broadband forum

TECHNICAL REPORT

TR-051
DSL Specific Conventions for the ITU-T Q.822.1 Performance Management Bulk Data File Structure

Issue: 1.0
Issue Date: April 2002

Notice

The Broadband Forum is a non-profit corporation organized to create guidelines for broadband network system development and deployment. This Broadband Forum Technical Report has been approved by members of the Forum. This Broadband Forum Technical Report is not binding on the Broadband Forum, any of its members, or any developer or service provider. This Broadband Forum Technical Report is subject to change, but only with approval of members of the Forum. This Technical Report is copyrighted by the Broadband Forum, and all rights are reserved. Portions of this Technical Report may be copyrighted by Broadband Forum members.

This Broadband Forum Technical Report is provided AS IS, WITH ALL FAULTS. ANY PERSON HOLDING A COPYRIGHT IN THIS BROADBAND FORUM TECHNICAL REPORT, OR ANY PORTION THEREOF, DISCLAIMS TO THE FULLEST EXTENT PERMITTED BY LAW ANY REPRESENTATION OR WARRANTY, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, ANY WARRANTY:

(A)   OF ACCURACY, COMPLETENESS, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, OR TITLE;
(B)   THAT THE CONTENTS OF THIS BROADBAND FORUM TECHNICAL REPORT ARE SUITABLE FOR ANY PURPOSE, EVEN IF THAT PURPOSE IS KNOWN TO THE COPYRIGHT HOLDER;
(C)   THAT THE IMPLEMENTATION OF THE CONTENTS OF THE DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

By using this Broadband Forum Technical Report, users acknowledge that implementation may require licenses to patents. The Broadband Forum encourages but does not require its members to identify such patents. For a list of declarations made by Broadband Forum member companies, please see http://www.broadband-forum.org. No assurance is given that licenses to patents necessary to implement this Technical Report will be available for license at all or on reasonable and non-discriminatory terms.

ANY PERSON HOLDING A COPYRIGHT IN THIS BROADBAND FORUM TECHNICAL REPORT, OR ANY PORTION THEREOF, DISCLAIMS TO THE FULLEST EXTENT PERMITTED BY LAW (A) ANY LIABILITY (INCLUDING DIRECT, INDIRECT, SPECIAL, OR CONSEQUENTIAL DAMAGES UNDER ANY LEGAL THEORY) ARISING FROM OR RELATED TO THE USE OF OR RELIANCE UPON THIS TECHNICAL REPORT; AND (B) ANY OBLIGATION TO UPDATE OR CORRECT THIS TECHNICAL REPORT.

Broadband Forum Technical Reports may be copied, downloaded, stored on a server or otherwise re-distributed in their entirety only, and may not be modified without the advance written permission of the Broadband Forum.

The text of this notice must be included in all copies.

Technical comments or questions about this Technical Report should be directed to:

EDITOR: Andrew J. Mayer, Ph.D. Telcordia Technologies, Inc. amayer@telcordia.com

Abstract:
This specifications presents a DSL bulk performance data file structure supporting performance monitoring statistics that is aligned with the ITU-T Q.822.1 file structure. Alignment with the Q.822.1 bulk PM data file structure will help to promote consistency across technologies. Normative requirements for a DSL bulk performance data file are presented.

TABLE OF CONTENTS

1. INTRODUCTION

This specification defines a structure for DSL bulk file performance monitoring statistics that is aligned with the ITU-T Q.822.1 file structure. Normative requirements for using the ITU-T Q.822.1 performance data file structure to represent DSL bulk performance data are presented in Section 3. The generation of PM bulk data files provides an alternative to the bulk retrieval methods currently defined within the CORBA MIB (ref to Broadband Forum TR-050), as the performance monitoring data is stored in files rather than transported over CORBA. The PM bulk file is independent of the network management protocol used by an NMS, EMS or NE. It is expected that the PM bulk data file will be generated via a network management request, as in the CORBA MIB, or automatically. The performance monitoring data files may be retrieved by an NMS using a separate bulk file transfer mechanism.

The DSL CORBA MIB and Q.822.1 provide mechanisms to support requests for generating bulk performance monitoring information directly to bulk data files. Once generated, these bulk data files may be retrieved from the agent by the manager outside the CORBA interface (or CMIP or SNMP interface) using a bulk file transfer mechanism, such as FTP. Q.822.1 Annex A defines a bulk file that may be used to for DSL performance data. This specification defines conventions for applying the Q.822.1 file format to DSL. This approach is extensible and allows the counters supported in each data set to be identified within the bulk PM data file.

2. ITU-T Q.822.1 Performance Data File Structure

The following is found in Annex A of ITU-T Q.822.1

2.1 Text File Format BNF

The BNF in this annex defines the format of performance measurement data file, named "q822d1FileText".

```
<File> := #file <FD> <FileType> <Newline>
 <Nodes> #endfile <Newline>

<FileType> := <FileFormat> <FD> <FormatVersion> <FD> <ModelParadigm>

<FileFormat> := <String>

<FormatVersion> := <String>

<ModelParadigm> := <String>

<Nodes> := <Node> <Nodes>
  | <Node>
```

```
<Node> := #node <FD> <NodeID> <FD> <MeasuredObjectIDPrefix> <Newline>
  <Tables> #endnode <Newline>

<NodeID> := <String>

<MeasuredObjectIDPrefix> := <ID>
  | <Empty>

<Tables> := <MeasuredObjectIDAliases> <Table> <Tables>
  | <MeasuredObjectIDAliases> <Table>

<MeasuredObjectIDAliases> := <MeasuredObjectIDAlias>
<MeasuredObjectIDAliases>
  | <MeasuredObjectIDAlias>
  | <Empty>

<MeasuredObjectIDAlias> := #idalias <FD> <ShortID> <FD>   <LongID>
<Newline>

<ShortID> := <ID>

<LongID> := <ID>

<Table> := #table <FD> <Header> <RecordGroups> #endtable <Newline>

<Header> := #header <FD> <MeasuredObjectType> <FD> <GranualarityPeriod>
  <Newline> <DataSetTypes> #endheader <Newline>

<MeasuredObjectType> := <String>

<GranualarityPeriod> := <TimePeriodValue>

<DataSetTypes> := <DataSetType> <DataSetTypes>
  | <DataSetType>

<DataSetType> := #dataset <FD> <DataSetTypeName> <FD> <DataSetTypeIndex>
  <Newline> suspect <FD> <ParameterTypes> <Newline>
  #enddataset <Newline>

<DataSetTypeName> := <String>

<DataSetTypeIndex> := <IntegerString>

<ParameterTypes> := <ParameterType> <ParameterTypes>
  | <ParameterType>

<ParameterType> := <String> <FieldSeparator>

<RecordGroups> := <RecordGroup> <RecordGroups>
  | <RecordGroup>

<RecordGroup> := #period <FD> <PeriodEndTime> <Newline>
  <Records> #endperiod <NewLine>

<PeriodEndTime> := <TimeValue>

<Records> := <Record> <Records>
  | <Record>

<Record> := <MeasuredObjectID> <FD> <DataSetValues> <Newline>

<MeasuredObjectID> := <ID>        // can be aliased and prefixed

<DataSetValues> := <DataSetValue> <DataSetValues>
  | <DataSetValue>

<DataSetValue> := <DataSetTypeIndex> <FD> <Suspect> <FD>
```

```
<ParameterValues>

<Suspect> := <BooleanValue>

<ParameterValues> := <ParameterValue> <ParameterValues>
  | <ParameterValue>

<ParameterValue> := <Value> <FD>

<Value> := <CounterValue>
  | <GaugeValue>
  | <TidemarkValue>
  | <BooleanValue>
  | <EnumValue>
  | <TimeValue>
  | <TimePeriodValue>

<FD> := :          // field delimiter

<EscapeCharacter> := \

<Newline> := '\n'// line delimiter

<ID> := <String>

<CounterValue> := <IntegerString>

<GaugeValue> := <FloatString>

<TidemarkValue> := <FloatString>

<BooleanValue> := <BooleanString>

<EnumValue> := <IntegerString>

<TimeValue> := <String>  // UTC in format "YYYYMMDDHHMMSS.fffZ"

<TimePeriodValue> := <IntegerString> <TimePeriodType>

<BooleanString> := T | F // true or false

<TimePeriodType> := days | hours | minutes | seconds

<String> := {ISO 8859-1(Latin-1) charatcters}

<IntegerString> := <IntegerString> <Digit>
  | <Digit>

<Digit> := 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0

<FloatString> := <Mantissa> | <Mantissa> <Exponent>

<Mantissa> := <IntegerString> | <IntegerString> . | . <IntegerString>
        | <IntegerString> . <IntegerString>

<Exponent> := <Exp> <Sign> <IntegerString>

<Sign> := + | -

<Exp> := E | e
```
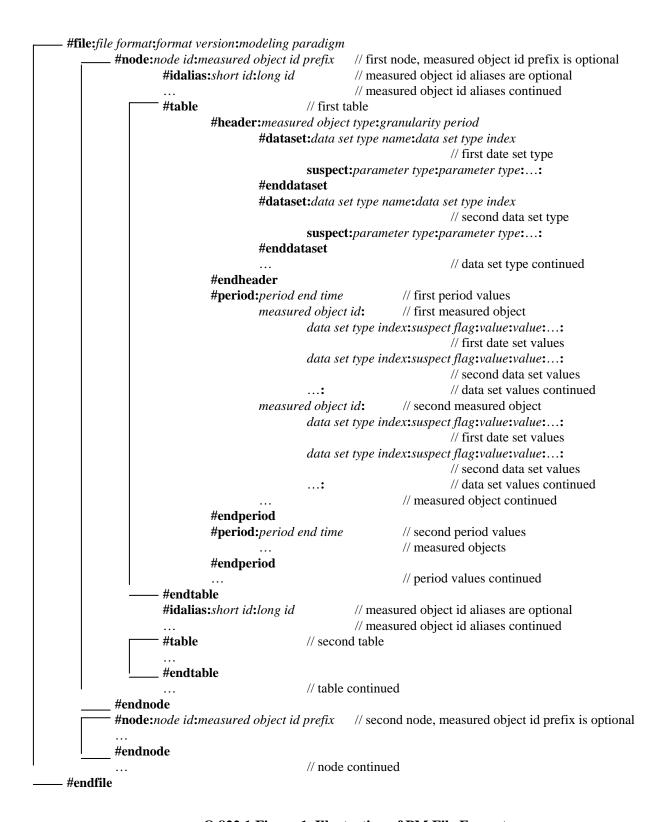
Notes:

1. White space in the file is meaningless. File generator shall not put white space into the file. File parser shall skip and ignore any white space encountered in the file.
2. A backward compatible file parser shall skip and ignore any data between unknown #<tag> and #end<tag>.
3. Extension to this file format shall require new #<tag> definitions.

2.2 Text File Format Description

The history data are contained in performance measurement file. The file consists of a file type definition and a set of measurement data. The order of the data is insignificant. There shall be at least one set of data in the file. Otherwise, file shall not be created.

The performance measurement file shall follow these rules for file generating. (See the figure below which depicts the file structure):

1. The file type is defined by its file format, format version, and modeling paradigm. FileFormat indicates which file format this file follows. FormatVersion specifies a version of particular file format. The value for FileFormat of this recommendation is "q822d1FileText", and FormatVersion is "version1".
2. The modeling paradigm is modeling paradigm of management system that generates the file. If the file is generated from a CORBA management system compliant with ITU-T CORBA framework, the value for ModelParadigm shall be "ITUTCORBA". Otherwise, an appropriate value shall be provided, for instance, "CMIP", "SNMP", "TL1", "abcCORBA", etc.
3. The performance management specification for a specific information modeling domain shall specify the semantics and naming convention of ModelParadigm, NodeID, MeasuredObjectType, DataSetTypeName, ParameterType, ID, MeasuredObjectIDPrefix, MeasuredObjectID.
4. MeasuredObjectIDPrefix defines the common prefix for MeasuredObjectID in a node section.
5. MeasuredObjectIDAlias defines the short alias MeasuredObjectID for long FDN MeasuredObjectID. The scope of an alias definition extends to end of file unless re-defined by another MeasuredObjectIDAlias.
6. DataSetTypeIndex of a DataSetValue indicates the corresponding data set type as defined in the table header.
7. All the measurement data within a table have same granularity period and represent measurements for the same measured object type.
8. If there is no data for a particular period, the file shall still contain #period and #endperiod with empty information for that period.

#file:*file format***:***format version***:***modeling paradigm*
    #node:*node id***:***measured object id prefix*   // first node, measured object id prefix is optional
        #idalias:*short id***:***long id*   // measured object id aliases are optional
        …   // measured object id aliases continued
        #table   // first table
            #header:*measured object type***:***granularity period*
                #dataset:*data set type name***:***data set type index*
                        // first date set type
                    suspect:*parameter type***:***parameter type***:***…***:**
                #enddataset
                #dataset:*data set type name***:***data set type index*
                        // second data set type
                    suspect:*parameter type***:***parameter type***:***…***:**
                #enddataset
                …   // data set type continued
            #endheader
            #period:*period end time*   // first period values
                *measured object id***:**   // first measured object
                        *data set type index***:***suspect flag***:***value***:***value***:***…***:**
                              // first date set values
                        *data set type index***:***suspect flag***:***value***:***value***:***…***:**
                              // second data set values
                    …***:**   // data set values continued
                *measured object id***:**   // second measured object
                        *data set type index***:***suspect flag***:***value***:***value***:***…***:**
                              // first date set values
                        *data set type index***:***suspect flag***:***value***:***value***:***…***:**
                              // second data set values
                    …***:**   // data set values continued
                …   // measured object continued
            #endperiod
            #period:*period end time*   // second period values
                …   // measured objects
            #endperiod
            …   // period values continued
        #endtable
        #idalias:*short id***:***long id*   // measured object id aliases are optional
        …   // measured object id aliases continued
        #table   // second table
        …
        #endtable
        …   // table continued
    #endnode
    #node:*node id***:***measured object id prefix*   // second node, measured object id prefix is optional
    …
    #endnode
    …   // node continued
#endfile

Q.822.1 Figure 1. Illustration of PM File Format

3. DSL PM Bulk Data Conventions

3.1 Information Model Paradigm Name (ModelParadigm)

The information model paradigm name is a string value included in the header section of a bulk performance data file. The information model paradigm name sets the context for the constant values (data set types and performance parameter types) used in the bulk performance data file.

> **R-BULK-CONV-1**   For bulk performance data files subscribing to the Broadband Forum's conventions the information model paradigm (ModelParadigm) name in the header section shall be set to "dslf_pm_v1".

3.2 Data Set Types

The performance data set represents a grouping of DSL performance counters that are commonly gathered together. For DSL, performance data sets include: ADSL Line ATU-C, ADSL Line ATU-R, ADSL Channel ATU-C, and ADSL Channel ATU-R.

Within a bulk performance data file, data sets are identified by a data set name (DataSetTypeName) and a corresponding integer identifier or index (DataSetTypeIndex). For DSL performance data, the each performance data set will be assigned a short text name and an integer identifer.

> **R-BULK-CONV-2** For bulk performance data files subscribing to the Broadband Forum's conventions the individual data sets shall use the assigned **data set type name** and **data set type index:**

Performance Data Set Description	DataSetTypeName	DataSetTypeIndex
ADSL Line ATU-C Performance Data	"ADSLLineAtucCurrentData"	1
ADSL Line ATU-R Performance Data	"ADSLLineAturCurrentData"	2
ADSL Channel ATU-C Performance Data	"ADSLChannelAtucCurrentData"	3
ADSL Channel ATU-R Performance Data	"ADSLChannelAturCurrentData"	4

3.3 Measured Object Types

The measured object type describes the class or type of the containing entity relative to the performance data represented in a table. For DSL, measured object types include: ADSL Line, and ADSL Channel.

Within a bulk performance data file, measured object types are represented in the table header by a string (MeasuredObjectType). For DSL performance data, the each measured object type is assigned a short text name.

> **R-BULK-CONV-3** For bulk performance data files subscribing to the Broadband Forum's conventions the table header shall use the following **measured object types** to describe the type of containing entity for the performance data in the table.

Container Entity Description	MeasuredObjectType
ADSL Line	"ADSLLine"
ADSL Channel	"ADSLChannel"

3.4 Parameter Types

The parameter types describe the individual types of measurements that may be members of a specific performance data set. For example, unavailable seconds.

Within a bulk performance data file, parameter types are represented in the data set description by a string (ParameterType). For DSL performance data, the each parameter type is assigned a short text name.

> **R-BULK-CONV-4** For bulk performance data files subscribing to The Broadband Forum's conventions each parameter type within the data set description shall use one of the following strings to represent the **parameter type.**

Parameter Type Description	ParameterType
Count of loss of frame events	"lossOfFrameEvents"
Count of loss of signal events	"lossOfSignalEvents"
Count of loss of link events	"lossOfLinkEvents"
Count of loss of power events	"lossOfPowerEvents"
Count of errored seconds	"erroredSeconds"

Count of severely errored seconds	"severelyErroredSeconds"
Count of unavailable seconds	"unavailableSeconds"
Count of initialization events	"initializationEvents"
Count of fast retrain attempts	"fastRetrainAttempts"
Count of failed fast retrain attempts	"failedFastRetrainAttempts"
Count of code violations	"codeViolations"
Count of forward error corrections	"forwardErrorCorrections"
Count of forward error correction seconds	"forwardErrorCorrectionSeconds"
Count of loss of signal seconds	"lossOfSignalSeconds"
Count of transmitted blocks	"transmittedBlocks"
Count of received blocks	"receivedBlocks"
Count of corrected blocks	"correctedBlocks"
Count of uncorrected blocks	"uncorrectedBlocks"

3.5 Performance Data Set and Parameter Types Relationship

The parameter types describe the individual types of measurements that may be members of a specific performance data set. The following requirement describes the relationship between performance data sets and parameter types.

R-BULK-CONV-5 For bulk performance data files subscribing to The Broadband Forum's conventions each performance data set shall provide support for the corresponding parameter types identified in the table below.

DataSetTypeName	Member ParameterTypes
"ADSLLineAtucCurrentData"	"lossOfFrameEvents"
	"lossOfSignalEvents"
	"lossOfLinkEvents"
	"lossOfPowerEvents"
	"erroredSeconds"

	"severelyErroredSeconds"
	"unavailableSeconds"
	"initializationEvents"
	"fastRetrainAttempts"
	"failedFastRetrainAttempts"
	"codeViolations"
	"forwardErrorCorrections"
	"forwardErrorCorrectionSeconds"
	"lossOfSignalSeconds"
"ADSLLineAturCurrentData"	"lossOfFrameEvents"
	"lossOfSignalEvents"
	"lossOfLinkEvents"
	"lossOfPowerEvents"
	"erroredSeconds"
	"severelyErroredSeconds"
	"unavailableSeconds"
	"initializationEvents"
	"fastRetrainAttempts"
	"failedFastRetrainAttempts"
	"codeViolations"
	"forwardErrorCorrections"
	"forwardErrorCorrectionSeconds"
	"lossOfSignalSeconds"
"ADSLChannelAtucCurrentData"	"transmittedBlocks"
	"receivedBlocks"
	"correctedBlocks"
	"uncorrectedBlocks"

"ADSLChannelAturCurrentData"	"transmittedBlocks"
	"receivedBlocks"
	"correctedBlocks"
	"uncorrectedBlocks"

4. DSL Specific Bulk PM Data File Example

The following is an example of a simple instance of a PM bulk data file following the proposed Broadband Forum Conventions.

```
#file: q822d1FileText: version1:dslf_pm_v1
    #node:BLTMM018BB1:node1          // first node, measured object id prefix is optional
        #table                        // first table
            #header: ADSLChannel:15 minutes
                #dataset: ADSLChannelAtucCurrentData:3
                                                    // first date set type
                    suspect:transmittedBlocks:receivedBlocks:
                            correctedBlocks:uncorrectedBlocks:
                #enddataset
                #dataset: ADSLChannelAturCurrentData:4
                                                    // second data set type
                    suspect:transmittedBlocks:receivedBlocks:
                            correctedBlocks:uncorrectedBlocks:
                #enddataset
                …                                   // data set type continued
            #endheader
            #period: 20010723091500.000Z           // first period values

            ManagedElementId=Node1*Bay=B*Shelf=C*CircuitPack=D*Port=E:
                            // first measured object
                    3:F: 51273:4258:14:2:
                                                    // first date set values
                    4:F:4258:51273:6:0:
                                                    // second data set values
                    …:                             // data set values continued
            ManagedElementId=Node1*Bay=B*Shelf=C*CircuitPack=D*Port=G:
                    // second measured object
                    3:F: 34273:2258:14:2:
                                                    // first date set values
                    4:F: 2258:34273:6:0:
                                                    // second data set values
                    …:                             // data set values continued
                …                                   // measured object continued
            #endperiod
        #endtable
    #endnode
#endfile
```