

Pierre Auger Project: Big Brother Interface

Version 1.1 19st September 2005

Author: V. Tunnicliffe
Change control & document ownership: V. Tunnicliffe

Updates : V. Tunnicliffe to v1.0 – First draft
V. Tunnicliffe to v1.1 – Added backbone network ping information

Distribution: P. D. J. Clark <p.d.j.clark@dspcomms.karoo.co.uk>
V. Tunnicliffe <V.Tunnicliffe@iee.org>
Pete Walker <p.walker@leeds.ac.uk>

The Institute of Integrated Information Systems
School of Electrical and Electronic Engineering
The University of Leeds
Woodhouse Lane
Leeds
LS2 9JT
UK

1. Table of Contents

Pierre Auger Project: Big Brother Interface	1
1. Table of Contents	2
2. Introduction.....	2
3. File Generation and Naming Conventions.....	2
4. File Format.....	3
5. Backbone Network Availability Information	3

2. Introduction

This document describes format of the files that are produced by the communications system's network monitoring software. These files are intended to be used by Auger's "big brother" monitoring system for displaying the status of the communications system in near-real-time on the world wide web.

3. File Generation and Naming Conventions.

The network monitoring software (NETMON) generates monitoring files and stores them in the C:\Netlog\Reports directory :-

- 1) At 15 minute intervals from when the NETMON software was started.
- 2) Daily, from when the NETMON software was started.

The files that are generated every 15 minutes are named *dd_hh.mm.rep*, where *dd*, *hh* and *mm* represent the day, hour and minute of when the file was created respectively. The minute field may take on the value of 00, 15, 30 or 45 depending on when the file was created within the hour. For instance, if the NETMON software was started at 10.03am, a file will be automatically created at 10.18am named *04_10.15.rep* and will contain the monitoring data obtained during the period 10.03am to 10.18am.

A file is also generated daily and is named *xxx.rep*, where *xxx* is the day of the year expressed as a decimal number in the range 001 to 366. The file will be generated 24 hours from the time netmon was started.

All monitoring files are stored in the C:\Netlog\Reports directory on the network monitoring PC. To remove the need of maintenance of NETMON's report files, old files are automatically deleted; for daily generated files, last year's file is deleted and for 15 minute average files, the previous day's file is deleted before the new file is created.

The logging of the monitoring files may be stopped by removing the check from the *Big Brother File Generation* radio button that appears on the *Communication System Performance* page of the NETMON software.

4. File Format.

The monitoring files generated every 15 minutes and the files generated daily follow an identical format. Each line in the file gives a tab-separated list of the operating parameters of each subscriber unit that is connected to the radio network. The format of this data is shown in Figure 1.

Subscriber ID	Parent Base-Station ID	Average Signal Strength (dBm)	Packet Count	Number of ARQ Requests
---------------	------------------------	-------------------------------	--------------	------------------------

Figure 1 - Monitoring Data Format

The *subscriber ID* identifies the serial number of the unit. The *parent base-station ID* shows which base-station is servicing the subscriber. The *average signal strength* field gives the average signal strength measured on the radio downlink over the measurement period. The *packet counter* field identifies the number of monitoring packets received from the subscriber unit and the *number of ARQ requests* field identifies the number of packets the BSU has re-requested because of an on-air packet failure.

All number formats are integers apart from the average signal strength field, which is a floating point number in 3.2 format. The following C style printf format specifier was used to write the data to the file:-

```
"%4d\t%2d\t%3.2f\t%7d\t%7d\n"
```

This is taken from the program line :-

```
OutputString.Format("%4d\t%2d\t%3.2f\t%7d\t%7d\n", pSU->GetSUSationID(),
    pSU->GetParentBaseStation()->GetBSUSationID(),
    pSU->Get15MinSigStrengthAverage(),
    pSU->Get15MinPacketCounter(),
    pSU->Get15MinARQPacketCounter());
```

5. Backbone Network Availability Information

On the 19th September 2005, the NETMON software was updated (version 3.20) to include information about the status of the backbone network. NETMON now uses the ICMP echo packet (ping) to test the availability of the backbone network by pinging each of the routers at the tower sites. In an identical manner to the way in which the signal strength and uplink packet count reports are generated and described in section 3, the software now also generates a “ping” packet count listing file. The files generated follow exactly the same naming conventions as described in section 3, with the “.rep” file extension being replaced with “.ping”. Please note that a “daily” file is not generated.

Every 10 seconds a new line of information will be written to the file for each of the 4 routers at the tower sites. Each line within the file provides the following information :-

Router ID (1 to 4)	Number of Ping Transmissions	Number of Ping Replies	Min Reply Time (ms)	Max. Reply Time (ms)	Average Reply Time (ms)
-----------------------	---------------------------------	---------------------------	---------------------------	----------------------------	----------------------------------

Figure 2 - Backbone Network Availability Format

The *Router ID* field may take on the value of 1, 2, 3, or 4, which corresponds to the routers at Los Leones, Los Morados, Loma Amarilla and Coihueco respectively. The *Number of Ping Transmissions* represents the number of ping requests sent by NETMON to the specified router during the 10 second measurement period. The *Number of Ping Replies* field shows how many ping replies were received by NETMON. The *Min Reply Time*, *Max. Reply Time* and *Average Reply Time* fields represent the minimum, maximum and average transit times for the measurement period of 10 seconds.

All numbers formats are integers apart from the *Average Reply Time* field, which is a floating point number in 5.2 format. The following C style printf format specifier was used to write the data to the file:-

```
Message.Format("%2d - %3d\t%3d\t%3d\t%5.2f\t\n", i+1,  
m_pPingMonitor->GetPtrToCPingArray()[i]->GetTxPacketCount(),  
m_pPingMonitor->GetPtrToCPingArray()[i]->GetRxPacketCount(),  
m_pPingMonitor->GetPtrToCPingArray()[i]->GetMinDelay(),  
m_pPingMonitor->GetPtrToCPingArray()[i]->GetMaxDelay(),  
m_pPingMonitor->GetPtrToCPingArray()[i]->GetAverageDelay());
```