BTSLOG
1.5-0
Tutorial

1. INTRODUCTION

This tutorial gives you instructions how to get started with BTSLOG and its features.
The main features of BTSLOG are:

- Tracing WCDMA and LTX nodes and recording message trace files in binary file format (*.bin)

- Recording system logs in ascii file format (*.log)

- Displaying run-time data of monitored nodes:
    - Trace state
    - Number of traced messages
    - Number of lost messages (only for WCDMA node)
    - Ping state
    - Ping time
    - Peak kbps
    - Current kbps

- Displaying run-time data of received system logs:
    - Number of recorded system logs
    - Number of lost system logs

- Viewing system logs in Real Time Views where you can:
    - Filter system logs by setting include and exclude filters
    - Select nodes from which system logs are displayed
    - Define watch texts for incoming system logs
    - Open a snapshot of Real Time Log - window content by double-clicking the grid lines in the view

- Defining filters for WCDMA nodes in BTS filters - window

- Defining filters for LTX nodes in LTX filters - window

- Configuring setup information for log file handling, tracing and windows

- Viewing recorded message trace files by opening Ida2 using shortcut from menu Tools-> Open in Ida2 or via dedicated Ida2 -button from main window

- Viewing recorded system logs using user defined filters

- Setting and getting R&D Parameters

- Fetching RM status tables from nodes (WN4 support only)

2. CONFIGURING BTSLOG

2.1 Installation

BTSlog package is msi - The windows installer. It is recommended to remove previous installation before installing a new version.

2.2 Command line arguments

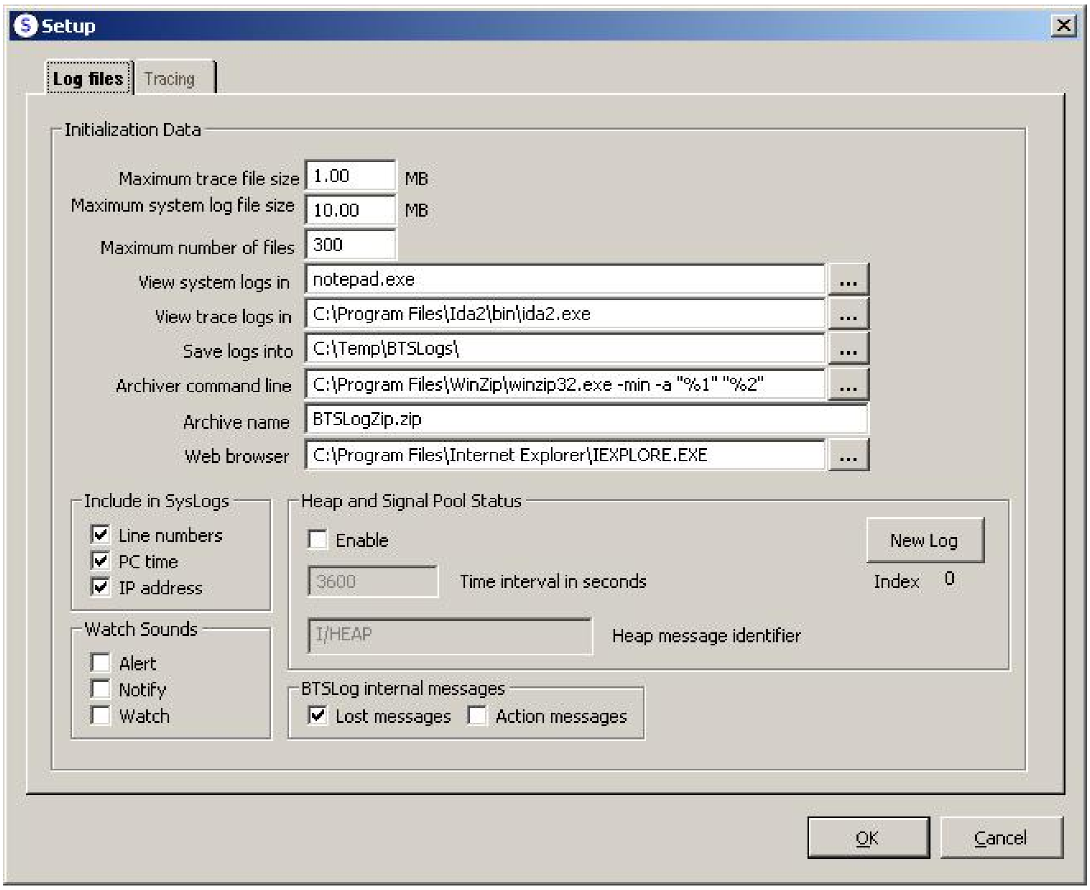
BTSlog can also start with following command line arguments:

```
start_bssigrec
start_udplog
start_both
```

If further external commanding is needed, BTSLog_launcher program is able send start & stop & clear messages to BTSlog. BTSlog can be this way started and stopped with any scripting language, which is capable of executing programs. More info in C:\Program Files\BTSlog\BtsLogLauncherTutorial.txt.

2.3 Configuring Setup information

1. Select the Tools->Setup command (or press F4).
2. Check the information on Log files -tab.



Initialization Data

Maximum trace file size 1.00 MB
Maximum system log file size 10.00 MB
Maximum number of files 300
View system logs in notepad.exe
View trace logs in C:\Program Files\Ida2\bin\ida2.exe
Save logs into C:\Temp\BTSLogs\
Archiver command line C:\Program Files\WinZip\winzip32.exe -min -a "%1" "%2"
Archive name BTSLogZip.zip
Web browser C:\Program Files\Internet Explorer\IEXPLORE.EXE

Include in SysLogs
☑ Line numbers
☑ PC time
☑ IP address

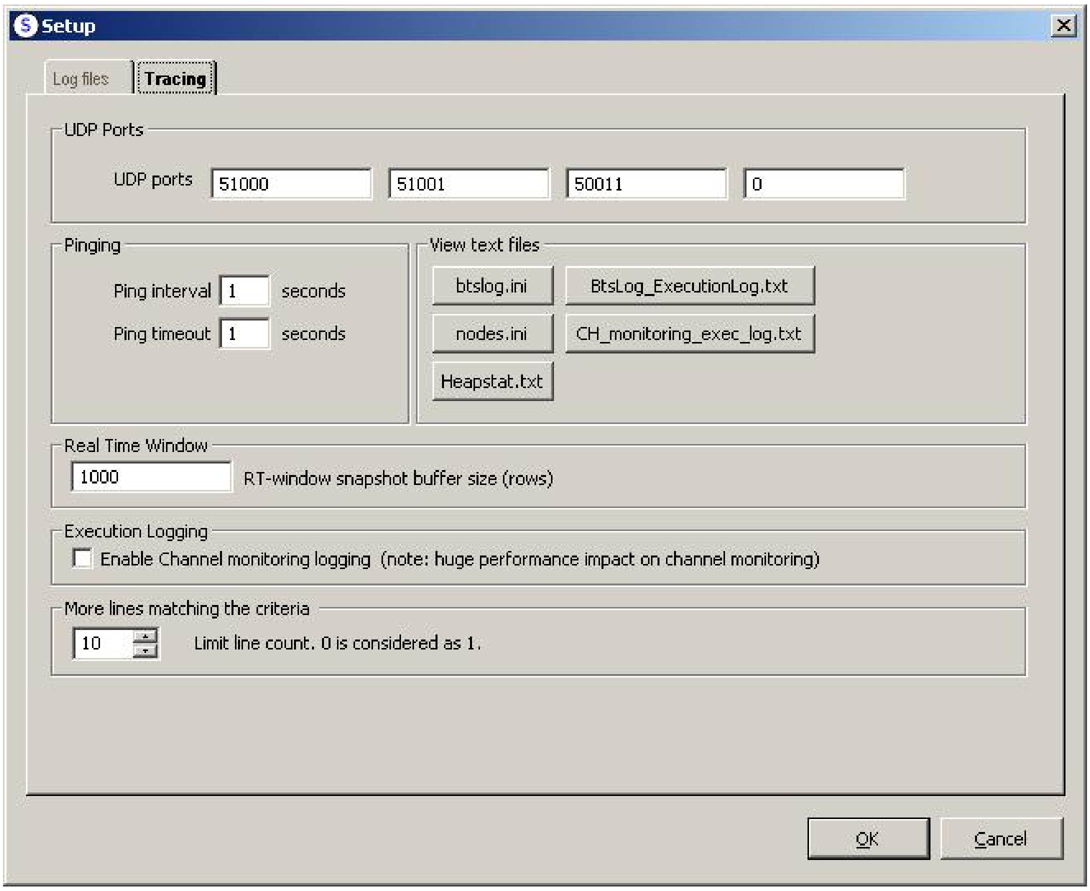
Watch Sounds
☐ Alert
☐ Notify
☐ Watch

Heap and Signal Pool Status
☐ Enable
3600 Time interval in seconds
Index 0
I/HEAP Heap message identifier

BTSLog internal messages
☑ Lost messages ☐ Action messages

Remarks:
All log files (including trace logs and system logs) will be saved into the directory given in **Save logs into** edit box.
Log files can be archived using an external archiver utility. BTSLOG will copy log files going to be archived into a temporary directory, and then it will call add one by one each log file from temporary directory to archive. **Archiver command line** edit box contains the command line used to add one file to archive. **%1** contains the full full path of the archive (output) file, **%2** contains the full path of the (log)file going to be added to the archive.
BTSLOG will add each of log files, one by one to archive.
The resulted archive file will be saved into the same directory like log files. The name of this archive file is given by **Archive name** edit box.

3. Check the information on Tracing –tab



UDP Ports
UDP ports 51000 51001 50011 0

Pinging
Ping interval 1 seconds
Ping timeout 1 seconds

Real Time Window
1000 RT-window snapshot buffer size (rows)

Execution Logging
□ Enable Channel monitoring logging (note: huge performance impact on channel monitoring)

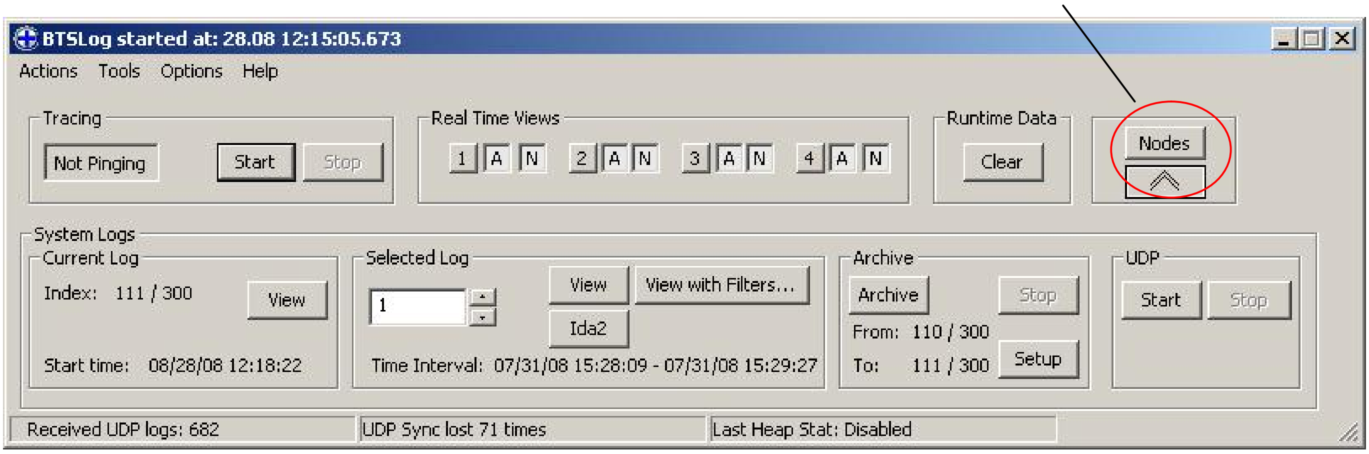
More lines matching the criteria
10 Limit line count. 0 is considered as 1.

4. Change the default values in setup - window if needed. You changes are automatically saved when OK- button is pressed.

3. TRACING WCDMA AND LTX NODES

3.1 Selecting nodes for monitoring

Activate the Nodes and Filters window. If the window is not visible press Nodes button in BTSLOG main window.



BTSLog started at: 28.08 12:15:05.673

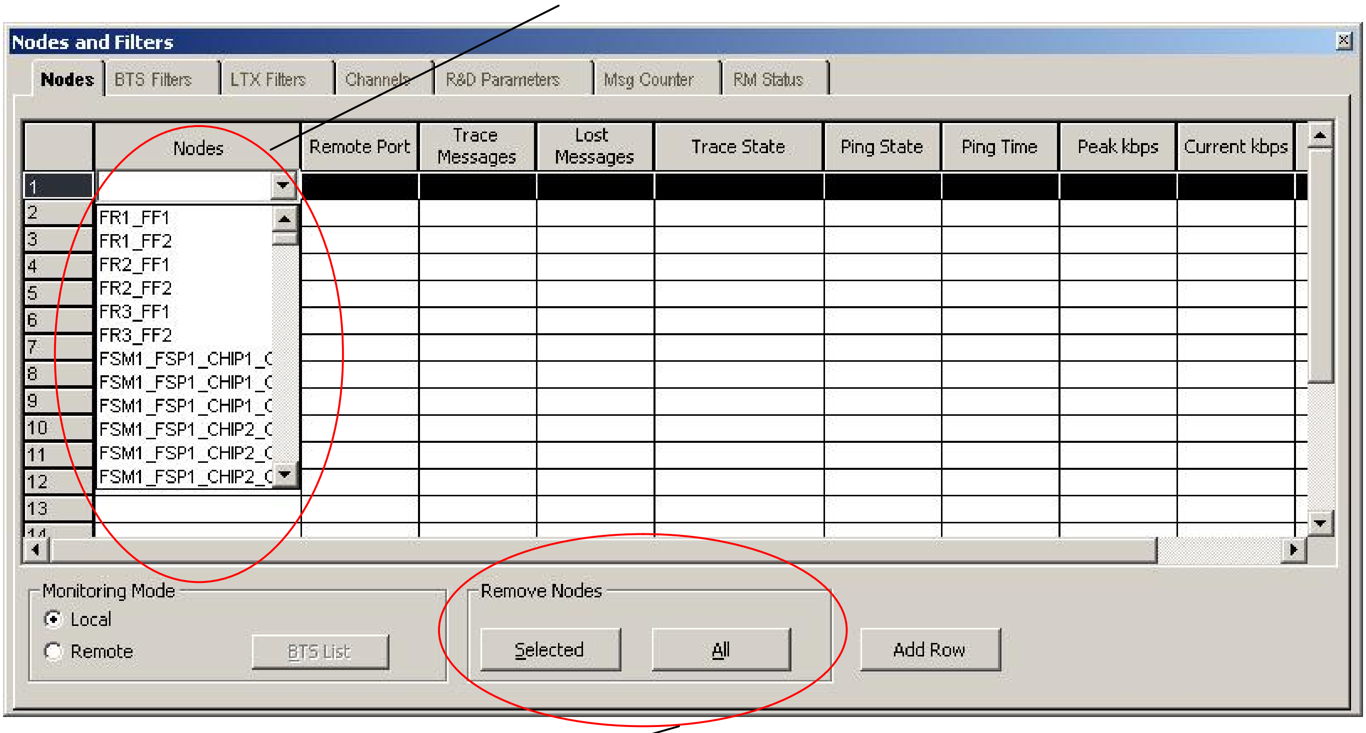
Current Log — Index: 111 / 300 — Start time: 08/28/08 12:18:22

Selected Log: 1 — Time Interval: 07/31/08 15:28:09 - 07/31/08 15:29:27

Archive — From: 110 / 300 — To: 111 / 300

Received UDP logs: 682 | UDP Sync lost 71 times | Last Heap Stat: Disabled

Select the nodes to be monitored from Nodes column drop-down list.



To remove a node from the grid:

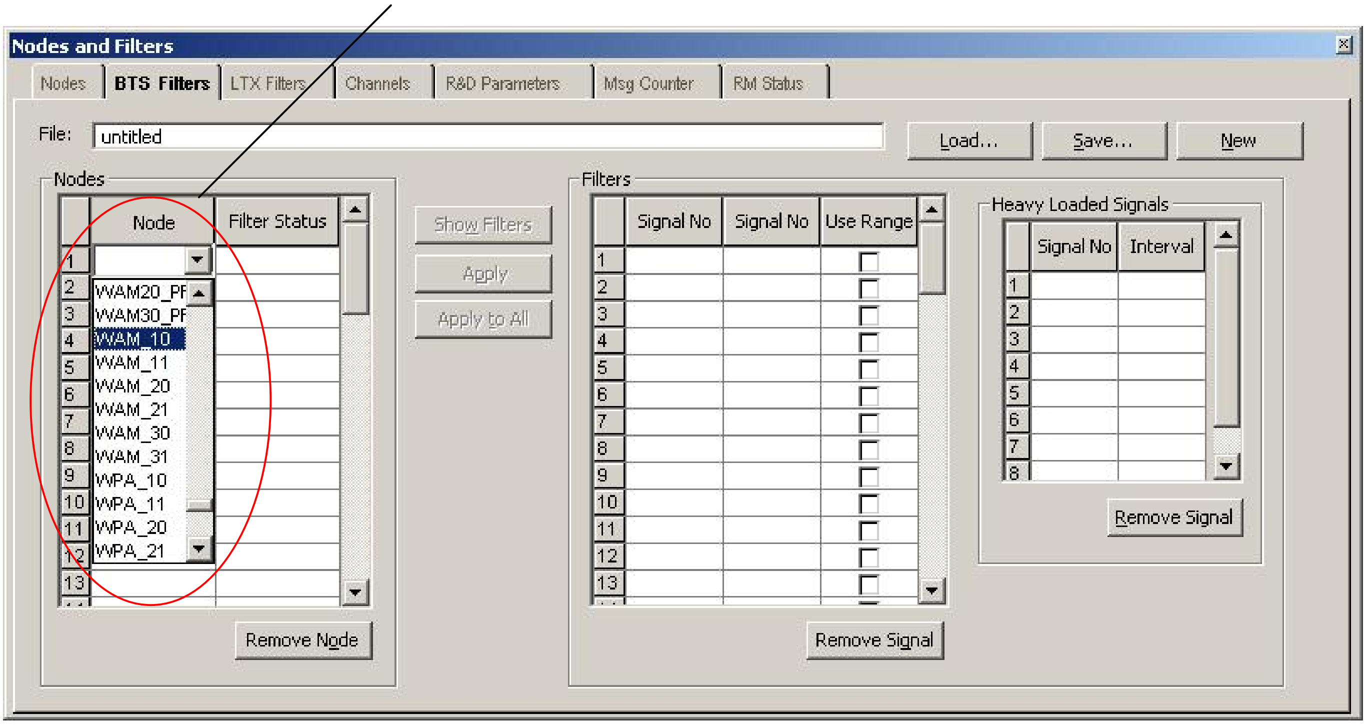
1. Select the row in the nodes grid by clicking it
2. Press Remove Selected button, or Remove all

3.2 Defining filters for nodes

3.2.1 Defining BTS Filters

Activate BTS filters tab in Nodes and Filters window.
To define a filter for a node:

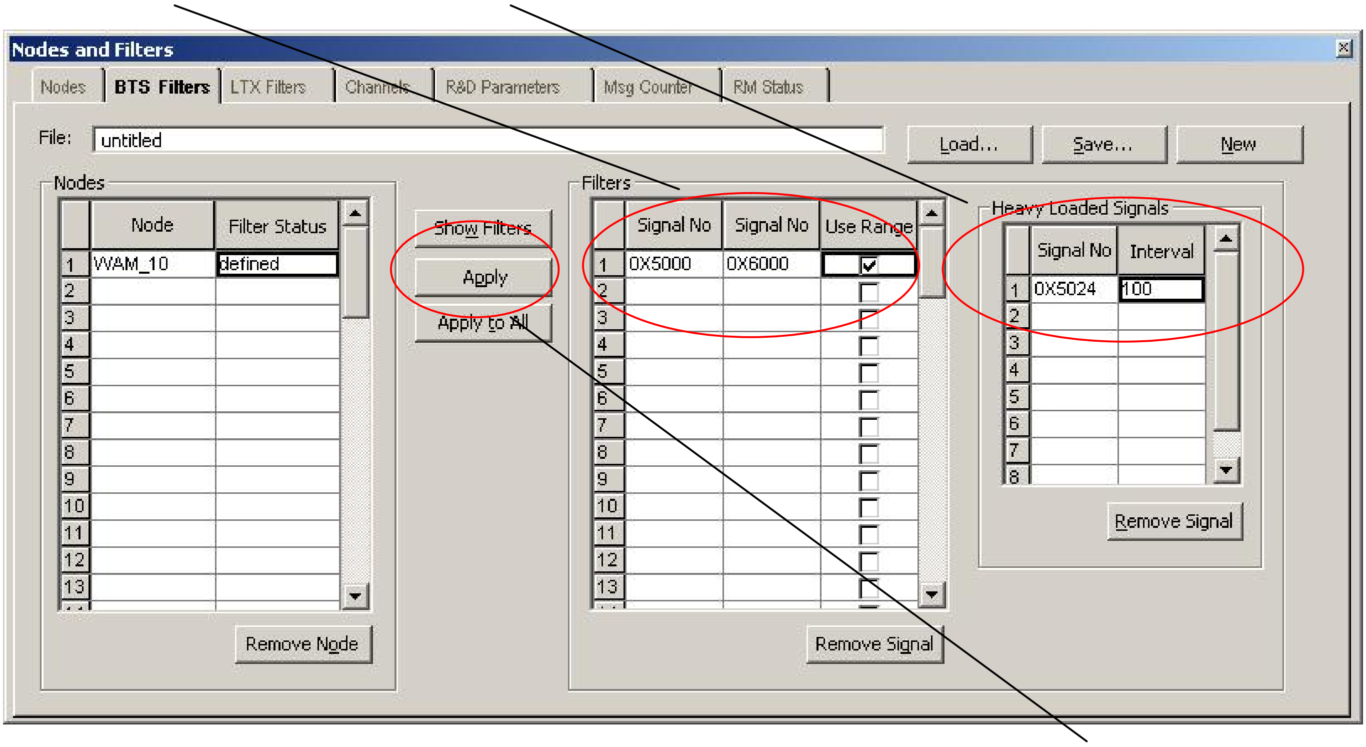
1. Select the nodes from Nodes column drop-down list.



Nodes and Filters

File: untitled

2. Define filters and heavy loaded signals for the selected node



File: untitled

Nodes

	Node	Filter Status
1	WAM_10	defined
2		
3		
4		
5		
6		
7		
8		
9		
10		
11		
12		
13		

Filters

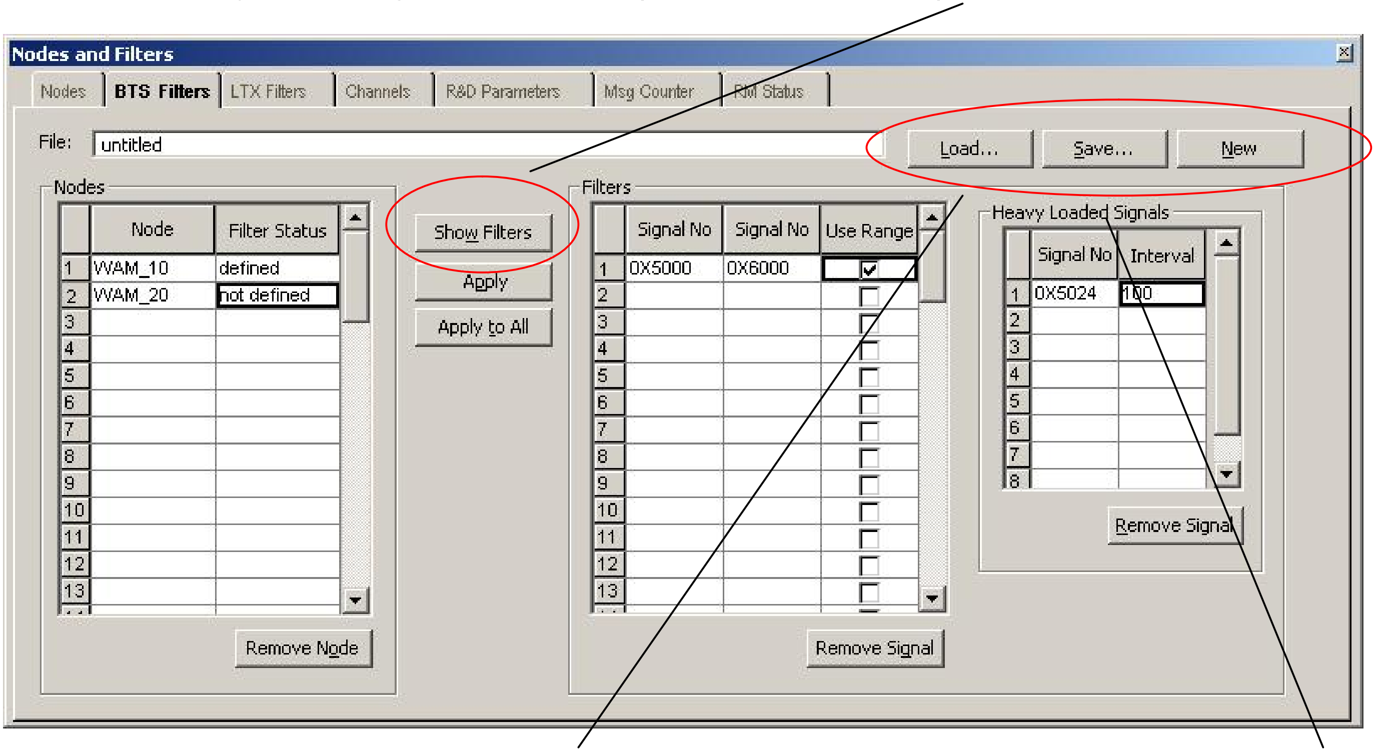
	Signal No	Signal No	Use Range
1	0X5000	0X6000	☑
2			☐
3			☐
4			☐
5			☐
6			☐
7			☐
8			☐
9			☐
10			☐
11			☐
12			☐
13			☐

Heavy Loaded Signals

	Signal No	Interval
1	0X5024	100
2		
3		
4		
5		
6		
7		
8		

3. Apply the filters and heavy loaded signals for the selected node by pressing Apply-button. Now the node filters status is changed from undefined to defined.

If you want to view filters for a single node, select the node and press Show Filters-button. The Filters and Heavy Loaded Signals - window is updated with the node specific filter.



File: untitled

Nodes

	Node	Filter Status
1	WAM_10	defined
2	WAM_20	not defined
3		
4		
5		
6		
7		
8		
9		
10		
11		
12		
13		

Filters

	Signal No	Signal No	Use Range
1	0X5000	0X6000	☑
2			☐
3			☐
4			☐
5			☐
6			☐
7			☐
8			☐
9			☐
10			☐
11			☐
12			☐
13			☐

Heavy Loaded Signals

	Signal No	Interval
1	0X5024	100
2		
3		
4		
5		
6		
7		
8		

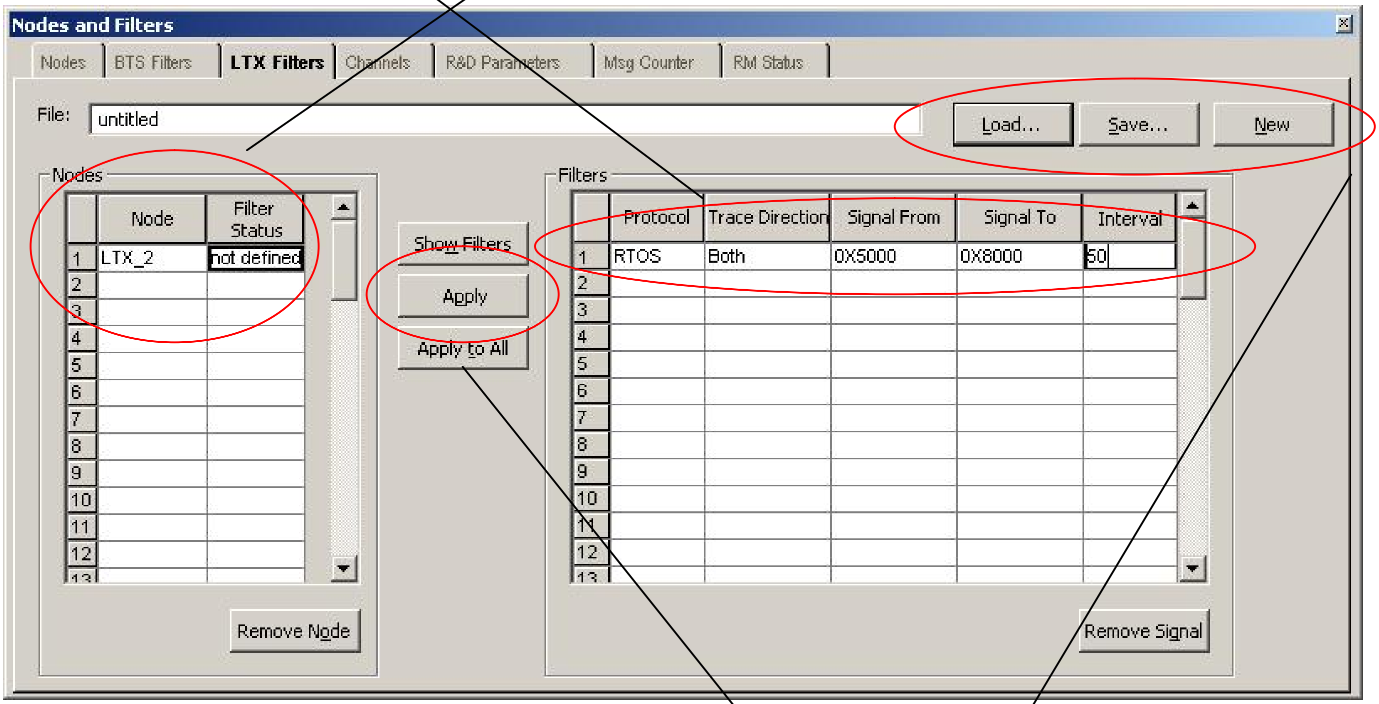
You can save the filter by pressing Save – button. You can load an existing filter by pressing Load – button. If you want to automatically load the filters on next BTSLog session, save the filters. You can start configuring a new filter by pressing New – button.

3.2.2 Defining LTX Filters

Activate LTX filters tab in Nodes and Filters window.
To define a filter for a node:

1. Select the nodes from Nodes column drop-down list.

2. Define filters for the selected node.



File: untitled

	Node	Filter Status
1	LTX_2	not defined
2		
3		
4		
5		
6		
7		
8		
9		
10		
11		
12		
13		

	Protocol	Trace Direction	Signal From	Signal To	Interval
1	RTOS	Both	0X5000	0X8000	50
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					

3. Apply the filters for the selected node by pressing Apply-button.

Now the node filters status is changed from undefined to defined.

Tips:
If you have many nodes in Nodes grid you can apply the filters window content to all nodes by pressing Apply to All-button.

If you want to view filters for a single node, select the node and press Show Filters-button. The Filters window is updated with the node specific filter.

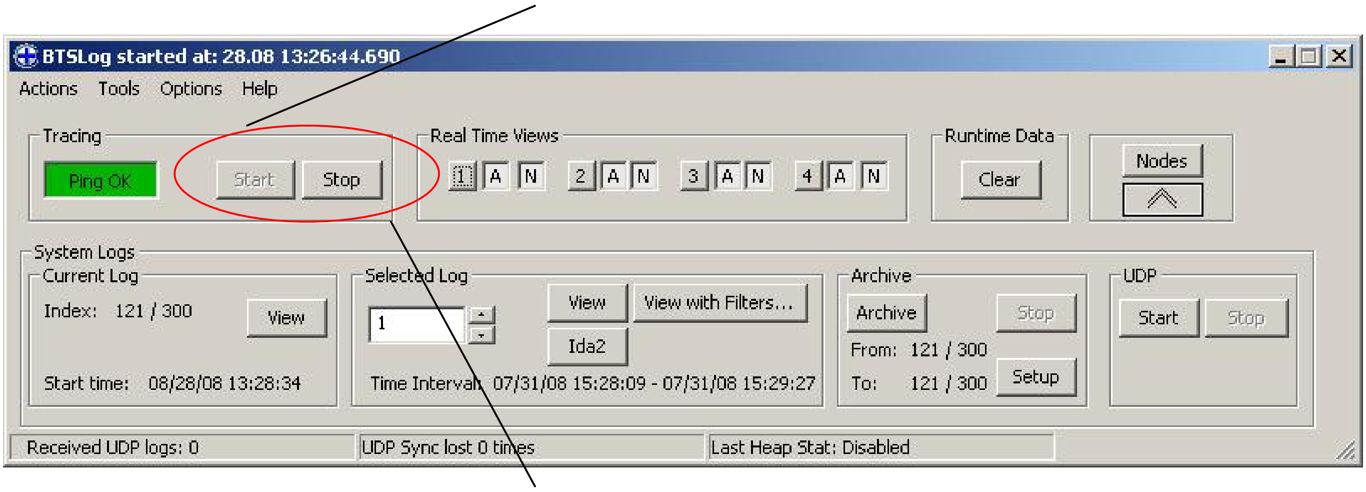
You can save the filter by pressing Save – button. You can load an existing filter by pressing Load – button. You can start configuring a new filter by pressing New – button.

3.3 Starting trace

Before starting trace, check that you have:

- Inspected setup information
- Selected nodes to be monitored
- Defined filters for selected nodes

Tracing will be started by pressing Start-button of BTSLOG main window.



Index: 121 / 300

Start time: 08/28/08 13:28:34

1

Time Interval: 07/31/08 15:28:09 - 07/31/08 15:29:27

From: 121 / 300

To: 121 / 300

Received UDP logs: 0

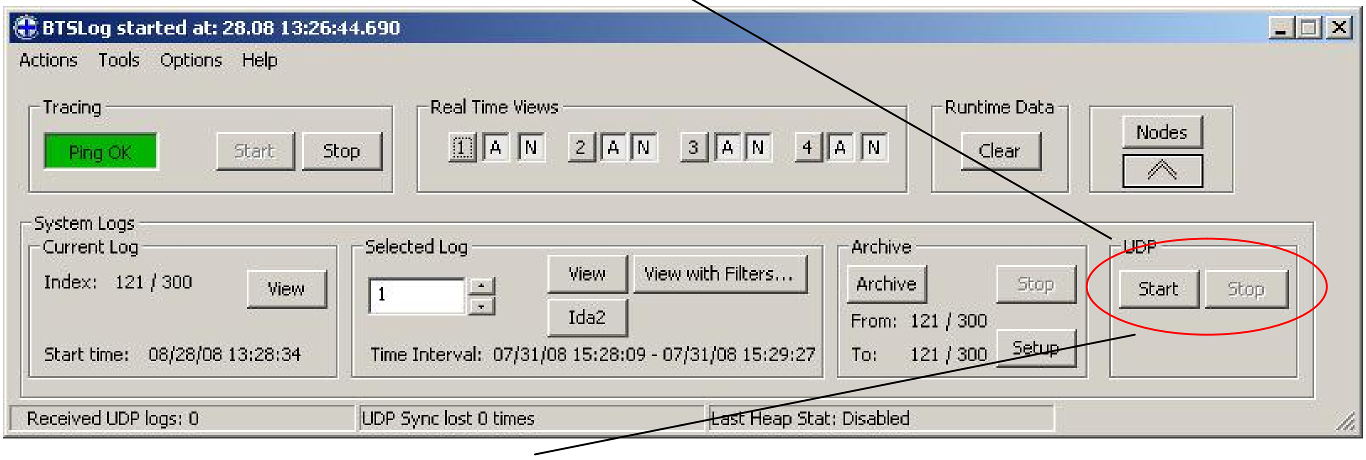
UDP Sync lost 0 times

Last Heap Stat: Disabled

Tracing will be stopped by pressing Stop-button of BTSLOG main window.
Tracing run-time data for selected nodes can be viewed in the Nodes and Filters window.

4. RECORDING AND VIEWING SYSTEM LOGS

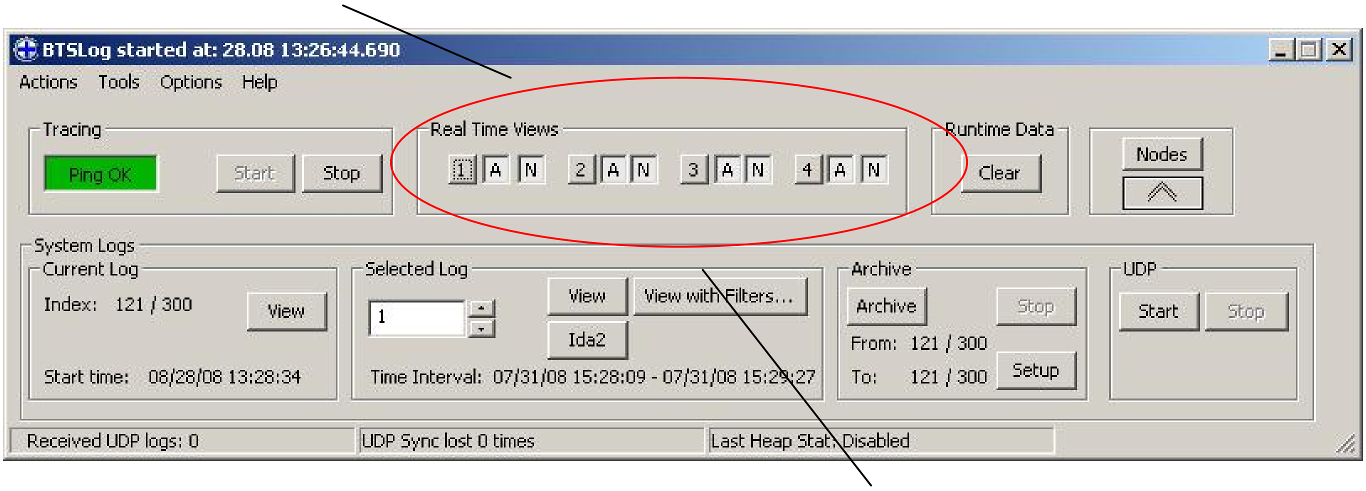
To start recording system logs press the Start -button in UDP section. All received system logs are recorded to ASCII text files.



To stop recording system logs press Stop -button.

4.1 Configuring Real Time Views

There are four Real Time views in BTSLOG for viewing system logs real time.

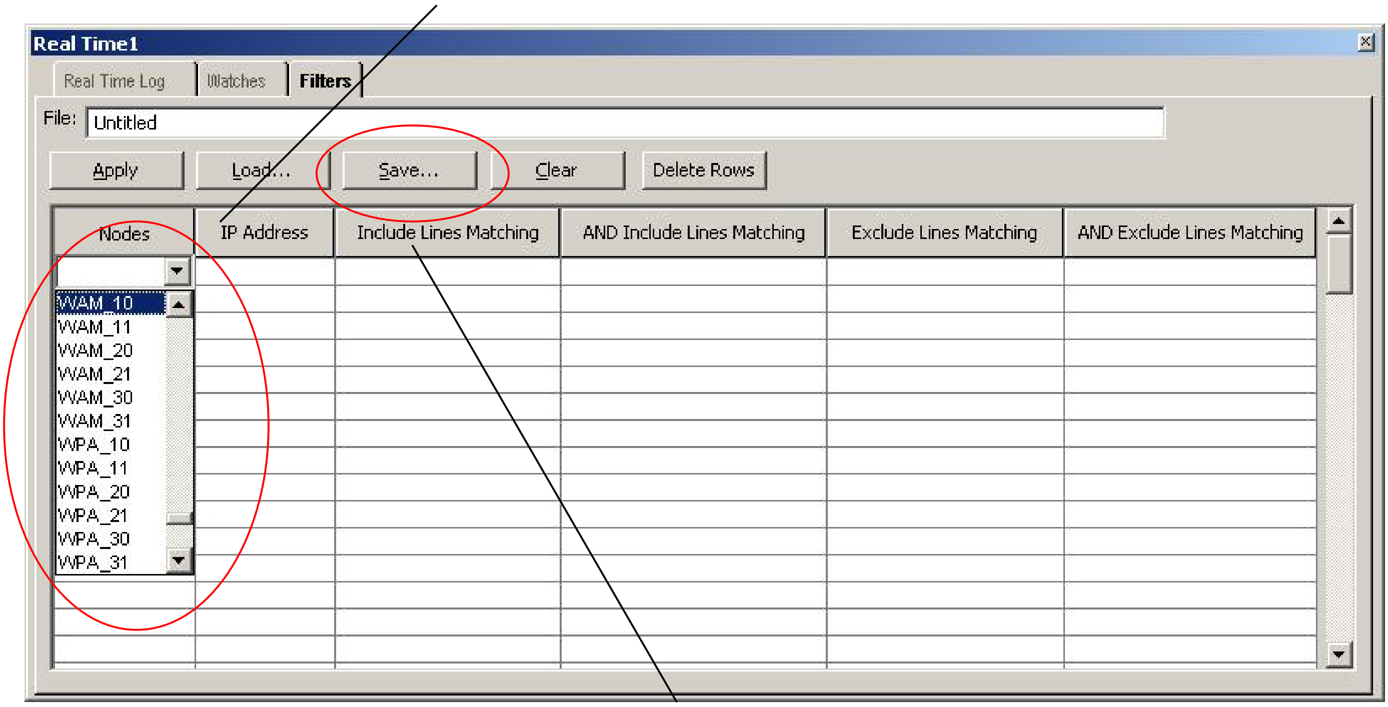


To show the Real Time view press the View -button you want to show (1, 2, 3 or 4)

4.1.1 Defining filters on Filters tab

You can define view filters for the system logs on RealTime View's Filters tab.

1. Select the nodes from Nodes column you want to view system logs from in this view. If you do select any node, by default all system logs are shown in the view.



2. Define Include and Exclude filters for system log lines. A received system log line is shown in the Real Time Log – window :
   - if it matches the include filters and
   - if it does NOT match the exclude filters

3. You can save the Real Time View filter by pressing Save-button.

Note: Real Time Filters are only filters for view content and they have not any affect on how logs are stored.

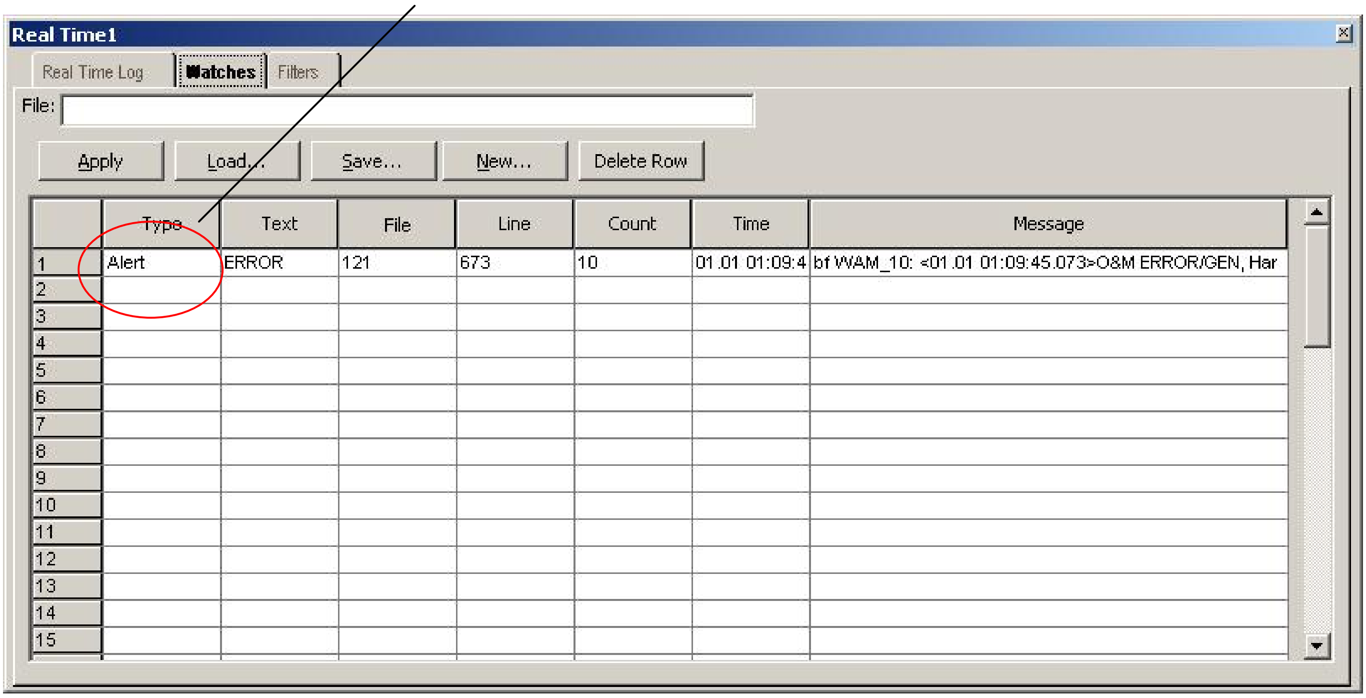
4.1.1.1 Modifying filters at run-time

- You can clear all the filters from the window by pressing Clear Filters- button .
- You can modify existing filters and take them in use on fly by pressing Apply-button.

4.1.2 Defining watches on Watches tab

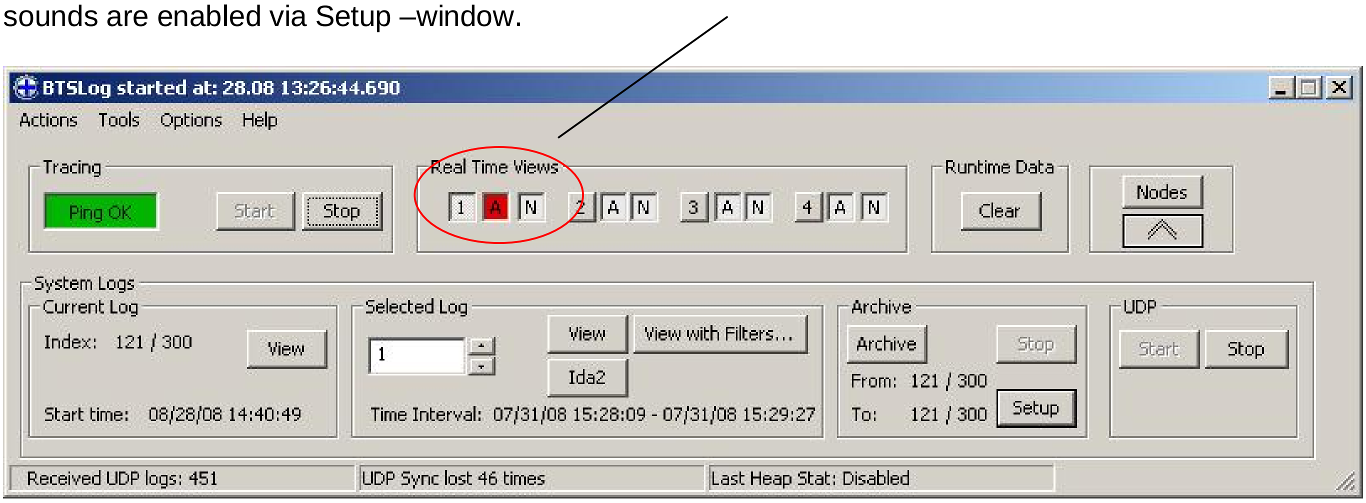
To define text to be searched from incoming system log lines you can define watches of three types:

- Alert (lines including alert texts are marked with read color in Real Time Log window)
- Notify (lines including notify texts are marked with green color in Real Time Log window)
- Watch (lines including watch texts are marked with blue color in Real Time Log window)



	Type	Text	File	Line	Count	Time	Message
1	Alert	ERROR	121	673	10	01.01 01:09:4	bf WAM_10: <01.01 01:09:45.073>O&M ERROR/GEN, Har
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							

When a watch is found at run-time from incoming system log line the Watches window File-, Line-, Time-, and Message-column are updated with the information of the last found watch information.
Also a notification is shown in the main window for Alert and Notify and watch sound are played if sounds are enabled via Setup –window.



Index: 121 / 300

Start time: 08/28/08 14:40:49

1

Time Interval: 07/31/08 15:28:09 - 07/31/08 15:29:27

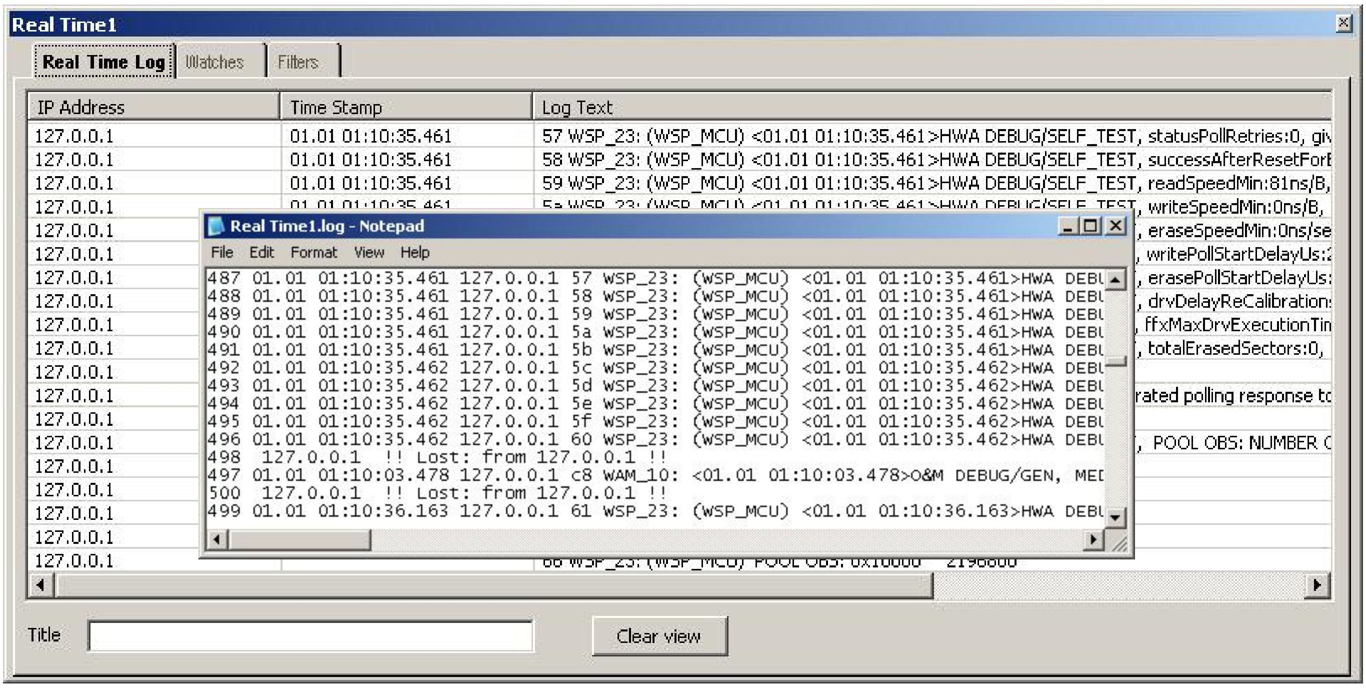
From: 121 / 300

To: 121 / 300

Received UDP logs: 451 | UDP Sync lost 46 times | Last Heap Stat: Disabled

4.1.3 Viewing Real Time Log window

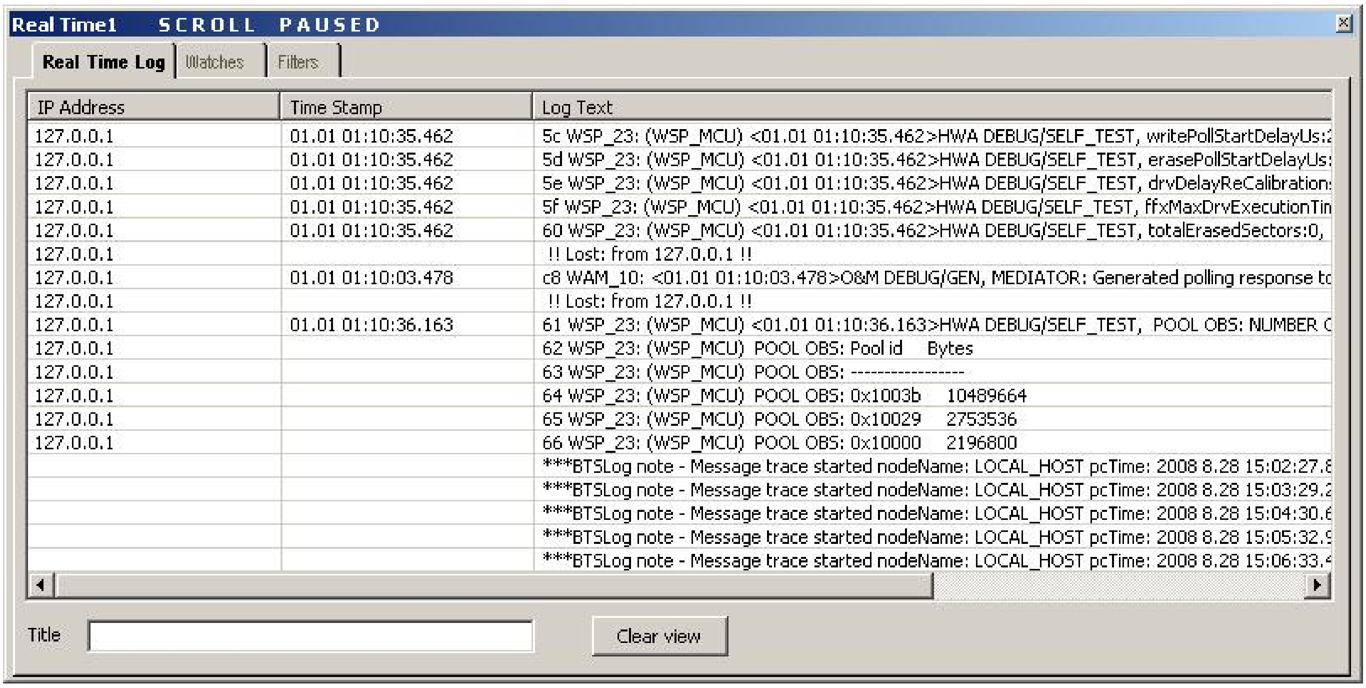
You can view a snapshot of the view including eg. last 1000 lines by double clicking the grid lines. User has possibility to configure snapshot line number via Setup –window.



IP Address	Time Stamp	Log Text
127.0.0.1	01.01 01:10:35.461	57 WSP_23: (WSP_MCU) <01.01 01:10:35.461>HWA DEBUG/SELF_TEST, statusPollRetries:0, giv
127.0.0.1	01.01 01:10:35.461	58 WSP_23: (WSP_MCU) <01.01 01:10:35.461>HWA DEBUG/SELF_TEST, successAfterResetForI
127.0.0.1	01.01 01:10:35.461	59 WSP_23: (WSP_MCU) <01.01 01:10:35.461>HWA DEBUG/SELF_TEST, readSpeedMin:81ns/B,
127.0.0.1	01.01 01:10:35.461	5a WSP_23: (WSP_MCU) <01.01 01:10:35.461>HWA DEBUG/SELF_TEST, writeSpeedMin:0ns/B,
127.0.0.1		, eraseSpeedMin:0ns/se
127.0.0.1		, writePollStartDelayUs:2
127.0.0.1		, erasePollStartDelayUs:
127.0.0.1		, drvDelayReCalibration:
127.0.0.1		, ffxMaxDrvExecutionTin
127.0.0.1		, totalErasedSectors:0,
127.0.0.1		
127.0.0.1		rated polling response tc
127.0.0.1		
127.0.0.1		, POOL OBS: NUMBER C
127.0.0.1		
127.0.0.1		
127.0.0.1		
127.0.0.1		
127.0.0.1		66 WSP_23: (WSP_MCU) POOL OBS: 0x10000 2196800

```
487 01.01 01:10:35.461 127.0.0.1 57 WSP_23: (WSP_MCU) <01.01 01:10:35.461>HWA DEBU
488 01.01 01:10:35.461 127.0.0.1 58 WSP_23: (WSP_MCU) <01.01 01:10:35.461>HWA DEBU
489 01.01 01:10:35.461 127.0.0.1 59 WSP_23: (WSP_MCU) <01.01 01:10:35.461>HWA DEBU
490 01.01 01:10:35.461 127.0.0.1 5a WSP_23: (WSP_MCU) <01.01 01:10:35.461>HWA DEBU
491 01.01 01:10:35.461 127.0.0.1 5b WSP_23: (WSP_MCU) <01.01 01:10:35.461>HWA DEBU
492 01.01 01:10:35.462 127.0.0.1 5c WSP_23: (WSP_MCU) <01.01 01:10:35.462>HWA DEBU
493 01.01 01:10:35.462 127.0.0.1 5d WSP_23: (WSP_MCU) <01.01 01:10:35.462>HWA DEBU
494 01.01 01:10:35.462 127.0.0.1 5e WSP_23: (WSP_MCU) <01.01 01:10:35.462>HWA DEBU
495 01.01 01:10:35.462 127.0.0.1 5f WSP_23: (WSP_MCU) <01.01 01:10:35.462>HWA DEBU
496 01.01 01:10:35.462 127.0.0.1 60 WSP_23: (WSP_MCU) <01.01 01:10:35.462>HWA DEBU
498  127.0.0.1  !! Lost: from 127.0.0.1 !!
497 01.01 01:10:03.478 127.0.0.1 c8 WAM_10: <01.01 01:10:03.478>O&M DEBUG/GEN, MED
500  127.0.0.1  !! Lost: from 127.0.0.1 !!
499 01.01 01:10:36.163 127.0.0.1 61 WSP_23: (WSP_MCU) <01.01 01:10:36.163>HWA DEBU
```

Tip: You can halt the scrolling in RT-view by clicking the view. And release by clicking again.

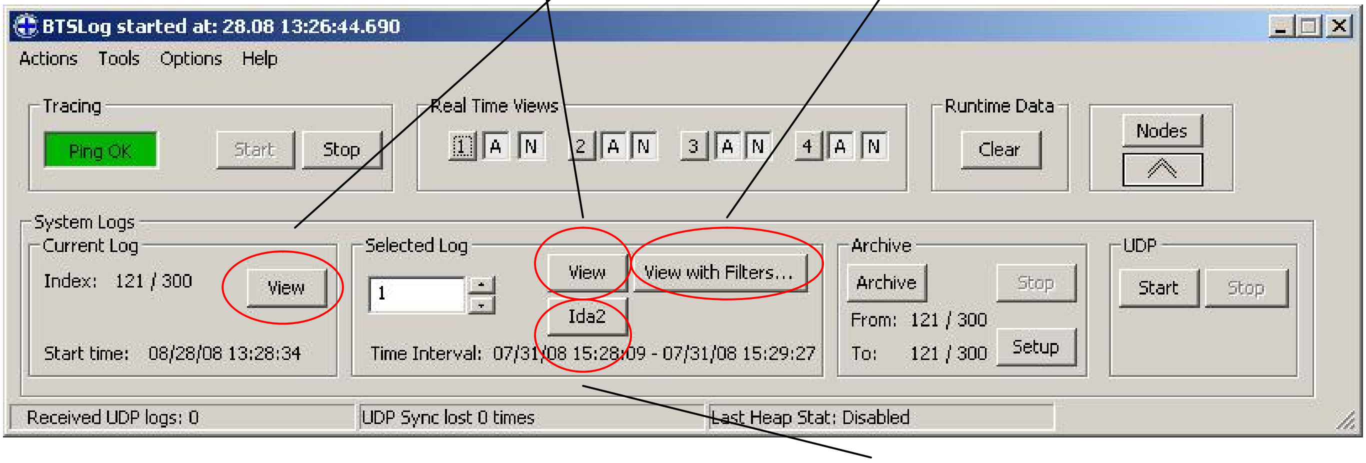


IP Address	Time Stamp	Log Text
127.0.0.1	01.01 01:10:35.462	5c WSP_23: (WSP_MCU) <01.01 01:10:35.462>HWA DEBUG/SELF_TEST, writePollStartDelayUs:2
127.0.0.1	01.01 01:10:35.462	5d WSP_23: (WSP_MCU) <01.01 01:10:35.462>HWA DEBUG/SELF_TEST, erasePollStartDelayUs:
127.0.0.1	01.01 01:10:35.462	5e WSP_23: (WSP_MCU) <01.01 01:10:35.462>HWA DEBUG/SELF_TEST, drvDelayReCalibration:
127.0.0.1	01.01 01:10:35.462	5f WSP_23: (WSP_MCU) <01.01 01:10:35.462>HWA DEBUG/SELF_TEST, ffxMaxDrvExecutionTin
127.0.0.1	01.01 01:10:35.462	60 WSP_23: (WSP_MCU) <01.01 01:10:35.462>HWA DEBUG/SELF_TEST, totalErasedSectors:0,
127.0.0.1		!! Lost: from 127.0.0.1 !!
127.0.0.1	01.01 01:10:03.478	c8 WAM_10: <01.01 01:10:03.478>O&M DEBUG/GEN, MEDIATOR: Generated polling response tc
127.0.0.1		!! Lost: from 127.0.0.1 !!
127.0.0.1	01.01 01:10:36.163	61 WSP_23: (WSP_MCU) <01.01 01:10:36.163>HWA DEBUG/SELF_TEST, POOL OBS: NUMBER C
127.0.0.1		62 WSP_23: (WSP_MCU) POOL OBS: Pool id Bytes
127.0.0.1		63 WSP_23: (WSP_MCU) POOL OBS: -----------------
127.0.0.1		64 WSP_23: (WSP_MCU) POOL OBS: 0x1003b 10489664
127.0.0.1		65 WSP_23: (WSP_MCU) POOL OBS: 0x10029 2753536
127.0.0.1		66 WSP_23: (WSP_MCU) POOL OBS: 0x10000 2196800
		***BTSLog note - Message trace started nodeName: LOCAL_HOST pcTime: 2008 8.28 15:02:27.8
		***BTSLog note - Message trace started nodeName: LOCAL_HOST pcTime: 2008 8.28 15:03:29.2
		***BTSLog note - Message trace started nodeName: LOCAL_HOST pcTime: 2008 8.28 15:04:30.6
		***BTSLog note - Message trace started nodeName: LOCAL_HOST pcTime: 2008 8.28 15:05:32.9
		***BTSLog note - Message trace started nodeName: LOCAL_HOST pcTime: 2008 8.28 15:06:33.4

5. VIEWING RECORDED SYSTEM LOGS

You can view recorded system logs in three different ways:

1. View the current system log by pressing View - button in Current Log - section.
2. View selected log by pressing View - button in Selected Log - section
3. View log content using user defined filters by pressing View with Filters… -button



Index: 121 / 300

Start time: 08/28/08 13:28:34

1

Time Interval: 07/31/08 15:28:09 - 07/31/08 15:29:27

From: 121 / 300

To: 121 / 300

Received UDP logs: 0

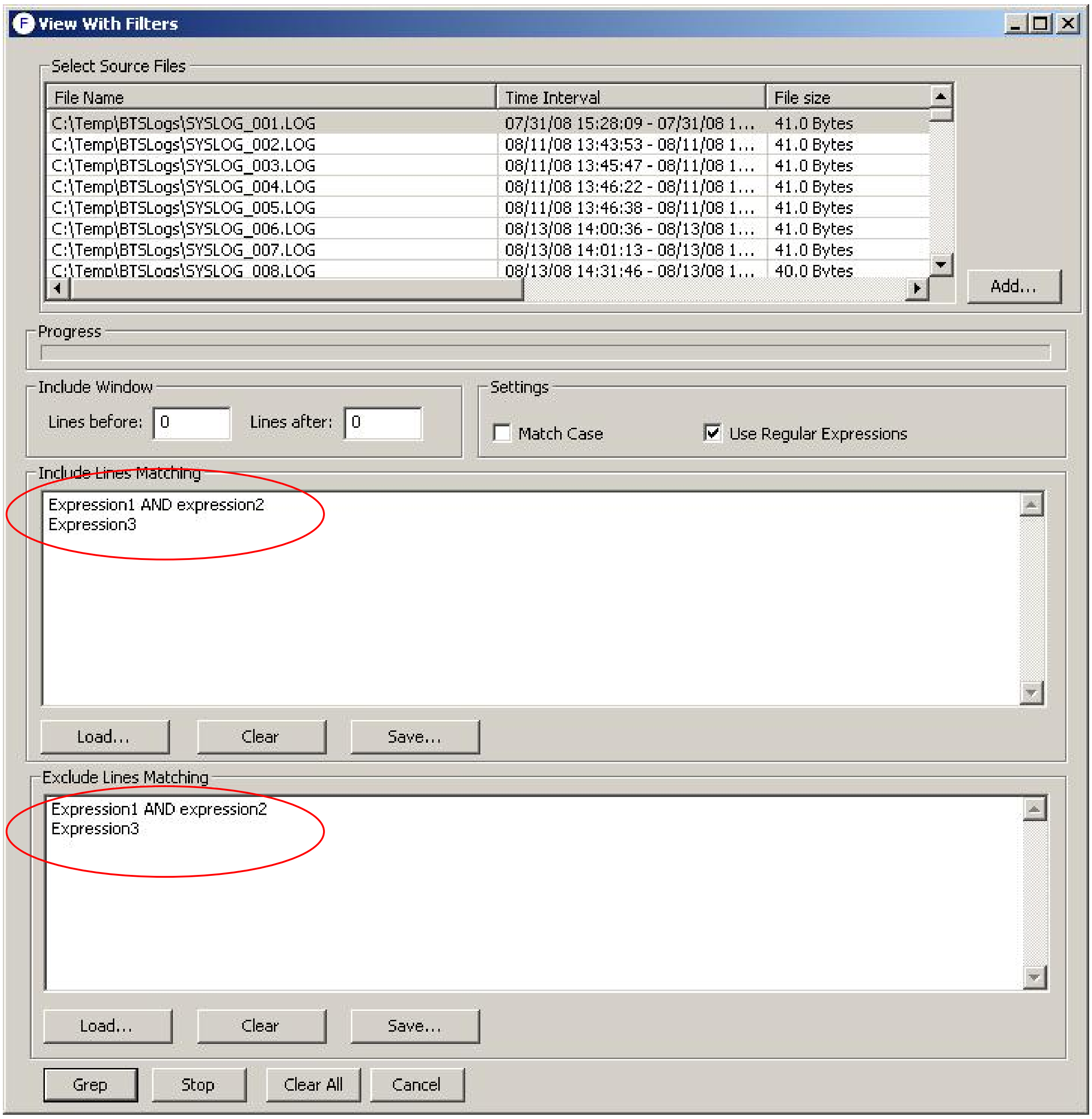
UDP Sync lost 0 times

Last Heap Stat: Disabled

There is also possibility to open selected trace log files by pressing Ida2 –button in Selected Log - section.

5.1 Viewing system log with filter

To open the Filters dialog press the View with Filters- button.



File Name	Time Interval	File size
C:\Temp\BTSLogs\SYSLOG_001.LOG	07/31/08 15:28:09 - 07/31/08 1...	41.0 Bytes
C:\Temp\BTSLogs\SYSLOG_002.LOG	08/11/08 13:43:53 - 08/11/08 1...	41.0 Bytes
C:\Temp\BTSLogs\SYSLOG_003.LOG	08/11/08 13:45:47 - 08/11/08 1...	41.0 Bytes
C:\Temp\BTSLogs\SYSLOG_004.LOG	08/11/08 13:46:22 - 08/11/08 1...	41.0 Bytes
C:\Temp\BTSLogs\SYSLOG_005.LOG	08/11/08 13:46:38 - 08/11/08 1...	41.0 Bytes
C:\Temp\BTSLogs\SYSLOG_006.LOG	08/13/08 14:00:36 - 08/13/08 1...	41.0 Bytes
C:\Temp\BTSLogs\SYSLOG_007.LOG	08/13/08 14:01:13 - 08/13/08 1...	41.0 Bytes
C:\Temp\BTSLogs\SYSLOG_008.LOG	08/13/08 14:31:46 - 08/13/08 1...	40.0 Bytes

Lines before: 0 Lines after: 0

☐ Match Case ☑ Use Regular Expressions

Include Lines Matching:

```
Expression1 AND expression2
Expression3
```

Exclude Lines Matching:

```
Expression1 AND expression2
Expression3
```

Syntax for both include and exclude is the same. You can use the reserved uppercase keyword AND as logical operator on a single row. Linefeed means logical operator OR.

Example:

Expression1 AND expression2
Expression3

Is the same as: (Expression1 AND Expression2) OR Expression3.

Include lines

If include lines box is empty, all data is passed. That is because some one might wan't to use only exclude criteria.

Exclude lines

Excludes overrun includes. If exclude criteria matches, includes are not inspected.

Lines after and Lines before

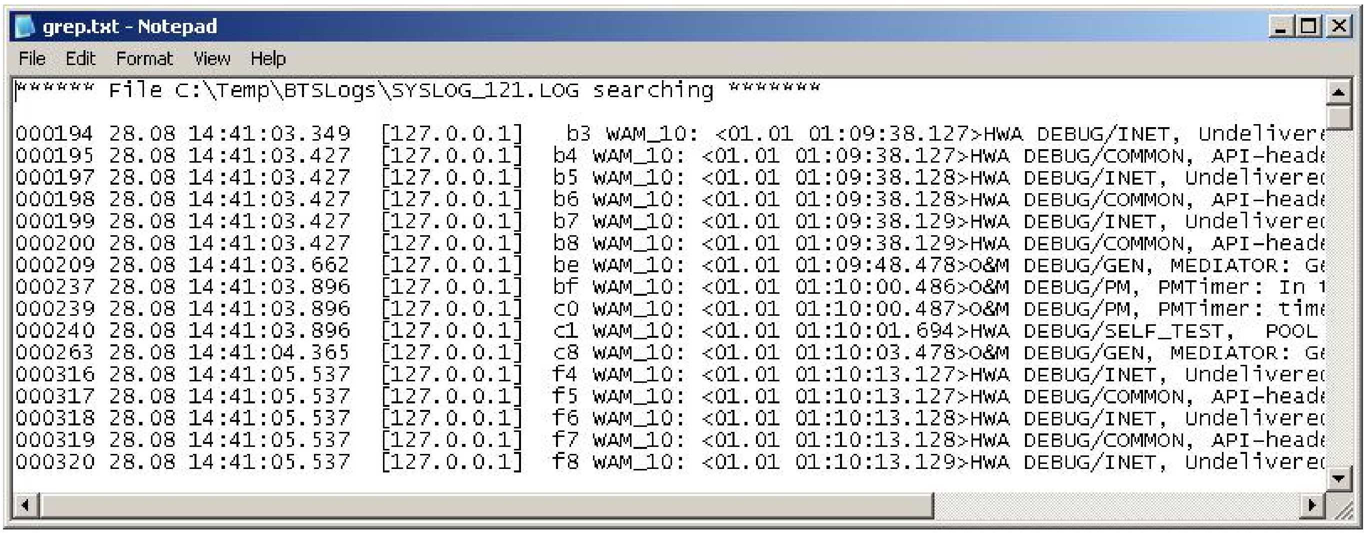
If you wan't to include to result file lines close to matching rows, put the numbers into boxes. No dublicate rows are written into result file if preceding lines make matches.

Regular Expressions

Regular expressions are made optional, because it uses certain characters (* . ? * +) in special purposes. This may conflict with some search criteria, like a search expression starting with '('.

Regular expressions don't support so called globbing. I.e. expanded wild char search, used for example by windows file search. More specs of Regexp can be found e.g. in wikipedia.
Btslog uses VBScript.dll for regexp.

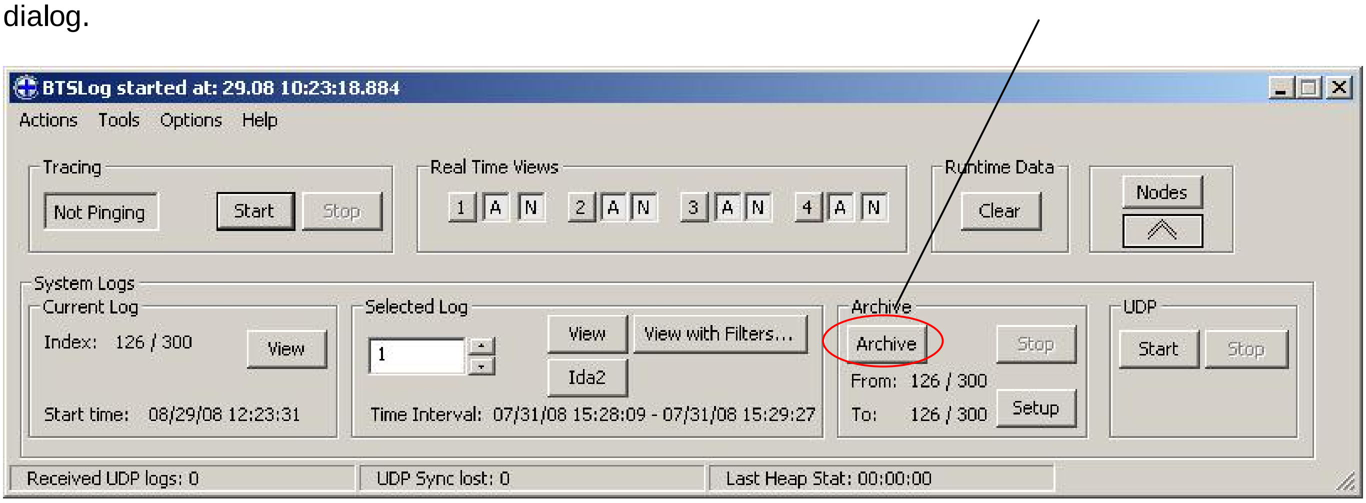
Example of filter result:



```
****** File C:\Temp\BTSLogs\SYSLOG_121.LOG searching ********

000194 28.08 14:41:03.349  [127.0.0.1]   b3 WAM_10: <01.01 01:09:38.127>HWA DEBUG/INET, Undeliver
000195 28.08 14:41:03.427  [127.0.0.1]  b4 WAM_10: <01.01 01:09:38.127>HWA DEBUG/COMMON, API-head
000197 28.08 14:41:03.427  [127.0.0.1]  b5 WAM_10: <01.01 01:09:38.128>HWA DEBUG/INET, Undelivere
000198 28.08 14:41:03.427  [127.0.0.1]  b6 WAM_10: <01.01 01:09:38.128>HWA DEBUG/COMMON, API-head
000199 28.08 14:41:03.427  [127.0.0.1]  b7 WAM_10: <01.01 01:09:38.129>HWA DEBUG/INET, Undelivere
000200 28.08 14:41:03.427  [127.0.0.1]  b8 WAM_10: <01.01 01:09:38.129>HWA DEBUG/COMMON, API-head
000209 28.08 14:41:03.662  [127.0.0.1]  be WAM_10: <01.01 01:09:48.478>O&M DEBUG/GEN, MEDIATOR: G
000237 28.08 14:41:03.896  [127.0.0.1]  bf WAM_10: <01.01 01:10:00.486>O&M DEBUG/PM, PMTimer: In
000239 28.08 14:41:03.896  [127.0.0.1]  c0 WAM_10: <01.01 01:10:00.487>O&M DEBUG/PM, PMTimer: tim
000240 28.08 14:41:03.896  [127.0.0.1]  c1 WAM_10: <01.01 01:10:01.694>HWA DEBUG/SELF_TEST,  POOL
000263 28.08 14:41:04.365  [127.0.0.1]  c8 WAM_10: <01.01 01:10:03.478>O&M DEBUG/GEN, MEDIATOR: G
000316 28.08 14:41:05.537  [127.0.0.1]  f4 WAM_10: <01.01 01:10:13.127>HWA DEBUG/INET, Undelivere
000317 28.08 14:41:05.537  [127.0.0.1]  f5 WAM_10: <01.01 01:10:13.127>HWA DEBUG/COMMON, API-head
000318 28.08 14:41:05.537  [127.0.0.1]  f6 WAM_10: <01.01 01:10:13.128>HWA DEBUG/INET, Undelivere
000319 28.08 14:41:05.537  [127.0.0.1]  f7 WAM_10: <01.01 01:10:13.128>HWA DEBUG/COMMON, API-head
000320 28.08 14:41:05.537  [127.0.0.1]  f8 WAM_10: <01.01 01:10:13.129>HWA DEBUG/INET, Undelivere
```

6. ARCHIVING LOG FILES

Log file (both trace log and system log files) can be archived by pushing Archive button of BTSLOG main dialog.



Using Archive's setup is optional. From and to - indexes are set dynamically like this:

When Clear is pressed
1. Current index increases with one
2. From = Current
3. To = Current

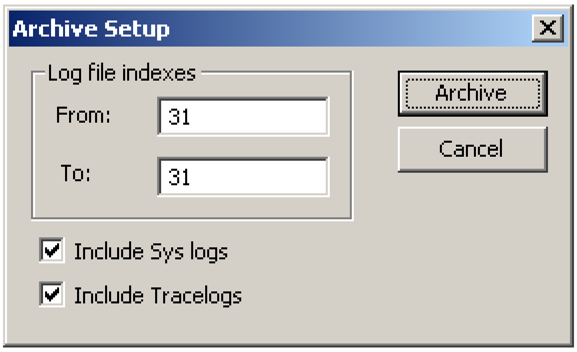
When start is pressed
1. Current index increases with one
2. From = Current
3. To = Current

When stop is pressed
1. Current remains the same
2. From remains the same
3. To remains the same

When BTSLog automatically creates new log files, when they reach their maximum size, ”current” increases with one and ”to” is sychronized with it.

If "Archive" is pressed straight away, From - To are zipped. If ”to” happens to be open and on the trace, it is flushed and put also to package.

If you need to change indexes, press setup and write desired indexes to "from" and "to" boxes, press OK, and the achiving starts.



Log file indexes

From: 31

To: 31

☑ Include Sys logs

☑ Include Tracelogs

- By default Include syslogs ja include tracelogs checkboxes are set
- If "From" is a bigger value than "to", BTSlog wraps around the indexes according to max logfile count.

7. HANDLING RUNTIME DATA

7.1 Auto sync of log files & log file indexes

Log files are automatically kept in sync with an index number in log name.
BTSLOG_TRACE_001.bin
SYSLOG_001.LOG

When the logfile exceeds the defined max size, a new log is created. The log file type which gets first filled, dictates the point of creation. Both trace and UDP logs are always created simultaneously. Even if only the other of the trace types is running, an empty log file is still created for the other one.

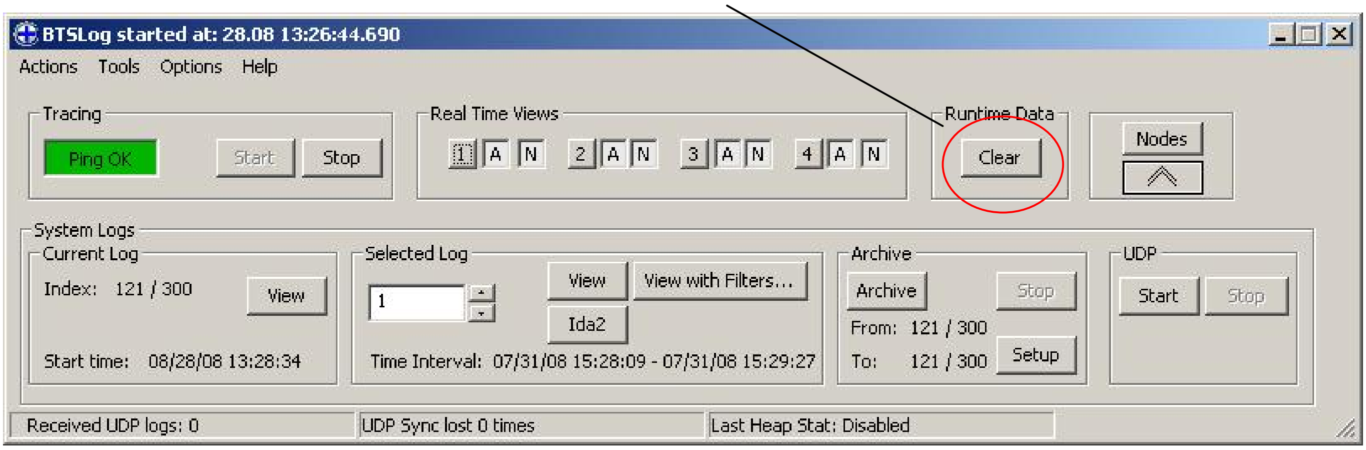
BTSLog adds zeroes to logfile index according to maximum number of files. This is to get log files sorted according to log file index, not to name.
For example:

maximum number of files. = 300
BTSLOG_TRACE_001.bin - BTSLOG_TRACE_300.bin

maximum number of files = 1000
BTSLOG_TRACE_0001.bin - BTSLOG_TRACE_1000.bin

7.2 Clear button

Clear button can be pressed at any point, during the trace or when stopped.
It closes current logs and creates new ones. No recorded data is lost. It also resets the counters and clears the views.
If you want to delete all recorded logs in your defined output folder, press Actions/ Delete Logs.

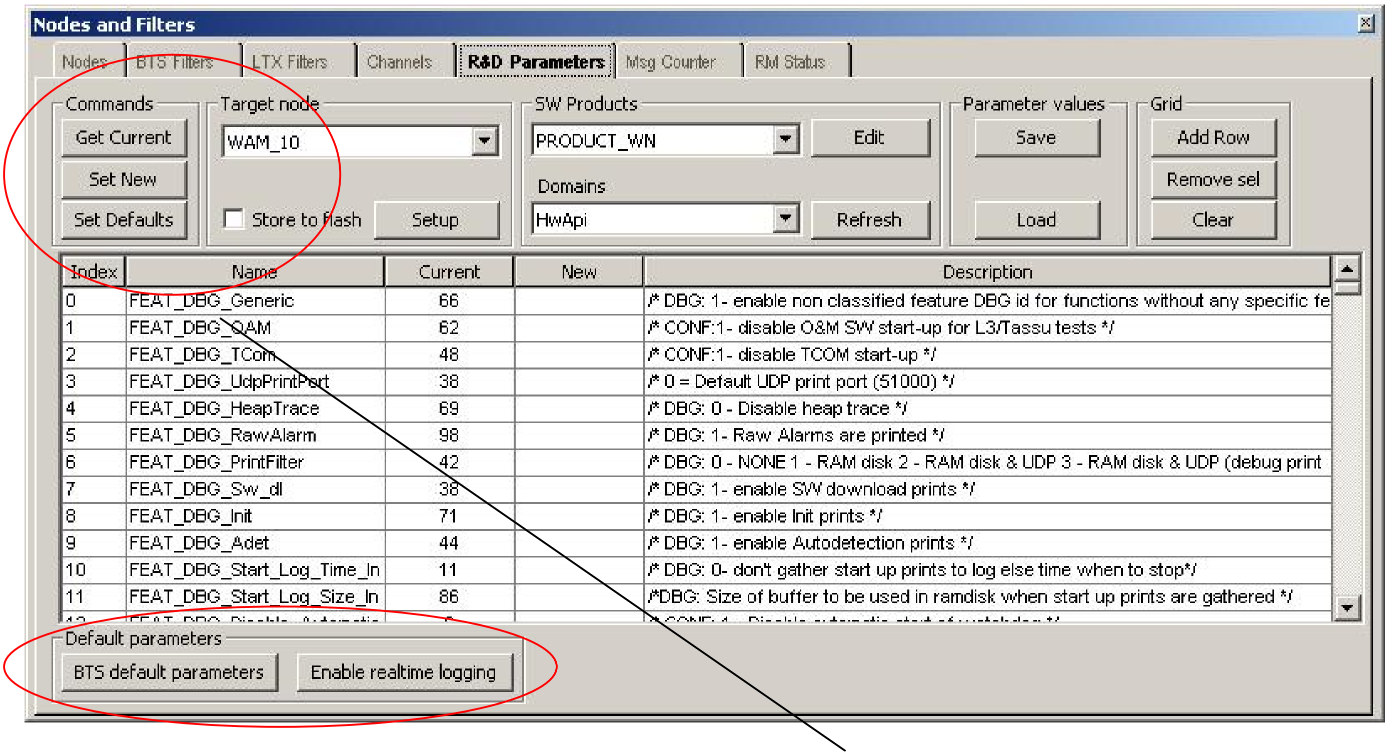

BTSLog started at: 28.08 13:26:44.690

Index: 121 / 300
Start time: 08/28/08 13:28:34
Time Interval: 07/31/08 15:28:09 - 07/31/08 15:29:27
From: 121 / 300
To: 121 / 300

Received UDP logs: 0
UDP Sync lost 0 times
Last Heap Stat: Disabled

8. R&D PARAMETERS

You can handle setting and getting the R&D parameters for selected domain in Nodes and Filters window on R&D tab.



Target node: WAM_10

☐ Store to flash

SW Products: PRODUCT_WN

Domains: HwApi

Index	Name	Current	New	Description
0	FEAT_DBG_Generic	66		/* DBG: 1- enable non classified feature DBG id for functions without any specific fe
1	FEAT_DBG_OAM	62		/* CONF:1- disable O&M SW start-up for L3/Tassu tests */
2	FEAT_DBG_TCom	48		/* CONF:1- disable TCOM start-up */
3	FEAT_DBG_UdpPrintPort	38		/* 0 = Default UDP print port (51000) */
4	FEAT_DBG_HeapTrace	69		/* DBG: 0 - Disable heap trace */
5	FEAT_DBG_RawAlarm	98		/* DBG: 1- Raw Alarms are printed */
6	FEAT_DBG_PrintFilter	42		/* DBG: 0 - NONE 1 - RAM disk 2 - RAM disk & UDP 3 - RAM disk & UDP (debug print
7	FEAT_DBG_Sw_dl	38		/* DBG: 1- enable SW download prints */
8	FEAT_DBG_Init	71		/* DBG: 1- enable Init prints */
9	FEAT_DBG_Adet	44		/* DBG: 1- enable Autodetection prints */
10	FEAT_DBG_Start_Log_Time_In	11		/* DBG: 0- don't gather start up prints to log else time when to stop*/
11	FEAT_DBG_Start_Log_Size_In	86		/*DBG: Size of buffer to be used in ramdisk when start up prints are gathered */

- Set new – Sends the values of Index- and in New- columns of selected domain
- Get Current – Fetches parameters of selected domain
- Set Defaults – Sends to BTS a request to restore defaults for selected domain.
- Store to flash – Sends to BTS a request to store params for selected domain, sent with Set New, and store to flash memory.

- BTS default parameters – Sets default parameter to BTS by sending empty SetRadParamsReq – message to selected node's selected SW product domain (Legacy) and finally sets parameters 6=5 and 244=1 for Legacy. If Legacy is not included in domains list, domain id zero (0) is used.

- Enable realtime logging – Reads user defined parameter(s) from user_radparams.ini file and sends them to BTS. User_radparams.ini file format is explained in chapter 8.2.

8.1 Miscellaneous

- RAD param requests establish an individual TCP connection, if selected node is not in trace
- If selected node is already in trace, rad param requests go via existing connection
- Node can be selected in dropdown list, or from rad param setup dialog, which allows e.g. overriding NMAP address of RAD param messages.
- If Master WAM is enabled, master WAM is first detected and RAD param messages are posted to the NMAP address gotten from detection reply. If Master WAM is disabled, Board CPU and task are read from their editboxes.

- If you cannot see the parameter names you're propably missing the radparamsacks. Check out the sack settings with edit button in SW Products section. Ini file explains its structure in the header.
- If you are using BTS default parameters or Enable realtime logging –buttons during trace it could take some time to send all parameter to BTS (there is a short delay between messages during trace).

8.2 BTS internal R&D RadParameter file format

A couple of examples of R&D Parameter saving format.
If Domain zero's 1st parameter would be one, the value would be:

0x00000000 = 1

if domain 3, parameter 5. is 34, the value would be:
0x00030005 = 34

Summary.
Both in the same file:

0x00000000 = 1
0x00030005 = 34

- If the Domain/index has 0x prefix, its HEX, oherwise it's DEC
- If the value has 0x prefix, its HEX, oherwise it's DEC

- If the Domain/index has only 4 bytes after 0x prefix, domain is 0.
- If the Domain/index has more than 4 bytes after 0x prefix, domain is the value of the exceeding part.

When saving parameters, there will be an option for appending or replacing,
if the file already exists.

- Saving to exisiting file with Append means that values are written to the end of the file.
- Saving to existing file with Overwrite, destroys the contents and writes new values to file.

User has the responsibility of managing the file. I.e. saving with append, it's possible to append the same param value several times.

If the param is written several times to file, only the last one is taken into use by BTSLog. BTS behaviour is undefined in this case.

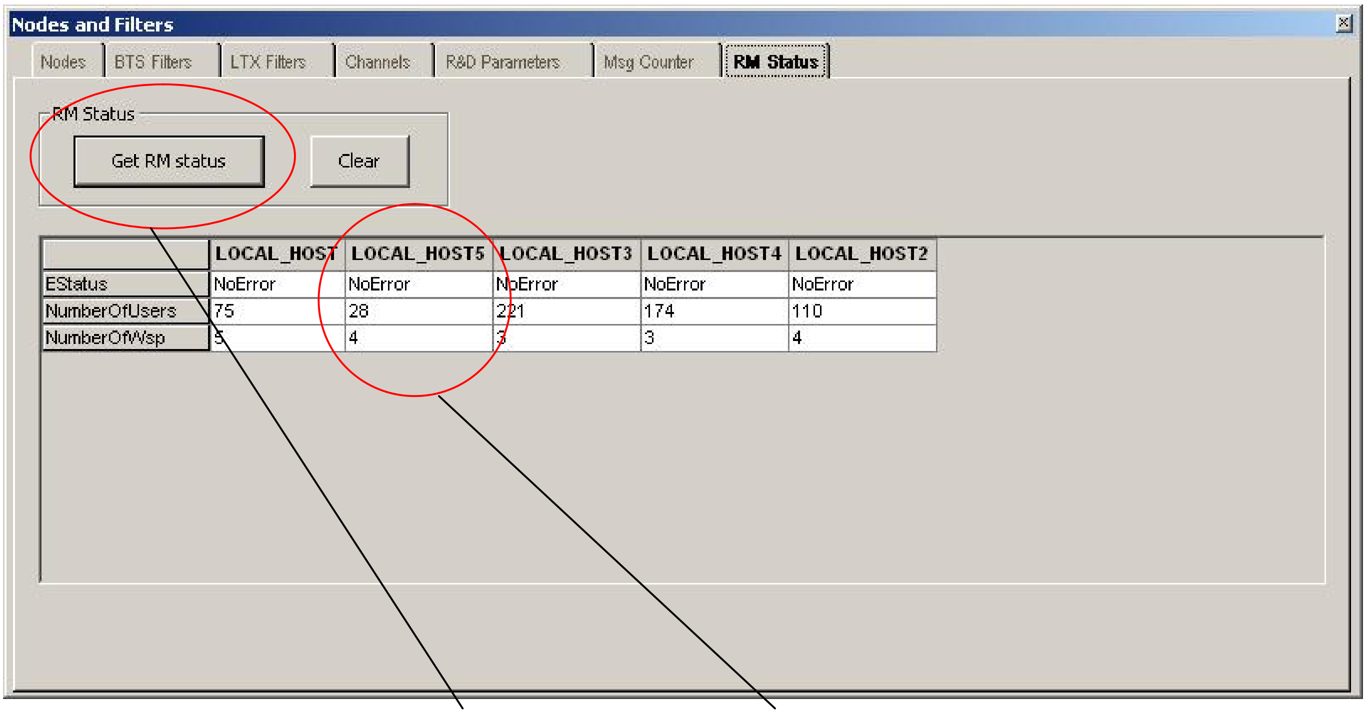
9. RM STATUS TABLES

You can fetch RM status tables from node(s) which have RM parameter in nodes.ini. See example below:

```
[WAM_10]
Connection=TCP
IPAddress=192.168.255.1
Board=0x10
CPU=0x11
Task=0x0
Port=BTSLOG
EchoTask=0x0300
MsgTask=0x0309
RadTask=0x0300
RMTask=0x151
Type=WCDMA
```

Please remember restart your BTSLog after editing nodes.ini file. File content is not refreshed automatically.

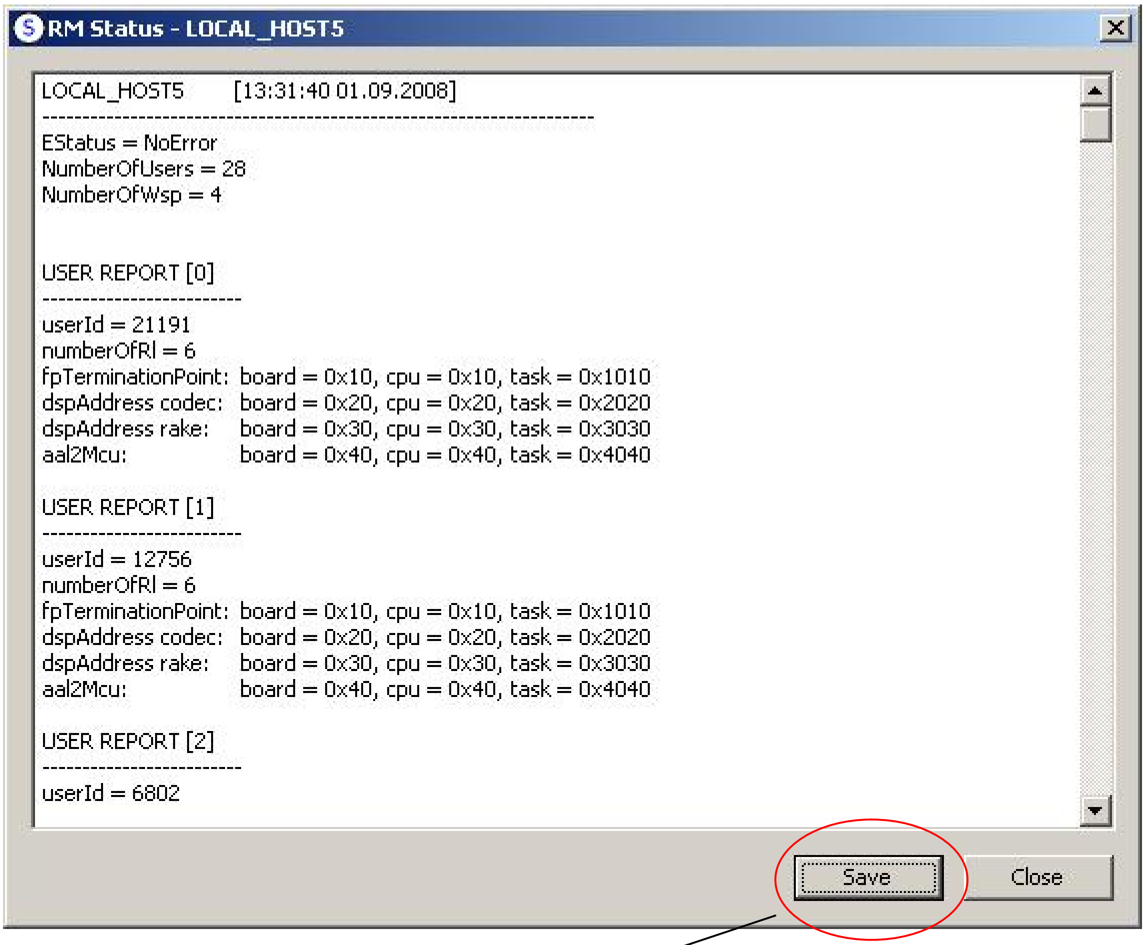
At the moment (BTSLog 1.5-0) only WN4.0 nodes are supported.



	LOCAL_HOST	LOCAL_HOST5	LOCAL_HOST3	LOCAL_HOST4	LOCAL_HOST2
EStatus	NoError	NoError	NoError	NoError	NoError
NumberOfUsers	75	28	221	174	110
NumberOfWsp	5	4	3	3	4

RM query will be executed to all nodes (which have RM parameter) at the same time when the “Get RM status” button is pressed. If node has an existing TCP connection (node in trace) it will be used for query, otherwise a new TCP connection will be created. If node is not answering in 10 second it will be ignored.

The query results are displayed in a grid tab and in text format in own popup text window. The grid display will contain summary information about each node query. Popup window will be opened when eg. node name (column) is clicked.



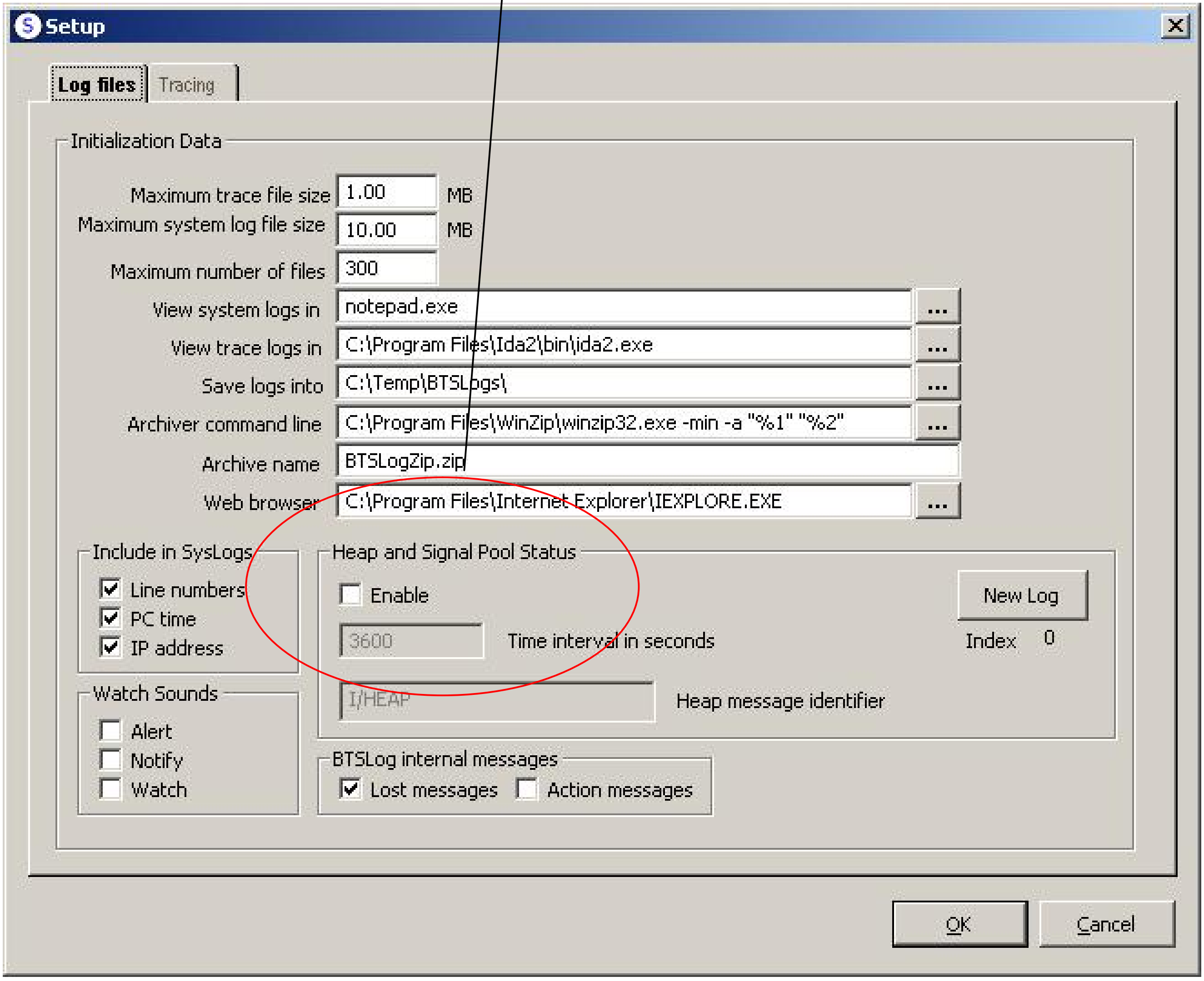
```
LOCAL_HOST5      [13:31:40 01.09.2008]
--------------------------------------------------------------------
EStatus = NoError
NumberOfUsers = 28
NumberOfWsp = 4


USER REPORT [0]
------------------------
userId = 21191
numberOfRl = 6
fpTerminationPoint:  board = 0x10, cpu = 0x10, task = 0x1010
dspAddress codec:    board = 0x20, cpu = 0x20, task = 0x2020
dspAddress rake:     board = 0x30, cpu = 0x30, task = 0x3030
aal2Mcu:             board = 0x40, cpu = 0x40, task = 0x4040

USER REPORT [1]
------------------------
userId = 12756
numberOfRl = 6
fpTerminationPoint:  board = 0x10, cpu = 0x10, task = 0x1010
dspAddress codec:    board = 0x20, cpu = 0x20, task = 0x2020
dspAddress rake:     board = 0x30, cpu = 0x30, task = 0x3030
aal2Mcu:             board = 0x40, cpu = 0x40, task = 0x4040

USER REPORT [2]
------------------------
userId = 6802
```

When node name is clicked it will open a popup window which shows full content of the query in text format and there is a possibility to save window content to text file by pressing Save -button.

10. COLLECTING SYSLOG HEAP AND SIGNAL POOL STATUS

Enabling Heap and signal pool status saves certain parts of UDP messages, which contain the heap message identifier, into heapstats.txt conatined int the same directory like btslog.exe.

Heapstats.txt can be read with Excel.



Initialization Data

Maximum trace file size 1.00 MB
Maximum system log file size 10.00 MB
Maximum number of files 300
View system logs in notepad.exe
View trace logs in C:\Program Files\Ida2\bin\ida2.exe
Save logs into C:\Temp\BTSLogs\
Archiver command line C:\Program Files\WinZip\winzip32.exe -min -a "%1" "%2"
Archive name BTSLogZip.zip
Web browser C:\Program Files\Internet Explorer\IEXPLORE.EXE

Include in SysLogs
☑ Line numbers
☑ PC time
☑ IP address

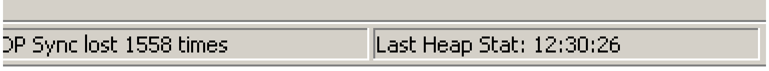
Watch Sounds
□ Alert
□ Notify
□ Watch

Heap and Signal Pool Status
□ Enable
3600 Time interval in seconds
Index 0
I/HEAP Heap message identifier

BTSLog internal messages
☑ Lost messages □ Action messages

- Time interval affects the collecting of messages from particular unit. For example, if an I/HEAP contained message arrives from WAM_10, it is stored. The next I/HEAP message from WAM_10 is not stored until the time interval has elapsed.
- Heap Message identifier is pretty much constant, INFO/HEAP, or I/HEAP in later WN-releases, but might change in the future. That's why the editbox is there.

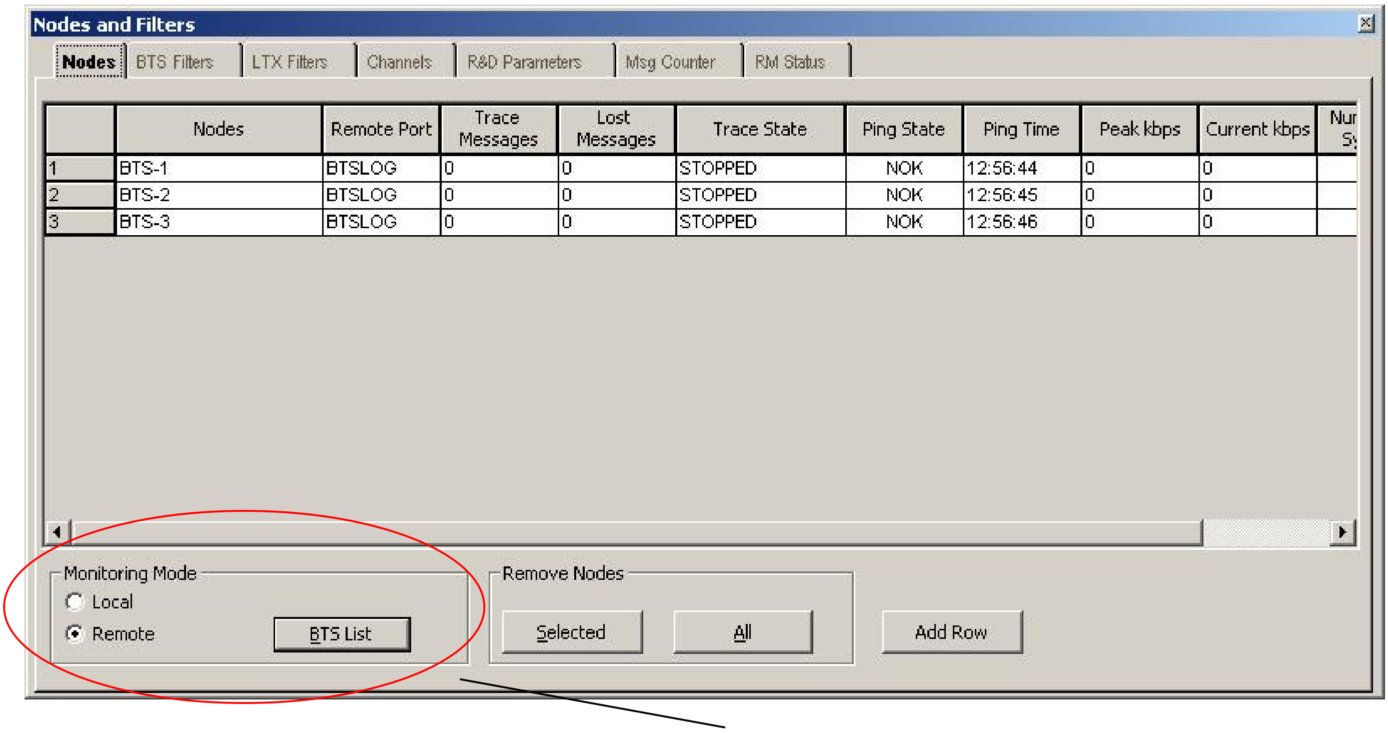
As additional information, timestamp of the latest collected heap stat is presented in main window's statusbar.



DP Sync lost 1558 times | Last Heap Stat: 12:30:26

11. REMOTE MONITORING

Remote monitoring is redesigned since version 1.3-0.
Now it’s possible to monitor several BTS’s simultaneously.

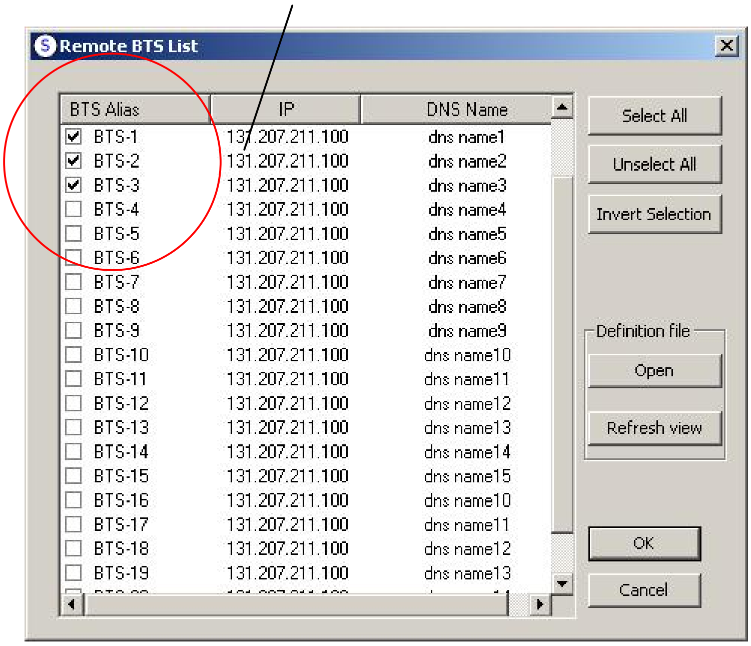


	Nodes	Remote Port	Trace Messages	Lost Messages	Trace State	Ping State	Ping Time	Peak kbps	Current kbps	Nur Sy
1	BTS-1	BTSLOG	0	0	STOPPED	NOK	12:56:44	0	0	
2	BTS-2	BTSLOG	0	0	STOPPED	NOK	12:56:45	0	0	
3	BTS-3	BTSLOG	0	0	STOPPED	NOK	12:56:46	0	0	

Monitoring Mode
○ Local
● Remote

In monitoring mode-section, select remote radio-button. Press BTS List-button.

Select desired BTS’s to trace from opened window.



BTS Alias	IP	DNS Name
☑ BTS-1	131.207.211.100	dns name1
☑ BTS-2	131.207.211.100	dns name2
☑ BTS-3	131.207.211.100	dns name3
☐ BTS-4	131.207.211.100	dns name4
☐ BTS-5	131.207.211.100	dns name5
☐ BTS-6	131.207.211.100	dns name6
☐ BTS-7	131.207.211.100	dns name7
☐ BTS-8	131.207.211.100	dns name8
☐ BTS-9	131.207.211.100	dns name9
☐ BTS-10	131.207.211.100	dns name10
☐ BTS-11	131.207.211.100	dns name11
☐ BTS-12	131.207.211.100	dns name12
☐ BTS-13	131.207.211.100	dns name13
☐ BTS-14	131.207.211.100	dns name14
☐ BTS-15	131.207.211.100	dns name15
☐ BTS-16	131.207.211.100	dns name10
☐ BTS-17	131.207.211.100	dns name11
☐ BTS-18	131.207.211.100	dns name12
☐ BTS-19	131.207.211.100	dns name13

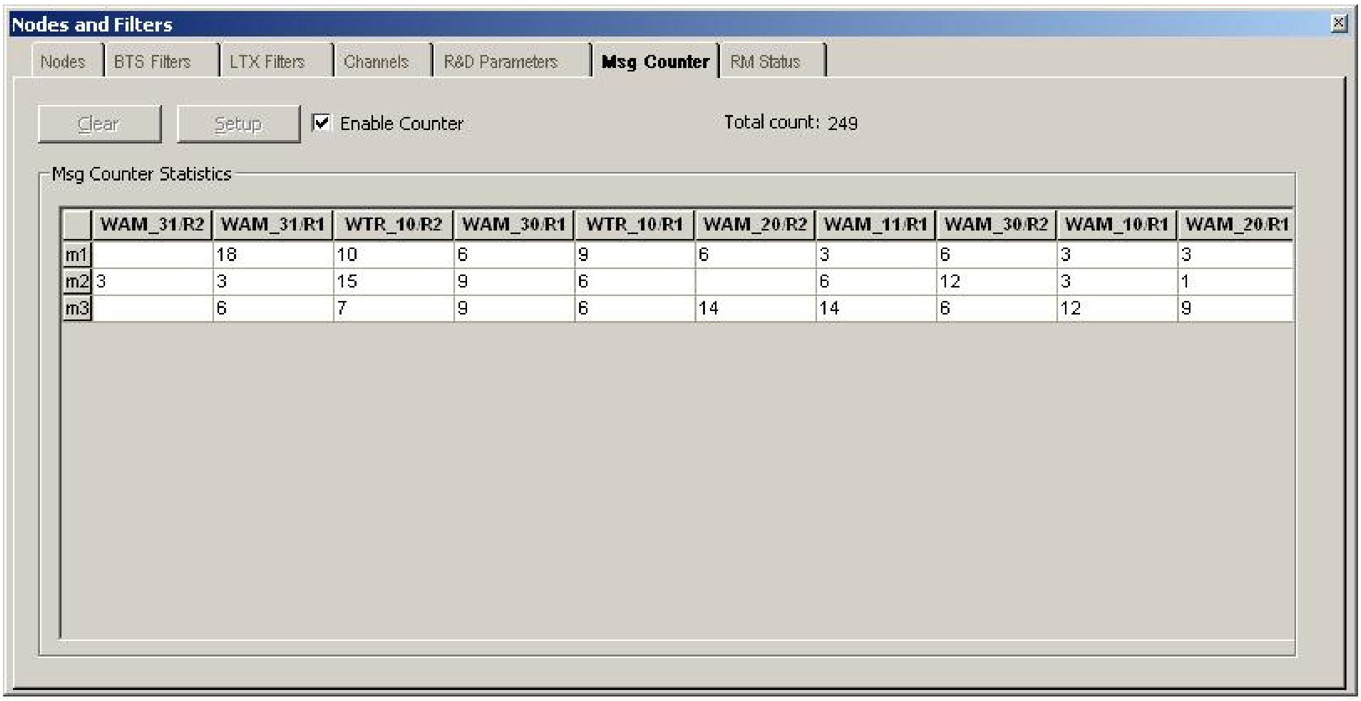
The picture on above is just an example. Press Open-button in definition file section for editing. Edit the file with correct BTS-IPs, save the file and press refresh. New list will appear.
File is located in:

C:\Program Files\BTSlog\config\RemoteBTSList.txt

Remote monitoring collects all messages to same trace log in the same manner than local monitoring does.

12. MESSAGE COUNTER

This feature enables calculating certain message ids just by message id or with id and with values from the hexdump.
Each sender unit / cpu is presented in own column. They appear during the trace.



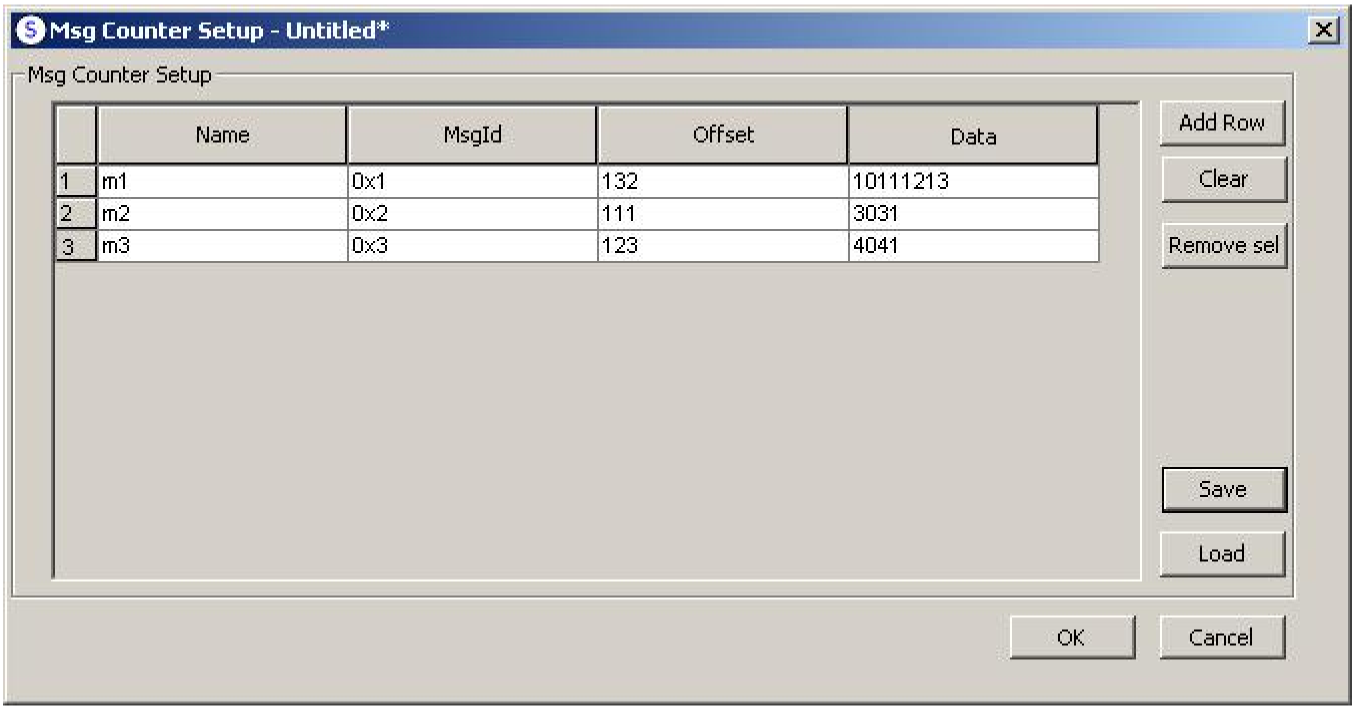
☑ Enable Counter

Total count: 249

Msg Counter Statistics

	WAM_31/R2	WAM_31/R1	WTR_10/R2	WAM_30/R1	WTR_10/R1	WAM_20/R2	WAM_11/R1	WAM_30/R2	WAM_10/R1	WAM_20/R1
m1		18	10	6	9	6	3	6	3	3
m2	3	3	15	9	6		6	12	3	1
m3		6	7	9	6	14	14	6	12	9

Setup is guite self explanatory. It paint syntax errors red and reports about them. The length of the data is calculated automatically. Text written in to name column appear in statistic view after leaving setup dialog with OK -button.

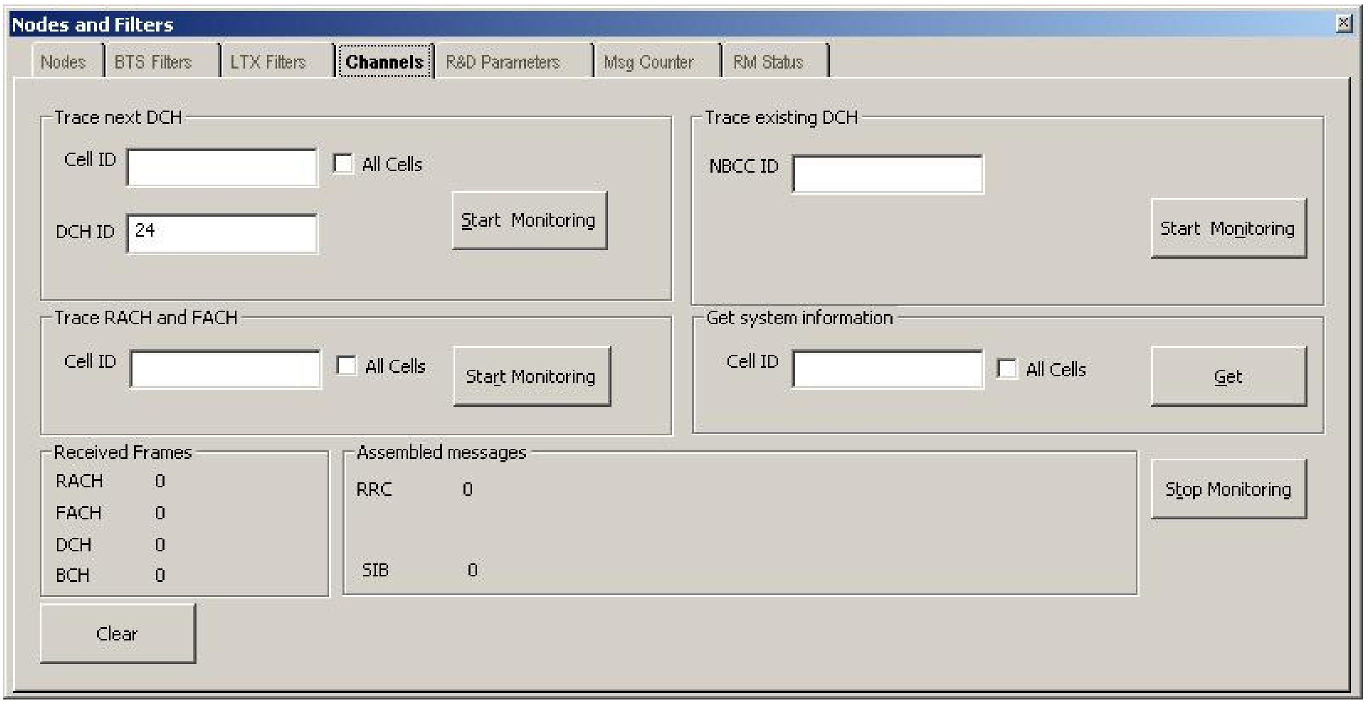


	Name	MsgId	Offset	Data
1	m1	0x1	132	10111213
2	m2	0x2	111	3031
3	m3	0x3	123	4041

Unfortunately message counter does not support LTX nodes yet in this version.

13. CHANNEL MONITORING

This feature makes it possible to monitor so called RRC, CC and MM messages which are transparent at Node B's point of view.
To collect those messages, the connection has to be first established by starting the message trace. When the trace is running and any of the start buttons in the picture pressed, the channel monitoring request is sent to all WAM-nodes that are currently in trace.
Stop command stops all channel monitoring.



Nodes and Filters

Trace next DCH

Cell ID

□ All Cells

DCH ID 24

Trace existing DCH

NBCC ID

Trace RACH and FACH

Cell ID

□ All Cells

Get system information

Cell ID

□ All Cells

Received Frames

RACH	0
FACH	0
DCH	0
BCH	0

Assembled messages

RRC	0
SIB	0

RRC, CC and MM messages arrive to BTSLog as FP-frames.
BTSLog combines the frames to Iub-messages and stores them to to same trace log with all other trace messages.
(Channel monitoring is under construction at the point this is written. Full implementation target in BTSLog 1.6-0 ???)

14. INI FILES

14.1 Btslog.ini

```
[MRU FILES]

NodesFile=C:\Program Files\Btslog\Config\nodes.ini  Set in install package.Edit manually

LTXFilter=C:\temp\bin\all.ltx   Saved automatically when filter is saved

BTSFilter=C:\temp\all.bts      Saved automatically when filter is saved

LogFileIndex=1      Current index of the log file ring buffer. Saved automatically


[PCNODE]   // Settings for UDPCP tracing

IPAddress=192.168.255.76  IP address of PC running BTSLog when communicating

Port=FCMUDPCP  IP port of PC running BTSLog when communicating

                with a BTS node using UDPCP protocol (see PORTS section)

Board=0x10     NMAP address of BTS node simulated by PC running

               BTSLog when with a BTS node using UDPCP protocol

CPU=0x11

Task=0x0001


[UDPCP]

CFlag= YES      YES - UDPCP protocol uses CRC checksum (Adler32 checksu

                NO  - UDPCP protocol does not use CRC (checksum field of UDPCP headers will be

                      filled with 0 bits)

NFlag=NO        NFlag = NO,  SFlag = NO     - every UDPCP packet will be acknowledged

                NFlag = NO,  SFlag = YES    - only the last UDPCP packet of message will be

                                              acknowledged

                NFlag = YES, SFlag = YES/NO - UDPCP packets will not be acknowledged

SFlag=NO

RetryLimit      number of retries when sending an UDPCP packet, before sending will be considered
```

failed

RetryTimeout maximum timeout to between sending an UDPCP packet and receiving the acknowledgement packet

```
 [PORTS]  Ports are aliased, that they are easier to recognize in nodes window
NONE=0
LTX=100
TASSU=15001
BTSLOG=15004
FCMUDPCP=12000
DSP=12001
DSP1=12002
DSP2=12003
DSP3=12004
FSPUDPCP1=771
FSP1=12021
FSP2=12022
FSP3=12023
```

```
[PROTOCOL] // Protocols for LTX filter settings
All=0
RTOS=1
SWBus=2
UDP=3
UDPCP=4
SOAP=5
```

```
[DIRECTION]  // Directions for LTX filter settings

Both=0

Send=1

Receive=2
```

14.2 Nodes.ini

Fields are explained in nodes.ini's header.

15. FAQ

If you have problems with BTSLog, see the typical problems and solutions below.

You can always contact BTSLog support mailto:BTSlog.Support@nsn.com.

It is recommended to straight away include to email BtsLog_ExecutionLog_x.txt and all other logs involved with the problem, because BTSLog support will ask for them anyway.

If the problem is involved with message traffic, it's recommended to record the problem with a network protocol analyzer (Etherreal - Wireshark) and inlude the libcap files to the support request.

If the problem is involved with GUI, snapshots and a complete explanation about the problem helps solving the case faster.

Frequently asked questions

Q: Where to get latest BTSLog install package?
A:
https://dominoext.access.nokiasiemensnetworks.com/bi/eeserviceshomepa.nsf/document/ES01X6MNF9M?OpenDocument

Q: What tools are capable of opening logs created by Btslog?
A: Syslogs are ASCII data. Those can be read with any text editor.
Tracelogs are binary data, stored in so called Ida2 format.
Ida2 is propably the only tool for opening BTSLog's trace logs in a readable form.
You can load ida2 from:
https://dominoext.access.nokiasiemensnetworks.com/bi/eeserviceshomepa.nsf/document/ESLNC0026C9ALH?OpenDocument

Sacks for **Ida2** can be found in:
https://dominoext.access.nokiasiemensnetworks.com/bi/eeserviceshomepa.nsf/document/ESLNC0026C9ALK?OpenDocument

Q: What WN-software versions does BTSLog support ?
A: WN2.0 CD2 and newer including WN3.x, WN4.x

Q: Is there a command line version of BTSLog available?
A:Not yet. Perhaps in the future. How ever, BTSLog can be commanded externally with BTSLog_launcher-tool, described in this tutorial in command line arguments-chapter.

Q: Can BTSLog be used over a network ?
A: Some parts of it yes. See chapter REMOTE MONITORING in this tutorial.

Q: How does BTSlog connect with BTS?
A:In TCP trace BTSLog connects to the node according the node's IP settings in nodes.ini.
In UDP trace BTSLog listens to ports defined in setup dialog.
BTS is broadcasting datagrams to these ports.
In future there'll be a feature in BTSLog for sending unicast IP address to BTS.

Q: What is LTX?
A: LTX means "radio module".
LTX is the name of the SW system component for radio module SW.
(before module BTS there was plan for a plug-in unit called LTX)

Q: What is the difference between WAM & LTX?
A: LTX testability interface and its platform SW is somewhat different from other units/submodules (WAM, WTR, FCM etc).
In BTSLog's point of view, LTX has a different message interface compared to all other units. For example, trying to send BTS filters to LTX unit won't work. And vice verca.

Q: Cannot get a node into trace. What to do?
A: 1. Check that selected node pings from command promt.
2. Make sure that your PC's network adapter's IP address is set to lab environment.
3. Inspect BtsLog_ExecutionLog_x.txt for detailed information
4. Inspect the handshakes between BTSLog and BTS with a network protocol analyzer tool (Etherreal, Wireshark, Commview).
4. If BTSLog is still not tracing, send the Ethereal-logs to BTSLog support.
5.. Firewalls on PC are also often causing problems.

Q: BTSLog doesn't receive UDP messages. What to do?
A: 1. Make sure that start button is pressed in UDP section.
(Message trace and syslog trace have individual start and stop commands.)
2. Check out that UDP ports in setup dialog match the current BTS environment.
default ports are 51000, 51001, 50011.
3. Firewalls on PC are also often causing problems.

Q: Cannot receive rad parameters. Grid remains empty. What to do?
A: 1. Make sure that the unit, which should respond, is up and running.
Try pinging and tracing it.
2. Inspect with a network protocol analyzer (Etherreal, Wireshark, Commview) if BTS is responding to R&D param requests, sent by BTSLog.

Q: Cannot see rad parameter names. What to do?
A: Inspect your radparam sack settings in radparamsackinfo.ini.
It has a description of the contents in its header.

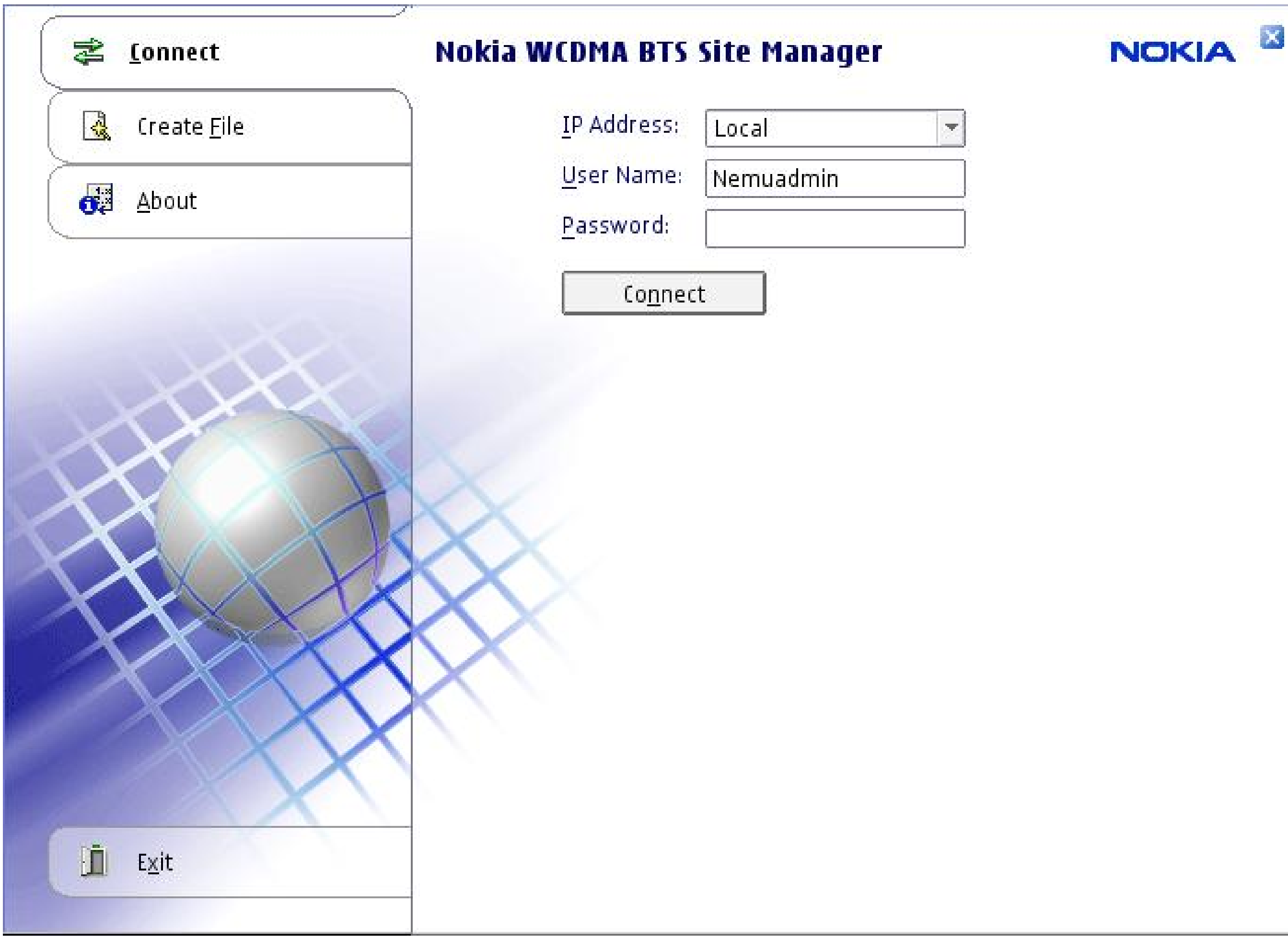
Q: How to define more than 8 heavyload signals for a BTS filter ?
A: Unfortunately it is no possible. Filter message has a static
structure for heavy loaded signals. Max is 8.

Q: Why does BTSLog receive 0x2015 messages (Seen in Ida2),
even though it's not defined in to the filter?
A: Message 0x2015 is an RRC-message assembled by BTSLog's channel monitoring
system. BCH-frames arrive from BTS to BTSLog with id 0x5004. BTSLog assembles
the frames and creates a new message with id 0x2015.

16. WCDMA LOOP TEST EXAMPLE

A brief example how Nokia WCDMA BTS Site Manager, BTSLog and Ida2 can be used together.

1. Make sure that FlexiBTS is up and running
2. Launch Element Manager in lab PC
3. Log in by pressing 'Connect'

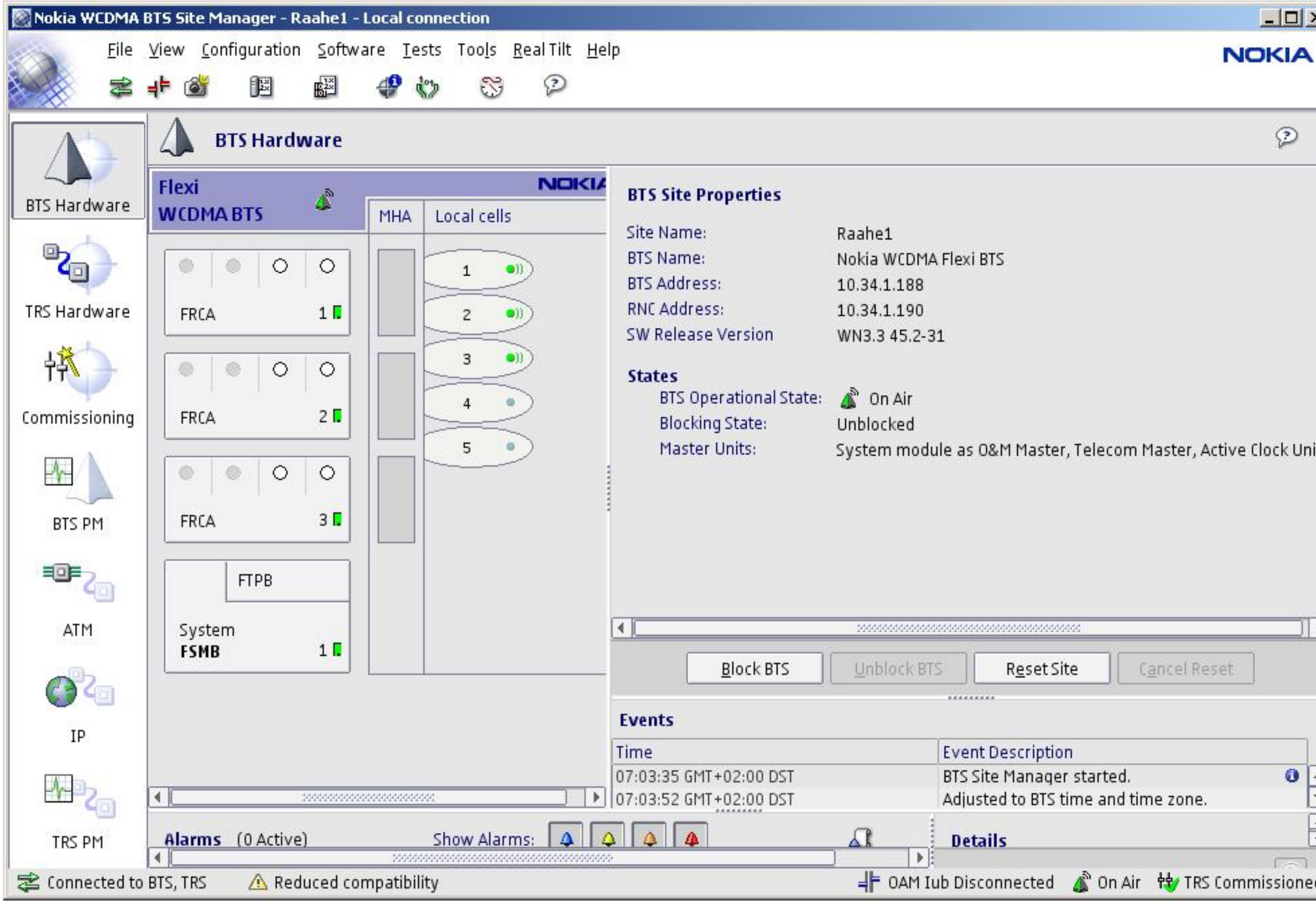


IP Address: Local

User Name: Nemuadmin

Password:

4. Wait until Element Manager starts up, Flexi is onAir and BTS operational state is OnAir.



Nokia WCDMA BTS Site Manager - Raahe1 - Local connection

BTS Site Properties

Site Name: Raahe1
BTS Name: Nokia WCDMA Flexi BTS
BTS Address: 10.34.1.188
RNC Address: 10.34.1.190
SW Release Version WN3.3 45.2-31

States

BTS Operational State: On Air
Blocking State: Unblocked
Master Units: System module as O&M Master, Telecom Master, Active Clock Uni

Events

Time	Event Description
07:03:35 GMT+02:00 DST	BTS Site Manager started.
07:03:52 GMT+02:00 DST	Adjusted to BTS time and time zone.

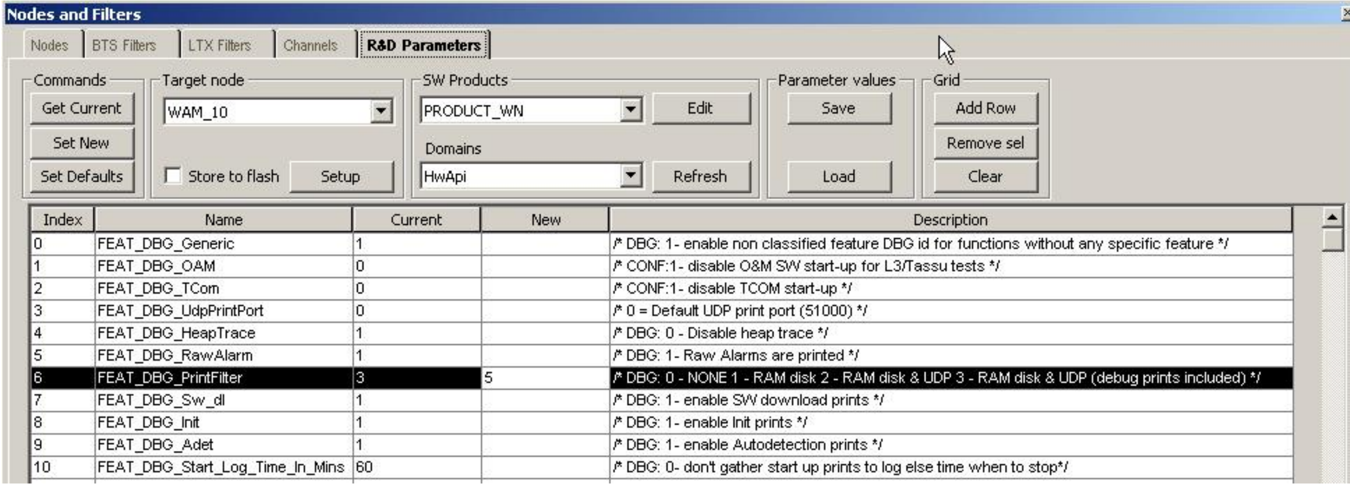
Alarms (0 Active)

Connected to BTS, TRS | Reduced compatibility | OAM Iub Disconnected | On Air | TRS Commissione

5. Launch BTSLog in Lab PC
6. Enabling print flags:
   Open Nodes and Filters window, goto R&D parameters page.

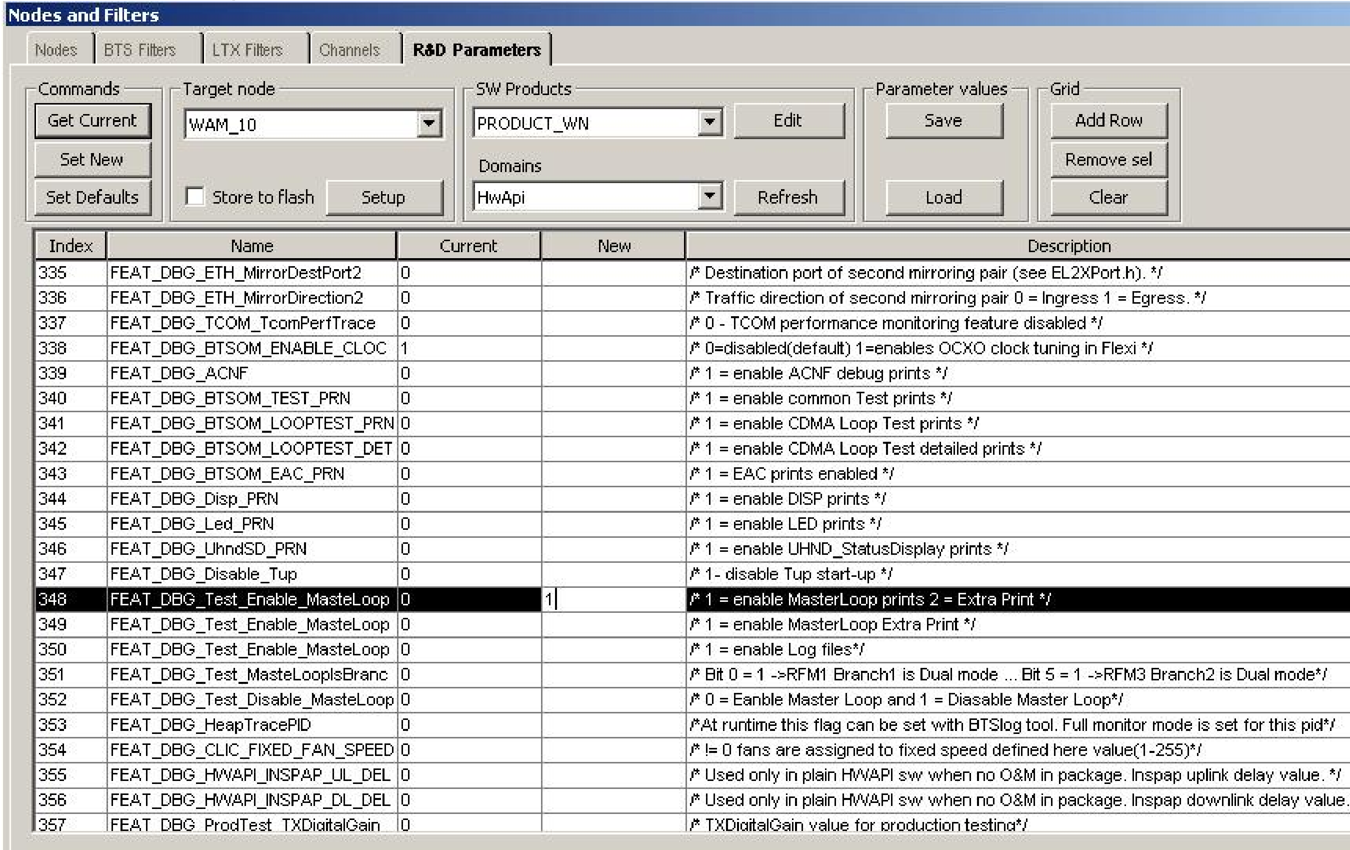
Choose WAM_10 Target node, PRODUCT_WN product and HWApi domain.
Press 'Get Current'

Enable prints with value 5.



Index	Name	Current	New	Description
0	FEAT_DBG_Generic	1		/* DBG: 1- enable non classified feature DBG id for functions without any specific feature */
1	FEAT_DBG_OAM	0		/* CONF:1- disable O&M SW start-up for L3/Tassu tests */
2	FEAT_DBG_TCom	0		/* CONF:1- disable TCOM start-up */
3	FEAT_DBG_UdpPrintPort	0		/* 0 = Default UDP print port (51000) */
4	FEAT_DBG_HeapTrace	1		/* DBG: 0 - Disable heap trace */
5	FEAT_DBG_RawAlarm	1		/* DBG: 1- Raw Alarms are printed */
6	FEAT_DBG_PrintFilter	3	5	/* DBG: 0 - NONE 1 - RAM disk 2 - RAM disk & UDP 3 - RAM disk & UDP (debug prints included) */
7	FEAT_DBG_Sw_dl	1		/* DBG: 1- enable SW download prints */
8	FEAT_DBG_Init	1		/* DBG: 1- enable Init prints */
9	FEAT_DBG_Adet	1		/* DBG: 1- enable Autodetection prints */
10	FEAT_DBG_Start_Log_Time_In_Mins	60		/* DBG: 0- don't gather start up prints to log else time when to stop*/

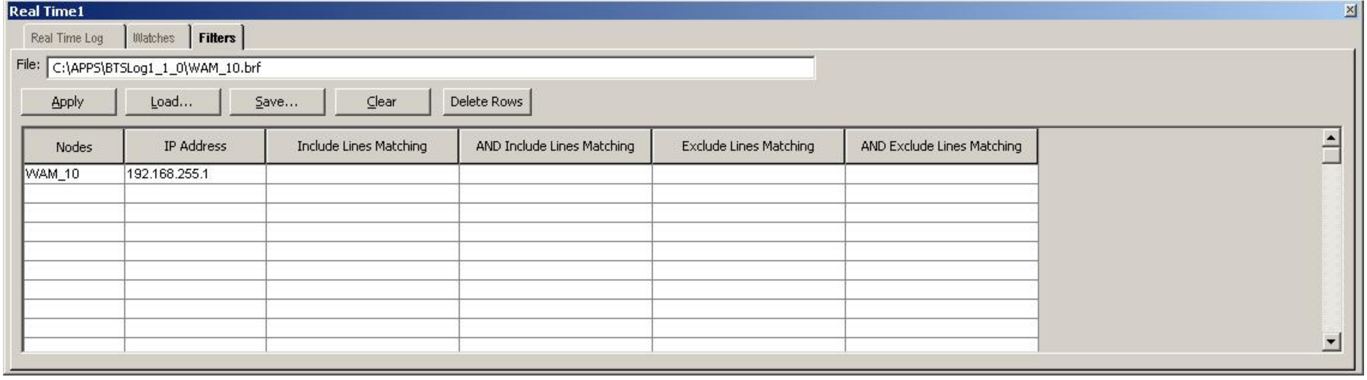
Enable Masterloop with value 1.



Index	Name	Current	New	Description
335	FEAT_DBG_ETH_MirrorDestPort2	0		/* Destination port of second mirroring pair (see EL2XPort.h). */
336	FEAT_DBG_ETH_MirrorDirection2	0		/* Traffic direction of second mirroring pair 0 = Ingress 1 = Egress. */
337	FEAT_DBG_TCOM_TcomPerfTrace	0		/* 0 - TCOM performance monitoring feature disabled */
338	FEAT_DBG_BTSOM_ENABLE_CLOC	1		/* 0=disabled(default) 1=enables OCXO clock tuning in Flexi */
339	FEAT_DBG_ACNF	0		/* 1 = enable ACNF debug prints */
340	FEAT_DBG_BTSOM_TEST_PRN	0		/* 1 = enable common Test prints */
341	FEAT_DBG_BTSOM_LOOPTEST_PRN	0		/* 1 = enable CDMA Loop Test prints */
342	FEAT_DBG_BTSOM_LOOPTEST_DET	0		/* 1 = enable CDMA Loop Test detailed prints */
343	FEAT_DBG_BTSOM_EAC_PRN	0		/* 1 = EAC prints enabled */
344	FEAT_DBG_Disp_PRN	0		/* 1 = enable DISP prints */
345	FEAT_DBG_Led_PRN	0		/* 1 = enable LED prints */
346	FEAT_DBG_UhndSD_PRN	0		/* 1 = enable UHND_StatusDisplay prints */
347	FEAT_DBG_Disable_Tup	0		/* 1- disable Tup start-up */
348	FEAT_DBG_Test_Enable_MasteLoop	0	1	/* 1 = enable MasterLoop prints 2 = Extra Print */
349	FEAT_DBG_Test_Enable_MasteLoop	0		/* 1 = enable MasterLoop Extra Print */
350	FEAT_DBG_Test_Enable_MasteLoop	0		/* 1 = enable Log files*/
351	FEAT_DBG_Test_MasteLoopIsBranc	0		/* Bit 0 = 1 ->RFM1 Branch1 is Dual mode ... Bit 5 = 1 ->RFM3 Branch2 is Dual mode*/
352	FEAT_DBG_Test_Disable_MasteLoop	0		/* 0 = Eanble Master Loop and 1 = Diasable Master Loop*/
353	FEAT_DBG_HeapTracePID	0		/*At runtime this flag can be set with BTSlog tool. Full monitor mode is set for this pid*/
354	FEAT_DBG_CLIC_FIXED_FAN_SPEED	0		/* != 0 fans are assigned to fixed speed defined here value(1-255)*/
355	FEAT_DBG_HWAPI_INSPAP_UL_DEL	0		/* Used only in plain HWAPI sw when no O&M in package. Inspap uplink delay value. */
356	FEAT_DBG_HWAPI_INSPAP_DL_DEL	0		/* Used only in plain HWAPI sw when no O&M in package. Inspap downlink delay value.
357	FEAT_DBG_ProdTest_TXDigitalGain	0		/* TXDigitalGain value for production testing*/

Press 'Set new'. You can check that values are set correctly by pressing 'Get current'.

Open realtime window. You can filter the prints of desired nodes on filters tab.

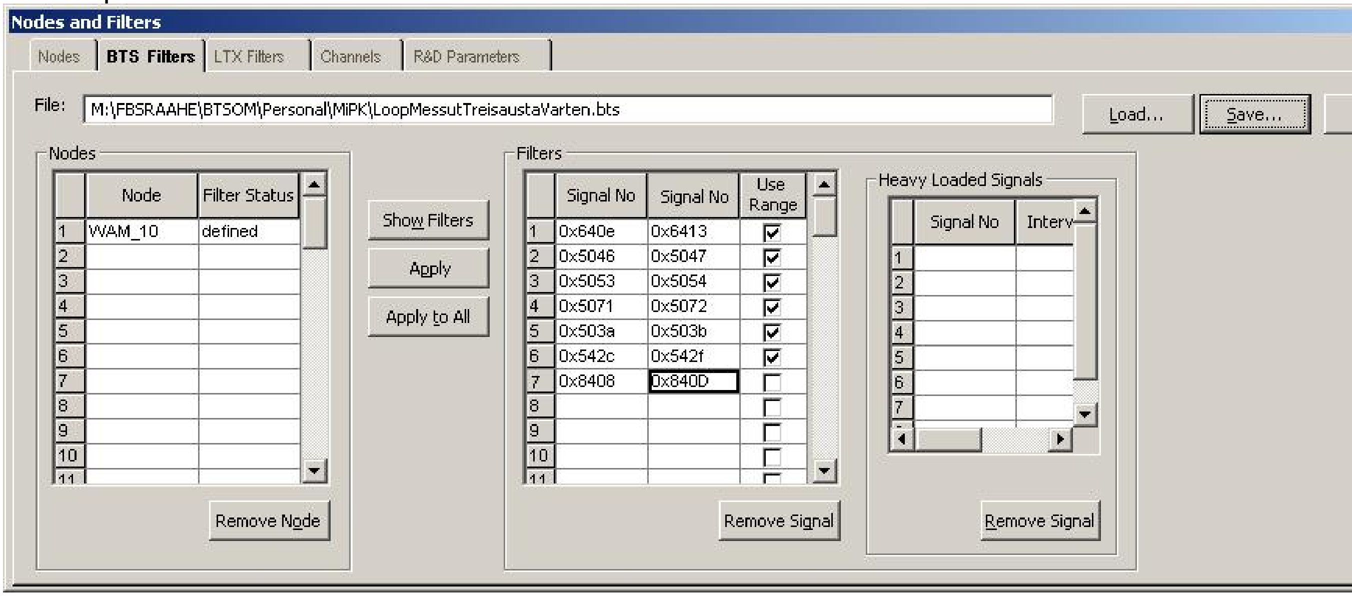


File: C:\APPS\BTSLog1_1_0\WAM_10.brf

Nodes	IP Address	Include Lines Matching	AND Include Lines Matching	Exclude Lines Matching	AND Exclude Lines Matching
WAM_10	192.168.255.1				

Set the message trace filters on filters page of Nodes and filters window

(0x640e - 0x6413, 0x5046-0x5047, 0x5053 - 0x5054, 0x5071 -0x5072, 0x503a -0x503b, 0x542c - 0x542f and (0x8408 - 0x840D) .
When ready, press apply. If you save the filter, it is automatically loaded, when BTSLog is closed and reopened.



File: M:\FBSRAAHE\BTSOM\Personal\MiPK\LoopMessutTreisaustaVarten.bts

Nodes

	Node	Filter Status
1	WAM_10	defined
2		
3		
4		
5		
6		
7		
8		
9		
10		
11		

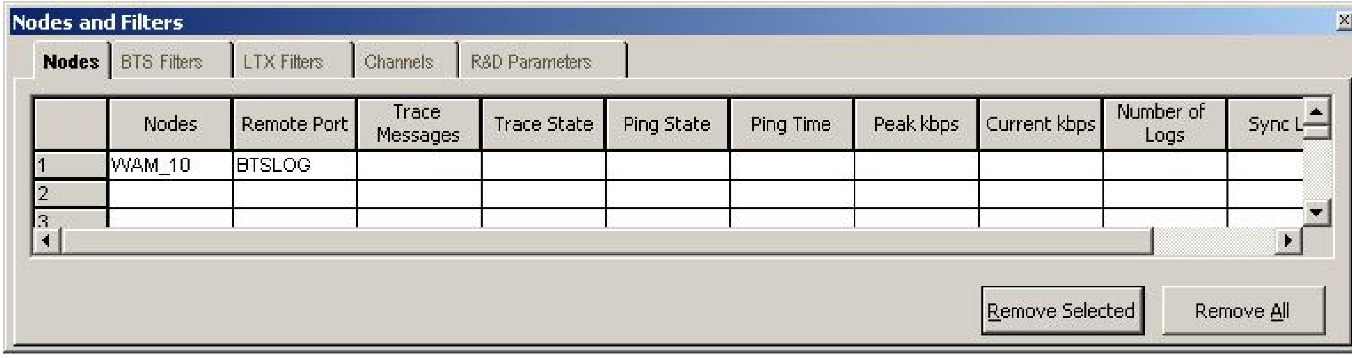
Filters

	Signal No	Signal No	Use Range
1	0x640e	0x6413	☑
2	0x5046	0x5047	☑
3	0x5053	0x5054	☑
4	0x5071	0x5072	☑
5	0x503a	0x503b	☑
6	0x542c	0x542f	☑
7	0x8408	0x840D	☐
8			☐
9			☐
10			☐
11			☐

Heavy Loaded Signals

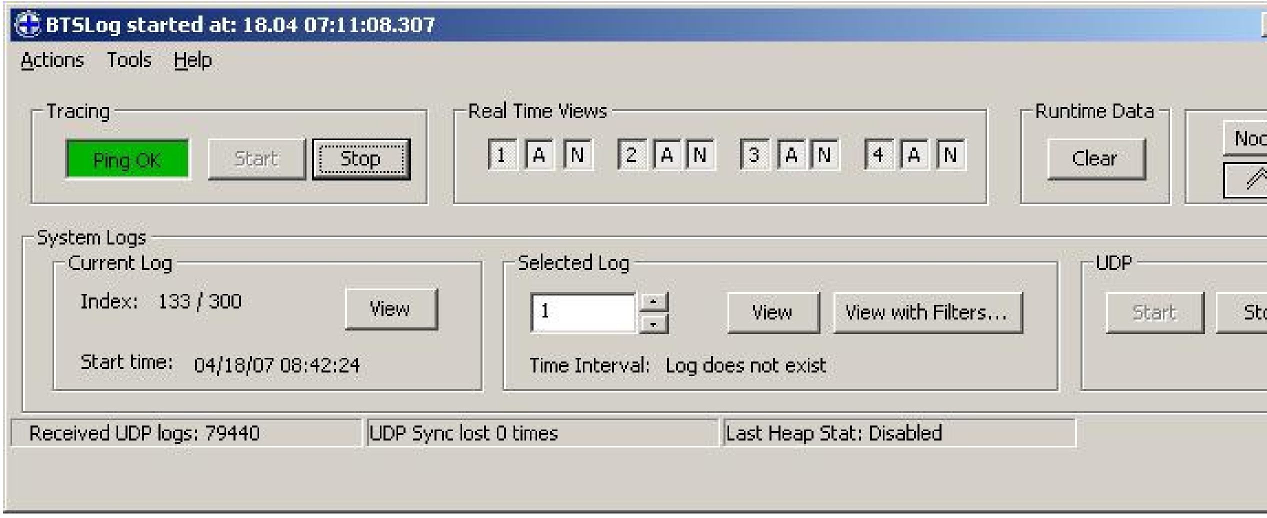
	Signal No	Interv
1		
2		
3		
4		
5		
6		
7		

Goto Nodes tab. Select WAM_10 from the dropdown list on *Nodes* column.



	Nodes	Remote Port	Trace Messages	Trace State	Ping State	Ping Time	Peak kbps	Current kbps	Number of Logs	Sync L
1	WAM_10	BTSLOG								
2										
3										

Press both start buttons in *Tracing* and in *UDP* sections.



BTSLog started at: 18.04 07:11:08.307

Index: 133 / 300

Start time: 04/18/07 08:42:24

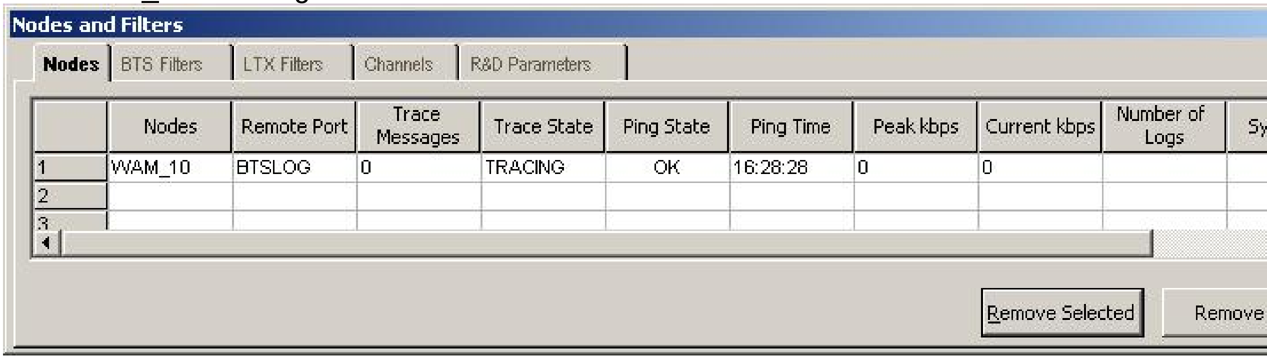
Time Interval: Log does not exist

Received UDP logs: 79440

UDP Sync lost 0 times

Last Heap Stat: Disabled

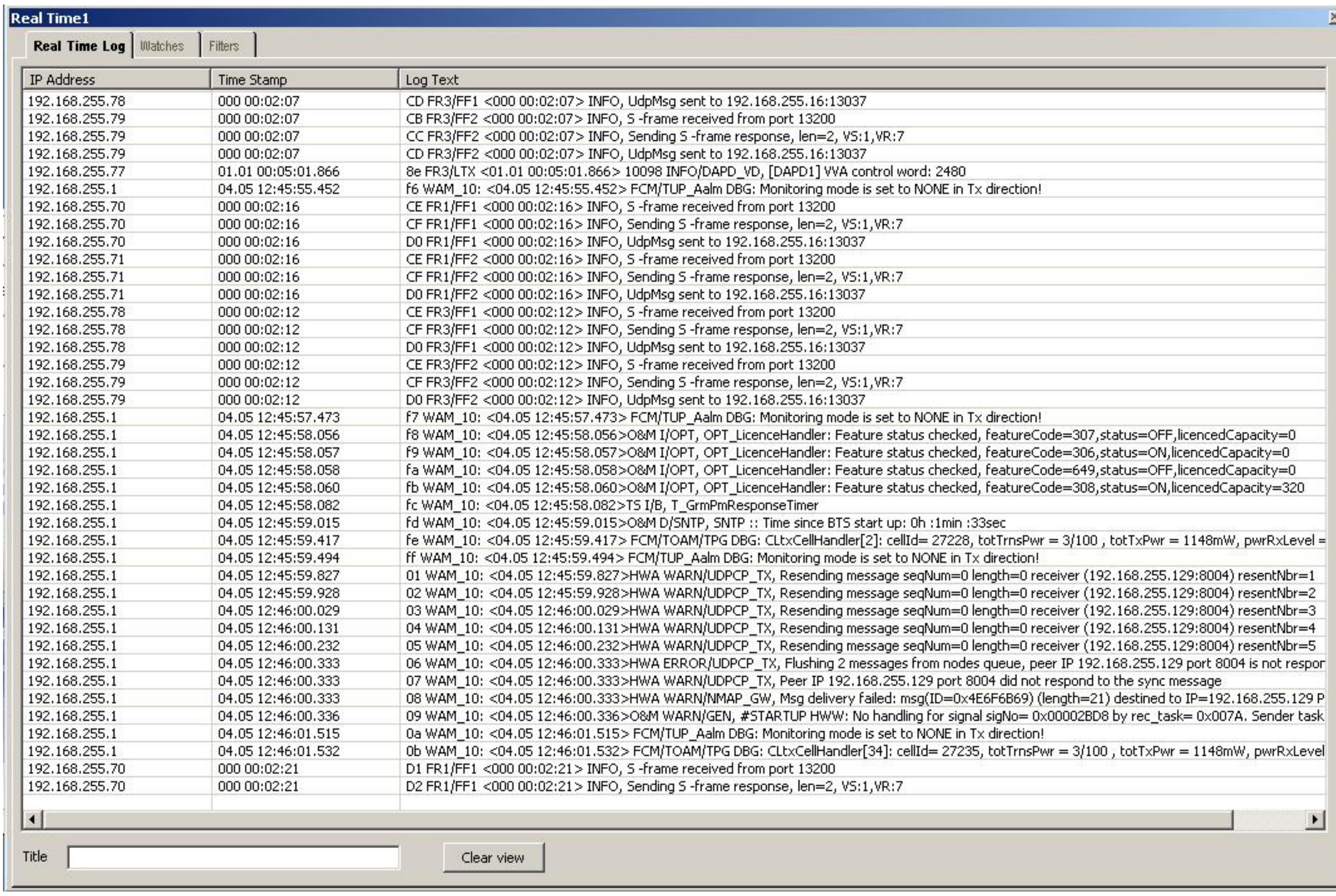
Now WAM_10 should go to TRACING-state.



Nodes and Filters

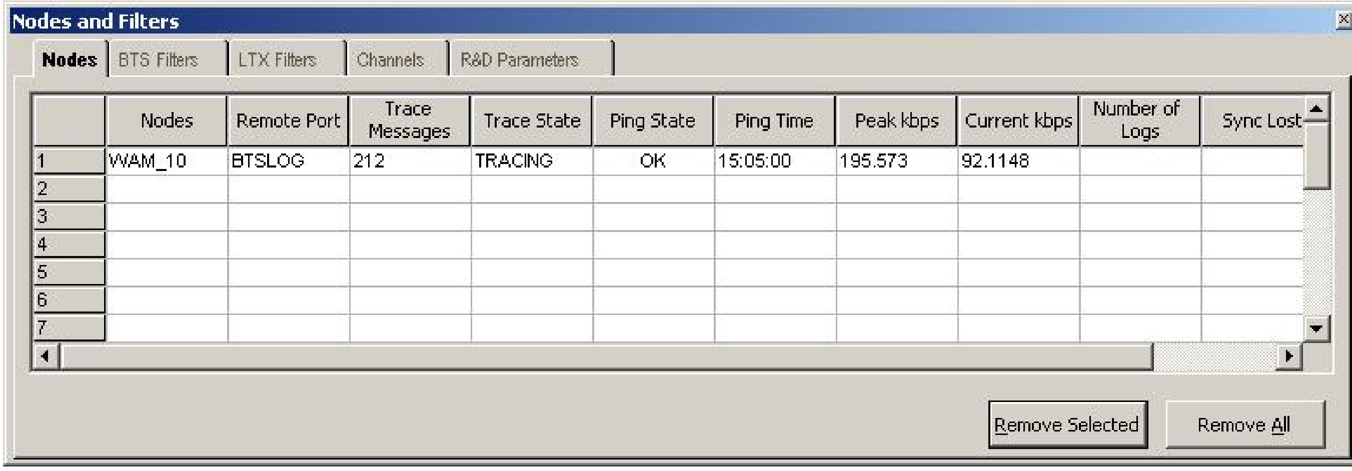
	Nodes	Remote Port	Trace Messages	Trace State	Ping State	Ping Time	Peak kbps	Current kbps	Number of Logs	Sy
1	WAM_10	BTSLOG	0	TRACING	OK	16:28:28	0	0		
2										
3										

You should also be able to see prints in realtime window.
(RT-windows open from buttons 1,2,3,4)



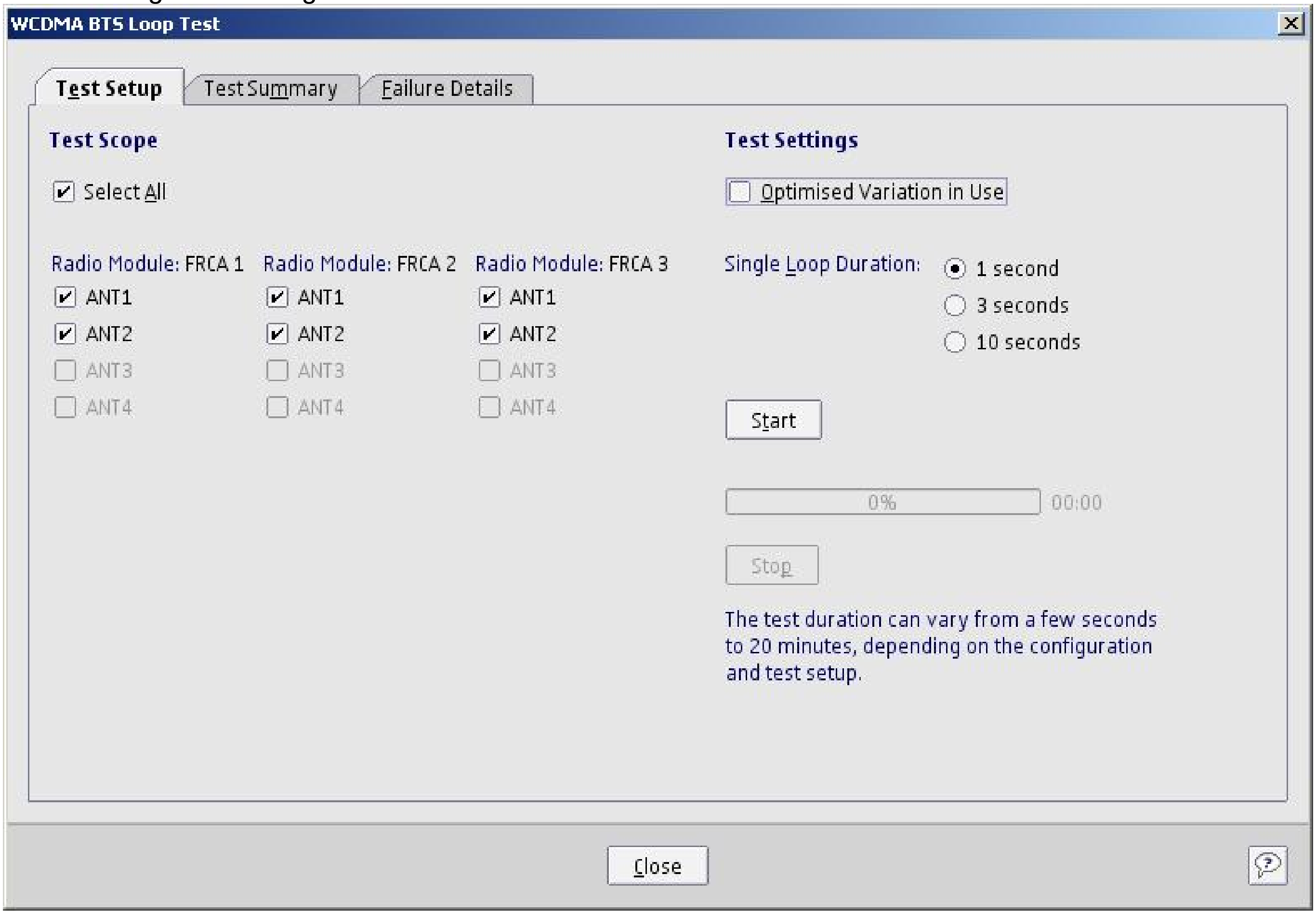
IP Address	Time Stamp	Log Text
192.168.255.78	000 00:02:07	CD FR3/FF1 <000 00:02:07> INFO, UdpMsg sent to 192.168.255.16:13037
192.168.255.79	000 00:02:07	CB FR3/FF2 <000 00:02:07> INFO, S -frame received from port 13200
192.168.255.79	000 00:02:07	CC FR3/FF2 <000 00:02:07> INFO, Sending S -frame response, len=2, VS:1,VR:7
192.168.255.79	000 00:02:07	CD FR3/FF2 <000 00:02:07> INFO, UdpMsg sent to 192.168.255.16:13037
192.168.255.77	01.01 00:05:01.866	8e FR3/LTX <01.01 00:05:01.866> 10098 INFO/DAPD_VD, [DAPD1] VVA control word: 2480
192.168.255.1	04.05 12:45:55.452	f6 WAM_10: <04.05 12:45:55.452> FCM/TUP_Aalm DBG: Monitoring mode is set to NONE in Tx direction!
192.168.255.70	000 00:02:16	CE FR1/FF1 <000 00:02:16> INFO, S -frame received from port 13200
192.168.255.70	000 00:02:16	CF FR1/FF1 <000 00:02:16> INFO, Sending S -frame response, len=2, VS:1,VR:7
192.168.255.70	000 00:02:16	D0 FR1/FF1 <000 00:02:16> INFO, UdpMsg sent to 192.168.255.16:13037
192.168.255.71	000 00:02:16	CE FR1/FF2 <000 00:02:16> INFO, S -frame received from port 13200
192.168.255.71	000 00:02:16	CF FR1/FF2 <000 00:02:16> INFO, Sending S -frame response, len=2, VS:1,VR:7
192.168.255.71	000 00:02:16	D0 FR1/FF2 <000 00:02:16> INFO, UdpMsg sent to 192.168.255.16:13037
192.168.255.78	000 00:02:12	CE FR3/FF1 <000 00:02:12> INFO, S -frame received from port 13200
192.168.255.78	000 00:02:12	CF FR3/FF1 <000 00:02:12> INFO, Sending S -frame response, len=2, VS:1,VR:7
192.168.255.78	000 00:02:12	D0 FR3/FF1 <000 00:02:12> INFO, UdpMsg sent to 192.168.255.16:13037
192.168.255.79	000 00:02:12	CE FR3/FF2 <000 00:02:12> INFO, S -frame received from port 13200
192.168.255.79	000 00:02:12	CF FR3/FF2 <000 00:02:12> INFO, Sending S -frame response, len=2, VS:1,VR:7
192.168.255.79	000 00:02:12	D0 FR3/FF2 <000 00:02:12> INFO, UdpMsg sent to 192.168.255.16:13037
192.168.255.1	04.05 12:45:57.473	f7 WAM_10: <04.05 12:45:57.473> FCM/TUP_Aalm DBG: Monitoring mode is set to NONE in Tx direction!
192.168.255.1	04.05 12:45:58.056	f8 WAM_10: <04.05 12:45:58.056>O&M I/OPT, OPT_LicenceHandler: Feature status checked, featureCode=307,status=OFF,licencedCapacity=0
192.168.255.1	04.05 12:45:58.057	f9 WAM_10: <04.05 12:45:58.057>O&M I/OPT, OPT_LicenceHandler: Feature status checked, featureCode=306,status=ON,licencedCapacity=0
192.168.255.1	04.05 12:45:58.058	fa WAM_10: <04.05 12:45:58.058>O&M I/OPT, OPT_LicenceHandler: Feature status checked, featureCode=649,status=OFF,licencedCapacity=0
192.168.255.1	04.05 12:45:58.060	fb WAM_10: <04.05 12:45:58.060>O&M I/OPT, OPT_LicenceHandler: Feature status checked, featureCode=308,status=ON,licencedCapacity=320
192.168.255.1	04.05 12:45:58.082	fc WAM_10: <04.05 12:45:58.082>TS I/B, T_GrmPmResponseTimer
192.168.255.1	04.05 12:45:59.015	fd WAM_10: <04.05 12:45:59.015>O&M D/SNTP, SNTP :: Time since BTS start up: 0h :1min :33sec
192.168.255.1	04.05 12:45:59.417	fe WAM_10: <04.05 12:45:59.417> FCM/TOAM/TPG DBG: CLtxCellHandler[2]: cellId= 27228, totTrnsPwr = 3/100 , totTxPwr = 1148mW, pwrRxLevel =
192.168.255.1	04.05 12:45:59.494	ff WAM_10: <04.05 12:45:59.494> FCM/TUP_Aalm DBG: Monitoring mode is set to NONE in Tx direction!
192.168.255.1	04.05 12:45:59.827	01 WAM_10: <04.05 12:45:59.827>HWA WARN/UDPCP_TX, Resending message seqNum=0 length=0 receiver (192.168.255.129:8004) resentNbr=1
192.168.255.1	04.05 12:45:59.928	02 WAM_10: <04.05 12:45:59.928>HWA WARN/UDPCP_TX, Resending message seqNum=0 length=0 receiver (192.168.255.129:8004) resentNbr=2
192.168.255.1	04.05 12:46:00.029	03 WAM_10: <04.05 12:46:00.029>HWA WARN/UDPCP_TX, Resending message seqNum=0 length=0 receiver (192.168.255.129:8004) resentNbr=3
192.168.255.1	04.05 12:46:00.131	04 WAM_10: <04.05 12:46:00.131>HWA WARN/UDPCP_TX, Resending message seqNum=0 length=0 receiver (192.168.255.129:8004) resentNbr=4
192.168.255.1	04.05 12:46:00.232	05 WAM_10: <04.05 12:46:00.232>HWA WARN/UDPCP_TX, Resending message seqNum=0 length=0 receiver (192.168.255.129:8004) resentNbr=5
192.168.255.1	04.05 12:46:00.333	06 WAM_10: <04.05 12:46:00.333>HWA ERROR/UDPCP_TX, Flushing 2 messages from nodes queue, peer IP 192.168.255.129 port 8004 is not respor
192.168.255.1	04.05 12:46:00.333	07 WAM_10: <04.05 12:46:00.333>HWA WARN/UDPCP_TX, Peer IP 192.168.255.129 port 8004 did not respond to the sync message
192.168.255.1	04.05 12:46:00.333	08 WAM_10: <04.05 12:46:00.333>HWA WARN/NMAP_GW, Msg delivery failed: msg(ID=0x4E6F6B69) (length=21) destined to IP=192.168.255.129 P
192.168.255.1	04.05 12:46:00.336	09 WAM_10: <04.05 12:46:00.336>O&M WARN/GEN, #STARTUP HWW: No handling for signal sigNo= 0x00002BD8 by rec_task= 0x007A. Sender task
192.168.255.1	04.05 12:46:01.515	0a WAM_10: <04.05 12:46:01.515> FCM/TUP_Aalm DBG: Monitoring mode is set to NONE in Tx direction!
192.168.255.1	04.05 12:46:01.532	0b WAM_10: <04.05 12:46:01.532> FCM/TOAM/TPG DBG: CLtxCellHandler[34]: cellId= 27235, totTrnsPwr = 3/100 , totTxPwr = 1148mW, pwrRxLevel
192.168.255.70	000 00:02:21	D1 FR1/FF1 <000 00:02:21> INFO, S -frame received from port 13200
192.168.255.70	000 00:02:21	D2 FR1/FF1 <000 00:02:21> INFO, Sending S -frame response, len=2, VS:1,VR:7

Count of received TCP-messages can be seen in 'Trace Messages'-column when loop test is ran.



	Nodes	Remote Port	Trace Messages	Trace State	Ping State	Ping Time	Peak kbps	Current kbps	Number of Logs	Sync Lost
1	WAM_10	BTSLOG	212	TRACING	OK	15:05:00	195.573	92.1148		
2										
3										
4										
5										
6										
7										

7. Now UDP and TCP traces are running.
Open Element Manager menu 'Test/WCDMA Loop Test. Select all, or desired antennas start loops.
Clear the logs in BTSLog with clear button.



Test Scope

☑ Select All

Radio Module: FRCA 1	Radio Module: FRCA 2	Radio Module: FRCA 3
☑ ANT1	☑ ANT1	☑ ANT1
☑ ANT2	☑ ANT2	☑ ANT2
□ ANT3	□ ANT3	□ ANT3
□ ANT4	□ ANT4	□ ANT4

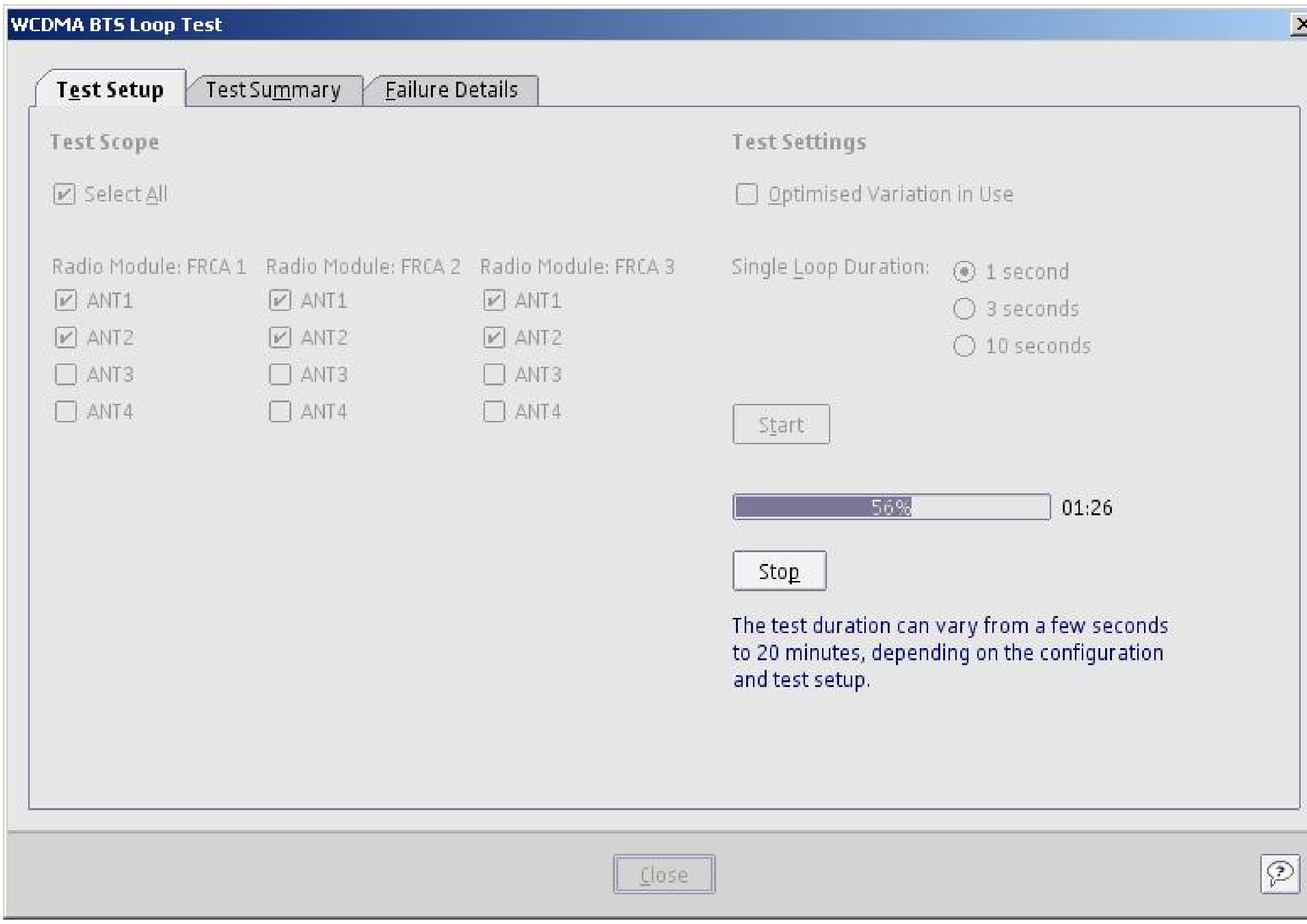
Test Settings

□ Optimised Variation in Use

Single Loop Duration: ● 1 second ○ 3 seconds ○ 10 seconds

The test duration can vary from a few seconds to 20 minutes, depending on the configuration and test setup.

8. Running loops are started
…



WCDMA BTS Loop Test

Test Scope

☑ Select All

Radio Module: FRCA 1	Radio Module: FRCA 2	Radio Module: FRCA 3
☑ ANT1	☑ ANT1	☑ ANT1
☑ ANT2	☑ ANT2	☑ ANT2
☐ ANT3	☐ ANT3	☐ ANT3
☐ ANT4	☐ ANT4	☐ ANT4

Test Settings

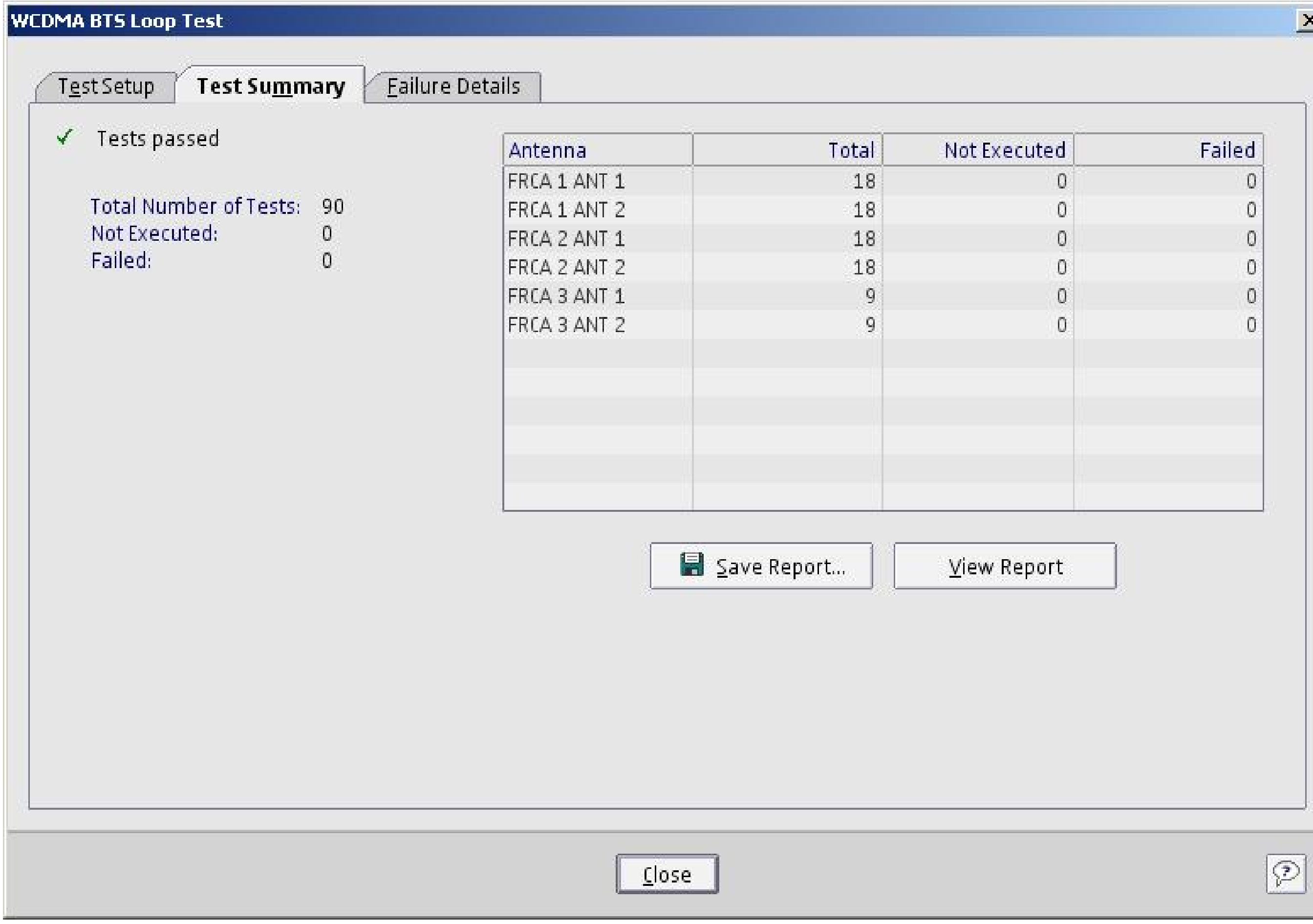
☐ Optimised Variation in Use

Single Loop Duration: ● 1 second ○ 3 seconds ○ 10 seconds

56% 01:26

The test duration can vary from a few seconds to 20 minutes, depending on the configuration and test setup.

9. When all loops are executed, the window seen below opens.



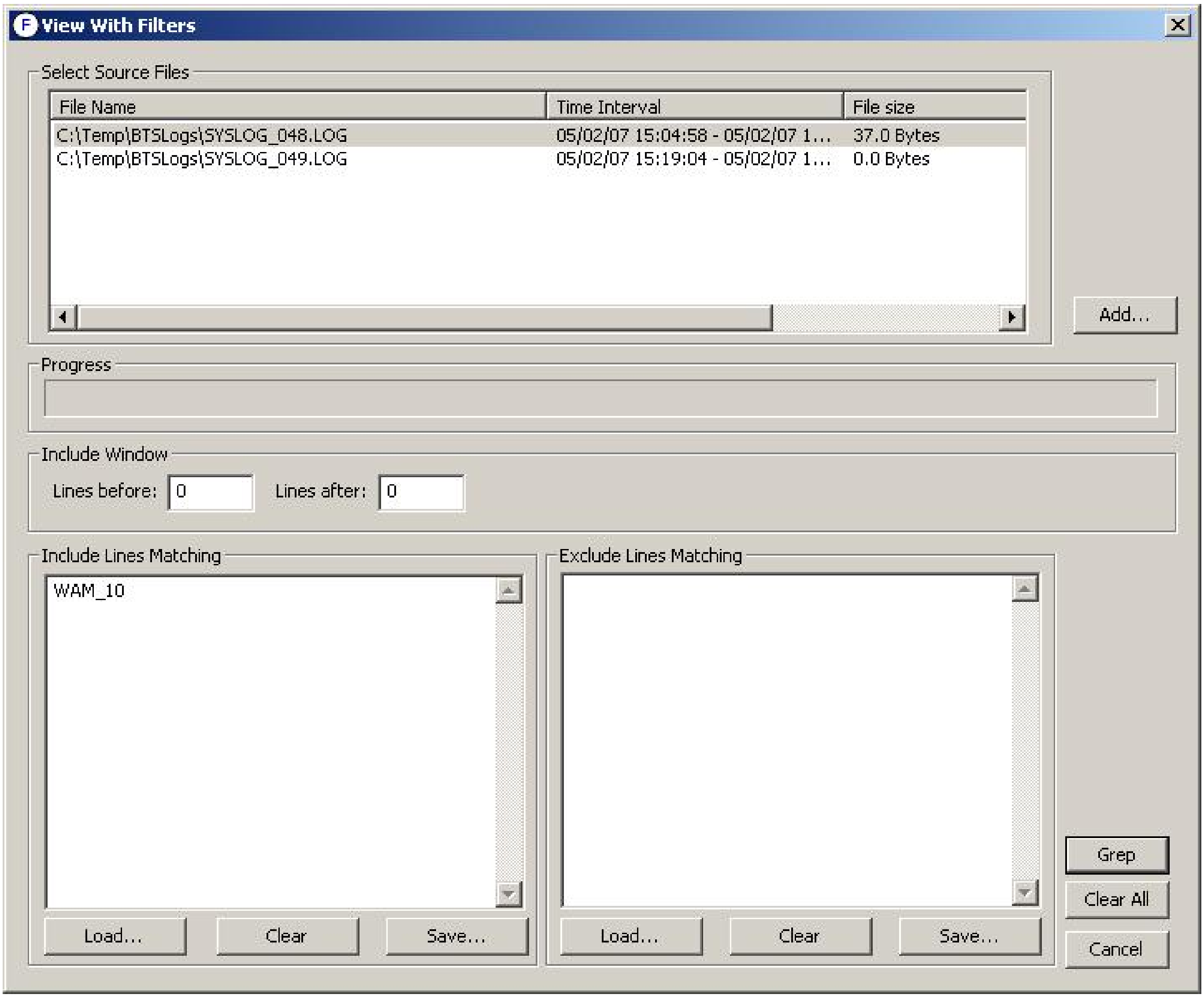
✓ Tests passed

Total Number of Tests:	90
Not Executed:	0
Failed:	0

Antenna	Total	Not Executed	Failed
FRCA 1 ANT 1	18	0	0
FRCA 1 ANT 2	18	0	0
FRCA 2 ANT 1	18	0	0
FRCA 2 ANT 2	18	0	0
FRCA 3 ANT 1	9	0	0
FRCA 3 ANT 2	9	0	0

10. Stop traces in BTSLog by pressing both stop-buttons.
Log files can be found in the directory defined in BTSLog -> tools / setup / Save logs into:<dir>.

You can inspect SYSLOGS with notepad or with any text editor.
If you want to filter the data, you can use
filegrep feature of BTSLog, by pressing 'View with filters'-button on the main window.
You can define keywords to be included or exluded from the ASCII data. Then press Grep
and filtered data is opened to notepad.



File Name	Time Interval	File size
C:\Temp\BTSLogs\SYSLOG_048.LOG	05/02/07 15:04:58 - 05/02/07 1...	37.0 Bytes
C:\Temp\BTSLogs\SYSLOG_049.LOG	05/02/07 15:19:04 - 05/02/07 1...	0.0 Bytes

Lines before: 0 Lines after: 0

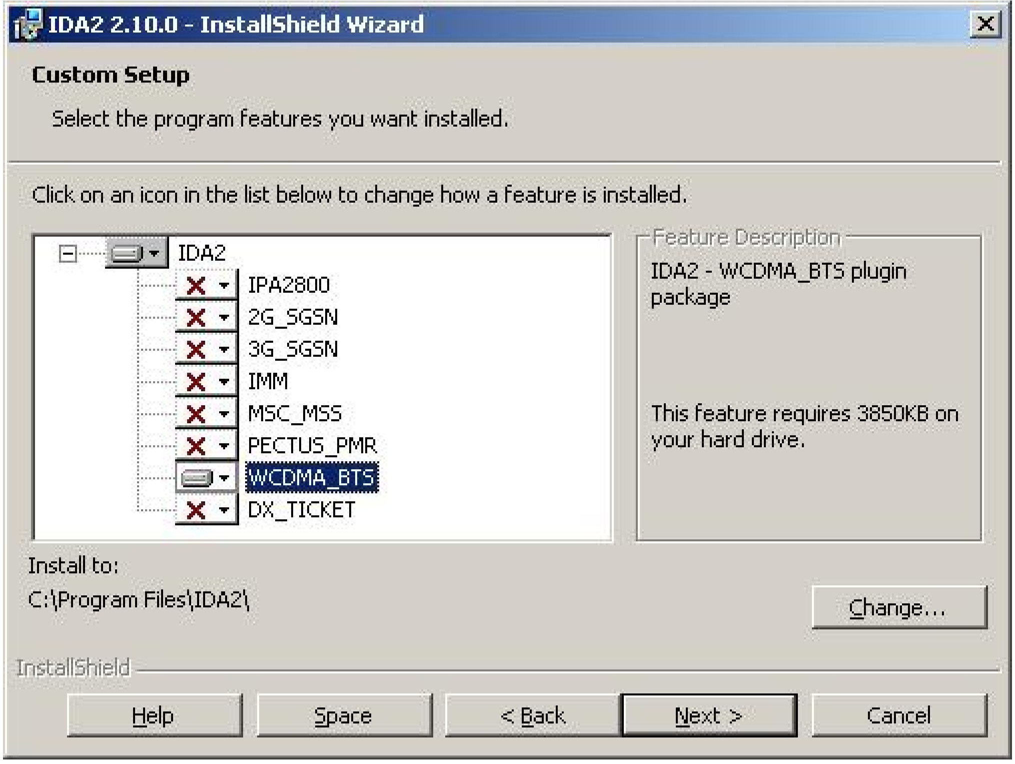
Include Lines Matching: WAM_10

Binary logs are saved in so called Ida2 format. Ida2 is an off line analysis tool for various kind of structured data. There's propably no other tool which is capable of opening BTSLog's trace logs in a readable form.

A brief run through how to install and set up Ida2 for opening log files recorded with BTSlog.

You can load Ida2 from here
https://dominoext.access.nokiasiemensnetworks.com/bi/eeserviceshomepa.nsf/document/ESLNC0026C9ALH?OpenDocument

Start installation. Choose WCDMA_BTS plugins and press next.



IDA2 2.10.0 - InstallShield Wizard

Custom Setup

Select the program features you want installed.

Click on an icon in the list below to change how a feature is installed.

- IDA2
  - ☒ IPA2800
  - ☒ 2G_SGSN
  - ☒ 3G_SGSN
  - ☒ IMM
  - ☒ MSC_MSS
  - ☒ PECTUS_PMR
  - WCDMA_BTS
  - ☒ DX_TICKET

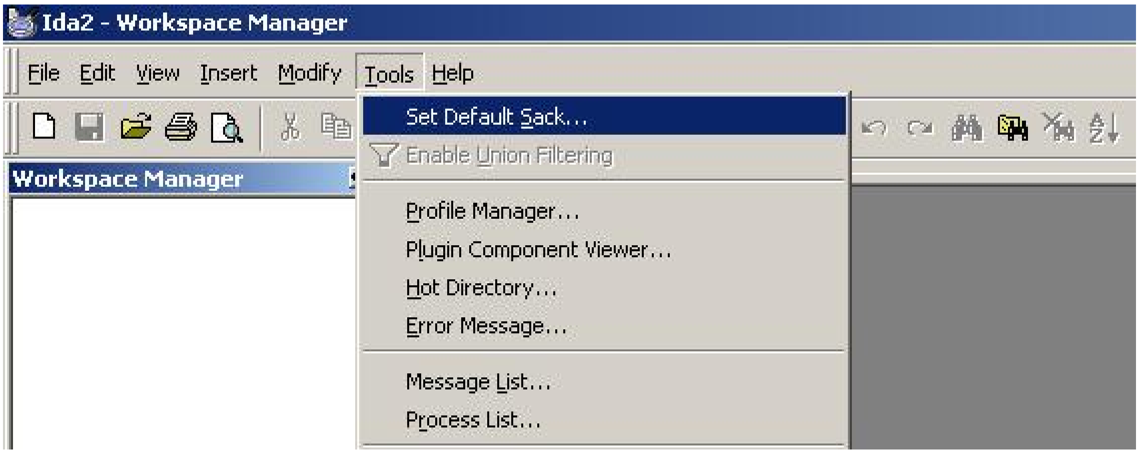
Feature Description

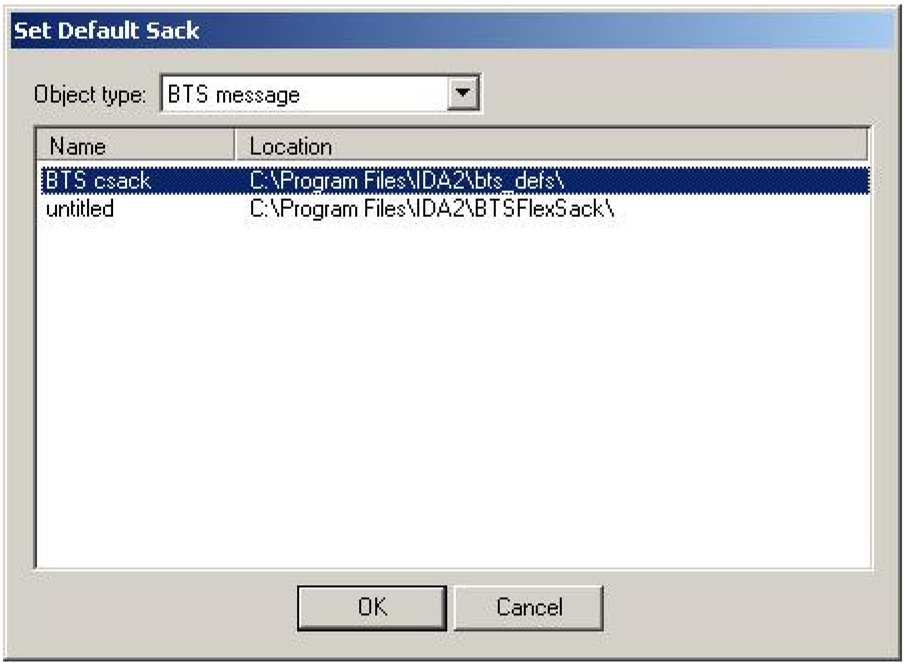
IDA2 - WCDMA_BTS plugin package

This feature requires 3850KB on your hard drive.

Install to:
C:\Program Files\IDA2\

Finish installation and start Ida2. First thing to do before opening BTSLog's tracelogs is to select the correct sack.





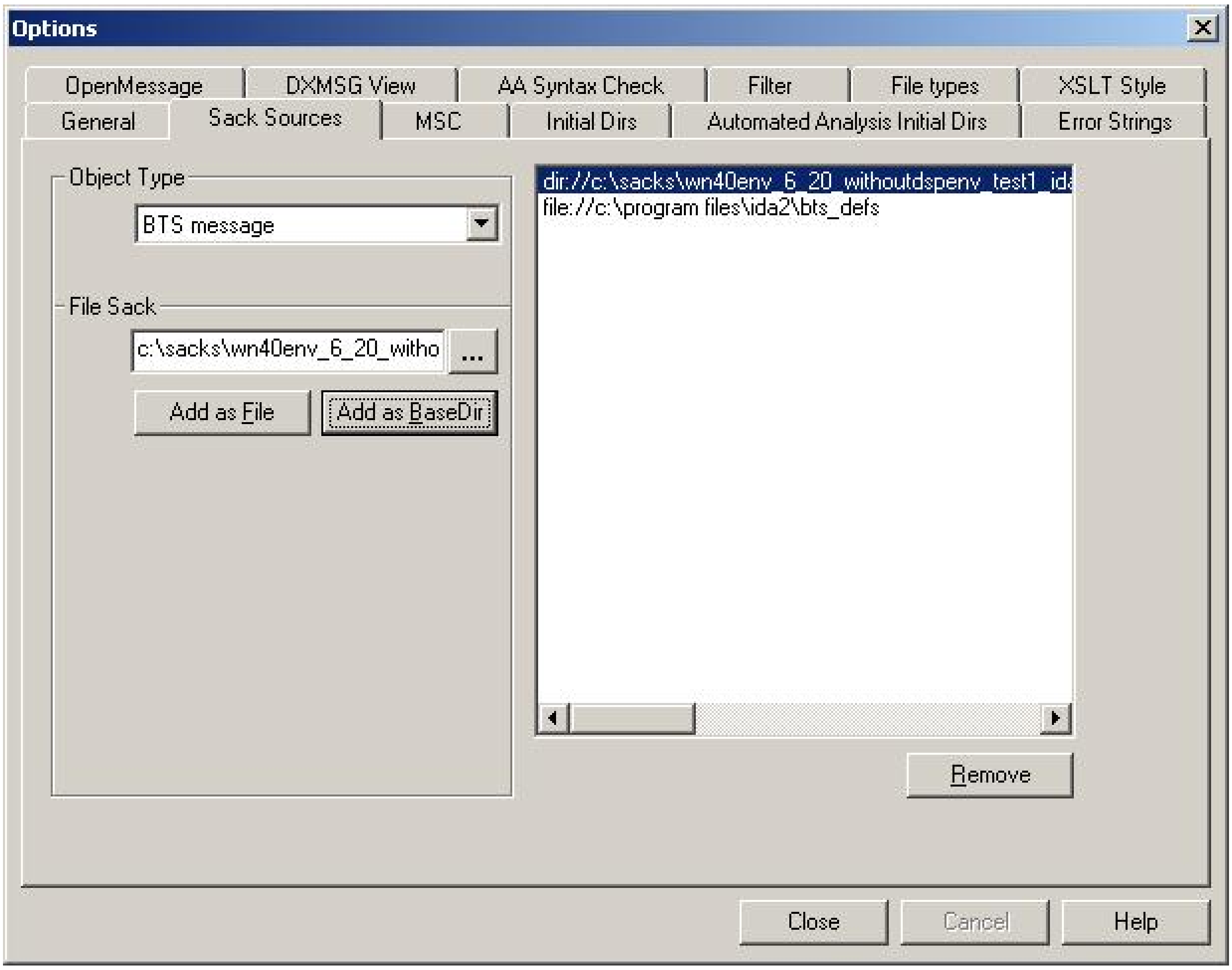
Set Default Sack

Object type: BTS message

Name	Location
BTS csack	C:\Program Files\IDA2\bts_defs\
untitled	C:\Program Files\IDA2\BTSFlexSack\

Press ok and your ready to open a tracelog.

Install package makes a default BTS-sack to c:\Program Files\IDA2\bts_defs.
If you need different sack versions, you can load them here.
\\VAGRP015\Groups1\RAN_BSENV\IDA2_Sacks\
To get a new sack available. Goto Tools/options -> sack sources.
Browse your sack and press 'Add as a File'. (Or 'Add as BaseDir', if you want to add a set of sacks in subdirs)
Press Close.

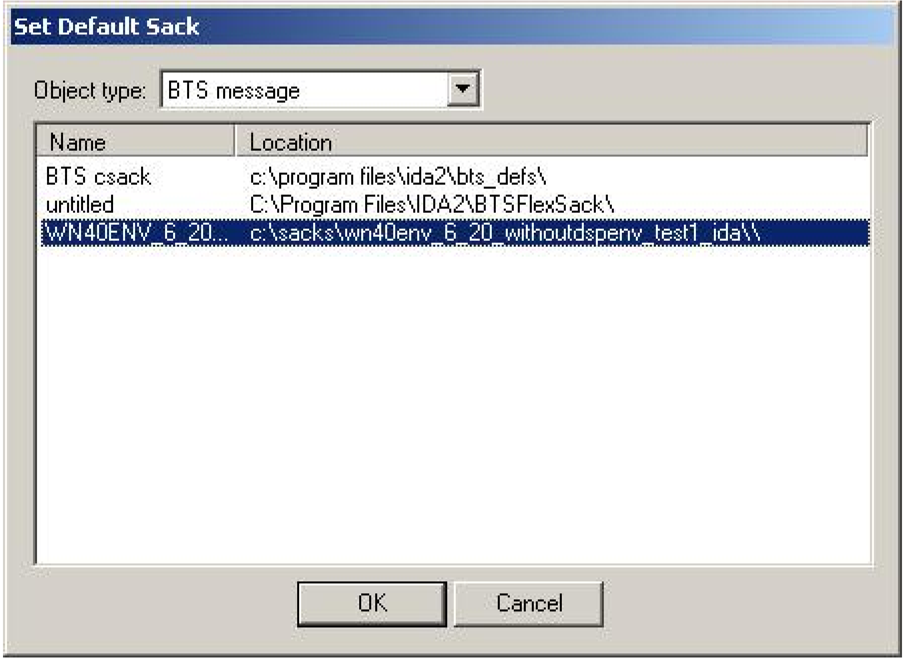


Object Type: BTS message

File Sack: c:\sacks\wn40env_6_20_witho

```
dir://c:\sacks\wn40env_6_20_withoutdspenv_test1_ida
file://c:\program files\ida2\bts_defs
```

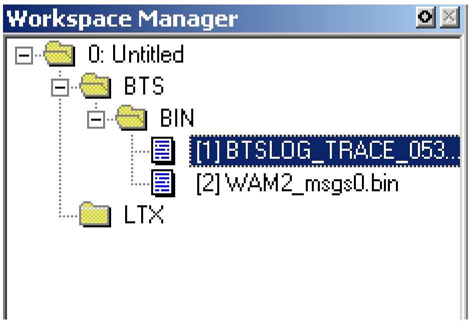
Now you can select a different sack in Tools / Set Default Sack.



Object type: BTS message

Name	Location
BTS csack	c:\program files\ida2\bts_defs\
untitled	C:\Program Files\IDA2\BTSFlexSack\
WN40ENV_6_20...	c:\sacks\wn40env_6_20_withoutdspenv_test1_ida\\

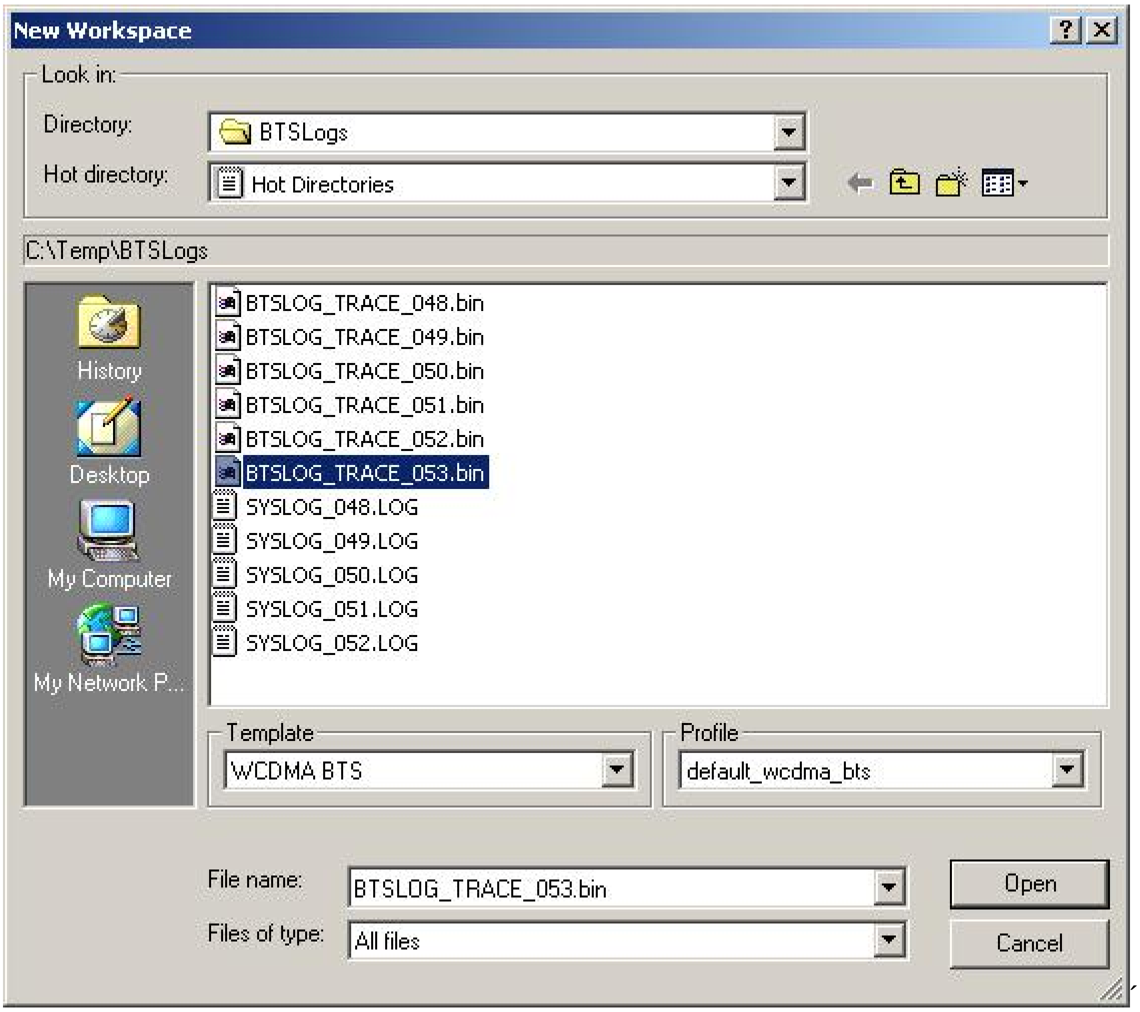
TIP: You can set different sack for each logfile by right clicking the file and selecting view properties.



```
0: Untitled
  BTS
    BIN
      [1] BTSLOG_TRACE_053..
      [2] WAM2_msgs0.bin
  LTX
```

Opening Logs

Choose:
File / New workspace.
Select Template and Profile as shown below.
After opening once with BTS template and profile, you can open current tracelog from BTSLog to Ida2 with Ctrl + i.
TIP: You can also drag & drop logfiles to Ida2.



Directory: BTSLogs

Hot directory: Hot Directories

C:\Temp\BTSLogs

```
BTSLOG_TRACE_048.bin
BTSLOG_TRACE_049.bin
BTSLOG_TRACE_050.bin
BTSLOG_TRACE_051.bin
BTSLOG_TRACE_052.bin
BTSLOG_TRACE_053.bin
SYSLOG_048.LOG
SYSLOG_049.LOG
SYSLOG_050.LOG
SYSLOG_051.LOG
SYSLOG_052.LOG
```

Template: WCDMA BTS

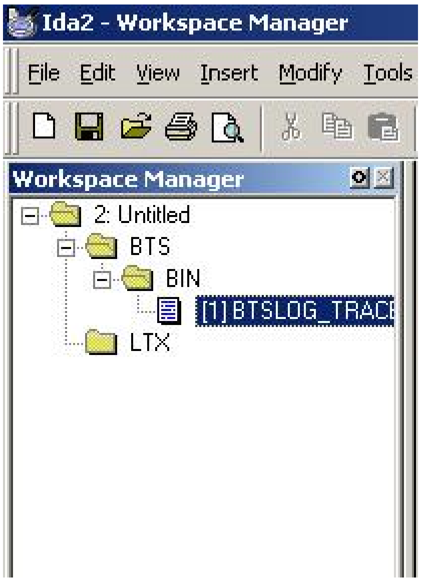
Profile: default_wcdma_bts

File name: BTSLOG_TRACE_053.bin

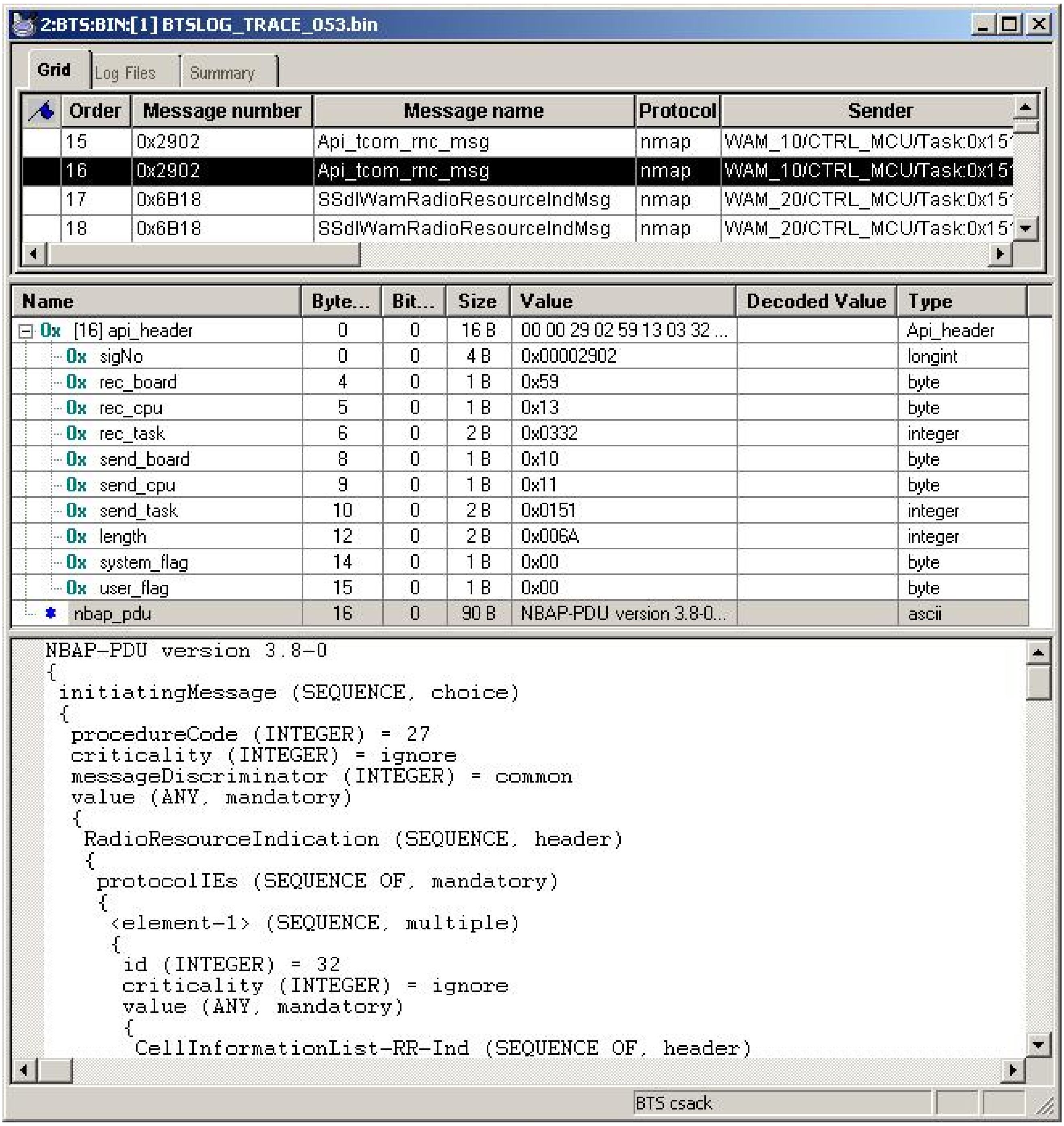
Files of type: All files

Press Open.

A new workspace is created including one tracelog file.
Double click the file and it opens to log window.



Here's an opened tracelog, recorded by Btslog. Double click messages in upper pane and their contents are presented in lower panes.



Order	Message number	Message name	Protocol	Sender
15	0x2902	Api_tcom_rnc_msg	nmap	WAM_10/CTRL_MCU/Task:0x15
16	0x2902	Api_tcom_rnc_msg	nmap	WAM_10/CTRL_MCU/Task:0x15
17	0x6B18	SSdlWamRadioResourceIndMsg	nmap	WAM_20/CTRL_MCU/Task:0x15
18	0x6B18	SSdlWamRadioResourceIndMsg	nmap	WAM_20/CTRL_MCU/Task:0x15

Name	Byte...	Bit...	Size	Value	Decoded Value	Type
0x [16] api_header	0	0	16 B	00 00 29 02 59 13 03 32 ...		Api_header
0x sigNo	0	0	4 B	0x00002902		longint
0x rec_board	4	0	1 B	0x59		byte
0x rec_cpu	5	0	1 B	0x13		byte
0x rec_task	6	0	2 B	0x0332		integer
0x send_board	8	0	1 B	0x10		byte
0x send_cpu	9	0	1 B	0x11		byte
0x send_task	10	0	2 B	0x0151		integer
0x length	12	0	2 B	0x006A		integer
0x system_flag	14	0	1 B	0x00		byte
0x user_flag	15	0	1 B	0x00		byte
nbap_pdu	16	0	90 B	NBAP-PDU version 3.8-0...		ascii

```
NBAP-PDU version 3.8-0
{
 initiatingMessage (SEQUENCE, choice)
 {
  procedureCode (INTEGER) = 27
  criticality (INTEGER) = ignore
  messageDiscriminator (INTEGER) = common
  value (ANY, mandatory)
  {
   RadioResourceIndication (SEQUENCE, header)
   {
    protocolIEs (SEQUENCE OF, mandatory)
    {
     <element-1> (SEQUENCE, multiple)
     {
      id (INTEGER) = 32
      criticality (INTEGER) = ignore
      value (ANY, mandatory)
      {
       CellInformationList-RR-Ind (SEQUENCE OF, header)
```

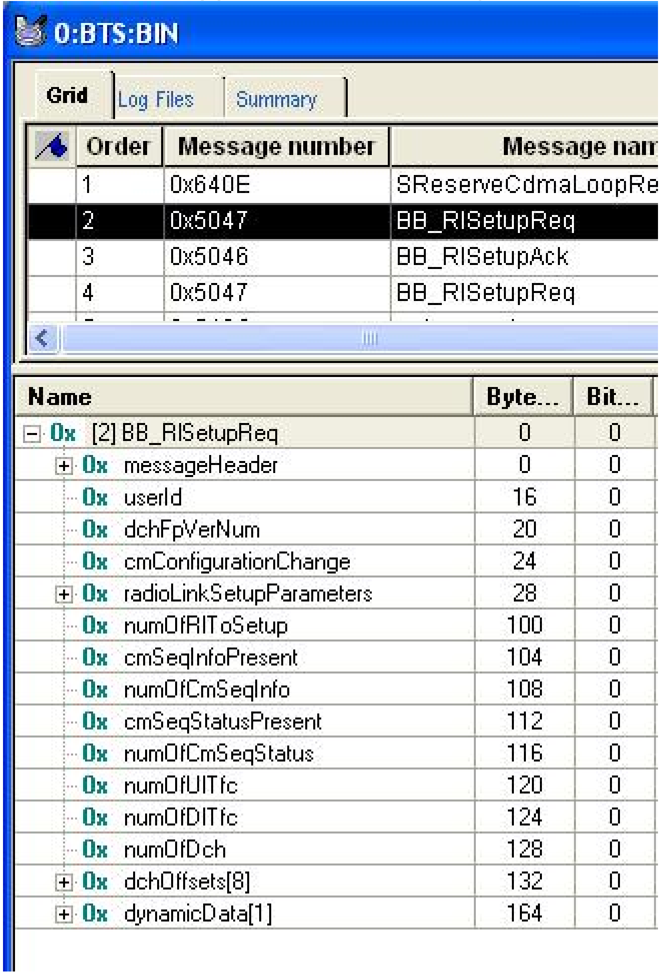
Just for an example, ASN.1 decoded field is selected.

DYNAMIC C - STRUCT FIELDS & CSACK_CONFIG.XML

When dynamic data structures are handled in Ida2, you need an XML-configuration file according to the sack version at hand.
For example: FILENAME : csack_config.xml ver. wn4_inc5#5.3

Load the csack_config.xml from the sack server...
\\VAGRP015\Groups1\RAN_BSENV\IDA2_Sacks\
...and copy it to C:\Program Files\IDA2\config.

Here's what happens, if csack_config.xml is missing.

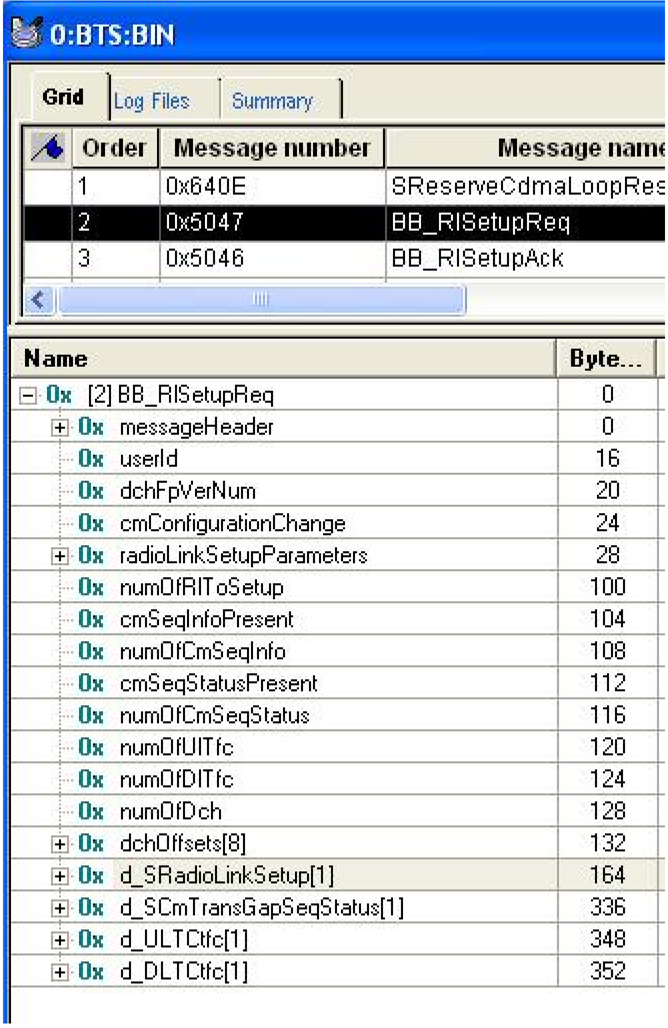


Order	Message number	Message nam
1	0x640E	SReserveCdmaLoopRe
2	0x5047	BB_RlSetupReq
3	0x5046	BB_RlSetupAck
4	0x5047	BB_RlSetupReq

Name	Byte...	Bit...
0x [2] BB_RlSetupReq	0	0
0x messageHeader	0	0
0x userId	16	0
0x dchFpVerNum	20	0
0x cmConfigurationChange	24	0
0x radioLinkSetupParameters	28	0
0x numOfRlToSetup	100	0
0x cmSeqInfoPresent	104	0
0x numOfCmSeqInfo	108	0
0x cmSeqStatusPresent	112	0
0x numOfCmSeqStatus	116	0
0x numOfUlTfc	120	0
0x numOfDlTfc	124	0
0x numOfDch	128	0
0x dchOffsets[8]	132	0
0x dynamicData[1]	164	0

In the end of the message can be seen the array of dynamic data, but the data is not decoded, because there's no csack_config.xml.

This is the same message, with the csack_config.xml copied to Ida2\config\.

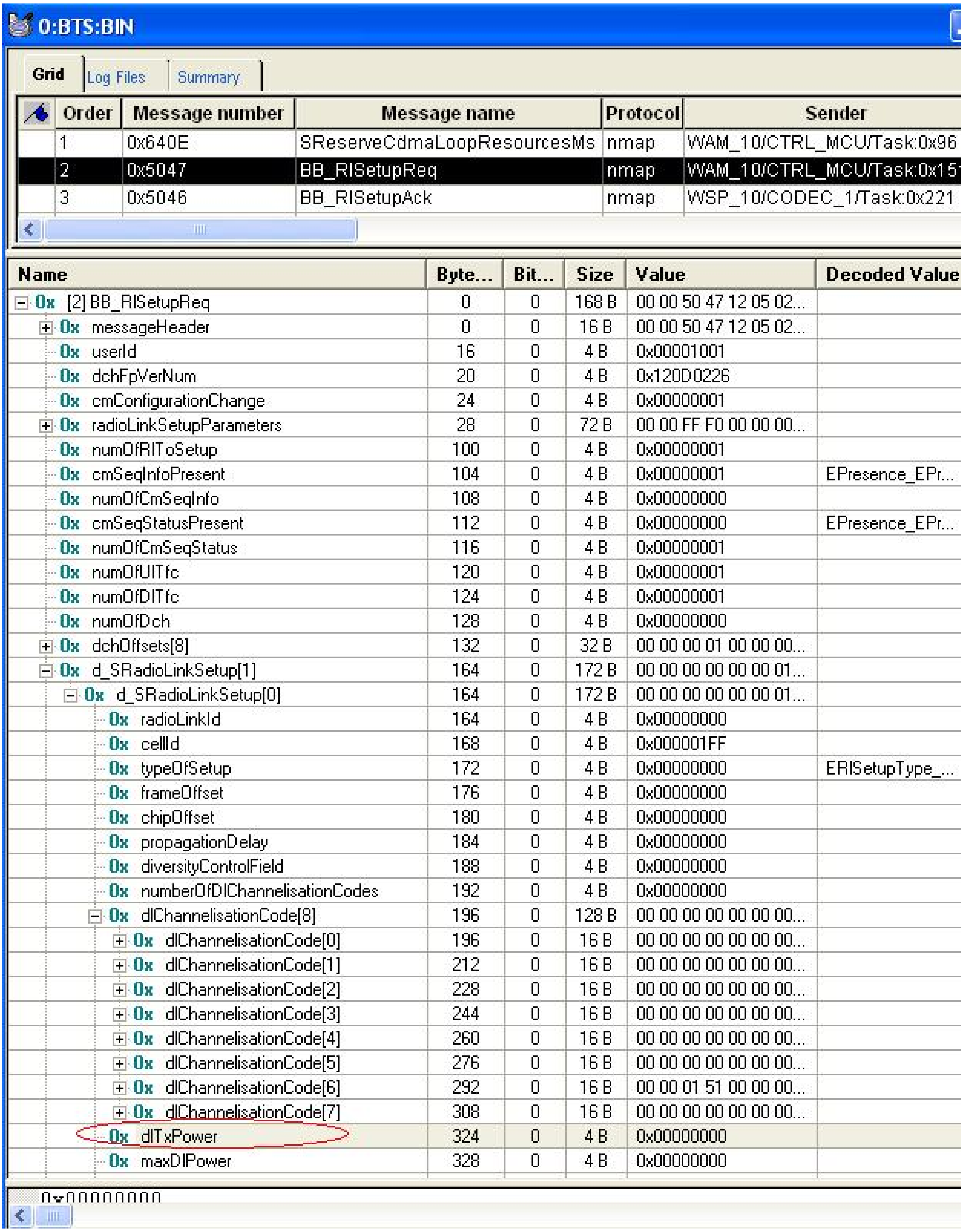


Order	Message number	Message name
1	0x640E	SReserveCdmaLoopRes
2	0x5047	BB_RlSetupReq
3	0x5046	BB_RlSetupAck

Name	Byte...
0x [2] BB_RlSetupReq	0
0x messageHeader	0
0x userId	16
0x dchFpVerNum	20
0x cmConfigurationChange	24
0x radioLinkSetupParameters	28
0x numOfRlToSetup	100
0x cmSeqInfoPresent	104
0x numOfCmSeqInfo	108
0x cmSeqStatusPresent	112
0x numOfCmSeqStatus	116
0x numOfUlTfc	120
0x numOfDlTfc	124
0x numOfDch	128
0x dchOffsets[8]	132
0x d_SRadioLinkSetup[1]	164
0x d_SCmTransGapSeqStatus[1]	336
0x d_ULTCtfc[1]	348
0x d_DLTCtfc[1]	352

Dynamic fields are decoded.

For example, Loop DCH Tx power level is a typical field to inspect in WCDMA-loop test. This field can be seen in the bottom of the picture below when the csack_config.xml is located in ida2\config dir.



Order	Message number	Message name	Protocol	Sender
1	0x640E	SReserveCdmaLoopResourcesMs	nmap	WAM_10/CTRL_MCU/Task:0x96
2	0x5047	BB_RlSetupReq	nmap	WAM_10/CTRL_MCU/Task:0x15
3	0x5046	BB_RlSetupAck	nmap	WSP_10/CODEC_1/Task:0x221

Name	Byte...	Bit...	Size	Value	Decoded Value
0x [2] BB_RlSetupReq	0	0	168 B	00 00 50 47 12 05 02...	
0x messageHeader	0	0	16 B	00 00 50 47 12 05 02...	
0x userId	16	0	4 B	0x00001001	
0x dchFpVerNum	20	0	4 B	0x120D0226	
0x cmConfigurationChange	24	0	4 B	0x00000001	
0x radioLinkSetupParameters	28	0	72 B	00 00 FF F0 00 00 00...	
0x numOfRlToSetup	100	0	4 B	0x00000001	
0x cmSeqInfoPresent	104	0	4 B	0x00000001	EPresence_EPr...
0x numOfCmSeqInfo	108	0	4 B	0x00000000	
0x cmSeqStatusPresent	112	0	4 B	0x00000000	EPresence_EPr...
0x numOfCmSeqStatus	116	0	4 B	0x00000001	
0x numOfUlTfc	120	0	4 B	0x00000001	
0x numOfDlTfc	124	0	4 B	0x00000001	
0x numOfDch	128	0	4 B	0x00000000	
0x dchOffsets[8]	132	0	32 B	00 00 00 01 00 00 00...	
0x d_SRadioLinkSetup[1]	164	0	172 B	00 00 00 00 00 00 01...	
0x d_SRadioLinkSetup[0]	164	0	172 B	00 00 00 00 00 00 01...	
0x radioLinkId	164	0	4 B	0x00000000	
0x cellId	168	0	4 B	0x000001FF	
0x typeOfSetup	172	0	4 B	0x00000000	ERlSetupType_...
0x frameOffset	176	0	4 B	0x00000000	
0x chipOffset	180	0	4 B	0x00000000	
0x propagationDelay	184	0	4 B	0x00000000	
0x diversityControlField	188	0	4 B	0x00000000	
0x numberOfDlChannelisationCodes	192	0	4 B	0x00000000	
0x dlChannelisationCode[8]	196	0	128 B	00 00 00 00 00 00 00...	
0x dlChannelisationCode[0]	196	0	16 B	00 00 00 00 00 00 00...	
0x dlChannelisationCode[1]	212	0	16 B	00 00 00 00 00 00 00...	
0x dlChannelisationCode[2]	228	0	16 B	00 00 00 00 00 00 00...	
0x dlChannelisationCode[3]	244	0	16 B	00 00 00 00 00 00 00...	
0x dlChannelisationCode[4]	260	0	16 B	00 00 00 00 00 00 00...	
0x dlChannelisationCode[5]	276	0	16 B	00 00 00 00 00 00 00...	
0x dlChannelisationCode[6]	292	0	16 B	00 00 01 51 00 00 00...	
0x dlChannelisationCode[7]	308	0	16 B	00 00 00 00 00 00 00...	
0x dlTxPower	324	0	4 B	0x00000000	
0x maxDlPower	328	0	4 B	0x00000000	

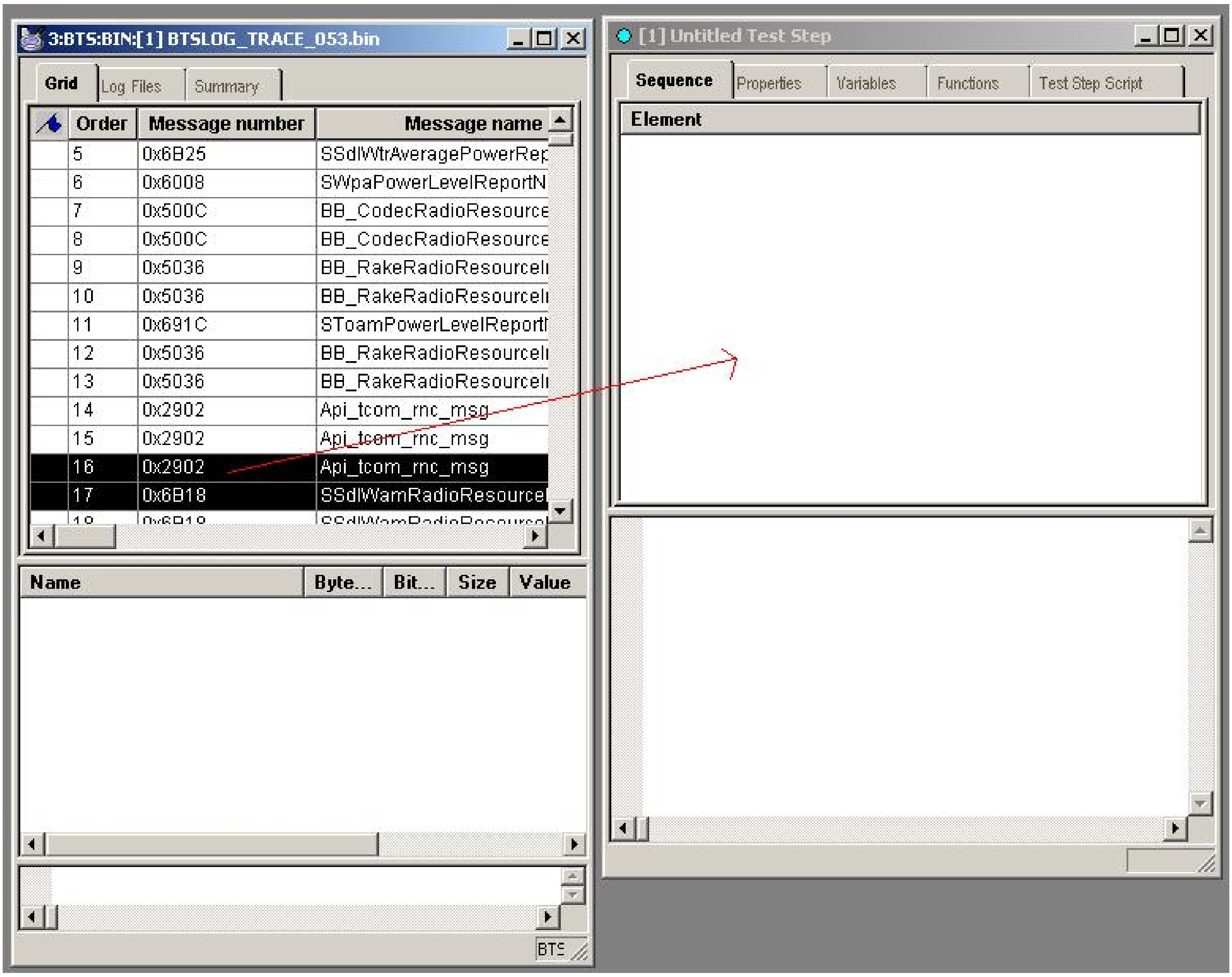
0x00000000

Avoiding futile work

You can browse message structures manually with Ida2.
How ever, when there's lots of data or same maneuvers are constantly repeated, learning *Ida2 Automated Analysis (AA)* is recommended at least at basic level.

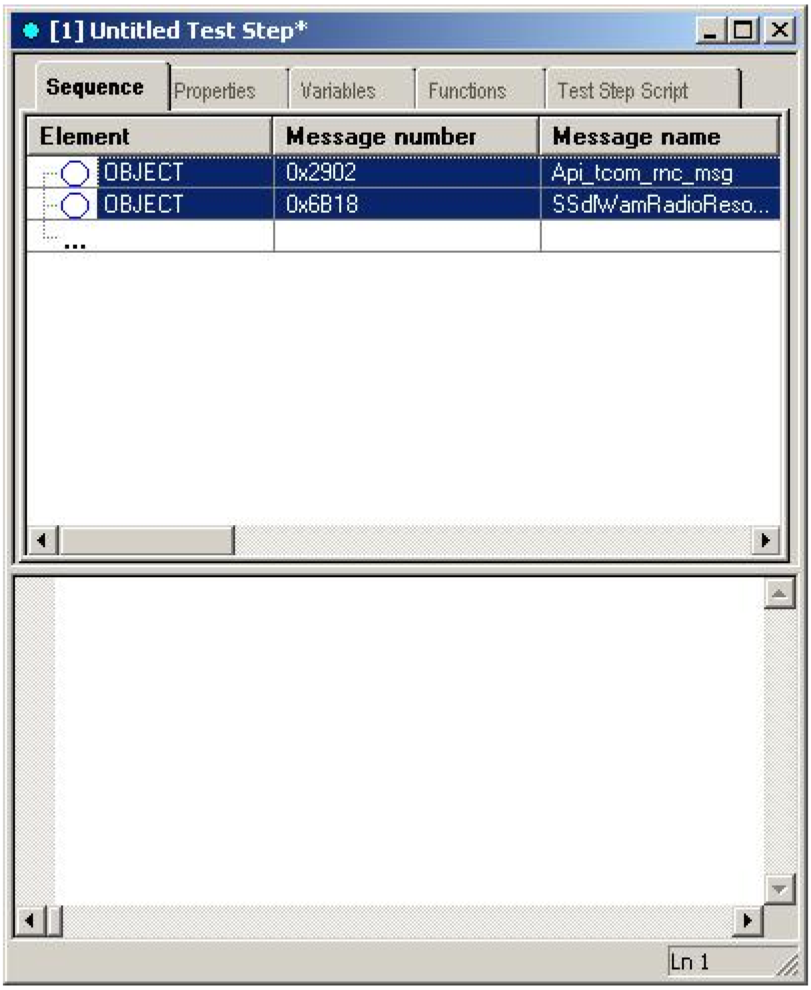
AA can find any kind of a field automatically and gives PASSED or FAILED verdict as a result. Ida2 install package includes complete tutorials about Ida2 AA.

Here's a simple example of using Ida2 AA.
Say, if you need to check that messages 16 and 17 exist in the log in chronological order.
Press F12, which opens so called 'Test Step' window.



Order	Message number	Message name
5	0x6B25	SSdlWtrAveragePowerRep
6	0x6008	SWpaPowerLevelReportN
7	0x500C	BB_CodecRadioResource
8	0x500C	BB_CodecRadioResource
9	0x5036	BB_RakeRadioResourceI
10	0x5036	BB_RakeRadioResourceI
11	0x691C	SToamPowerLevelReportI
12	0x5036	BB_RakeRadioResourceI
13	0x5036	BB_RakeRadioResourceI
14	0x2902	Api_tcom_rnc_msg
15	0x2902	Api_tcom_rnc_msg
16	0x2902	Api_tcom_rnc_msg
17	0x6B18	SSdlWamRadioResource
18	0x6B18	SSdlWamRadioResource

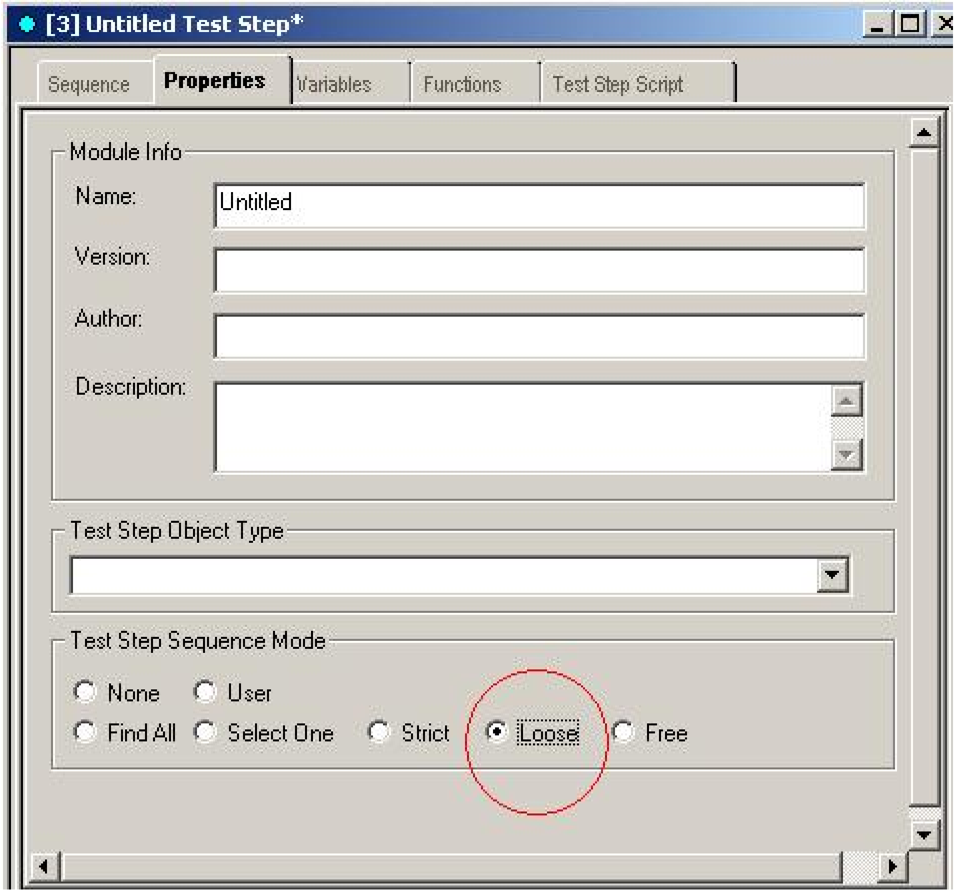
Drag & Drop messages 16 and 17 to upper pane of test step window...
(TIP: to get the cursor into drag mode, move it close to border of the grid cell and a box appears under cursor.)



Element	Message number	Message name
OBJECT	0x2902	Api_tcom_rnc_msg
OBJECT	0x6B18	SSdlWamRadioReso...
...		

Now you've got a test step with two objects.

Before Test Step is valid for running, there's one mandatory thing to do.
Goto properties page and select a **sequence mode**. Make it loose this time.
It means that those two messages under inspection have to appear chronologically in the log file, but can contain other messages before, after or in between.



[3] Untitled Test Step*

Module Info

Name: Untitled

Version:

Author:

Description:

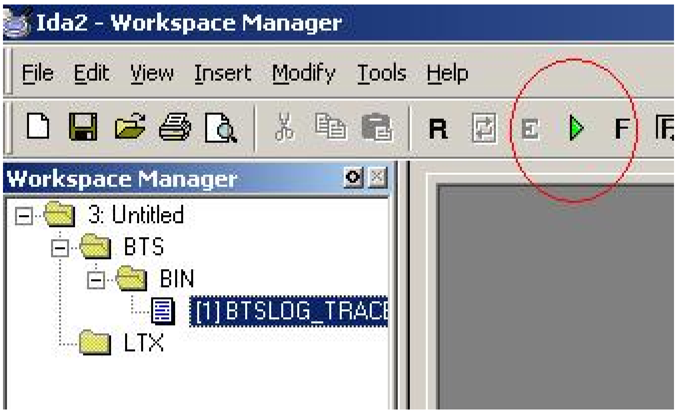
Test Step Object Type

Test Step Sequence Mode

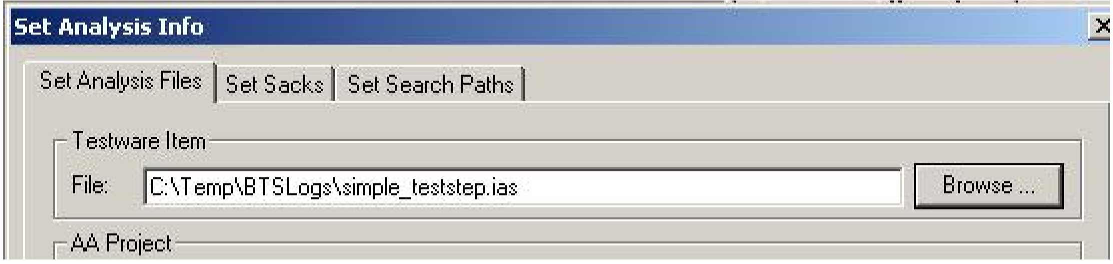
○ None ○ User

○ Find All ○ Select One ○ Strict ● Loose ○ Free

Save the TestStep.
Select the file leaf from workspace tree and an arrow in the toolbar turns to green and is now enabled. Press the arrow.

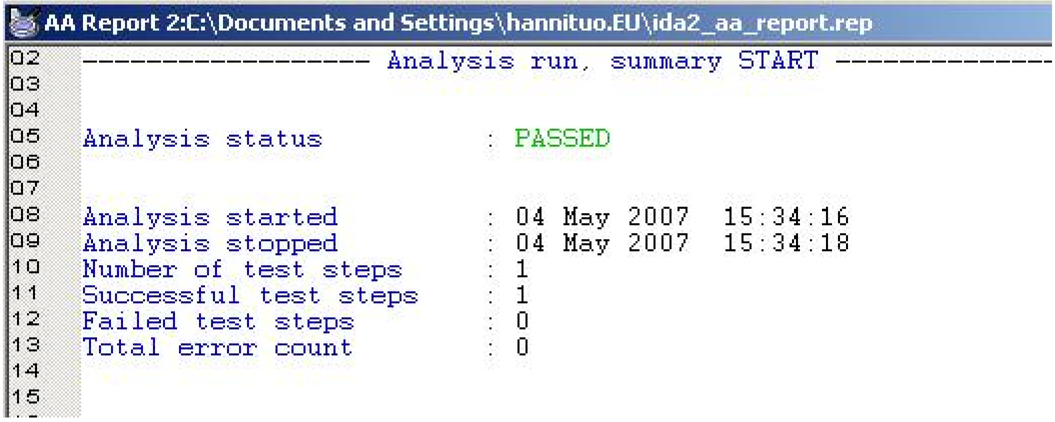


Browse the previously saved teststep and press ok in the bottom of the dialog.



File: C:\Temp\BTSLogs\simple_teststep.ias

When analysis completes, close the analyzer info dialog.
As a result you see the report.


AA Report 2:C:\Documents and Settings\hannituo.EU\ida2_aa_report.rep

```
------------------- Analysis run, summary START ---------------


Analysis status           : PASSED


Analysis started          : 04 May 2007  15:34:16
Analysis stopped          : 04 May 2007  15:34:18
Number of test steps      : 1
Successful test steps     : 1
Failed test steps         : 0
Total error count         : 0
```

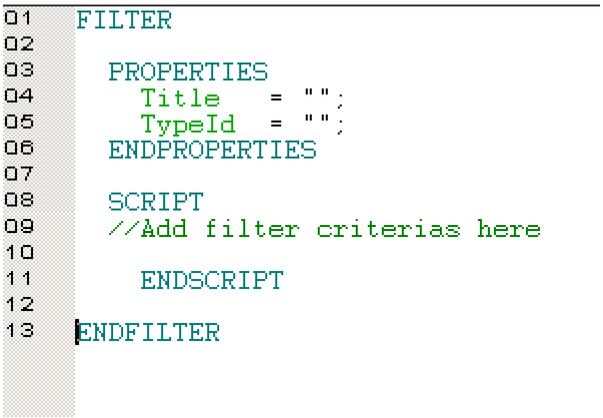
That's about it. To get more advantage of the AA, see the tutorials, but just to mention, Ida2 AA is designed to be co-operative with any other tools or languages.
System function executeapp() in Ida2 AA scripting language enables utilizing other programs during the AA execution.
For example Ida2 can run HIT macros - and vice versa.
You can utilize Perl, Python, Java, VB interpreters or anything that suits the situation.

You can also filter out irrelevant data from the trace logs with Ida2 AA.

Example of filtering log files With AA.

Typically main part of the data in trace logs is not interesting. There's a simple way to filter irrelevant data away.

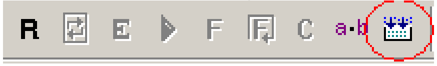
Choose File / New Filter.

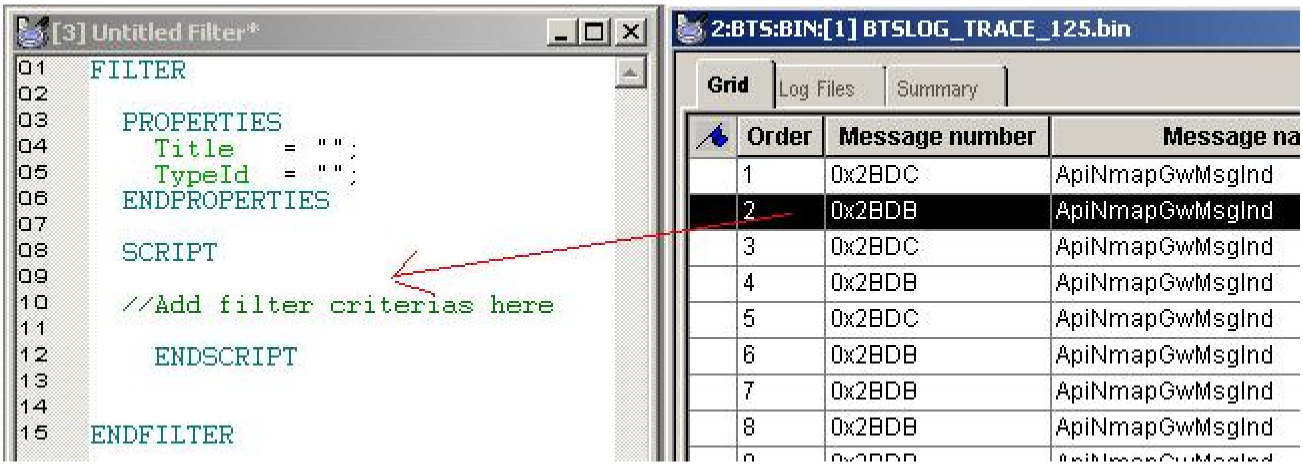


```
FILTER

  PROPERTIES
    Title    = "";
    TypeId   = "";
  ENDPROPERTIES

  SCRIPT
  //Add filter criterias here

    ENDSCRIPT

ENDFILTER
```

Drag & drop desired message to filter window.
TIP: text is dropped exactly to the point of drop. Drop it between tags SCRIPT and ENDSCRIPT

TIP: You can run syntax check of the script with the rounded toolbar button

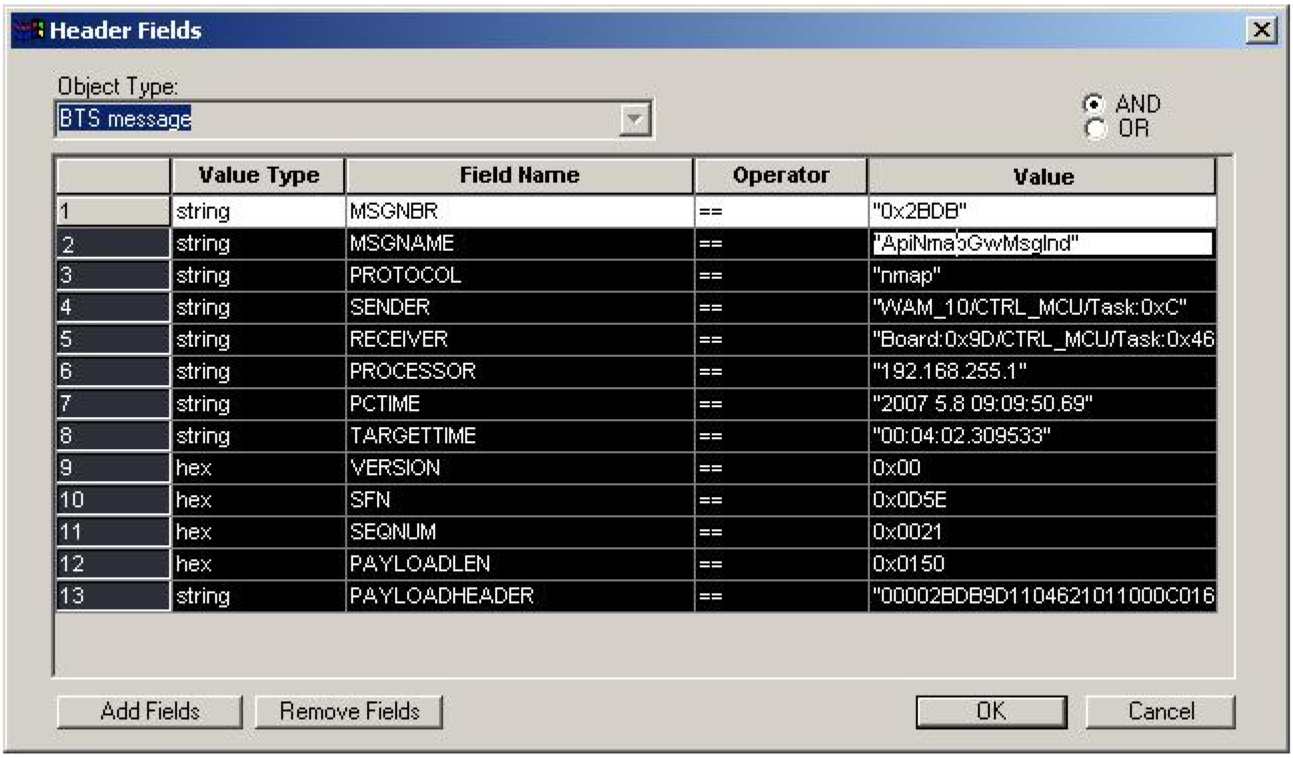




```
FILTER

  PROPERTIES
    Title    = "";
    TypeId   = "";
  ENDPROPERTIES

  SCRIPT

  //Add filter criterias here

    ENDSCRIPT


ENDFILTER
```

Order	Message number	Message na
1	0x2BDC	ApiNmapGwMsgInd
2	0x2BDB	ApiNmapGwMsgInd
3	0x2BDC	ApiNmapGwMsgInd
4	0x2BDB	ApiNmapGwMsgInd
5	0x2BDC	ApiNmapGwMsgInd
6	0x2BDB	ApiNmapGwMsgInd
7	0x2BDB	ApiNmapGwMsgInd
8	0x2BDB	ApiNmapGwMsgInd

Now you get to choose what fields will be the filtering criteria. This time we are interested only in MSGNBR. Other fields are removed with 'Remove Field' command.

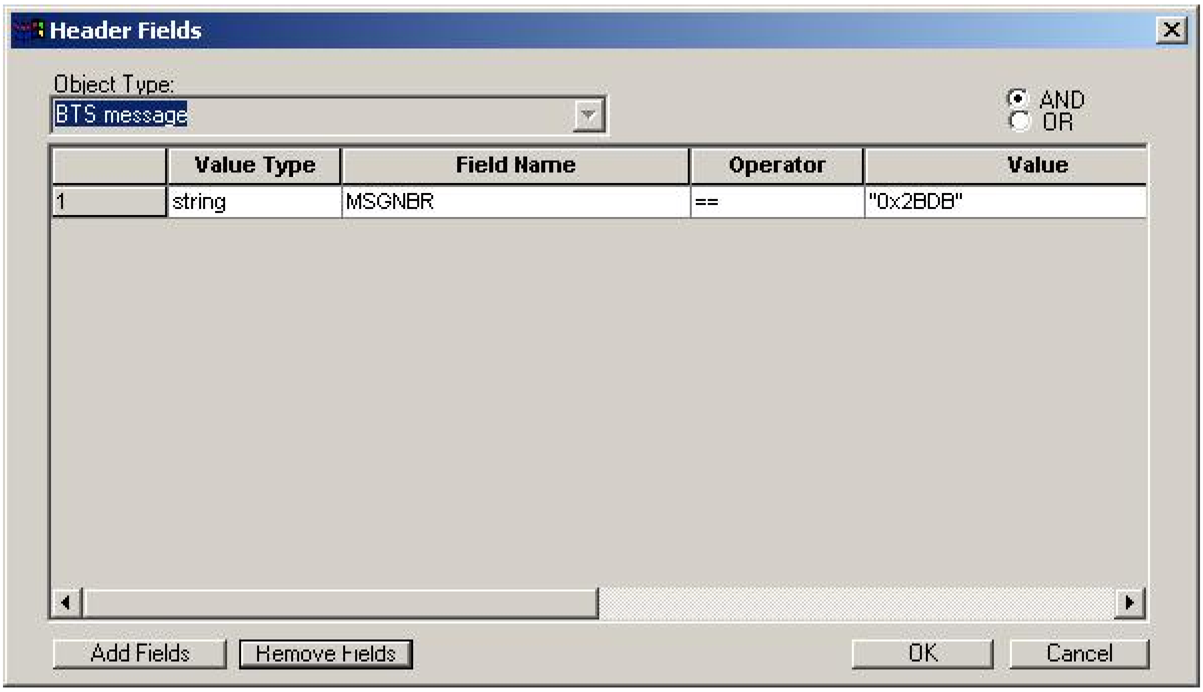


Object Type: BTS message

● AND
○ OR

	Value Type	Field Name	Operator	Value
1	string	MSGNBR	==	"0x2BDB"
2	string	MSGNAME	==	"ApiNmapGwMsgInd"
3	string	PROTOCOL	==	"nmap"
4	string	SENDER	==	"WAM_10/CTRL_MCU/Task:0xC"
5	string	RECEIVER	==	"Board:0x9D/CTRL_MCU/Task:0x46
6	string	PROCESSOR	==	"192.168.255.1"
7	string	PCTIME	==	"2007 5.8 09:09:50.69"
8	string	TARGETTIME	==	"00:04:02.309533"
9	hex	VERSION	==	0x00
10	hex	SFN	==	0x0D5E
11	hex	SEQNUM	==	0x0021
12	hex	PAYLOADLEN	==	0x0150
13	string	PAYLOADHEADER	==	"00002BDB9D1104621011000C016

Press remove fileds.



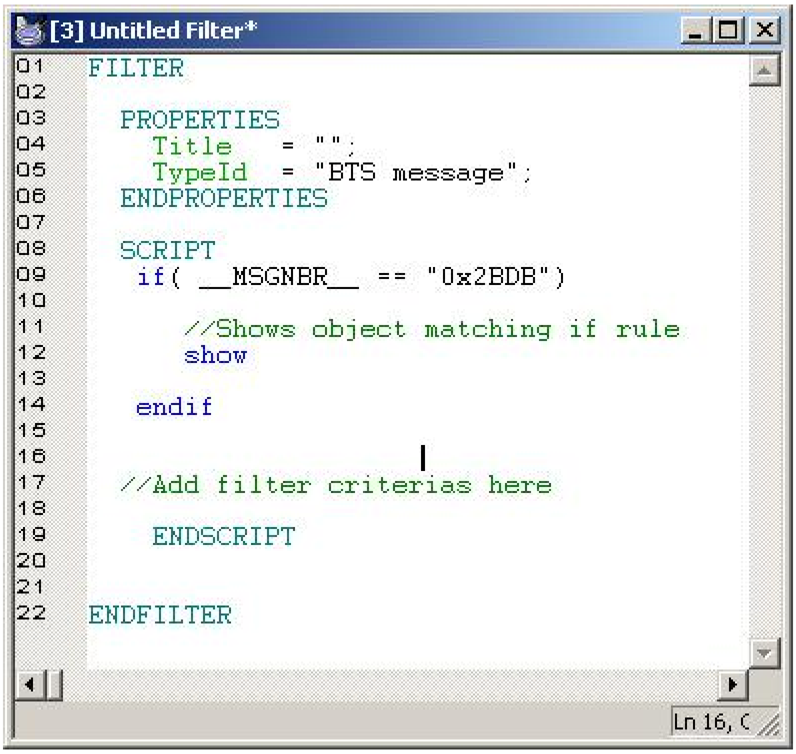
Object Type: BTS message

● AND
○ OR

	Value Type	Field Name	Operator	Value
1	string	MSGNBR	==	"0x2BDB"

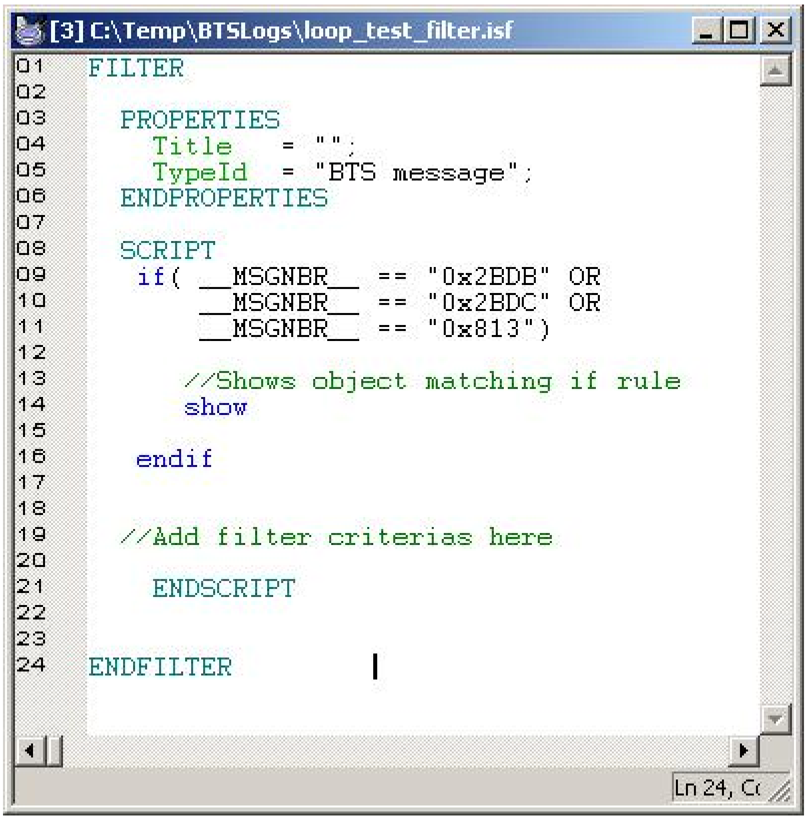
Press ok.

Here is a valid filter. Save it.



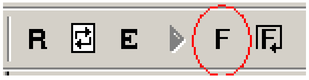
```
FILTER

  PROPERTIES
    Title    = "";
    TypeId   = "BTS message";
  ENDPROPERTIES

  SCRIPT
   if( __MSGNBR__ == "0x2BDB")

      //Shows object matching if rule
      show

   endif


  //Add filter criterias here

    ENDSCRIPT


ENDFILTER
```

But, let's add manually some more interesting messages.



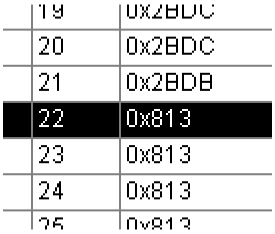
```
FILTER

  PROPERTIES
    Title    = "";
    TypeId   = "BTS message";
  ENDPROPERTIES

  SCRIPT
   if( __MSGNBR__ == "0x2BDB" OR
       __MSGNBR__ == "0x2BDC" OR
       __MSGNBR__ == "0x813")

      //Shows object matching if rule
      show

   endif


  //Add filter criterias here

    ENDSCRIPT


FILTER
```

Save the filter and run it by activating the log file window and pressing F in toolbar.



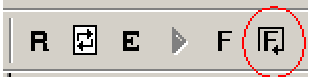
Browse your filter in 'Run Filter'-dialog.
Press 'Run filter'.

Now the log file window contents is filtered and only desired messages are presented.

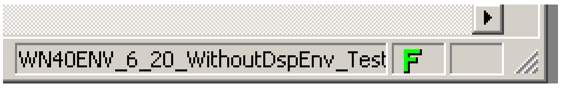


19	0x2BDC
20	0x2BDC
21	0x2BDB
22	0x813
23	0x813
24	0x813
25	0x813

TIP: if you wan't to modify your filter and run it again, just press 'Re-execute filter':

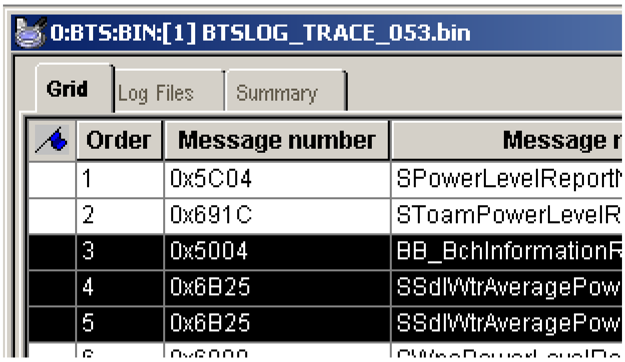


TIP: Green F in status bar indicates that the contents of the log window has been filtered.

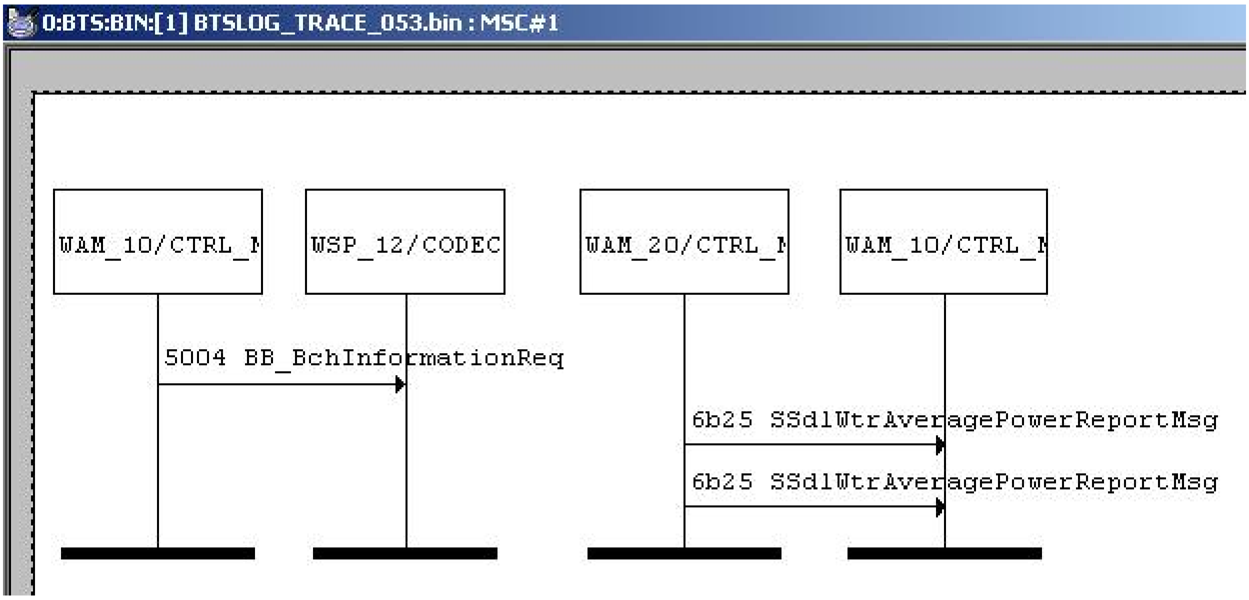


MSC - Message Sequence Chart

With Ida2, you can also create message sequence charts from the log file contents by painting messages from logfile window, and selecting 'create new MSC' from right click popup menu.

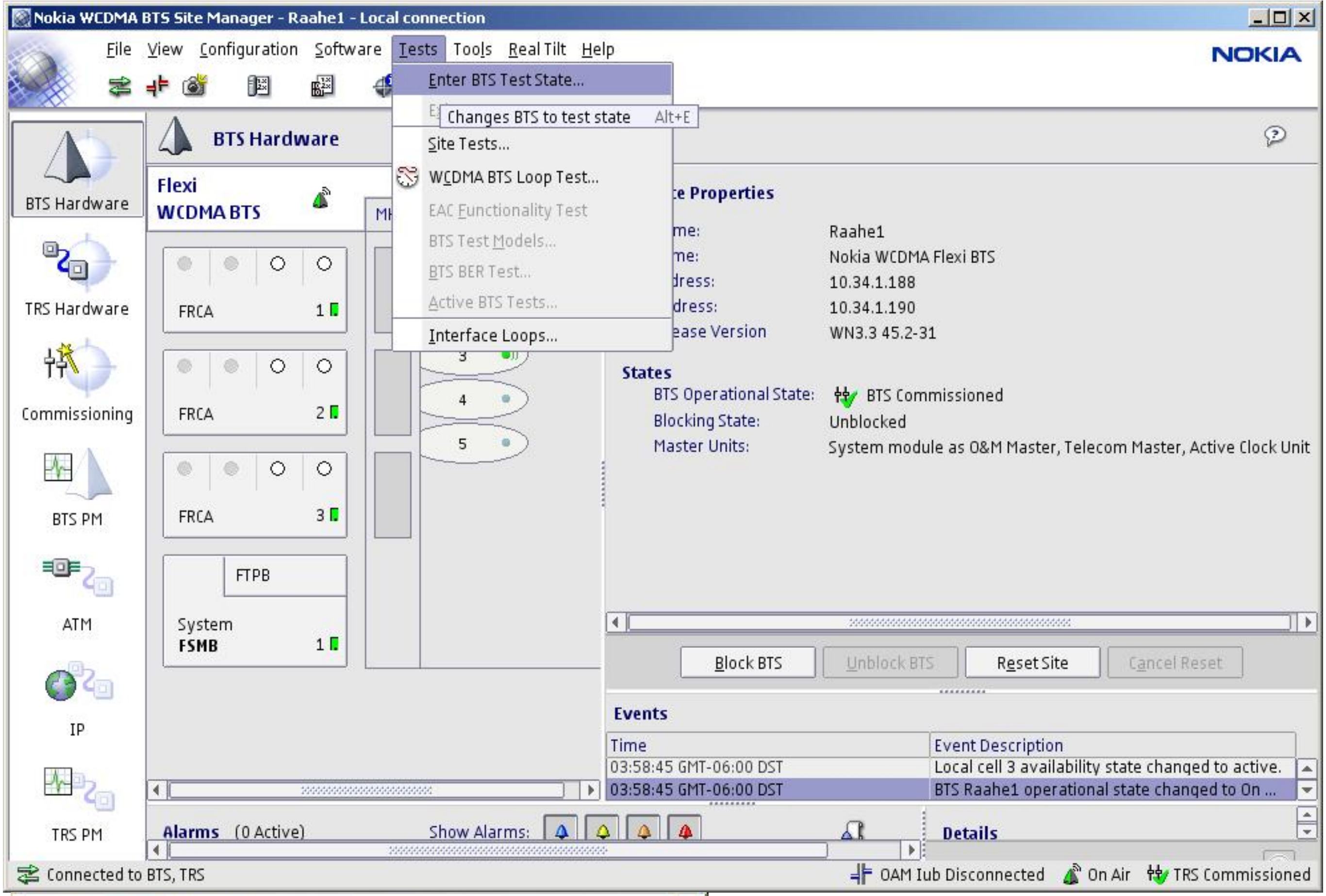

0:BTS:BIN:[1] BTSLOG_TRACE_053.bin

	Order	Message number	Message r
	1	0x5C04	SPowerLevelReportM
	2	0x691C	SToamPowerLevelR
	3	0x5004	BB_BchInformationR
	4	0x6B25	SSdlWtrAveragePow
	5	0x6B25	SSdlWtrAveragePow


0:BTS:BIN:[1] BTSLOG_TRACE_053.bin : MSC#1

WAM_10/CTRL_M
WSP_12/CODEC
WAM_20/CTRL_M
WAM_10/CTRL_M
5004 BB_BchInformationReq
6b25 SSdlWtrAveragePowerReportMsg
6b25 SSdlWtrAveragePowerReportMsg

You can swap to testing state with Element manager's command below.



e Properties

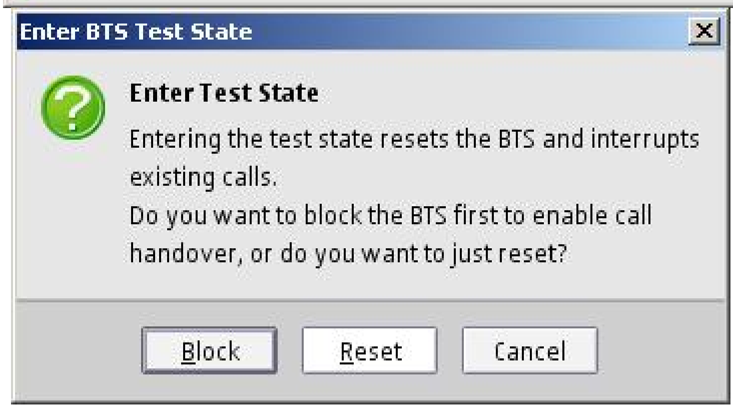
me:	Raahe1
me:	Nokia WCDMA Flexi BTS
dress:	10.34.1.188
dress:	10.34.1.190
ease Version	WN3.3 45.2-31

States

BTS Operational State:	BTS Commissioned
Blocking State:	Unblocked
Master Units:	System module as O&M Master, Telecom Master, Active Clock Unit

Events

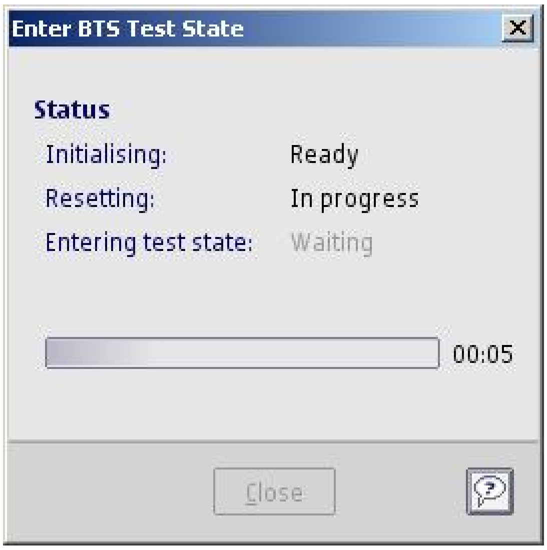
Time	Event Description
03:58:45 GMT-06:00 DST	Local cell 3 availability state changed to active.
03:58:45 GMT-06:00 DST	BTS Raahe1 operational state changed to On ...



Enter Test State

Entering the test state resets the BTS and interrupts existing calls.
Do you want to block the BTS first to enable call handover, or do you want to just reset?

15. When Flexi is resetted, BTS is in test dedicated mode.

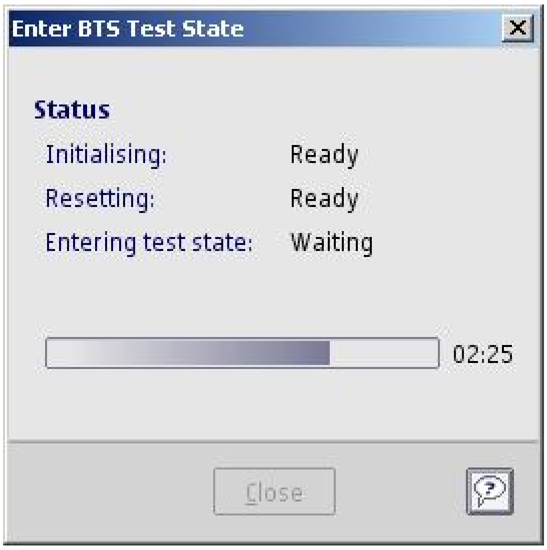


Enter BTS Test State

Status

Initialising: Ready
Resetting: In progress
Entering test state: Waiting

00:05



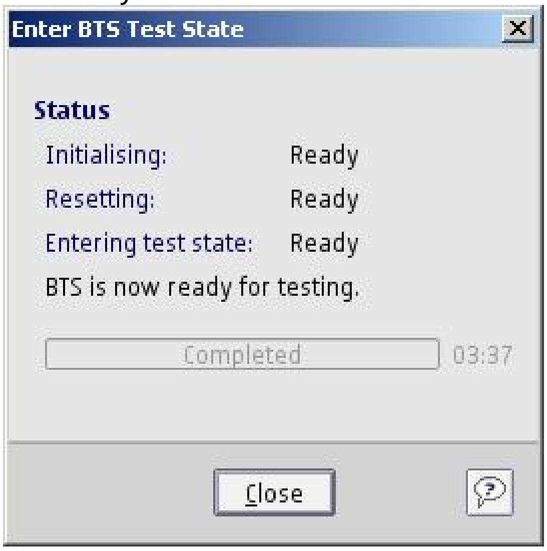
Enter BTS Test State

Status

Initialising: Ready
Resetting: Ready
Entering test state: Waiting

02:25

16. Now you can close this window...



Enter BTS Test State

Status

Initialising: Ready
Resetting: Ready
Entering test state: Ready
BTS is now ready for testing.

Completed 03:37

You can run loop in test dedicated mode or in OnAir mode.