

YD

中华人民共和国通信行业标准

YD/T 1587.3-2007

2GHz cdma2000 数字蜂窝移动通信网

网络管理技术要求(第一阶段)

第3部分 基于CORBA技术的网络资源模型设计

Technical Specification for 2GHz cdma2000 Digital Cell Mobile
Communication Network Management (Phase I)

Part 3 CORBA Based Network Resource Model Design

2007-05-16 发布

2007-05-16 实施

中华人民共和国信息产业部 发布

目 次

前 言	II
1 范围	1
2 规范性引用文件	1
3 术语、定义和缩略语	1
3.1 缩略语	1
4 配置网络资源模型设计	1
4.1 通用配置资源模型的 IDL 定义	2
4.2 无线接入网网络资源模型的 IDL 定义	19
4.3 核心网网络资源模型的 IDL 定义	24
5 性能网络资源模型设计	36
5.1 性能参数的 IDL 定义	36
6 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<measCollec.xsd>	46
6.2 性能测量数据文件的 XML header 定义	57
附录 A (规范性附录) Schema 文档补充说明	58
附录 B (资料性附录) 性能管理功能相关 XML 文件示例	60

前 言

本部分是《2GHz 数字蜂窝移动通信网网络管理技术要求》系列标准之一。该系列标准的预计结构为：系列标准由四项标准组成，各项标准又分为多个部分标准。该系列标准和部分标准的名称如下：

1. YD/T1584.1-2007 2GHz 数字蜂窝移动通信网网络管理通用技术要求 第 1 部分 基本原则
2. YD/T1584.2-2007 2GHz 数字蜂窝移动通信网网络管理通用技术要求 第 2 部分 接口功能
3. YD/T1584.3-2007 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
4. YD/T1584.4-2007 2GHz 数字蜂窝移动通信网网络管理通用技术要求 第 4 部分 基于 CORBA 技术的管理接口设计
5. YD/T1586.1-2007 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
6. YD/T1586.2-2007 2GHz WCDMA 数字蜂窝移动通信网网络管理技术要求（第一阶段） 第 2 部分 性能网络资源模型
7. YD/T1586.3-2007 2GHz WCDMA 数字蜂窝移动通信网网络管理技术要求（第一阶段） 第 3 部分 基于 CORBA 技术的网络资源模型设计
8. YD/T1587.1-2007 2GHz cdma2000 数字蜂窝移动通信网网络管理技术要求（第一阶段） 第 1 部分 配置网络资源模型
9. YD/T1587.2-2007 2GHz cdma2000 数字蜂窝移动通信网网络管理技术要求（第一阶段） 第 2 部分 性能网络资源模型
10. YD/T1587.3-2007 2GHz cdma2000 数字蜂窝移动通信网网络管理技术要求（第一阶段） 第 3 部分 基于 CORBA 技术的网络资源模型设计
11. 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 配置网络资源模型
12. 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 性能网络资源模型
13. YD/T1585.3-2007 2GHz TD-SCDMA 数字蜂窝移动通信网网络管理技术要求（第二阶段） 第 3 部分 基于 CORBA 技术的网络资源模型设计

本部分参考第三代移动通信伙伴项目第二组（3GPP2）的以下标准：

3GPP2 S.S0028-A (Version 1.0) OAM&P for cdma2000 (3GPP R4 Delta Specification) (cdma2000 网络的操作、管理、维护和指配)

本部分与上述国际标准之间的主要差异为：

- 增强了 IDL 文件的注释说明。
- 增加了配置和性能网络资源模型新增部分内容的 IDL 定义
- 增强了配置和性能网络资源模型参数数据类型的定义
- 配置资源模型的 IDL 定义增加了一类文件 xxxNRMPProfile.idl，包括 GenericNRMPProfile.idl、

IMDataProfile.idl、RanNRMPProfile.idl 和 CoreNRMPProfile.idl。这类文件只是用来描述配置网络资源对象的属性名称及其数据类型的对应关系，实现时并不使用此类 idl 文件

—完善配置和性能网络资源模型 Schema 文件定义。

—对配置和性能网络资源模型的名称取值进行规范。

本部分与上述 3GPP2 标准的一致性程度为非等效。

本部分的附录 A 为规范性附录，附录 B 为资料性附录。

本部分由中国通信标准化协会提出并归口。

本部分起草单位：中国联合通信有限公司、中兴通讯股份有限公司、北京邮电大学

本部分主要起草人：高柏峰、李俊、王勇、芮兰兰、姚羿志、范小磊、文海龙、杨国、刘金龙、
腾伟

广东省网络空间安全协会受控资料

2GHz cdma2000 数字蜂窝移动通信网网络管理技术要求(第一阶段)

第 3 部分 基于 CORBA 技术的网络资源模型设计

1 范围

本部分规定了 2GHz 数字蜂窝移动通信网（以下简称 3G）网络管理接口中采用 cdma2000 1x 技术的网络资源模型的 IDL 定义。

本部分适用于采用 cdma2000 1x 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

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

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

YD/T 1587.1-2007 2GHz cdma2000 数字蜂窝移动通信网网络管理技术要求（第一阶段） 第 1 部分 配置网络资源模型

YD/T 1587.2-2007 2GHz cdma2000 数字蜂窝移动通信网网络管理技术要求（第一阶段） 第 2 部分 性能网络资源模型

3 术语、定义和缩略语

3.1 缩略语

下列缩略语适用于本部分。

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

4 配置网络资源模型设计

注：配置网络资源模型设计中有 3 类 IDL 文件，这 3 类文档及其用途如下：

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

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

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

4.1 通用配置资源模型的 IDL 定义

4.1.1 GenericNRMDefs

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

// #pragma prefix "3gppsa5.org"

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

module GenericNRMDefs
{
    //Definitions for abstract MO class Top

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

    //Definitions for MO class IRPAgent

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

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

    //Definitions for abstract MO class GenericIRP

    interface GenericIRP: Top
```

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

    // Attribute Names
    //
    const string iRPIId = "iRPIId";
};

//Definitions for MO class SubNetwork

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

    // Attribute Names
    //
    const string subNetworkId = "subNetworkId";
    const string dnPrefix = "dnPrefix";
    const string userLabel = "userLabel";
    const string setOfMcc = "setOfMcc";
    const string userDefinedNetworkType = "userDefinedNetworkType ";
};

//Definitions for MO class MeContext

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

    // Attribute Names
    //
    const string meContextId = "meContextId";
    const string dnPrefix = "dnPrefix";
};

//Definitions for MO class ManagementNode
```

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

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

//Definitions for MO class ManagedElement

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

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

//Definitions for abstract MO class ManagedFunction

interface ManagedFunction : Top
```



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

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

//Definitions for MO class VsDataContainer

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

    //Attribute Names
    //
    const string vsDataContainerId = "vsDataContainerId";
    const string vsDataType = "vsDataType";
    const string vsData = "vsData";
    const string vsDataFormatVersion = "vsDataFormatVersion";
};

//Defination for MO class InventoryUnit

interface InventoryUnit: 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";
};
```

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

```

```

};
#endif

```

4.1.2 GenericNRMPProfile

```

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

#include "GenericNRMSystem.idl"

#pragma prefix "3gppsa5.org"

```

```

/**
 * This module defines the attribute names and
 * correspondig attribute types for all defined
 * MO class. This module is used for reference.
 */

```

```

module GenericNRMPProfile
{
    interface Top
    {
        readonly attribute string objectClass;
        readonly attribute string objectInstance;
    };

    interface IRPAgent : Top
    {

```

```
readonly attribute GenericNRMSystem::ObjectIdType iRPAgentId;  
readonly attribute GenericNRMSystem::DN systemDN;
```

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

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

```
};
```

```
interface GenericIRP : Top
```

```
{
```

```
    readonly attribute string 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::NetworkType 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
        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;
        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 InventoryUnit : Top
{
    readonly attribute GenericNRMSSystem::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
};

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;
        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 IMDataProfile

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

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

module IMDataProfile
{
    interface InventoryUnit:GenericNRMProfile::Top
    {
        readonly attribute GenericNRMSystem::ObjectIdType inventoryUnitId;
        readonly attribute string inventoryUnitType ;
        readonly attribute string vendorUnitFamilyType;
        readonly attribute string vendorUnitTypeNumber;
        readonly attribute string vendorName;
        readonly attribute string serialNumber;
        readonly attribute string versionNumber;
        readonly attribute string dateOfManufacture;
        readonly attribute string dateOfLastService;
        readonly attribute wstring unitPosition;
        readonly attribute string manufacturerData;

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

#endif
```

4.2 无线接入网网络资源模型的 IDL 定义

4.2.1 RanNRMDefs

```
//File "RanNRMDefs.idl"
//The IRP document version number is "RAN NRM V1.0"
#ifndef RanNRMDefs_idl
#define RanNRMDefs_idl

#include "GenericNRMDefs.idl"

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

        // Attribute Names
        //
        const string bscFunctionId = "bscFunctionId";
        const string controlledBtsList = "controlledBtsList";
        const string relatedMsc= "relatedMsc";
        const string maxNumE1T1Port= "maxNumE1T1Port";
        const string numE1T1Port="numE1T1Port";
        const string numSignallingPort = "numSignallingPort";
        const string relatedPcfList = "relatedPcfList";
        const string maxNumStm1Port = "maxNumStm1Port";
        const string numStm1Port = "numStm1Port";
        const string maxNumVocoder="maxNumVocoder";
        const string mcc="mcc";
        const string mnc="mnc";
    };

    /**
     * Definitions for MO class BtsFunction
     */
    interface BtsFunction : GenericNRMDefs::ManagedFunction
    {
```

```

const string CLASS = "BtsFunction";

// Attribute Names
//
const string btsFunctionId = "btsFunctionId";
const string relatedBsc = "relatedBsc";
const string numFa = "numFa";
};

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

    // Attribute Names
    //
    const string sectorId = "sectorId";
    const string cellIdentifier = "cellIdentifier";
    const string sectorCapacity = "sectorCapacity";
    const string relatedBts = "relatedBts";
};

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

    // Attribute Names
    //
    const string pcfFunctionId = "pcfFunctionId";
    const string ipAddress = "ipAddress";
    const string relatedPdsnAddrList = "relatedPdsnAddrList";
    const string relatedBscList = "relatedBscList";
    const string throughputCapacity = "throughputCapacity";
    const string maxNumSupportedSession = "maxNumSupportedSession";
    const string maxNumSupportedActiveSession = "maxNumSupportedActiveSession";
};
};

```



```
#endif
```

4.2.2 RanNRMPProfile

```
//File "RanNRMPProfile.idl"
//The IRP document version number is "RAN NRM V1.0"
#ifndef RanNRMPProfile_idl
#define RanNRMPProfile_idl

#include "GenericNRMSystem.idl"
#include "GenericNRMPProfile.idl"
#include "RanNRMSystem.idl"
/**
 * This module defines the attribute names and
 * corresponding attribute types for all defined
 * MO class in Ran network. This module is
 * used for reference.
 */
module RanNRMPProfile
{
    interface BscFunction : GenericNRMPProfile::ManagedFunction
    {
        readonly attribute GenericNRMSystem::ObjectIdType bscFunctionId;
        attribute RanNRMSystem::ControlledBtsListType controlledBtsList;
        attribute RanNRMSystem::relatedMscType relatedMsc;
        readonly attribute unsigned long maxNumE1T1Port;
        readonly attribute unsigned long numE1T1Port;
        readonly attribute unsigned long numSignallingPort;
        attribute RanNRMSystem::relatedPcfListType relatedPcfList;
        readonly attribute unsigned long maxNumStm1Port;
        readonly attribute unsigned long numStm1Port;
        readonly attribute unsigned long maxNumVocoder;
        readonly attribute unsigned long mcc;
        readonly attribute unsigned long mnc;

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

```

```

    // notifyPotentialFaultyAlarmList
};

interface BtsFunction : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType btsFunctionId;
    attribute RanNRMSystem::relatedBscType relatedBsc;
    readonly attribute unsigned long numFa;

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

```

```

interface Sector:GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType sectorId;
    readonly attribute string cellIdentifier;
    readonly attribute float sectorCapacity;
    readonly attribute RanNRMSystem::relatedBtsType relatedBts;

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

```

```

interface PcfFunction: GenericNRMPProfile::ManagedFunction

```

```

{
    readonly attribute GenericNRMSystem::ObjectIdType pcFunctionId;
    readonly attribute RanNRMSystem::IPAddressType iAddress;
    readonly attribute RanNRMSystem:: IPAddressListType relatedPdsnAddrList;
        attribute RanNRMSystem:: relatedBscListType relatedBscList;
    readonly attribute unsigned long throughputCapacity;
    readonly attribute unsigned long maxNumSupportedSession;
    readonly attribute unsigned long maxNumSupportedActiveSession;

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

```

4.2.3 RanNRMSystem

```

//File "RanNRMSystem.idl"
//The IRP document version number is "RAN NRM V1.0"
#ifndef RanNRMSystem_idl
#define RanNRMSystem_idl

#include "GenericNRMSystem.idl"

module RanNRMSystem
{
    /**
     * This module adds datatype definitions for types
     * used in the Ran NRM which are not basic datatypes defined
     * already in CORBA and datatypes defined already in
     * GenericNRMSystem.
     */
    typedef GenericNRMSystem::DN ControlledBts;
    typedef sequence<ControlledBts> ControlledBtsListType;
    typedef GenericNRMSystem::DN relatedMscType;

```

```

typedef GenericNRMSystem::DN relatedPcf;
typedef sequence<relatedPcf> relatedPcfListType;
typedef GenericNRMSystem::DN relatedBscType;
typedef sequence<relatedBscType> relatedBscListType;
typedef GenericNRMSystem::DN relatedBtsType;
typedef string IPAddressType;
typedef sequence<IPAddressType> IPAddressListType;
};
#endif

```

4.3 核心网网络资源模型的 IDL 定义

4.3.1 CoreNRMDefs

```

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

#include "GenericNRMDefs.idl"

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

        // Attribute Names
        //
        const string mscFunctionId = "mscFunctionId";
        const string mscType = "mscType";
        const string maxNumUsers = "maxNumUsers";
        const string maxMscBhca = "maxMscBhca";
        const string controlledBscList = "controlledBscList";
        const string maxNumTldn = "maxNumTldn";
        const string maxNumE1T1Port = "maxNumE1T1Port";
        const string numE1T1Circuits = "numE1T1Circuits";
        const string maxNumStm1Port = "maxNumStm1Port";
    }
}

```

```

const string numStm1Port = "numStm1Port";
const string relatedIwf = "relatedIwf";
};

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

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

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

    // Attribute Names
    //
    const string acFunctionId = "acFunctionId";
    const string maxNumUsers = "maxNumUsers";
};

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

    // Attribute Names

```

```

//
const string eirFunctionId = "eirFunctionId";
const string eirNumber = "eirNumber";
const string maxNumImei = "maxNumImei";
};

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

    // Attribute Names
    //
    const string hlrFunctionId = "hlrFunctionId";
    const string maxNumUsers = "maxNumUsers";
    const string imsiRange = "imsiRange";
    const string maxNumMdn = "maxNumMdn";
    const string mdnRange = "mdnRange";
};

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

    // Attribute Names
    //
    const string vlrFunctionId = "vlrFunctionId";
    const string maxNumUsers = "maxNumUsers";
};

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

    // Attribute Names

```

```

//
const string iwFunctionId="iwFunctionId";
const string iwCapability="iwCapability";
};

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

    // Attribute Names
    //
    const string mcFunctionId="mcFunctionId";
};

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

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

```

```
#endif
```

4.3.2 CoreNRMProfile

```

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

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

```

```

/**
 * This module defines the attribute names and
 * correspondig attribute types for all defined
 * MO class in core network. This module is
 * used for reference.
 */
module CoreNRMPProfile
{
    interface MscFunction : GenericNRMPProfile::ManagedFunction
    {
        readonly attribute GenericNRMSystem::ObjectIdType mscFunctionId;
        readonly attribute CoreNRMSystem::MscTypeType mscType;
        readonly attribute unsigned long maxNumUsers;
        readonly attribute unsigned long maxMscBhca;
        attribute CoreNRMSystem:: controlledBscListType controlledBscList;
        readonly attribute unsigned long maxNumTldn;
        readonly attribute unsigned long maxNumE1T1Port;
        readonly attribute unsigned long numE1T1Circuits;
        readonly attribute unsigned long maxNumStm1Port;
        readonly attribute unsigned long numStm1Port;
        readonly attribute CoreNRMSystem::relatedIwfType relatedIwf;
    };

    interface CircuitEndPointSubgroup: GenericNRMPProfile::Top
    {
        readonly attribute GenericNRMSystem::ObjectIdType circuitEndPointSubgroupId;
        readonly attribute unsigned long numCircuits;
        readonly attribute CoreNRMSystem::CircuitDirectionalityType circuitDirectionality;
        readonly attribute CoreNRMSystem::TransmissionCharacteristicsType transmissionCharacteristics;
        attribute wstring userLabel;
        readonly attribute CoreNRMSystem::TrunkGroupTypeType trunkGroupType;
        attribute GenericNRMSystem::SignallingInfoType signallingInfoOfFarEnd;
    };

    interface AcFunction : GenericNRMPProfile::ManagedFunction
    {
        readonly attribute GenericNRMSystem::ObjectIdType acFunctionId;
        readonly attribute unsigned long maxNumUsers;
    };

    interface EirFunction : GenericNRMPProfile::ManagedFunction
    {
        readonly attribute GenericNRMSystem::ObjectIdType eirFunctionId;
    };
}

```



```

    readonly attribute string eirNumber;
    readonly attribute unsigned long maxNumImei;
};

interface HlrFunction : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType hlrFunctionId;
    readonly attribute unsigned long maxNumUsers;
        attribute CoreNRMSystem:: imsiRangeType imsiRange;
    readonly attribute unsigned long maxNumMdn;
        attribute CoreNRMSystem:: mdnRangeType mdnRange;
};

interface VlrFunction : GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType vlrFunctionId;
    readonly attribute unsigned long maxNumUsers;
};

interface IwfFunction: GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType iwfFunctionId;
    readonly attribute unsigned long iwfCapability;
};

interface McFunction: GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType mcFunctionId;
};

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

```

```
#endif
```

4.3.3 CoreNRMSystem

```

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

```

```

#define CoreNRMSystem_idl

#include "GenericNRMSystem.idl"

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

    typedef unsigned long MscTypeType;
    const MscTypeType TMSC1=1;
    const MscTypeType TMSC2=2;
    const MscTypeType MSCVLR=3;
    const MscTypeType GMSC=4;
    const MscTypeType MSCGMSC=5;
    const MscTypeType MSC=6;
    const MscTypeType OTHER=7;

    typedef GenericNRMSystem::DN ControlledBsc;
    typedef sequence<ControlledBsc> controlledBscListType;

    typedef GenericNRMSystem::DN relatedIwfType;

    enum CircuitDirectionalityType
    {
        onewayOut,
        onewayIn,
        twoway
    };

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

```

//the eighth bit is meaningless.

```
typedef unsigned long TrunkGroupTypeType;
const TrunkGroupTypeType toTmsc1=1;
const TrunkGroupTypeType toTmsc2=2;
const TrunkGroupTypeType toCdmaGmsc=3;
const TrunkGroupTypeType toMsc=4;
const TrunkGroupTypeType to193=5;
const TrunkGroupTypeType toIp=6;
const TrunkGroupTypeType toCncPstn=7;
const TrunkGroupTypeType toCtPstn=8;
const TrunkGroupTypeType toCmcc=9;
const TrunkGroupTypeType toCtt=10;
const TrunkGroupTypeType toBsc=11;
const TrunkGroupTypeType toGsm=12;
const TrunkGroupTypeType toOthers=13;
```

```
struct imsiRange
{
    string startImsi;
    string endImsi;
};
typedef sequence<imsiRange> imsiRangeType;
```

```
struct mdnRange
{
    string startMsisdn;
    string endMsisdn;
};
typedef sequence<mdnRange> mdnRangeType;
```

```
typedef string ipAddressType;
```

```
typedef char DestCodeElementType;
// the possible value for DestCodeElementType are:
// '0', '1', '2', '3', '4', '5', '6', '7', '8', '9',
// 'A', 'B', 'C', 'D', 'E', 'F', '*', '#'
typedef string DestCodeType; //a string type formed from DestCodeElementType
```

```
enum DestType
{
    international_dest,
    national_dest,
```

```

        local_dest,
        other_dest
    };
typedef octet NatureOfAddressType;
//From right to left, the eighth bit is meaningless, the meaning of seventh bit to first bit is
following:
//0000000 is spare;
//0000001 is subscriber number (national use);
//0000010 is unknown (national use);
//0000011 is national (significant) number;
//0000100 is international number;
//0000101 is network-specific number (national use);
//0000110 is network routing number in national (significant) number format (national use);
//0000111 is network routing number in network-specific number format (national use);
//0001000 is network routing number concatenated with Called Directory Number (national use);
//0001001 to 1101111 are spare;
//1110000 to 1111110 are reserved for national use;
//1111111 is spare.

enum DestinationTypeChoiceType
{
    NatureOfAddressChoice,
    DestTypeChoice
};
union DestinationTypeType switch(DestinationTypeChoiceType)
{
    case NatureOfAddressChoice : NatureOfAddressType natureOfAddress;
    case DestTypeChoice : DestType dest;
};
};
#endif

```

4.3.4 CorePSNRMDefs

```

//File "CorePSNRMDefs.idl"
//The IRP document version number is "CNPS NRM V1.0"
#ifndef CorePSNRMDefs_idl
#define CorePSNRMDefs_idl

#include "GenericNRMDefs.idl"

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

```

```

module CorePSNRMDefs
{
    /**
     * Definitions for MO class PdsnFunction
     */
    interface PdsnFunction : GenericNRMDefs::ManagedFunction
    {
        const string CLASS = "PdsnFunction";

        // Attribute Names
        //
        const string pdsnFunctionId = "pdsnFunctionId";
        const string intIpAddress = "intIpAddressList";
        const string extIpAddress = "extIpAddressList";
        const string careOfAddress = "careOfAddressList";
        const string maxNumPPPConns = "maxNumPPPConns";
        const string maxNumPPPConnsEstabedPerSecond="maxNumPPPConnsEstabedPerSecond";
        const string maxNumPcfs = "maxNumPcfs";
        const string authProtocol = "authProtocol";
        const string maxSipRpRegLifetime = "maxSipRpRegLifetime";
        const string maxMipRpRegLifetime = "maxMipRpRegLifetime";
        const string maxThroughput="maxThroughput";
    };
    /**
     * Definitions for MO class HaFunction
     */
    interface HaFunction: GenericNRMDefs::ManagedFunction
    {
        const string CLASS = "HaFunction";

        // Attribute Names
        //
        const string haFunctionId = "haFunctionId";
        const string intIpAddress = "intIpAddressList";
        const string extIpAddress = "extIpAddressList";
        const string maxNumSessions = "maxNumSessions";
        const string dataThroughput="dataThroughput";
    };

    /**
     * Definitions for MO class AaaFunction
     */
}

```

```

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

    // Attribute Names
    //
    const string aaaFunctionId = "aaaFunctionId";
    const string extIpAddress = "extIpAddressList";
    const string maxNumUsers = "maxNumUsers";
    const string maxNumAccAuthProcessedPerSecond="maxNumAccAuthProcessedPerSecond";
};

};

#endif

```

4.3.5 CorePSNRMPProfile

```

//File "CorePSNRMPProfile.idl"
//The IRP document version number is "CNPS NRM V1.0"
#ifndef CorePSNRMPProfile_idl
#define CorePSNRMPProfile_idl

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

/**
 * This module defines the attribute names and
 * correspondig attribute types for all defined
 * MO class in core PS NRM. This module is
 * used for reference.
 */
module CorePSNRMPProfile
{
    interface PdsnFunction: GenericNRMProfile::ManagedFunction
    {
        readonly attribute GenericNRMSystem::ObjectIdType pdsnFunctionId;
        readonly attribute CorePSNRMSSystem::IPAddressType intIpAddressList;
        readonly attribute CorePSNRMSSystem::IPAddressType extIpAddressList;
        readonly attribute CorePSNRMSSystem::IPAddressType careOfAddressList;
        readonly attribute unsigned long maxNumPPPCOnns;
        readonly attribute unsigned long maxNumPPPCOnnsEstabedPerSecond;
        readonly attribute unsigned long maxNumPcfs;
        readonly attribute CorePSNRMSSystem::authProtocolType authProtocol;
    }
}

```

```

    readonly attribute unsigned long maxSipRpRegLifetime;
    readonly attribute unsigned long maxMipRpRegLifetime;
    readonly attribute unsigned long maxThroughput;
};

interface HaFunction: GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType haFunctionId;
    readonly attribute CorePSNRMSystem::IPAddressType intIpAddressList;
    readonly attribute CorePSNRMSystem::IPAddressType extIpAddressList;
    readonly attribute unsigned long maxNumSessions;
    readonly attribute unsigned long dataThroughput;
};

interface AaaFunction: GenericNRMPProfile::ManagedFunction
{
    readonly attribute GenericNRMSystem::ObjectIdType aAAFunctionId;
    readonly attribute CorePSNRMSystem::IPAddressType extIpAddressList;
    readonly attribute unsigned long maxNumUsers;
    readonly attribute unsigned long maxNumAccAuthProcessedPerSecond;
};

};

#endif

```

4.3.6 CorePSNRMSystem

```

//File "CorePSNRMSystem.idl"
//The IRP document version number is "CNPS NRM V1.0"
#ifndef CorePSNRMSystem_idl
#define CorePSNRMSystem_idl

#include "GenericNRMSystem.idl"

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

    typedef string IPAddressType;

```

```

typedef unsigned short authProtocol;
const authProtocol MSID=0;
const authProtocol CHAP=1;
const authProtocol PAP=2;
typedef sequence<authProtocol> authProtocolType;

};
#endif

```

5 性能网络资源模型设计

5.1 性能参数的 IDL 定义

5.1.1 CdmaMeasurementDefs

```

//File CdmaMeasurementDefs.idl

#ifndef CdmaMeasurementDefs_idl
#define CdmaMeasurementDefs_idl

/**
 * This module defines measurementType names constants
 */
module CdmaMeasurementDefs
{
    // msc measurement
    module mscBasicMeasurement
    {
        // handoff forward
        const string attOutHosFw = "attOutHosFw";
        const string succOutHosFw = "succOutHosFw";
        const string attIncHosFw = "attIncHosFw";
        const string succIncHosFw = "succIncHosFw";
        // handoff backward
        const string attOutHosBw = "attOutHosBw";
        const string succOutHosBw = "succOutHosBw";
        const string attIncHosBw = "attIncHosBw";
        const string succIncHosBw = "succIncHosBw";
        // handoff to third
        const string attOutHosThird = "attOutHosThird";
        const string succOutHosThird = "succOutHosThird";
        const string attIncHosThird = "attIncHosThird";
        const string succIncHosThird = "succIncHosThird";
        // paging request
        const string attPageReqs = "attPageReqs";

```



```

const string succPageReqs = "succPageReqs";
// location update
const string attLuReqs = "attLuReqs";
const string succLuReqs = "succLuReqs";
// MSC call drop by core network
const string nbrCallDropByNetwork = "nbrCallDropByNetwork";
// fail calling
const string failCallingByUserBusy = "failCallingByUserBusy";
const string failCallingByNoAns = "failCallingByNoAns";
const string failCallingByUnallNum = "failCallingByUnallNum";
const string failCallingByCallingAuthFail = "failCallingByCallingAuthFail";
const string failCallingByCallingAssignFail = "failCallingByCallingAssignFail";
const string failCallingByOtherReason = "failCallingByOtherReason";
const string failCalledByCalledAuthFail = "failCalledByCalledAuthFail";
const string failCalledByCalledAssignFail = "failCalledByCalledAssignFail";
const string failCalledByOtherReason = "failCalledByOtherReason";
};

```

```

module mscQos

```

```

{
    const string meanDurOfCallSetup = "meanDurOfCallSetup";
    const string meanCallDur = "meanCallDur";
    const string meanDurOfTrunkSeizure = "meanDurOfTrunkSeizure";
};

```

```

//exchange measurement

```

```

module mobileTrafficFlow

```

```

{
    // originating call
    const string attOrigCalls = "attOrigCalls";
    const string succOrigCalls = "succOrigCalls";
    const string ansOrigCalls = "ansOrigCalls";
    const string failOrigCallsPerCause = "failOrigCallsPerCause";
    const string attOrigCallTraffic = "attOrigCallTraffic";
    const string succOrigCallTraffic = "succOrigCallTraffic";
    const string ansOrigCallTraffic = "ansOrigCallTraffic";
    // internal call
    const string attInternalCalls = "attInternalCalls";
    const string succInternalCalls = "succInternalCalls";
    const string ansInternalCalls = "ansInternalCalls";
    const string failInternalCallsPerCause = "failInternalCallsPerCause";
    const string attInternalCallTraffic = "attInternalCallTraffic";
    const string succInternalCallTraffic = "succInternalCallTraffic";
};

```

```
const string ansInternalCallTraffic = "ansInternalCallTraffic";
// terminating call
const string attTermCalls = "attTermCalls";
const string succTermCalls = "succTermCalls";
const string ansTermCalls = "ansTermCalls";
const string failTermCallsPerCause = "failTermCallsPerCause";
const string attTermCallTraffic = "attTermCallTraffic";
const string succTermCallTraffic = "succTermCallTraffic";
const string ansTermCallTraffic = "ansTermCallTraffic";
// incoming call
const string attIncCalls = "attIncCalls";
const string succIncCalls = "succIncCalls";
const string ansIncCalls = "ansIncCalls";
const string failIncCallsPerCause = "failIncCallsPerCause";
const string attIncCallTraffic = "attIncCallTraffic";
const string succIncCallTraffic = "succIncCallTraffic";
const string ansIncCallTraffic = "ansIncCallTraffic";
// outgoing call
const string attOutCalls = "attOutCalls";
const string succOutCalls = "succOutCalls";
const string ansOutCalls = "ansOutCalls";
const string failOutCallsPerCause = "failOutCallsPerCause";
const string attOutCallTraffic = "attOutCallTraffic";
const string succOutCallTraffic = "succOutCallTraffic";
const string ansOutCallTraffic = "ansOutCallTraffic";
// transit call
const string attTransCalls = "attTransCalls";
const string succTransCalls = "succTransCalls";
const string ansTransCalls = "ansTransCalls";
const string failTransCallsPerCause = "failTransCallsPerCause";
const string attTransCallTraffic = "attTransCallTraffic";
const string succTransCallTraffic = "succTransCallTraffic";
const string ansTransCallTraffic = "ansTransCallTraffic";
// originating outgoing call
const string attOrigOutCalls = "attOrigOutCalls";
const string succOrigOutCalls = "succOrigOutCalls";
const string ansOrigOutCalls = "ansOrigOutCalls";
const string failOrigOutCallsPerCause = "failOrigOutCallsPerCause";
const string attOrigOutCallTraffic = "attOrigOutCallTraffic";
const string succOrigOutCallTraffic = "succOrigOutCallTraffic";
const string ansOrigOutCallTraffic = "ansOrigOutCallTraffic";
// terminating incoming call
const string attTermIncCalls = "attTermIncCalls";
```

```

const string succTermIncCalls = "succTermIncCalls";
const string ansTermIncCalls = "ansTermIncCalls";
const string failTermIncCallsPerCause = "failTermIncCallsPerCause";
const string attTermIncCallTraffic = "attTermIncCallTraffic";
const string succTermIncCallTraffic = "succTermIncCallTraffic";
const string ansTermIncCallTraffic = "ansTermIncCallTraffic";
};

//circuit end point subgroup measurement
module outCircuitMeasurement
{
    const string outBids = "outBids";
    const string succOutSeizures = "succOutSeizures";
    const string ansOutSeizures = "ansOutSeizures";
    const string succOutSeizureTraffic = "succOutSeizureTraffic";
    const string ansOutSeizureTraffic = "ansOutSeizureTraffic";
    const string nbrOutAvailTrunks = "nbrOutAvailTrunks";
    const string nbrOutBlockedTrunks = "nbrOutBlockedTrunks";
    const string failOutCallByUserBusy = "failOutCallByUserBusy";
    const string failOutCallByNoAns = "failOutCallByNoAns";
    const string failOutCallByUnallNum = "failOutCallByUnallNum";
    const string failOutCallByCongestion = "failOutCallByCongestion";
    const string failOutCallByOtherReason = "failOutCallByOtherReason";
};

module incCircuitMeasurement
{
    const string succIncSeizures = "succIncSeizures";
    const string ansIncSeizures = "ansIncSeizures";
    const string succIncSeizureTraffic = "succIncSeizureTraffic";
    const string ansIncSeizureTraffic = "ansIncSeizureTraffic";
    const string nbrIncAvailTrunks = "nbrIncAvailTrunks";
    const string nbrIncBlockedTrunks = "nbrIncBlockedTrunks";
    const string failIncCallByUserBusy = "failIncCallByUserBusy";
    const string failIncCallByNoAns = "failIncCallByNoAns";
    const string failIncCallByUnallNum = "failIncCallByUnallNum";
    const string failIncCallByCongestion = "failIncCallByCongestion";
    const string failIncCallByOtherReason = "failIncCallByOtherReason";
};

// observed destination measurement
module observedDestination
{

```

```

const string attCalls = "attCalls";
const string succCalls = "succCalls";
const string ansCalls = "ansCalls";
const string succCallTraffic = "succCallTraffic";
const string ansCallTraffic = "ansCallTraffic";
const string failCallByUserBusy="failCallByUserBusy";
const string failCallByNoAns = "failCallByNoAns";
const string failCallByUnallNum = "failCallByUnallNum";
const string failCallByCongestion = "failCallByCongestion";
const string failCallByOtherReason = "failCallByOtherReason";
};

```

```
//No.7 signalling measurement
```

```
module signallingLinkSetTP
```

```

{
    const string nbrBlockedSigLinks = "nbrBlockedSigLinks";
    const string sigLinkBlocked = "sigLinkBlocked";
    const string sigLinkSetTraffic = "sigLinkSetTraffic";
};

```

```
module signallingLinkTP
```

```

{
    const string durSigLinkOutOfService = "durSigLinkOutOfService";
    const string nbrSigLinkOutOfService = "nbrSigLinkOutOfService";
    const string nbrOutMsus = "nbrOutMsus";
    const string nbrOutSifAndSioOctets = "nbrOutSifAndSioOctets";
    const string outSigLinkTraffic = "outSigLinkTraffic";
    const string nbrInMsus = "nbrInMsus";
    const string nbrInSifAndSioOctets = "nbrInSifAndSioOctets";
    const string inSigLinkTraffic = "inSigLinkTraffic";
};

```

```
//AC measurement
```

```
module acMeasurement
```

```

{
    const string attAuth = "attAuth";
    const string failAuth = "failAuth";
    const string currentNumSubsInAc = "currentNumSubsInAc";
};

```

```
//HLR measurement
```

```
module hlrMeasurement
```

```
{
```

```

const string nbrStaticSubs = "nbrStaticSubs";
const string nbrDynamicSubs = "nbrDynamicSubs";
const string attRegnots = "attRegnots";
const string succRegnots = "succRegnots";
};

//VLR measurement
module vlrMeasurement
{
    const string attRegnots = "attRegnots";
    const string succRegnots = "succRegnots";
    const string nbrCurrentRegSubs = "nbrCurrentRegSubs";
    const string nbrCurrentPowerOnSubs = "nbrCurrentPowerOnSubs";
};

//EIR measurement
module eirMeasurement
{
    const string nbrWhiteListEntries = "nbrWhiteListEntries";
    const string nbrBlackListEntries = "nbrBlackListEntries";
    const string nbrGreyListEntries = "nbrGreyListEntries";
};

//BSC measurement
module bscHandoffMeasurement
{
    const string attIncHardHoInterBs = "attIncHardHoInterBs";
    const string succIncHardHoInterBs = "succIncHardHoInterBs";
    const string attIncHardHo = "attIncHardHo";
    const string succIncHardHo = "succIncHardHo";
    const string attSoftHoAdditionInterBs = "attSoftHoAdditionInterBs";
    const string succSoftHoAdditionInterBs = "succSoftHoAdditionInterBs";
    const string attSoftHoAddition = "attSoftHoAddition";
    const string succSoftHoAddition = "succSoftHoAddition";
};

module bscPsTrafficMeasurement
{
    const string attCallsPs = "attCallsPs";
    const string nbrCallDropPs = "nbrCallDropPs";
    const string attFschAssign = "attFschAssign";
    const string succFschAssign = "succFschAssign";
    const string attRschAssign = "attRschAssign";
};

```

```

const string succRschAssign = "succRschAssign";
const string trafficByWalshPs = "trafficByWalshPs";
const string schCeTraffic = "schCeTraffic";
const string nbrForwardRlpPkts = "nbrForwardRlpPkts";
const string nbrResendedForwardRlpPkts = "nbrResendedForwardRlpPkts";
const string nbrReverseRlpPkts = "nbrReverseRlpPkts";
const string nbrResendedReverseRlpPkts = "nbrResendedReverseRlpPkts";
const string nbrCongestionRadioPart = "nbrCongestionRadioPart";
const string attHoPs = "attHoPs";
const string succHoPs = "succHoPs";
};

```

```

module bscSysResMeasurement

```

```

{
    const string nbrAvailVocoders = "nbrAvailVocoders";
    const string durVocodersSeizure = "durVocodersSeizure";
    const string nbrAvailACircuits = "nbrAvailACircuits";
    const string durACircuitsSeizure = "durACircuitsSeizure";
};

```

```

//BTS measurement

```

```

module btsSysResMeasurement

```

```

{
    const string nbrAvailCe = "nbrAvailCe";
    const string durCeSeizure = "durCeSeizure";
    const string nbrAvailTch = "nbrAvailTch";
    const string failCallsByBlock = "failCallsByBlock";
};

```

```

//Sector measurement

```

```

module circuitServiceMeasurement

```

```

{
    const string attOrigCallsCs = "attOrigCallsCs";
    const string attAssignCs = "attAssignCs";
    const string succAssignCs = "succAssignCs";
    const string succPageRspsCs = "succPageRspsCs";
    const string nbrRadioDropCs = "nbrRadioDropCs";
};

```

```

module packetServiceMeasurement

```

```

{
    const string attOrigCallsPs = "attOrigCallsPs";
    const string attAssignPs = "attAssignPs";
};

```

```

    const string succAssignPs = "succAssignPs";
    const string succPageRspsPs = "succPageRspsPs";
    const string nbrRadioDropPs = "nbrRadioDropPs";
};

module sectorTrafficMeasurement
{
    const string trafficIncludeHo = "trafficIncludeHo";
    const string trafficExcludeHo = "trafficExcludeHo";
    const string trafficByWalsh = "trafficByWalsh";
};

//PCF measurement
module pcfTrafficMeasurement
{
    const string succAccessPs = "succAccessPs";
    const string nbrCongestionPcfPart = "nbrCongestionPcfPart";
    const string maxPppConns = "maxPppConns";
    const string forwardThroughput = "forwardThroughput";
    const string reverseThroughput = "reverseThroughput";
    const string nbrDiscardedPktsFromRan = "nbrDiscardedPktsFromRan";
    const string nbrDiscardedPktsFromPdsn = "nbrDiscardedPktsFromPdsn";
};
};

#endif

```

5.1.2 CorePSMeasurementDefs

```

//File CorePSMeasurementDefs.idl
#ifndef CorePSMeasurementDefs_idl
#define CorePSMeasurementDefs_idl

/**
 * This module defines measurementType names constants
 */
module CorePSMeasurementDefs
{
    //PDSN measurement
    module pdsnMeas
    {
        const string AttRegistration = "AttRegistration";
        const string RegistrationAccepted = "RegistrationAccepted";
        const string RegistrationDeniedPerCause = "RegistrationDeniedPerCause";
        const string AttRelConn = "AttRelConn";
    }
};

```

```

const string SuccRelConn = "SuccRelConn";
const string throughputUsingSimpleIp = "throughputUsingSimpleIp";
const string throughputUsingMobileIp = "throughputUsingMobileIp";
const string GreStatsRx = "GreStatsRx";
const string GreStatsTx = "GreStatsTx";
const string SessionFailures = "SessionFailures";
const string CurrentSessions = "CurrentSessions";
const string attEstabPppConns = "attEstabPppConns";
const string succEstabPppConns = "succEstabPppConns";
const string maxPppConns = "maxPppConns";
const string maxEstabPppConnsPerSec = "maxEstabPppConnsPerSec";
const string faRegRequestsReceived = "faRegRequestsReceived";
const string faRegRequestsRelayed = "faRegRequestsRelayed";
const string faRegRequestsDeniedPerCause = "faRegRequestsDeniedPerCause";
const string faRegRepliesRecieved = "faRegRepliesRecieved";
const string faRegRepliesRelayed = "faRegRepliesRelayed";
};

//HA measurement
module haMeas
{
    const string haServiceRequestsAccepted = "haServiceRequestsAccepted";
    const string haRegistrationAccepted = "haRegistrationAccepted";
    const string haMultiBindingUnsupported = "haMultiBindingUnsupported";
    const string haServiceRequestsDeniedPerCause = "haServiceRequestsDeniedPerCause";
    const string haRegRequestsReceived = "haRegRequestsReceived";
    const string haDeRegRequestsReceived = "haDeRegRequestsReceived";
    const string haRegRepliesSent = "haRegRepliesSent";
    const string haDeRegRepliesSent = "haDeRegRepliesSent";
};

//AAA measurement
module aaaMeas
{
    const string radiusAccServTotalRequests = "radiusAccServTotalRequests";
    const string radiusAccServTotalResponses = "radiusAccServTotalResponses";
    const string radiusAccServTotalRequestsDeniedPerCause
        = "radiusAccServTotalRequestsDeniedPerCause";
    const string radiusAccServTotalRequestsFwToServiceGw
        = "radiusAccServTotalRequestsFwToServiceGw";
    const string radiusAccServTotalResponsesFwFromServiceGw
        = "radiusAccServTotalResponsesFwFromServiceGw";
    const string radiusAccServTotalRequestsFwToHomeRegion

```



```

    = "radiusAccServTotalRequestsFwToHomeRegion";
const string radiusAccServTotalResponsesFwFromHomeRegion
    = "radiusAccServTotalResponsesFwFromHomeRegion";
const string radiusAuthServTotalAccessRequests = "radiusAuthServTotalAccessRequests";
const string radiusAuthServTotalAccessAccepts = "radiusAuthServTotalAccessAccepts";
const string radiusAuthServTotalAccessRejects = "radiusAuthServTotalAccessRejects";
const string radiusAuthServTotalAccessRequestsDeniedPerCause
    = "radiusAuthServTotalAccessRequestsDeniedPerCause";
const string numProcessAccAuthPerSecond = "numProcessAccAuthPerSecond";
};

```

```
};
```

```
#endif
```

5.2 数据类型的 IDL 定义

5.2.1 CdmaMeasurementSystem

```
//File CdmaMeasurementSystem.idl
```

```
#ifndef CdmaMeasurementSystem_idl
```

```
#define CdmaMeasurementSystem_idl
```

```
/**
```

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

```
 */
```

```
module CdmaMeasurementSystem
```

```
{
```

```
    typedef unsigned long CDMAMeasurementType1;
```

```
    typedef float CDMAMeasurementType2;
```

```
    typedef unsigned long CauseType;
```

```
    const CauseType sum = 0;
```

```
    const CauseType other=65535;
```

```
    typedef CauseType CallLostReasonType;
```

```
    const CallLostReasonType callLostBecauseOfTrunkUnavailable = 1;
```

```
    const CallLostReasonType callLostBecauseOfExchangeCongested = 2;
```

```
    const CallLostReasonType abnormalCalls = 3;
```

```
};
```

```
#endif
```

5.2.2 CorePSMeasurementSystem

```
//File CorePSMeasurementSystem.idl
```

```
#ifndef CorePSMeasurementSystem_idl
```

```

#define CorePSMeasurementSystem_idl

/**
 * This module defines type definitions for performance measurements
 */
module CorePSMeasurementSystem
{
    typedef unsigned long CDMAPSMeasurementType1;
    typedef float CDMAPSMeasurementType2;
};

#endif

```

6 性能管理接口功能相关的文件

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

下面的Schema文件中用到的字段的说明参见附录A，示例参见附录B。

版本号：PM FILE V1.0

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- Measurement collection data file XML schema measCollec.xsd -->
<schema targetNamespace="http://latest/nmc-omc/cmNrm.doc#measCollec" elementFormDefault="qualified"
xmlns="http://www.w3.org/2001/XMLSchema" xmlns:mc="http://latest/nmc-omc/cmNrm.doc#measCollec">
  <!-- Measurement collection data file root XML element -->
  <element name="measCollecFile">
    <complexType>
      <sequence>
        <element name="fileHeader">
          <complexType>
            <sequence>
              <element name="fileSender">
                <complexType>
                  <attribute name="localDn" type="string" use="optional"/>
                  <attribute name="elementType" type="string" use="optional"/>
                </complexType>
              </element>
            </sequence>
          </complexType>
        </element>
        <element name="measCollec">
          <complexType>
            <attribute name="beginTime" type="dateTime" use="required"/>
          </complexType>
        </element>
      </sequence>
    </complexType>
  </element>
</schema>

```

```

</sequence>
  <attribute name="fileFormatVersion" type="string" use="required"/>
  <attribute name="vendorName" type="string" use="optional"/>
  <attribute name="dnPrefix" type="string" use="optional"/>
</complexType>
</element>
<element name="measData" minOccurs="0" maxOccurs="unbounded">
  <complexType>
    <sequence>
      <element name="managedElement">
        <complexType>
          <attribute name="localDn" type="string" use="optional"/>
          <attribute name="userLabel" type="string" use="optional"/>
          <attribute name="swVersion" type="string" use="optional"/>
        </complexType>
      </element>
      <element name="measInfo" minOccurs="0" maxOccurs="unbounded">
        <complexType>
          <sequence>
            <element name="job" minOccurs="0">
              <complexType>
                <attribute name="jobId" type="string" use="required"/>
              </complexType>
            </element>
            <element name="granPeriod">
              <complexType>
                <attribute name="duration" type="duration" use="required"/>
                <attribute name="endTime" type="dateTime" use="required"/>
              </complexType>
            </element>
            <element name="repPeriod" minOccurs="0">
              <complexType>
                <attribute name="duration" type="duration" use="required"/>
              </complexType>
            </element>
          </sequence>
        </complexType>
      </element>
    </choice>
  </complexType>
</element>
<choice>
  <element name="measTypes">

```

```

    <simpleType>
      <list itemType="mc:measName"/>
    </simpleType>
  </element>
<element name="measType" minOccurs="0" maxOccurs="unbounded">
  <complexType>
    <simpleContent>
      <extension base="mc:measName">
        <attribute name="p" type="positiveInteger" use="required"/>
      </extension>
    </simpleContent>
  </complexType>
</element>
</choice>
<element name="measValue" minOccurs="0" maxOccurs="unbounded">
  <complexType>
    <sequence>
      <choice>
        <element name="measResults">
          <simpleType>
            <list itemType="mc:measResultType"/>
          </simpleType>
        </element>
        <element name="r" minOccurs="0" maxOccurs="unbounded">
          <complexType>
            <simpleContent>
              <extension base="mc:measResultType">
                <attribute name="p" type="positiveInteger" use="required"/>
              </extension>
            </simpleContent>
          </complexType>
        </element>
      </choice>
      <element name="suspect" type="boolean" minOccurs="0"/>
    </sequence>
    <attribute name="measObjLdn" type="string" use="required"/>
  </complexType>

```

```

        </element>
    </sequence>
</complexType>
</element>
</sequence>
</complexType>
</element>
<element name="fileFooter">
    <complexType>
        <sequence>
            <element name="measCollec">
                <complexType>
                    <attribute name="endTime" type="dateTime" use="required"/>
                </complexType>
            </element>
        </sequence>
    </complexType>
</element>
</sequence>
</complexType>
</element>
</sequence>
</complexType>
</element>
<simpleType name="measNameWithSubCounter">
    <restriction base="string">
        <pattern
value="(mobileTrafficFlow.failOrigCallsPerCause. | mobileTrafficFlow.failInternalCallsPerCause. | mobileTrafficFlow.failTermCallsPerCause. | mobileTrafficFlow.failIncCallsPerCause. | mobileTrafficFlow.failOutCallsPerCause. | mobileTrafficFlow.failTransCallsPerCause. | mobileTrafficFlow.failOrigOutCallsPerCause. | mobileTrafficFlow.failTermIncCallsPerCause.)\d{1,5}"/>
    </restriction>
</simpleType>
<simpleType name="measNameWithoutSubCounter">
    <restriction base="string">
        <enumeration value="mscBasicMeasurement.attOutHosFw"/>
        <enumeration value="mscBasicMeasurement.succOutHosFw"/>
        <enumeration value="mscBasicMeasurement.attIncHosFw"/>
        <enumeration value="mscBasicMeasurement.succIncHosFw"/>
        <enumeration value="mscBasicMeasurement.attOutHosBw"/>
    </restriction>
</simpleType>

```

<enumeration value="mscBasicMeasurement.succOutHosBw"/>
<enumeration value="mscBasicMeasurement.attIncHosBw"/>
<enumeration value="mscBasicMeasurement.succIncHosBw"/>
<enumeration value="mscBasicMeasurement.attOutHosThird"/>
<enumeration value="mscBasicMeasurement.succOutHosThird"/>
<enumeration value="mscBasicMeasurement.attIncHosThird"/>
<enumeration value="mscBasicMeasurement.succIncHosThird"/>
<enumeration value="mscBasicMeasurement.attPageReqs"/>
<enumeration value="mscBasicMeasurement.succPageReqs"/>
<enumeration value="mscBasicMeasurement.attLuReqs"/>
<enumeration value="mscBasicMeasurement.succLuReqs"/>
<enumeration value="mscBasicMeasurement.nbrCallDropByNetwork"/>
<enumeration value="mscBasicMeasurement.failCallingByUserBusy"/>
<enumeration value="mscBasicMeasurement.failCallingByNoAns"/>
<enumeration value="mscBasicMeasurement.failCallingByUnallNum"/>
<enumeration value="mscBasicMeasurement.failCallingByCallingAuthFail"/>
<enumeration value="mscBasicMeasurement.failCallingByCallingAssignFail"/>
<enumeration value="mscBasicMeasurement.failCallingByOtherReason"/>
<enumeration value="mscBasicMeasurement.failCalledByCalledAuthFail"/>
<enumeration value="mscBasicMeasurement.failCalledByCalledAssignFail"/>
<enumeration value="mscBasicMeasurement.failCalledByOtherReason"/>
<enumeration value="mscQos.meanDurOfCallSetup"/>
<enumeration value="mscQos.meanCallDur"/>
<enumeration value="mscQos.meanDurOfTrunkSeizure"/>
<enumeration value="mobileTrafficFlow.attOrigCalls"/>
<enumeration value="mobileTrafficFlow.succOrigCalls"/>
<enumeration value="mobileTrafficFlow.ansOrigCalls"/>
<enumeration value="mobileTrafficFlow.attOrigCallTraffic"/>
<enumeration value="mobileTrafficFlow.succOrigCallTraffic"/>
<enumeration value="mobileTrafficFlow.ansOrigCallTraffic"/>
<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.attTermCalls"/>

<enumeration value="mobileTrafficFlow.succTermCalls"/>
<enumeration value="mobileTrafficFlow.ansTermCalls"/>
<enumeration value="mobileTrafficFlow.attTermCallTraffic"/>
<enumeration value="mobileTrafficFlow.succTermCallTraffic"/>
<enumeration value="mobileTrafficFlow.ansTermCallTraffic"/>
<enumeration value="mobileTrafficFlow.ansTermCallTraffic"/>
<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="outCircuitMeasurement.outBids"/>
<enumeration value="outCircuitMeasurement.succOutSeizures"/>

```

<enumeration value="outCircuitMeasurement.ansOutSeizures"/>
<enumeration value="outCircuitMeasurement.succOutSeizureTraffic"/>
<enumeration value="outCircuitMeasurement.ansOutSeizureTraffic"/>
<enumeration value="outCircuitMeasurement.nbrOutAvailTrunks"/>
<enumeration value="outCircuitMeasurement.nbrOutBlockedTrunks"/>
<enumeration value="outCircuitMeasurement.failOutCallByUserBusy"/>
<enumeration value="outCircuitMeasurement.failOutCallByNoAns"/>
<enumeration value="outCircuitMeasurement.failOutCallByUnallNum"/>
<enumeration value="outCircuitMeasurement.failOutCallByCongestion"/>
<enumeration value="outCircuitMeasurement.failOutCallByOtherReason"/>
<enumeration value="incCircuitMeasurement.succIncSeizures"/>
<enumeration value="incCircuitMeasurement.ansIncSeizures"/>
<enumeration value="incCircuitMeasurement.succIncSeizureTraffic"/>
<enumeration value="incCircuitMeasurement.ansIncSeizureTraffic"/>
<enumeration value="incCircuitMeasurement.nbrIncAvailTrunks"/>
<enumeration value="incCircuitMeasurement.nbrIncBlockedTrunks"/>
<enumeration value="incCircuitMeasurement.failIncCallByUserBusy"/>
<enumeration value="incCircuitMeasurement.failIncCallByNoAns"/>
<enumeration value="incCircuitMeasurement.failIncCallByUnallNum"/>
<enumeration value="incCircuitMeasurement.failIncCallByCongestion"/>
<enumeration value="incCircuitMeasurement.failIncCallByOtherReason"/>
<enumeration value="observedDestination.attCalls"/>
<enumeration value="observedDestination.succCalls"/>
<enumeration value="observedDestination.ansCalls"/>
<enumeration value="observedDestination.succCallTraffic"/>
<enumeration value="observedDestination.ansCallTraffic"/>
<enumeration value="observedDestination.failCallByUserBusy"/>
<enumeration value="observedDestination.failCallByNoAns"/>
<enumeration value="observedDestination.failCallByUnallNum"/>
<enumeration value="observedDestination.failCallByCongestion"/>
<enumeration value="observedDestination.failCallByOtherReason"/>
<enumeration value="signallingLinkSetTP.nbrBlockedSigLinks"/>
<enumeration value="signallingLinkSetTP.sigLinkBlocked"/>
<enumeration value="signallingLinkSetTP.sigLinkSetTraffic"/>
<enumeration value="signallingLinkTP.durSigLinkOutOfService"/>
<enumeration value="signallingLinkTP.nbrSigLinkOutOfService"/>
<enumeration value="signallingLinkTP.nbrOutMsus"/>

```



```
<enumeration value="signallingLinkTP.nbrOutSifAndSioOctets"/>
<enumeration value="signallingLinkTP.outSigLinkTraffic"/>
<enumeration value="signallingLinkTP.nbrInMsus"/>
<enumeration value="signallingLinkTP.nbrInSifAndSioOctets"/>
<enumeration value="signallingLinkTP.inSigLinkTraffic"/>
<enumeration value="acMeasurement.attAuth"/>
<enumeration value="acMeasurement.failAuth"/>
<enumeration value="acMeasurement.currentNumSubsInAc"/>
<enumeration value="hlrMeasurement.nbrStaticSubs"/>
<enumeration value="hlrMeasurement.nbrDynamicSubs"/>
<enumeration value="hlrMeasurement.attRegnots"/>
<enumeration value="hlrMeasurement.succRegnots"/>
<enumeration value="vlrMeasurement.attRegnots"/>
<enumeration value="vlrMeasurement.succRegnots"/>
<enumeration value="eirMeasurement.nbrWhiteListEntries"/>
<enumeration value="eirMeasurement.nbrBlackListEntries"/>
<enumeration value="eirMeasurement.nbrGreyListEntries"/>
<enumeration value="pdsnMeas.attRegistration"/>
<enumeration value="pdsnMeas.registrationAccepted"/>
<enumeration value="pdsnMeas.registrationDenied"/>
<enumeration value="pdsnMeas.registrationDeniedByUnspecifiedCause"/>
<enumeration value="pdsnMeas.registrationDeniedByExternalCause"/>
<enumeration value="pdsnMeas.registrationDeniedByInsufficientResources"/>
<enumeration value="pdsnMeas.attRelConn"/>
<enumeration value="pdsnMeas.succRelConn"/>
<enumeration value="pdsnMeas.throughputUsingSimpleIp"/>
<enumeration value="pdsnMeas.throughputUsingMobileIp"/>
<enumeration value="pdsnMeas.greStatsRx"/>
<enumeration value="pdsnMeas.greStatsTx"/>
<enumeration value="pdsnMeas.sessionFailures"/>
<enumeration value="pdsnMeas.currentSessions"/>
<enumeration value="pdsnMeas.sessionDurationMean"/>
<enumeration value="pdsnMeas.attEstabPppConns"/>
<enumeration value="pdsnMeas.succEstabPppConns"/>
<enumeration value="pdsnMeas.maxPppConns"/>
<enumeration value="pdsnMeas.maxEstabPppConnsPerSec"/>
<enumeration value="pdsnMeas.faRegRequestsReceived"/>
```

<enumeration value="pdsnMeas.faRegRequestsRelayed"/>
<enumeration value="pdsnMeas.faReasonUnspecified"/>
<enumeration value="pdsnMeas.faAdmProhibited"/>
<enumeration value="pdsnMeas.faInsufficientResource"/>
<enumeration value="pdsnMeas.faMNAAuthenticationFailure"/>
<enumeration value="pdsnMeas.faRegLifetimeTooLong"/>
<enumeration value="pdsnMeas.faPoorlyFormedRequests"/>
<enumeration value="pdsnMeas.faEncapsulationUnavailable"/>
<enumeration value="pdsnMeas.faVJCompressionUnavailable"/>
<enumeration value="pdsnMeas.faHAUnreachable"/>
<enumeration value="pdsnMeas.faRegRepliesRecieved"/>
<enumeration value="pdsnMeas.faRegRepliesRelayed"/>
<enumeration value="pdsnMeas.faHAAAuthenticationFailure"/>
<enumeration value="pdsnMeas.faPoorlyFormedReplies"/>
<enumeration value="haMeas.haServiceRequestsAccepted"/>
<enumeration value="haMeas.haServiceRequestsDenied"/>
<enumeration value="haMeas.haOverallServiceTime"/>
<enumeration value="haMeas.haRegistrationAccepted"/>
<enumeration value="haMeas.haMultiBindingUnsupported"/>
<enumeration value="haMeas.haReasonUnspecified"/>
<enumeration value="haMeas.haAdmProhibited"/>
<enumeration value="haMeas.haInsufficientResource"/>
<enumeration value="haMeas.haMNAAuthenticationFailure"/>
<enumeration value="haMeas.haFAAAuthenticationFailure"/>
<enumeration value="haMeas.haIDMismatch"/>
<enumeration value="haMeas.haPoorlyFormedRequest"/>
<enumeration value="haMeas.haTooManyBindings"/>
<enumeration value="haMeas.haUnknownHA"/>
<enumeration value="haMeas.haGratuitiousARPsSent"/>
<enumeration value="haMeas.haProxyARPsSent"/>
<enumeration value="haMeas.haRegRequestsReceived"/>
<enumeration value="haMeas.haDeRegRequestsReceived"/>
<enumeration value="haMeas.haRegRepliesSent"/>
<enumeration value="haMeas.haDeRegRepliesSent"/>
<enumeration value="aaaMeas.radiusAccServTotalRequests"/>
<enumeration value="aaaMeas.radiusAccServTotalInvalidRequests"/>
<enumeration value="aaaMeas.radiusAccServTotalDupRequests"/>

```

<enumeration value="aaaMeas.radiusAccServTotalResponses"/>
<enumeration value="aaaMeas.radiusAccServTotalMalformedRequests"/>
<enumeration value="aaaMeas.radiusAccServTotalBadAuthenticators"/>
<enumeration value="aaaMeas.radiusAccServTotalPacketsDropped"/>
<enumeration value="aaaMeas.radiusAccServTotalRequestsFwToServiceGw"/>
<enumeration value="aaaMeas.radiusAccServTotalResponsesFwFromServiceGw"/>
<enumeration value="aaaMeas.radiusAccServTotalRequestsFwToHomeRegion"/>
<enumeration value="aaaMeas.radiusAccServTotalResponsesFwFromHomeRegion"/>
<enumeration value="aaaMeas.radiusAuthServTotalAccessRequests"/>
<enumeration value="aaaMeas.radiusAuthServTotalInvalidRequests"/>
<enumeration value="aaaMeas.radiusAuthServTotalDupAccessRequests"/>
<enumeration value="aaaMeas.radiusAuthServTotalAccessAccepts"/>
<enumeration value="aaaMeas.radiusAuthServTotalAccessRejects"/>
<enumeration value="aaaMeas.radiusAuthServTotalAccessChallenges"/>
<enumeration value="aaaMeas.radiusAuthServTotalMalformedAccessRequests"/>
<enumeration value="aaaMeas.radiusAuthServTotalBadAuthenticators"/>
<enumeration value="aaaMeas.radiusAuthServTotalUnknownTypes"/>
<enumeration value="aaaMeas.radiusAuthServTotalAccessRejectsByInputErr"/>
<enumeration value="aaaMeas.numProcessAccAuthPerSecond"/>
<enumeration value="bscHandoffMeasurement.attIncHardHoInterBs"/>
<enumeration value="bscHandoffMeasurement.succIncHardHoInterBs"/>
<enumeration value="bscHandoffMeasurement.attIncHardHo"/>
<enumeration value="bscHandoffMeasurement.succIncHardHo"/>
<enumeration value="bscHandoffMeasurement.attSoftHoAdditionInterBs"/>
<enumeration value="bscHandoffMeasurement.succSoftHoAdditionInterBs"/>
<enumeration value="bscHandoffMeasurement.attSoftHoAddition"/>
<enumeration value="bscHandoffMeasurement.succSoftHoAddition"/>
<enumeration value="bscPsTrafficMeasurement.attCallsPs"/>
<enumeration value="bscPsTrafficMeasurement.nbrCallDropPs"/>
<enumeration value="bscPsTrafficMeasurement.attFschAssign"/>
<enumeration value="bscPsTrafficMeasurement.succFschAssign"/>
<enumeration value="bscPsTrafficMeasurement.attRschAssign"/>
<enumeration value="bscPsTrafficMeasurement.succRschAssign"/>
<enumeration value="bscPsTrafficMeasurement.trafficByWalshPs"/>
<enumeration value="bscPsTrafficMeasurement.schCeTraffic"/>
<enumeration value="bscPsTrafficMeasurement.nbrForwardRlpPkts"/>
<enumeration value="bscPsTrafficMeasurement.nbrResendedForwardRlpPkts"/>

```

```

    <enumeration value="bscPsTrafficMeasurement.nbrReverseRlpPkts"/>
    <enumeration value="bscPsTrafficMeasurement.nbrResendedReverseRlpPkts"/>
    <enumeration value="bscPsTrafficMeasurement.nbrCongestionRadioPart"/>
    <enumeration value="bscPsTrafficMeasurement.attHoPs"/>
    <enumeration value="bscPsTrafficMeasurement.succHoPs"/>
    <enumeration value="bscSysResMeasurement.nbrAvailVocoders"/>
    <enumeration value="bscSysResMeasurement.durVocodersSeizure"/>
    <enumeration value="bscSysResMeasurement.nbrAvailACircuits"/>
    <enumeration value="bscSysResMeasurement.durACircuitsSeizure"/>
    <enumeration value="btsSysResMeasurement.nbrAvailCe"/>
    <enumeration value="btsSysResMeasurement.durCeSeizure"/>
    <enumeration value="btsSysResMeasurement.nbrAvailTch"/>
    <enumeration value="btsSysResMeasurement.failCallsByBlock"/>
    <enumeration value="circuitServiceMeasurement.attOrigCallsCs"/>
    <enumeration value="circuitServiceMeasurement.attAssignCs"/>
    <enumeration value="circuitServiceMeasurement.succAssignCs"/>
    <enumeration value="circuitServiceMeasurement.succPageRspsCs"/>
    <enumeration value="circuitServiceMeasurement.nbrRadioDropCs"/>
    <enumeration value="packetServiceMeasurement.attOrigCallsPs"/>
    <enumeration value="packetServiceMeasurement.attAssignPs"/>
    <enumeration value="packetServiceMeasurement.succAssignPs"/>
    <enumeration value="packetServiceMeasurement.succPageRspsPs"/>
    <enumeration value="packetServiceMeasurement.nbrRadioDropPs"/>
    <enumeration value="sectorTrafficMeasurement.trafficIncludeHo"/>
    <enumeration value="sectorTrafficMeasurement.trafficExcludeHo"/>
    <enumeration value="sectorTrafficMeasurement.trafficByWalsh"/>
    <enumeration value="pcfTrafficMeasurement.succAccessPs"/>
    <enumeration value="pcfTrafficMeasurement.nbrCongestionPcfPart"/>
    <enumeration value="pcfTrafficMeasurement.maxPppConns"/>
    <enumeration value="pcfTrafficMeasurement.forwardThroughput"/>
    <enumeration value="pcfTrafficMeasurement.reverseThroughput"/>
    <enumeration value="pcfTrafficMeasurement.nbrDiscardedPktsFromRan"/>
    <enumeration value="pcfTrafficMeasurement.nbrDiscardedPktsFromPdsn"/>
  </restriction>
</simpleType>
<simpleType name="measName">
  <union memberTypes="mc:measNameWithSubCounter mc:measNameWithoutSubCounter"/>

```

```

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

```

6.2 性能测量数据文件的 XML header 定义

在实际性能测量数据文件中应该使用下面的XML header定义：

```

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

```

广东省网络空间安全协会受控资料

附录 A
(规范性附录)
Schema 文档补充说明

A.1 XML Schema文档结构标记约定如图 1 所示。

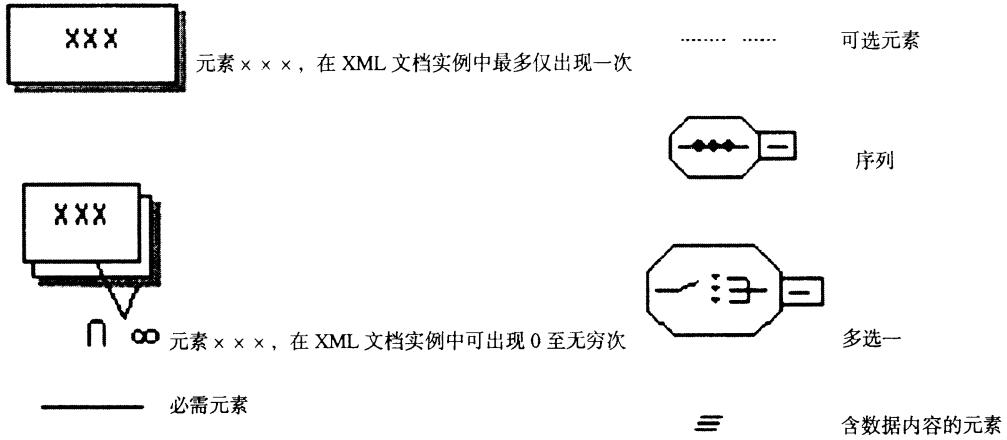


图 1 XML Schema 文档结构标记约定

A.2 性能测量数据文件的Schema定义<measCollec.xsd>

a) XML Schema 文档结构如图 2 所示。

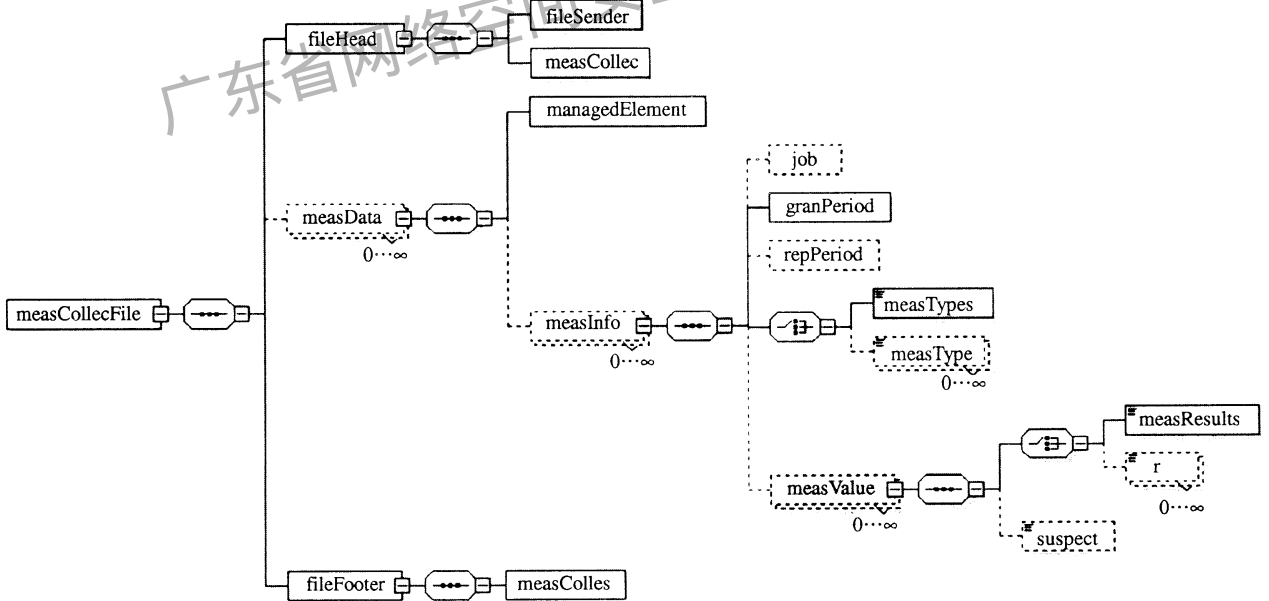


图 2 XML Schema 文档结构

b) XML Schema 文档元素/属性说明如表 1 所示。

表 1 XML Schema 文档元素/属性说明

元素/属性名称		元素/属性描述
measCollecFile		性能数据采集文件。是该 Schema 的根元素，由三个子元素组成：文件头部（fileHeader）、采集数据(measData)以及文件尾部(fileFooter)
fileHeader	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）。自身包含三个属性：文件格式版本（fileFormatVersion）、制造商名称（vendorName）和识别名前缀（dnPrefix）
	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	文件发送方。包含两个属性：本地识别名（localDN）、网元类型（elementType）
	elementType	
managedElement	localDn	被管网元。包括三个属性本地识别名（localDn）、用户友好名（userLabel）、软件版本（swVersion）
	userLabel	
	swVersion	
measInfo		测量信息。由四个子元素组成：测量任务（job）、测量粒度周期（granPeriod）、测量上报周期（repPeriod）、测量类型（measType/measTypes）和测量值（measValue）
job		测量任务。该元素为可选元素，由其类型 JobID 唯一标识
granPeriod	duration	测量粒度周期。包含两个属性：持续时间（duration）、结束时间（endTime）
	endTime	
repPeriod	duration	测量上报周期。该元素为可选元素，包含惟一属性：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 以列表方式呈现，且只出现一次；measType 可出现多次，由属性值为非负数的 p 加以区分
measType p		p 为属性限定。属性用于区分不同的 measType
measResults/r		采集结果。均由 measResultType 扩展而来。在 XML 文件实例中，两个元素择一使用。值为空表示该采集项的取值无法获得。不同的是 measResults 以列表方式呈现，且只出现一次；r 可出现多次，由属性值为非负数的 p 加以区分。r 的 p 属性应与 measType 的 p 属性一一对应
r p		p 为属性限定。表示对 <measType p> 的一个采集结果应答。<r p> 需和 <measType p> 一一对应
measValue	measObjLdn	采集值。由两个子元素组成：采集结果列表(measResults/r)和一个标记采集数据是否可信的标志位(suspect)。本身还包含一个属性：测量对象本地识别名(measObjLdn)
suspect		用于标记采集值是否可信。默认值为 False（即可信）
measCollec	beginTime	性能采集开始时间
	endTime	性能采集结束时间
measName		性能测量项名称。分为包含 SubCounter（measNameWithSubCounter）和不含 SubCounter（measNameWithoutSubCounter）两类。从 3GPP 规范中扩展而来
measNameWithSubCounter		含 SubCounter 的数据测量项名称。表示为 familyname.measurename.subcounter 形式。从 3GPP 规范中扩展而来
measNameWithoutSubCounter		不含 SubCounter 的数据测量项名称。表示为 familyname.measurename 形式。从 3GPP 规范中扩展而来

附录 B
(资料性附录)

性能管理功能相关 XML 文件示例

B.1 性能管理功能相关XML文件示例一

```

<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="MeasDataCollection.xsl"?>
<!-- The following is an example of a XML schema based XML measurement report file without use of optional
positioning attributes on measurement types and results -->
<measCollecFile xmlns="http://latest/nmc-omc/cmNrm.doc#measCollec"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://latest/nmc-omc/cmNrm.doc#measCollec
D:\Downloads\GB\CDMA20~1.XSD">
  <fileHeader fileFormatVersion="PM FILE V1.0" 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"/>
    <measCollec beginTime="2000-03-01T14:00:00+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="2000-03-01T14:15:00+02:00"/>
</fileFooter>
</measCollecFile>

```

B.2 性能管理功能相关XML文件示例二

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- The following is an example of a XML schema based XML measurement report file with use of optional
positioning attributes on measurement types and results -->
<measCollecFile xmlns="http://latest/nmc-omc/cmNrm.doc#measCollec"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://latest/nmc-omc/cmNrm.doc#measCollec
D:\Downloads\GB\CDMA20~1.XSD">
  <fileHeader fileFormatVersion="PM FILE V1.0" 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"/>
      <measCollec beginTime="2000-03-01T14:00:00+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="2000-03-01T14:15:00+02:00"/>
</fileFooter>
</measCollecFile>
```

广东省网络空间安全协会受控资料

中华人民共和国
通信行业标准

2GHz cdma2000 数字蜂窝移动通信网网络管理技术要求（第一阶段）
第3部分 基于CORBA技术的网络资源模型设计

YD/T 1587.3-2007

*

人民邮电出版社出版发行
北京市崇文区夕照寺街14号A座
邮政编码：100061

*

版权所有 不得翻印

*

本书如有印装质量问题，请与本社联系 电话：(010)67114922