The IRP document version number is "UTRAN NRM for HSDPA V1.0"
#ifndef UtranNRMDefs_idl
#define UtranNRMDefs_idl

#include "GenericNRMDefs.idl"

/**
 * 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 maxThroughput = "maxThroughput";
    };

    //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";
// HSPA+ specific attribute(s)
const string mimoFlag = "mimoFlag";
const string qam64Flag = "qam64Flag";
const string qam16Flag = "qam16Flag";
const string hsscchlessFlag = "hsscchlessFlag";
const string mimoStat = "mimoStat";
const string qam64Stat = "qam64Stat";
const string qam16Stat = "qam16Stat";
};

//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 cellMode = "cellMode";
    const string userLabel = "userLabel";
};

//Definitions for MO class ExternalUtranCell

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

    // Attribute Names
    //
    const string externalUtranCellId = "externalUtranCellId";
    const string 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";
// HSPA+ specific attribute(s)
const string mimoFlag = "mimoFlag";
const string qam64Flag = "qam64Flag";
const string qam16Flag = "qam16Flag";
const string hsscchlessFlag = "hsscchlessFlag";
const string mimoStat = "mimoStat";
const string qam64Stat = "qam64Stat";
const string qam16Stat = "qam16Stat";
};

```

//Definitions 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";
};

```

//Definitions for MO class 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

```

4.2.2 UtranNRMPProfile

```

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

#include "GenericNRMPProfile.idl"
#include "GenericNRMDefs.idl"
#include "UtranNRMSystem.idl"

/**
 * 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 maxThroughput;

```

```

    // 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 : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSysyem::ObjectIdType utranCellId;
    readonly attribute GenericNRMSysyem::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;
    readonly attribute float primaryCpichPower;
    readonly attribute float maximumTransmissionPower;
    readonly attribute float primarySchPower;
    readonly attribute float secondarySchPower;
    readonly attribute float bchPower;
    readonly attribute unsigned long lac;
        readonly attribute unsigned long rac;
        readonly attribute unsigned long sac;
    readonly attribute UtranNRMSysyem::UraListType uraList;
    readonly attribute UtranNRMSysyem::CellModeEnumType cellMode;
    // HSPA+ specific attribute(s)
        readonly attribute unsigned long mimoFlag;
        readonly attribute unsigned long qam64Flag;
        readonly attribute unsigned long qam16Flag ;
        readonly attribute unsigned long hsscchlessFlag ;
            attribute unsigned long mimoStat;
            attribute unsigned long qam64Stat;
            attribute unsigned long qam16Stat;

    // 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;
    readonly attribute float primaryCpichPower;
    readonly attribute unsigned long lac;
    readonly attribute UtranNRMSystem::CellModeEnumType cellMode;
        attribute string userLabel;

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

interface ExternalUtranCell : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType externalUtranCellId;
        attribute unsigned long mcc;
        attribute unsigned long mnc;
        attribute unsigned long cId;
        attribute unsigned long rnId;
        attribute unsigned long uarfcnUl;
        attribute unsigned long uarfcnDl;
        attribute unsigned long primaryScramblingCode;
        attribute float primaryCpichPower;
        attribute unsigned long lac;
        attribute unsigned long rac;
    readonly attribute UtranNRMSystem::CellModeEnumType cellMode;
    // HSPA+ specific attribute(s)
    readonly attribute unsigned long mimoFlag;

```

```

        readonly attribute unsigned long qam64Flag;
        readonly attribute unsigned long qam16Flag ;
        readonly attribute unsigned long hsscchlessFlag ;
            attribute unsigned long mimoStat;
            attribute unsigned long qam64Stat;
            attribute unsigned long qam16Stat;

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

interface GsmRelation : GenericNRMPProfile::Top
{
    readonly attribute GenericNRMSystem::ObjectIdType gsmRelationId;
    attribute GenericNRMSystem::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 GenericNRMSystem::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

```

4.2.3 UtranNRMSysytem

```

//File "UtranNRMSysytem.idl"
//The IRP document version number is "UTRAN NRM for HSDPA V1.0"
#ifndef UtranNRMSysytem_idl
#define UtranNRMSysytem_idl

#include "GenericNRMSysytem.idl"

module UtranNRMSysytem
{
    /**
     * 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
     * GenericNRMSysytem.
     */

    union AdjacentCellType switch(boolean)
    {
        case TRUE: GenericNRMSysytem::DN    utranCell;
        case FALSE: string cellGlobalId;
    };

    typedef GenericNRMSysytem::ULongSet UraListType;

    enum CellModeEnumType
    {
        FDDMode,
        TDDMode_3_84Mcps,
        TDDMode_1_28Mcps
    };
};
#endif

```


5 WCDMA 性能网络资源模型设计

5.1 性能参数的 IDL 定义

5.1.1 HspaeMeasCollecDefs

```

//File HspaeMeasCollecDefs.idl
#ifndef HspaeMeasCollecDefs_idl
#define HspaeMeasCollecDefs_idl

/**
 * This module defines measurementType names constants
 */

module HspaeMeasCollecDefs
{
    //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 mncHardHandoverMeasurement
{
    const string attHho= "attHho";
    const string failHhoPerCause= "failHhoPerCause";
};

module mncRelocationMeasurement
{
    //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 rncInterSystemHandoverMeasurement
{
    //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 iuDlSigThroughputCs = "iuDlSigThroughputCs";
    const string iuUIDataThroughputCsPerType = "iuUIDataThroughputCsPerType";
    const string iuDlDataThroughputCsPerType = "iuDlDataThroughputCsPerType";
    const string iuUISigThroughputPs = "iuUISigThroughputPs";
    const string iuDlSigThroughputPs = "iuDlSigThroughputPs";
    const string iuUIDataThroughputPsPerType = "iuUIDataThroughputPsPerType";
    const string iuDlDataThroughputPsPerType = "iuDlDataThroughputPsPerType";
};

module iurInterfaceThroughputMeasurement
{
    const string iurUISigThroughput = "iurUISigThroughput";
    const string iurDlSigThroughput = "iurDlSigThroughput";
    const string iurUIDataThroughput = "iurUIDataThroughput";
    const string iurDlDataThroughput = "iurDlDataThroughput";
};

```

```

};

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

//Cell 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 iubRlManagementMeasurement
{
    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";
};

//UtranCell measurement for HSPA+
module HSPAЕ
{
    const string NbrMimoSingleStr = " NbrMimoSingleStr ";
    const string NbrMimoDoubleStr = "Nbr MimoDoubleStr ";
    const string NbrQam64Scheduled = " NbrQam64Scheduled ";
    const string NbrQam16Scheduled = " NbrQam16Scheduled ";
    const string NbrCpcDtxUser = " NbrCpcDtxUser ";
    const string NbrCpcDrxUser = "Nbr CpcDrxUser ";
    const string AttEfachEstab = " AttEfachEstab ";
    const string SuccEfachEstab = " SuccEfachEstab ";
    const string EfachMaxDelay = " EfachMaxDelay ";
    const string EfachMeanDelay = " EfachMeanDelay ";
    const string NbrEfachMaxUser = " NbrEfachMaxUser ";
    const string NbrEfachMeanUser = " NbrEfachMeanUser ";
    const string NbrEpchMaxUser = " NbrEpchMaxUser ";
}

```

```

const string NbrEpchMeanUser = " NbrEpchMeanUser ";
const string NbrOctEfachIubPdu = " NbrOctEfachIubPdu ";
const string SuccRabSetup = " SuccRabSetup";
const string FailRabSetup = " FailRabSetup ";
const string NbrRabRelUsrInact = " NbrRabRelUsrInact ";
const string NbrRabRelCn = " NbrRabRelCn ";
const string DualCellSchedulingDuration = " DualCellSchedulingDuration ";
const string IndividualCellSchedulingDuration = " IndividualCellSchedulingDuration ";
const string EstabAttDCHSDSCH = " EstabAttDCHSDSCH ";
const string EstabSuccDCHSDSCH = " EstabSuccDCHSDSCH ";
const string EstabFailDCHSDSCH = " EstabFailDCHSDSCH ";
const string AttRabEstabPS = " AttRabEstabPS ";
const string SuccRabEstabPS = " SuccRabEstabPS ";
const string UpDown11="<1><1>";
const string UpDown12="<1><2>";
const string UpDown13="<1><3>";
const string UpDown14="<1><4>";
const string UpDown15="<1><5>";
const string UpDown16="<1><6>";
const string UpDown21="<2><1>";
const string UpDown22="<2><2>";
const string UpDown23="<2><3>";
const string UpDown24="<2><4>";
const string UpDown25="<2><5>";
const string UpDown26="<2><6>";
const string UpDown31="<3><1>";
const string UpDown32="<3><2>";
const string UpDown33="<3><3>";
const string UpDown34="<3><4>";
const string UpDown35="<3><5>";
const string UpDown36="<3><6>";
const string UpDown41="<4><1>";
const string UpDown42="<4><2>";
const string UpDown43="<4><3>";
const string UpDown44="<4><4>";
const string UpDown45="<4><5>";
const string UpDown46="<4><6>";
const string UpDown51="<5><1>";
const string UpDown52="<5><2>";
const string UpDown53="<5><3>";
const string UpDown54="<5><4>";
const string UpDown55="<5><5>";
const string UpDown56="<5><6>";

```

```

const string UpDown61="<6><1>";
const string UpDown62="<6><2>";
const string UpDown63="<6><3>";
const string UpDown64="<6><4>";
const string UpDown65="<6><5>";
const string UpDown66="<6><6>";

};

//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";
};

//GsmRelation measurement
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

5.2.1 HspaeMeasCollecSystem

```

//File "HspaeMeasCollecSystem.idl"
#ifndef HspaeMeasCollecSystem_idl
#define HspaeMeasCollecSystem_idl

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

    // 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; // HSPA+ specified
const RANAPCause requestedCipheringAndOrIntegrityProtectionAlgorithmsNotSupported = 12;
const RANAPCause conflictWithAlreadyExistingIntegrityProtectionAndOrCipheringInformation = 13;
const RANAPCause failureInTheRadioInterfaceProcedure = 14; // HSPA+ specified
const RANAPCause releaseDueToUtranGeneratedReason = 15;
const RANAPCause userInactivity_RANAP = 16; // HSPA+ specified
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 requestedMaximumBitRateForDlNotAvailable = 33;
const RANAPCause requestedMaximumBitRateForUlNotAvailable = 34;
const RANAPCause requestedGuaranteedBitRateForDlNotAvailable = 35;
const RANAPCause requestedGuaranteedBitRateForUlNotAvailable = 36;
const RANAPCause repeatedIntegrityCheckingFailure = 37;
const RANAPCause requestedRequestTypeNotSupported = 38;
const RANAPCause requestSuperseded = 39;
const RANAPCause releaseDueToUeGeneratedSignallingConnectionRelease = 40; // HSPA+ specified
const RANAPCause resourceOptimisationRelocation = 41; // HSPA+ specified
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; // HSPA+ specified

//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 riAlreadyActivatedAllocated_RNSAP = 27;
const RNSAPCause numberOfULCodesNotSupported_RNSAP = 28;
const RNSAPCause cellReservedForOperatorUse = 29;
const RNSAPCause dpcModeChangeNotSupported_RNSAP = 30;
const RNSAPCause informationTemporarilyNotAvailable_RNSAP = 31;
const RNSAPCause informationProvisionNotSupportedForTheObject_RNSAP = 32;
const RNSAPCause powerBalancingStatusNotCompatible_RNSAP = 33;
const RNSAPCause delayedActivationNotSupported_RNSAP = 34;
const RNSAPCause riTimingAdjustmentNotSupported_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 rlAlreadyActivatedAllocated_NBAP = 6;
const NBAPCause nodeBResourcesUnavailable = 7;
const NBAPCause measurementNotSupportedForTheObject_NBAP = 8;
const NBAPCause combiningResourcesNotAvailable_NBAP = 9;
const NBAPCause requestedConfigurationNotSupported_NBAP = 10;
const NBAPCause synchronizationFailure = 11;
const NBAPCause priorityTransportChannelEstablished = 12;
const NBAPCause sibOriginationInNodeBNotSupported = 13;
const NBAPCause requestedTxDiversityModeNotSupported_NBAP = 14;
const NBAPCause unspecified_RNL_NBAP = 15;
const NBAPCause bcchSchedulingError = 16;
const NBAPCause measurementTemporarilyNotAvailable_NBAP = 17;
const NBAPCause invalidCmSetting = 18;
const NBAPCause reconfigurationCfnNotElapsed_NBAP = 19;
const NBAPCause numberOfDlCodesNotSupported_NBAP = 20;
const NBAPCause scpichNotSupported = 21;
const NBAPCause combiningNotSupported_NBAP = 22;
const NBAPCause ulSfNotSupported = 23;
const NBAPCause dlSfNotSupported = 24;
const NBAPCause commonTransportChannelTypeNotSupported_NBAP = 25;
const NBAPCause dedicatedTransportChannelTypeNotSupported_NBAP = 26;

```



```

const NBAPCause downlinkSharedChannelTypeNotSupported = 27;
const NBAPCause uplinkSharedChannelTypeNotSupported = 28;
const NBAPCause cmNotSupported_NBAP = 29;
const NBAPCause txDiversityNoLongerSupported = 30;
const NBAPCause unknownLocalCellId = 31;
const NBAPCause numberOfUICodesNotSupported_NBAP = 32;
const NBAPCause informationTemporarilyNotAvailable_NBAP = 33;
const NBAPCause informationProvisionNotSupportedForTheObject_NBAP = 34;
const NBAPCause cellSynchronisationNotSupported = 35;
const NBAPCause cellSynchronisationAdjustmentNotSupported = 36;
const NBAPCause dpcModeChangeNotSupported_NBAP = 37;
const NBAPCause ipdlAlreadyActivated = 38;
const NBAPCause ipdlNotSupported = 39;
const NBAPCause ipdlParametersNotAvailable = 40;
const NBAPCause frequencyAcquisitionNotSupported = 41;
const NBAPCause powerBalancingStatusNotCompatible_NBAP = 42;
const NBAPCause requestedTypeOfBearerRearrangementNotSupported = 43;
const NBAPCause signallingBearerRearrangementNotSupported = 44;
const NBAPCause bearerRearrangementNeeded = 45;
const NBAPCause delayedActivationNotSupported_NBAP = 46;
const NBAPCause 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 conditionalError_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 requestRelatedToPossiblyDuplicatedPacketsAlreadyFulfilled = 252;
const ActPdpContextUtmsFailureCause requestAlreadyFulfilled = 253;
const ActPdpContextUtmsFailureCause sequenceNumbersOfReleasedOrCancelledPacketsIeIncorrect = 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 conditionalError_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 WCDMA 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<HspaeMeasCollec.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： 1.0

```
<?xml version="1.0" encoding="UTF-8"?>
<!--REFERENCE: 3GPP TS 32.435 V7.2.0 measCollec.xsd-->
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:mc="std://yd-t/hsdpa/itf-n/mc/2007"
targetNamespace="std://yd-t/hsdpa/itf-n/mc/2007" elementFormDefault="qualified"
attributeFormDefault="unqualified" version="1.0">
  <xsd:element name="measCollecFile">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="fileHeader">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="fileSender">
                <xsd:complexType>
```

ME resides-->

```
<xsd:attribute name="localDn" type="xsd:string" use="optional"/>
<xsd:attribute name="elementType" type="xsd:string" use="optional"/>
<xsd:attribute name="inDomain" type="mc:domainType" use="optional"/>
<!--add an optional attribute "inDomain" indicating in which domain the
ME resides-->
</xsd:complexType>
</xsd:element>
<xsd:element name="measCollec">
  <xsd:complexType>
    <xsd:attribute name="beginTime" type="xsd:dateTime" use="required"/>
  </xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="fileFormatVersion" type="xsd:string" use="required"/>
<xsd:attribute name="vendorName" type="xsd:string" use="optional"/>
<xsd:attribute name="dnPrefix" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="measData" minOccurs="0" maxOccurs="unbounded">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="managedElement">
        <xsd:complexType>
          <xsd:attribute name="localDn" type="xsd:string" use="optional"/>
          <xsd:attribute name="userLabel" type="xsd:string" use="optional"/>
          <xsd:attribute name="swVersion" type="xsd:string" use="optional"/>
          <xsd:attribute name="inDomain" type="mc:domainType" use="optional"/>
          <!--add an optional attribute "inDomain" indicating in which domain
the ME resides-->
```

the ME resides-->

use="required"/>

```
</xsd:complexType>
</xsd:element>
<xsd:element name="measInfo" minOccurs="0" maxOccurs="unbounded">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="job" minOccurs="0">
        <xsd:complexType>
          <xsd:attribute name="jobId" type="xsd:string"
use="required"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="granPeriod">
        <xsd:complexType>
          <xsd:attribute name="duration" type="xsd:duration"
```


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


```

maxOccurs="unbounded">
                                <xsd:element name="r" minOccurs="0"
                                <xsd:complexType>
                                    <xsd:simpleContent>
                                        <xsd:extension
base="mc:measResultType">
                                            <xsd:attribute
name="p" type="xsd:positiveInteger" use="required"/>
                                                </xsd:extension>
                                                </xsd:simpleContent>
                                                </xsd:complexType>
                                            </xsd:element>
                                        </xsd:choice>
                                        <xsd:element name="suspect"
type="xsd:boolean" minOccurs="0"/>
                                            </xsd:sequence>
                                            <xsd:attribute name="measObjLdn"
type="xsd:string" use="required"/>
                                                </xsd:complexType>
                                                </xsd:element>
                                            </xsd:sequence>
                                            </xsd:complexType>
                                        </xsd:element>
                                    </xsd:sequence>
                                    </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="fileFooter">
                                    <xsd:complexType>
                                        <xsd:sequence>
                                            <xsd:element name="measCollec">
                                                <xsd:complexType>
                                                    <xsd:attribute name="endTime" type="xsd:dateTime"
use="required"/>
                                                        </xsd:complexType>
                                                        </xsd:element>
                                                    </xsd:sequence>
                                                    </xsd:complexType>
                                                </xsd:element>
                                            </xsd:sequence>
                                            </xsd:complexType>
                                        </xsd:element>
                                    <!--New attribute type-->
                                    <xsd:simpleType name="domainType">

```

```

    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="CN_CS"/>
      <xsd:enumeration value="CN_PS"/>
      <xsd:enumeration value="UTRAN"/>
      <xsd:enumeration value="IMS"/>
      <!--Legacy compatibility -->
      <xsd:enumeration value="EMS"/>
      <xsd:enumeration value="CAMEL"/>
      <xsd:enumeration value="OTHER"/>
    </xsd:restriction>
  </xsd:simpleType>
  <!--PM name extensions-->
  <xsd:simpleType name="measName">
    <xsd:union memberTypes="mc:measNameWithSubCounter mc:measNameWithoutSubCounter"/>
  </xsd:simpleType>
  <xsd:simpleType name="measResultType">
    <xsd:union memberTypes="xsd:decimal">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="NIL"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:union>
  </xsd:simpleType>
  <xsd:simpleType name="measNameWithSubCounter">
    <xsd:restriction base="xsd:string">
      <xsd:pattern
value="(rabAssignmentMeasurement.attRabAssignEstabCsPerType.|rabAssignmentMeasurement.succRabAssignE
stabCsPerType.|rabAssignmentMeasurement.failRabAssignEstabCsPerCause.|rabAssignmentMeasurement.attRabA
ssignEstabPsPerType.|rabAssignmentMeasurement.succRabAssignEstabPsPerType.|rabAssignmentMeasurement.fa
ilRabAssignEstabPsPerCause.|rabAssignmentMeasurement.attRabAssignModCsPerType.|rabAssignmentMeasurem
ent.succRabAssignModCsPerType.|rabAssignmentMeasurement.failRabAssignModCsPerCause.|rabAssignmentMea
surement.attRabAssignModPsPerType.|rabAssignmentMeasurement.succRabAssignModPsPerType.|rabAssignme
ntMeasurement.failRabAssignModPsPerCause.|rabAssignmentMeasurement.attRabAssignRelCsPerType.|rabAssign
mentMeasurement.succRabAssignRelCsPerType.|rabAssignmentMeasurement.failRabAssignRelCsPerCause.|rab
AssignmentMeasurement.attRabAssignRelPsPerType.|rabAssignmentMeasurement.succRabAssignRelPsPerType.|r
abAssignmentMeasurement.failRabAssignRelPsPerCause.|rabReleaseRequestMeasurement.nbrRncRelCsRabPerCa
use.|rabReleaseRequestMeasurement.nbrRncRelPsRabPerCause.|iuConnectionMeasurement.nbrRncRelCsIuConnP
erCause.|iuConnectionMeasurement.nbrRncRelPsIuConnPerCause.|iuConnectionMeasurement.attRelCsIuConnP
erCause.|iuConnectionMeasurement.attRelPsIuConnPerCause.|iuInterfaceMeasurement.nbrResetCsByRncPerCause.|i
uInterfaceMeasurement.nbrResetPsByRncPerCause.|iuInterfaceMeasurement.nbrResetCsByCnPerCause.|iuInterfac
eMeasurement.nbrResetPsByCnPerCause.|iuInterfaceMeasurement.nbrResetResCsByRncPerCause.|iuInterfaceMea
surement.nbrResetResPsByRncPerCause.|iuInterfaceMeasurement.nbrResetResCsByCnPerCause.|iuInterfaceMeasu
rement.nbrResetResPsByCnPerCause.|iuInterfaceMeasurement.nbrErrorIndCsByRncPerCause.|iuInterfaceMeasure

```


ment.nbrErrorIndPsByRncPerCause.|iuInterfaceMeasurement.nbrErrorIndCsByCnPerCause.|iuInterfaceMeasureme
nt.nbrErrorIndPsByCnPerCause.|rncSoftHandoverMeasurement.failRIAddInSofterHoPerCause.|rncSoftHandoverM
easurement.failRIDelInSofterHoPerCause.|rncSoftHandoverMeasurement.failRIAddInShoPerCause.|rncSoftHandov
erMeasurement.failRIDelInShoPerCause.|rncHardHandoverMeasurement.failHhoPerCause.|rncRelocationMeasure
ment.attRelocOutPrepWithUeNotInvCsPerCause.|rncRelocationMeasurement.failRelocOutPrepWithUeNotInvCsPe
rCause.|rncRelocationMeasurement.failRelocOutWithUeNotInvCsPerCause.|rncRelocationMeasurement.attRelocO
utPrepWithUeInvCsPerCause.|rncRelocationMeasurement.failRelocOutPrepWithUeInvCsPerCause.|rncRelocation
Measurement.failRelocOutWithUeInvCsPerCause.|rncRelocationMeasurement.attRelocOutPrepWithUeNotInvPsPe
rCause.|rncRelocationMeasurement.failRelocOutPrepWithUeNotInvPsPerCause.|rncRelocationMeasurement.failRe
locOutWithUeNotInvPsPerCause.|rncRelocationMeasurement.attRelocOutPrepWithUeInvPsPerCause.|rncRelocati
onMeasurement.failRelocOutPrepWithUeInvPsPerCause.|rncRelocationMeasurement.failRelocOutWithUeInvPsPer
Cause.|rncRelocationMeasurement.attRelocInWithUeNotInvCsPerCause.|rncRelocationMeasurement.failRelocInW
ithUeNotInvCsPerCause.|rncRelocationMeasurement.attRelocInWithUeInvCsPerCause.|rncRelocationMeasuremen
t.failRelocInWithUeInvCsPerCause.|rncRelocationMeasurement.attRelocInWithUeNotInvPsPerCause.|rncRelocatio
nMeasurement.failRelocInWithUeNotInvPsPerCause.|rncRelocationMeasurement.attRelocInWithUeInvPsPerCause
.|rncRelocationMeasurement.failRelocInWithUeInvPsPerCause.|rncInterSystemHandoverMeasurement.attRelocOut
PrepInterSysCsPerCause.|rncInterSystemHandoverMeasurement.failRelocOutPrepInterSysCsPerCause.|rncInterSys
temHandoverMeasurement.failRelocOutInterSysCsPerCause.|rncInterSystemHandoverMeasurement.attRelocInInte
rSysCsPerCause.|rncInterSystemHandoverMeasurement.failRelocInInterSysCsPerCause.|rncInterSystemHandover
Measurement.failRelocOutInterSysPsPerCause.|iuInterfaceThroughputMeasurement.iuUlDataThroughputCsPerTyp
e.|iuInterfaceThroughputMeasurement.iuDlDataThroughputCsPerType.|iuInterfaceThroughputMeasurement.iuUlDa
taThroughputPsPerType.|iuInterfaceThroughputMeasurement.iuDlDataThroughputPsPerType.|rlcConnectionMeasu
rement.nbrRlcBlockSentPerMode.|rlcConnectionMeasurement.nbrRlcBlockRecvedPerMode.|cellRrcConnectionMea
surement.attRrcConnSetupPerCause.|cellRrcConnectionMeasurement.succRrcConnSetupPerCause.|cellRrcConnec
tionMeasurement.failRrcConnSetupPerCause.|cellRrcConnectionMeasurement.failRrcConnReestabPerCause.|cellS
oftHandoverMeasurement.failRIAddInShoPerCause.|cellSoftHandoverMeasurement.failRIDelInShoPerCause.|hard
HandoverIntraCellMeasurement.failHhoOutIntraCellPerCause.|iubRlManagementMeasurement.failRISetupIubPerC
ause.|iubRlManagementMeasurement.failRIAddIubPerCause.|iurRlManagementMeasurement.failRISetupIurPerCau
se.|iurRlManagementMeasurement.failRIAddIurPerCause.|HSPA.E.SuccRabSetup.|HSPA.E.FailRabSetup.|HSPA.E.
NbrRabRelCn.|HSPA.E.EstabFailDCHSDSCH.|HSPA.E.AttRabEstabPS.|HSPA.E.SuccRabEstabPS.|hardHandoverIn
terCellIntraNodeBMeasurement.failHhoOutInterCellIntraNodeBPerCause.|hardHandoverInterNodeBIntraRncMeas
urement.failHhoOutInterNodeBIntraRncPerCause.|hardHandoverInterRncViaIurMeasurement.failHhoOutInterRnc
ViaIurPerCause.|hardHandoverInterRncMeasurement.failHhoOutInterRncCnPerCause.|hardHandoverInterSystem
Measurement.attRelocOutPrepInterSysCsPerCause.|hardHandoverInterSystemMeasurement.failRelocOutPrepInter
SysCsPerCause.|hardHandoverInterSystemMeasurement.failRelocOutInterSysCsPerCause.|hardHandoverInterSyste
mMeasurement.attRelocInInterSysCsPerCause.|hardHandoverInterSystemMeasurement.failRelocInInterSysCsPerC
ause.|hardHandoverInterSystemMeasurement.failRelocOutInterSysPsPerCause.)

([a-zA-Z0-9.<>\- _]{1,63})"/>

</xsd:restriction>

</xsd:simpleType>

<xsd:simpleType name="measNameWithoutSubCounter">

<xsd:restriction base="xsd:string">

<xsd:enumeration value="iuConnectionMeasurement.attRncEstabCsIuConn"/>

<xsd:enumeration value="iuConnectionMeasurement.attRncEstabPsIuConn"/>

<xsd:enumeration value="iuInterfaceMeasurement.nbrOverloadControlCsByRnc"/>


```

<xsd:enumeration value="iuInterfaceMeasurement.nbrOverloadControlPsByRnc"/>
<xsd:enumeration value="iuInterfaceMeasurement.nbrOverloadControlCsByCn"/>
<xsd:enumeration value="iuInterfaceMeasurement.nbrOverloadControlPsByCn"/>
<xsd:enumeration value="rncSoftHandoverMeasurement.attRIAddInSofterHo"/>
<xsd:enumeration value="rncSoftHandoverMeasurement.attRIDelInSofterHo"/>
<xsd:enumeration value="rncSoftHandoverMeasurement.attRIAddInSho"/>
<xsd:enumeration value="rncSoftHandoverMeasurement.attRIDelInSho"/>
<xsd:enumeration value="rncHardHandoverMeasurement.attHho"/>
<xsd:enumeration value="rncRelocationMeasurement.attRelocOutWithUeNotInvCs"/>
<xsd:enumeration value="rncRelocationMeasurement.attRelocOutWithUeInvCs"/>
<xsd:enumeration value="rncRelocationMeasurement.attRelocOutWithUeNotInvPs"/>
<xsd:enumeration value="rncRelocationMeasurement.attRelocOutWithUeInvPs"/>
<xsd:enumeration value="rncInterSystemHandoverMeasurement.attRelocOutInterSysCs"/>
<xsd:enumeration value="rncInterSystemHandoverMeasurement.attRelocOutInterSysPs"/>
<xsd:enumeration value="rncInterSystemHandoverMeasurement.attRelocInInterSysPs"/>
<xsd:enumeration value="rncInterSystemHandoverMeasurement.succRelocInInterSysPs"/>
<xsd:enumeration value="iuInterfaceThroughputMeasurement.iuUISigThroughputCs"/>
<xsd:enumeration value="iuInterfaceThroughputMeasurement.iuDlSigThroughputCs"/>
<xsd:enumeration value="iuInterfaceThroughputMeasurement.iuUISigThroughputPs"/>
<xsd:enumeration value="iuInterfaceThroughputMeasurement.iuDlSigThroughputPs"/>
<xsd:enumeration value="iurInterfaceThroughputMeasurement.iurUISigThroughput"/>
<xsd:enumeration value="iurInterfaceThroughputMeasurement.iurDlSigThroughput"/>
<xsd:enumeration value="iurInterfaceThroughputMeasurement.iurUIDataThroughput"/>
<xsd:enumeration value="iurInterfaceThroughputMeasurement.iurDIDataThroughput"/>
<xsd:enumeration value="rlcConnectionMeasurement.nbrDiscardedRlcBlocksByRnc"/>
<xsd:enumeration value="rlcConnectionMeasurement.nbrRetransmittedRlcBlocksToUe"/>
<xsd:enumeration value="cellRrcConnectionMeasurement.attRrcConnReestab"/>
<xsd:enumeration value="cellSoftHandoverMeasurement.attRIAddInSho"/>
<xsd:enumeration value="cellSoftHandoverMeasurement.attRIDelInSho"/>
<xsd:enumeration value="hardHandoverIntraCellMeasurement.attHhoOutIntraCell"/>
<xsd:enumeration value="iubRlManagementMeasurement.attRISetupIub"/>
<xsd:enumeration value="iubRlManagementMeasurement.attRIAddIub"/>
<xsd:enumeration value="iubRlManagementMeasurement.attRIDelIub"/>
<xsd:enumeration value="iubRlManagementMeasurement.succRIDelIub"/>
<xsd:enumeration value="iurRlManagementMeasurement.attRISetupIur"/>
<xsd:enumeration value="iurRlManagementMeasurement.attRIAddIur"/>
<xsd:enumeration value="iurRlManagementMeasurement.attRIDelIur"/>
<xsd:enumeration value="iurRlManagementMeasurement.succRIDelIur"/>
<xsd:enumeration value="cellTrafficMeasurement.cellCcchTraffic"/>
<xsd:enumeration value="cellTrafficMeasurement.cellCtchTraffic"/>
<xsd:enumeration value="cellTrafficMeasurement.cellDcchTraffic"/>
<xsd:enumeration value="cellTrafficMeasurement.cellDtchTraffic"/>
<xsd:enumeration value="cellPagingMeasurement.attPagingType1FromUtran"/>
<xsd:enumeration value="cellPagingMeasurement.succPagingType1FromUtran"/>

```



```

        <xsd:enumeration value="cellPagingMeasurement.attPagingType2FromUtran"/>
        <xsd:enumeration
value="hardHandoverInterCellIntraNodeBMeasurement.attHhoOutInterCellIntraNodeB"/>
        <xsd:enumeration
value="hardHandoverInterNodeBIntraRncMeasurement.attHhoOutInterNodeBIntraRnc"/>
        <xsd:enumeration value="hardHandoverInterRncViaIurMeasurement.attHhoOutInterRncViaIur"/>
        <xsd:enumeration value="hardHandoverInterRncMeasurement.attHhoOutInterRncCn"/>
        <xsd:enumeration value="hardHandoverInterSystemMeasurement.attRelocOutInterSysCs"/>
        <xsd:enumeration value="hardHandoverInterSystemMeasurement.attRelocOutInterSysPs"/>
        <xsd:enumeration value="hardHandoverInterSystemMeasurement.attRelocInInterSysPs"/>
        <xsd:enumeration value="hardHandoverInterSystemMeasurement.succRelocInInterSysPs"/>
        <xsd:enumeration value=" HSPA.E.NbrMimoSingleStr "/>
        <xsd:enumeration value=" HSPA.E.NbrMimoDoubleStr "/>
        <xsd:enumeration value=" HSPA.E.NbrQam64Scheduled "/>
        <xsd:enumeration value=" HSPA.E.NbrQam16Scheduled "/>
        <xsd:enumeration value=" HSPA.E.NbrCpcDtxUser "/>
        <xsd:enumeration value=" HSPA.E.NbrCpcDrxUser "/>
        <xsd:enumeration value=" HSPA.E. AttEfachEstab "/>
        <xsd:enumeration value=" HSPA.E. SuccEfachEstab "/>
        <xsd:enumeration value=" HSPA.E.EfachMaxDelay "/>
        <xsd:enumeration value=" HSPA.E.EfachMeanDelay "/>
        <xsd:enumeration value=" HSPA.E.NbrEfachMaxUser "/>
        <xsd:enumeration value=" HSPA.E.NbrEfachMeanUser "/>
        <xsd:enumeration value=" HSPA.E.NbrEpchMaxUser "/>
        <xsd:enumeration value=" HSPA.E.NbrEpchMeanUser "/>
        <xsd:enumeration value=" HSPA.E.NbrOctEfachIubPdu "/>
        <xsd:enumeration value=" HSPA.E.NbrRabRelUsrInact "/>
        <xsd:enumeration value=" HSPA.E.DualCellSchedulingDuration "/>
        <xsd:enumeration value=" HSPA.E.IndividualCellSchedulingDuration "/>
        <xsd:enumeration value=" HSPA.E.EstabAttDCHSDSCH "/>
        <xsd:enumeration value=" HSPA.E.EstabSuccDCHSDSCH "/>
        <!--HSPA+ measurements on UtranCell beginning with family name "HSPA.E"-->
    </xsd:restriction>
</xsd:simpleType>
</xsd: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="HspaeMeasCollec.xsl"?>
<measCollecFile xmlns="std://yd-t/hspae/itf-n/mc/2007">

```

7 TD-SCDMA 网络资源模型设计

网络资源模型设计中有3类idl文件，这3类文件及其用途如下：

- a) xxxNRMDefs.idl包括GenericNRMDefs.idl、IMDataDefs.idl、UtranNRMDefs.idl和CoreNRMDefs.idl，用来定义网络资源对象及其属性名称；
- b) xxxNRMSysytem.idl包括GenericNRMSysytem.idl、UtranNRMSysytem.idl和CoreNRMSysytem.idl，用来定义网络资源对象的属性使用的数据类型；
- c) xxxNRMPProfile.idl 包 括 GenericNRMPProfile.idl 、 IMDataProfile.idl 、 UtranNRMPProfile.idl 和 CoreNRMPProfile.idl，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）。

7.1 通用网络资源模型的 IDL 定义

见YD/T 1586.3—2006中4.1的通用配置资源模型的IDL定义。

7.2 无线接入网网络资源模型的 IDL 定义

7.2.1 UtranNRMDefs

```
//File "UtranNRMDefs.idl"
#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 UserLabel = "UserLabel";
};

//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 ConnectedRncId = "ConnectedRncId";
    const string RelatedIubLink = "RelatedIubLink";
    const string AdministrativeState = "AdministrativeState";
    const string OperationalState = "OperationalState";
    const string Latitude= "Latitude";
    const string Longitude= "Longitude";
    const string UserLabel = "UserLabel";
};

//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 UserLabel = "UserLabel";
    const string RelatedNodeBFunction = "RelatedNodeBFunction";
    const string RelatedUtranCell = "RelatedUtranCell";
    const string IubBandwidth = "IubBandwidth";
};

```

```
//Definitions for MO class UtranCell
```

```
interface UtranCell : GenericNRMDefs::ManagedFunction
```

```
{
```

```
    const string CLASS = "UtranCell";
```

```
    // Attribute Names
```

```
    //
```

```
    const string UtranCellId = "UtranCellId";
```

```
    const string UserLabel = "UserLabel";
```

```
    const string CellLevel = "CellLevel";
```

```
    const string RelatedIubLink = "RelatedIubLink";
```

```
    const string Cid = "Cid";
```

```
    const string LocalCellId = "LocalCellId";
```

```
    const string CellMode = "CellMode";
```

```
    const string MaximumTransmissionPower = "MaximumTransmissionPower";
```

```
    const string CellParameterId = "CellParameterId";
```

```
    const string PrimaryCcpchPower = "PrimaryCcpchPower";
```

```
    const string DwPchPower = "DwPchPower";
```

```
    const string Lac = "Lac";
```

```
    const string Rac = "Rac";
```

```
    const string Sac = "Sac";
```

```
    const string UraList = "UraList";
```

```
    const string OperationalState = "OperationalState";
```

```
    const string PchPower = "PchPower";
```

```
    const string FachPower = "FachPower";
```

```
    const string SacList = "SacList";
```

```
    const string HcsPrio = "HcsPrio";
```

```
    const string MaximumAllowedUITxPower = "MaximumAllowedUITxPower";
```

```
    const string DpchConstantValue = "DpchConstantValue";
```

```
    const string QrxlevMin = "QrxlevMin";
```

```
    const string DeltaQrxlevmin = "DeltaQrxlevmin";
```

```
    const string Qhcs = "Qhcs";
```

```
        // HSPA+ specific attribute(s)
```

```
    const string MimoFlag = "MimoFlag";
```

```
    const string Qam64Flag = "Qam64Flag";
```

```
    const string SpsFlag = "SpsFlag";
```

```
    const string MimoStat = "MimoStat";
```

```
    const string Qam64Stat = "Qam64Stat";
```



```
};
```

```
//Definitions for MO class Carrier
```

```
interface Carrier : GenericNRMDefs::ManagedFunction
```

```
{
```

```
    const string CLASS = "Carrier";
```

```
    // Attribute Names
```

```
    //
```

```
    const string CarrierId = "CarrierId";
```

```
    const string UserLabel = "UserLabel";
```

```
    const string UarfcnType = "UarfcnType";
```

```
    const string Uarfcn = "Uarfcn";
```

```
    const string TimeSlotList = "TimeSlotList";
```

```
    const string OperationalState = "OperationalState";
```

```
};
```

```
//Definitions for MO class UtranRelation
```

```
interface UtranRelation : GenericNRMDefs::Top
```

```
{
```

```
    const string CLASS = "UtranRelation";
```

```
    // Attribute Names
```

```
    //
```

```
    const string UtranRelationId = "UtranRelationId";
```

```
    const string AdjacentCell = "AdjacentCell";
```

```
};
```

```
//Definition for MO class GsmRelation
```

```
interface GsmRelation : GenericNRMDefs::Top
```

```
{
```

```
    const string CLASS = "GsmRelation";
```

```
    //Attribute Names
```

```
    //
```

```
    const string GsmRelationId = "GsmRelationId";
```

```

    const string AdjacentCell = "AdjacentCell";

};

//Definition for MO AntennaFunction

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

    //Attribute Names
    //
    const string AntennaFunctionId = "AntennaFunctionId";
    const string RetFlag = "RetFlag";
    const string AntUtranCellList = "AntUtranCellList";
    const string RetTiltValue = "RetTiltValue";
    const string Bearing = "Bearing";
    const string MaxTiltValue = "MaxTiltValue";
    const string MinTiltValue = "MinTiltValue";
    const string MechanicalOffset = "MechanicalOffset";
    const string RetGroupName = "RetGroupName";
    const string Height = "Height";
    const string BaseElevation= "BaseElevation";
    const string Latitude= "Latitude";
    const string Longitude= "Longitude";
    const string MaxAzimuthValue= "MaxAzimuthValue";
    const string MinAzimuthValue= "MinAzimuthValue";
    const string HorizBeamwidth= "HorizBeamwidth";
    const string VertBeamwidth= "VertBeamwidth";
    const string PatternLabel= "PatternLabel";
};

//Definitions for MO class IurLink

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

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

```

```

const string IurLinkId = "IurLinkId";
const string UserLabel = "UserLabel";
const string ConnectedRncId = "ConnectedRncId";
const string IurBandwidth = "IurBandwidth";
};

//Definitions for MO class IucsLinkRnc

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

    // Attribute Names
    //
    const string IucsLinkRncId = "IucsLinkRncId";
    const string UserLabel = "UserLabel";
    const string ConnectedMgw = "ConnectedMgw";
    const string ConnectedMss = "ConnectedMss";
    const string IucsBandwidth = "IucsBandwidth";
};

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

    // Attribute Names
    //
    const string IupsLinkRncId = "IupsLinkRncId";
    const string UserLabel = "UserLabel";
    const string ConnectedSgsn = "ConnectedSgsn";
    const string IupsBandwidth = "IupsBandwidth";
};

//Definitions for MO class RncExternalUtranCell

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

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

```

```

//
const string RncExternalUtranCellId = "RncExternalUtranCellId";
const string CellGlobalId = "CellGlobalId";
const string CellMode = "CellMode";
const string Uarfcn = "Uarfcn";
const string CellParameterId = "CellParameterId";
const string Lac = "Lac";
const string Rac = "Rac";
const string UserLabel = "UserLabel";
// HSPA+ specific attribute(s)
const string MimoFlag = "MimoFlag";
const string Qam64Flag = "Qam64Flag";
const string SpsFlag = "SpsFlag";
const string MimoStat = "MimoStat";
const string Qam64Stat = "Qam64Stat";

};

//Definitions for MO class RncExternalGsmCell

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

    // including all Attribute Names from
    // MO Class GenericNRMDefs::ManagedFunction
    // additional Attribute Names is as follows
    //
const string RncExternalGsmCellId = "RncExternalGsmCellId";
const string CellGlobalId = "CellGlobalId";
const string BcchFrequency = "BcchFrequency";
const string Ncc = "Ncc";
const string Bcc = "Bcc";
const string Lac = "Lac";
const string Rac = "Rac";
const string Mcc = "Mcc";
const string Mnc = "Mnc";
const string UserLabel = "UserLabel";
};

};

#endif // _UtranNRMDefs_idl

```


7.2.2 UtranNRMSysyem

```

//File "UtranNRMSysyem.idl"

#ifndef _UtranNRMSysyem_idl
#define _UtranNRMSysyem_idl

#include <GenericNRMSysyem.idl>
// This statement must appear after all include statements
#pragma prefix "3gppsa5.org"

module UtranNRMSysyem
{
    /**
     * 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
     * GenericNRMSysyem.
     */

    enum CellLevelType
    {
        MacroCell, //0
        MicroCell, //1
        PicoCell //2
    };

    enum CellModeEnumType
    {
        FDDMode, //0
        TDDMode_1_28Mcps, //1
        TDDMode_3_84Mcps //2
    };

    enum TimeSlotDirectionType
    {
        UL,
        DL
    };

    enum TimeSlotStatusType
    {
        Active,

```

```

        Not_Active
    };
    enum uarfcnType
    {
        MainCarrier, //0 主载波
        SubCarrier //1 辅载波
    };

    enum FlagType
    {
        Disable,
        Enable
    };

    enum AdministrativeStateType
    {
        LOCKED,
        UNLOCKED,
        SHUTTING_DOWN
    };
    enum OperationalStateType
    {
        DISABLED,
        ENABLED
    };

    struct TimeSlotConfigStructType
    {
        unsigned short timeSlotId;
        TimeSlotDirectionType timeSlotDirection;
        TimeSlotStatusType timeSlotStatus;
    };
    typedef sequence<TimeSlotConfigStructType> TimeSlotListConfigStructType;

    typedef GenericNRMSysytem::ULongSet UraListType;

};
#endif // _UtranNRMSysytem_idl

```

7.2.3 UtranNRMPProfile

```

//File "UtranNRMPProfile.idl"

#ifndef _UtranNRMPProfile_idl
#define _UtranNRMPProfile_idl

#include <GenericNRMPProfile.idl>

#include <GenericNRMDefs.idl>

#include <UtranNRMSystem.idl>
// This statement must appear after all include statements
#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;
        readonly attribute unsigned long RncId;
        readonly attribute unsigned long Mcc;
        readonly attribute unsigned long Mnc;
        attribute string UserLabel;

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

```

```
interface NodeBFunction : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType NodeBFunctionId;
    readonly attribute unsigned long ConnectedRncId;
    readonly attribute string RelatedIubLink;
    attribute UtranNRMSystem::AdministrativeStateType AdministrativeState;
    readonly attribute UtranNRMSystem::OperationalStateType OperationalState;
    readonly attribute float Latitude;
    readonly attribute float Longitude;
    attribute string UserLabel;
    // The following notifications may be sent from this MO,
        // notifyObjectCreation
        // notifyObjectDeletion
        // notifyAttributeValueChange
        // notifyAckStateChanged
        // notifyChangedAlarm
        // notifyClearedAlarm
        // notifyNewAlarm
        // notifyComments
        // notifyStateChange
        // notifyAlarmListRebuilt
        // notifyPotentialFaultyAlarmList
};
```

```
interface IubLink : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType IubLinkId;
    attribute string UserLabel;
    attribute string RelatedNodeBFunction;
    attribute GenericNRMSystem::DNListType RelatedUtranCell;
    readonly attribute float IubBandwidth;

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



```

};

interface UtranCell : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSysyem::ObjectIdType UtranCellId;
    attribute string UserLabel;
    readonly attribute UtranNRMSysyem::CellLevelType CellLevel;
    attribute GenericNRMSysyem::DN RelatedIubLink;
    readonly attribute unsigned long Cid;
    attribute unsigned long LocalCellId;
    readonly attribute UtranNRMSysyem::CellModeEnumType CellMode;
    attribute float MaximumTransmissionPower; //0dBm..50dBm
    readonly attribute unsigned long CellParameterId; //0..127
    attribute float PrimaryCcpchPower; //-15dBm..+40dBm
    attribute float DwPchPower; //-15dBm..+40dBm
    attribute unsigned long Lac;
    attribute unsigned long Rac;
    attribute unsigned long Sac;
    attribute UtranNRMSysyem::UraListType UraList;
    readonly attribute UtranNRMSysyem::OperationalStateType OperationalState;
    attribute float PchPower; //-10..+46
    attribute float FachPower; //-10..+46
    readonly attribute GenericNRMSysyem::ULongSet SacList; //0~65535
    readonly attribute unsigned long HcsPrio; //0..7
    readonly attribute float MaximumAllowedUITxPower; //-50..33
    readonly attribute float DpchConstantValue; //-10..10,..
    readonly attribute float QrxlevMin; //-115..-25
    readonly attribute float DeltaQrxlevmin; //-4..-2
    readonly attribute unsigned long Qhcs; //0..99
    // HSPA+ specific attribute(s)
    readonly attribute unsigned long MimoFlag;
    readonly attribute unsigned long Qam64Flag;
    readonly attribute unsigned long SpsFlag ;
    attribute unsigned long MimoStat;
    attribute unsigned long Qam64Stat;

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

```

```

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

interface Carrier : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType CarrierId;
    attribute string UserLabel;
    readonly attribute UtranNRMSystem::uarfcnType  UarfcnType;
    readonly attribute unsigned long Uarfcn;
    attribute UtranNRMSystem::TimeSlotListConfigStructType TimeSlotList;
    readonly attribute UtranNRMSystem::OperationalStateType OperationalState;

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

interface UtranRelation : GenericNRMPProfile::Top
{
    readonly attribute GenericNRMSystem::ObjectIdType UtranRelationId;
    attribute string AdjacentCell; //cellGlobalId

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

```

```

interface GsmRelation : GenericNRMPProfile::Top
{
    readonly attribute GenericNRMSystem::ObjectIdType GsmRelationId;
    attribute string AdjacentCell; //cellGlobalId

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

interface AntennaFunction : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType AntennaFunctionId;
    attribute GenericNRMSystem::DNListType AntUtranCellList;
    readonly attribute UtranNRMSystem::FlagType RetFlag; //Enable, Disable
    attribute unsigned long RetTiltValue;
    attribute unsigned long Bearing;
    attribute unsigned long MaxTiltValue;
    attribute unsigned long MinTiltValue;
    attribute unsigned long MechanicalOffset;
    attribute string RetGroupName;
    attribute unsigned long Height;
    attribute unsigned long BaseElevation;
    attribute float Latitude;
    attribute float Longitude;
    attribute unsigned long MaxAzimuthValue;
    attribute unsigned long MinAzimuthValue;
    attribute unsigned long HorizBeamwidth;
    attribute unsigned long VertBeamwidth;
    attribute string PatternLabel;

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

```

```

        // notifyAlarmListRebuilt
        // notifyPotentialFaultyAlarmList

};

interface IurLink : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType IurLinkId;
    attribute string UserLabel;
    attribute unsigned long ConnectedRncId;
    readonly attribute float IurBandwidth;

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

interface IucsLinkRnc : GenericNRMPProfile::Top
{
    readonly attribute GenericNRMSystem::ObjectIdType IucsLinkRncId;

    attribute string UserLabel;
    readonly attribute unsigned long ConnectedMgw;
    readonly attribute unsigned long ConnectedMss;
    readonly attribute float IucsBandwidth;

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

```



```

        // notifyAlarmListRebuilt
        // notifyPotentialFaultyAlarmList
    };
interface IupsLinkRnc : GenericNRMPProfile::Top
{
    readonly attribute GenericNRMSystem::ObjectIdType IupsLinkRncId;
    attribute string UserLabel;
    readonly attribute unsigned long ConnectedSgsn;
    readonly attribute float IupsBandwidth;

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

interface RncExternalUtranCell : GenericNRMDefs::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType RncExternalUtranCellId;
    readonly attribute string CellGlobalId;
    readonly attribute UtranNRMSysyem::CellModeEnumType CellMode;
    attribute unsigned long Uarfcn;
    readonly attribute unsigned long CellParameterId; //0..127
    readonly attribute unsigned long Lac;
    readonly attribute unsigned long Rac;
    attribute string UserLabel;
    // HSPA+ specific attribute(s)
    readonly attribute unsigned long MimoFlag;
    readonly attribute unsigned long Qam64Flag;
    readonly attribute unsigned long SpsFlag ;
    attribute unsigned long MimoStat;
    attribute unsigned long Qam64Stat;

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

```

```
        // notifyAttributeValueChange
    };

    interface RncExternalGsmCell : GenericNRMDefs::ManagedFunction
    {

        readonly attribute GenericNRMSystem::ObjectIdType RncExternalGsmCellId;
        readonly attribute string CellGlobalId;
        readonly attribute unsigned long BcchFrequency;
        readonly attribute unsigned long Ncc;
        readonly attribute unsigned long Bcc;
        readonly attribute unsigned long Lac;
        readonly attribute unsigned long Rac;
        readonly attribute unsigned long Mcc;
        readonly attribute unsigned long Mnc;
        attribute string UserLabel;

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

    };
};
#endif // _UtranNRMPProfile_idl
```

8 TD-SCDMA 性能测量数据模型设计

8.1 性能测量数据模型的 IDL 定义

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MeasurementName的某个subcounter值；
- “family.measurementName”可用于获取一个MeasurementName的值。如果该MeasurementName有subcounters，那么所有subcounter的值都应该返回。当设备不支持所有的subcounter时，需要返回所有能支持的subcounter，以及总和值。

—“family”可用于获取该family下的所有MeasurementName的值。

其中，family为YD/T1585.2—2007中4.1定义的族名；subcounter设置时选相应的整数值（取值见下节数据类型的IDL定义“TDSCDMANRMMMeasurementSystem.idl”中的定义，“Sum”表示总和）。

例如：

- (1) “RAB.FailEstabCs.114”可用于获取某个cause（noResourceAvailable）的“FailEstabCs”值。
- (2) “RAB.FailEstabCs”可用于获取“FailEstabCs”的所有subcounters的值。
- (3) “IU.AttRncEstabCsIuConn”可用于获取“AttRncEstabCsIuConn”的值。
- (4) “IU”可用于获取该family下所有MeasurementType的值。

TD-SCDMA无线网性能测量数据模型IDL文件如下:

TDSCDMANRMMeasurementDefs.idl

```
//TDSCDMANRMMeasurementDefs.idl

//File "TDSCDMANRMMeasurementDefs.idl"
#ifndef TDSCDMANRMMeasurementDefs_idl
#define TDSCDMANRMMeasurementDefs_idl

// #pragma prefix "3gppsa5.org"

/**
 * This module defines measurementType names constants
 */
module TDSCDMANRMMeasurementDefs
{
    // for family RAB
    module RAB
    {
        // RNCfunction
        // RAB assignment
        const string AttEstabCs="AttEstabCs";
        const string SuccEstabCs="SuccEstabCs";
        const string FailEstabCs="FailEstabCs";
        const string AttEstabPs="AttEstabPs";
        const string SuccEstabPs="SuccEstabPs";
        const string FailEstabPs="FailEstabPs";
        // RAB Release Request
        const string RelReqCsPerCause="RelReqCsPerCause";
        const string RelReqPsPerCause="RelReqPsPerCause";
        const string RelReqCsPerTraffic="RelReqCsPerTraffic";
        const string RelReqPsPerTraffic="RelReqPsPerTraffic";
        // Utrancell
        const string AttRabAssnEstabCs="AttRabAssnEstabCs";
        const string SuccRabAssnEstabCs="SuccRabAssnEstabCs";
        const string FailRabAssnEstabCs="FailRabAssnEstabCs";
        const string AttRabAssnEstabPs="AttRabAssnEstabPs";
        const string SuccRabAssnEstabPs="SuccRabAssnEstabPs";
        const string FailRabAssnEstabPs="FailRabAssnEstabPs";
        const string RelReqCs="RelReqCs";
        const string RelReqPs="RelReqPs";
        const string NbrRncRelCsRab="NbrRncRelCsRab";
        const string NbrRncRelPsRab="NbrRncRelPsRab";
        const string NbrRabCsRelIuConnPerCause="NbrRabCsRelIuConnPerCause";
```

```

const string NbrRabPsRelIuConnPerCause="NbrRabPsRelIuConnPerCause";
const string NbrRabCsRelIuConnPerTraffic="NbrRabCsRelIuConnPerTraffic";
const string NbrRabPsRelIuConnPerTraffic="NbrRabPsRelIuConnPerTraffic";
};
//for family IU
module IU
{
    // IU connection
    const string AttRncEstabCsIuConn="AttRncEstabCsIuConn";
    const string AttRncEstabPsIuConn="AttRncEstabPsIuConn";
    const string AttConnRelReqUtranCs="AttConnRelReqUtranCs";
    const string AttConnRelReqUtranPs="AttConnRelReqUtranPs";
    const string AttConnRelCnCs="AttConnRelCnCs";
    const string AttConnRelCnPs="AttConnRelCnPs";
    const string NbrRabCsRelIuConnPerCause="NbrRabCsRelIuConnPerCause";
    const string NbrRabPsRelIuConnPerCause="NbrRabPsRelIuConnPerCause";
    const string NbrRabCsRelIuConnPerTraffic="NbrRabCsRelIuConnPerTraffic";
    const string NbrRabPsRelIuConnPerTraffic="NbrRabPsRelIuConnPerTraffic";
};
//for family IUEXCEPT
module IUEXCEPT
{
    const string NbrResetCsByRnc="NbrResetCsByRnc";
    const string NbrResetPsByRnc="NbrResetPsByRnc";
    const string NbrResetCsByCn="NbrResetCsByCn";
    const string NbrResetPsByCn="NbrResetPsByCn";
    const string NbrResetResCsByRnc="NbrResetResCsByRnc";
    const string NbrResetResPsByRnc="NbrResetResPsByRnc";
    const string NbrResetResCsByCn="NbrResetResCsByCn";
    const string NbrResetResPsByCn="NbrResetResPsByCn";
    const string NbrOverloadCtrlCsByRnc="NbrOverloadCtrlCsByRnc";
    const string NbrOverloadCtrlPsByRnc="NbrOverloadCtrlPsByRnc";
    const string NbrOverloadCtrlCsByCn="NbrOverloadCtrlCsByCn";
    const string NbrOverloadCtrlPsByCn="NbrOverloadCtrlPsByCn";
    const string NbrErrorIndCsByRnc="NbrErrorIndCsByRnc";
    const string NbrErrorIndPsByRnc="NbrErrorIndPsByRnc";
    const string NbrErrorIndCsByCn="NbrErrorIndCsByCn";
    const string NbrErrorIndPsByCn="NbrErrorIndPsByCn";
};
//for family BHO
module BHO
{
    // RNCfunction

```



```

const string AttIntraFreqIntraRnc="AttIntraFreqIntraRnc";
const string AttInterFreqIntraRnc="AttInterFreqIntraRnc";
const string FailIntraFreqIntraRnc="FailIntraFreqIntraRnc";
const string FailInterFreqIntraRnc="FailInterFreqIntraRnc";
// Utrancell
// IntraRnc BHO
const string AttOutIntraRncIntraFreq="AttOutIntraRncIntraFreq";
const string AttOutIntraRncInterFreq="AttOutIntraRncInterFreq";
const string FailOutIntraRncIntraFreq="FailOutIntraRncIntraFreq";
const string FailOutIntraRncInterFreq="FailOutIntraRncInterFreq";
const string AttInIntraRncIntraFreq="AttInIntraRncIntraFreq";
const string AttInIntraRncInterFreq="AttInIntraRncInterFreq";
const string FailInIntraRncIntraFreq="FailInIntraRncIntraFreq";
const string FailInIntraRncInterFreq="FailInIntraRncInterFreq";
// UtranRelation
//const string AttOutIntraRncIntraFreq="AttOutIntraRncIntraFreq";
//const string AttOutIntraRncInterFreq="AttOutIntraRncInterFreq";
//const string FailOutIntraRncIntraFreq="FailOutIntraRncIntraFreq";
//const string FailOutIntraRncInterFreq="FailOutIntraRncInterFreq";
};
//for family HHO
module HHO
{
    // RNCfunction
    const string AttIntraFreqIntraRnc="AttIntraFreqIntraRnc"; //与 BHO 相同
    const string AttInterFreqIntraRnc="AttInterFreqIntraRnc"; //与 BHO 相同
    const string FailIntraFreqIntraRnc="FailIntraFreqIntraRnc"; //与 BHO 相同
    const string FailInterFreqIntraRnc="FailInterFreqIntraRnc"; //与 BHO 相同
    const string AttOutInterRnc="AttOutInterRnc";
    const string SuccOutInterRnc="SuccOutInterRnc";
    const string AttInInterRnc="AttInInterRnc";
    const string SuccInInterRnc="SuccInInterRnc";
    // Utrancell
    // InterRnc HHO
    //const string AttInInterRnc="AttInInterRnc";
    //const string SuccInInterRnc="SuccInInterRnc";
    //const string AttOutInterRnc="AttOutInterRnc";
    //const string SuccOutInterRnc="SuccOutInterRnc";
    // IntraRnc HHO
    const string AttOutIntraRncIntraFreq="AttOutIntraRncIntraFreq";
    const string AttOutIntraRncInterFreq="AttOutIntraRncInterFreq";
    const string FailOutIntraRncIntraFreq="FailOutIntraRncIntraFreq";
    const string FailOutIntraRncInterFreq="FailOutIntraRncInterFreq";

```

```

const string AttInIntraRncIntraFreq="AttInIntraRncIntraFreq";
const string AttInIntraRncInterFreq="AttInIntraRncInterFreq";
const string FailInIntraRncIntraFreq="FailInIntraRncIntraFreq";
const string FailInIntraRncInterFreq="FailInIntraRncInterFreq";
// InterRncintercell HHO
const string AttOutInterRncCnIntraFreq="AttOutInterRncCnIntraFreq";
const string AttOutInterRncCnInterFreq="AttOutInterRncCnInterFreq";
const string FailOutInterRncCnIntraFreq="FailOutInterRncCnIntraFreq";
const string FailOutInterRncCnInterFreq="FailOutInterRncCnInterFreq";
// UtranRelation
const string AttOutInterCellIntraFreq="AttOutInterCellIntraFreq";
const string AttOutInterCellInterFreq="AttOutInterCellInterFreq";
const string FailOutInterCellIntraFreq="FailOutInterCellIntraFreq";
const string FailOutInterCellInterFreq="FailOutInterCellInterFreq";
};
//for family RELOC
module RELOC
{
    const string AttOutPrepUeInvolCs="AttOutPrepUeInvolCs";
    const string FailOutPrepUeInvolCs="FailOutPrepUeInvolCs";
    const string AttOutWithUeInvolCs="AttOutWithUeInvolCs";
    const string FailRelocOutWithUeInvolCs="FailRelocOutWithUeInvolCs";
    const string SuccOutWithUeInvolCs="SuccOutWithUeInvolCs";
    const string AttOutPrepWithUeInvolPs="AttOutPrepWithUeInvolPs";
    const string FailOutPrepWithUeInvolPs="FailOutPrepWithUeInvolPs";
    const string AttOutWithUeInvolPs="AttOutWithUeInvolPs";
    const string FailOutWithUeInvolPs="FailOutWithUeInvolPs";
    const string SuccOutWithUeInvolPs="SuccOutWithUeInvolPs";
    const string AttInUeInvolCs="AttInUeInvolCs";
    const string FailInUeInvolCs="FailInUeInvolCs";
    const string AttInUeInvolPs="AttInUeInvolPs";
    const string FailInUeInvolPs="FailInUeInvolPs";
};
//for family IRATHO
module IRATHO
{
    // RNCfunction
    const string AttRelocPrepOutCs="AttRelocPrepOutCs";
    const string FailRelocPrepOutCs="FailRelocPrepOutCs";
    const string AttOutCs="AttOutCs";
    const string FailOutCs="FailOutCs";
    const string AttIncCs="AttIncCs";
    const string FailIncCs="FailIncCs";

```

```

const string AttOutPsUtran="AttOutPsUtran";
const string FailOutPsUtran="FailOutPsUtran";
const string AttRelocInInterSysPs="AttRelocInInterSysPs";
const string SuccRelocInInterSysPs="SuccRelocInInterSysPs";
// Utrancell
    //const string AttIncCs="AttIncCs";
    //const string FailIncCs="FailIncCs";
    //const string AttRelocInInterSysPs="AttRelocInInterSysPs";
    //const string SuccRelocInInterSysPs="SuccRelocInInterSysPs";
// GsmRelation
    //const string AttRelocPrepOutCs="AttRelocPrepOutCs";
    //const string FailRelocPrepOutCs="FailRelocPrepOutCs";
    //const string AttOutCs="AttOutCs";
    //const string FailOutCs="FailOutCs";
    //const string AttOutPsUtran="AttOutPsUtran";
    //const string FailOutPsUtran="FailOutPsUtran";
};
//for family IUCSOCT
module IUCSOCT
{
    const string RanapOutCs="RanapOutCs";
    const string RanapInCs="RanapInCs";
    const string UpOutCs="UpOutCs";
    const string UpInCs="UpInCs";
    // common
    const string Conv="Conv";
    const string Strm="Strm";
    const string Intact="Intact";
    const string Bgrd="Bgrd";
};
//for family IUPSOCT
module IUPSOCT
{
    const string RanapOutPs="RanapOutPs";
    const string RanapInPs="RanapInPs";
    const string UpOutPs="UpOutPs";
    const string UpInPs="UpInPs";
    // common
    const string Conv="Conv";
    const string Strm="Strm";
    const string Intact="Intact";
    const string Bgrd="Bgrd";
};

```

```

//for family IUBOCT
module IUBOCT
{
    const string NbapIn="NbapIn";
    const string NbapOut="NbapOut";
    const string UpFpIn="UpFpIn";
    const string UpFpOut="UpFpOut";
};

//for family ATM
module ATM
{
    const string OutOctIucs="OutOctIucs";
    const string InOctIucs="InOctIucs";
    const string OutOctIups="OutOctIups";
    const string InOctIups="InOctIups";
    const string OctInIub="OctInIub";
    const string OctOutIub="OctOutIub";
};

//for family RLC
module RLC
{
    // RNCfunction
    const string CsTraffic="CsTraffic";
    const string PsUIOct="PsUIOct";
    const string PsDIOct="PsDIOct";
    // Utrancell
    const string TrafficUI="TrafficUI";
    const string TrafficDI="TrafficDI";
    // const string CsTraffic="CsTraffic";
    // const string PsUIOct="PsUIOct";
    // const string PsDIOct="PsDIOct";
    const string DTCH="DTCH";
    const string CTCH="CTCH";
    //common
    const string Traf4="<4>";
    const string Traf5="<5>";
    const string Traf6="<6>";
    const string Traf7="<7>";
    const string Traf8="<8>";
    const string Traf10="<10>";
};

//for family RRC
module RRC

```



```

{
    // RRC setup
    const string AttConnEstab="AttConnEstab";
    const string SuccConnEstab="SuccConnEstab";
    const string FailConnEstab="FailConnEstab";
    // RRC num
    const string MeanConn="MeanConn";
    const string MaxConn="MaxConn";
};

//for family DCA
module DCA
{
    const string AttIntraCellIntraFreq="AttIntraCellIntraFreq";
    const string FailIntraCellIntraFreq="FailIntraCellIntraFreq";
    const string AttIntraCellInterFreq="AttIntraCellInterFreq";
    const string FailIntraCellInterFreq="FailIntraCellInterFreq";
};

//for family RLM
module RLM
{
    const string AttRlSetupIubUtranSide="AttRlSetupIubUtranSide";
    const string FailRlSetupIubUtranSide="FailRlSetupIubUtranSide";
    const string AttRlAddIubUtranSide="AttRlAddIubUtranSide";
    const string FailRlAddIubUtranSide="FailRlAddIubUtranSide";
    const string AttRlDelIubUtranSide="AttRlDelIubUtranSide";
    const string SuccRlDelIubUtranSide="SuccRlDelIubUtranSide";
};

//for family PAGING
module PAGING
{
    const string AttInReCellPagingType1="AttInReCellPagingType1";
    const string SuccCellPagingType1="SuccCellPagingType1";
    const string AttCellPagingType2="AttCellPagingType2";
    const string FailTranCellPagingType1="FailTranCellPagingType1";
    const string CongCellPagingType1="CongCellPagingType1";
};

//for family HSPA
module HSPA
{
    const string NbrMimoSingleStr = " NbrMimoSingleStr ";
    const string NbrMimoDoubleStr = "Nbr MimoDoubleStr ";
    const string NbrQam64Scheduled = " NbrQam64Scheduled ";
    const string NbrQam16Scheduled = " NbrQam16Scheduled ";
};

```

```
const string NbrCpcDrxUser = "Nbr CpcDrxUser ";
const string AttEfachEstab = " AttEfachEstab ";
const string SuccEfachEstab = " SuccEfachEstab ";
const string EfachMaxDelay = " EfachMaxDelay ";
const string EfachMeanDelay = " EfachMeanDelay ";
const string NbrEfachMaxUser = " NbrEfachMaxUser ";
const string NbrEfachMeanUser = " NbrEfachMeanUser ";
const string NbrEpchMaxUser = " NbrEpchMaxUser ";
const string NbrEpchMeanUser = " NbrEpchMeanUser ";
const string NbrOctEfachIubPdu = " NbrOctEfachIubPdu ";
const string SuccRabSetup = " SuccRabSetup";
const string FailRabSetup = " FailRabSetup ";
const string NbrRabRelUsrInact = " NbrRabRelUsrInact ";
const string NbrRabRelCn = " NbrRabRelCn ";
const string AttRabEstabPS = " AttRabEstabPS ";
const string SuccRabEstabPS = " SuccRabEstabPS ";

const string UpDown11="<1><1>";
const string UpDown12="<1><2>";
const string UpDown13="<1><3>";
const string UpDown14="<1><4>";
const string UpDown15="<1><5>";
const string UpDown16="<1><6>";
const string UpDown21="<2><1>";
const string UpDown22="<2><2>";
const string UpDown23="<2><3>";
const string UpDown24="<2><4>";
const string UpDown25="<2><5>";
const string UpDown26="<2><6>";
const string UpDown31="<3><1>";
const string UpDown32="<3><2>";
const string UpDown33="<3><3>";
const string UpDown34="<3><4>";
const string UpDown35="<3><5>";
const string UpDown36="<3><6>";
const string UpDown41="<4><1>";
const string UpDown42="<4><2>";
const string UpDown43="<4><3>";
const string UpDown44="<4><4>";
const string UpDown45="<4><5>";
const string UpDown46="<4><6>";
const string UpDown51="<5><1>";
const string UpDown52="<5><2>";
const string UpDown53="<5><3>";
```

```

const string UpDown54="<5><4>";
const string UpDown55="<5><5>";
const string UpDown56="<5><6>";
const string UpDown61="<6><1>";
const string UpDown62="<6><2>";
const string UpDown63="<6><3>";
const string UpDown64="<6><4>";
const string UpDown65="<6><5>";
const string UpDown66="<6><6>";

};

//for family CARR
module CARR
{
    const string TddMeanTcp="TddMeanTcp";
    const string TddMaxTcp="TddMaxTcp";
    const string TddMeanRtwp="TddMeanRtwp";
    const string TddMaxRtwp="TddMaxRtwp";
    const string Ts0="Ts0";
    const string Ts1="Ts1";
    const string Ts2="Ts2";
    const string Ts3="Ts3";
    const string Ts4="Ts4";
    const string Ts5="Ts5";
    const string Ts6="Ts6";
};

//for family CR
module CR
{
    const string CodeUpResUsed="CodeUpResUsed";
    const string CodeDownResUsed="CodeDownResUsed";
    const string NbrAssnBruUl="NbrAssnBruUl";
    const string NbrAssnBruDl="NbrAssnBruDl";
    const string R4="R4";
    const string EDCH="EDCH";
    const string Ctrl="Ctrl";
    const string Hsdpa="Hsdpa";
    const string Mbms="Mbms";
    const string SF1="SF1";
    const string SF2="SF2";

```

```
const string SF4="SF4";
const string SF8="SF8";
const string SF16="SF16";
};
const string Sum="Sum";
const string Conv="Conv";
const string Strm="Strm";
const string Intact="Intact";
const string Bgrd="Bgrd";
const string UpDown11="<1><1>";
const string UpDown22="<2><2>";
const string UpDown44="<4><4>";
const string UpDown55="<5><5>";
const string UpDown56="<5><6>";
const string UpDown58="<5><8>";
const string UpDown66="<6><6>";
const string UpDown67="<6><7>";
const string UpDown68="<6><10>";
};
#endif
```

8.2 数据类型的 IDL 定义

TD-SCDMA无线网性能测量数据模型数据类型的IDL定义文件如下：

TDSCDMANRMMeasurementSystem.idl

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

// #pragma prefix "3gppsa5.org"

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

    typedef unsigned long TDSCDMAMeasurementType1;
    typedef float TDSCDMAMeasurementType2;

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



```

//Radio Network Layer Cause. Value range is 1 - 64.
const RANAPCause rabPreempted = 1;
const RANAPCause trelocoverallExpiry = 2;
const RANAPCause trelocprepExpiry = 3;
const RANAPCause treloccompleteExpiry = 4;
const RANAPCause tqueingExpiry = 5;
const RANAPCause relocationTriggered = 6;
const RANAPCause trelocallocExpiry = 7;
const RANAPCause unableToEstablishDuringRelocation = 8;
const RANAPCause unknownTargetRnc = 9;
const RANAPCause relocationCancelled = 10;
const RANAPCause successfulRelocation = 11; // HSPA+ specified
const RANAPCause requestedCipheringAndOrIntegrityProtectionAlgorithmsNotSupported = 12;
const RANAPCause conflictWithAlreadyExistingIntegrityProtectionAndOrCipheringInformation = 13;
const RANAPCause failureInTheRadioInterfaceProcedure = 14; // HSPA+ specified
const RANAPCause releaseDueToUtranGeneratedReason = 15;
const RANAPCause userInactivity_RANAP = 16; // HSPA+ specified
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 requestedMaximumBitRateForDlNotAvailable = 33;
const RANAPCause requestedMaximumBitRateForUlNotAvailable = 34;
const RANAPCause requestedGuaranteedBitRateForDlNotAvailable = 35;
const RANAPCause requestedGuaranteedBitRateForUlNotAvailable = 36;
const RANAPCause repeatedIntegrityCheckingFailure = 37;
const RANAPCause requestedRequestTypeNotSupported = 38;
const RANAPCause requestSuperseded = 39;
const RANAPCause releaseDueToUeGeneratedSignallingConnectionRelease = 40; // HSPA+ specified
const RANAPCause resourceOptimisationRelocation = 41; // HSPA+ specified

```

```

const RANAPCause requestedInformationNotAvailable = 42;
const RANAPCause relocationDesirableForRadioReasons = 43;
const RANAPCause relocationNotSupportedInTargetRncOrTargetSystem = 44;
const RANAPCause directedRetry = 45;
const RANAPCause radioConnectionWithUeLost = 46;
const RANAPCause rncUnableToEstablishAllRfcs = 47;
const RANAPCause decipheringKeysNotAvailable = 48;
const RANAPCause dedicatedAssistanceDataNotAvailable = 49;
const RANAPCause relocationTargetNotAllowed = 50;
const RANAPCause locationReportingCongestion = 51;
const RANAPCause reduceLoadInServingCell = 52;
const RANAPCause noRadioResourcesAvailableInTargetCell = 53;
const RANAPCause geranIuModeFailure = 54;
const RANAPCause accessRestrictedDueToSharedNetworks = 55;
const RANAPCause incomingRelocationNotSupportedDueToPuesbineFeature = 56;
//Transport Layer Cause. Value range is 65 - 80.
const RANAPCause signallingTransportResourceFailure = 65;
const RANAPCause iuTransportConnectionFailedToEstablish = 66;

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

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

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

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

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

```

//Radio Network Layer Cause.

```
const RNSAPCause unknownCid_RNSAP = 1;
const RNSAPCause cellNotAvailable_RNSAP = 2;
const RNSAPCause powerLevelNotSupported_RNSAP = 3;
const RNSAPCause ulScramblingCodeAlreadyInUse = 4;
const RNSAPCause dlRadioResourcesNotAvailable_RNSAP = 5;
const RNSAPCause ulRadioResourcesNotAvailable_RNSAP = 6;
const RNSAPCause measurementNotSupportedForTheObject_RNSAP = 7;
const RNSAPCause combiningResourcesNotAvailable_RNSAP = 8;
const RNSAPCause combiningNotSupported_RNSAP = 9;
const RNSAPCause reconfigurationNotAllowed = 10;
const RNSAPCause requestedConfigurationNotSupported_RNSAP = 11;
const RNSAPCause synchronisationFailure = 12;
const RNSAPCause requestedTxDiversityModeNotSupported_RNSAP = 13;
const RNSAPCause measurementTemporarilyNotAvailable_RNSAP = 14;
const RNSAPCause unspecified_RNL_RNSAP = 15;
const RNSAPCause invalidCmSettings = 16;
const RNSAPCause 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 numberOfULCodesNotSupported_RNSAP = 28;
const RNSAPCause cellReservedForOperatorUse = 29;
const RNSAPCause dpcModeChangeNotSupported_RNSAP = 30;
const RNSAPCause informationTemporarilyNotAvailable_RNSAP = 31;
const RNSAPCause informationProvisionNotSupportedForTheObject_RNSAP = 32;
const RNSAPCause powerBalancingStatusNotCompatible_RNSAP = 33;
const RNSAPCause delayedActivationNotSupported_RNSAP = 34;
const RNSAPCause rlTimingAdjustmentNotSupported_RNSAP = 35;
const RNSAPCause unknownRnti = 36;
```

//Transport Layer Cause.

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



```

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

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

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

//Radio Network Layer Cause.
const NBAPCause unknownCid_NBAP = 1;
const NBAPCause cellNotAvailable_NBAP = 2;
const NBAPCause powerLevelNotSupported_NBAP = 3;
const NBAPCause dlRadioResourcesNotAvailable_NBAP = 4;
const NBAPCause ulRadioResourcesNotAvailable_NBAP = 5;
const NBAPCause rlAlreadyActivatedAllocated_NBAP = 6;
const NBAPCause nodeBResourcesUnavailable = 7;
const NBAPCause measurementNotSupportedForTheObject_NBAP = 8;
const NBAPCause combiningResourcesNotAvailable_NBAP = 9;
const NBAPCause requestedConfigurationNotSupported_NBAP = 10;
const NBAPCause synchronizationFailure = 11;
const NBAPCause priorityTransportChannelEstablished = 12;
const NBAPCause sibOriginationInNodeBNotSupported = 13;
const NBAPCause requestedTxDiversityModeNotSupported_NBAP = 14;
const NBAPCause unspecified_RNL_NBAP = 15;
const NBAPCause bcchSchedulingError = 16;
const NBAPCause measurementTemporarilyNotAvailable_NBAP = 17;
const NBAPCause invalidCmSetting = 18;
const NBAPCause reconfigurationCfnNotElapsed_NBAP = 19;
const NBAPCause numberOfDlCodesNotSupported_NBAP = 20;
const NBAPCause scpichNotSupported = 21;
const NBAPCause combiningNotSupported_NBAP = 22;
const NBAPCause ulSfNotSupported = 23;

```



```

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

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

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

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

```

```
const NBAPCause unspecified_Misc_NBAP = 61;
```

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

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

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

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

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

```
typedef unsigned short FailureCause;
const FailureCause configurationUnsupported = 1;
const FailureCause physicalChannelFailure_Failure = 2;
const FailureCause incompatibleSimultaneousReconfiguration = 3;
const FailureCause protocolError_Failure = 4;
const FailureCause compressedModeRuntimeError = 5;
const FailureCause cellUpdateOccurred = 6;
```

```

const FailureCause invalidConfiguration = 7;
const FailureCause configurationIncomplete = 8;
const FailureCause unsupportedMeasurement = 9;

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

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

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

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

//The following call failure causes are used in the category "mobileTrafficFlow".
typedef unsigned short CallFailureCause;
const CallFailureCause callingPartAuthFail = 1;
const CallFailureCause callingPartCipherModeFail = 2;
const CallFailureCause interfaceABusy = 3;
const CallFailureCause callingPartAssignFail = 4;

```



```

const CallFailureCause exchangeCongestion = 5;
const CallFailureCause userEarlyRelease = 6;
const CallFailureCause calledPartAssignFail = 7;
const CallFailureCause calledPartDetermineBusy = 8;
const CallFailureCause userUnreachable = 9;
const CallFailureCause alertingEarlyRelease = 10;
const CallFailureCause outCircuitOverflow = 11;
const CallFailureCause calledPartBusy = 12;
const CallFailureCause noAnswer = 13;

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

```



```

const ActPdpContextMsFailureCause operatorDeterminedBarring_Ms = 8;
const ActPdpContextMsFailureCause llcOrSndcpFailure= 25;
const ActPdpContextMsFailureCause insufficientResources = 26;
const ActPdpContextMsFailureCause unknownOrMissingAccessPointName = 27;
const ActPdpContextMsFailureCause unknownPdpAddressOrPdpType_Ms = 28;
const ActPdpContextMsFailureCause userAuthenticationFailed_Ms = 29;
const ActPdpContextMsFailureCause activationRejectedByGgsn = 30;
const ActPdpContextMsFailureCause activationRejected = 31; //unspecified
const ActPdpContextMsFailureCause serviceOptionNotSupported_Ms = 32;
const ActPdpContextMsFailureCause requestedServiceOptionNotSubscribed_Ms = 33;//redefined
const ActPdpContextMsFailureCause serviceOptionTemporarilyOutOfOrder_Ms = 34;//redefined
const ActPdpContextMsFailureCause nsapiAlreadyUsed = 35;
const ActPdpContextMsFailureCause regularPdpContextDeactivation = 36;
const ActPdpContextMsFailureCause qosNotAccepted = 37;
const ActPdpContextMsFailureCause networkFailure_Ms = 38;
const ActPdpContextMsFailureCause reactivationRequested = 39;
const ActPdpContextMsFailureCause featureNotSupported = 40;
const ActPdpContextMsFailureCause semanticErrorInTheTftOperation_Ms = 41;
const ActPdpContextMsFailureCause syntacticalErrorInTheTftOperation = 42;
const ActPdpContextMsFailureCause unknownPdpContext = 43;
const ActPdpContextMsFailureCause semanticErrorsInPacketFilters_Ms= 44;
const ActPdpContextMsFailureCause syntacticalErrorInPacketFilters= 45;
const ActPdpContextMsFailureCause PdpContextWithoutTftAlreadyActivated_Ms = 46;
const ActPdpContextMsFailureCause InvalidTransactionIdentifierValue = 81;
const ActPdpContextMsFailureCause semanticallyIncorrectMessage_Ms = 95;
//const ActPdpContextMsFailureCause invalidMandatoryInformation_Ms = 96;//redefined
const ActPdpContextMsFailureCause messageTypeNon_existentOrNotImplemented_Ms = 97;
const ActPdpContextMsFailureCause messageTypeNotCompatibleWithTheProtocolState_Ms = 98;
const ActPdpContextMsFailureCause informationElementNon_existentOrNotImplemented_Ms = 99;
const ActPdpContextMsFailureCause conditionalIeError_Ms = 100;
const ActPdpContextMsFailureCause messageNotCompatibleWithTheProtocolState_Ms = 101;
const ActPdpContextMsFailureCause protocolError_Ms = 111; // unspecified

```

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

```

typedef unsigned short ActPdpContextUtmsFailureCause;
const ActPdpContextUtmsFailureCause non_existent = 192;
const ActPdpContextUtmsFailureCause invalidMessageFormat = 193;
const ActPdpContextUtmsFailureCause imsiNotKnown = 194;
const ActPdpContextUtmsFailureCause msIsGprsDetached = 195;
const ActPdpContextUtmsFailureCause msIsNotGprsResponding = 196;
const ActPdpContextUtmsFailureCause msRefuses = 197;
const ActPdpContextUtmsFailureCause versionNotSupported = 198;

```

```

const ActPdpContextUtmsFailureCause noResourcesAvailable = 199;
const ActPdpContextUtmsFailureCause serviceNotSupported = 200;
const ActPdpContextUtmsFailureCause mandatoryIeIncorrect = 201;
const ActPdpContextUtmsFailureCause mandatoryIeMissing = 202;
const ActPdpContextUtmsFailureCause optionalIeIncorrect = 203;
const ActPdpContextUtmsFailureCause systemFailure = 204;
const ActPdpContextUtmsFailureCause roamingRestriction = 205;
const ActPdpContextUtmsFailureCause p_tmsiSignatureMismatch = 206;
const ActPdpContextUtmsFailureCause gprsConnectionSuspended = 207;
const ActPdpContextUtmsFailureCause authenticationFailure = 208;
const ActPdpContextUtmsFailureCause userAuthenticationFailed_Utms = 209;
const ActPdpContextUtmsFailureCause contextNotFound = 210;
const ActPdpContextUtmsFailureCause allDynamicPdpAddressesAreOccupied = 211;
const ActPdpContextUtmsFailureCause noMemoryIsAvailable = 212;
const ActPdpContextUtmsFailureCause relocationFailure = 213;
const ActPdpContextUtmsFailureCause unknownMandatoryExtensionHeader = 214;
const ActPdpContextUtmsFailureCause semanticErrorInTheTftOperation_Utms = 215;
const ActPdpContextUtmsFailureCause syntacticErrorInTheTftOperation = 216;
const ActPdpContextUtmsFailureCause semanticErrorsInPacketFilters_Utms = 217;
const ActPdpContextUtmsFailureCause syntacticErrorsInPacketFilters = 218 ;
const ActPdpContextUtmsFailureCause missingOrUnknownApn = 219;
const ActPdpContextUtmsFailureCause unknownPdpAddressOrPdpType_Utms = 220;
const ActPdpContextUtmsFailureCause pdpContextWithoutTftAlreadyActivated_Utms = 221;
const ActPdpContextUtmsFailureCause apnAccessDenied_noSubscription = 222;
//value range 223-240 is for future use;
//value range 241-255 is reserved for GPRS charging protocol use;
const ActPdpContextUtmsFailureCause requestRelatedToPossiblyDuplicatedPacketsAlreadyFulfilled =
252;
const ActPdpContextUtmsFailureCause requestAlreadyFulfilled = 253;
const ActPdpContextUtmsFailureCause sequenceNumbersOfReleasedOrCancelledPacketsIeIncorrect =
254;
const ActPdpContextUtmsFailureCause requestNotFulfilled = 255;

//The following GPRS attach failure causes are defined in the section 10.5.5.14 of 3GPP TS 24.008
v6.1.0.
typedef unsigned short gprsAttathFailureCause;
const gprsAttathFailureCause imsiUnknownInHLR_Gprs = 2;
const gprsAttathFailureCause illegalMS_Gprs = 3;
const gprsAttathFailureCause illegalME_Gprs = 6;
const gprsAttathFailureCause gprsServicesNotAllowed = 7;
const gprsAttathFailureCause gprsServicesAndNon_GprsServicesNotAllowed = 8;
const gprsAttathFailureCause msIdentityCannotBeDerivedByTheNetwork = 9;
const gprsAttathFailureCause implicitlyDetached = 10;

```



```

const gprsAttathFailureCause plmnNotAllowed_Gprs = 11;
const gprsAttathFailureCause locationAreaNotAllowed_Gprs = 12;
const gprsAttathFailureCause roamingNotAllowedInThisLocationArea_Gprs = 13;
const gprsAttathFailureCause noSuitableCellsInLocationArea_Gprs = 15;
const gprsAttathFailureCause networkFailure_Gprs = 17;
const gprsAttathFailureCause macFailure_Gprs = 20;
const gprsAttathFailureCause synchFailure_Gprs = 21;
const gprsAttathFailureCause congestion_Gprs = 22;
const gprsAttathFailureCause gsmAuthenticationUnacceptable_Gprs = 23;
const gprsAttathFailureCause NoPdpContextActivated = 40;
const gprsAttathFailureCause failRetryUponEntryIntoANewCell_Gprs = 48;
//value range 48 - 63 is used to retry upon entry into a new cell;
const gprsAttathFailureCause semanticallyIncorrectMessage_Gprs = 95;
const gprsAttathFailureCause invalidMandatoryInformation_Gprs = 96;
const gprsAttathFailureCause messageTypeNon_existentOrNotImplemented_Gprs = 97;
const gprsAttathFailureCause messageTypeNotCompatibleWithTheProtocolState_Gprs = 98;
const gprsAttathFailureCause informationElementNon_existentOrNotImplemented_Gprs = 99;
const gprsAttathFailureCause conditionalError_Gprs = 100;
const gprsAttathFailureCause messageNotCompatibleWithTheProtocolState_Gprs = 101;
const gprsAttathFailureCause protocolError_Gprs = 111; // unspecified

// The following originating and terminating SMS failure causes are defined in the section 8.2.5.4 of
3GPP TS 24.011 v5.2.0.

typedef unsigned short smsFailureCause;
// Cause values in a mobile originating SM_transfer attempt failure
const smsFailureCause unassignedOrUnallocatedNumber = 1;
const smsFailureCause operatorDeterminedBarring_Sms = 8;
const smsFailureCause callBarred = 10;
const smsFailureCause reserved = 11;
const smsFailureCause shortMessageTransferRejected = 21;
const smsFailureCause destinationOutOfOrder = 27;
const smsFailureCause unidentifiedSubscriber = 28;
const smsFailureCause facilityRejected = 29;
const smsFailureCause unknownSubscriber = 30;
const smsFailureCause networkOutOfOrder = 38;
const smsFailureCause temporaryFailure = 41;
const smsFailureCause congestion_Sms = 42;
const smsFailureCause resourcesUnavailable = 47; //unspecified
const smsFailureCause requestedFacilityNotSubscribed = 50;
const smsFailureCause requestedFacilityNotImplemented = 69;
const smsFailureCause invalidShortMessageTransferReferenceValue = 81;

```

```

const smsFailureCause semanticallyIncorrectMessage_Sms = 95;
const smsFailureCause invalidMandatoryInformation_Sms = 96;
const smsFailureCause messageTypeNon_existentOrNotImplemented_Sms = 97;
const smsFailureCause messageNotCompatibleWithShortMessageProtocolState = 98;
const smsFailureCause informationElementNon_existentOrNotImplemented_Sms = 99;
const smsFailureCause protocolError_Sms = 111; //unspecified
const smsFailureCause interworking = 127; //unspecified
// Cause values in a mobile terminating SM_transfer attempt failure
const smsFailureCause memoryCapacityExceeded = 22;
//const smsFailureCause invalidShortMessageTransferReferenceValue = 81; //redefined
//const smsFailureCause semanticallyIncorrectMessage_Sms = 95; //redefined
//const smsFailureCause invalidMandatoryInformation_Sms = 96; //redefined
//const smsFailureCause messageTypeNon_existentOrNotImplemented_Sms = 97; //redefined
//const smsFailureCause messageNotCompatibleWithShortMessageProtocolState = 98; //redefined
//const smsFailureCause informationElementNon_existentOrNotImplemented_Sms = 99; //redefined
//const smsFailureCause protocolError_Sms = 111; //unspecified //redefined

typedef unsigned short CauseType;
const CauseType sum = 0;
const CauseType other = 65535;
const CauseType noResponse = 65534;
struct CausePairType
{
    CauseType cause;
    unsigned long value;
};
typedef sequence<CausePairType> TDSCDMAMeasurementType3;

typedef unsigned short TrafficClass;
const TrafficClass conversational = 1;
const TrafficClass streaming = 2;
const TrafficClass interactive = 3;
const TrafficClass background = 4;
struct ClassPairType
{
    TrafficClass class;
    unsigned long value;
};
typedef sequence<ClassPairType> TDSCDMAMeasurementType4;

```



```
typedef string LocationAreaIdentificationType;
//LocationAreaIdentificationType is composed of MCC, MNC and LAC;
struct LocationAreaMeasurementType
{
    LocationAreaIdentificationType LocationAreaIdentification;
    unsigned long value;
};
typedef sequence<LocationAreaMeasurementType> TDSCDMAMeasurementType5;

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

#endif
```

9 TD-SCDMA 性能测量数据接口功能相关的文件

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

下面Schema文件中用到的字段说明参见附录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
xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:mc="http://latest/nmc-omc/cmNrm.doc#measCollec"
targetNamespace="http://latest/nmc-omc/cmNrm.doc#measCollec" elementFormDefault="qualified">
<!-- 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"/>
        </complexType>
      </element>
      <attribute name="userLabel" type="string" use="optional"/>
      <attribute name="swVersion" type="string" use="optional"/>
    </sequence>
  </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>
    </sequence>
  </complexType>
</element>

```

```

</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>
<!--<attribute name="measInfoId" type="string" use="optional"/>-->

```

```

        </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.|mscBasicMeasurement.failLocationUpdatesIntr
aMscPerCause.|mscBasicMeasurement.failLocationUpdatesInterMscPerCause.|mscBasicMeasurement.fail
OrigSmsCsPerCause.|mscBasicMeasurement.failTermSmsCsPerCause.|mscBasicMeasurement.attExternal
HostPerCause.|mscBasicMeasurement.attPageReqsPerLa.|mscBasicMeasurement.succPageReqsPerLa.|ms
cBasicMeasurement.attRepageReqsPerLa.|mobileTrafficFlow.failOrigCallsGsmPerCause.|mobileTrafficFl
ow.failOrigCallsUmtsPerCaus
e.|mobileTrafficFlow.failInternalCallsPerCause.|mobileTrafficFlow.failTermCallsGsmPerCause.|mobileTr
afficFlow.failTermCallsUmtsPerCause.|mobileTrafficFlow.failIncCallsPerCause.|mobileTrafficFlow.failO
utCallsPerCause.|mobileTrafficFlow.failTransCallsPerCause.|mobileTrafficFlow.failOrigOutCallsPerCaus
e.|mobileTrafficFlow.failTermIncCallsPerCause.|vlrSubscriberData.nbrCurrentSubsInVlrPerHlr.|hlrSubscr
iberData.nbrCurrentSubsWithPowerOnInHlrPerVlr.|sessionManagementMeasurement.failActPdpContext
MsPerCause.|sessionManagementMeasurement.failActPdcContextnetworkPerCause.|subscriberManagem
entMeasurement.meanAttachedSubs
PcrRa.|subscriberManagementMeasurement.maxAttachedSubsPerRa.|mobileManagementMeasurement.failGp
rsAttachPerCause.|mobileManagementmeasurement.failCombiAttachPerCause.|mobileManagementMeas
urement.failGprsAttachWithImsiAttachedPerCause.|mobileManagementMeasurement.failIntraSgsnRaUpd
atePerCause.|mobileManagementMeasurement.failCombiIntraSgsnRaUpdatePerCause.|mobileManagem
entMeasurement.failInterSgsnRaUpdatePerCause.|mobileManagementMeasurement.failCombiInterSgsnRa
UpdatePerCause.|apnSessionManagementMeasurement.failActPdpContextUmtsPerCause.|
RAB.AttEstabCs.|RAB.SuccEstabCs.|RAB.FailEstabCs.|RAB.AttEstabPs.|RAB.SuccEstabPs.|RAB.FailE
stabPs.|RAB.RelReqCsPerCause.|RAB.RelReqPsPerCause.|RAB.RelReqCsPerTraffic.|RAB.RelReqPsPer

```


Traffic.|IU.AttConnRelReqUtranCs.|IU.AttConnRelReqUtranPs.|IU.AttConnRelCnCs.|IU.AttConnRelCnPs.
 Ps.|IU.NbrRabCsRelIuConnPerCause.|IU.NbrRabPsRelIuConnPerCause.|IU.NbrRabCsRelIuConnPerTraf-
 fic.|IU.NbrRabPsRelIuConnPerTraffic.|IUEXCEP.NbrResetCsByRnc.|IUEXCEP.NbrResetPsByRnc.|IUE-
 XCEP.NbrResetCsByCn.|IUEXCEP.NbrResetPsByCn.|IUEXCEP.NbrResetResCsByRnc.|IUEXCEP.NbrR-
 esetResPsByRnc.|IUEXCEP.NbrResetResCsByCn.|IUEXCEP.NbrResetResPsByCn.|IUEXCEP.NbrErrorI-
 ndCsByRnc.|IUEXCEP.NbrErrorIndPsByRnc.|IUEXCEP.NbrErrorIndCsByCn.|IUEXCEP.NbrErrorIndPs-
 ByCn.|BHO.FailIntraFreqIntraRnc.|BHO.FailInterFreqIntraRnc.|HHO.FailIntraFreqIntraRnc.|HHO.FailInt-
 erFreqIntraRnc.|RELOC.AttOutPrepUeInvolCs.|RELOC.FailOutPrepUeInvolCs.|RELOC.FailRelocOutWi-
 thUeInvolCs.|RELOC.AttOutPrepWithUeInvolPs.|RELOC.FailOutPrepWithUeInvolPs.|RELOC.FailOut-
 WithUeInvolPs.|RELOC.AttInUeInvolCs.|RELOC.FailInUeInvolCs.|RELOC.AttInUeInvolPs.|RELOC.Fa-
 ilInUeInvolPs.|IRATHO.AttRelocPrepOutCs.|IRATHO.FailRelocPrepOutCs.|IRATHO.FailOutCs.|IRATH-
 O.AttIncCs.|IRATHO.FailIncCs.|IRATHO.FailOutPsUtran.|IUCSOCT.UpOutCs.|IUCSOCT.UpInCs.|IUP-
 SOCT.UpOutPs.|IUPSOCT.UpInPs.|RLC.CsTraffic.|RLC.PsUIOct.|RLC.PsDIOct.|RAB.AttRabAssnEstab-
 Cs.|RAB.SuccRabAssnEstabCs.|RAB.FailRabAssnEstabCs.|RAB.AttRabAssnEstabPs.|RAB.SuccRabAss-
 nEstabPs.|RAB.FailRabAssnEstabPs.|RAB.RelReqCs.|RAB.RelReqPs.|RAB.NbrRncRelCsRab.|RAB.Nbr-
 RncRelPsRab.|RAB.NbrRabCsRelIuConnPerCause.|RAB.NbrRabPsRelIuConnPerCause.|RAB.NbrRabCs-
 RelIuConnPerTraffic.|RAB.NbrRabPsRelIuConnPerTraffic.|RRC.AttConnEstab.|RRC.SuccConnEstab.|R-
 RC.FailConnEstab.|DCA.FailIntraCellIntraFreq.|DCA.FailIntraCellInterFreq.|HHO.FailOutIntraRncIntraF-
 req.|HHO.FailOutIntraRncInterFreq.|HHO.FailInIntraRncIntraFreq.|HHO.FailInIntraRncInterFreq.|BHO.F-
 ailOutIntraRncIntraFreq.|BHO.FailOutIntraRncInterFreq.|BHO.FailInIntraRncIntraFreq.|BHO.FailInIntra-
 RncInterFreq.|HHO.FailOutInterRncCnIntraFreq.|HHO.FailOutInterRncCnInterFreq.|IRATHO.AttIncCs.|I-
 RATHO.FailIncCs.|RLM.FailRISetupIubUtranSide.|RLM.FailRIAddIubUtranSide.|RLC.TrafficUI.|RLC.T-
 rafficDI.|RLC.PsUIOct.|RLC.PsDIOct.|HHO.FailOutInterCellIntraFreq.|HHO.FailOutInterCellInterFreq.|B-
 HO.FailOutIntraRncIntraFreq.|BHO.FailOutIntraRncInterFreq.|IRATHO.AttRelocPrepOutCs.|IRATHO.Fa-
 ilRelocPrepOutCs.|IRATHO.FailOutCs.|IRATHO.FailOutPsUtran.|HSPA.E.SuccRabSetup.|HSPA.E.FailRa-
 bSetup.|HSPA.E.NbrRabRelCn.|HSPA.E.AttRabEstabPS.|HSPA.E.SuccRabEstabPS.|CARR.TddMeanTcp.|
 CARR.TddMaxTcp.|CARR.TddMeanRtwp.|CARR.TddMaxRtwp.|CR.CodeUpResUsed.|CR.CodeDownR-
 esUsed.|CR.NbrAssnBruUI.|CR.NbrAssnBruDI.)([a-zA-Z0-9.<>_]{1,63})"/>

</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.nbrImsiSetachs"/>

<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.attTermSmsCs"/>
<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.succOutCalls"/>
<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.succLucIntraVlr"/>
<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.nbrAlertServiceCenter"/>
<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.succGetRoutingInfo"/>
<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.masReadySubs"/>
<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.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="shortMeassageServiceMeasurement.attOrigSmsPs"/>
<enumeration value="shortMeassageServiceMeasurement.succOrigSmsPs"/>
<enumeration value="shortMeassageServiceMeasurement.attTermSmsPs"/>
<enumeration value="shortMeassageServiceMeasurement.succTermSmsPs"/>
<enumeration value="shortMeassageServiceMeasurement.attSmsMsPresentPs"/>
<enumeration value="shortMeassageServiceMeasurement.succSmsMsPresentPs"/>

```



```

<enumeration value="shortMeassageServiceMeasurement.attSmsMemoryAvailPs"/>
<enumeration value="shortMeassageServiceMeasurement.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="IU.AttRncEstabCsIuConn"/>
<enumeration value="IU.AttRncEstabPsIuConn"/>
<enumeration value="IUEXCEP.NbrOverloadCtrlCsByRnc"/>
<enumeration value="IUEXCEP.NbrOverloadCtrlPsByRnc"/>
<enumeration value="IUEXCEP.NbrOverloadCtrlCsByCn"/>

```

<enumeration value="IUEXCEP.NbrOverloadCtrlPsByCn"/>
<enumeration value="BHO.AttIntraFreqIntraRnc"/>
<enumeration value="BHO.AttInterFreqIntraRnc"/>
<enumeration value="HHO.AttIntraFreqIntraRnc"/>
<enumeration value="HHO.AttInterFreqIntraRnc"/>
<enumeration value="HHO.AttOutInterRnc"/>
<enumeration value="HHO.SuccOutInterRnc"/>
<enumeration value="HHO.AttInInterRnc"/>
<enumeration value="HHO.SuccInInterRnc"/>
<enumeration value="RELOC.AttOutWithUeInvolCs"/>
<enumeration value="RELOC.SuccOutWithUeInvolCs"/>
<enumeration value="RELOC.AttOutWithUeInvolPs"/>
<enumeration value="RELOC.SuccOutWithUeInvolPs"/>
<enumeration value="IRATHO.AttOutCs"/>
<enumeration value="IRATHO.AttOutPsUtran"/>
<enumeration value="IRATHO.AttRelocInInterSysPs"/>
<enumeration value="IRATHO.SuccRelocInInterSysPs"/>
<enumeration value="IUCSOCT.RanapOutCs"/>
<enumeration value="IUCSOCT.RanapInCs"/>
<enumeration value="ATM.OutOctIucs"/>
<enumeration value="ATM.InOctIucs"/>
<enumeration value="IUPSOCT.RanapOutPs"/>
<enumeration value="IUPSOCT.RanapInPs"/>
<enumeration value="ATM.OutOctIups"/>
<enumeration value="ATM.InOctIups"/>
<enumeration value="IUBOCT.NbapIn"/>
<enumeration value="IUBOCT.NbapOut"/>
<enumeration value="IUBOCT.UpFpIn"/>
<enumeration value="IUBOCT.UpFpOut"/>
<enumeration value="ATM.OctInIub"/>
<enumeration value="ATM.OctOutIub"/>
<enumeration value="RRC.MeanConn"/>
<enumeration value="RRC.MaxConn"/>
<enumeration value="HHO.AttInInterRnc"/>
<enumeration value="HHO.SuccInInterRnc"/>
<enumeration value="HHO.AttOutInterRnc"/>
<enumeration value="HHO.SuccOutInterRnc"/>
<enumeration value="DCA.AttIntraCellIntraFreq"/>
<enumeration value="DCA.AttIntraCellInterFreq"/>
<enumeration value="HHO.AttOutIntraRncIntraFreq"/>
<enumeration value="HHO.AttOutIntraRncInterFreq"/>
<enumeration value="HHO.AttInIntraRncIntraFreq"/>
<enumeration value="HHO.AttInIntraRncInterFreq"/>


```

<enumeration value="BHO.AttOutIntraRncIntraFreq"/>
<enumeration value="BHO.AttOutIntraRncInterFreq"/>
<enumeration value="BHO.AttInIntraRncIntraFreq"/>
<enumeration value="BHO.AttInIntraRncInterFreq"/>
<enumeration value="HHO.AttOutInterRncCnIntraFreq"/>
<enumeration value="HHO.AttOutInterRncCnInterFreq"/>
<enumeration value="IRATHO.AttRelocInInterSysPs"/>
<enumeration value="IRATHO.SuccRelocInInterSysPs"/>
<enumeration value="RLM.AttRISetupIubUtranSide"/>
<enumeration value="RLM.AttRIAddIubUtranSide"/>
<enumeration value="RLM.AttRIDelIubUtranSide"/>
<enumeration value="RLM.SuccRIDelIubUtranSide"/>
<enumeration value="PAGING.AttInReCellPagingType1"/>
<enumeration value="PAGING.SuccCellPagingType1"/>
<enumeration value="PAGING.AttCellPagingType2"/>
<enumeration value="PAGING.FailTranCellPagingType1"/>
<enumeration value="PAGING.CongCellPagingType1"/>
<enumeration value="HSPA.E.NbrMimoSingleStr"/>
<enumeration value="HSPA.E.NbrMimoDoubleStr"/>
<enumeration value="HSPA.E.NbrQam64Scheduled"/>
<enumeration value="HSPA.E.NbrQam16Scheduled"/>
<enumeration value="HSPA.E.NbrCpcDrxUser"/>
<enumeration value="HSPA.E.AttEfachEstab"/>
<enumeration value="HSPA.E.SuccEfachEstab"/>
<enumeration value="HSPA.E.EfachMaxDelay"/>
<enumeration value="HSPA.E.EfachMeanDelay"/>
<enumeration value="HSPA.E.NbrEfachMaxUser"/>
<enumeration value="HSPA.E.NbrEfachMeanUser"/>
<enumeration value="HSPA.E.NbrEpchMaxUser"/>
<enumeration value="HSPA.E.NbrEpchMeanUser"/>
<enumeration value="HSPA.E.NbrOctEfachIubPdu"/>
<enumeration value="HSPA.E.NbrRabRelUsrInact"/>
<enumeration value="HHO.AttOutInterCellIntraFreq"/>
<enumeration value="HHO.AttOutInterCellInterFreq"/>
<enumeration value="BHO.AttOutIntraRncIntraFreq"/>
<enumeration value="BHO.AttOutIntraRncInterFreq"/>
<enumeration value="IRATHO.AttOutCs"/>
<enumeration value="IRATHO.AttOutPsUtran"/>
  </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>
```

9.2 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)

XML Schema 文档补充说明

A.1 XML Schema文档结构标记

XML Schema文档结构标记约定如图A.1所示。

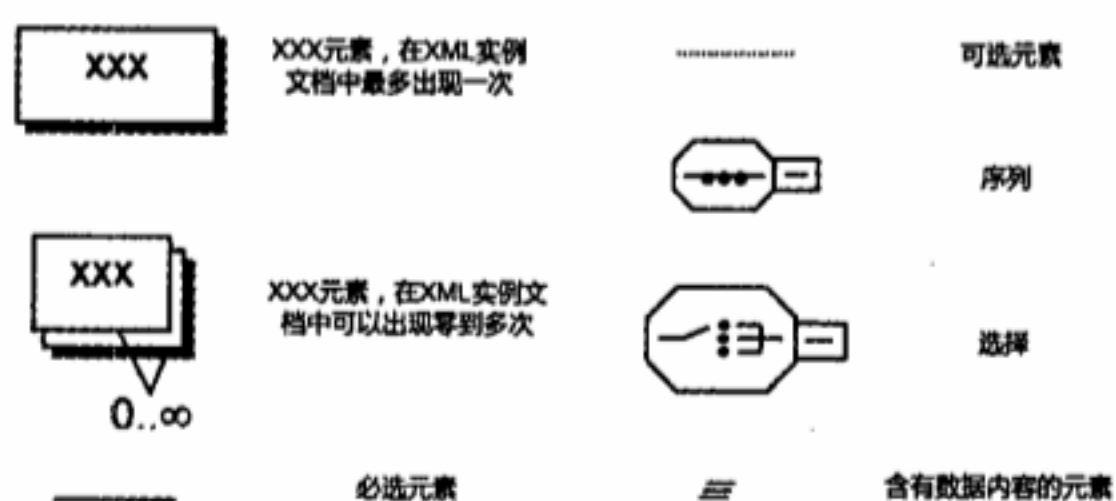


图 A.1 XML Schema 文档结构标记

A.2 性能测量数据文件的Schema定义

XML Schema文档<HspaeMeasCollec.xsd>结构如图A.2所示。

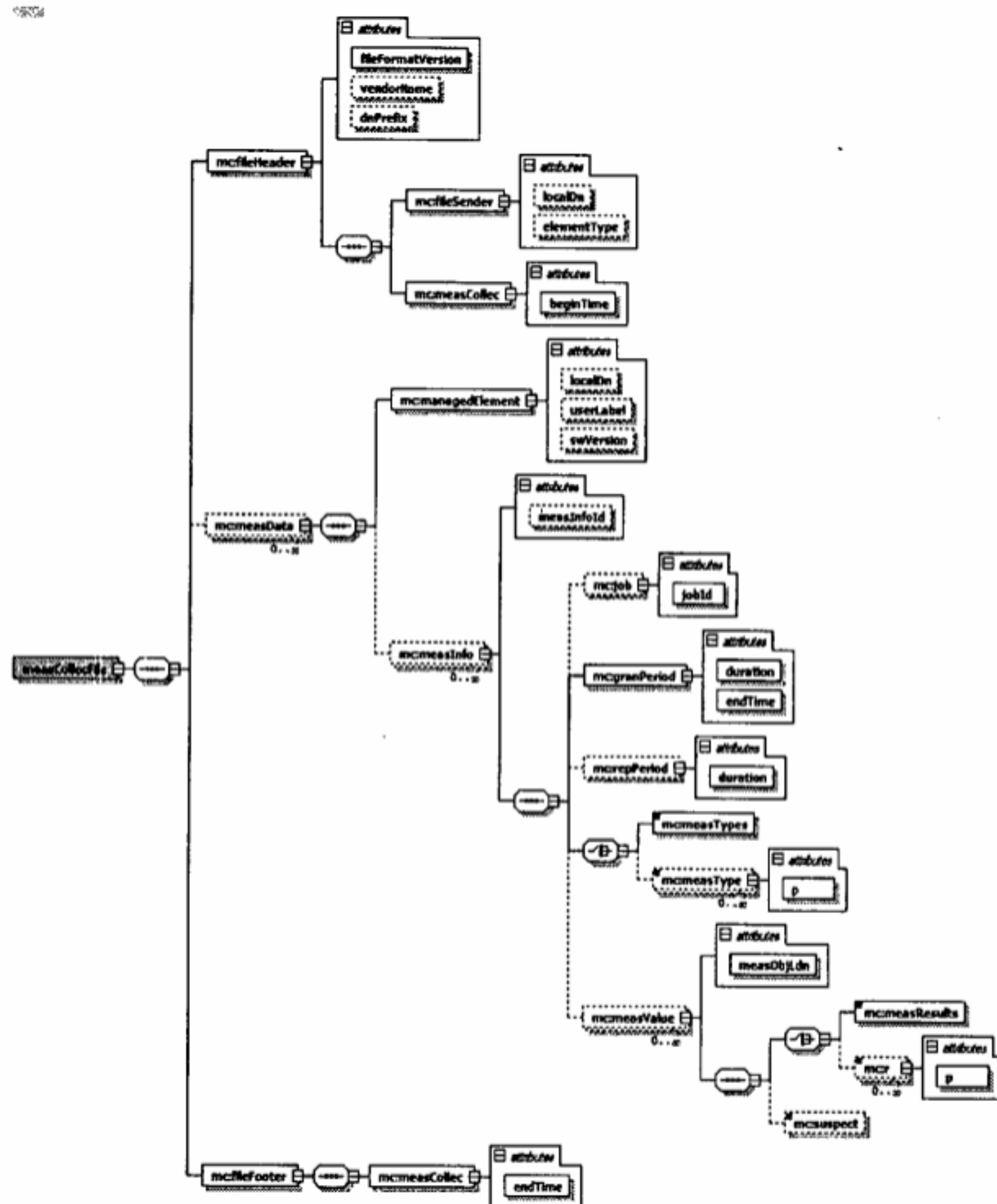


图 A.2 HspaeMeasCollec.xsd 文档结构

XML Schema文档元素/属性说明如表A.1所示。

表 A.1 HspaeMeasCollec.xsd 文档元素/属性说明

元素/属性名称		元素/属性描述
元素名称	包含属性	
measCollecFile		性能数据采集文件。是该Schema的根元素。由三个子元素组成：文件头部(fileHeader)、采集数据(measData)以及文件尾部(FileFooter)
fileHeader	fileFormatVersion	文件头部。由两个子元素组成：文件发送方(fileSender)、测量采集开始时间(measCollec)。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)
	vendorName	
	dnPrefix	
measData		性能测量数据。在一份采集上报文件中可出现零(未采集到数据)至多次。由两个子元素组成：管理网元(managedElement)及其性能采集结果(measInfo)
fileFooter		文件尾部。包含子元素：测量采集结束时间(measCollec)
fileSender	localDn	文件发送发。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	measInfoId	测量信息。由四个子元素组成：测量任务(job)、测量粒度周期(granPeriod)、测量上报周期(repPeriod)、测量类型(measType/measTypes)和测量值(measValue)。包含一个可选属性：测量信息标识符(measInfoId)
job	jobId	测量任务。该元素为可选元素。由其属性jobId唯一标识
granPeriod	duration	测量粒度周期。包含两个属性：持续时间(duration)、结束时间(endTime)
	endTime	
repPeriod	duration	测量上报周期。该元素为可选元素。包含唯一属性：持续时间(duration)
measTypes/measType		采集类型。均由measName扩展而来。在XML文件实例中，两个元素择一使用。不同的是measTypes是以列表方式呈现，且只出现一次；measType可出现多次，由属性值为非负数的p加以区分
measType	p	p为属性限定(position)。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p加以区分。r的p属性应与measType的p属性一一对应
r	p	p为属性限定。表示对<measType p>的一个采集结果应答。<r p>需和<measType p>一一对应
measValue	measObjLdn	采集值。由两个子元素组成：采集结果列表(measResults/r)和一个标记采集数据是否可信的标志位(suspect)。本身还包含一个属性：测量对象本地识别名(measObjLdn)
suspect		用于标记采集值是否可信。默认值为False(即可信)
measCollec	beginTime	性能采集开始时间
	endTime	性能采集结束时间
measName		性能测量项名称。分为包含子测量项(measNameWithSubCounter)和不含子测量项(measNameWithoutSubCounter)两类。从3GPP TS 32.435中扩展而来
measNameWithSubCounter		含子测量项的数据测量项名称。表示为familyname.measurename.subcounter形式。从3GPP TS 32.435中扩展而来
measNameWithoutSubCounter		不含子测量项的数据测量项名称。表示为familyname.measurename形式。从3GPP TS 32.435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 性能测量数据XML文件示例一：不使用可选的p属性

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="HspaeMeasCollec.xsl"?>
<!-- XML schema based XML measurement report file without use of optional positioning attributes on
measurement types and results. All values are hypothetical but syntactically correct -->
<measCollecFile xmlns="std://yd-t/hspae/itf-n/mc/2007"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="std://yd-t/hspae/itf-n/mc/2007
HspaeMeasCollec.xsd">
  <fileHeader fileFormatVersion="1.0" vendorName="Company NN"
dnPrefix="DC=a1.companyNN.com,SubNetwork=1,IRPAgent=1">
    <fileSender
localDn="SubNetwork=CountryNN,MeContext=MEC-Gbg-1,ManagedElement=BSC-Gbg-1"
elementType="BSC" inDomain="UTRAN"/>
    <measCollec beginTime="2007-03-01T14:00:00+02:00"/>
  </fileHeader>
  <measData>
    <managedElement
localDn="SubNetwork=CountryNN,MeContext=MEC-Gbg-1,ManagedElement=BSC-Gbg-1" userLabel="BSC
Telecomville"/>
    <measInfo>
      <job jobId="1231"/>
      <granPeriod duration="PT900S" endTime="2000-03-01T14:14:30+02:00"/>
      <repPeriod duration="PT1800S"/>
      <measTypes> mobileTrafficFlow.failOrigCallsPerCause.0
mobileTrafficFlow.failTermIncCallsPerCause.1 mobileTrafficFlow.failTermIncCallsPerCause.2
mobileTrafficFlow.failTermIncCallsPerCause.3 </measTypes>
      <measValue measObjLdn="BscFunction=RF-1,Sector=Gbg-997">
        <measResults>234 345 567 789</measResults>
      </measValue>
      <measValue measObjLdn="BscFunction=RF-1,Sector=Gbg-998">
        <measResults>890 901 123 234</measResults>
      </measValue>
      <measValue measObjLdn="BscFunction=RF-1,Sector=Gbg-999">
        <measResults>456 567 678 789</measResults>
        <suspect>true</suspect>
      </measValue>
    </measInfo>
  </measData>
```

```
<fileFooter>
    <measCollec endTime="2007-03-01T14:15:00+02:00"/>
</fileFooter>
</measCollecFile>
```

B.2 性能测量数据XML文件示例二：使用可选的p属性

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="HspaeMeasCollec.xsl"?>
<!-- XML schema based XML measurement report file with use of optional positioning attributes on measurement
types and results. All values are hypothetical but syntactically correct. -->
<measCollecFile xmlns="std://yd-t/hspae/itf-n/mc/2007"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="std://yd-t/hspae/itf-n/mc/2007
HspaeMeasCollec.xsd">
    <fileHeader fileFormatVersion="1.0" vendorName="Company NN"
dnPrefix="DC=a1.companyNN.com,SubNetwork=1,IRPAgent=1">
        <fileSender
localDn="SubNetwork=CountryNN,MeContext=MEC-Gbg-1,ManagedElement=BSC-Gbg-1"
elementType="BSC" inDomain="UTRAN"/>
            <measCollec beginTime="2007-03-01T14:00:00+02:00"/>
        </fileHeader>
        <measData>
            <managedElement
localDn="SubNetwork=CountryNN,MeContext=MEC-Gbg-1,ManagedElement=BSC-Gbg-1" userLabel="BSC
Telecomville"/>
                <measInfo>
                    <job jobId="1231"/>
                    <granPeriod duration="PT900S" endTime="2000-03-01T14:14:30+02:00"/>
                    <repPeriod duration="PT1800S"/>
                    <measType p="1">mobileTrafficFlow.failOrigCallsPerCause.0</measType>
                    <measType p="2">mobileTrafficFlow.failOrigCallsPerCause.1</measType>
                    <measType p="3">mobileTrafficFlow.failOrigCallsPerCause.2</measType>
                    <measType p="4">mobileTrafficFlow.failTermIncCallsPerCause.3</measType>
                    <measValue measObjLdn="BscFunction=RF-1,Sector=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="BscFunction=RF-1,Sector=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="BscFunction=RF-1,Sector=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="2007-03-01T14:15:00+02:00"/>
</fileFooter>
</measCollecFile>

```

B.3 性能测量数据XML文件示例三：使用可选的measInfoId属性

```

<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="HspaeMeasCollec.xsl"?>
<!-- XML schema based XML measurement report file with use of optional measInfoId attribute. All values are
hypothetical but syntactically correct. -->
<measCollecFile xmlns="std://yd-t/hspae/itf-n/mc/2007"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="std://yd-t/hspae/itf-n/mc/2007
HspaeMeasCollec.xsd">
  <fileHeader fileFormatVersion="1.0" vendorName="Company NN" dnPrefix="SubNetwork=1">
    <fileSender localDn="OMC_PS=10" elementType="Element Manager"/>
    <measCollec beginTime="2007-03-01T14:00:00+02:00"/>
  </fileHeader>
  <measData>
    <managedElement localDn="ManagedElement=PS_Core" userLabel="SGSN" swVersion="R30.1.5"
inDomain="UTRAN"/>
    <measInfo measInfoId="Category A">
      <job jobId="01"/>
      <granPeriod duration="PT900S" endTime="2007-03-01T14:15:00+02:00"/>
      <repPeriod duration="PT1800S"/>
      <measTypes>MM.AttGprsAttach MM.SuccGprsAttach MM.AbortedGprsAttach
MM.AttIntraSgsnRaUpdate</measTypes>
      <measValue measObjLdn="SgsnFunction=1">
        <measResults>10 20 30 40</measResults>
      </measValue>
    </measInfo>
    <measInfo measInfoId="Category B">
      <job jobId="02"/>
      <granPeriod duration="PT900S" endTime="2007-03-01T14:15:00+02:00"/>
    </measInfo>
  </measData>
</measCollecFile>

```



```
<repPeriod duration="PT1800S"/>
  <measTypes>MM.AttCombiAttach MM.SuccCombiAttach MM.
MM.AbortedCombiAttachMM.AttCombiDetachMs</measTypes>
  <measValue measObjLdn="SgsnFunction=2">
    <measResults>10 20 30 40</measResults>
  </measValue>
</measInfo>
<measInfo measInfoId="Category C">
  <job jobId="03"/>
  <granPeriod duration="PT1800S" endTime="2007-03-01T14:15:00+02:00"/>
  <repPeriod duration="PT900S"/>
  <measTypes>MM.AttPsPagingProcIu MM.SuccPsPagingProcIu</measTypes>
  <measValue measObjLdn="SgsnFunction=3">
    <measResults>25 25</measResults>
  </measValue>
</measInfo>
</measData>
<fileFooter>
  <measCollec endTime="2007-03-01T14:15:00+02:00"/>
</fileFooter>
</measCollecFile>
```

参 考 文 献

3GPP TS 32.432 V7.0.0 性能测量——文件格式定义
3GPP TS 32.435 V7.2.0 性能测量——XML文件格式定义

中 华 人 民 共 和 国
通 信 行 业 标 准

2GHz TD-SCDMA/WCDMA 数字蜂窝移动通信网
演进版高速分组接入 (HSPA+) 网络管理技术要求
第 2 部分: 基于 CORBA 技术的信息模型设计

YD/T 2333.2-2011

*

人民邮电出版社出版发行

北京市崇文区夕照寺街 14 号 A 座

邮政编码: 100061

宝隆元 (北京) 印刷技术有限公司印刷

版权所有 不得翻印

*

开本: 880 × 1230 1/16

2012 年 1 月第 1 版

印张: 7.5

2012 年 1 月北京第 1 次印刷

字数: 206 千字

ISBN 978 - 7 - 115 - 2517/ 12 - 95

定价: 75 元

本书如有印装质量问题, 请与本社联系 电话: (010)67114922