MOMI®

When Your System Demands Quick Attention...®

MOMI is a System Management tool for the HPE NonStop Computer Systems (i.e. a Tandem) running the NonStop OS (i.e. Guardian Operating System or NSK).

MOMI software, written by BlackWood Systems, Inc., provides an easy way to obtain useful system information yet minimizing the impact to system resources.

MOMI is a Client / Server design (known long ago as Requester / Server). The Server portion runs on the NonStop System as a fault tolerant nonstop multi-tasking process-pair with a passive backup. The Client runs on Windows 10 or later and is written in Delphi by Borland (now known as Embarcadero). TCP/IP sockets with AES encryption are used to communicate between the client and server.

The document reflects MOMI Release Package 6.19 (PC Client 6.19 - NonStop Server 6.19).

Quick start Links -

Table of Contents

System requirements

MINIMUM SYSTEM REQUIREMENTS

PC Client

- Processor 4.0 GHz (or better)
- 8 GB RAM
- 10 GB free disk space
- Windows 10 or later client O/S
- Windows Server 2019 or later server O/S
- 1024 x 768 display with 24 bit color (minimum)
- 1280 x 1024 display with 32 bit color (recommended)
- Sound card and speakers (if Alarm sound desired)

NonStop Server

- NonStop X or Virtual NonStop
- Integrity NonStop Blade with J06.04 or later
- Integrity NonStop with H06.09 or later
- TCP/IP connection from PC to NonStop Server
- MEASURE installed and STARTED
- 600 MB disk space without History active or
- 22 GB initial default size (user selectable) with History active

Version 6 considerations

Overview

Version 6 of the MOMI software introduces a new logon dialog and default security setting more restrictive than in previous releases.

Generally speaking, the enhanced security mode does not affect existing installations of MOMI. New installations of MOMI, by default, operate in the enhanced security mode and includes users of mini-MOMI. An existing installation of MOMI, or an update of an existing installation, is when the CONFMOMI file on the NonStop System originates from a version 5 or earlier release.

The enhancements were made as the result of the ever changing world of security and user feedback which said 1) add support for multi-factor authentication, and 2) no data should be displayed prior to a user log on.

Support of multi-factor authentication, from MOMI's perspective, was to change our logon window to support a dialog or a series steps to logon. This means that each step of the logon process is provided to the NonStop Operating System and if accepted causes MOMI prompts for the next *piece* of logon information. A *piece* may be a password, pin or some other item of information as directed by the O/S security subsystem. The logon process initially prompts for the User ID. This change has no setting within MOMI and is entirely driven by the O/S security subsystem.

Support for not displaying data prior to logon means the MOMI PC Client displays a virtually blank screen prior to logon, the default for new installations. Existing installations, may see data prior to logon as was true in previous releases. Here is what this means:

- An existing installation of MOMI, after *upgrade* of the client and server **but** leaving the CONFMOMI file alone, may continue to see data such as the MOMI **Overview** screen prior to user logon.

- A new installation of MOMI or mini-MOMI, where a CONFMOMI file is installed, will not see data displayed in the client until user logon.

In both of these situations, the display of data prior to logon is initially controlled by the CONFMOMI keyword CLIENT-LOCKDOWN-MODE. Additionally, Client_Access continues to allow control over what screens are visible to individual users or groups.

Client compatibility

A version 5 client can be used with the version 6 server. The client must be 5.21 or later (older clients will crash shortly after start-up).

A version 6 client can be used with the version 5 server, however new screens and some features dependent on the server will not function or automatically downgrade.

In both of these *mixed version* situations, support of multi-factored authentication is not available and the older logon process is used.

Server compatibility - upgrade version 5 to 6

An existing installation, where MOMI 5 is installed and running, may upgrade the server to version 6 using their existing configuration and files. Database files created by version 5 are compatible with version 6. Clients must be 5.21 or later.

Server compatibility - fallback version 6 to 5

An existing installation, where MOMI 6 was installed over an existing version 5, may fallback to version 5 server software.

A new installation, where MOMI 6 is installed in a subvolume for the first time, **may NOT** fallback and use the version 5 server software. Database files created by Version 6 are not compatible with version 5 and the MOMI server will fail during start up should this situation occur.

Overall best performance

For overall best performance, a new installation of the MOMI server software is required.

Licensing Information

MOMI PC Client software may be freely installed without consideration of any per-user, per-client, or number of PCs restrictions.

MOMI NonStop Server software may be freely installed without consideration of the number processors in the system, number of cores (active or available) in the processor, nor usage of the system (i.e. production, development, QA, etc...).

Features of MOMI are controlled by a password (i.e. license key) placed in the CONFMOMI file located on the NonStop System normally keyed by System serial number.

- When operated with a password, additional or full features are available as determined by the password.
- When operated without a password, the software operates in a limited or mini-MOMI mode. The limited mode is available for (currently) 2-years after the release date of the MOMI Server. Upgrade to the latest version to continue to use mini-MOMI after the period expires. As a standard disclaimer, we reserve the right to alter or discontinue mini-MOMI features.
- The current state of the MOMI password for the system displayed may be seen on the MOMI PC Client screen Configure / Diagnostics.

A trial password for MOMI activates additional or full features but contains an expiration date. On the expiration date MOMI will stop or limit operation. Remove the password and restart the MOMI server to return to mini-MOMI mode (if within the period mentioned above).

Multiple MOMI Servers may be installed on the NonStop System. Place the software in a different subvolume, change the main MOMI process $name, listen on a different port, and don't share files across the instances on the NonStop System. See Duplicate an existing server environment for additional details. This configuration is sometimes used to test new versions of MOMI software.

Installation

Overview

MOMI software is distributed in a MSI file created by InstallShield. This file is the place to start for a MOMI install or upgrade. It provides capabilities to install to the PC (all users or current user), copy MOMI software to the NonStop System, and/or copy MOMI software to a shared network location. All supported NonStop platforms (NonStop X (physical and virtual), Nonstop i, NonStop Blade, and S-Series Systems) are included.

There are several ways MOMI software may be installed:

- install to the local PC for all users (requires PC Administrator access)
- install to the local PC for the current user
- copy MOMI software to a shared network location where users may access the software via short-cuts
- copy the Server portion of MOMI software to the NonStop System via FTP/SFTP using MomiFTP or manually

Particularity for a first install, there are additional considerations:

- On the NonStop System, the default process name for the main MOMI process is $MOMI (all other processes started by the main MOMI process are randomly selected by the NonStop system). This default may be changed in the OBYMOMI file.
- On the NonStop System, the main MOMI process listens for incoming connections from MOMI PC Client(s). In the CONFMOMI file the keyword TCPIP-LISTEN defines the $TCP/IP process name, IP address of the NonStop System, and a port (default 2010 but user selectable) to receive the incoming connections. The CONFMOMI file contains a sample entry to modify as needed.
- In most corporate network environments, in-between users PC and the NonStop System is a firewall. This situation requires the network administrator to authorize or allow communication and typically the information asked:
    - IP address of the NonStop Server (what the client connects to)
    - type of communication : TCP
    - port used : 2010 (this is the default and is defined in the CONFMOMI file)
    - direction communication is established : from the PC to the NonStop System
    - is the PC in the corporate office and/or connecting via a VPN

- Corporate PC administration *may* also need to authorize the MOMI PC based applications. All MOMI PC based executables are digitally signed.

- See here for additional information.

First installation / Update of an existing installation

The procedures for installing MOMI the first time or updating an existing installation are virtually the same.

In prior releases, MOMI installation software was packaged as an EXE or ZIP. The new MSI format performs all of the same features, plus allows the client to be installed under the current user which does not normally require administrator access.

The MOMI PC Client installs by default into its own version specific subdirectory and is accessed from the Windows Start menu under **BWS MOMI n.nn** (where n.nn is the version number).

It is recommended that a new version of the MOMI server software or an update to an existing version is installed into the same subvolume on the NonStop System. By default, existing startup and configuration information is not changed. Existing files are renamed as necessary to allow for installation while the MOMI server is running and to provide for fallback.

The installation process starts by obtaining MOMI software from either our website or perhaps a flash drive handed out a conference (we can also send you the software on a flash drive on request). MOMI installation software is a standard Windows MSI file and can perform either a new install or update an existing installation. The installation program also contains utilities for transferring MOMI server software to the NonStop System (MomiFTP) and for placing MOMI software on a shared drive location (CentralServer).

The following steps describe installing MOMI software on the local PC and on the NonStop System:

1) Obtain MOMI software and double click on the **MSI** file. The initial screen provides allows display of the ReadMe and detailed installer Help.

2) Click though the **Welcome** screen. Please read and accept the **License Agreement**. Click through to continue.

3) The **Utilities** screen allows the option to run **MOMIFTP** and **CentralServer** (can be run now or later). Click **Install Client >** to continue.

4) The next screen selects the **Setup Type**. The default is **Complete**. Click through to continue.

5) The **Ready to Install the Program** screen allows MOMI software on this PC to be installed **Only for me (<user>)** or **Anyone who uses this computer (all users)**. Choose the desired option. Note the **Anyone...** selection prompts for an administrator logon if needed. After the installation completes, click **Finish**.

6) Press the Windows Start button and under BWS MOMI n.nn (where n.nn is the version just installed) select **MomiFTP**.

a. On the screen **1st Connect to HPE NonStop**, select the **Transport type** (FTP or SFTP), fill in the **Host Address** (NonStop System IP address or DNS), **User ID**, **Password** and **Initial $Vol.SubVol** where the MOMI files are placed. The User ID specified owns the files after the transfer takes place. Press **Connect to HPE NonStop**. If the connection was successful, you will see a big green **Connected..**.

b. Press **Next >>** to advance the screen.

c. On the screen **2nd Select Vol.SubVol on HPE NonStop** a list of the files present (if any) in the selected location is displayed. For a new installation, you should choose an empty subvolume. Change the location as needed **Change** to update the display.

d. Press **Next >>** to advance the screen.

e. On the screen **3rd Select New Install or Update** press **Put Files on HPE NonStop** to start the transfer. The status boxes indicate the outcome of the transfer.

f. Press the **Exit** button to disconnect and stop the program.

7) From a TACL prompt, logon with the User ID and password that $MOMI will operate under. Volume to the location where the MOMI files were placed.

8) Edit the file **OBYMOMI**. This is a TACL obey file used to start MOMI and provide initial configuration information. Page down to the section "**vvv set values below here vvv**". This marked portion of the obey file is used to change the initial start up values for the $MOMI server portion.

Check the process name, CPU $MOMI operates in, priority and home terminal. Default values are usually valid for most systems.

9) Edit the file CONFMOMI. This edit file is used to set various MOMI attributes using keywords. Set the System-Description and check TCPIP-LISTEN which defines the TCP/IP process name, optionally the address and port that MOMI will *listen* on for PC Client connections.

- mini-MOMI users, the above should be about the only item of concern.
- MOMI users, add your MOMI password. Also, define any history files desired. MOMI automatically creates the history files.

10) SQL/MP compile MOMI (if you don't use SQL, ignore this step):
from a TACL prompt: OBEY OBYCSQL

11) Create / update BWSSG (the Super Group helper process) (see Security / General considerations to determine if you need this step):

(BWSSG is a PROGID'ed copy of the MOMI server. The following steps show you how to rename out any existing BWSSG, keeping a backup copy, create a new file - details such as required logon security to perform these operations is not covered).

Purge BWSSG2
Rename BWSSG1, BWSSG2
Rename BWSSG, BWSSG1
FUP DUP BWMOMI, BWSSG, sourcedate (BWMOMIi on TNS/E - BWMOMIx on TNS/X)
FUP Give BWSSG, <Super.Group>
FUP secure BWSSG, NGNG, PROGID

If used, BWSSG ***must*** be updated any time BWMOMI is updated.
The FUP DUP option of SOURCEDATE or SAVEALL **must** be used.

12) Start MOMI:

From a TACL prompt: OBEY OBYMOMI

(wait at least 1 minute before proceeding to the next step)

13) On your PC, press the Windows Start button and under **BWS MOMI n.nn** (where n.nn is the version just installed) select MOMI. The first time a Client is started on a PC, it will prompt for the TCP/IP address and port of the NonStop System (the port is defined in the CONFMOMI file and has a default of 2010).

Shortly after startup, MOMI on the NonStop System defines a measurement (DCPUS file) configured with a variety of entities (CPU, Process, TMF, etc...). This measurement does not specify an interval but is effectively used for LISTACTIVE purposes. This file may become 100% full which is not a concern.

On older NonStop Systems, sometime after starting MOMI, the following EMS message, one per processor, may be logged:

07-08-19 09:47:46.200 \SYSTEM.$XMnn TANDEM.MEASURE.G09 4016 Measurement 0 $vol.subvol.DCPUS: write to datafile failed. File error 45.

These messages are not a problem and may be ignored. No action is required.

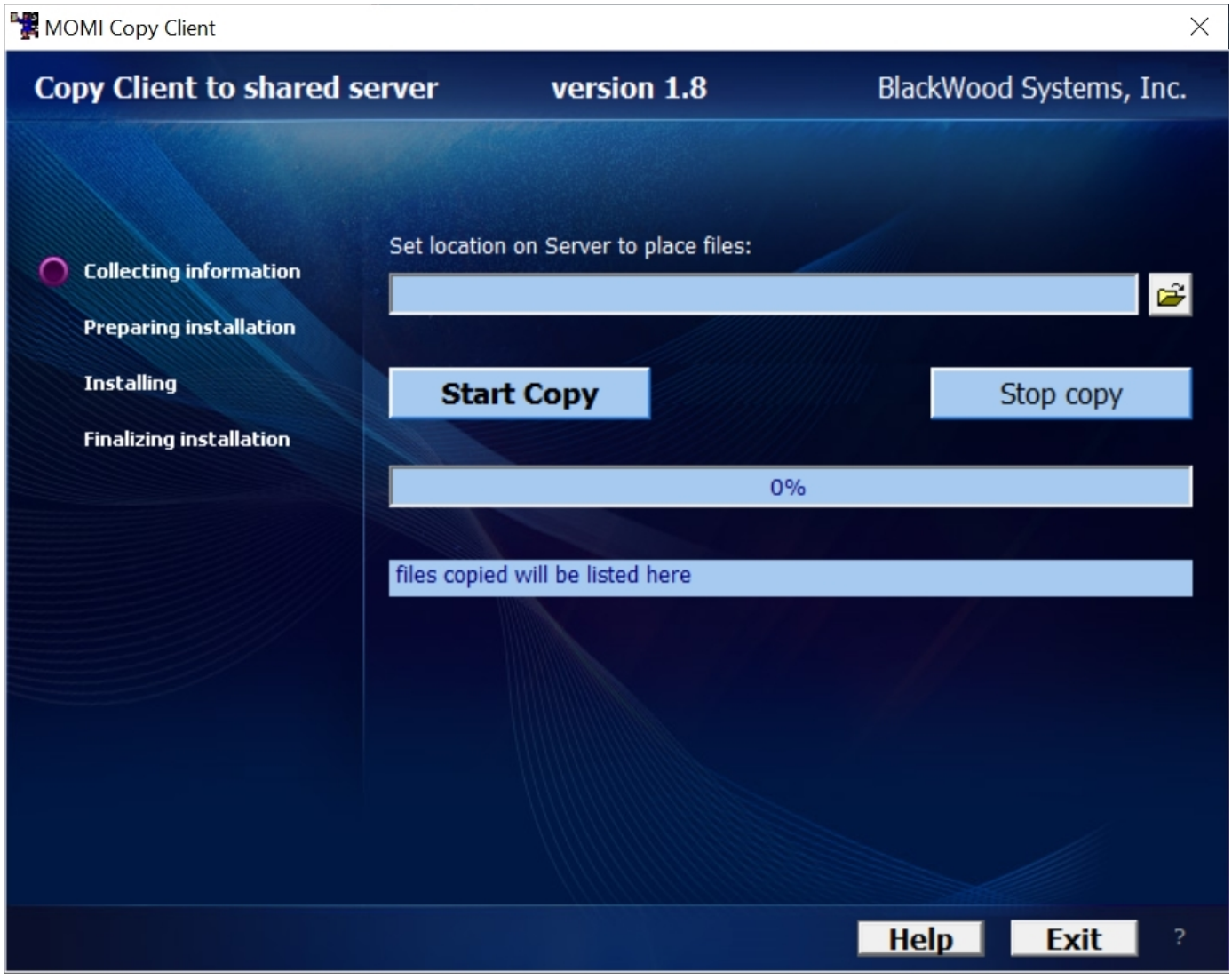
PC Client

PC Client on a central server

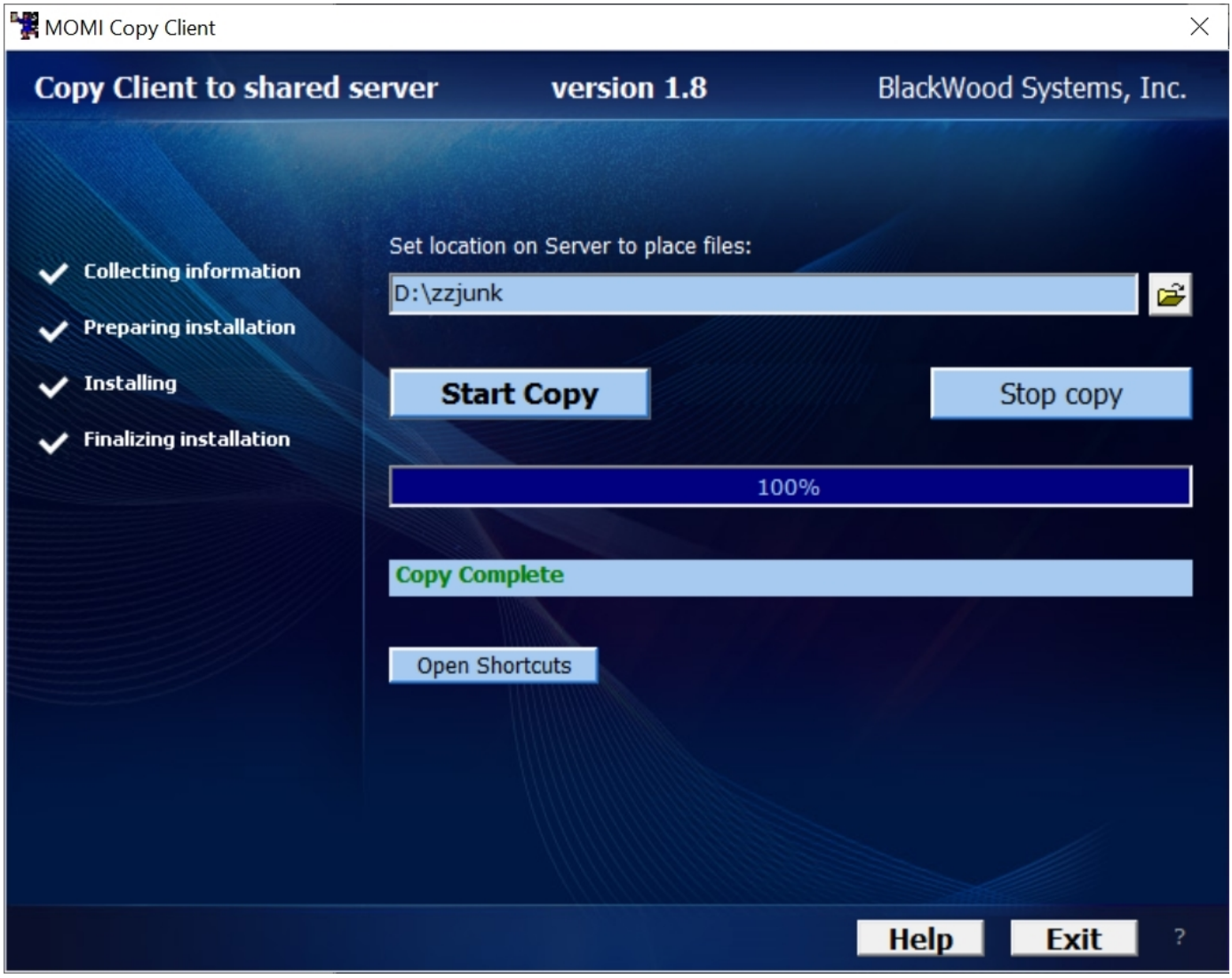
Some shops may wish to place a single copy of the PC MOMI Client software on a shared server to simplify distribution and upgrades.

The PC MOMI Client executable (MOMI.EXE) has no special installation requirements. There are no registry, DLLs or OCXs directly associated with the MOMI Client itself (other than built-in Windows components).

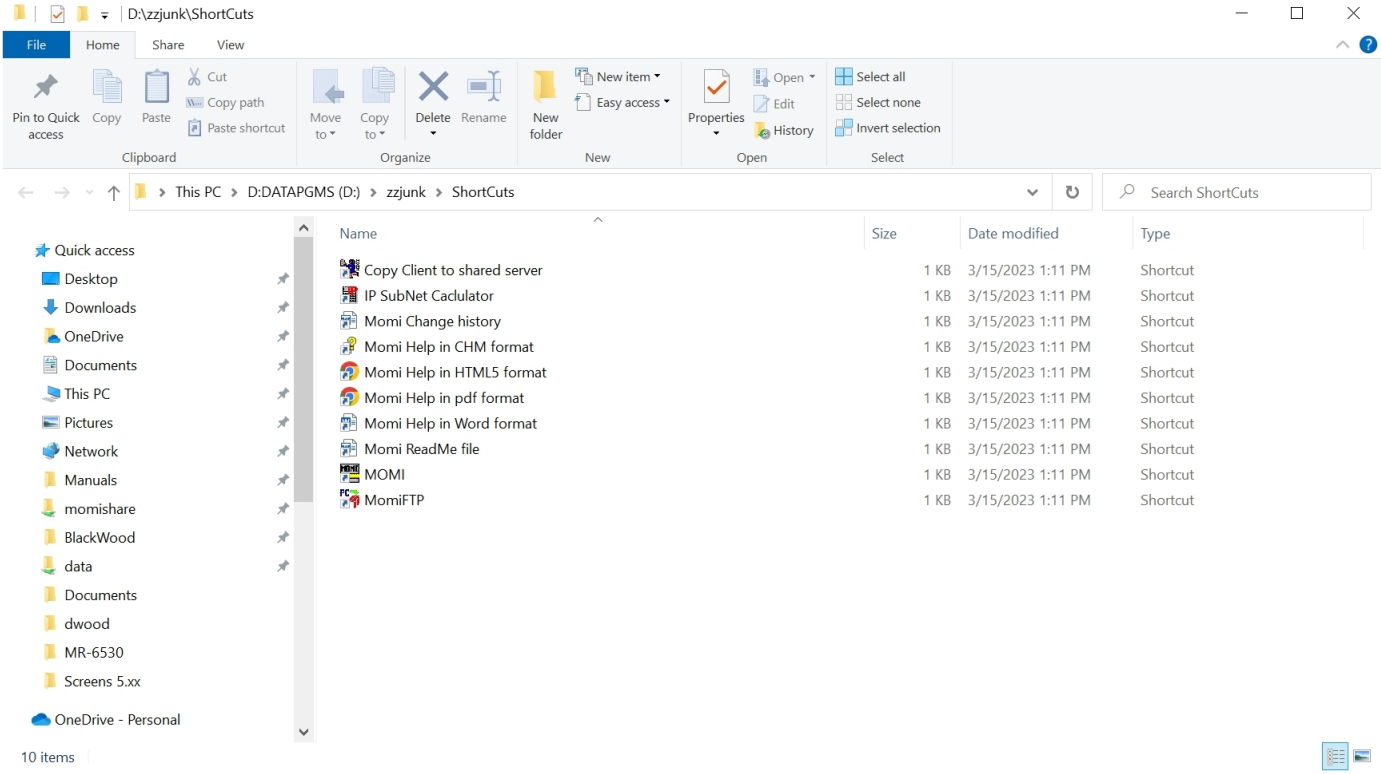
During or after MOMI installation, a utility called CentralServer is available to copy MOMI software to user specified location.



Enter the location where to place files and push **Start Copy**.



After the operation completes, the button **Open Shortcuts** is available to display windows short-cuts to the copied software. Share the short-cuts with users needing to access MOMI PC Client. .


D:\zzjunk\ShortCuts
File
Home
Share
View
Pin to Quick access
Copy
Paste
Cut
Copy path
Paste shortcut
Clipboard
Move to
Copy to
Delete
Rename
Organize
New folder
New item
Easy access
New
Properties
Open
Edit
History
Open
Select all
Select none
Invert selection
Select
This PC
D:DATAPGMS (D:)
zzjunk
ShortCuts
Search ShortCuts
Quick access
Desktop
Downloads
OneDrive
Documents
This PC
Pictures
Network
Manuals
momishare
BlackWood
data
Documents
dwood
MR-6530
Screens 5.xx
OneDrive - Personal
Name
Size
Date modified
Type
Copy Client to shared server 1 KB 3/15/2023 1:11 PM Shortcut
IP SubNet Caculator 1 KB 3/15/2023 1:11 PM Shortcut
Momi Change history 1 KB 3/15/2023 1:11 PM Shortcut
Momi Help in CHM format 1 KB 3/15/2023 1:11 PM Shortcut
Momi Help in HTML5 format 1 KB 3/15/2023 1:11 PM Shortcut
Momi Help in pdf format 1 KB 3/15/2023 1:11 PM Shortcut
Momi Help in Word format 1 KB 3/15/2023 1:11 PM Shortcut
Momi ReadMe file 1 KB 3/15/2023 1:11 PM Shortcut
MOMI 1 KB 3/15/2023 1:11 PM Shortcut
MomiFTP 1 KB 3/15/2023 1:11 PM Shortcut
10 items

PC Client initial configuration deployment

(client version 5.23 or later)

The MOMI PC Client has the ability to be deployed with an initial default configuration. This default configuration allows new users to begin using the MOMI client without first defining the connection to each NonStop Systems. The default configuration capability is best suited where the client is accessed from a central server.

The feature is activated by placing a copy of the **MOMI.INI** named as **MOMI_INIT.INI** in the same subdirectory were the MOMI.EXE executable is located.

When the MOMI PC Client is launched and a MOMI.INI file is not found, it checks to see if a MOMI_INIT.INI file is present in the same subdirectory as the MOMI.EXE executable. If so, it copies the file and uses it as the initial configuration. If a MOMI_INIT.INI file is not present, a MOMI.INI file is created with minimal default values.

Note that once a MOMI PC Client is started that Windows User ID will have a MOMI.INI present in the following location:

Windows 10 and later -

C:\Users\<User ID>\AppData\Roaming\BlackWood Systems\MOMI\

Where <User ID> is the name used at logon to Windows. You must enable *Show hidden files, folders and drives* under Windows Folder Options to enable their display.

To use this feature, a MOMI administrator would:

1) start the MOMI PC Client.
2) set all screens and options to the desired default settings.
3) stop the client (important-to insure all settings are saved).
4) restart the client and export the MOMI.INI file named as **MOMI_INIT.INI**.
5) place the exported file in the same subdirectory as the MOMI.EXE executable in the shared location.

The location of the MOMI.INI file may be redirected via the run-time param INIPATH.

Server

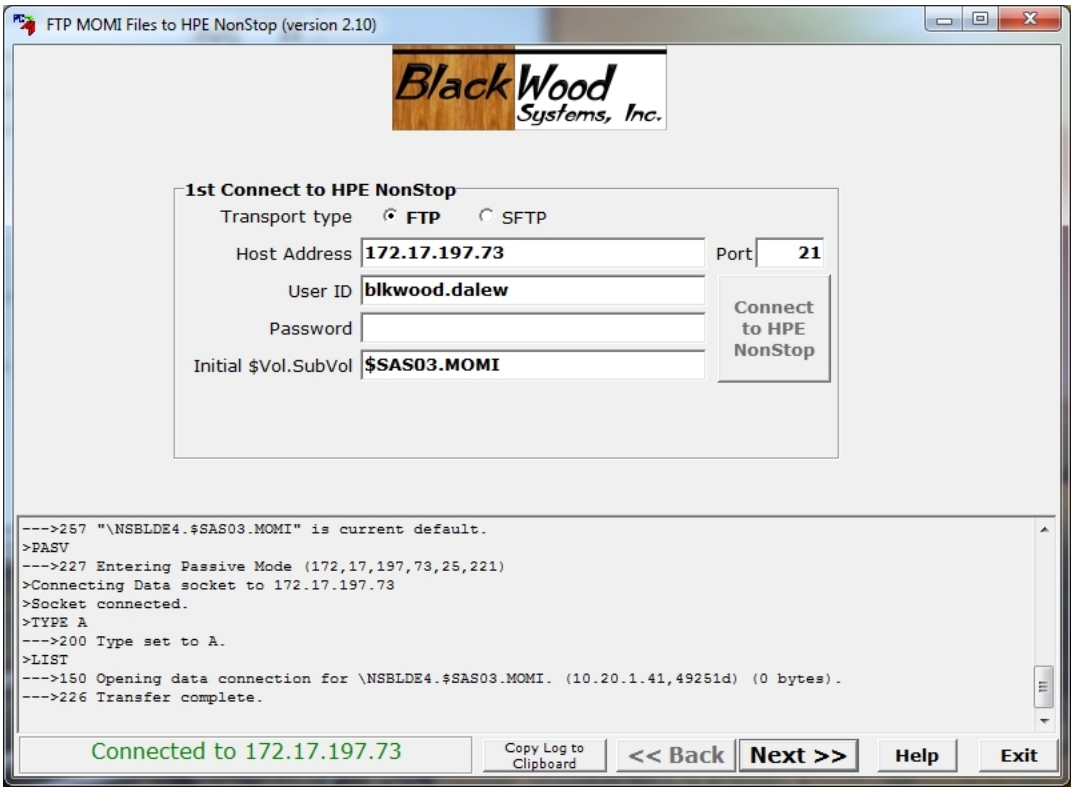
Installation of NonStop Server based files (using MomiFTP)

The MomiFTP utility is provided to aid in the transfer of MOMI files from the PC to the NonStop System using the FTP/SFTP protocol. Transfer type (ASCII or Binary) and File codes settings are automatically perform by the utility.

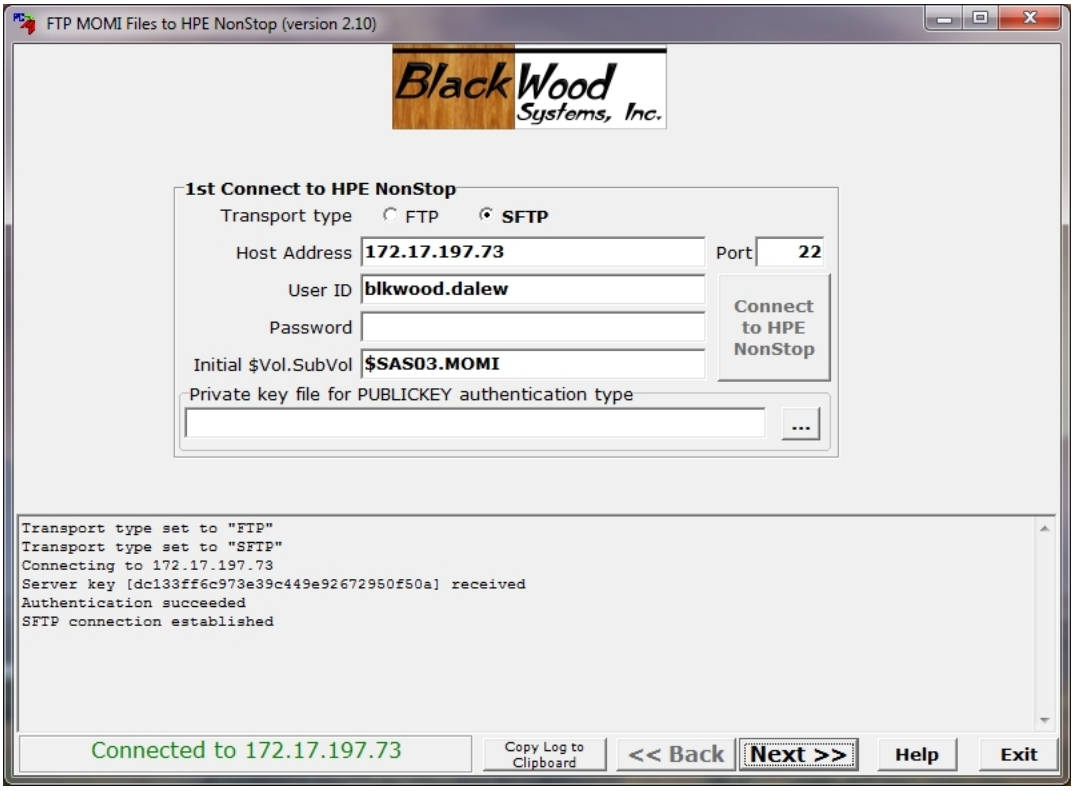
The utility is located on the PC in the "\Tandem Install" subdirectory where the PC Client was installed if the option of **Complete** or **Custom** was selected. This subdirectory contains configuration files necessary to drive MomiFTP and a subdirectory of the NonStop System based files.

MomiFTP is comprised of three screens. The first screen determines the system to receive file and User ID to transfer the files, the second screen determines the destination subvolume on the receiving system, and the third screen performs the transfer.

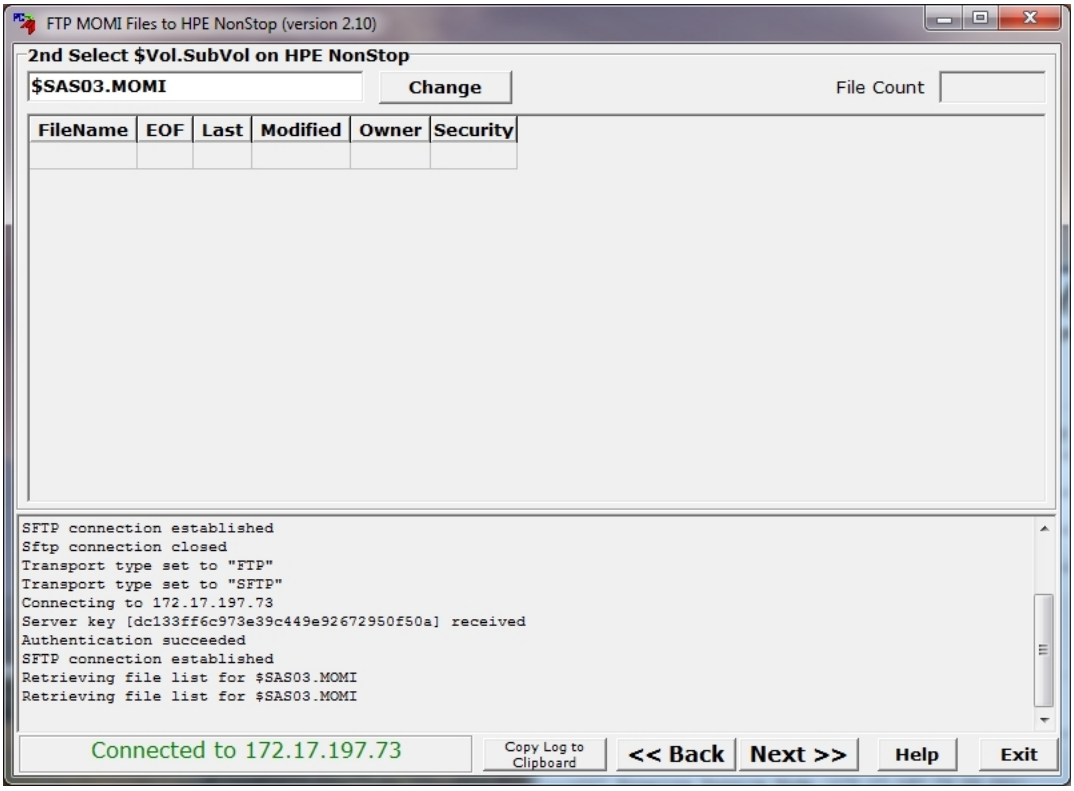
On the first screen enter the **Host Address** or DNS name of the NonStop System to receive the MOMI files. Enter a **User ID** and **Password** along with (optionally) the location (**Initial $Vol.SubVol**). Press **Connect to HPE NonStop** to establish the FTP connection. Press **Next >>** to advance to the next screen once a connection is established.



If a secure connection is available, SFTP may be selected and then enter either a **Password** or select an optional **Private key file**.



The second screen displays the current contents of the location previously selected (or the default location) and also provides the opportunity to **Change** the location. Press **Next >>** to advance to the next screen.



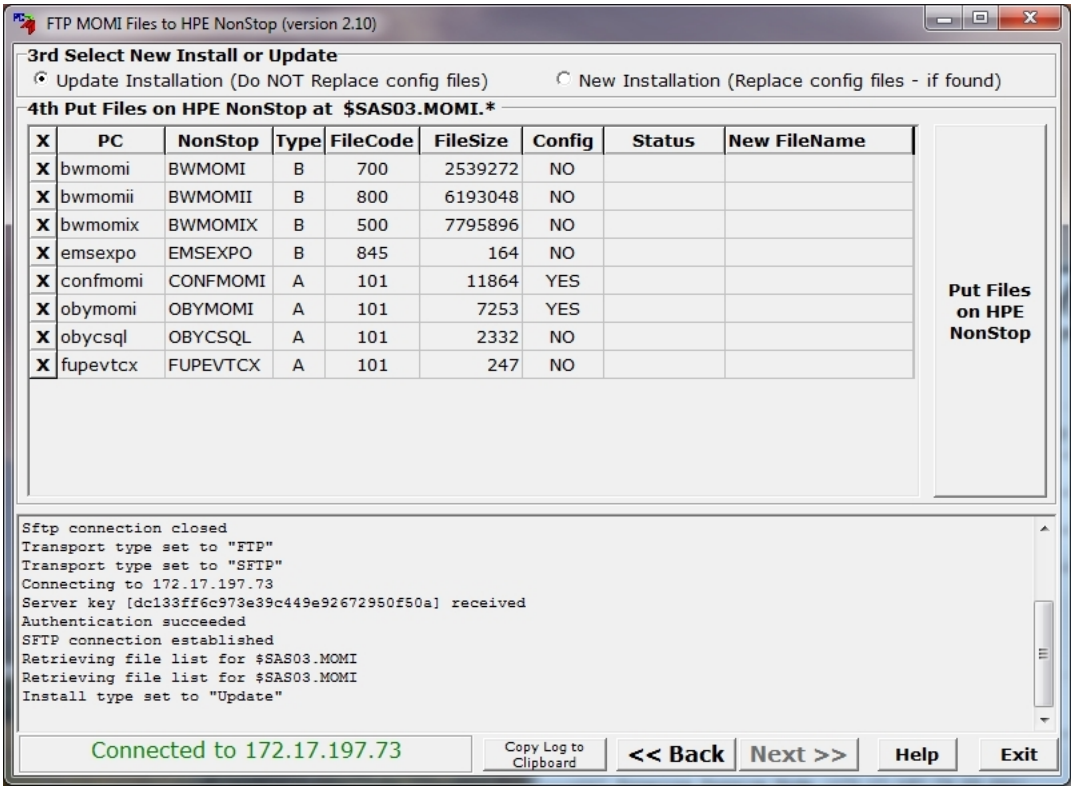
The third screen displays the files to be transferred and the logic to use with existing configuration files.

The **Update Installation...** option (the default) leaves existing configuration files alone. Existing object files are renamed, with a 1 or 2 placed on the end of the name and is suitable for transferring files while the MOMI server is running. For all files with a YES in the **Config** column, when an existing file exists, the new file has a 0 (zero) placed at the end of the name (indicating it is a new file). Restarting MOMI with the OBYMOMI file will then cause the new object files to be used. *This selection also works correctly for a new installation (i.e. the destination is empty).*

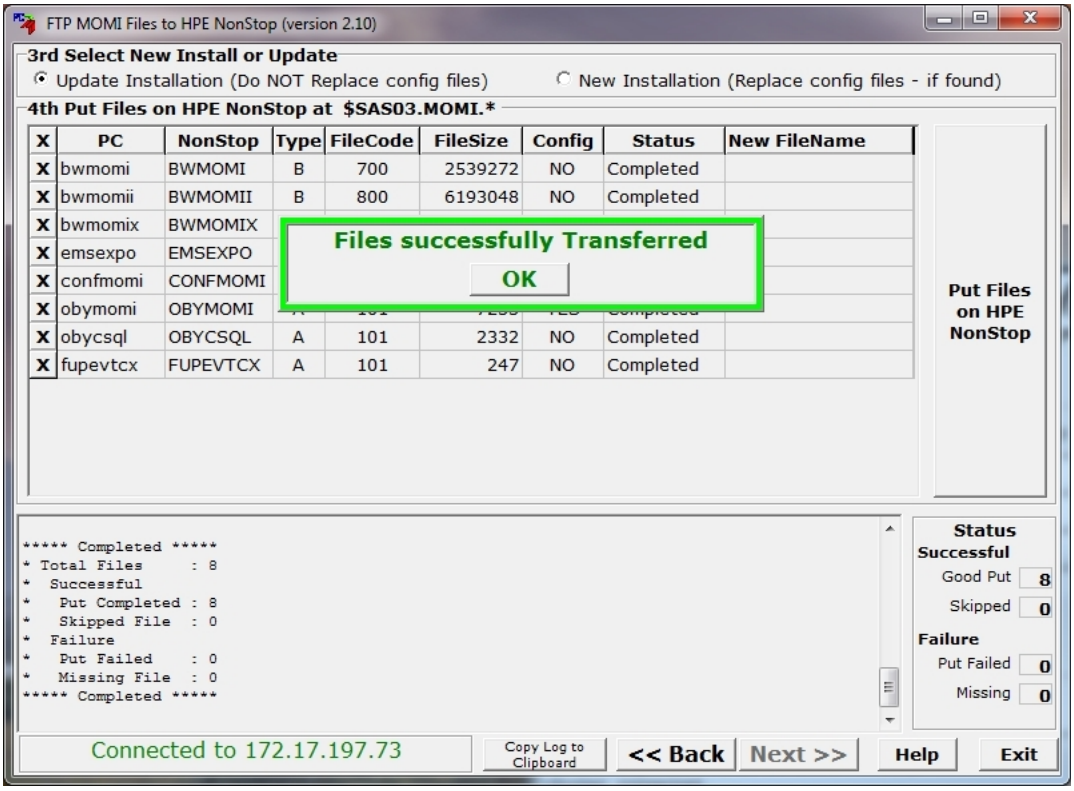
The **New Installation...** option will rename any existing files (add a 1 to the end) so that any existing file is updated. This selection is only performed with a new installation or if directed by the ReadMe (which is seldom). **The OBYMOMI file and CONFMOMI file need to be edited and settings confirmed for your environment.**

The X in the first column indicates which files are transferred (left & right mouse click toggles the setting).

Press **Put Files on HPE NonStop** to start the transfer process. During the transfer, a flashing indicator (...) is present in the **Status** column to signal the transfer progress .



After the transfer, the **Status** column indicates the completion. Status counts are located in the lower left hand corner of the screen. The **New FileName** column indicates the name of any configuration file added.



Press **Exit** to leave the program.

If any problems occur, scroll through the log window to help determine the reason (usually security related). The **Copy Log to Clipboard** should be used to provide the log to support.

Installation of NonStop Server based files (manual method)

If the MOMIFTP utility cannot be used to transfer files from the PC to the NonStop server, these instructions describe how an alternate file transfer program would be used. The MOMIFTP utility does not perform any *special* actions, but simply renames files (if existing are present) and transfers MOMI files from the PC to the NonStop System using FTP or SFTP.

The first step is to locate the source files -

PC install all users:

C:\Program Files (x86)\BlackWood Systems\BWS MOMI n.nn\Tandem Install\NonStop\

PC install user only:

C:\Users\<User ID>\AppData\Local\Programs\BlackWood Systems\BWS MOMI n.nn\Tandem Install\NonStop\

Via shared server (varies based on how shared server is organized):

\Tandem Install\NonStop\

(where n.nn is the release package number)

If MOMI is not installed, download and execute the MSI file. Select **Launch CentralServer** and place the files in an empty subdirectory. In the destination subdirectory under **Tandem Install\NonStop** are the subdirectories **\ascii** and **\Binary**. The grouping indicates the needed file transfer mode (i.e. ASCII or Binary).

The second step, using a file transfer utility, copy files from the PC to the NonStop (all to the same destination subvolume) -

```
Binary

      BWMOMI
      BWMOMIi
     BWMOMIx
      EMSEXPO


ASCII
```

```
FUPEVTCX
OBYMOMI
CONFMOMI
OBYCSQL
OBYALTER
```

The third step is to logon to the NonStop System, volume to the location where the files were placed and OBEY the TACL obey file OBYALTER to alter the file codes (or manually perform the steps below):

```
a) FUP alter BWMOMI  ,code 700
b) FUP alter BWMOMIi ,code 800
c) FUP alter BWMOMIx ,code 500
d) FUP alter EMSEXPO ,code 845
```

At this point the files are transferred. Continue the instructions in the section First Installation Update an existing installation after the step using MOMIFTP.

Server Files

The following files make up the MOMI Server environment when first started -

- BWMOMI | BWMOMIi | BWMOMIx

  MOMI Server executable

  ```
       BWMOMIx - 32-bit TNS/X native (NonStop X & Virtual
  NonStop)
       BWMOMIi - 32-bit TNS/E native (NonStop i & NonStop
  Blade)
       BWMOMI  - 32-bit TNS/R native (S-Series)
  ```

- CONFMOMI

  Edit file containing password (i.e. or license key), file locations and general configuration parameters for the MOMI subsystem. The file is edited by the MOMI administrator. See Overview for additional information.

- EMSEXPO

  A compiled EMS filter used by MOMI to monitor certain Expand events.

- FUPEVTCX

  A FUP obey file used to optionally create the file EVENTCX. This file is used by Viewpt, but MOMI can display, to store EMS Event Cause/Effect/Recovery action.

- OBYCSQL

  A TACL OBEY file used to SQL/MP compile the MOMI Server executable. This file is used once after a new server in installed and before the MOMI subsystem is started.

- OBYMOMI

  A TACL OBEY file used to start the MOMI subsystem on the NonStop System and automatically selects the correct executable. The file is edited by the MOMI administrator if needed to set the CPU and priority for the main $MOMI process.

The following file are automatically created after server startup -

- CNF01DB

  The MOMI configuration database. See Where MOMI stores data for addition information.

- DCPUS

  MEASURE Master Measurement file (file code 175). See DEFAULT-WORK-LOCATION for additional information.

- D<various-alpha-numeric>

  Dynamic MEASURE measurement file(s) (file code 175). This files are created by MOMI as various measurements are needed, usually the result of a client request. See DEFAULT-WORK-LOCATION for additional information.

- HSTnnDB

  Optional History file(s). See HSTnnDB for additional information.

- LOG01DB

  Log file. See LOG01DB for additional information.

Duplicate an existing server environment

The following describes how to duplicate an existing MOMI environment. If the duplicate is on a different NonStop System, a password is required and file placement should be checked but otherwise the files may be used as is. If the duplicate is on the same NonStop System, some changes are necessary as data files cannot be shared.

Take note of the explanations following as avoiding some files may speed up the creation of the duplicate environment.

1. stop the existing MOMI process.

2. using FUP duplicate all of the files in the existing MOMI subvolume to another subvolume.

3. restart the existing MOMI process.

4. In the duplicate OBYMOMI file:

   - on the same system:

     assign a new process name to MOMI (each MOMI process must have unique process name).

   - on a different system:

     change the PASSWORD to one assigned for that system.

5. In the duplicate CONFMOMI file:

   - on the same system:

     for the keyword TCPIP-LISTEN, assign a new port number (each MOMI server must listen on a unique port).

     for the keyword HSTnnDB, assign new vol/subvol locations (each MOMI server must have unique history files).

     check other data file names to insure make sure they are not duplicated (the keywords above are the most common but a few others exist).

   - on a different system:

     check volume names to insure they are valid on the new system.

The following are brief explanations of certain files within the MOMI subvolume. MOMI automatically creates new one(s) at start up if they are not present:

- The CNF01DB file is the MOMI configuration database. It stores configuration information entered online such as Alarm definitions. It does not generally store information found in the OBYMOMI or CONFMOMI files. To start a duplicate environment 'clean', simply delete this file.
- The LOG01DB file is the MOMI log file. It stores messages logged by MOMI. If you want to start the duplicate environment with an empty log file, simply delete this file.
- The HSTnnDB files are the MOMI history files. If you want to start the duplicate environment with empty history files, simply delete them. These files are usually large and may take some time to duplicate. It is usually faster to duplicate them from one disk volume to another disk volume (to avoid drive thrashing).

Start / Stop MOMI Server

TO START MOMI ON THE NONSTOP SYSTEM

The TACL obey file OBYMOMI provides the means to start MOMI.

1) Insure the MEASURE subsystem is running on your NonStop System.

   *Failure to start MEASURE will cause a **MEASURE NOT RUNNING** error in the MOMI PC Client.*

2) Start MOMI from a TACL prompt:
   volume MOMI (subvolume where files were placed)
   obey OBYMOMI

3) Once MOMI is started, you are ready to run the PC based MOMI Client.

TO STOP MOMI ON THE NONSTOP SYSTEM

MOMI only requires a Guardian stop of one process from a TACL prompt for an orderly shutdown.

1) From a TACL prompt, logon with the User ID MOMI was started under.

2) STOP $MOMI (or use the process name specified in the OBYMOMI file)

3) Other MOMI processes will perform cleanup and stop themselves automatically after 10 to 15 seconds.

System start up and shutdown files

The System administrator should add the necessary references to MOMI in the system start up and shutdown procedures.

Generally, the System administrator will use a TACL running under the Super.Super ID to reference a stream of other TACL files that perform the start up of various software components. The original TACL obey file will use a TACL / in ... / of an intermediary obey file. The intermediary obey file created by the System administrator performs a log-down (from Super.Super) to the selected MOMI User ID and then obeys the OBYMOMI file. Below is an obey file fragment (details are omitted):

...

... main start up obey file executing as super.super

TACL /in STRTMOMI, out strtlog, name/

... in the STRTMOMI file ...

```
logon user.momi
volume $vol.momi
obey OBYMOMI
```

... continue system start up

...

In the system start up files, MOMI should be started after TCP/IP, MEASURE, and the home terminal specified in OBYMOMI are all started, perhaps near the end of the system start up sequence.

In the system shutdown files, MOMI should be stopped early in the shutdown sequence and before the MEASURE subsystem is stopped.

Security

General considerations

(updated for server version 5.24 or later)

The MOMI subsystem on the NonStop system is initially launched via a TACL obey file which starts the main process named by default $MOMI. The main process starts other server processes that collectively support the MOMI environment.

When a user 'logs on' to a MOMI PC Client, a logon server process is launched that assumes given User ID. When a sensitive command is issued at the client, such as viewing the content of a Spooler Job, a process is launched from the logon server to perform the sensitive commands under the users authority - not under the authority of the main $MOMI process.

MOMI does NOT contain privileged code and should NOT be licensed via FUP. However, execution of certain operations, such as the generation of an ICMP PING message, requires a Super.Group level of authority to open a raw socket. If MOMI is started under the Super.Super (255,255) or a Super.Group (255,*) user the necessary level of authority is available. In the case where the administrator of MOMI does not desire to operate the majority of MOMI under a Super.* user, the needed "Super Group" level of authority can be obtained as needed via a separate object file named BWSSG (this object is discussed below). BWSSG is manually created during installation.

The simplest configuration is to start MOMI under Super.Super, under a Super.Group, or lastly under a "normal" User ID, perhaps one specifically created for MOMI. When MOMI is started under a "normal" User ID BWSSG is also needed (see here on how to create).

Below are the security guidelines for various files / subvolume:

BWMOMI | BWMOMIi | BWMOMIx

The MOMI executable.
Must be secured to allow Execute for all users. For example, a Guardian security string of "UUNU". Additionally, in order to allow the creation of SAVEABEND files (used in troubleshooting), READ access should also be considered for a resulting security string of "NUNU".

BWSSG

If $MOMI is started under Super.Super or the Super.Group, **this file is not needed.**

BWSSG is created by the user and is a copy (i.e. FUP DUP) of BWMOMI (or BWMOMIi, BWMOMIx) and functions as a helper program to perform operations such as TCP/IP PING (the ICMP Echo command) and adjustment (if enabled) of the System time .

File security must allow Execute for all users (i.e. "UUNU"). To allow the creation of SAVEABEND files (used in troubleshooting), READ access should also be considered for a resulting security string of "NUNU".

Subvolume of BWMOMI

MOMI creates configuration, log and work files in the subvolume where the object resides. Additionally, other files distributed with the executable may be accessed. The User ID MOMI runs under must have read / write / execute / purge / create access to this subvolume. Consider not using SAFEGUARD for this subvolume.

The files created by default in this location can be relocated with the following CONFMOMI keywords:

CNF01DB
DEFAULT-WORK-LOCATION
LOG01DB

To relocate existing files - stop MOMI, manually move the file(s), add the keyword(s) to CONFMOMI, and then restart MOMI.

Subvolume of MOMI history files (CONFMOMI keyword HSTnnDB)

MOMI must be given read / write / create / purge access to the subvolume(s) specified for these files.

$SYSTEM.SYSnn.MEAS*

MEASURE support files (such as MEASFH).
MOMI makes extensive use of MEASURE. These files must allow read / execute access.

$SYSTEM.SYSnn.EMSDIST

EMS distributor program (used to read EMS log files).
The file must allow execute access.

$SYSTEM.SYSnn.INITDLL

ProcessH support file.
MOMI uses this file in ProcessH "System" code report on Integrity and later systems. The file must allow read access.

$SYSTEM.SYSnn.MCPDLL

ProcessH support file.
MOMI uses this file in ProcessH "System" code report on Integrity and later systems. The file must allow read access.

$SYSTEM.SYSnn.MEDIASVR

Tape programmatic server.
MOMI uses this to report on tape status. The file must allow execute access.

$SYSTEM.SYSnn.NSKCOM

Virtual memory access utility.
MOMI uses this to report on virtual memory usage. The file must allow execute access.

$SYSTEM.SYSnn.RLSEID

O/S release information (i.e. G06.29.02).
The file must allow read access.

$SYSTEM.SYSnn.TMFSERV

TMF programmatic server.
MOMI uses this to report on TMF status. The file must allow execute access.

$SYSTEM.SYSnn.TSC

ProcessH support file - 16-bit code.
MOMI uses this file in ProcessH "System" code report on S-Series systems. The file must allow read access.

$SYSTEM.SYSnn.TSL

ProcessH support file.
MOMI uses this file in ProcessH "System" code report. The file must allow read access.

$SYSTEM.SYSnn.TSYSCLR

ProcessH support file.
MOMI uses this file in ProcessH "System" code report on S-Series systems.
The file must allow read access.

$SYSTEM.SYSTEM.SQLCOMP

SQL/MP compilation utility.
The display of SQL/MP information is the result of dynamic SQL statements.
This file must allow read / execute access.

SQL/MP catalog subvolumes

MOMI provides SQL/MP information by reading this subsystem's catalogs.
MOMI needs read access to the **Catalog of the System** to provide information on the SQL/MP screens. Optionally, but recommended, MOMI should be granted read access to all other SQL/MP catalogs on the system. Where access is not granted, a security error is displayed.

$SYSTEM.SYSTEM.EVENTCX *(this file is optional)*

EMS user defined cause / effect / recovery information.
MOMI makes use of this file to display EMS user defined cause / effect / recovery information. MOMI needs read / write access to this file. MOMI Client access controls the ability to display / alter information. The location of this file may be overridden with the CONFMOMI keyword EVENTCX.

$SYSTEM.SYSTEM.EVENTTX

EMS HTML cause / effect / recovery information.
MOMI makes use of this file to display EMS cause / effect / recovery information. MOMI needs read access to this file.

$SYSTEM.ZLOGnn

EMS log file subvolume.
In order to display EMS messages from $0, the log files must allow read access. Use EMSCINFO $0 to display the current log file settings and EMSCCTRL $0,<command> to alter the settings. Existing files will need to have their file security manually altered via FUP.

$SYSTEM.ZSERVICE

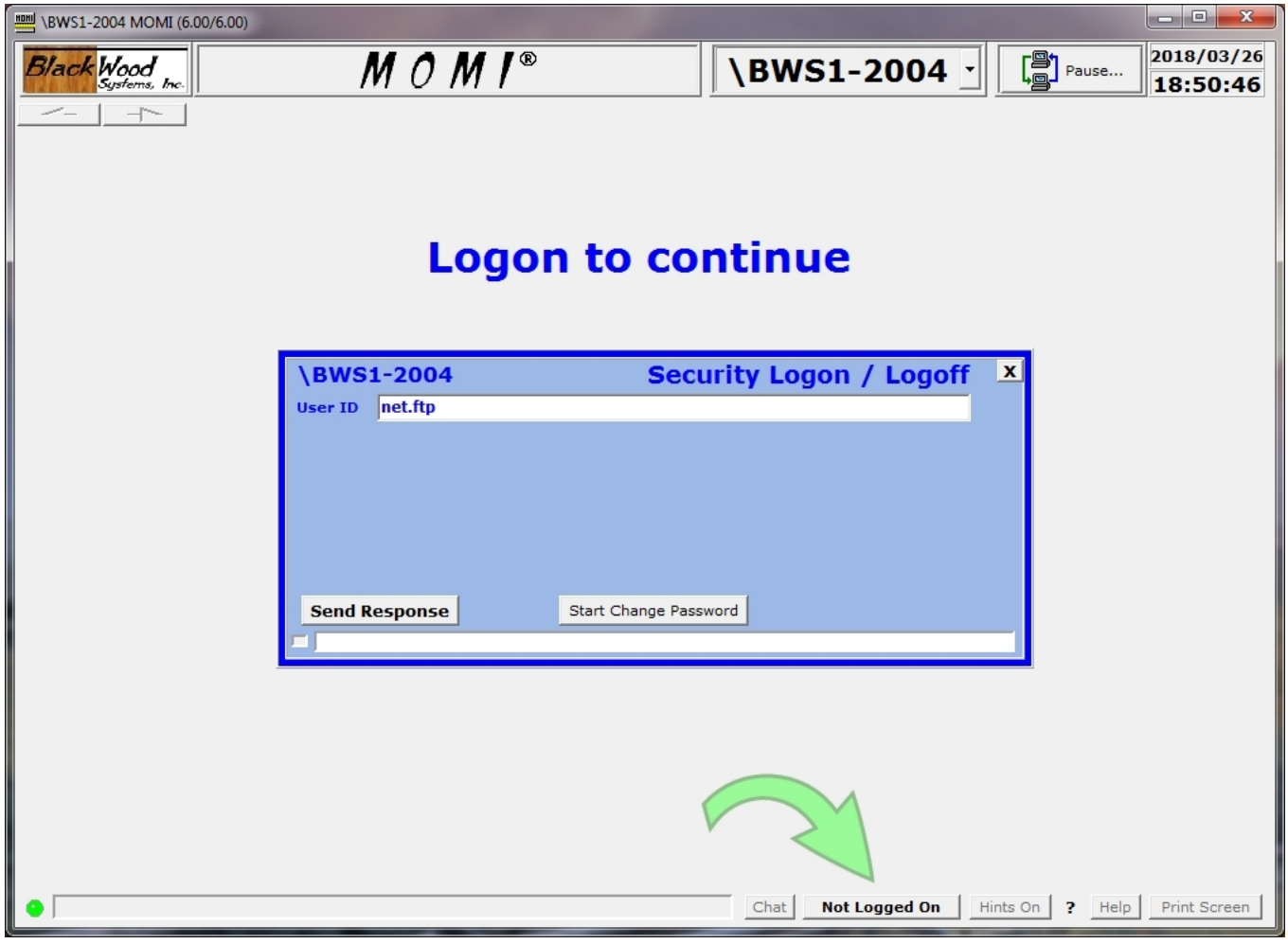
EMS log file subvolume for hardware events.

In order to display EMS messages from $ZLOG, the log files must allow read access. Use EMSCINFO $ZLOG to display the current log file settings and EMSCCTRL $ZLOG,<command> to alter the settings. Existing files will need to have their file security manually altered via FUP.

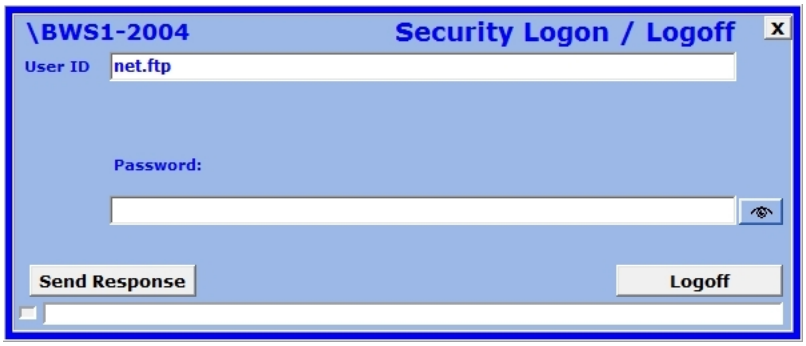
Logon / Logoff

Many operations within MOMI, such as viewing of performance information, listing volumes, files and Spooler collector status are considered non-sensitive commands and are executed in the context of the User ID that started $MOMI. Commands that perform an action, such as purging a file, viewing the contents of a file or viewing the contents of a Spooler job, are considered sensitive and must be executed under the context of a User ID and Password entered at the Client.

The Security Logon / Logoff pop-up provides a means to logon to the NonStop System. Press the button at the bottom of the screen (as shown by an arrow below) initially labeled **Not Logged On** to display the Security Logon / Logoff pop-up. The button alternately shows the name of user currently logged on. The process of logon on is conducted in the form of a dialog or a sequence of prompts, where requested data is entered, press **Send Response**, enter data, press **Send Response** and so on. The first prompt is for a **User ID** which may be a Guardian *group.user* or Safeguard alias as determined by the NonStop System security administrator. After entering the appropriate User ID press **Send Response**.



The screen will them request a **Password** or other information as directed by the prompt. Enter the requested information and press **Send Response**. This process may repeat a number of times as controlled by the security subsystem.



Once a logon is successful and considered complete, the **OK Close** button is displayed along with any informational text provided by the security subsystem. Push **OK - Close** to hide the pop-up.

A logon may not be successful if a password change is required. Follow the prompts and enter the requested information and push **Send Response**. Below, the logon is not successful until an expired password is replaced with a new one.

Passwords and other information are normally hidden from view. Press the **eye** to the right of the field to temporarily make the text visible.

Logon Notes

The MOMI server imposes the following to a logon:

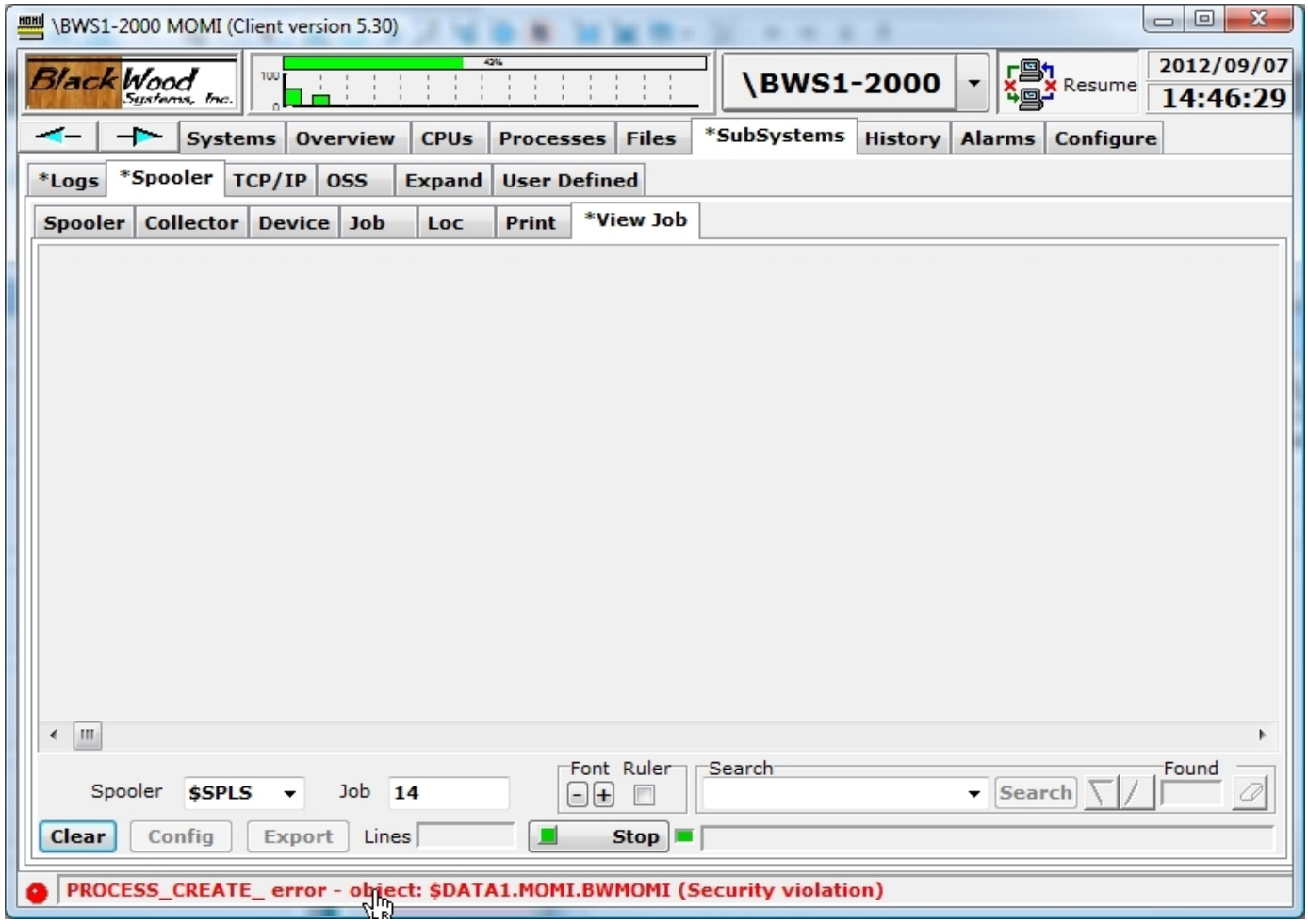
- CMON pre-logon message is requested
- CMON logon message is requested
- numeric form of User ID is **not** allowed (example 255,100)

If Client Access is enabled, a logon may be denied by MOMI after a successful logon to the NonStop System. The following messages may be returned by Client Access:

Client Access DB - User Record Not Found

Client Access DB - User Not Allowed to Logon To NonStop System

Special Note: The MOMI server creates a *logon server process* locked to the requesting MOMI PC Client. This server performs the logon operation to the system and operates under the User ID specified. It is used to start additional processes under the same User ID which perform any sensitive operation on behalf of the client. The MOMI object file must be secured to allow execute access to all users permitted to logon to MOMI, otherwise a PROCESS_CREATE_ error results. The error does not occur during logon but later during the execution of the first sensitive command. Below is an example of this error condition while attempting to view the contents of a Spooler Job:


\BWS1-2000 MOMI (Client version 5.30)
BlackWood Systems, Inc.
\BWS1-2000
Resume
2012/09/07
14:46:29
Systems
Overview
CPUs
Processes
Files
*SubSystems
History
Alarms
Configure
*Logs
*Spooler
TCP/IP
OSS
Expand
User Defined
Spooler
Collector
Device
Job
Loc
Print
*View Job
Spooler
$SPLS
Job
14
Font
Ruler
Search
Found
Search
Clear
Config
Export
Lines
Stop
PROCESS_CREATE_ error - object: $DATA1.MOMI.BWMOMI (Security violation)

Client Access

Overview

MOMI has the ability to limit the screens and features available to users of the PC Client. Virtually every screen (with some exceptions) may be turned on or off. Client Access configuration is optional but is provided to allow system administrators a means to tailor MOMI more closely to the requirements of their environment.

Users are identified to MOMI using the standard Guardian User Id or Safeguard Alias. Client Access relies on host to perform authentication. MOMI does not maintain a database of passwords.

Client Access is configured and enabled after MOMI is installed and operational. Client Access configuration information is stored on the currently connected NonStop System in the file CNF01DB.

This section describes Client Access and gives examples of turning Client Access on and configuring.

What Client Access does not do

Client Access does not grant users accessing the Nonstop System security for sensitive commands such as: stop a process, purge a file, view file contents, delete spooler jobs, etc.... Operations of this type are performed within the context of the user logon so the 'security success' of the operation is determined by the Operating System as discussed in Security Logon / Logoff.

The User ID's defined within Client Access do not store passwords.

Default Security User | Security User

The overall security of MOMI is controlled by two users known as the Default Security User and the Security User.

The user that starts the MOMI server on the Nonstop System is considered the Default Security User. There may be only one Default Security User. This user can perform and has full control over internal MOMI functions. The Default Security User is the only user that can activate or deactivate Client Access. The default may be overridden with the CONFMOMI keyword DEFAULT-SECURITY-USER.

Any user, except ones that are predefined, created within Client Access may also be enabled as a Security User. There may be zero or more Security Users. A Security User has the same authority as the Default Security User except it does not have the ability to turn off Client Access nor perform emergency database actions (see bottom of page here for details).

The Initial state of Client Access

The initial state Client Access is OFF meaning that the MOMI PC Client displays all of its screens to any user of the client and that any User with a valid User ID on the NonStop System may logon. Sensitive functions external to MOMI, such stopping a process, deleting a file, viewing the contents of a file, etc... require the user to have sufficient authority granted by the Operating System. Internal functions to MOMI, such as Alarm configuration (i.e. add / delete / change) and enabling Client Access, may only be performed by the user that starts the MOMI server.

In this state, the user that starts the MOMI server is the only user with full control and full access over the MOMI environment.

Enable Client Access

To enable Client Access, the Default Security User logs on to the MOMI PC Client, navigates to the screen Configure / Client Access / Global Settings and checks the box **Enable Client Access Checking on this System**, and also usually checks **Enable User Access Checking on this System.** Press **Change Global Client Access Settings** at the bottom of the screen to save the settings. Client Access is enabled immediately (no restart is required).

Special predefined users

Two users are automatically created and may not be deleted within Client Access:

NOT LOGGED ON determines what screens / functions are available prior to a logon or after a logoff. This is also the initial client state.

USER NOT DEFINED determines what screens / function are available when the User ID entered for logon is not found in the Client Access database. This predefined user provides a default environment or may be set to prevent logon to MOMI.

Client Access order of precedence

When the MOMI PC Client connects to a system, the predefined user *NOT LOGGED ON* determines what screens / functions are initially available. When a user attempts to log on, the Client Access database is searched in the following order, stopping at the first "match":

1) an exact match to the User ID with a case insensitive comparison
2) match User ID by wild card
3) default to *USER NOT DEFINED*

How MOMI 6.00 and later affects Client Access

Prior to MOMI version 6.00, the MOMI PC Client could display screens of meaningful data prior to logon.

MOMI 6.00 and later, by default, severely limits data displayed prior to logon. The operation is controlled by the CONFMOMI keyword CLIENT-LOCKDOWN-MODE which directs the client to display a virtually blank screen prior to logon and after logoff. This functional change effectively overrides the Client Access predefined user *NOT LOGGED ON*.

The administrator of MOMI can restore the previous manner in which MOMI operated by setting CLIENT-LOCKDOWN-MODE to false (also see this setting for additional information).

Examples

The following examples (denoted by a ➞) describe the sequence of steps necessary to perform certain activities. With the exception of the first two examples, which assume an initial configuration, all other steps assume that Client Access is enabled.

➞ The MOMI server is initially started

Results
- All screens within MOMI are available to all users
- Only the Default Security User may add/delete/operate an Alarm
- Only the Default Security User may enable Client Access checking

➞ Enable Client Access checking
- Log on to Client as User that started the MOMI server on NonStop System
- Go to the screen Configure/Client Access/Global Settings
- Check **Enable Client Access Checking on this System**
- Check **Enable User Access Checking on this System**
- Click the button **Change Global Client Access Settings**

Results
- Only the Default Security User may add/delete/operate an Alarm
- Only the Default Security User may disable Client Access checking
- The predefined user *NOT LOGGED ON* determines logged off access
- The predefined user *USER NOT DEFINED* determines logged on access

➞ Change the predefined user *NOT LOGGED ON*
- Log on to Client as User that started the MOMI server on NonStop System
- Go to the screen Configure/Client Access/Define
- Select User ID (in the upper left hand corner of the screen)
- In the drop-down box pick *NOT LOGGED ON*
- Configure as desired
- Save settings by clicking button **Change User**

Results
- Users that have not logged on to the client have settings as configured

➞ Change the predefined user *USER NOT DEFINED*
- Log on to Client as User that started the MOMI server on NonStop System
- Go to the screen Configure/Client Access/Define
- Select User ID (in the upper left hand corner of the screen)
- In the drop-down box pick *USER NOT DEFINED*

- Configure as desired
- Save settings by clicking button **Change User**

  Results
  - Users that are logged on but not known to Client Access have settings as configured

➞ Add an individual User ID
- Log on to Client as User that started the MOMI server on NonStop System
- Go to the screen Configure/Client Access/Define
- Select User ID (in the upper left hand corner of the screen)
- Enter User ID and Description
- Do not enter or select a **Member of Group**
- Configure as desired
- Save settings by clicking button **Save New User**

  Results
  - User ID when logged on has settings as configured

➞ Add a Group ID
- Log on to Client as User that started the MOMI server on NonStop System
- Go to the screen Configure/Client Access/Define
- Select Group ID (in the upper left hand corner of the screen)
- Enter Group ID and Description
- Configure as desired
- Save settings by clicking button **Save New Group**

  Results
  - A new Group ID is now available

➞ Add an User ID to a group
- Log on to Client as User that started the MOMI server on NonStop System
- Go to the screen Configure/Client Access/Define
- Select User ID (in the upper left hand corner of the screen)
- Enter User ID and Description
- Select in the drop-down box **Member of Group**
- Save settings by clicking button **Save New User**

  Results
  - The User ID is now a member of a group

➞ Allow a user to add/delete/operate an Alarm *(assumes the user was already added)*

- Log on to Client as User that started the MOMI server on NonStop System
- Go to the screen Configure/Client Access/Define
- Select User ID (in the upper left hand corner of the screen)
- Enable settings under Alarms to Display Active Alarm & Display Define & Display List
- Save settings by clicking button **Change User**

  Results
  - User ID can now Display / Create / Update alarms

➞ Assign a Security User *(assumes the user was already added)*
- Log on to Client as either a) User that started the MOMI server on NonStop System, or b) an existing Security User
- Go to the screen Configure/Client Access/Define
- Select User ID (in the upper left hand corner of the screen)
- Enable Security User
- Save settings by clicking button **Change User**

  Results
  - **User ID can now perform Display / Create / Update alarms**

➞ Limit AutoUpdate time
- Log on to Client as User that started the MOMI server on NonStop System
- Go to the screen Configure/Client Access/Global Settings
- Check box and **Limit AutoUpdate minutes to** value
- Save settings by clicking button **Change Global Client Access Settings**

  Results
  - AutoUpdate is limited by default for all users. *(This may be overridden on an individual or group basis)*

➞ Restore the default state of Client Access and Disable (use only if you really mess up)
- Log on to Client as User that started the MOMI server on NonStop System
- Go to the screen Configure / Client / Actions
- Press **Perform Emergency Database Actions**
- Press **Delete ALL Client Access and User Access Records...**
- Press confirmation button

  Results
  - Client Access disabled. Default functionality restored.

➞ Disable Client Access Checking
- Log on to Client as User that started the MOMI server on NonStop System

- Go to the screen Configure/Client Access/Global Settings
- Uncheck **Enable Client Access Checking on this System**
- Click the button **Change Global Client Access Settings**

  Results
  - Client Access disabled. Default functionality restored.
  - User profiles are still present.

Server

Overview

Information presented by the MOMI PC Client is obtained from server processes running on the NonStop System. Certain MOMI processes run continually and some are started on demand.

The MOMI subsystem on the NonStop System is initially started via the TACL obey file OBYMOMI located in subvolume where MOMI is installed. This obey file starts the main $MOMI process which serves as the overseer for the subsystem. This main process:

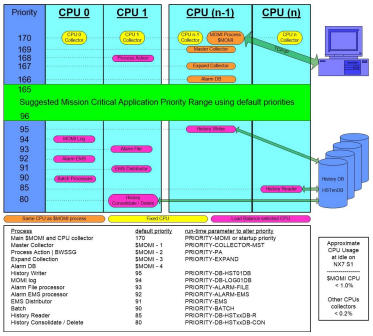
- configures a master measurement in the MEASURE subsystem
- starts an Expand collector process
- starts a collector in each processor of the system
- starts a process that writes history to the HST01DB file
- starts a process that performs the history consolidation and deletion
- starts an NSKCOM process for obtaining virtual memory statistics
- posts a 'listen' on the assigned TCP/IP process(es) and port(s) for incoming connections from the client

The MOMI PC Client communicates with the server by issuing a TCP/IP connect to the configured address and port of the $MOMI server. This main server either provides information directly or passes the request to another server it started. The response is returned to the main server which in turn is sent back to the PC Client.

Sensitive commands that require security are executed in the context of the User ID provided. When a user logs on, a server assigned to that particular client executes the logon command and assumes the authority of that particular user. From that point on, sensitive commands are executed by that server or a server it started. When a user logs off the assigned server and additional server it started all stop.

The MOMI server object may execute dynamic SQL/MP statements to obtain information from the system and user catalogs. The statements are used in support the PC MOMI Client screens Files / SQL/MP / *. The TACL obey file OBYCSQL performs an SQL/MP compile of the MOMI server object with the attribute of NOREGISTER ON. This means that the SQL/MP catalog do not reflect the MOMI executable.

Process Priority



In order for MOMI to collect information on a timely basis, it must run at a priority high enough so that the other programs on the system do not 'walk over' or interrupt operation.

The diagram above shows the relative process priorities of $MOMI and its server programs.

MOMI, to divide its workload, starts several copies of itself. The main program $MOMI, is the master traffic cop, data collector and data reporter. MOMI also starts a copy of itself in every CPU in order to collect information about that CPU. All information collected is reported back to $MOMI. $MOMI and the CPU collectors consume a fairly consistent portion of CPU resources, with $MOMI varying the most, based on incoming requests from the PC Clients.

By default, $MOMI and the collectors in each CPU run at a rather high priority of 170. This priority is set in the OBYMOMI file.

MOMI also starts other processes based on needs of the PC Client and generally run at default priorities in the 80 to 95 range and can be adjusted in the CONFMOMI file.

If you have concerns with the priority of the MOMI environment, you could run $MOMI at a lower priority than your critical production programs. Please be aware that running $MOMI at a low priority could prevent the timely collection and reporting of data (usually showing up as random interruptions). In the Main Overview screen you may see MOMI CPU collectors reporting as HUNG or STOP. Once you have tested and are comfortable with the operation of MOMI on your system, MOMI should run at a rather high priority to help to insure consistent data collection and reporting.

One of the common questions asked is, what is the CPU impact of MOMI on my system? The discussion above describes at what priorities MOMI operates. Batch type of operations that can consume large amounts of CPU are pushed to lower priorities, while processes that must operate on a regular timely basis operate at higher priorities.

So, what is the impact? With MOMI running at an idle, On an NS16000, $MOMI is at 0.20% and the collector is 0.04%. On an NB50000c, $MOMI is at 0.03% and the collector is 0.016%.

Certainly, you will see higher values. When MOMI is reading and transferring, for example, an EMS records you will see the EMS distributor and the MOMI server communicating with it both at priority 91 burning cycles. $MOMI, which handles TCP/IP communication, will be using cycles, particularly with the TCP/IP v6 stack, to transfer the resulting data. We limit the bytes per second, by processor type, $MOMI can transfer over the TCP/IP connection so place a lid on communication cycle costs.

Adjust System time via SNTP

(server version 4.17 or later)

Overview

MOMI has the capability to adjust the Nonstop System time via a Network time source.

This feature is not enabled by default.

A Network Time Protocol (NTP) server is specified to MOMI and this server queried periodically using the Simple Network Time Protocol (SNTP). SNTP is defined in RFC 2030 and MOMI specifies version 3 in the time request. This protocol provides a simplified means for obtaining time from a reliable source located either on the local network or via the Internet. NTP servers usually obtain their time from master sources such as atomic clocks or from over-the-air sources such as GPS.

How it works

MOMI queries or takes many samples from the NTP server over several minutes. The samples are processed to determine a network time. The network time is compared to the time on the Nonstop System to determine if an adjustment forward or backward is required. The Operating System call

a

SETSYSTEMCLOCK is used to perform the adjustment and is given 1) the amount to adjust and 2)

mode of 6 which directs a clock adjustment (not a clock set) regardless of the clock error.

The Nonstop Operating System provides two methods for manipulating the System time. The first method is to set the clock to a specific time, which could be an abrupt operation and is not generally performed with a 'live' System. The second method is to adjust the rate at which the clock operates without disturbing the live environment. MOMI uses the second method by default.

A way to think of the clock adjustment is "nudging" either forward or backward by slightly speeding up or slightly slowing down the System clock, but **not** jumping or forcing it to a specific time. Time nudging may be performed on an active system generally without concern. An actual set of the clock, especially going backwards in time, could have a really undesirable affect on certain subsystems such as TMF (i.e. you really don't want to do that).

A side effect of clock "nudging" is that if the System is off by more that a few minutes, particularly if the system is ahead of the network source, it may take some time (no pun intended) before an accurate system time is achieved. If the System time is really 'off', a manual operation should be considered during a maintenance window.

Should I use it?

The SNTP protocol and the MOMI implementation does not the provide the highest level of accuracy theoretically possible, but can in most situations maintain the System time much better than manual input. The code implementation strives for an accuracy of .001 seconds, which is the smallest adjustment attempted, but a resulting accuracy of about 0.1 seconds is recommended for "should I use it" purposes. However, if any of the following are true:

a. A very high level of accuracy is required, or
b. System time management is exceptionally critical

then this feature of MOMI should not be enabled.

Systems that are just plain "slow" and are constantly behind, may be closer to the network source as a result of MOMI requesting time adjustments, but may still remain behind.

What is the best time source?

Ideally, an NTP server / device on your local Network should be used to provide the highest levels of accuracy. The local network should have far less traffic and response time fluctuations than a remote device accessed via the Internet. However, the MOMI SNTP implementation attempts to compensate for response fluctuations (i.e. delays, lags).

The local NTP server / device may be a dedicated NTP server, a Windows Server or a UNIX Server. A Windows Server must be a domain controller or have the NTP server function enabled.

A public time server (NTP) may be used by searching via an Internet search engine or from the following URL -

www.pool.ntp.org

Network firewall information

SNTP uses a UDP socket over port 123. A MOMI server on the NonStop System opens the socket and sends a time request. The response is received over the same socket and port. It may be necessary to have your firewall (i.e. network environment) allow this outbound request.

Security requirement

In order to activate this feature, even for preview mode, MOMI must operate under a Super.Group or better level security.

Please see this section for more information about BWSSG if required.

How to activate

The System time management feature of MOMI may be placed in a 'preview' mode prior to full activation.

The preview mode allows confirmation of access to an NTP server and the time difference between Nonstop System and the NTP server are written to the MOMI Log. The first three steps below place MOMI in the preview mode and the fourth step allows adjustment of the System clock:

1. determine the NTP server to use.

2. add the following keywords to the CONFMOMI file (restart MOMI to take effect):

```
                    == specify TCP/IP stack name
                    ==  (not always needed)
SNTP-TCPIP-NAME  <tcpip-process-name>

                    == NTP server
SNTP-SERVER-ADDR <DNS-name>|<IP-address>
```

3. view the MOMI Log to see the difference between the System time and the Network Source (the messages are logged once an hour by default).

4. if desired, direct MOMI to perform required adjustments to the System clock by adding the following keyword to the CONFMOMI file (restart MOMI to take effect):

```
                   == allow MOMI to adjust
                   ==  the System time
SNTP-ALLOW-ADJUSTMENT
```

Common questions

Can our operators still set the System time if MOMI is configured to make adjustments?

Yes, with a consideration. The operating system forces a set of the system clock under two situations. 1) a SETTIME is performed twice within two minutes, or 2) the SETTIME value specifies a time two minutes or more off from the current System time. If the operator only issues a single SETTIME within two minutes of the current System time and expects only an adjustment, a clock set may result if MOMI is also performing an adjustment.

Can this feature be activated even if the System time is off by more than several minutes?

Yes. It may take weeks or months to finally arrive at the correct time, particularly if the System time is ahead of the Network Source.

How long do time adjustments take?

Adjusting the clock forward two minutes takes about 33 hours. Adjusting the clock back two minutes takes about 14 days.
(information from the HPE Knowledgebase)

Can an operator perform a SETTIME while MOMI is adjusting the System time?

Yes. Take note of the first question above. MOMI's time computation may be affected if the System time is externally set but the condition would self correct after a few automatic time updates.

Can I have another process (or method) also perform adjustments to the System time in addition to MOMI?

Not recommended. Only one process (or method) should be selected to maintain the System time.

Once the System time is correct and MOMI is performing adjustments, will manual adjustments be required?

Usually not. The System time should be checked periodically by an operator to insure proper operation. Messages are written to the MOMI Log and/or EMS Msgs log for normal and error conditions detected by MOMI (messages are described below).

After a maintenance window, particularly if the outage was extended or if any hardware was replaced, the System time should be checked after processors are loaded but before System is 'started'.

Will messages appear in my EMS log every hour (assuming MOMI is checking once an hour)?

Only if a time adjustment is required.

Initially, you will see messages logged to EMS every hour while your System time is brought into sync. Two messages are logged when a time adjustment occurs. The first message notifies that MOMI is going to perform an adjustment. The second message is from the Operating System indicating the time was adjusted (System ID: TANDEM.CLOCK, event number 107). Additionally, a message is also logged should the Operating System reject the attempt by MOMI to perform the time adjustment.

The System time does not seem to be as accurate as it should be and EMS event 104 (Error reading Service Processor clock, error code :nnn) is logged.

This may indicate a problem with the service processor (SP). Consider resetting the service processors in CPU 0 and 1.

MOMI logs the following messages

Normal messages

SNTP:System time n.nnnnnn seconds ahead of Network Source (<ip-address>)
SNTP:System time n.nnnnnn seconds behind Network Source (<ip-address>)
SNTP:System time 0.000000 seconds equal to Network Source (<ip-address>)

This message is written to the MOMI Log and reflects the difference between the NTP server and the Nonstop System.

SNTP:System time does not require adjustment by MOMI

This message is written to the MOMI Log and indicates that the required adjustment is below an internal threshold so MOMI does not perform any action to the System time.

SNTP:System time will be adjusted by MOMI forward n.nnnnnn seconds
SNTP:System time will be adjusted by MOMI backward n.nnnnnn seconds

This message is written to the MOMI Log and EMS Msgs log prior to performing the adjustment to the System Clock. The operating system will also generate an EMS message when MOMI issues the call to adjust the System Clock.

Error messages

SNTP:Unable to open socket - TCP/IP stack $tcpip-name

This message is written to the MOMI Log and EMS Msgs log and indicates that the TCP/IP process specified in the CONFMOMI keyword SNTP-TCPIP-NAME was either invalid or not available.

SNTP:Unable to obtain TCP/IP address from <DNS-name> | <IP-address>

This message is written to the MOMI Log and EMS Msgs log and indicates that the NTP server specified in the CONFMOMI keyword SNTP-SERVER-ADDR was invalid or the DNS entry could not be converted to a binary address (like the Nonstop System did not have access to an DNS server).

SNTP:Unable to obtain enough valid time samples-Used <count> Attempted <count>

This message is written to the MOMI Log and indicates that an insufficient number of time samples were could be used so the program continues to obtain them. No other error occurred. This message is reported in MOMI sever 5.37 or later.

This message is not expected and usually indicates the information received from the time server could not be used. Try another time server known to work with other systems.

SNTP:Unable to obtain reliable time from Network Source

This message is written to the MOMI Log and indicates that MOMI could not process the responses from the NTP server. This either indicates a temporary condition, such as an extremely poor communications link or problems at the NTP server. Occasional messages (particularly if the Internet connection is down) are expected.

Check with your network folks to insure that any and all firewalls in-between the NonStop System and the NTP server allow UDP port 123. You will probably need to provide them the TCP/IP address of the TCP/IP stack specified in SNTP-TCPIP-NAME. Companies usually place multiple firewalls within their networks and each firewall must allow the NTP data flow.

SNTP:MOMI failed to adjust System time

This message is written to the MOMI Log and EMS Msgs log and indicates that the call to adjust the System Clock failed. MOMI requires the proper security for the MOMI helper program BWSSG to perform this activity.

CONFMOMI

Overview

The CONFMOMI file is used to configure / define settings and overall features for the MOMI environment during program start-up. Settings in this file are not usually changed after initial configuration. The file is read once after each start-up of MOMI and is a simple edit format file (code 101) with this general structure.

At a minimum, the PASSWORD (unless running as mini-MOMI) and TCPIP-LISTEN are defined in this file.

After making any changes to the CONFMOMI file, in order for them to take effect, restart MOMI then check the EMS log. Any errors encountered in the CONFMOMI file generates an EMS message and possibly an ABEND.

Two sections in this document describe settings within the CONFMOMI file referred to as **Keywords** and **Advanced Keywords**. **Keywords** are what most configurations reference. **Advanced Keywords** deal with special features or situations and are not generally required.

Below are selected categories and their keywords. Not all keywords are needed to enable a particular function.

Alarms
- Overview
  - ALARM-DOMAIN-NAME
  - ALARM-EMAIL-ADDRESS-FROM
  - ALARM-EMAIL-SUBJECT
  - ALARM-SMTP-SERVER-ADDR

History
- Overview
  - HSTnnDB
  - HSTnnDB-DELETE-TIME
  - HSTnnDB-HISTORY-DUMP

MOMI Process Priorities
- Overview
  - PRIORITY-BATCH
  - PRIORITY-DB-HST01DB
  - PRIORITY-DB-HSTxxDB-CON
  - PRIORITY-DB-HSTxxDB-R
  - PRIORITY-EMS
  - PRIORITY-EXPAND
  - PRIORITY-MOMI
  - PRIORITY-PA

SNTP

General structure

The file has the following basic format of a keyword (case insensitive) usually followed by a value separated by one or more spaces.

```
keyword  <value>
```

The < > symbols indicate the location of a user defined value and the word an indication of the type of data expected. The < > symbols are **are not** included, but are used for documentation purposes. For example:

```
CPU <cpu-nbr>

actual entry -

        CPU 2
```

An optional value is indicated in the documentation by enclosing it with the [ ] symbols. For example:

```
keyword  [<value>]

documentation -

        ENABLED [<False>]

actual entries -

        ENABLED          or

        ENABLED false
```

Strings, if indicated by the keyword, must be enclosed in quotes or tics if it contains an embedded space. For example:

```
                                                == Description of system
System-description   "Production System"
```

Comments may be included in the file. Lines that begin with the following identifiers are considered comment lines:

```
!
==
```

comment

Generally, data is extracted from lines of the CONFMOMI file when a keyword is recognized. Characters past expected data are usually ignored. Comments are not supported within valid input lines (i.e. you can't embedded a comment between a keyword and an expected value).

The following are some examples how comments are used:

```
comment *** this section documents something ***

========================================
== this is the MOMI configuration file ==
========================================

Hst01db    $data1.momi.hst01db       == MOMI history database
```

Only one keyword is allowed per line. Spaces preceding or trailing the keyword are ignored. Use the sample CONFMOMI file as a formatting template (it is automatically placed in the same subvolume as MOMI object file during install).

It is not necessary to specify all the available keywords as most have default values. Advanced keywords are provided for limited or special situations.

Some keywords, in order to make it unique, use a numeric value documented as "nn" that represents a number such as 01, 02, 03, etc... . In other words, where you see "nn", such as HSTnnDB, replace it with 01, 02, 03, etc... as indicated in its documentation.

Where you see xx in a keyword, such as PRIORITY-DB-HSTxxDB-CON , it is used 'as is' and applies to all instances.

If a syntax error is encountered: a) that line is ignored, and b) two error messages are logged where the first identifies the file and the second contains the line of ignored text. Servers prior to version 6.18 log one message and later versions log two:

Line of data in Config file bad : <text>
Line of data in $vol.sub.confmomi file bad (line ignored) : <ignored text in 2nd EMS message>

Keywords

ALARM-DOMAIN-NAME

(Default = none)

ALARM-DOMAIN-NAME <domain-name>

Defines the users domain name which MOMI uses in alarm email. Specifying this keyword activates in the SMTP (i.e. outbound email) the protocol HELO, or EHLO if ALARM-SMTP-SERVER-USERNAME is specified. Some SMTP servers require this protocol (perhaps in conjunction with SPAM detection).

The Domain Name specified here is usually the last portion of your email address (i.e. what is after the @ in your email address).

By default, MOMI does not use the HELO | EHLO protocol.

```
Example:
                                          == SMTP domain name
                                          ==  My email address is
                                          ==  fred@fredsco.com so
                                          ==  my domain name is
                                          ==  fredsco.com
    ALARM-DOMAIN-NAME fredsco.com
```

ALARM-EMAIL-ADDRESS-FROM

(Default = none)

ALARM-EMAIL-ADDRESS-FROM <email-address>

Defines the return email address in an alarm email. This email address should be valid monitored address so that problems in sending email or replies from users receiving email are seen.

This keyword is required in order to support Alarm email.

```
Example:
                                  == return address in email sent
                                  == by MOMI
   ALARM-EMAIL-ADDRESS-FROM  momi-alarm@fredsco.com
```

ALARM-EMAIL-SUBJECT

(Default = none)

ALARM-EMAIL-SUBJECT <"subject-line"> | <"">

Defines the email subject line which MOMI uses in alarm email.

There are three possibilities in using this keyword:

1. If this keyword is not present, the default subject reads: MOMI Alarm
2. Use this keyword and specify a particular subject line.
3. Use this keyword but specify an empty double quote (i.e. ""). This causes the first line of the alarm message to appear as the subject (in other words the "fixed subject" is eliminated). Some email systems, particularly cell phones, limit the length of the subject line more so than the body of the email.

```
Examples:

                              == my alarm subject
    ALARM-EMAIL-SUBJECT  "Alarm from MOMI"


                              == no 'fixed' email subject, use
                              == first line of alarm text
    ALARM-EMAIL-SUBJECT  ""
```

ALARM-SMTP-BIND-ADDR

(server version 5.18 or later)

(Default = determined by system)

ALARM-SMTP-BIND-ADDR <DNS-name> | <IP-address>

Defines the DNS name or IP address of the specific NonStop TCP/IP stack that MOMI should bind to when sending alarm EMAIL. The address specified here forces the 'source' address for the message.

Note that the address specified must be valid for the TCP/IP stack MOMI uses as specified by the keyword ALARM-SMTP-TCPIP-NAME. If the IP address is not valid for the TCP/IP stack MOMI uses to send the email, a failure to send occurs.

Some NonStop Systems use communication hardware which have multiple physical Ethernet ports controlled by a single NonStop process. For example, the TCP/IP CLIM has several physical Ethernet ports and all communication occurs through only one NonStop process, for example $ZTC0. By default, a TCP/IP listen on a TCP/IP stack process (again for example $ZTC0) would allow incoming connection on every physical Ethernet port in the CLIM and an outbound connection would use a path chosen by the system, which may not be desirable. The specification of a particular IP address (or DNS name) would limit and direct all communication to a single physical Ethernet card.

Technically, this keyword causes a socket level call to bind() after a socket is opened to force an association with IP address specified.

```
Example:
                                        == alarm outbound email
                                        ==  source IP address
    ALARM-SMTP-BIND-ADDR 67.12.65.12
```

ALARM-SMTP-SERVER-ADDR

(server version 4.03 or later)

(Default = none)

ALARM-SMTP-SERVER-ADDR <DNS-name>| <IP-address>

Defines the name or IP address of the SMTP server which MOMI uses to send EMAIL. This is the server MOMI uses to send outbound email. Note that the NonStop System must have the resolver active if a DNS-Name is used.

MOMI does not specify any authentication. The SMTP server must allow unauthenticated access from the source NonStop IP address. TCP port 25 is used to communicate from the NonStop to the SMTP server.

A DNS entry that resolves to more than one IP address is interpreted as a primary and backup(s) SMTP servers. MOMI connects to the first IP address returned but if the connect fails, a connect is attempted on the second address and so on. Note that the criteria for using the additional IP addresses in a DNS name is a failure to connect, not a failure of the email. The conversion of a DNS name to an IP address(es) occurs when a EMAIL is sent (this allows changes to the address while MOMI is running).

This keyword is required in order to support Alarm email.

```
Example:
                                        == alarm outbound SMTP
                                        ==  server which resolves
                                        ==  into 64.158.128.147
    ALARM-SMTP-SERVER-ADDR smtp.fredsco.com
```

ALARM-SMTP-SERVER-PASSWORD

(server version 6.04 or later)

(Default = none)

ALARM-SMTP-SERVER-PASSWORD <password>

Defines the password used with SMTP (i.e. outbound email) authentication.

Note that password is case sensitive with most email servers.

See the keyword ALARM-SMTP-SERVER-USERNAME for details.

```
Example:
                                        == alarm outbound SMTP
                                        ==  server password
    ALARM-SMTP-SERVER-PASSWORD MySmptPassword
```

ALARM-SMTP-SERVER-TIMEOUT

(server version 4.05 or later)

(default = 30)

ALARM-SMTP-SERVER-TIMEOUT <seconds>

Specifies the maximum amount of time (in seconds) that MOMI waits for a single I/O to complete on the SMTP server. The SMTP server is used for sending Email.

A server that does not respond within the specified time generates an error 40 (Timeout).

Multiple I/Os are required in order to converse with the SMTP server. A server responding slowly may take much longer than the value specified in order to complete an Email transmission. No error is generated as long as each I/O does not take longer than this value.

```
Example:
                                == increase I/O timeout to SMTP
    ALARM-SMTP-SERVER-TIMEOUT 60
```

ALARM-SMTP-SERVER-USERNAME

(server version 6.04 or later)

(Default = none)

ALARM-SMTP-SERVER-USERNAME <user>

Defines the User Name used with SMTP (i.e. outbound email) authentication.

If the email server MOMI is using for alarm emails requires authentication, this keyword along with ALARM-SMTP-SERVER-PASSWORD specify the information. Normally, the user name is an email address.

SMTP authentication also requires that keyword ALARM-DOMAIN-NAME is also specified.

The type of authentication MOMI uses with the email server is PLAIN with no encryption. The email server may need this setting specifically enabled.

```
Example:
                                        == alarm outbound SMTP
                                        ==  server user name
    ALARM-SMTP-SERVER-USERNAME fred@fredsco.com
```

ALARM-SMTP-TCPIP-NAME

(Default = first TCPIP-LISTEN)

ALARM-SMTP-TCPIP-NAME <tcpip-process-name>

Defines the TCP/IP process name that MOMI uses to access the SMTP server which MOMI uses to send EMAIL. SMTP is outbound email.

By default, the first TCP/IP process defined to MOMI for establishing client connections is used.

This keyword would generally be used in the situation where the SMTP server is isolated or not accessible from the same subnet MOMI PC Clients connect.

Example:

```
                                    == TCP/IP process used
                                    ==  for alarm output
                                    ==  SMTP email
    ALARM-SMTP-TCPIP-NAME $ZSAM1
```

ALARM-SUSPEND-DELAY

(default = none)

ALARM-SUSPEND-DELAY <minutes>

Specifies the amount of time (in minutes) that MOMI should delay alarm processing when the subsystem is started. During this time, alarms are not checked.

The time specified is a minimum and may vary somewhat as at MOMI start up additional delays are imposed to help minimize start-up system overhead.

The MOMI PC Client may alter or even resume alarm processing delayed affected by this keyword on the screens Alarms / Active or Configure / Client / Actions.

```
Example:

    ALARM-SUSPEND-DELAY 30        == delay alarms 30 minutes
```

ALARMS-MAXIMUM-PER-DEFINITION

(server version 4.06 or later)

(Default = 100)

ALARMS-MAXIMUM-PER-DEFINITION <value16>

Specifies the maximum number of Alarms that a single definition is allowed to generate.

This value is primarily a protection mechanism to prevent a single definition from generating an excessive number of Alarms. Once the limit is reached, a special 'really red' or purple alarm is displayed and further alarms from the definition are disabled. Update the Alarm definition to 'reset' or clear the 'really red' alarm.

Example:

```
                                    == Alarms maximum to 175
    ALARMS-MAXIMUM-PER-DEFINITION 175
```

CLIENT-LOCKDOWN-MODE

(server version 5.57 or later - updated 6.00)

(Default = FALSE)

CLIENT-LOCKDOWN-MODE TRUE | FALSE

Specifies if the MOMI PC Client should operate in a mode where limited data is requested and displayed prior to a user logon. Note that this setting has no effect on earlier versions of the client (earlier client versions can be prevented from connecting with the keyword CLIENT-MINIMUM-V6).

The processing of CLIENT-LOCKDOWN-MODE occurs prior to the settings of Client Access (if enabled).

True directs the MOMI PC Client to generally limit the display to a blank screen prior to a user logon.

False allows the MOMI PC Client to display data prior to a logon. However, Client Access (if enabled) still determines what screens are available prior to a logon (known as the **NOT LOGGED ON** state).

Installations of MOMI prior to version 6 do not contains this keyword by default, which means it effectively has the value of FALSE. New installations contain this keyword and it is set to TRUE. The MOMIFTP utility, used to install MOMI software on the NonStop System, by default does not alter an existing CONFMOMI file.

This keyword also affects mini-MOMI which is the limited feature set of the software when operated without a password. Either remove or set this keyword to False to allow the display of data without requiring logon.

Example:

```
                              == client may NOT display
                              ==  data prior to user
                              ==  logon
    CLIENT-LOCKDOWN-MODE true
```

CLIENT-MINIMUM-V6

(server version 6.00)

(Default = FALSE)

CLIENT-MINIMUM-V6 TRUE | FALSE

Determines if the server accepts connections from a MOMI PC Client (or the screen saver) with a version less than 6.00. Enabling this setting effectively means that MOMI client software prior to version 6.00 is not permitted.

True causes the server to reject connections from MOMI clients (or the screen saver) prior to version 6.00.

False means the server does not check the client version when it connects.

Example:

```
                              == clients prior to version 6.00
                              ==  are not permitted to connect
    CLIENT-MINIMUM-V6 TRUE
```

CLIENT-SCREEN-SAVER

(server version 6.00)

(Default = TRUE)

CLIENT-SCREEN-SAVER TRUE | FALSE

Determines if the MOMI PC screen saver is allowed to operate even when CLIENT-LOCKDOWN-MODE has been enable. Note that this setting has no effect on screen saver versions prior to 6.00.

True permits the screen saver to operate, effectively allowing it to override CLIENT-LOCKDOWN-MODE and display data.

False causes the screen saver to display a message that the function has been disabled.

```
Example:

                                == screen saver not
                                ==  permitted
    CLIENT-SCREEN-SAVER FALSE
```

CNF01DB

(Default = subvolume of BWMOMI object)

CNF01DB <file-name>

Specifies the name of the MOMI configuration database. This database is used to store MOMI configuration information.

This keyword is not required. The file is automatically created if not found at the location (or default) specified. MOMI must have create/write/read security access to this file.

The I/O activity on this file is generally low.

Note that if this keyword is used and an existing file is present in the default location, you must stop MOMI, manually move the file, add the keyword and then restart MOMI.

Example:

```
                                == MOMI configuration database
    CNF01DB  $data1.momi.cnf01db
```

CPU-NOT-PRESENT

(server version 5.35 or later)

(Default = none)

CPU-NOT-PRESENT <cpu-nbr>

This parameter is intended for special circumstances where a nonstandard configuration is required.

Specifies if MOMI should consider a processor AS not present.

By default, and under normal systems configurations, processors in a NonStop System are contiguously populated starting with CPU 0 and sequentially going up to a maximum of CPU 15 (16 total for the system). However, certain configurations may have a hole or gap and processors are not physically populate sequentially. MOMI at start-up considers any 'missing' processor as down between (and inclusive) CPU 0 and the highest numerical value determined to be running (i.e. the CPU's O/S is loaded and responding).

A processor defined with this keyword is omitted or not initialized within MOMI and in turn does not appear on MOMI screens. After MOMI is started, if a processor is physically loaded (i.e. O/S responding) MOMI will see the CPU UP system message and report its activity regardless of this keyword setting.

Note that at MOMI start-up if a processor is defined with this keyword and is physically present (i.e. O/S is responding) it is not reported.

Multiple processors s are defined by multiple uses of this keyword in the CONFMOMI file (only one cpu-nbr per line may be specified).

Example:

```
                                        == System missing CPU 4 & 5
    CPU-NOT-PRESENT 4
    CPU-NOT-PRESENT 5
```

DEFAULT-SECURITY-USER

(server version 4.03 or later)
(updated client version 6.09)

(Default = user that started MOMI)

DEFAULT-SECURITY-USER <User-ID>

Specifies the User ID, i.e. Guardian or Safeguard alias, that is the Default Security User for MOMI.

The User ID is not case sensitive. It may contain a wild-card (as of client 6.09).

The user that starts the MOMI server on the NonStop System is considered the Default Security User. This user has the initial ability to perform configuration and other operations such as creating alarms in the MOMI PC Client. Other users may also be given full or partial authority to perform these actions. See Client Access for detailed information about the Default Security User.

In the situation where the User ID that started the MOMI server, for example Super.Super, is not directly accessible (i.e. no one may logon to that ID) this keyword solves the chicken-and-egg problem by providing an alternate User ID with the initial authority to "configure and define" within the client.

Examples:

```
                              == fred is default security user
    DEFAULT-SECURITY-USER admin.fred

                              == any super.group is the
                              ==  default security user
    DEFAULT-SECURITY-USER super.*
```

DEFAULT-WORK-LOCATION

(server version 4.11 or later)

(Default = vol/subvol when MOMI was started)

DEFAULT-WORK-LOCATION <$vol.subvol>

Specifies an alternate location for MOMI to place work files which typically are MEASURE data files. The initial file created here is DCPUS.

The default value for this keyword is the default vol / subvol specified when MOMI was started which normally is the subvolume of the BWMOMI program.

Shortly after startup, MOMI on the NonStop System defines a measurement (DCPUS file) configured with a variety of entities (CPU, Process, TMF, etc...). This measurement does not specify an interval but is effectively used for LISTACTIVE purposes. This file may become 100% full which is not a concern.

The User ID MOMI runs under must have read / write / execute / purge / create access in this subvolume. It is recommended that SAFEGUARD is not used for this subvolume.

Example:

```
                              == $data1.momiwork for temp files
    DEFAULT-WORK-LOCATION $data1.momiwork
```

DISABLE-MEAS-SQLPROC

(server version 4.12 or later)

(Default = true)

DISABLE-MEAS-SQLPROC TRUE | FALSE

Determines if MOMI is allowed to report on the SQLPROC (SQL Process) entity via MEASURE.

True means this entity is not allowed to operate. False allows this entity to operate.

Certain versions of the operating system may have issues with this MEASURE entity. As a precaution, it is not enabled by default.

Example:

```
    DISABLE-MEAS-SQLPROC FALSE              == enable SQLPROC
```

DISABLE-MEAS-SQLSTMT

(default changed in server version 4.06 or later)
(Default = true)

DISABLE-MEAS-SQLSTMT TRUE | FALSE

Determines if MOMI is allowed to operate an SQL Statement measurement.

True means this entity is not allowed to operate. False allows this entity to operate.

Historically, some versions of the operating system may have issues with this MEASURE entity. As a precaution, it is not enabled by default. Prior to enabling, check your operating system version for any applicable updates or alerts.

MEASURE documentation warns that measuring SQL Statement incurs a higher CPU cost than other entities

Example:

```
    DISABLE-MEAS-SQLSTMT FALSE        == enable SQL statement
```

DNS-TCPIP-NAME

(server version 5.31 or later)

(Default = first TCPIP-LISTEN)

DNS-TCPIP-NAME <tcpip-process-name>

Defines the TCP/IP process stack name that MOMI uses when resolving DNS names into an IP address.

By default, the first TCP/IP process defined to MOMI for establishing client connections is used.

When MOMI encounters a DNS name and requires an IP address, the socket library is directed to use this stack name to issue the query.

This keyword would generally be used when an explicate TCP/IP stack should be used for DNS resolution or the subnet(s) defined to listen for MOMI PC Client connections does not support DNS.

Example:

```
                                    == TCP/IP process used
                                    == for alarm output
    ALARM-SMTP-TCPIP-NAME $ZSAM1
```

ENCRYPTION-ALGORITHM

(server version 5.06 or later - updated 5.57)

(Default = 2 / S-Series or older 5)

ENCRYPTION-ALGORITHM 0 .. 5

Specifies the type of data encryption used between the MOMI PC Client and the MOMI Server in the TCP/IP data flow.

The following settings are available:

0 - None

1 - FAS

2 - AES

3 - FAS No Port

4 - AES No Port

5 - FAS 2

0 - None means that normal MOMI traffic is not encrypted. This provides the highest level of performance with the lowest CPU cost.

1 - FAS uses a proprietary algorithm with a 256 bit asymmetric key to provide a "Fast and Simple" encryption of the data. A fairly high level of performance is obtained with a modest CPU cost.

2 - AES uses the Advance Encryption Standard industry algorithm with a 256 bit key.

3 - FAS No Port uses a proprietary algorithm with a 256 bit asymmetric key to provide a "Fast and Simple" encryption of the data. A fairly high level of performance is obtained with a modest CPU cost. This algorithm may be automatically selected if the TCP/IP dynamic port varies between the MOMI PC Client and MOMI Server.

4 - AES No Port uses the Advance Encryption Standard industry algorithm with a 256 bit key. This algorithm may be automatically selected if the TCP/IP dynamic port varies between the MOMI PC Client and MOMI Server.

5 - FAS 2 uses a proprietary algorithm with a 256 bit asymmetric key to provide a "Fast and Simple" encryption of the data. A fairly high level of performance is obtained with a modest CPU cost. This implementation is improved over the original 1 - FAS algorithm.

Regardless of the chosen setting note the following:

- Logon information is always (and always has been) encrypted.
- Header and control information in the data flow may not be encrypted.
- The encryption level may be automatically downgraded to comply with US export controls.

Generally speaking, MOMI takes advantage of the multiple processing running within its environment and pushes encryption processing down-to the level where a request is actually serviced. This push helps to limit the amount of encryption overhead that occurs in the higher priority processes of a MOMI subsystem. See Process Priority for additional information on how MOMI divides its workload.

```
Example:
                                                  == Encrypt type AES
    ENCRYPTION-ALGORITHM 2
```

EVENTCX

(server version 4.10 or later)

(Default = $SYSTEM.SYSTEM.EVENTCX)

EVENTCX <file-name>

Specifies an alternate location an existing EMS user defined cause and recommended action file. Usage of this keyword means the file is exclusively used by MOMI since the location is non-standard.

This optional file, defined by HPE for the Viewpt subsystem, provides a means of user defined verbiage for an EMS event. If the file is 1) present and 2) if a record is found with the same owner, subsystem and event number for a given EMS event then the content of the record is displayed.

On a system that does not have an EVENTCX file, the FUP obey file FUPEVTCX (located in the subvolume where MOMI is installed) is provided to allow its creation. The following command would be used to create the file:

FUP / in FUPEVTCX /

See EMS EVENTCX for additional information.

```
Example:

                                          == alternate location
                                          == for EVENTCX
    EVENTCX  $data1.momi.eventcx
```

EXPAND-IO-DELAY

(Default = 5)

EXPAND-IO-DELAY <.01 seconds>

Specifies the amount of time, in an implied .01 seconds, of delay in-between individual I/Os used to gather information about Expand objects.

This delay reduces the CPU in gathering information by spreading the I/Os out over a longer period of time

```
Example:
                                    == delay 2 seconds between
                                    ==  I/Os
    EXPAND-IO-DELAY 200
```

EXPAND-NETWORK-NAME

(Default = spaces)

EXPAND-NETWORK-NAME <name>

Specifies a unique name on an Expand network that MOMI uses to group systems for reporting purposes.

The name may be up to eight characters.

It is possible that a MOMI Client could be connected to more than one Expand network. For network diagram or other reporting purposes, MOMI needs to know how to group this information. This parameter provides that grouping.

All MOMI servers in this 'network' must use the same name in order for the client to properly establish network node relationships.

```
Example:

    EXPAND-NETWORK-NAME USA                == network name USA
```

EXPAND-UPDATE-INTERVAL-ERR

(Default = 30)

EXPAND-UPDATE-INTERVAL-ERR <seconds>

Specifies the frequency in seconds in which Expand objects are updated after an error condition has been detected.

By default, MOMI uses a faster update interval during error conditions such as when a line or path down are present. This allows MOMI to report updates more quickly. After the error condition is corrected and detected by MOMI, the normal update interval is used.

See the parameter EXPAND-UPDATE-INTERVAL for additional considerations

Example:

```
                                    == update every 15 seconds
    EXPAND-UPDATE-INTERVAL-ERR 15
```

EXPAND-UPDATE-INTERVAL

(Default = 60)

EXPAND-UPDATE-INTERVAL <seconds>

Specifies the frequency in seconds in which Expand objects are updated. The higher the value, the longer in-between updates.

Lower values result in more frequent updates, but result in higher CPU usage.

The time spent gathering data also lengthens the update frequency.

Example:

```
                                    == update every 30 seconds
    EXPAND-UPDATE-INTERVAL 30
```

FNAME-SPOOLER-n

(Default = $SYSTEM.SYSTEM.SPOOL)

FNAME-SPOOLER-n <object-file-name-pattern>

Specifies the standard SPOOLER supervisor object file name. MOMI identifies SPOOLER subsystems by object file name.

Usage of this parameter will override the default value. If the default is required along with other entries, at least two entries are required (see example below).

Multiple file names may be specified when more than one object file is in use, such as during a version upgrade. A file name pattern may also be specified

The n in the parameter name provides a means to create unique parameter names. Up to 9 object file names may be specified.

```
Examples:

                                == Define spooler supervisor files
    FNAME-SPOOLER-1 $SYSTEM.SYSTEM.SPOOL

    FNAME-SPOOLER-2 $DATA1.SPLFILES.SPOOL
```

FNAME-SPOOLER-PLUS-n

(Default = none)

FNAME-SPOOLER-PLUS-n <object-file-name-pattern>

Specifies the SPOOLER 'plus' supervisor object file name. MOMI identifies SPOOLER subsystems by object file name.

Use of this parameter is required to identify a SPOOLER 'plus' subsystem. SPOOLER 'plus' is an optional product that requires certain changes in the access method versus the standard SPOOLER.

Multiple file names may be specified when more than one object file is in use, such as during a version upgrade. A file name pattern may also be specified.

The n in the parameter name provides a means to create unique parameter names. Up to 9 object file names may be specified.

Examples:

```
    FNAME-SPOOLER-PLUS-1   $SYSTEM.SPLUS.SPOOL
    FNAME-SPOOLER-PLUS-2   $*.*.SPOOL
```

HSTnnDB-DELETE-TIME

(revised in server version 5.20 or later)

HST01DB-DELETE-TIME <days> (default = 14)
HST02DB-DELETE-TIME <days> (default = 28)
HST03DB-DELETE-TIME <days> (default = 60)
HST04DB-DELETE-TIME <days> (default = 600)

Specifies how long (in days) data is retained before it is deleted from the history database. Another way to think about it is this value determines how far back history is allowed to accumulate.

If the number of days specified is greater than zero, data is automatically deleted based on the specified time criteria.

If the number of days specified is zero, data is not deleted from the file based on its age. This setting allows a history file to fill all its available file disk space. The oldest data in the file is automatically deleted to make room for new data.

If the number of days specified is -1, data is not deleted from the file based on its age and new data is no longer written to the file. This setting is provided to support the situation where it is necessary to examine a saved history file (usually in a second copy of the MOMI server environment).

When data is written to a history file but the file had reached its maximum capacity, either maximum number extents or no more disk space is available, MOMI automatically and always deletes the oldest data to make room for the new data. MOMI does not generate any out of disk space or file full messages.

Examples:

```
hst01db-delete-time 7    == keep HST01DB history 7 days
hst02db-delete-time 30   == keep HST02DB history 30 days
```

HSTnnDB-HISTORY-DUMP

HST01DB-HISTORY-DUMP <seconds> (default = 120) 2 minutes
HST02DB-HISTORY-DUMP <seconds> (default = 600) 10 minutes
HST03DB-HISTORY-DUMP <seconds> (default = 3600) 1 hour
HST04DB-HISTORY-DUMP <seconds> (default = 86400) 1 day

Specifies the amount of time (in seconds) for MOMI history windows.

MOMI initially writes history to the HST01DB file at the interval specified by HST01DB-HISTORY-DUMP. MOMI reads data from HST01DB and consolidates it into HST02DB at the interval specified by HST02DB-HISTORY-DUMP. This process is repeated for the other history files.

For example, using the default values, MOMI reads 10 minutes of history from the HST01DB file and writes the consolidated data into HST02DB. MOMI then reads 1 hour of history from HST02DB and writes the consolidated data into HST03DB. MOMI reads 1 day of history from HST03DB and writes the consolidated data into HST04DB. If a history file reaches capacity or no more disk space is available, the oldest data in the file is automatically deleted to free up space.

A value of zero (or less than an internal minimum) turns history dumping off for that file.

Values given for each history file should be progressively larger than the next and a multiple of the previous (i.e. follow the pattern as seen in the default values). History always starts with HST01DB, followed by HST02DB and so on.

The value entered here will effectively still be a multiple of the raw data collection time, which defaults to 10 seconds.

```
Examples:

                                    = dump HST01DB every 30 secs
    hst01db-history-dump 30

                                    = dump HST01DB every 2 mins
    hst02db-history-dump 120
```

HSTnnDB

HST01DB <file-name> (optional but must be first)
HST02DB <file-name> (optional but must be second)
HST03DB <file-name> (optional but must be third)
HST04DB <file-name> (optional but must be fourth)

Specifies the name of the MOMI history database(s) which are used to maintain various statistics.

The file is automatically created if not found at the location specified. MOMI must have create/write/read security access to the files and the subvolume location.

The HST01DB file is used by MOMI to dump the initial history statistics. If history is enabled, the HST01DB file must be specified first, followed optionally by HST02DB as second and so on.

Volumes such as $SYSTEM and $AUDIT are not recommend for MOMI history files as performance of these volumes should be reserved for the overall system.

See Resizing_History_Files for default and resizing information.

Examples:

```
                              == assign MOMI History Files
HST01DB $data1.momi.hst01db
HST02DB $data1.momi.hst02db
HST03DB $data1.momi.hst03db
HST04DB $data1.momi.hst04db
```

IIT-NORMAL-INTERVAL-INCREMENT

(Default = 10, minimum 10)

IIT-NORMAL-INTERVAL-INCREMENT <seconds>

Specifies the number of seconds that a normal Measure interval is to increase. An attempt to set the value below the minimum is forced to the minimum.

Measurements started on demand run for an initial period of time, stop, report data, restart the measurement but increase the time interval by an increment value. The computed time interval is capped at a maximum amount. This keyword determines the amount a measurement interval is increased.

A normal Measure are the time periods associated with measurements such as Disk Entity, DiskFile Entity and File Entity. It is not associated with time periods for details such as Process Detail.

Example:

```
                                    == increase by 20 seconds
    IIT-NORMAL-INTERVAL-INCREMENT 20
```

IIT-NORMAL-MAXIMUM

(Default = 240, minimum 30)

IIT-NORMAL-MAXIMUM <seconds>

Specifies the maximum number of seconds of a normal measurement interval. An attempt to set the value below the minimum is forced to the minimum.

Measurements started on demand run for an initial period of time, stop, report data, restart the measurement but increase the time interval by an increment value. The computed time interval is capped at a maximum amount. This keyword determines the maximum amount of a measurement interval.

A normal Measure are the time periods associated with measurements such as Disk Entity, DiskFile Entity and File Entity. It is not associated with time periods for details such as Process Detail.

Example:

```
                              == increase stops at 60 seconds
    IIT-NORMAL-MAXIMUM 60
```

IIT-NORMAL-STARTING

(updated server version 5.57)

(Default = 15, minimum 10)

IT-NORMAL-STARTING <seconds>

Specifies the initial number of seconds for a normal measurement interval. An attempt to set the value below the minimum is forced to the minimum.

Measurements started on demand run for an initial period of time, stop, report data, restart the measurement but increase the time interval by an increment value. The computed time interval is capped at a maximum amount. This keyword determines the first measurement interval.

A normal Measure are the time periods associated with measurements such as Disk Entity, DiskFile Entity and File Entity. It is not associated with time periods for details such as Process Detail.

Example:

```
                              == first interval is 20 seconds
    IIT-NORMAL-STARTING 20
```

INTERVAL

(default = 10)

INTERVAL <seconds>

Specifies the amount of time (in seconds) in-between each gathering of the main body of information collected by MOMI. This main body of information is primarily the status of each CPU, process, disk, OSS CPU and OSS Name Server on the system.

The value has a range of 5 to 180 seconds.

A value smaller or larger than allowed is set to the respective minimum or maximum permitted.

Internally, MOMI will generally round this value. The accepted values are 5, 10, 20, etc...

The default value is recommended.

```
Example:

    INTERVAL 5                      == stats every 5 seconds
```

IO-STUCK-EXCLUDE-n

(server version 4.06 or later)
(Default = none)

IO-STUCK-EXCLUDE-n <object-file-name-pattern>

Specifies object file names to exclude from the determination if a process is 'stuck' in an I/O.

Processes with object files located in $SYSTEM.SYSTEM and $SYSTEM.SYSnn are excluded automatically.

A process which remains in a wait state of %004, and is not consuming any processor time (i.e. no CPU busy time), may possibly be caught in some sort of I/O which is not completing. This may or may not indicate a problem.

Multiple file names may be specified when more than one object file is in use, such as during a version upgrade. A file name pattern may also be specified.

The n in the parameter name provides a means to create unique parameter names. Up to 9 object file names may be specified.

Examples:

```
                                        == exclude fredobj
    IO-STUCK-EXCLUDE-1 $DATA1.DEVEL.FREDOBJ
                                        == exclude prodcatb
    IO-STUCK-EXCLUDE-2 $*.*.PRODCATB
```

IO-STUCK-INTERVAL

(default = 30)

IO-STUCK-INTERVAL <seconds>

Specifies the amount of time (in seconds) that a process must be in a wait state of %004 before it is declared as possibly 'stuck' in an I/O.

The value has a lower range of twice the INTERVAL, which defaults to 10 seconds.

A process which remains in a wait state of %004, and is not consuming any processor time (i.e. no CPU busy time), may possibly be caught in some sort of I/O which is not completing. This may or may not indicate a problem.

See IO-STUCK-EXCLUDE-n for additional information.

Example:

```
    IO-STUCK-INTERVAL 60      == define 'stuck' as 60 seconds
```

LICENSE-KEY

(server version 6.17h or later)

(REQUIRED for full operation)

LICENSE-KEY license-string

Specifies a license key string used to control the features, options, and operation of MOMI. The license key may allows MOMI to operate only on a specific NonStop System, or on any NonStop system for a limited period of time.

The license key must be entered EXACTLY as supplied on one single line. There is a single space after the keyword LICENSE-KEY and no quotes around entered text.

After the Server is started and the Client connects, the status of the license key may be viewed on the Diagnostics page.

References to the keyword PASSWORD will be phased out in favor of LICENSE-KEY but either refer to the same thing.

Example:

```
                                          == MOMI license key
    LICENSE-KEY 2932938045092384...420394852093845848
```

LOG01DB

(Default = subvolume of BWMOMI object)

LOG01DB <file-name>

Specifies the name of the MOMI log file. This file stores events generated by MOMI.

This keyword is not required. The file is automatically created if not found at the location (or default) specified. MOMI must have create/write/read security access to this file.

The I/O activity on this file can be moderate if logging is enabled on high activity alerts.

Data is retained until the file is full (i.e. reaches 100%). When a full file is encountered, oldest data is deleted to make room for a new record. If desired, to store a larger set of data increase the maximum number of extents for the file.

Note that if this keyword is used and an existing file is present in the default location, you must stop MOMI, manually move the file, add the keyword and then restart MOMI.

```
Example:
                                    == MOMI log file location
    LOG01DB $data1.momi.log01db
```

PASSWORD

(REQUIRED for full operation)

PASSWORD password-string

Specifies a password string used to control the operation of MOMI. The password either allows MOMI to operate on a specific NonStop System, or on any NonStop system for a limited period of time. Optional features and other program capabilities can also be controlled by the password.

Password must be entered EXACTLY as supplied. The entire password is on one single line. There is a single space after the keyword PASSWORD and no quotes around password.

After the Server is started and the Client connects, the status of the password may be viewed on the Diagnostics page.

Usage of this keyword will be phased out and replaced with the keyword LICENSE-KEY but either refer to the same thing.

```
Example:

                                          == MOMI password
    PASSWORD 2932938045092384...420394852093845848
```

PATHWAY-TCP

(Default = none)

PATHWAY-TCP <pathtcp2-object-file-name>

Defines an additional, or secondary, object file location for a Pathway TCP.

By default, MOMI only considers $SYSTEM.SYSTEM.PATHTCP2 as the object file for Pathway TCP process.

In order for MOMI to identify a Pathway TCP process, the object file name is examined. This keyword supports the situation where an additional or perhaps customer modified copy of the original Pathway TCP object is maintained. MOMI considers the standard location and this secondary location, if specified, as Pathway TCP object files.

Example:

```
                                    == Location of our copy
                                    == of PATHTCP2
    PATHWAY-TCP $data1.alttcp.pathtcp2
```

PRIORITY-ALARM-EMS

(server version 4.06 or later)

(Default = 92)

PRIORITY-ALARM-EMS <value-1-199>

Specifies the process priority of a MOMI server for Alarm operations relating to EMS messages.

This server generally reads the EMS log and scans for events defined to trigger and clear Alarms.

Example:

```
                                == Alarm EMS priority at 101
    PRIORITY-ALARM-EMS 101
```

PRIORITY-ALARM-FILE

(server version 4.06 or later)

(Default = 93)

PRIORITY-ALARM-FILE <value-1-199>

Specifies the process priority of a MOMI server for Alarm operations relating to Disk and File.

This server generally issues commands to obtain disk volume and file information to trigger and clear defined Alarms.

Example:

```
                                == Alarm File priority at 102
    PRIORITY-ALARM-FILE 102
```

PRIORITY-BATCH

(Default = 90)

PRIORITY-BATCH <value-1-199>

Specifies the process priority of a MOMI server for 'batch' type operations within MOMI. Batch operations are operations where a server MOMI is started for demand type of operations such as measuring files, processes, ProcessH, etc...

```
Example:

    PRIORITY-BATCH 89                   == batch at priority 89
```

PRIORITY-DB-HST01DB

(Default = 95)

PRIORITY-DB-HST01DB <value-1-199>

Specifies the process priority of a MOMI server for database operations relating to the HST01DB file. This file is used as the initial location for storing performance information.

Specifically, this keyword controls the priority for writing new data to the file and reading the most current data. It is recommended that the priority specified here be the highest of the MOMI servers, operating in the batch priority area, to insure data is written timely to disk.

```
Example:

    PRIORITY-DB-HST01DB 94           == hst01db priority at 94
```

PRIORITY-DB-HSTxxDB-CON

(Default = 80)

PRIORITY-DB-HSTxxDB-CON <value-1-199>

Specifies the process priority of a MOMI server for database operations relating to the data consolidation of the HSTxxDB files. These files are used for storing performance information.

Specifically, this keyword controls the priority for consolidating history from one file into another and deleting old data. This is an ongoing but fairly consistent operation and should occur at a fairly low priority.

Example:

```
                              == history consolidation at 75
    PRIORITY-DB-HSTXXDB-CON 75
```

PRIORITY-DB-HSTxxDB-R

(Default = 85)

PRIORITY-DB-HSTxxDB-R <value-1-199>

Specifies the process priority of a MOMI server for database operations relating to the HSTxxDB files. These files are used for storing performance information.

Specifically, this keyword controls the priority for initial, or possibly long running, read operations against the files. It is recommended that this priority be rather low.

Example:

```
    PRIORITY-DB-HSTXXDB-R 87          == history read at 87
```

PRIORITY-DB-LOG01DB

(server version 4.06 or later)

(Default = 94)

PRIORITY-DB-LOG01DB <value-1-199>

Specifies the process priority of a MOMI server for database operations (write and delete of old data) relating to the LOG01DB file.

This file stores events generated by MOMI.

```
Example:

    PRIORITY-DB-LOG01DB 100          == log01db priority at 100
```

PRIORITY-EMS

(Default = 91)

PRIORITY-EMS <value-1-199>

Specifies the process priority of a MOMI server for EMS message retrieval within MOMI. EMS distributors and a MOMI server are started for obtaining this data.

Example:

```
    PRIORITY-EMS 50                    == EMS at priority 50
```

PRIORITY-EXPAND

(Default = Main MOMI - 3)

PRIORITY-EXPAND <value-1-199>

Specifies the process priority for the Expand Server/Collector. This process collects Expand information and services Client requests.

The default value is recommended.

Example:

```
    PRIORITY-EXPAND 167             == expand at priority 167
```

PRIORITY-MOMI

(Default = as started)

PRIORITY-MOMI <value-1-199>

Specifies the priority for $MOMI.

While the process priority of $MOMI is set when the process is started, sometimes $CMON can alter or prevent the priority given at run-time. This keyword causes the program to alter its own priority directly via a GUARDIAN procedure.

This keyword will override the value specified when the program was started.

Example:

```
    PRIORITY-MOMI 168              == MOMI at priority 168
```

PRIORITY-PA

(Default = Main MOMI - 2)

PRIORITY-PA <value-1-199>

Specifies the process priority for Process Actions.

Process Actions are short duration 'command' type of activities. For example:

logon
process stop / abend / altpri / etc...
BWSSG (which handles Ping, Trace Route, maintain system time, etc...)

The handling of Process Actions at a rather high priority help insure they are processed in a timely manner. A low priority could cause them to be hindered by low-priority intense batch type of activity in a CPU.

Example:

```
    PRIORITY-PA 168                 == Process Actions at 168
```

SELECTION-DEFAULT-VHS

(server version 4.11 or later)

(Default = none)

SELECTION-DEFAULT-VHS <$vol.subvol>

Specifies the default location the file selector displays when selecting a VHS log file to display. VHS, or the Virtual Hometerm Subsystem, provides a stable location for processes to log message usually related to a problem or status update.

Usage of this parameter may aid users in selecting the proper log file as the File Locator on the screen VHS Log as it will automatically drill-down to the location specified.

```
Example:

    SELECTION-DEFAULT-VHS $DATA1.VHSLOG
```

SNTP-ALLOW-ADJUSTMENT

(server version 4.16 or later)

(Default = no adjustment)

SNTP-ALLOW-ADJUSTMENT

Allows MOMI to adjust the time on the Nonstop System.

This keyword is provided as an 'extra' step to insure that it is desired for MOMI to make adjustments to the System time.

MOMI adjusts the System time, it does not set the time. Adjusting the System time means that MOMI compares the current time at the Network time source to the Nonstop System and based on that result "nudges" the System clock either forward or backward. "Nudges" is effectively slightly speeding up or slowing down the system clock, but not forcing it to a specific time. This action may be performed on an active system without concern for setting the clock backward, which certain systems such as TMF could not tolerate.

By default, MOMI does not alter the System time.

```
Example:
                                        == Allow MOMI to adjust
                                        ==  the system time
    SNTP-ALLOW-ADJUSTMENT
```

SNTP-BIND-ADDR

(server version 5.19 or later)

(Default = determined by system)

SNTP-BIND-ADDR <DNS-name><IP-address>

Defines the DNS name or IP address of the specific NonStop TCP/IP stack that MOMI should bind to when sending SNTP messages. The address specified here forces the 'source' address for the message.

Note that the address specified must be valid for the TCP/IP stack MOMI uses as specified by the keyword SNTP-TCPIP-NAME. If the IP address is not valid for the TCP/IP stack MOMI uses to send the email, a failure to send occurs.

Some NonStop Systems use communication hardware which have multiple physical Ethernet ports controlled by a single NonStop process. For example, the TCP/IP CLIM has several physical Ethernet ports and all communication occurs through only one NonStop process, for example $ZTC0. By default, a TCP/IP listen on a TCP/IP stack process (again for example $ZTC0) would allow incoming connection on every physical Ethernet port in the CLIM and an outbound connection would use a path chosen by the system, which may not be desirable. The specification of a particular IP address (or DNS name) would limit and direct all communication to a single physical Ethernet card.

Internally, this keyword causes a socket level call to bind() after a socket is opened to force an association with the IP address specified.

```
Example:
                                          == SNTP message
                                          ==   source IP address
    SNTP-BIND-ADDR 10.21.36.124
```

SNTP-OFFSET

(server version 5.18 or later)

(Default = zero)

SNTP-OFFSET <+/-hmm>

Specifies an offset added to the NTP server reported time.

The value specified may either be positive (the default) or negative and is encoded as hours and 2 digits for minutes. For example, 1 hour is represented as 100. 2 hours and 15 minutes is represented as 215.

This keyword provides for the situation when the original NTP server time value, which is generally GMT, is not desired but requires an adjustment. Literally, MOMI adds the value specified here to the time reported by the NTP server and the result is used in the time adjustment calculations.

It is recommended that if this feature is required, MOMI should be placed into preview mode, via commenting out the keyword SNTP-ALLOW-ADJUSTMENT, to confirm the value specified here is correct. Time adjustment information is reported in the MOMI Log when operating in preview mode.

Example:

```
                                == add 1 hour to NTP
SNTP-OFFSET 100                 == reported time


                                ==  subtract 2 hours and
                                == 15 minutes from
SNTP-OFFSET -215                == NTP reported time
```

SNTP-SERVER-ADDR

(server version 4.16 or later)

(Default = none)

SNTP-SERVER-ADDR <DNS-name>| <IP-address>

Defines the DNS name or IP address of the NTP server (i.e. network time source). The NTP server is queried by MOMI to determine if the System time requires adjustment. Note that the NonStop System must have the resolver active if a DNS-Name is used.

MOMI uses the SNTP protocol to access the NTP server.

Fault tolerance is provided if a DNS name is used and resolves into multiple IP addresses. MOMI connects to the first IP address returned but if communication fails over a certain threshold, a connect is attempted on the second address and so on.

This keyword is required in order to have MOMI adjust the System time.

```
Examples:
                                                == NTP server name
    SNTP-SERVER-ADDR us.pool.ntp.org

                                                == NTP server address
    SNTP-SERVER-ADDR 10.10.1.1
```

SNTP-TCPIP-NAME

(server version 4.16 or later)

(Default = first TCPIP-LISTEN)

SNTP-TCPIP-NAME <tcpip-process-name>

Defines the TCP/IP process name that MOMI uses to access the NTP server. The NTP server provides a network time source and MOMI uses this to determine if the System time should be adjusted.

By default, the first TCP/IP process defined to MOMI for establishing client connections is used.

This keyword would generally be used in the situation where the NTP server is isolated or not accessible from the same subnet MOMI PC Clients connect.

```
Example:
                                    == specify SNTP TCP/IP stack
    SNTP-TCPIP-NAME $ZSAM1
```

SNTP-UPDATE-INTERVAL

(server version 4.16 or later)

(Default = 1)

SNTP-UPDATE-INTERVAL <hours>

Specifies the frequency that MOMI accesses the NTP server to obtain a network time. The NTP server provides a network time source and MOMI uses this to determine if the System time should be adjusted.

The frequency of access has a minimum value of 1 hour and a maximum value of 744 hours (which is approximately 1 month). A value less than the minimum is set to the minimum. A value over the maximum is set to the maximum.

MOMI commences the time update process approximately on an LCT multiple of the value specified here. What this means, for example, if a value of 24 (hours) is specified MOMI starts its time update processing at 12:00 am. If the value of 2 (hours) is specified, time update processing occurs at 12:00 am. 2:00 am, 4:00 am and so on. Processing may take several minutes so any adjustment would normally occur a few minutes after the multiple.

Example:

```
                                        == check system time
                                        ==  every 2 days
    SNTP-UPDATE-INTERVAL 48
```

SPOOLER-DEFINE

(server version 6.17d or later)

SPOOLER-DEFINE $sname [PLUS]

Specifies a SPOOLER supervisor process name that MOMI should include in SPOOLER related screens.

Usage of this parameter, if present, overrides and disables the CONFMOMI keywords FNAME-SPOOLER-n and FNAME-SPOOLER-PLUS- n. Use one line for each SPOOLER Supervisor. The optional keyword PLUS identifies to MOMI a SPOOLER Plus. A total of up to 100 process names may be specified.

It is necessary to identify a SPOOLER Plus to MOMI as this optional product that requires certain changes in the access method versus the standard SPOOLER.

Examples:

```
                                  == Define Spooler Supervisor $SPLS
    SPOOLER-DEFINE $SPLS

                                 == Define Spooler Supervisor $SPL1
    SPOOLER-DEFINE $SPL1

                                 == Define Spooler Plus $PLUS1
    SPOOLER-DEFINE $PLUS1 PLUS
```

SPOOLER-IO-DELAY

(server version 4.05 or later)

(Default = 5)

SPOOLER-IO-DELAY <.01 seconds>

Specifies the amount of time, in an implied .01 seconds, of delay in-between individual I/Os used to gather information about Spooler objects.

This delay reduces the CPU in gathering information by spreading the I/Os out over a longer period of time.

```
Example:

    SPOOLER-IO-DELAY 200          == delay 2 seconds between
```

SPOOLER-UPDATE-INTERVAL

(server version 4.05 or later)

(Default = 60)

SPOOLER-UPDATE-INTERVAL <seconds>

Specifies the frequency in seconds in which Spooler objects are updated for Alarm purposes while the object is in a non-alarm (i.e. normal state).

The higher the value, the longer in-between updates. Lower values result in more frequent updates, but result in higher CPU usage.

The time spent gathering data also lengthens the update frequency.

The parameter SPOOLER-IO-DELAY determines the amount of time in-between each I/O during an update interval.

Example:

```
    SPOOLER-UPDATE-INTERVAL 30      == update every 30 seconds
```

SPOOLER-UPDATE-INTERVAL-ERR

(server version 4.05 or later)

(Default = 30)

SPOOLER-UPDATE-INTERVAL-ERR <seconds>

Specifies the frequency in seconds in which Spooler objects are updated for Alarm purposes after an error condition has been detected.

By default, MOMI uses a faster update interval during error conditions such as when a line or path down are present. This allows MOMI to report updates more quickly. After the error condition is corrected and detected by MOMI, the normal update interval is used.

See the parameter SPOOLER-UPDATE-INTERNAL for additional considerations.

```
Example:

    SPOOLER-UPDATE-INTERVAL-ERR 15  == update every 15 seconds
```

SYSTEM-DESCRIPTION

(Default = none)

SYSTEM-DESCRIPTION <string>

An informational field that specifies the nature or characteristics of the NonStop System on which MOMI is running.

The <string> is limited in length to the first 30 bytes.

The string may be quoted to include an embedded space.

Examples:

```
    SYSTEM-DESCRIPTION  Production
    SYSTEM-DESCRIPTION  'Communication Front End'
    SYSTEM-DESCRIPTION  "Production 1"
```

TAPE-UPDATE-INTERVAL

(server version 6.18d or later)

(Default = 30)

TAPE-UPDATE-INTERVAL <seconds>

Specifies the frequency that MOMI accesses the TAPE subsystem to support alarms.

The frequency of access has a minimum value of 10 seconds and a maximum value of 1 hour. A value less than the minimum is set to the minimum. A value over the maximum is set to the maximum.

If no TAPE related alarms are defined, the subsystem is only queried once but will resume if alarms are defined.

Example:

```
                                        == Query subsystem every
                                        ==  every 2 minutes
    TAPE-UPDATE-INTERVAL 120
```

TCPIP-LISTEN

(server version 5.15 or later)

(At least 1 is REQUIRED)

TCPIP-LISTEN <tcpip-process-name> * | <IP-address> | <DNS-name> <port>

Specifies the TCP/IP process name, optional IP address / DNS name and TCP/IP port to accept connections from the MOMI PC Client.

A total of up to 20 specifications may be made. A minimum of one is required.

Avoid usage of port 2000 (see Network Information Troubleshooting) for additional information.

Some NonStop Systems use communication hardware which have multiple physical Ethernet ports controlled by a single NonStop process. For example, the TCP/IP CLIM has several physical Ethernet ports and all communication occurs through only one NonStop process, example $ZTC0. By default, a TCP/IP listen on a TCP/IP stack process (again for example $ZTC0) would allow incoming connection on every physical Ethernet port in the CLIM and an outbound connection would use a path chosen by the system, which may not be desirable. The specification of a particular IP address (or DNS name) would limit and direct all communication to a single physical Ethernet card.

If a DNS name is specified, it is resolved once at MOMI start up time.

The first occurrence of this keyword in the CONFMOMI file sets defaults for the following keywords:

- ALARM-SMTP-TCPIP-NAME
- DNS-TCPIP-NAME
- SNTP-TCPIP-NAME

Examples:

```
                                == TCP/IP stack $ZSAM1
                                ==  all IP addresses
                                ==  port 2010
    TCPIP-LISTEN  $ZSAM1  *  2010

                                == TCP/IP stack $ZTC1
                                ==  address 192.168.1.10
                                ==  port 3000
    TCPIP-LISTEN  $ZTC1  192.168.1.10  3000
```

```
                                == TCP/IP stack $ZTC1
                                ==  address by DNS
                                ==  port 3019
TCPIP-LISTEN  $ZTC1  tdm.mycompany.com  3010


                                == TCP/IP stack $ZTCP0
                                ==  address 10.11.54.21
                                ==  port 2040
TCPIP-LISTEN  $ZTCP0  10.11.54.21  2040
```

TCPIP-THREADS

(default & maximum changed 6.11)

(Default = 250)

TCPIP-THREADS <value16>

Specifies the maximum number of TCP/IP threads or concurrent Client connections supported. The maximum allowed is 450.

A value smaller or larger than allowed is set to the respective minimum or maximum permitted.

Resources are allocated dynamically for TCP/IP connections, so using this keyword will not generally 'save' anything nor is it generally required.

The number of connections current, high water and the maximum available may be viewed in the PC Client.

Example:

```
TCPIP-THREADS 350                    == TCP/IP threads 350
```

TCPIP-WRITE-BYTES-SEC

(Default = varies-see below)

TCPIP-WRITE-BYTES-SEC <value64>

Specifies the maximum number of TCP/IP bytes per second (Bps) that may be sent from the MOMI server to all PC Clients. When data traffic exceeds the specified limit, I/Os are limited (i.e. throttled) to honor the limit.

The default value is determined based on the processor type which $MOMI is running and may vary in releases.

This parameter has the indirect effect of limiting the amount of CPU usage consumed by $MOMI in handling TCP/IP traffic to and from the MOMI PC Clients.

A value of zero disables limitation.

The following table lists the default value:

Processor	Default Bps
NS5000T	800,000
NS1000, NS1200, NS3000AC	900,000
NS2300	1,000,000
NS2400	1,500,000
NS14000, NS14200	2,000,000
NS16000, NS16200	3,000,000
NB50000C, NB54000C	4,000,000
NB56000C	4,500,000
Later NS-Series processors	2,000,000
TNS/X processors	5,000,000
Unknown processors	6,000,000

Examples:

```
    TCPIP-WRITE-BYTES-SEC 0          == remove limitation

                                     == limit to 350,000
                                     == bytes/sec
```

```
TCPIP-WRITE-BYTES-SEC 350000
```

Advanced Keywords

ALARM-BREAKPOINTS-MAXIMUM

(server version 5.17 or later)

(Default = 50)

ALARM-BREAKPOINTS-MAXIMUM <value>

Defines the maximum number of breakpoints available within each processor of the system.

A breakpoint is used by a programmer to stop the execution of a process at a particular location. The operating system maintains a table of breakpoints currently defined. If the processor breakpoint table becomes full, additional breakpoints are not allowed.

The current number of breakpoints defined within a processor is available to MOMI via a system call. However, the maximum number available within a processor is not available to MOMI. This keyword defines a maximum that MOMI uses for computational purposes (used by the alarm **CPU / Breakpoints**).

Current systems do not have a practical limit on the number of breakpoints that may be set. The older S-Series systems had a fixed limit defined within SYSGEN.

```
Example:
                                    == Set MAXIMUM breakpoints
                                    ==  to 75

    ALARM-BREAKPOINTS-MAXIMUM 75
```

ALARM-EMS-EVENT-NBR

(server version 5.00 or later)

(Default = zero)

ALARM-EMS-EVENT-NBR <value>

Defines the EMS event number for messages logged by MOMI that are a result of an alarm.

Value has a recommended range of 20000 through 32767.

MOMI does generally create EMS messages in a 'formatted' manner, but logs its information as text with an event number of zero. However, to allow for any possible future enhancement in this area the recommended range of values should be followed.

```
Example:
                                        == Set event number
                                        ==  for EMS alarm messages

    ALARM-EMS-EVENT-NBR 22000
```

ALARM-SMTP-PORT

(server version 5.50 or later)

default = 25

ALARM-SMTP-PORT <value>

Overrides the default port for output email (i.e. SMTP).

Example:

```
                                    == Set SMTP to 2026

    ALARM-SMTP-PORT  2026
```

CLIENT-ACC

(Default = TRUE)

CLIENT-ACC TRUE | FALSE

Determines if PC Client acceleration is activated. This optimization allows the server to construct multiple responses to a Client a request (i.e. block up responses into fewer transmissions).

By default, this optimization is turned on. The keyword is provided for internal testing.

Example:

```
Client-acc false                    == acceleration off
```

CLIENT-INITIAL-MSG

(client/server version 5.24 or later-default changed 6.00)

(Default = TRUE)

CLIENT-INITIAL-MSG TRUE | FALSE

Determines if the PC Client (or screen saver) is required to use extra level of initial protocol to successfully communication with the MOMI server on the NonStop System.

The keyword is provided for internal testing purposes only.

The MOMI Server on the NonStop System opens and listens a TCP/IP port as specified by TCPIP-LISTEN. Certain security testing programs may open the port and attempt communication. This feature was introduced to help insure that these testing programs were more quickly disconnected.

A side effect of this feature is that no MOMI PC Client prior to version 5.08 or MOMI screen saver prior to version 4.13 (issued in 5.24 release package) can communicate with the server.

Example:

```
Client-initial-msg  false          == initial message off
```

COLLECT-ALL-PROCESSES

(server version 4.17 or later)

(Default = FALSE)

COLLECT-ALL-PROCESSES TRUE | FALSE

Directs MOMI to collect all processes on the system regardless of its creation time

By default, processes created in the future are not collected.

MOMI, in order to prevent possible process collection hangs, in its normal data collection INTERVAL does not monitor processes with a process creation time greater than the start of a particular collection interval. In the situation where numerous processes are continually being created (i.e. 100's or 1000's), trying to pick up newly created processes could hang a MOMI CPU collector. The default operation of MOMI, in order to work-around this possible situation, is to not pick-up processes created in the "future".

Systems used for test purposes that may adjust or play with (rather dramatically) the system time can use the keyword to alter the normal data collection operation. A side effect of using this keyword, is that HUNG or STOP may be see on the Main Overview screen for CPUs if the situations described above occur.

Example:

```
                                  == collect all processes
                                  ==  regardless of process
                                  ==  creation time
    COLLECT-ALL-PROCESSES true
```

CPU-COLLECTOR-NOREPORT-TIMEOUT

(changed server version 6.11 or later)

(default = the larger of 60 or (3 * INTERVAL))

CPU-COLLECTOR-NOREPORT-TIMEOUT <seconds>

Specifies the maximum amount of time (in seconds) that MOMI waits for a CPU collector process to report.

A CPU collector process that does not report in a timely manner is stopped and its data cleared. The client initially reports the CPU as HUNG (or STOP). The collector is automatically restarted about 1 minute later.

It is not recommended to increase this value as data reported by the client for that affected processor may be old and incorrect. However, certain system configurations may not be able to provide MOMI the CPU cycles needed to operate timely and this keyword allows for a greater timeout value.

The minimum value allowed is the default. Any value less than the default is set to the default. No maximum is imposed.

This value is not 'exact' and the effective value will usually be higher.

Example:

```
                                == set timeout to 30 seconds
    CPU-COLLECTOR-NOREPORT-TIMEOUT 30
```

CPU-NOT-PRESENT

(server version 5.35 or later)

(Default = none)

CPU-NOT-PRESENT <cpu-nbr>

This parameter is intended for special circumstances where a nonstandard configuration is required.

Specifies if MOMI should consider a processor AS not present.

By default, and under normal systems configurations, processors in a NonStop System are contiguously populated starting with CPU 0 and sequentially going up to a maximum of CPU 15 (16 total for the system). However, certain configurations may have a hole or gap and processors are not physically populate sequentially. MOMI at start-up considers any 'missing' processor as down between (and inclusive) CPU 0 and the highest numerical value determined to be running (i.e. the CPU's O/S is loaded and responding).

A processor defined with this keyword is omitted or not initialized within MOMI and in turn does not appear on MOMI screens. After MOMI is started, if a processor is physically loaded (i.e. O/S responding) MOMI will see the CPU UP system message and report its activity regardless of this keyword setting.

Note that at MOMI start-up if a processor is defined with this keyword and is physically present (i.e. O/S is responding) it is not reported.

Multiple processors s are defined by multiple uses of this keyword in the CONFMOMI file (only one cpu-nbr per line may be specified).

Example:

```
                                        == System missing CPU 4 & 5
    CPU-NOT-PRESENT 4
    CPU-NOT-PRESENT 5
```

CPU-LIMIT

(Default = no limit)

CPU-LIMIT <percent-limit>

Imposes a maximum percentage of CPU busy each MOMI process is allowed to consume.

This parameter sets a default global limit for all MOMI processes except where another CPU-LIMIT-* keyword is defined or has a default. In other words, this parameter is a 'catch all', or starting point, for all MOMI programs including the main $MOMI.

By default, no limitation is imposed. It is generally not advisable or required to use this parameter.

Example:

```
CPU-LIMIT 22                    == limit CPU usage to 22%
```

CPU-LIMIT-ALARM-DETECT

(Default = 5)

CPU-LIMIT-ALARM-DETECT <percent-limit>

Specifies the percentage of CPU busy a MOMI server is allowed to consume in alarm detection activity.

The value may range from 0 to 99. A value of 0 disables CPU limit.

Alarm detection is the processing in stand alone servers associated determining if a threshold has been reached. For example, this could be scanning through disk files and obtaining attributes, requesting information from the SPOOLER, performing PING, etc...

Example:

```
CPU-LIMIT-ALARM-DETECT 15      == limit CPU usage to 15%
```

CPU-LIMIT-DB-ALARM

(server version 6.17h or later)

(Default = 10)

CPU-LIMIT-DB-ALARM <percent-limit>

Specifies the percentage of CPU busy a MOMI server is allowed to consume in Database Alarm activity.

The value may range from 0 to 99. A value of 0 disables CPU limit.

Database Alarm activity is the process of maintaining an in memory copy of alarm definitions, active alarms, providing copies to MOMI PC Clients, and issuing notification of active alarms.

Example:

```
CPU-LIMIT-DB-ALARM 15      == limit CPU usage to 15%
```

CPU-LIMIT-EMS

(Default = none)

CPU-LIMIT-EMS <percent-limit>

Specifies the percentage of CPU busy a MOMI server is allowed to consume in reading from EMS.

The value may range from 0 to 99. A value of 0 disables CPU limit.

Reading from EMS is in response to a MOMI PC Client request to obtain information from either $0, other collectors or an EMS log file. By limiting or slowing down the MOMI process requesting from EMSDIST, indirectly the CPU usage consumed by both may be reduced.

Example:

```
CPU-LIMIT-EMS 16              == limit CPU usage to 16%
```

CPU-LIMIT-HSTxxDB-CONSOL

(Default = 4)

CPU-LIMIT-HSTxxDB-CONSOL <percent-limit>

Specifies the percentage of CPU busy the MOMI history consolidate and delete processing is allowed to consume.

The value may range from 0 to 99. A value of 0 disables CPU limit.

History consolidation is the process of moving data from one history file to another. This process takes data from many history periods and adds them together to a more consolidated or 'less detailed' view. Delete processing is removing old data from the history file after consolidation has completed.

Example:

```
CPU-LIMIT-HSTXXDB-CONSOL 8      == limit CPU usage to 8%
```

DEBUG1

(Default=not enabled)

DEBUG1

This parameter is intended for internal use only.

If present, activates special debug code used to determine if server is sending invalid data to Client. Will cause MOMI to ABEND if the condition is found.

DEBUG2

(Default=not enabled)

DEBUG2

This parameter is intended for internal use only.

If present, activates special debug code used to determine if MOMI is trying to terminate due to an unexpected error condition. Will cause MOMI to ABEND if the condition is found.

DEBUG3

(Default=not enabled)

DEBUG3

This parameter is intended for internal use only.

If present, activates special debug code used to confirm if data compression is working correctly. Data is compressed, decompressed and compared to the original. Will cause MOMI to ABEND if a mismatch is found.

DEVICE-INFO-SENDMSG-TIMEOUT

(server version 6.11 or later)

(default = 10)

DEVICE-INFO-SENDMSG-TIMEOUT <seconds>

Specifies the maximum amount of time (in seconds) that MOMI waits for a response to a DEVICE_INFO query for subtype 30 processes.

A process that does not respond within the specified time has the device type set to an error indication.

A subtype 30 process has the ability to simulate other processes and the executable has a subtype 30 attribute. This designation causes the O/S to send a message to the process requesting its device type/subtype. However, some processes do not respond to this query.

It is not recommended to increase the value greatly as this could slow down other processing within MOMI.

Example:

```
DEVICE-INFO-SENDMSG-TIMEOUT 12              == set timeout to 12
seconds
```

DISABLE-LIST-LOCKS

(server version 5.03 or later)

(Default = false)

DISABLE-LIST-LOCKS TRUE | FALSE

Determines if MOMI is allowed to obtain information about the presence and status of locks on files.

True means this entity is not allowed to operate. False allows this entity to operate, which is the default.

The keyword is not normally used but is provided in the event that the listing of locks is not recommended for the current release of the operating system.

Example:

```
DISABLE-LIST-LOCKS TRUE             == prevent list locks
```

DISABLE-LOGON-COMMANDS

(Default = false)

DISABLE-LOGON-COMMANDS TRUE | FALSE

Specifies if MOMI allows commands to function that require LOGON with a User ID and password.

Certain sensitive commands require the user to LOGON with a GUARDIAN and password. This setting can be used to disable LOGON commands and prevent such commands from being executed.

True causes all sensitive commands requiring LOGON to be disabled. False allows the user to enter a User ID and password and if authentication is accepted, execute sensitive commands.

Example:

```
DISABLE-LOGON-COMMANDS TRUE           == disable LOGON
```

DISABLE-MEAS-COUNTER-WRITE

(server version 5.05 or later)

(Default = see below)

DISABLE-MEAS-COUNTER-WRITE TRUE | FALSE

Determines if MOMI disables writing of MEASURE counters to any data file where counters are accessed online (i.e. via LISTACTIVE).

True means MEASURE counters are not written to disk. False means MEASURE counters are written to disk.

The default value for this keyword varies based on the operating system release. See the table below:

Operating System release	Default value
G Series	False
H Series prior to H06.14	False
H Series H06.14 or later	True
J Series J06.06 or later	True
any other release	True

A master measurement is started shortly after MOMI startup collects information for CPU's, Processes, TMF, Expand, etc... This measurement is left running the entire time MOMI operates. The measure has no collection interval as data is retrieved via the LISTACTIVE function. However, MEASURE will write records such as process start / stop to the file until it fills. At that time each processor in the CPU will log an error 45 to EMS. This error, however, does not impact MOMI's ability to access the data via the LISTACTIVE function. It simply means that data is no longer written to the MEASURE data file. This feature of MEASURE to disable recording of counter records to disk effectively prevents the error 45 situation.

Setting this value to True, when MEASURE support is not available, will cause the initial measurement started by MOMI to fail. This would be seen in the MOMI PC Client on the screen Overview as the error indication MEAS.

Example:

```
                                    == counters written to disk
    DISABLE-MEAS-COUNTER-WRITE FALSE
```

DISABLE-MEAS-OSSCPU

(server version 4.09 or later)

(Default = false)

DISABLE-MEAS-OSSCPU TRUE | FALSE

Determines if MOMI is allowed to report on the OSS CPU entity via MEASURE.

True means this entity is not allowed to operate. False allows this entity to operate.

Certain versions of the operating system may have issues with this MEASURE entity so this keyword provides a disable capability.

Example:

```
DISABLE-MEAS-OSSCPU TRUE                == disable OSS CPU
```

DISABLE-MEAS-OSSNS

(server version 4.09 or later)

(Default = false)

DISABLE-MEAS-OSSNS TRUE | FALSE

Determines if MOMI is allowed to report on the OSS Name Server (NS) entity via MEASURE.

True means this entity is not allowed to operate. False allows this entity to operate.

Certain versions of the operating system may have issues with this MEASURE entity so this keyword provides a disable capability.

Example:

```
DISABLE-MEAS-OSSNS TRUE          == disable OSS Name Server
```

DISABLE-MEAS-RM-DISC

(server version 5.58 or later)

(Default = false)

DISABLE-MEAS-RM-DISC TRUE | FALSE

Determines if MOMI is allowed to gather the DISC entity of MEASURE via the system procedure call MEAS_READACTIVE_MANY_.

True means this operation is not allowed to be performed. False allows this operation to be performed.

Certain versions of the operating system may have issues with this MEASURE operation so this keyword provides a disable capability.

Example:

```
                                                == disable reading DISC via
                                                ==  MEAS_READACTIVE_MANY_
    DISABLE-MEAS-RM-DISC TRUE
```

DISABLE-TAPE-REPORT

(server version 5.18c or later)

(Default = false)

DISABLE-TAPE-REPORT TRUE | FALSE

Determines if MOMI is allowed to query TAPE to obtain information such as outstanding TAPE mounts.

True means this entity is not queried. False, the default, allows this entity to be queried.

The keyword would be used if for some reason the subsystem should not be queried.

Example:

```
                          == prevent automatic TAPE reporting
    DISABLE-TAPE-REPORT TRUE
```

DISABLE-TMF-REPORT

(server version 4.20 or later)

(Default = false)

DISABLE-TMF-REPORT TRUE | FALSE

Determines if MOMI is allowed to query TMF to obtain information such as the overall transactions per second and the state of TMF.

True means this entity is not queried. False, the default, allows this entity to be queried.

The keyword would be used if for some reason the subsystem should not be queried.

Example:

```
                          == prevent automatic TMF reporting
    DISABLE-TMF-REPORT TRUE
```

DISK-B-MB-NOT-CONFIGURED

(server version 6.18d or later)

(Default = false)

DISK-B-MB-NOT-CONFIGURED TRUE | FALSE

Specifies if all disk Backup and Mirror Backup should be reported by MOMI as not configured meaning they are not considered present. This situation may be desirable within a virtual Nonstop environment.

Example:

```
                                    == Disk Backup and Mirror Backup
                                    ==  is not reported
    DISK-B-MB-NOT-CONFIGURED TRUE
```

DISK-COMPRESSION-TYPE

(default changed server version 6.00)

(Default = 3)

DISK-COMPRESSION-TYPE 0 | 2 | 3

Specifies or disables the compression algorithm used for storing data on disk, primarily history.

The following selections are available:

0 - None

2 - Type 2

3 - Type 3 (Default)

None causes data to be saved without compression and consumes the most disk space.

Type 2 is present in MOMI Server versions 3.xx, 4.xx and 5.xx.

Type 3 is available in MOMI server version 5.00 or later. This type compresses longer strings of repeat data than Type 2 and generally provides a higher level of compression.

A newer version of the MOMI server can read data compressed from older server versions. However, older versions of the MOMI server cannot read data created by a newer server version if the compression type is not known.

MOMI's ability to read existing data is not affected by a change in this value as the compression selection is stored in each data record.

The default is recommended.

Example:

```
                                    == use old disk compression
    DISK-COMPRESSION-TYPE 2
```

ENCRYPTION-ALGORITHM

(server version 5.06 or later - updated 5.57)

(Default = 2 / S-Series or older 5)

ENCRYPTION-ALGORITHM 0 .. 5

Specifies the type of data encryption used between the MOMI PC Client and the MOMI Server in the TCP/IP data flow.

The following settings are available:

0 - None

1 - FAS

2 - AES

3 - FAS No Port

4 - AES No Port

5 - FAS 2

0 - None means that normal MOMI traffic is not encrypted. This provides the highest level of performance with the lowest CPU cost.

1 - FAS uses a proprietary algorithm with a 256 bit asymmetric key to provide a "Fast and Simple" encryption of the data. A fairly high level of performance is obtained with a modest CPU cost.

2 - AES uses the Advance Encryption Standard industry algorithm with a 256 bit key.

3 - FAS No Port uses a proprietary algorithm with a 256 bit asymmetric key to provide a "Fast and Simple" encryption of the data. A fairly high level of performance is obtained with a modest CPU cost. This algorithm may be automatically selected if the TCP/IP dynamic port varies between the MOMI PC Client and MOMI Server.

4 - AES No Port uses the Advance Encryption Standard industry algorithm with a 256 bit key. This algorithm may be automatically selected if the TCP/IP dynamic port varies between the MOMI PC Client and MOMI Server.

5 - FAS 2 uses a proprietary algorithm with a 256 bit asymmetric key to provide a "Fast and Simple" encryption of the data. A fairly high level of performance is obtained with a modest CPU cost. This implementation is improved over the original 1 - FAS algorithm.

Regardless of the chosen setting note the following:

- Logon information is always (and always has been) encrypted.
- Header and control information in the data flow may not be encrypted.
- The encryption level may be automatically downgraded to comply with US export controls.

Generally speaking, MOMI takes advantage of the multiple processing running within its environment and pushes encryption processing down-to the level where a request is actually serviced. This push helps to limit the amount of encryption overhead that occurs in the higher priority processes of a MOMI subsystem. See Process Priority for additional information on how MOMI divides its workload.

```
Example:
                                                == Encrypt type AES
    ENCRYPTION-ALGORITHM 2
```

EXPAND-IO-DELAY

(Default = 5)

EXPAND-IO-DELAY <.01 seconds>

Specifies the amount of time, in an implied .01 seconds, of delay in-between individual I/Os used to gather information about Expand objects.

This delay reduces the CPU in gathering information by spreading the I/Os out over a longer period of time

```
Example:
                                    == delay 2 seconds between
                                    ==  I/Os
    EXPAND-IO-DELAY 200
```

G0612-OR-LATER

(Default = automatic)

G0612-OR-LATER TRUE | FALSE

This parameter is intended for internal use only.

Specifies if MOMI uses certain procedural values to obtain information available in the G06.12 or later version of the operating system.

True instructs MOMI that the O/S specific features are available. False instructs MOMI that the O/S specific features are not available.

By default, the proper setting is automatically determined. Fields reported in the client that are not available are reported as zero or spaces (as appropriate). Usage of this parameter will override the automatic determination.

G0626-OR-LATER

(Default = automatic)

G0626-OR-LATER TRUE | FALSE

This parameter is intended for internal use only.

Specifies if MOMI uses certain procedural values to obtain information available in the G06.26 or later version of the operating system.

True instructs MOMI that the O/S specific features are available. False instructs MOMI that the O/S specific features are not available.

By default, the proper setting is automatically determined. Fields reported in the client that are not available are reported as zero or spaces (as appropriate). Usage of this parameter will override the automatic determination.

H0604-OR-LATER

(server version 4.07 or later)

(Default = automatic)

H0604-OR-LATER TRUE | FALSE

This parameter is intended for internal use only.

Specifies if MOMI uses certain procedural values to obtain information available in the H06.04 or later version of the operating system.

True instructs MOMI that the O/S specific features are available. False instructs MOMI that the O/S specific features are not available.

By default, the proper setting is automatically determined. Fields reported in the client that are not available are reported as zero or spaces (as appropriate). Usage of this parameter will override the automatic determination.

H0614-OR-LATER

(server version 5.05 or later)

(Default = automatic)

H0614-OR-LATER TRUE | FALSE

This parameter is intended for internal use only.

Specifies if MOMI uses certain procedural values to obtain information available in the H06.14 or later version of the operating system.

True instructs MOMI that the O/S specific features are available. False instructs MOMI that the O/S specific features are not available.

By default, the proper setting is automatically determined. Fields reported in the client that are not available are reported as zero or spaces (as appropriate). Usage of this parameter will override the automatic determination.

H0624-J0613-OR-LATER

(server version 5.32 or later)

(Default = automatic)

H0624-J0613-OR-LATER TRUE | FALSE

This parameter is intended for internal use only.

Specifies if MOMI uses certain procedural values to obtain information available in the H06.24/J06.13 or later version of the operating system.

True instructs MOMI that the O/S specific features are available. False instructs MOMI that the O/S specific features are not available.

By default, the proper setting is automatically determined. Fields reported in the client that are not available are reported as zero or spaces (as appropriate). Usage of this parameter will override the automatic determination.

H0625-J0614-OR-LATER

(server version 5.32 or later)

(Default = automatic)

H0625-J0614-OR-LATER TRUE | FALSE

This parameter is intended for internal use only.

Specifies if MOMI uses certain procedural values to obtain information available in the H06.25/J06.14 or later version of the operating system.

True instructs MOMI that the O/S specific features are available. False instructs MOMI that the O/S specific features are not available.

By default, the proper setting is automatically determined. Fields reported in the client that are not available are reported as zero or spaces (as appropriate). Usage of this parameter will override the automatic determination.

H0627-J0616-OR-LATER

(server version 5.32 or later)

(Default = automatic)

H0627-J0616-OR-LATER TRUE | FALSE

This parameter is intended for internal use only.

Specifies if MOMI uses certain procedural values to obtain information available in the H06.27/J06.16 or later version of the operating system.

True instructs MOMI that the O/S specific features are available. False instructs MOMI that the O/S specific features are not available.

By default, the proper setting is automatically determined. Fields reported in the client that are not available are reported as zero or spaces (as appropriate). Usage of this parameter will override the automatic determination.

H0629-J0618-L1708-LATER

(server version 6.00 or later)

(Default = automatic)

H0629-J0618-L1708-LATER TRUE | FALSE

This parameter is intended for internal use only.

Specifies if MOMI uses certain procedural values to obtain information available in the H06.29/J06.18/L17.08 or later version of the operating system.

True instructs MOMI that the O/S specific features are available. False instructs MOMI that the O/S specific features are not available.

By default, the proper setting is automatically determined. Fields reported in the client that are not available are reported as zero or spaces (as appropriate). Usage of this parameter will override the automatic determination.

HIGHEST-RUNNING-CPU

(Default = automatic)

HIGHEST-RUNNING-CPU <cpu>

Specifies the numerically highest CPU number that should be running on the system.

When MOMI is started, by default it uses the current highest 'up' CPU as the number of processors to 'expect' as running on the system.

This keyword over-rides the default computation. If the value specified is less than the highest 'up' CPU it is not used. Any time a new processor is added to the system MOMI bumps its own internal value and will expect the processor in the system configuration.

The purpose of this keyword is to insure that a minimum number of processors exists on the system. It is possible that if MOMI is started while a CPU is down, no alarm indication would result.

Example:

```
                                        == 6 processor system
    HIGHEST-RUNNING-CPU 5
```

L2509-OR-LATER

(server version 6.19 or later)

(Default = automatic)

L2509-OR-LATER TRUE | FALSE

This keyword is intended for internal use only. Original keyword KLT is also accepted.

Specifies if MOMI uses certain procedural values to obtain information available in the L25.09 or later version of the operating system.

True instructs MOMI that the O/S specific features are available. False instructs MOMI that the O/S specific features are not available.

By default, the proper setting is automatically determined. Fields reported in the client that are not available are reported as zero or spaces (as appropriate). Usage of this keyword will override the automatic determination.

Generally, this keyword activates MOMI gathering and reporting of Nonstop KLT (Kernel Level Threading) related information.

LOGON-PSTATE

(server version 6.17 or later)

(Default = TRUE)

LOGON-PSTATE TRUE | FALSE

Determines if the MOMI requires a user to logon in order to execute PSTATE/PSTATE2 commands.

Setting this keyword to FALSE means that the User ID used to start $MOMI is used to execute this command. By default, the command is executed under the authority of the user logon.

The keyword is provided for environments where it is sufficient or desirable to use MOMI's Client Access feature instead of Guardian / SafeGuard security.

PSTATE/PSTATE2, as of T0705L01^AAT 12APR21, runs under enhanced security as follows:

- Super.group can see all targets.
- Group.manager can see all targets on its group.
- A non-super userid can only see its targets.

Example:

```
                                        == PSTATE/PSTATE2 operates
under
    LOGON-PSTATE false                  ==  the User ID of $MOMI
```

LOGON-SPOOLER

(server version 6.17f or later)

(Default = FALSE)

LOGON-SPOOLER TRUE | FALSE

Determines if the MOMI requires a user to logon access Spooler related functions. This keyword overrides the setting of keyword LOGON-SPOOLER-VIEW-JOB.

TRUE means that ALL Spooler related command are executed under the authority of the user logon.

FALSE means that the User ID $MOMI is started under is used execute non-sensistive Spooler related commands such as listing Spooler supervisors, collectors jobs, devices, etc... Viewing the contents of a Spooler job is considered sensitive.

The keyword is provided for environments where Spooler processes are secured by Safeguard process ACLs.

Example:

```
                                        == logon not required
                                         == to view spooler job
    logon-spooler true
```

LOGON-SPOOLER-VIEW-JOB

(server version 5.24 or later)

(Default = TRUE)

LOGON-SPOOLER-VIEW-JOB TRUE | FALSE

Determines if the MOMI requires a user to logon to view the contents of a Spooler job.

Setting this keyword to FALSE means that the User ID $MOMI is started under is used to access the contents of a spooler job. By default, the command is executed under the authority of the user logon.

The keyword is provided for environments where it is sufficient to use MOMI's Client Access feature instead of Guardian / SafeGuard security.

Setting this keyword to FALSE has the following effects:

- Viewing of a Spooler job occurs under the logon security used to start $MOMI
- If $MOMI is started under Super.Group (or better) ID the contents any Spooler job may be displayed
- By default, the NOT LOGGED ON user can display the contents of any Spooler jobs

Example:

```
                                        == logon not required
logon-spooler-view-job false            == to view spooler job
```

MAX-COUNT-CONTEXT-ALLOWED

(server version 5.05 or later)

(default = 100)

MAX-COUNT-CONTEXT-ALLOWED <value32>

This parameter is intended for internal use only.

Specifies the maximum number of context control blocks a single client may request from a single MOMI server.

The server will not set a value lower than an internal minimum value.

The MOMI server usually creates context to service individual Client requests. As a protection, the maximum number client context control blocks are limited. This prevents, in a worse case situation, the appearance of a MOMI server looping or consuming a great deal of CPU resources when the client disconnects and internal clean up occurs.

Example:

```
MAX-COUNT-CONTEXT-ALLOWED 60        == limit context to 60
```

MEAS-SQLSTMT-WILD-CARD

(Default = Space)

MEAS-SQLSTMT-WILD-CARD <"character">

Defines the character used as a wild-card for the field run^unit on the MEASURE entity SQL Statements.

MOMI always retrieves all SQL Statements in a process when requested. However, at some point the MEASURE interface changed and an asterisk ("*") no longer functioned as a wild-card. Instead, a space was required.

This keyword allows the definition of the character used for this particular MEASURE entity field.

Examples:

```
                            == set SQL Statements wild-card
                            ==  to an asterisk
    MEAS-SQLSTMT-WILD-CARD  "*"
```

OSS01-API-VERSION

(server version 4.20 or later)

(Default = automatically determined)

OSS01-API-VERSION 1 | 2

Determines the API set used by MOMI when accessing OSS.

Version 1 uses 32-bit OSS file system calls. Version 2 uses 64-bit OSS file system calls.

Later versions of the Operating System provides for OSS files larger than 2 Gigabytes in size. However, this requires that an extended set of API calls be used that generally return 64-bit instead of 32-bit values. The 64-bit OSS file system calls may be used on 64 and 32-bit files. However, 32-bit OSS file system calls cannot be used on 64-bit files.

Usage of this keyword is NOT recommended as the default is automatically determined by MOMI based on the current operating system version. The keyword is provided primarily for internal testing.

Example:

```
OSS01-API-VERSION 2    == use 64-bit OSS file system calls
```

PRIORITY-COLLECTOR-MST

(Default = Main MOMI - 1)

PRIORITY-COLLECTOR-MST <value-1-199>

Specifies the priority for the Collector Master. This process configures the master measurement used by all MOMI CPU collectors and performs other coordination activities.

The default value is recommended.

Example:

```
PRIORITY-COLLECTOR-MST 165      == master at priority 165
```

PROCESS-ENTITY-EXCLUDE

(server version 5.21 or later)

(Default = None)

PROCESS-ENTITY-EXCLUDE <file-name>

Defines object file names that are excluded from the MEASURE entity Process.

File-name is a MEASURE Guardian format object file name. Wild-cards are supported but only as a whole volume, subvolume or file portion of the name. Up to 10 entries are supported.

This capability is provided to as a work-around where certain object files must be excluded from the process entity measurements made by MOMI. Any exclusion specified causes a Process Entity and SQLPROC Entity exclusion from 1) the master measurement started when MOMI is started, and 2) the measurement started on the client screen Processes / Process Entity.

Note that processes with object files specified with this keyword will no longer be reported on MOMI screens where their information is obtained from MEASURE. However, the process may still be reported where information is obtained from Guardian resulting in discrepancies between client screens.

Examples:

```
                                    == exclude these objects
                                    == from process entity
    PROCESS-ENTITY-EXCLUDE $SYSTEM.*.mxosrvr
    PROCESS-ENTITY-EXCLUDE $OSS.*.*
    PROCESS-ENTITY-EXCLUDE $data1.test.fredobj
```

SNTP-OFFSET

(server version 5.18 or later)

(Default = zero)

SNTP-OFFSET <+/-hmm>

Specifies an offset added to the NTP server reported time.

The value specified may either be positive (the default) or negative and is encoded as hours and 2 digits for minutes. For example, 1 hour is represented as 100. 2 hours and 15 minutes is represented as 215.

This keyword provides for the situation when the original NTP server time value, which is generally GMT, is not desired but requires an adjustment. Literally, MOMI adds the value specified here to the time reported by the NTP server and the result is used in the time adjustment calculations.

It is recommended that if this feature is required, MOMI should be placed into preview mode, via commenting out the keyword SNTP-ALLOW-ADJUSTMENT, to confirm the value specified here is correct. Time adjustment information is reported in the MOMI Log when operating in preview mode.

Example:

```
                                == add 1 hour to NTP
SNTP-OFFSET 100                 == reported time


                                ==  subtract 2 hours and
                                == 15 minutes from
SNTP-OFFSET -215                == NTP reported time
```

SNTP-SET-IF-BEHIND

(server version 5.00 or later)

(Default = FALSE)

SNTP-SET-IF-BEHIND TRUE | FALSE

Directs MOMI to perform a SET of the system time if a correction is needed that would move the system time forward.

This setting provides a capability to 'force' an adjustment of the system time. Moving the system time forward is not considered 'dangerous', where moving the system time backward (on an active system) is not desirable, especially to TMF.

By default, MOMI will only adjust or request a gradual correction to the system time.

Customers generally should NOT use this capability.

Example:

```
                                    == Allow MOMI to set
                                    == the system time forward
    SNTP-SET-IF-BEHIND  TRUE
```

SNTP-VERBOSE

(server version 4.16 or later)

(Default = not verbose)

SNTP-VERBOSE

Directs MOMI to write additional information about SNTP operations into the MOMI Log.

This information would be used to help diagnose a problem with SNTP processing.

By default, MOMI will not generate this additional information.

Example:

```
                                        == Generate additional
                                        == information about
                                        == SNTP processing
    SNTP-VERBOSE
```

SPI-IO-TIMEOUT

(server version 4.06 or later)

(default = 5)

SPI-IO-TIMEOUT <seconds>

Specifies the maximum amount of time (in seconds) that MOMI waits for a single SPI I/O to complete. SPI is a programmatic interface used to obtain information about various subsystems such as Expand, TMF, Tape, etc... .

A server that does not respond within the specified time generates an error 40 (Timeout).

It is not recommended to increase the value greatly as this could slow down other processing within MOMI.

Example:

```
SPI-IO-TIMEOUT 12              == set timeout to 12 seconds
```

SPOOLER-IO-DELAY

(server version 4.05 or later)

(Default = 5)

SPOOLER-IO-DELAY <.01 seconds>

Specifies the amount of time, in an implied .01 seconds, of delay in-between individual I/Os used to gather information about Spooler objects.

This delay reduces the CPU in gathering information by spreading the I/Os out over a longer period of time.

```
Example:

    SPOOLER-IO-DELAY 200          == delay 2 seconds between
```

TCPIP-TCP-NODELAY

(default changed and keyword added in server version 6.08a or later)

(Default = false)

TCPIP-TCP-NODELAY TRUE | FALSE

Specifies if MOMI should issue a TCP socket option (TCP_NODELAY) that causes any data sent from the server to send immediately to the client. Sockets, by default, impose a slight delay in the hope that additional data is following and the overall number of transmissions may be reduced.

True instructs MOMI to issue the TCP_NODELAY socket option. False instructs MOMI to not issue the socket option.

MOMI clients normally receive a fairly steady stream of data and default socket delays are usually in milliseconds thus eliminating the need for immediate socket data sends.

Example:

```
                              == TCP/IP data sends
                              ==  immediately
    TCPIP-TCP-NODELAY true
```

TCPIP-THREADS

(default & maximum changed 6.11)

(Default = 250)

TCPIP-THREADS <value16>

Specifies the maximum number of TCP/IP threads or concurrent Client connections supported. The maximum allowed is 450.

A value smaller or larger than allowed is set to the respective minimum or maximum permitted.

Resources are allocated dynamically for TCP/IP connections, so using this keyword will not generally 'save' anything nor is it generally required.

The number of connections current, high water and the maximum available may be viewed in the PC Client.

Example:

```
TCPIP-THREADS 350                    == TCP/IP threads 350
```

TCPIP-TIMEOUT

(server version 4.19 or later)

(default = 600)

TCPIP-TIMEOUT <seconds>

Specifies the maximum amount of time (in seconds) that MOMI waits for any data to arrive from the MOMI PC Client. This value acts as a "sanity check" to insure that the connection between the client and server has viable with periodic activity and not hung.

If no data arrives within the specified time, a timeout occurs and the connection is closed.

This value also indirectly determines the maximum amount of time that the MOMI PC Client may be globally paused (i.e. all activity suspended) before the connection is closed.

The minimum value of 60 (1 minute). Any value less than the minimum is set to the minimum.

Example:

```
TCPIP-TIMEOUT 60              == set timeout to 1 minute
```

WAITED-FILENAME-FIND

(introduced in server version 5.28 - default changed in server version 6.06 or later)

(Default = FALSE)

WAITED-FILENAME-FIND TRUE | FALSE

Directs MOMI to perform a waited request for file name information instead of the no-wait operation.

This keyword was introduced due to bug introduced in the initial releases of H06.24 and J06.13 with the tag parameter. This resulted in MOMI appearing to not return look-ups of certain process and file name requests.

By default, MOMI uses no-wait operation.

Example:

```
                                    == Directs MOMI to use
                                    ==  waited request for file
                                    ==  name information
    WAITED-FILENAME-FIND  TRUE
```

Obsolete Keywords

NOCAIL

NOCAIL (default = *enabled*)

Directs the MOMI PC Client to not enable the terminal emulator.

The option is provided if the CAIL terminal emulator is not desired or cannot be loaded.

This option is similar to the MOMI PC Client run-time parameter, but is server based.

Example:

```
NOCAIL                                                    == disable
emulator
```

PRIORITY-DB-HSTxxDB-DEL

(Default = 80)

PRIORITY-DB-HSTxxDB-DEL <value-1-199>

NOTE: This parameter is not currently used, but is kept for compatibility. See PRIORITY-DB-HSTxxDB-CON.

Specifies the process priority of a MOMI server for database operations relating to the HSTxxDB files. These files are used for storing performance information.

Specifically, this keyword controls the priority for deleting old data or cleanup of the files and as such, this is an ongoing but fairly consistent operation and should occur at a fairly low priority.

TCPIP-INFO-n

TCPIP-INFO-n <tcpip-process-name> <port>

Specifies the TCP/IP process name and TCP/IP port to accept incoming connections from the MOMI PC Client. Up to 20 specifications may be made.

The 'n' is a value portion of the keyword is present to make the parameter name unique.

This keyword has been superseded by TCPIP-LISTEN.

```
Examples:

    TCPIP-INFO-1  $ZSAM1 2010   == TCP/IP $ZSAM1 2010

    TCPIP-INFO-2  $ZTC1 3000    == TCP/IP $ZTC1 port 3000

    TCPIP-INFO-3  $ZTCP0 2040   == TCP/IP $ZTCP0 port 2040
```

History

Enabling History

MOMI has the ability to store many of its statistics for later review. This feature is enabled in the CONFMOMI file. By defining history file (s), MOMI will automatically create and begin recording predetermined statistical information. If a file fills up, either for lack of disk space or maximum extents, the oldest data is automatically deleted to make room for newest data.

Up to four history files may be created. These are referred to as HST01DB -> HST04DB. History is initially written to HST01DB and then consolidated into the other files at different intervals. This gives the User the ability to maintain high resolution history for a short period of time and lower resolution history for a longer period of time.

The line to define each history file name (HSTnnDB) is the only line required. Other history attributes all have default values.

History is configured on the NonStop as follows:

1. Determine the disk volume(s) where the history files are to reside.

2. Edit CONFMOMI and un-comment out or add the following lines (HST01DB is required, others optional):

```
                == MOMI History Files
HST01DB $vol.subvol.HST01DB
HST02DB $vol.subvol.HST02DB
HST03DB $vol.subvol.HST03DB
HST04DB $vol.subvol.HST04DB
```

3. Optional - add keyword(s) : HSTnnDB-DELETE-TIME

4. Optional - add keyword(s) : HSTnnDB-HISTORY-DUMP

5. Start (or restart) MOMI via the TACL obey file OBYMOMI.

History files are created automatically. If a file fills up, its oldest data in the file is automatically deleted to make room for newest data.

Resizing History Files

(changed server version 6.17f or later)

MOMI automatically creates history files when enabled. The following table lists the initial size of these files:

File	Primary Extent	Secondary Extents	Default Maximum Extents	Default Size Gigabytes
HST01DB	80000	80000	60	9.8
HST02DB	70000	70000	50	7.2
HST03DB	60000	60000	40	4.9
HST04DB	50000	50000	30	3.1

Two methods are available to re-size a history file.

Alter maximum extents

The utility FUP may be used, while $MOMI is running, to alter the maximum extents of the history files.

To increase the amount of history disk space allowed, increase MAXEXTENTS of the history file(s).

To decrease the amount of history disk space allowed, reduce the MAXEXTENTS of the history file (s). The file system will not allow a reduction in the maximum extents below the End of File (EOF).

If the default amount of space consumed by history files (as shown above) is too large for the NonStop System, start MOMI, wait a few seconds to allow the files to automatically create, and then stop MOMI. In FUP "set like" a history file, purge the old file, adjust the MAXEXTENTS and/or primary/secondary extent sizes, then create a new file. MOMI must be able to allocate at least one extent on every file defined or an error will result.

See the screen Server_Info for information on how to alter the history files MAXEXTENTS from within the MOMI PC Client.

Load data into a larger file

Once a history file reaches the maximum number of extents allowed by the system, usually 978, storing additional history requires creation of a new file. Optionally, load data from the old file into the new file (if the old data is desired and you have time to wait for the load to complete).

Some math is required to determine the size of the new file, but a good starting point is its current size as reported by: FUP INFO <file-name>,detail. The value to the right of EOF is the number of bytes the file occupies. Take this number and perform the following math:

(eof / 2048) / 32 = ext-size *(use this result below)*

This formula takes the file EOF in bytes, divides by the number of bytes per disk page (2048), and then divides that by the number of extents for the file to initially occupy (32). This last number is not much more than a good guess but if you are loading data from a file with many hundreds of extents the disk may not have one very large chunk of contiguous free space. So, instead of one computing for one large chunk of free disk space this formula tries for at least 32 smaller ones.

The following steps describe how to create a new file and then optionally load it with data. The values in green are dependent on which history file is being modified and your configuration needs. Substitute the HST01DB name (used in this example) with the desired file on the NonStop System:

1. Logon to a TACL prompt on the NonStop System with sufficient authority to stop & restart $MOMI plus access its files.

2. Stop $MOMI. Wait until all its processes have automatically stopped.

3. Volume to the location of existing MOMI files, starting with FUP execute the following sequence of commands -

FUP

```
RENAME HST01DB,HST01DB1

SET LIKE HST01DB1
SET FORMAT 2
SET EXT (ext-size,ext-size)
SET MAXEXTENTS 64                          (double current size)
CREATE HST01DB

LOAD HST01DB1,HST01DB,SORTED               (optional)

EXIT
```

4. Restart MOMI. Check the EMS log to insure the new file was accepted (an error message is generated if a problem occurs).

5. At a later time, delete the old file.

Considerations:

a) absolutely, positively, and without a doubt insure that other applications on the system will not be impacted by the disk space consumed. Use the DSAP utility to confirm the availability of the extent size chosen. Remember that total free disk space does not indicate the availability of the extent size chosen.

b) depending on the file size and system performance, the LOAD command may take 45 minutes or more. One way to speed the process is to perform the LOAD from one disk volume to another disk volume then after the load completes FUP DUP the file back (if necessary). This may be faster than a LOAD when both the input & output are on the same volume.

Leave all other attributes of a MOMI file 'as is'. MOMI will not use the file if an expected attribute is changed. Check the EMS log immediately after restarting MOMI to insure that the converted history file is accepted (MOMI writes a message to the EMS log (i.e. $0) if the history file is not accepted). If this new file is not accepted fallback to the original.

History Considerations

The collection of history may require consideration before activation on smaller systems or when disk space is limited.

History files (HSTxxDB) are simple Enscribe keyed sequence files with a single primary key. No alternate keys are used and the files are not TMF audited. If necessary, history files may be purged. Stop MOMI, purge the desired files (or purge data) and restart MOMI. MOMI automatically creates files if needed. No special recovery procedures are required.

The activation of history primarily 1) will consume disk space, and 2) increase the I/O load to the disk volume(s) where the history files reside. Systems may notice a slight increase in CPU utilization. However, disk space and I/O load is really the primary considerations.

The default amount of disk space consumed by MOMI history files may be found in Resizing History Files. Care should be given to not impact disk space required by other applications.

Volumes such as $SYSTEM and $AUDIT are not recommend for MOMI history files as performance of these volumes should be reserved for the overall system.

MOMI initially writes history to the HST01DB file. History is consolidated from HST01DB into HST02DB, from HST02DB into HST03DB and finally from HST03DB into HST04DB. The consolidation process generally a) reads all needed records from one file, b) writes all needed records into the second file and c) goes back and deletes oldest eligible data from the first file. Spreading MOMI history files out across multiple disk does not really 'help' to provide any performance improvements at least to MOMI, however it may help other applications also residing on the same volume if it is heavily loaded.

History is written to HST01DB every few minutes by default. All MOMI history is compressed and consolidated to help avoid unnecessary I/O. Consolidation into the other files occur at their history dump interval (default history dump intervals are found in HSTnnDB-HISTORY-DUMP).

History Questions and Answers

Where do I start with the history parameters?

The defaults are recommended. Remove comments or add the lines to the CONFMOMI file as referenced elsewhere in this document.

What if my file is not large enough? Will MOMI generate an error 45 (file is full) message?

If MOMI needs to write data to the history file and space is not available, some of the oldest data in the file is automatically deleted to make room for new. MOMI does not generate error 45 (file is full) messages nor error 43 (no disk space) for history files. This action occurs regardless of any other settings pertaining to how long data is retained.

Does dumping data more frequently cause MOMI to work harder?

Directly no, but indirectly yes. The information MOMI dumps to the history file has already been collected. Activation of history is the process of dumping this data to disk, allowing the PC Client to read the history data and maintain or delete old data from the history file. Therefore, no additional CPU cost is directly imposed in the collection process, just in what you do with the data after it is collected.

At what process priority will the history be dumped to disk? Will it be high like MOMI (by default) is?

MOMI uses a server process, by default running at priority 95, to write history data to disk. It does not write at the priority of $MOMI. MOMI uses three different servers to manage the history file. One server to write data, a second server to read data for PC Client requests, and a third server to delete or cleanup old data. Refer to the documentation for the following CONFMOMI keywords for additional information:

PRIORITY-DB-HST01DB	`priority write`
PRIORITY-DB-HSTxxDB-R	`priority read`
PRIORITY-DB-HSTxxDB-CON	`priority delete/cleanup`

What will happen if the server process that MOMI uses to write data to disk is bogged down or otherwise not able to get its data written out to disk?

MOMI will initially queue in memory the data waiting to be written to disk. This queue will continue for a period of time, before new data is discarded. Overall, MOMI will try to protect itself and the System by using more memory and lastly discarding history data if necessary. In order to insure that all history data is collected, the MOMI server used to write data to disk should be given sufficient CPU so it will not normally be bogged down by batch or other long running programs.

I see a high CPU utilization of a MOMI process running at a low priority, and by using MOMI I tracked it down as performing large number of disk reads from a history file. What is going on?

Normally, that is normal. The PC Client has a great deal of flexibility in accessing the history data. The PC Client could be searching for a named process, reading data for display or perhaps performing a history export operation. MOMI protects your System by using a separate, configurable priority process to perform "bulk" history reads. If the default priority is too high for your System, lower the priority by using the keyword PRIORITY-DB-HSTxxDB-R.

I cannot find history records for a process?

To conserve disk space MOMI history records are written when a process 'changes' state or is active. If a process is active, inactive for a period of time, then active, there may be no history records for a portion of the inactive period of time. If you were to view all of the history records for this process, at the start of the inactive period of time there would be at least one record showing zero CPU busy, a gap in real time and then followed by a record showing activity. Activity is generally defined as start/stop, CPU Busy, or other changes in the process state.

Can I read the history files directly?

No. The data format of the history files is not published and the data is compressed. History file data is available to import into your favorite spreadsheet or database via the Export function.

System Quirks

The following links describes various situations or "quirks" present on the NonStop System:

- CPU Up/Down determination
- Expand throughput
- O/S version determination
- SQL/MP - DiskFile Entity
- SQL/MP errors
- Virtual Memory changes via NSKCOM / Virtual memory statistics

CPU Up/Down determination

NonStop Systems are not configured with the expected number of processors. From a system configuration perspective, the operating system is always ready to accept (software wise) a maximum of 16 processors.

Every time MOMI is started it uses the numerically highest **up** processor in the system as the expected number of processors in the system. For example, if the numerically highest **up** processor is 5, which indicates 6 CPUs (0 through 5), and processor 4 crashes it is reported as down.

Additional processors added to the system while MOMI is operating automatically increase the numerically highest **up** processor determination. In the above example, if an additional two processors were added, processors 6 and 7, MOMI now assumes the system is 8 processors numbered 0 through 7.

If processors are removed from the system *on-line*, i.e. without a cold-load, restart MOMI to recompute a new up/down calculation.

If processors are removed from the system, for example CPUs 4 & 5 in a 6 processors system, and it is necessary to operate in that manner for an extended period of time, add to the CONFMOMI file the keyword CPU-NOT-PRESENT and restart MOMI. This keyword directs MOMI to treat the indicated CPUS as not present in the system (i.e. they are not flagged as DOWN). **Be sure** and remove this keyword when the system configuration is restored to normal.

Expand throughput

Byte rates for Expand lines are generally only available for Expand over TCP/IP. Byte rates for other types of lines such as Expand over X.25 are estimated by using the Frame rate times the average bytes per frame. Expand over Fox and Supercluster (Expand over ServerNet which is reported as a NAM connection) do not provide a byte rate.

O/S version determination

Operating Systems versions prior to G06.24 do not provide the exact Operating System version as known by Users (i.e. G06.21) via supported system calls. You can determine the major portion (i.e. G06), but not the minor portion (i.e. .21).

The documented System procedure call PROCESSOR_GETINFOLIST_ has an item code defined to return the minor version number, but it always returns zero on systems prior to G06.24. To work around this limitation MOMI reads an undocumented file $SYSTEM.SYSnn.RLSEID. This file appears to be used by the utility SYSINFO and by TACL to obtain this information.

On Operating Systems G06.24 and later, the documented System procedure PROCESSOR_ GETINFOLIST_ correctly returns the minor version number. The undocumented file $SYSTEM.SYSnn.RLSEID is not accessed.

The TNS/E Integrity System must use the RLSEID file. Hopefully, this may be removed in a future O/S release.

SQL/MP - DiskFile Entity

Secondary partitions on SQL/MP files do not always show up under this measurement as MEASURE is not providing data. Try using the DiskOpen Entity.

SQL/MP errors

The SQL/MP screens may display various SQL error codes, with 4060 as the most common. The appearance of these codes may indicate a discrepancy in the SQL catalog itself. MOMI assumes that the catalog structure and data within the catalog has full integrity.

In the PC Client, on the SQL/MP / Catalog screen, the column Status Message shows an error code if a simple query of that catalog fails. Any Status Message displayed here indicates that MOMI will have problems with SQL/MP reporting. This is the first place to check and perhaps identify the catalog in question.

We suggest consulting with your Database Administrator (DBA) if you have any integrity issues with your SQL/MP catalogs.

Below are some additional steps you can perform to check your catalog integrity. From a TACL prompt:

1) SQLCI GET CATALOG OF SYSTEM;exit

2) Volume to this location.

3) SQLCI SELECT * FROM CATALOGS;exit

4) The first column (CATALOGNAME) in the output consists of all of the catalogs on your system registered to SQL. For each of those entries do a fileinfo of the $vol.subvol.*

5) If a $vol.subvol is listed in the CATALOGS table that does not exist, or does not contain SQL catalog objects, an inconsistency is present. Consult with your DBA and/or HPE to resolve the problem.

If your SQL/MP catalogs do not show a problem, please send us an email with a screen shot and description of the SQL/MP error encountered.

Virtual Memory changes via NSKCOM / Virtual memory statistics

Virtual memory configuration and statistics are not available from either the Guardian Operating System procedures nor MEASURE. To work around this limitation, MOMI starts a copy of NSKCOM and periodically parses its text output to determine the amount of virtual memory available, the highest usage (i.e. high water mark), and the current usage.

Some users have reported that they were unable to alter the virtual memory on their system due to NSKCOM reporting:

> "... ZSYSCFG is locked by another user..."
>
> Note: The file in question is really $SYSTEM.SYSTEM.KMSFLOCK

They corrected this problem by stopping MOMI, making changes via NSKCOM and then restarting MOMI.

MOMI does not make any changes to virtual memory but just uses NSKCOM for 'reading' virtual memory status. However, we believe this situation may occur if a delete of a swapfile is pending after a system cold-load and if the NSKCOM started by MOMI is the first instance of this utility after a cold-load of the system.

The suggested work-around is to start an NSKCOM, under a Super.Group User ID or better, rather early in your system startup files a) after all processors have been loaded, but b) before MOMI is started. The purpose of this NSKCOM is to cleanup any pending swapfile operations after a cold-load. It is possible that the pending operation may take some time if a great deal of cleanup is involved. The recommended command to consider in your system startup file is:

```
comment                         == start NSKCOM, allow any
comment                         == pending cleanup then exit

NSKCOM  EXIT
```

Update ➔ As of MOMI Server version 5.00 and later the NSKCOM used by MOMI is restarted after the initial request of swapfile information. This restart should prevent, or at least automatically correct, a locked configuration file. Note that pending changes to the swapfile may keep the initial instance of NSKCOM 'busy' for a while. This may delay the initial report swapfile information within MOMI. If your site changes the swapfile often, placing an NSKCOM EXIT in your startup files, as described above, is still recommended.

PC Client

Overview

The MOMI PC Client provides the presentation/access interface to MOMI. It is a standard Windows 32-bit program that uses TCP/IP (i.e. Winsock) to communicate with the MOMI server on the NonStop System. No additional communication software such as RSC is required.

The Client is written in Borland Delphi that generates a high performance stand-alone executable and does not require 'registration' within the Windows environment in order to function. The Client makes extensive use of Windows graphics and other resources. As such, a professional version of the Windows operating system (or better) is recommended.

The MOMI PC Client *technically* does not require any Windows installation or setup. You can run the PC Client by double-clicking on the MOMI.EXE executable. This stand-alone executable feature helps support various deployment options. The MOMI PC Client activates additional features when expected subdirectories are present in the location where the executable is present. For example, if the subdirectory \Help\HTML5 (or one the other flavors) is present, local web format help is invoked when pressing the Help button.

When a Client is started for the first time, it creates the MOMI.INI file. This file contains the configuration information for the program. We chose to use an INI file versus registry entries because a User can easily look at or relocate an INI file. Registry entries are a little more involved.

The default location where MOMI stores its INI and other working files:

Windows 10 and later -

C:\Users\<User ID>\AppData\Roaming\BlackWood Systems\MOMI\

Where <User ID> is the name used at logon to Windows. You must enable *Show hidden files, folders and drives* under Windows Folder Options to enable their display.

After the Client creates the MOMI.INI file, it will prompt the User for a TCP/IP address and port. This is an address and port the MOMI server on the NonStop System is 'listening' on. After entering this information the Client will establish a socket connection and begin exchanging data with the MOMI Server. Additional NonStop Systems may be defined in the Client. Each TCP/IP address and port is defined on this same screen.

After the initial connection, the User is taken to the Main Overview screen. This main screen of MOMI provides an overview of the general System health.

The available screens are organized by functional groups. Select the Tab or button of the group to navigate to the desired screen. The mouse will change to a pointing hand when over a jump point. The letters L and/or R will appear to indicate what mouse keys (Left / Right) are available.

Tool Tips or pop-up windows are used extensively throughout the MOMI PC Client to describe fields and actions. Move the mouse over any field or column heading to obtain additional information.

Security is imposed when access to the data or an operation requires user specific authorization. For example, specific security is required to view the contents of a file, but specific security is not required to view the names of files in a directory. Additionally, commands that perform an action such as purging a SPOOLER job or renaming a file require security. The User obtains security by logging on. This is obtained by pressing the LOGON button in the lower right hand corner of the screen. A Guardian User ID or Safeguard Alias and password is entered.

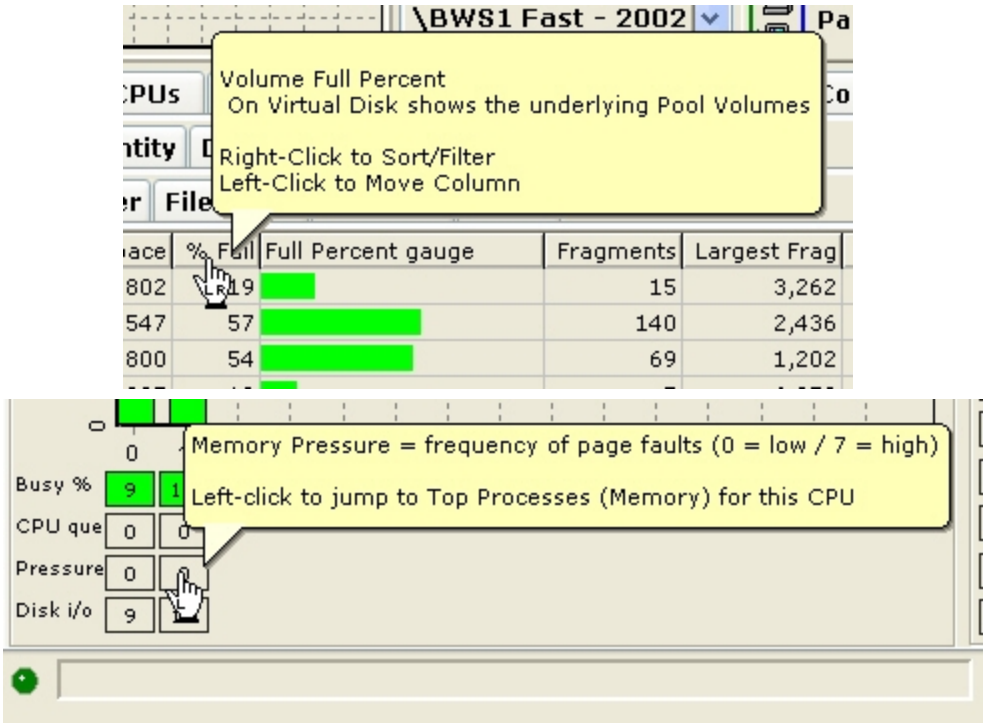
Client Access provides a means to restrict access to particular screens within the MOMI PC Client. Client Access does not replace Guardian/Safeguard security, but is a separate subsystem specific to MOMI.

During the course of accessing various screens and requesting information, additional MOMI servers on the NonStop System are started. This action allows MOMI to prioritize and multi-thread operations. After a period of time (3-4 minutes) unused servers are stopped. Additionally, when the MOMI PC Client is stopped or disconnects from the System, unused servers are also stopped.

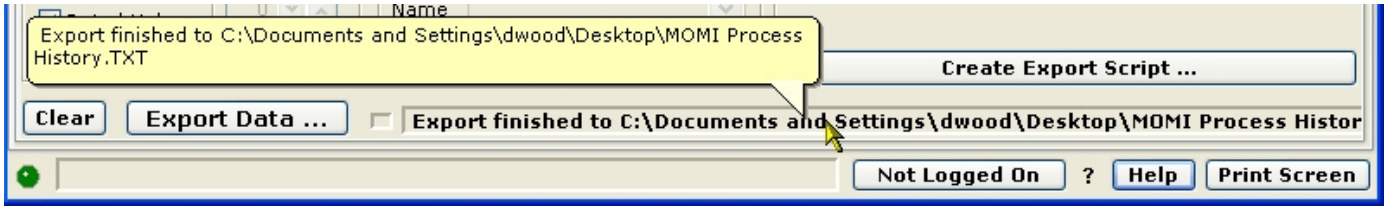
Common screen items

The …mouse pointer displays an indication when a jump point or 'hot spot' is available. It will change to a hand with a L and/or R indicating when a Left and/or Right mouse click is available. Left clicks generally perform the default action and a right click brings up a context sensitive menu with the default action in **bold**. A mouse click in this document refers to the left mouse button.

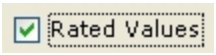
Tool tips, or Hints, are yellow text windows that automatically display when the mouse is moved over certain fields. Tool tips are present throughout the PC MOMI Client. They provide information such as field descriptions, column descriptions and navigational assistance. Tool tips may be disabled or the amount of time before they appear are adjusted on the screen Settings / System Wide. Below are some sample tool tips:



Tool tips are also available to display the entire message in notification areas located at the bottom of most screens. In the example below, the text not visible on screen is shown in the tool tip.



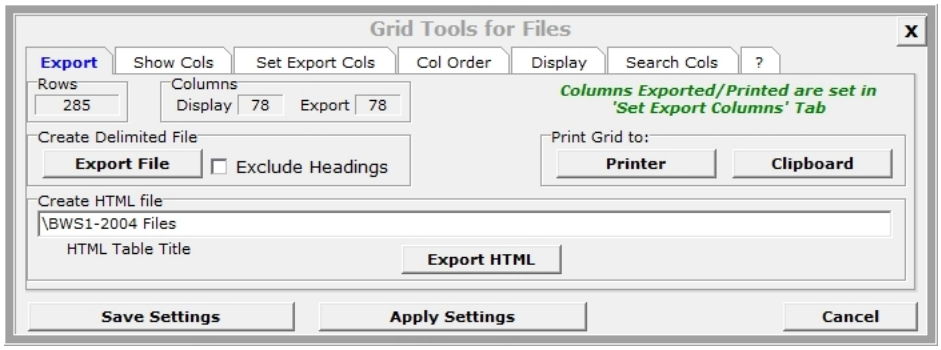
All MOMI screens have common items at the top and bottom of the screen. Place your mouse over items below and click for a description:



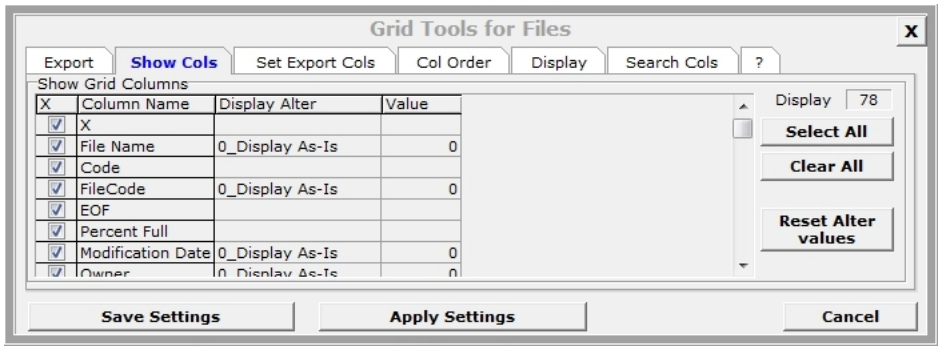
The Rated Values option allows the selection of either values on a per-second basis, or raw values for the measurement window. The measurement window is usually displayed in the status area or on the end of the data line.

The Tool button in the lower left hand corner of most grids presents a pop-up window called Grid Tools and provides options for exporting, printing, hiding columns, setting the columns to export, and saving or restoring the column order. The number of tabs visible may vary from screen-to-screen. Press the **Save Settings** button to have any setting changes apply the next time the MOMI PC Client is started (otherwise the changes only apply during the current instance). Settings within the Grid Tools window affect only the current screen from which the Tool button was pressed.

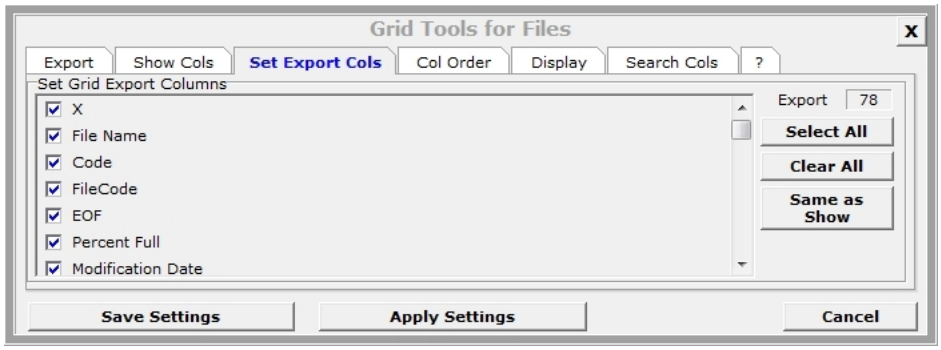
The **Export Grid** tab shows some general stats but primarily provides the ability to export, print and send the data to the clipboard. These functions are generic to the grid.



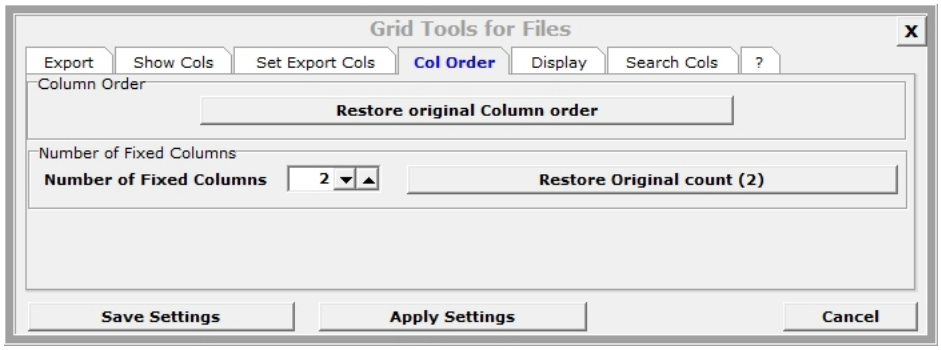
The **Show Columns** tab provides the ability to show or hid columns displayed within the grid. Some fields also provide the option to modify their contents. String fields provide the ability to remove a constant number of leading or trailing characters.



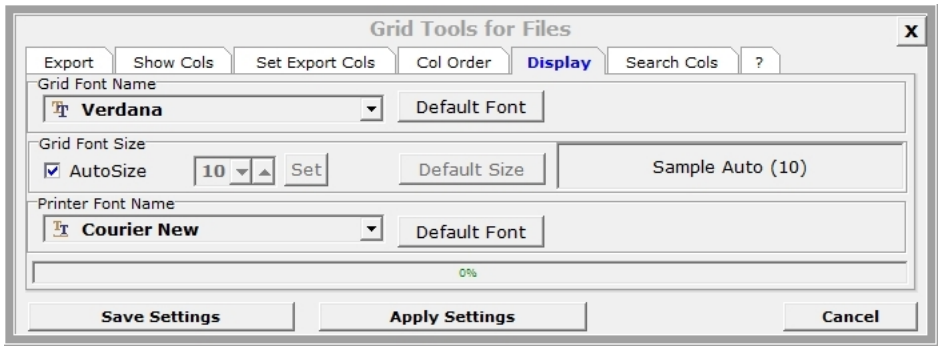
The **Set Export Columns** tab determines which columns are present (i.e. not exported) when the data is exported.



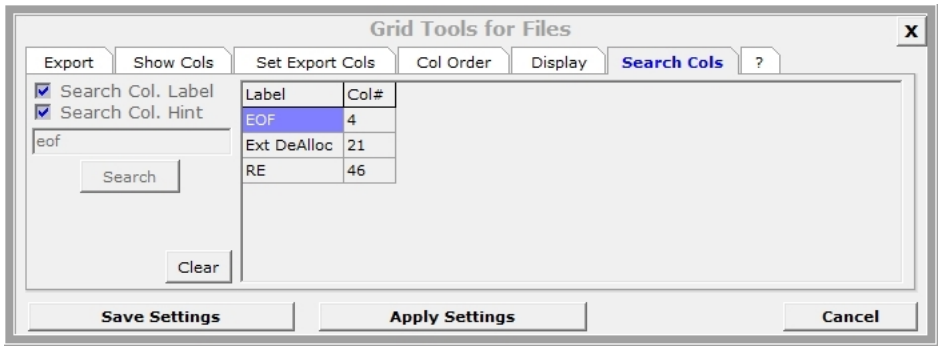
The **Column Order/Fixed** tab determines how many columns from the left hand side of the grid do not move when a grid is scrolled left/right. Typically, the default are field(s) which identify the data on the line such as a process or file name.



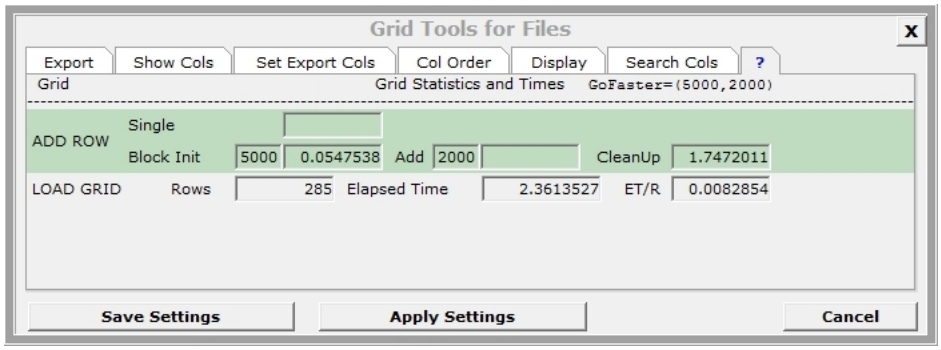
The **Display** tab of the Grid Tools window changes the font for viewing data within the grid and font used for printing data from the grid. The font size for viewing may also be increased or decreased in size by pushing the +/- keys.



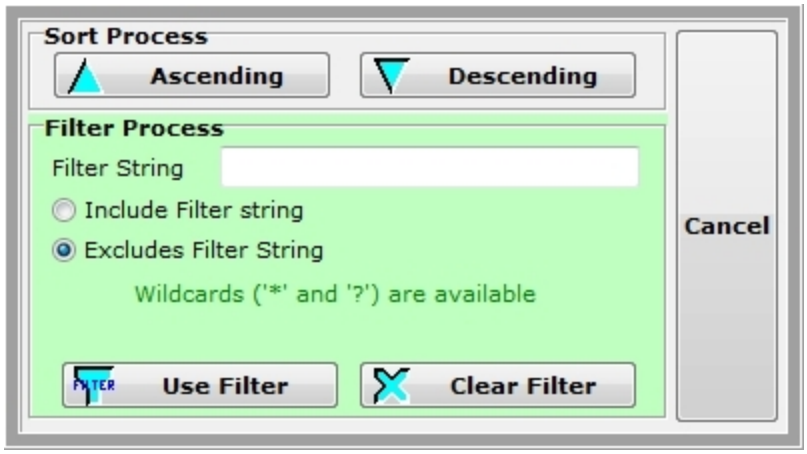
The **Search Cols** tab of the Grid Tools searches for entered text and displays (by column name) where the text was located. Click on the field (for example EOF below) in the **Label** column to cause the grid to highlight and position to it.

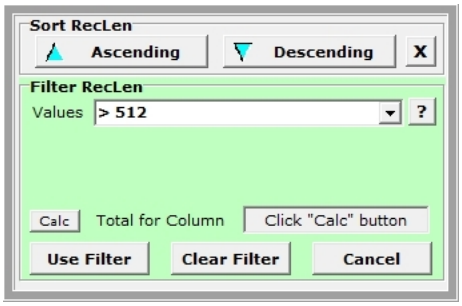


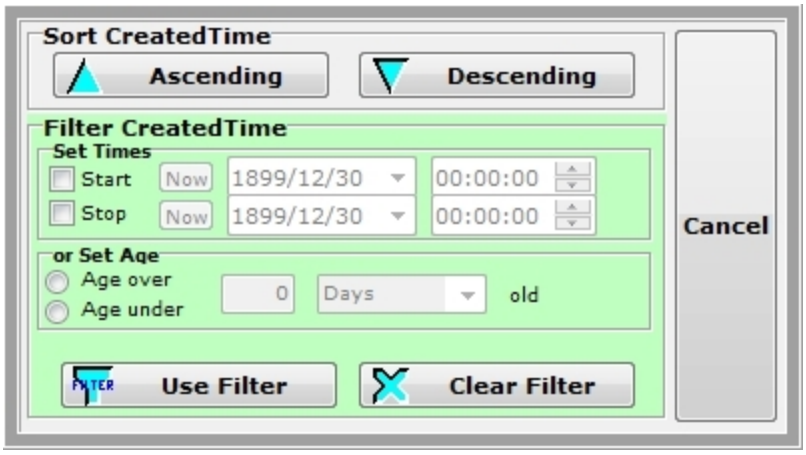
The **Question Mark** tab of the Grid Tools displays internal statistics pertaining to the grid and is used for internal development.



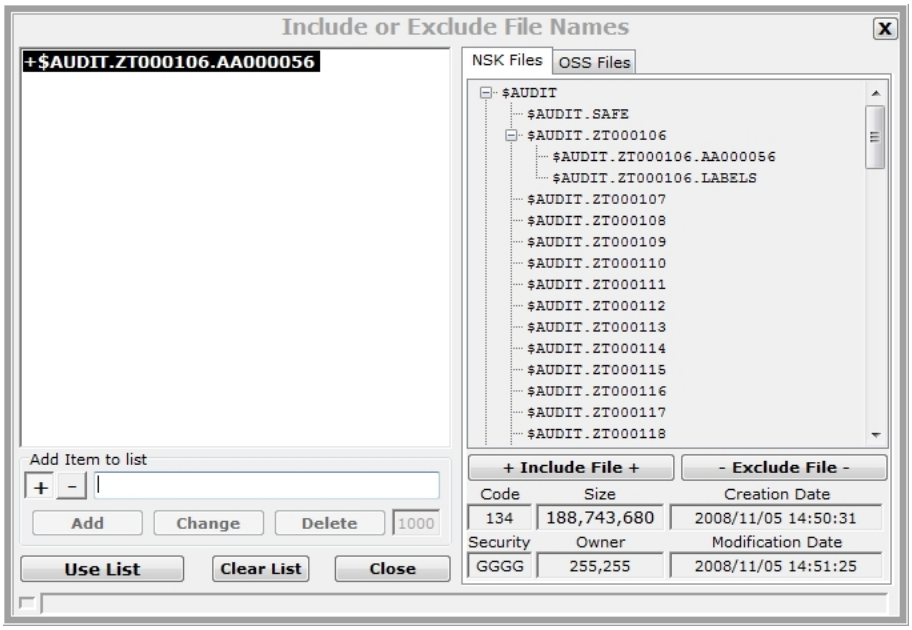
Right-click on a column heading usually provides data sorting (ascending or descending) and a filter specific to the data type. Columns of string data have a single filter item. The entered filter string is exactly matched unless wild card characters * or ? are used. Columns of numeric values may limit display to values that match, or meet a minimum / maximum (click the **?** button for syntax). Additionally, the **Calc** button displays the total sum of all visible rows in the column. Date ranges may be selected and columns with a single character may be selected / deselected.



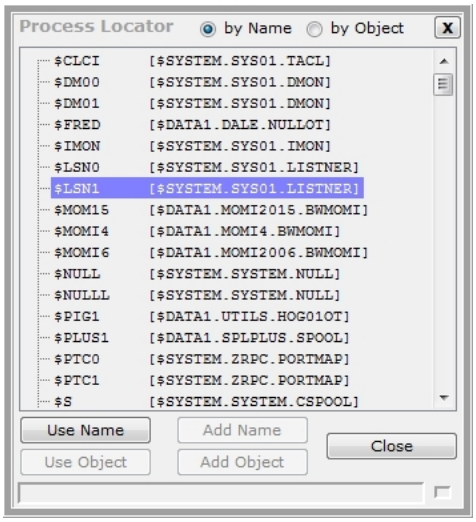




 or  These screen graphic buttons represents a $volume, subvolume and/or file selector option. It can represent an open of a PC file or of a NonStop System based file. Pressing this button for a NonStop System based file brings up a tree view window that allows the user to expand the Volume, Subvolume and down to the file level (where appropriate).



 This screen graphic button opens a tree view window that allows the user select a process name. The types of processes presented are usually limited to the type applicable for the given situation within the MOMI PC Client (for example a listing of all Tape processes) and may be shown by process name or by object file name.



Alarm Overview

MOMI Alarms are the ability to set thresholds and trigger notifications.

Alarms are setup with reusable objects which define a Window of Time to monitor and the notification Action. The notification Action may specify that (any or all of the following) the alarm generate a message on the MOMI PC Client Alarms / Active, generate a message to the MOMI Log, generate a message to the EMS Log, and/or generate an Email to one or more recipients.

An alarm starts by specifying a the general category of Type and Sub-Type which then displays field(s) to define the threshold or state(s) to be monitored. This could be a CPU down or a process not running. Where MOMI regularly gathers data related alarms are automatically checked. Where MOMI must retrieve data to process an alarm the definition will specify a time interval of how often the necessary data is gathered and checked. By default, an Alarm is monitored continuously and displays red on the Alarms / Active screen. A window of time to monitor may be defined specified in 1/2 hour slots specified over 7 days in a week.

Alarm processing may be deferred for a period of time when the MOMI Subsystem is started on the NonStop System by using the CONFMOMI keyword ALARM-SUSPEND-DELAY.

Alarm Security

By default, only the user that started the MOMI server is allowed to configure an Alarm. Additional users may be granted this authority by activating Client Access.

CONFMOMI keywords

One of the available Actions to an Alarm is the generation of Email or Email to a Group. To enable this function, the following CONFMOMI keywords must be defined (or review its default value).

Email requires an external SMTP server (i.e. outgoing email) and the return Email address:

ALARM-SMTP-SERVER-ADDR
ALARM-EMAIL-ADDRESS-FROM

Some SMTP (i.e. outgoing email) servers require the HELO sequence which requires a portion of your email address defined in this keyword:

ALARM-DOMAIN-NAME

If you have CLIMs on your system, you may need to direct which physical Ethernet is used for sending email:

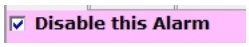
ALARM-SMTP-BIND-ADDR

The TCP/IP process used to communicate with the SMTP server has a default which may be overridden:

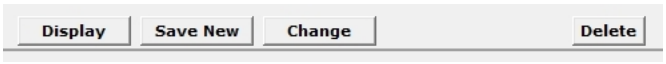
ALARM-SMTP-TCPIP-NAME

Common screen items

Definition of alarm objects usually have several common screen commands:



Disable this <text varies> when checked (push **Change** to update) causes the record to no longer function. Uncheck (and push Change) to re-activate.



Display causes the selected record (or entered text) to appear.

Save New creates a record. If a record with the same name is already present, an error is displayed.

Change updates an existing record. If a record with the same name does not exist, an error is displayed.

Delete removes an existing record.

International Issues

Dates

All Dates in the Client, where the value is output, are in the format of yyyy/mm/dd (year/month/day) except where the date is contained within a text message. An example of this exception is EMS messages.

In certain regional language selections, the Calendar display for Dates displays some invalid characters, although the calendar still works. On the Config screen the User may select a roll up/down style instead of a calendar to prevent this display irregularity.

Export data

Data exported contains a leading HardSpace (ASCII value 160) for fields that contain numeric information intended to be treated as a string field. An example is the CPU,PIN field. The default character may have adverse implications when using a multi-byte character set. The leading character used for these fields may be altered on the Settings / System Wide page.

Specifying run-time parameters

Various parameters may be given to the MOMI PC Client at start up. These parameters control certain aspects of the program operation. Multiple parameters are separated by spaces and may be specified in any order.

Syntax formats may enclose fields in the characters < >, however these characters are not used. For example:

inipath="<location>"

is used as follows at a command line:

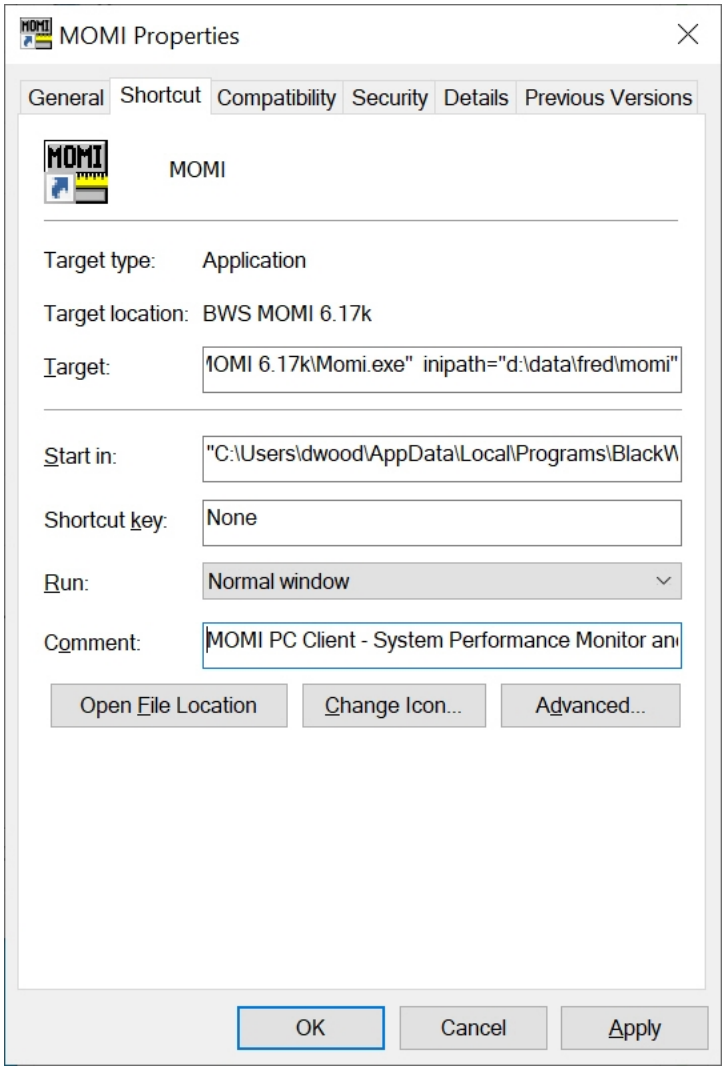
```
MOMI.EXE inipath="d:\data\fred\momi"
```

Examples:

```
MOMI.EXE script-history=ProcessScript     executes script ProcessScript

MOMI.EXE nocail                           don't load the CAIL emulator

                                          sets INI to d:\data\fred\momi
MOMI.EXE inipath=d:\data\fred\momi
```

As viewed from the Target field of a Windows shortcut:

"C:\Users\dwood\AppData\Local\Programs\BlackWood Systems\BWS MOMI 6.17k\Momi.exe" inipath="d:\data\fred\momi"


MOMI Properties
General
Shortcut
Compatibility
Security
Details
Previous Versions
MOMI
Target type:
Application
Target location:
BWS MOMI 6.17k
Target:
1OMI 6.17k\Momi.exe" inipath="d:\data\fred\momi"
Start in:
"C:\Users\dwood\AppData\Local\Programs\BlackW
Shortcut key:
None
Run:
Normal window
Comment:
MOMI PC Client - System Performance Monitor an
Open File Location
Change Icon...
Advanced...
OK
Cancel
Apply

Run-Time Parameters

CLIENTID

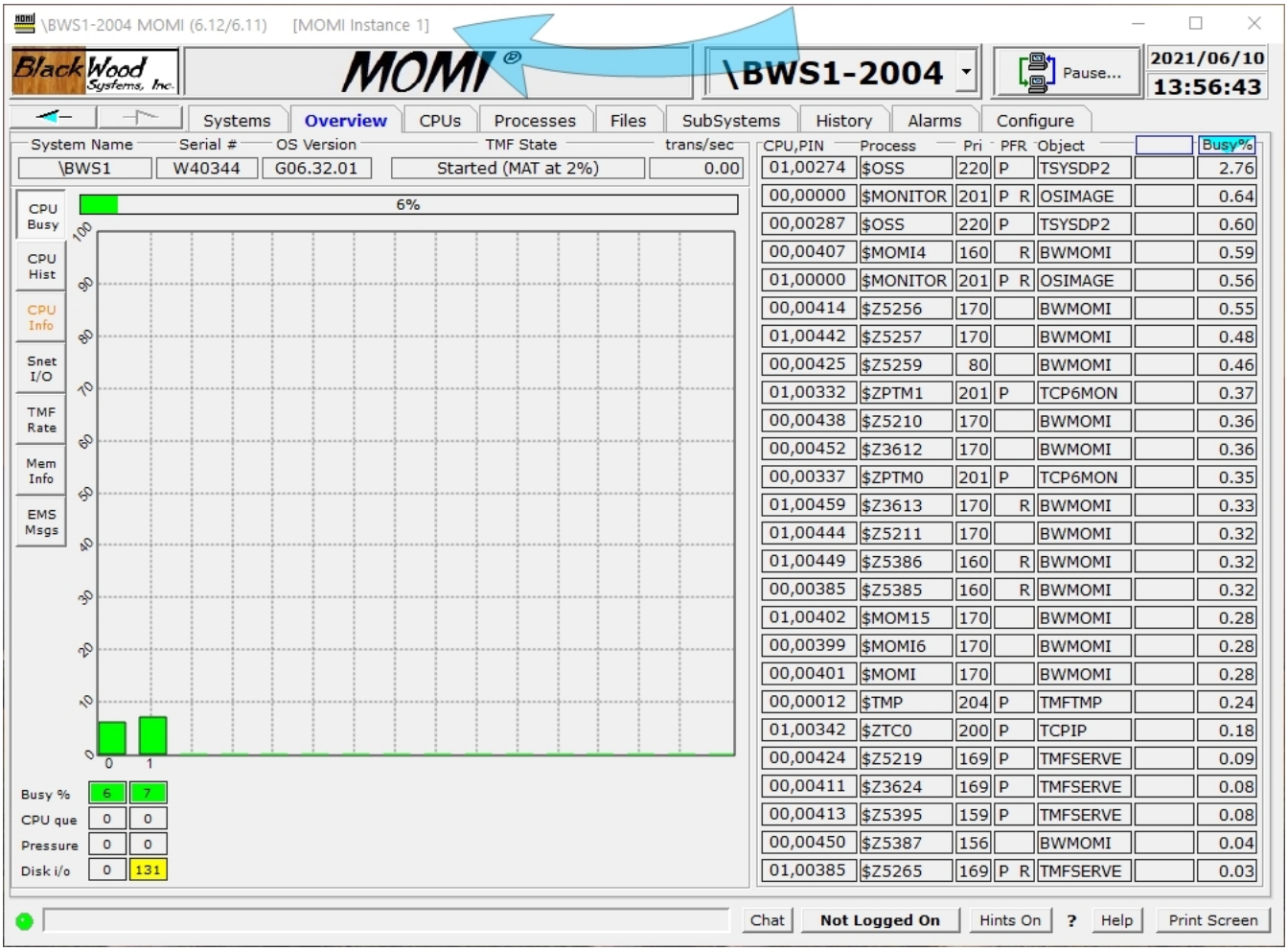
CIIENTID="<text>"

This parameter directs the MOMI PC Client to display the specified <text> on the window title bar.

The text is informational only.

Example:

```
MOMI.EXE clientid="MOMI Instance 1"
```



DONTSHOWSPACES

DONTSHOWSPACES

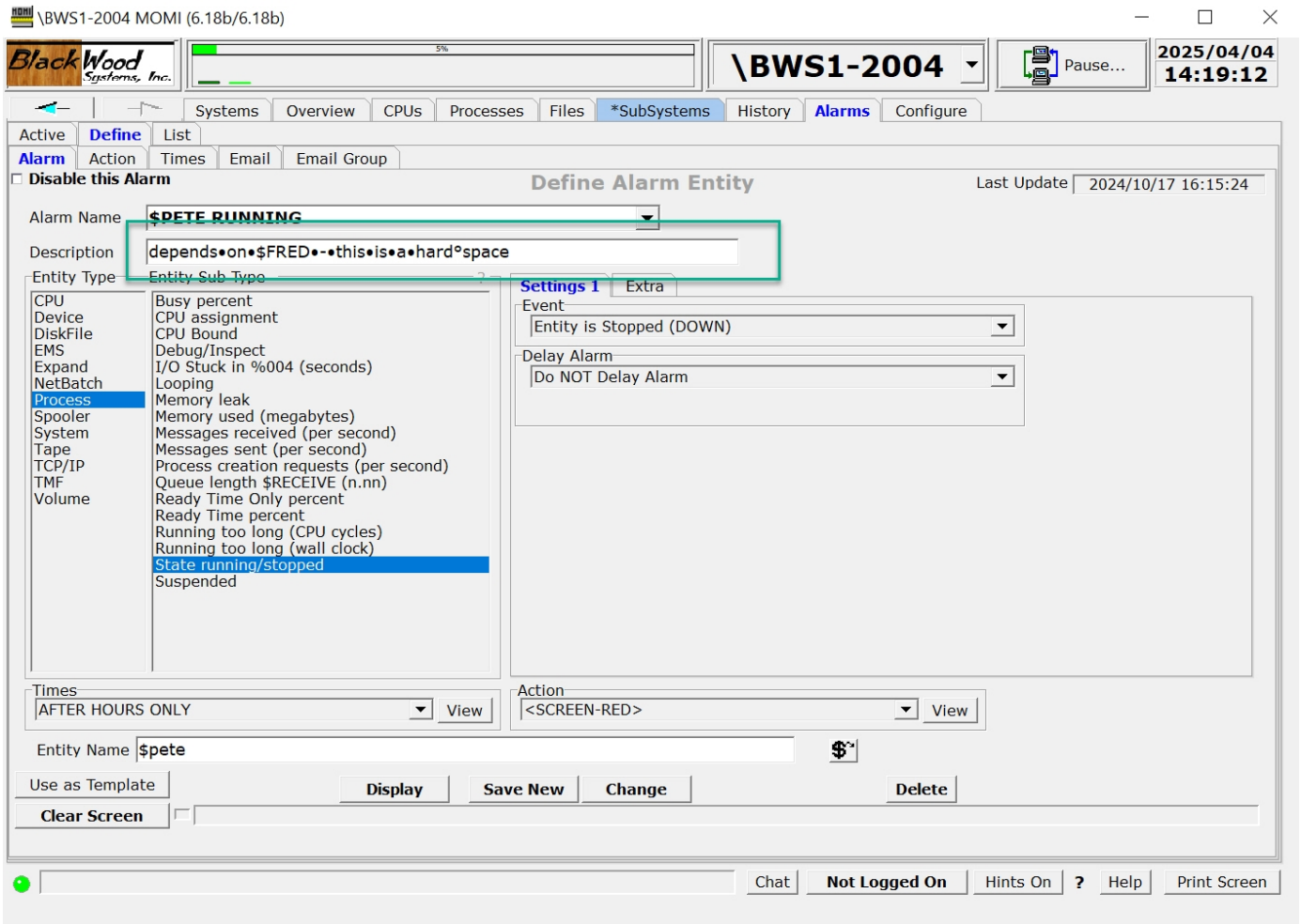
This parameter disables the use of the ESC key to visualize spaces and other characters.

By default, press and hold the ESC key causes the currently selected field (i.e. click within the field) to show spaces as a dot and hard spaces as a degree (sample image below). The field returns to normal when the ESC key is released. Not all fields support this function.

This ability requires that the setting Character Display (found under Fonts) be set to **None**.

Example:

```
MOMI.EXE DONTSHOWSPACES
```



FUNCTION-EVENTBX-EXPORT-UNFORMATTED

(client version 5.27 or later)

Function-EventBX-Export-Unformatted="<pc-file-name>"

This parameter directs the client to export all EVENTBX information as ASCII text.

The MOMI PC Client stops at the completion of processing this parameter. No other parameter is executed.

Note there are no spaces between the keyword, equal sign and the file name. If the file name contains spaces it must be enclosed within quotes.

Format of the exported data:

- one line per record
- each line is "owner ~ SSID ~ event number ~ text <CRLF>"
- event number is leading sign(+/-) and 5 digit leading zeros
- default item separator is Tilde ~ (#126) (may be changed in Configure)
- default embedded line-feeds within the text are <ETX> (#3) (may be changed in Configure)
- each line ends with <CRLF> (#10#13)

This parameter supersedes the parameter SCRIPT-EVENTBX-EXPORT-UNFORMATTED used during the BETA release of the software.

The syntax is as follows:

MOMI.EXE function-eventbx-export-unformatted=<pc-file-name>

Example:

```
                                exports EventBX
                                records into
                                'myBXs'

MOMI.EXE function-eventbx-export-unformatted=d:\data\fred\myBXs
```

FUNCTION-USERHINTS-IMPORT

(client version 5.27 or later)

Function-UserHints-Import="<pc-file-name>"

This parameter directs the client to import User Hints from a PC file into MOMI. See UserHints for the format of the file imported.

The MOMI PC Client stops at the completion of processing this parameter. No other parameter is executed.

Note there are no spaces between the keyword, equal sign and the file name. The file name must be enclosed in quotes if it contains a space.

This parameter supersedes the parameter SCRIPT-USERHINTS-IMPORT.

The syntax is as follows:

MOMI.EXE function-userhints-import=<pc-file-name>

Example:

```
                                        imports userhints file 'myhints'
MOMI.EXE function-userhints-import=d:\data\fred\myhints
```

INIPATH

INIPATH="<location>" (default = user profile area)

This parameter directs the MOMI PC Client to store its configuration information at the location specified. The parameter only specifies the path as the client always uses the name MOMI.INI.

Note there are no spaces between the keyword, equal sign and the location.

The Client configuration information in stored in a simple text file using standard INI format and by default located in the in the users profile area. This location is typically:

Windows 10 and later -

C:\Users\<User ID>\AppData\Roaming\BlackWood Systems\MOMI\

Where <User ID> is the name used at logon to Windows. You must enable *Show hidden files, folders and drives* under Windows Folder Options to enable their display.

MOMI also stores in the location specified by this parameter the files Calendar.mom, CustErrors.mom and the directories MomiData and Backups.

The default location provides user specific storage for MOMI settings and usually has no security implications. However, altering this location may be desirable if more than one User accesses the PC and common settings are needed or the client is started multiple times and each need their own specific settings.

The User could have different INI files to store different configurations for multiple Systems. Each System could have its own filters, color-coded EMS messages, thresholds, etc... specific a different location for each MOMI shortcut.

The MOMI.INI file may also be initialized with customer specific data. See PC Client initial configuration deployment for additional information.

Example:

```
MOMI.EXE inipath="d:\data\fred\momi"                redirects INI to
d:\data\fred\momi
```

LOG

Log

This parameter activates a logging feature in the MOMI PC Client to obtain information in special problem situations. When enabled, a file called **MOMI-Actions.log** is created on the desktop and **"-*** Action Logging ***"** is appended to the information in the programs title bar (upper left hand corner of the screen).

Usage of this parameter would be requested by BlackWood support. The information in the log would not normally be of any benefit to a user.

The syntax is as follows:

MOMI.EXE log

Example:

```
MOMI.EXE log                              activate special logging
```

NOCAIL

Nocail

This parameter directs the client to NOT load the CAIL terminal emulator.

The option is provided if the CAIL terminal emulator is not desired or cannot be loaded.

The syntax is as follows:

MOMI.EXE nocail

Examples:

(do not load the cail emulator)

MOMI.EXE nocail

NOSPEECH

(client version 4.07 or later)

Nospeech

Directs the client to NOT attempt speech activity.

The option is provided if the normal detection mechanism used by the MOMI PC Client fails on speech detection.

The syntax is as follows:

MOMI.EXE nospeech

Examples:

```
MOMI.EXE nospeech                                   don't enable speech
```

SAME

Same

The MOMI.INI file is placed in the same location as the MOMI.EXE.

The syntax is as follows:

MOMI.EXE same

Examples:

redirects INI to the same location as EXE

MOMI.EXE same

```
                    redirects INI to same location
                    as EXE, run script FRED

MOMI.EXE same script=fred
```

SCRIPT

Script="<script-name>"

This parameter directs the client to export information from the MOMI history files. See History / Export Data or information on defining the script of "predefined" commands that are executed.

The MOMI PC Client stops at the completion of processing this parameter. No other parameter is executed.

Note there are no spaces between the keyword, equal sign and the file name. The file name must be enclosed in quotes if it contains a space.

This parameter is replaced by SCRIPT-HISTORY.

The syntax is as follows:

MOMI.EXE script=<script-name>

Examples:

```
                                          executes script ProcessScript
MOMI.EXE script=ProcessScript

                                          sets INI to d:\data\fred\momi, execute
                                           script ProcessScript
MOMI.EXE inipath=d:\data\fred\momi  script=ProcessScript
```

SCRIPT-HISTORY

(client version 4.20 or later)

Script-History="<script-name>"

This parameter directs the client to export information from the MOMI history files. See History / Export Data for information on defining the script of "predefined" commands that are executed.

The MOMI PC Client stops at the completion of processing this parameter. No other parameter is executed.

Note there are no spaces between the keyword, equal sign and the file name. The file name must be enclosed in quotes if it contains a space.

This parameter supersedes the obsolete parameter SCRIPT.

The syntax is as follows:

MOMI.EXE script-history=<script-name>

Examples:

```
                                          executes script ProcessScript
MOMI.EXE script-history=ProcessScript

                                          sets INI to d:\data\fred\momi, execute
                                           script ProcessScript
MOMI.EXE inipath=d:\data\fred\momi  script-history=ProcessScript
```

SCRIPT-LOG

(client version 4.20 or later)

Script-Log="<script-name>"

This parameter directs the client to export information from the MOMI log file. See MOMI Log or information on defining the script of "predefined" commands that are executed.

The MOMI PC Client stops at the completion of processing this parameter. No other parameter is executed.

Note there are no spaces between the keyword, equal sign and the file name. The file name must be enclosed in quotes if it contains a space.

The syntax is as follows:

MOMI.EXE script-log=<script-name>

Examples:

```
                                          executes script ProcessScript
MOMI.EXE script-log=LogScript

                                          sets INI to d:\data\fred\momi, execute
                                           script LogScript
MOMI.EXE inipath=d:\data\fred\momi  script-log=LogScript
```

SCRIPT-USERHINTS-IMPORT

(client version 5.16 or later)

Script-UserHints-Import="<pc-file-name>"

This parameter directs the client to import User Hints from a PC file into MOMI. See UserHints for the format of the file imported.

The MOMI PC Client stops at the completion of processing this parameter. No other parameter is executed.

Note there are no spaces between the keyword, equal sign and the file name. The file name must be enclosed in quotes if it contains a space.

This parameter is superseded by the parameter FUNCTION-USERHINTS-IMPORT.

The syntax is as follows:

MOMI.EXE script-userhints-import=<pc-file-name>

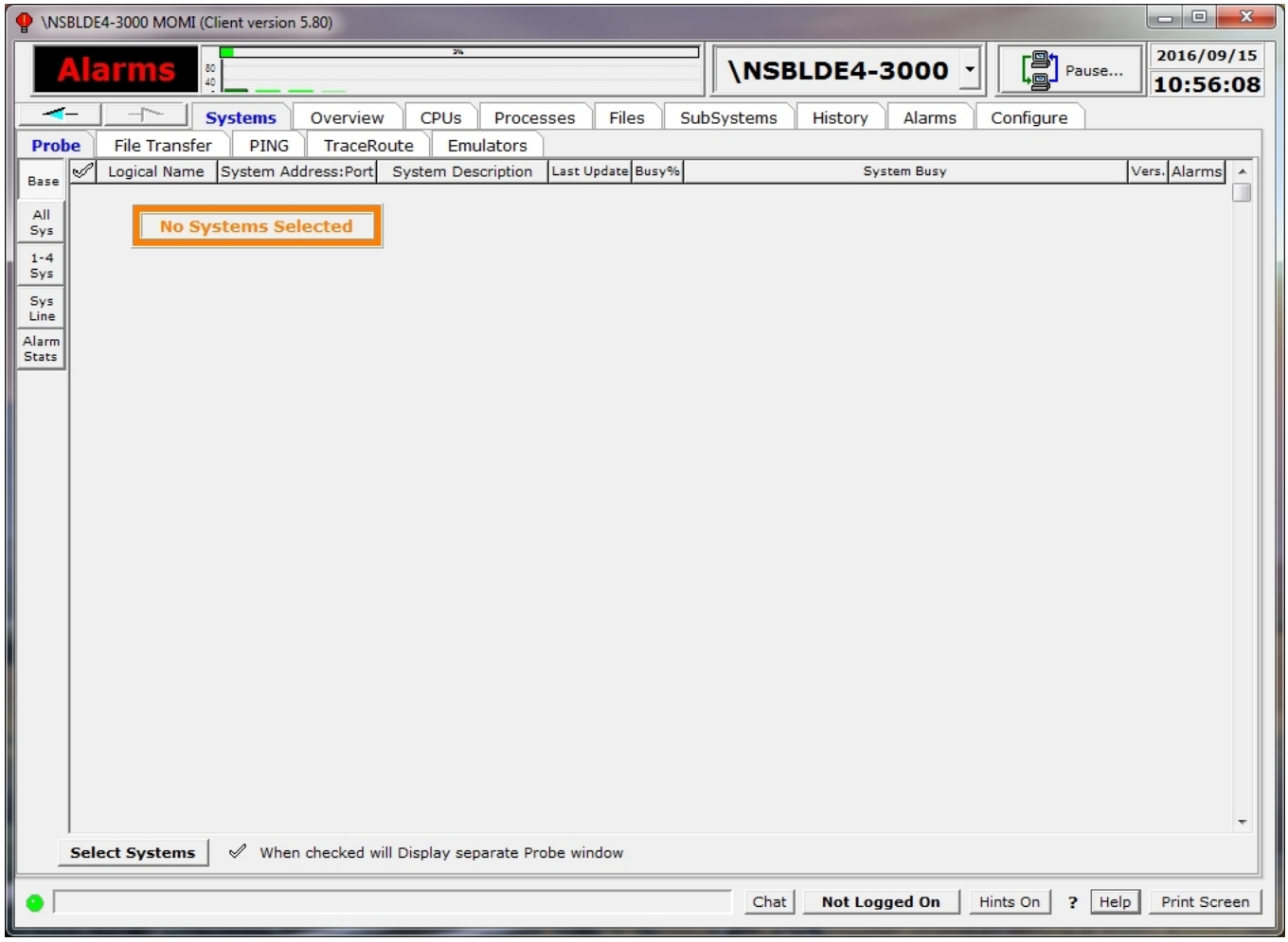
Example:

```
                                        imports userhints file 'myhints'
MOMI.EXE script-userhints-import=d:\data\fred\myhints
```

Screens

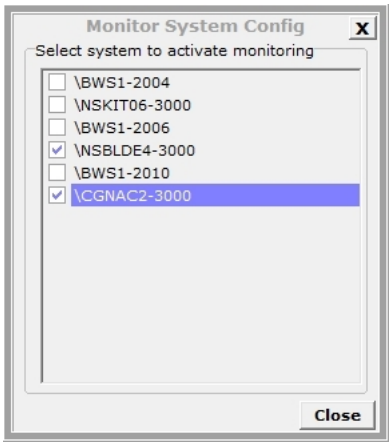
Systems

Systems / Probe

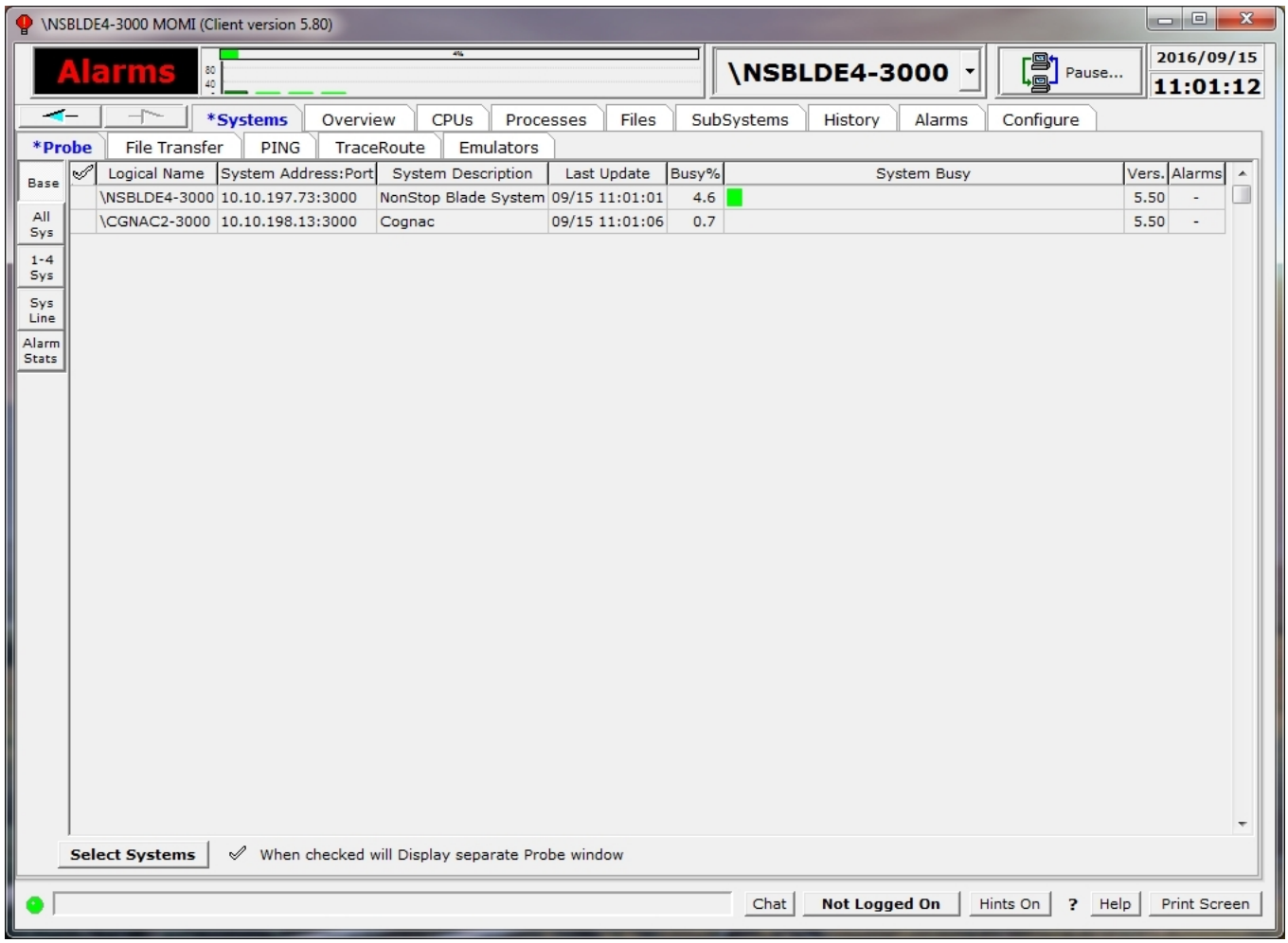


The Probe screen provides a method to configure the MOMI PC Client to display high level details on more than one system. The systems available on this screen were initially configured to MOMI on the Define Systems screen.

To configure the display, press the **Select Systems** screen. Check each system desired.



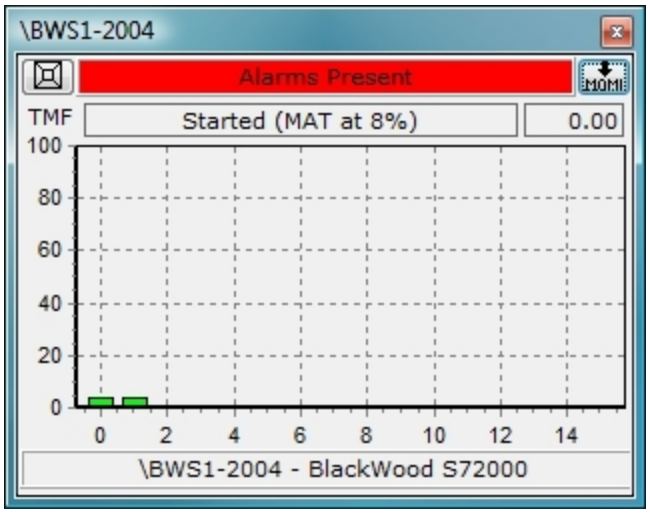
Information is displayed for each system selected. The column "Alarms" provides a general indication of any Alarms present on a system being probed. This column is not displayed by default. It is enabled on the screen Settings / Single Screen.



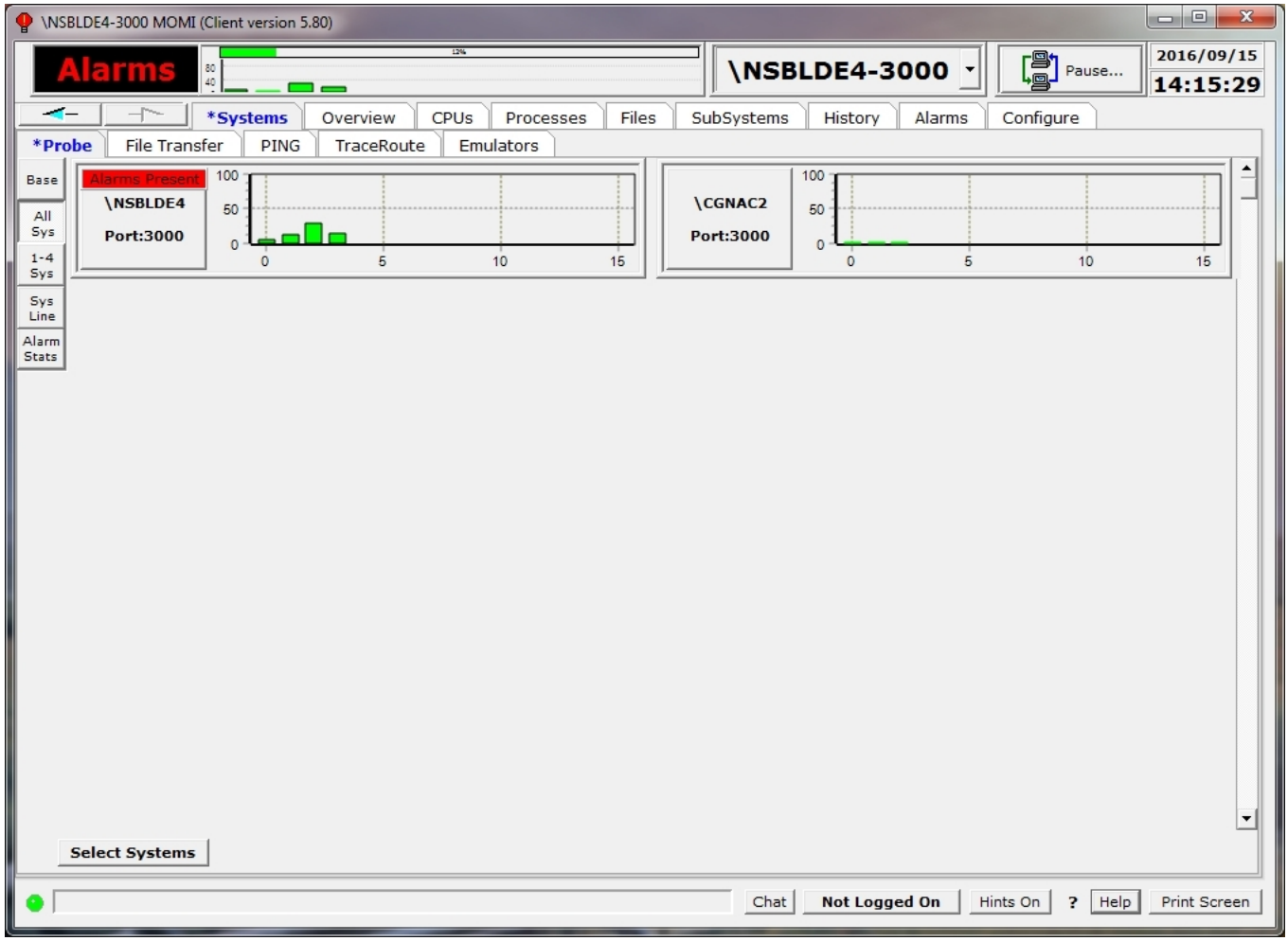
A floating independent window is available by checking provides basic information about a system. Thisresizable (grab the window edge and drag) window may be positioned anywhere on the

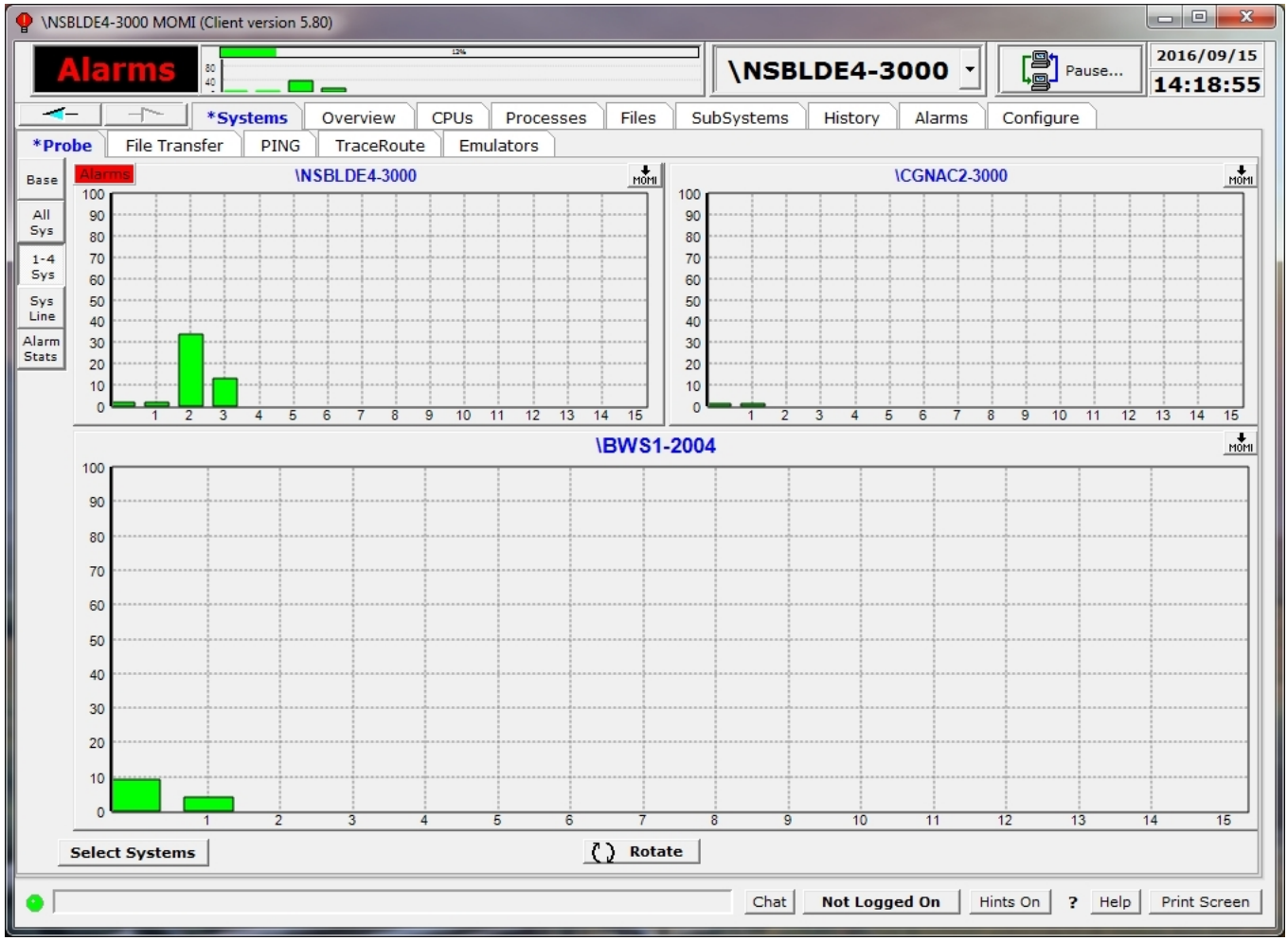
desktop. Clicking on the cpu busy graph bar displays the top few processes busy in that processor.

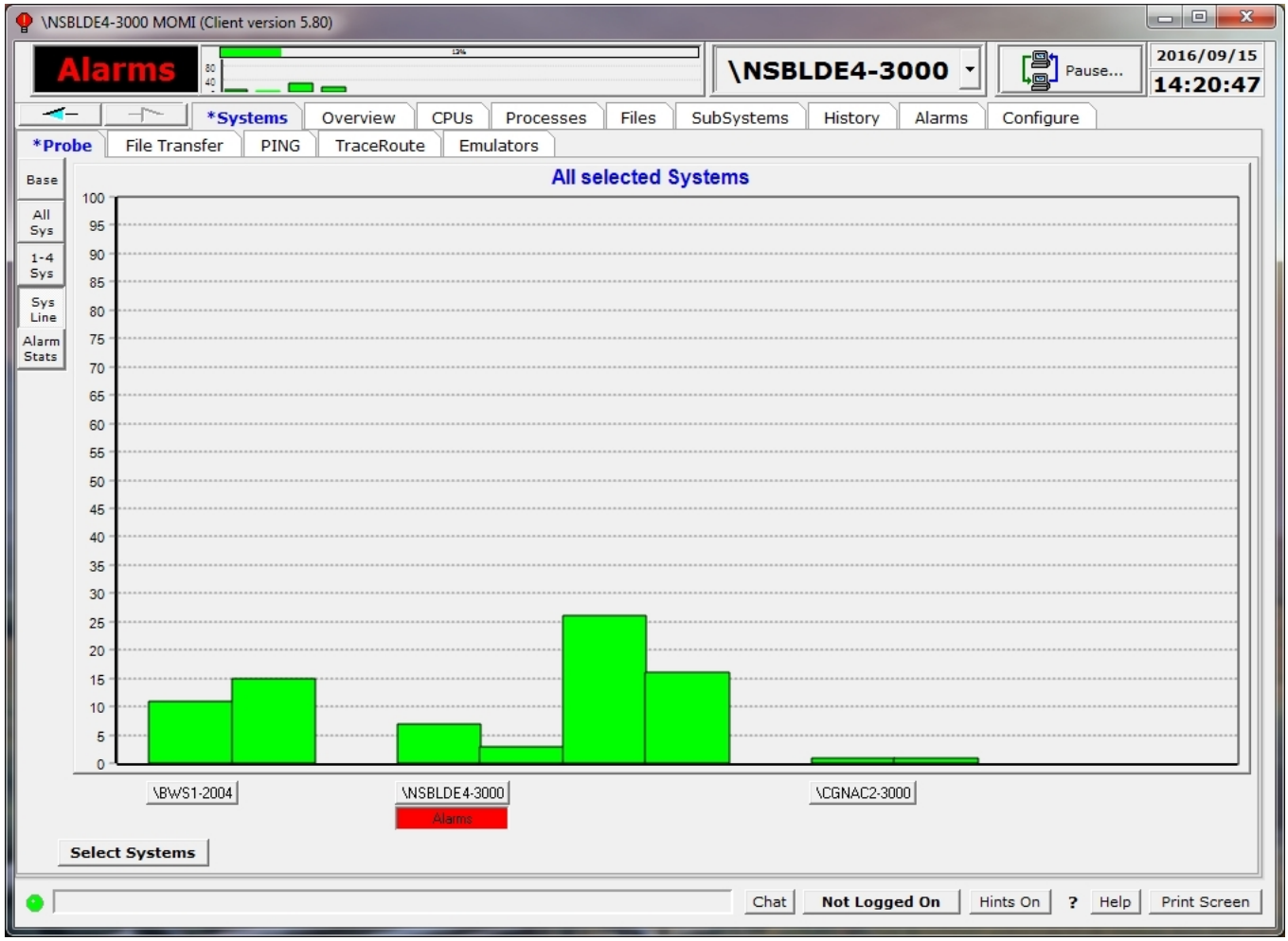
on the MOMI arrow button will select the main MOMI screen to that system.



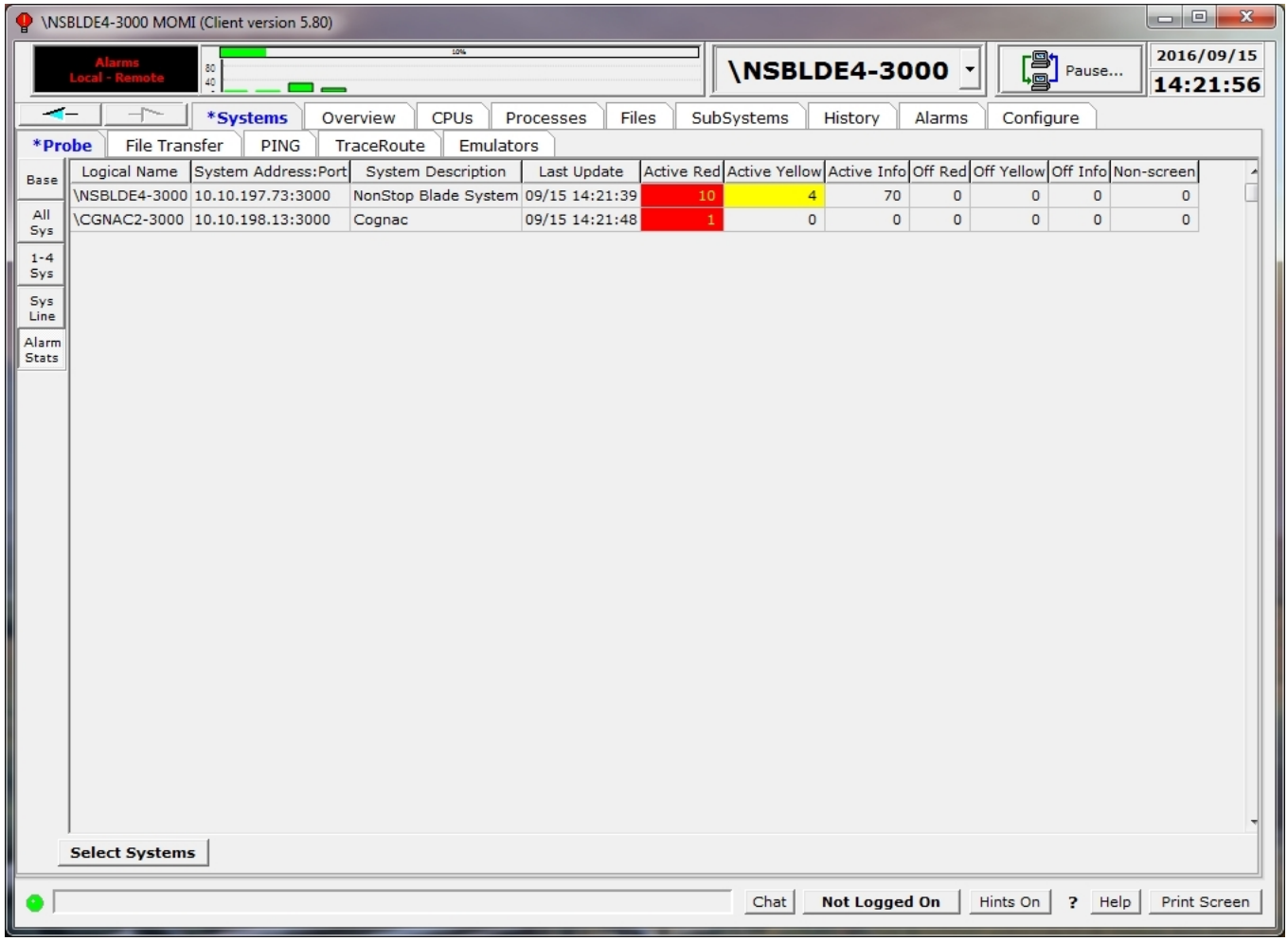
Additional sub-screens are available to present multiple systems in different formats. Press the All Sys, 1-4 Sys and Sys Line buttons to display these screens. Press the **Select Systems** button to choose which systems are displayed.


\NSBLDE4-3000 MOMI (Client version 5.80)
Alarms
\NSBLDE4-3000
Pause...
2016/09/15
14:15:29
*Systems
Overview
CPUs
Processes
Files
SubSystems
History
Alarms
Configure
*Probe
File Transfer
PING
TraceRoute
Emulators
Base
All Sys
1-4 Sys
Sys Line
Alarm Stats
Alarms Present
\NSBLDE4
Port:3000
\CGNAC2
Port:3000
Select Systems
Chat
Not Logged On
Hints On
?
Help
Print Screen

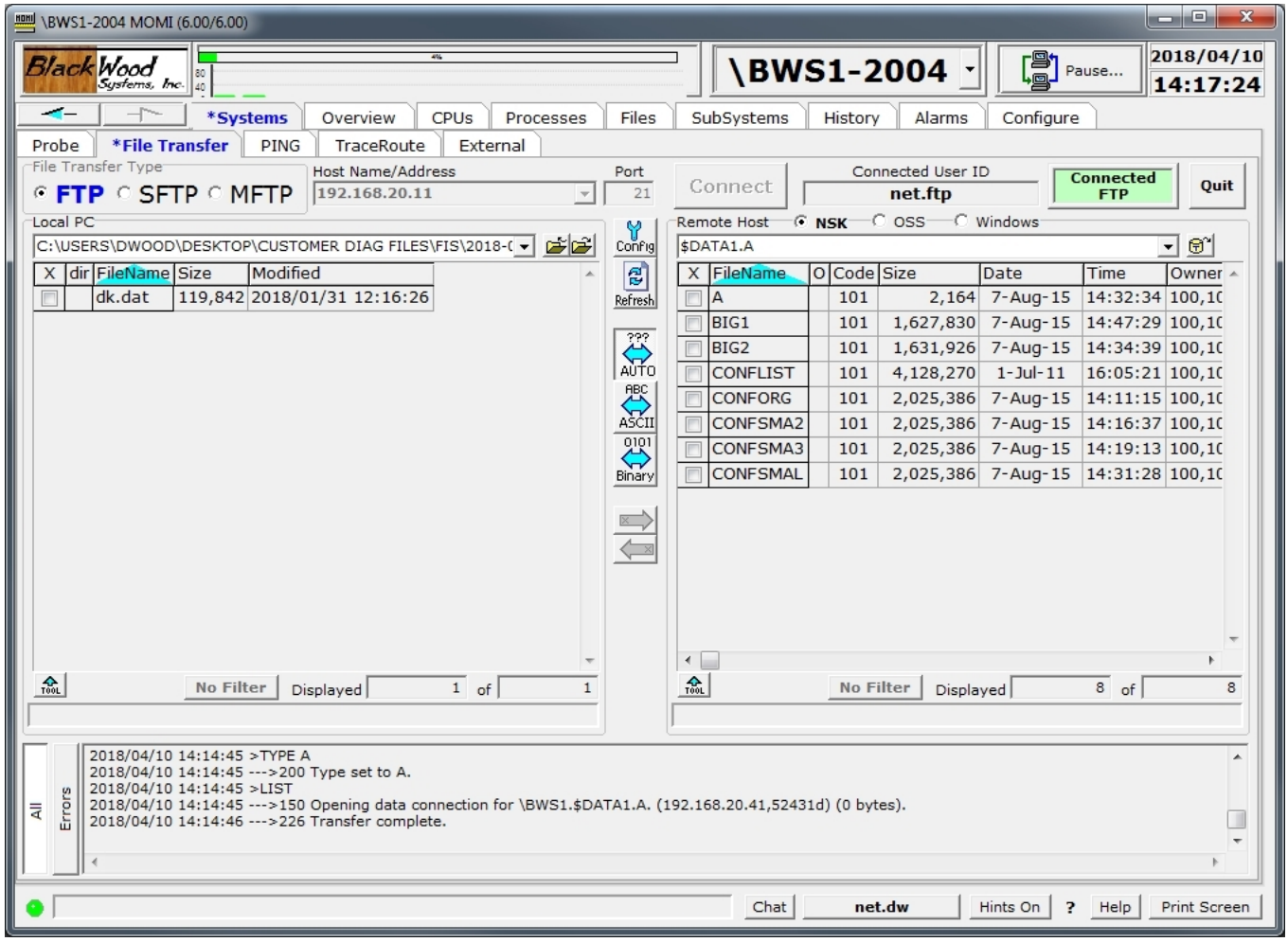

\NSBLDE4-3000 MOMI (Client version 5.80)
Alarms
\NSBLDE4-3000
Pause...
2016/09/15
14:18:55
*Systems
Overview
CPUs
Processes
Files
SubSystems
History
Alarms
Configure
*Probe
File Transfer
PING
TraceRoute
Emulators
Base
All Sys
1-4 Sys
Sys Line
Alarm Stats
Alarms
\NSBLDE4-3000
\CGNAC2-3000
\BWS1-2004
MOMI
Select Systems
Rotate
Chat
Not Logged On
Hints On
?
Help
Print Screen



The Alarm Stats sub-screen performs two functions. First, it activates which systems are probed for alarm information. Second, the information is displayed here and on other screens.


\NSBLDE4-3000 MOMI (Client version 5.80)
Alarms
Local - Remote
\NSBLDE4-3000
Pause...
2016/09/15
14:21:56
*Systems
Overview
CPUs
Processes
Files
SubSystems
History
Alarms
Configure
*Probe
File Transfer
PING
TraceRoute
Emulators
Base
All Sys
1-4 Sys
Sys Line
Alarm Stats
Logical Name
System Address:Port
System Description
Last Update
Active Red
Active Yellow
Active Info
Off Red
Off Yellow
Off Info
Non-screen
\NSBLDE4-3000 10.10.197.73:3000 NonStop Blade System 09/15 14:21:39 10 4 70 0 0 0 0
\CGNAC2-3000 10.10.198.13:3000 Cognac 09/15 14:21:48 1 0 0 0 0 0 0
Select Systems
Chat
Not Logged On
Hints On
?
Help
Print Screen

Systems / File Transfer



The Systems / File Transfer screen provides multiple methods for transferring files to and from the local PC to a remote system. The methods available for FTP, SFTP and MFTP. Connections to Tandem hosts may display files in Guardian or OSS format.

FTP (unsecured File Transfer Protocol) and SFTP (Secured File Transfer Protocol) are industry standard methods which require the appropriate server operating on the remote host. MFTP (MOMI File Transfer Protocol) uses and requires a MOMI server previously configured on the screen Defined Systems.

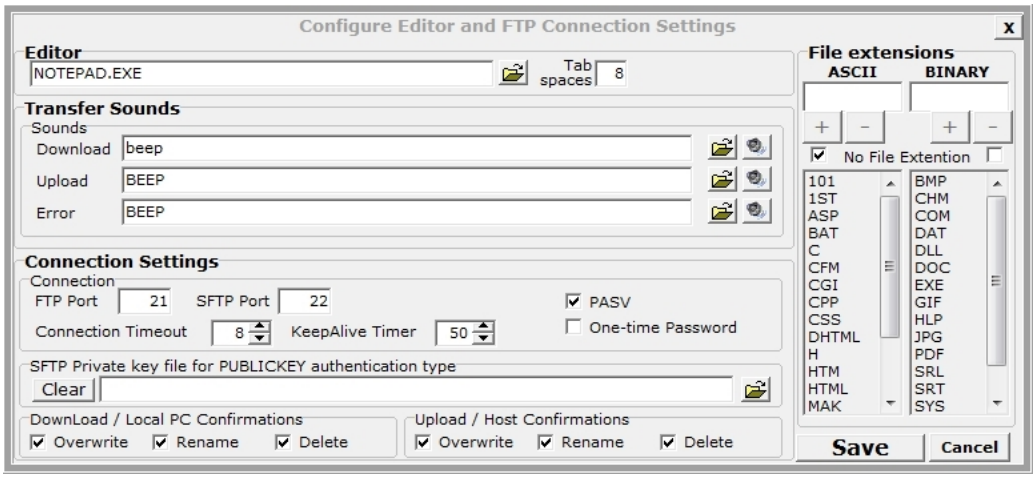
All file transfer methods on this screen generally operate in the same manner, select the method, enter the destination IP address or select the MOMI system, User ID and Password then press **Connect**. Files at the default location are displayed after a successfully connection.

Files are transferred by drag and drop or by selecting the file(s) then press a direction arrow. The manner in which the files are interpreted when read is either ASCII or binary. ASCII transfers assumes the file contains printable characters and uses CR/LF as line delimiters. Files on the NonStop System with a file code of 101 are edit format and are read/written using O/S supplied

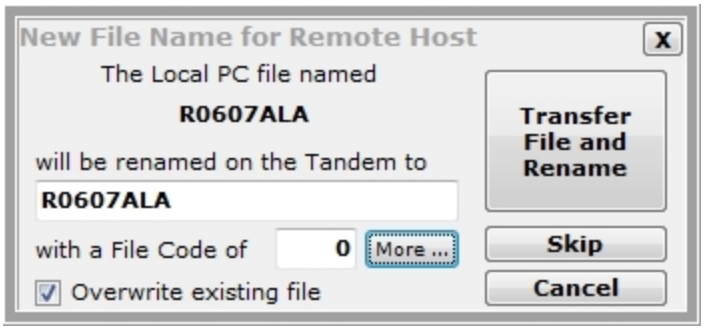
routines to maintain the internal ASCII structure. Binary files are transferred 'as-is' or byte-for-byte. AUTO or Automatic uses the file extension to determine the ASCII or binary format. File extensions are defined by pressing the **Config** button. The pop-up window also sets other features such as the editor, sounds and other default connection settings.

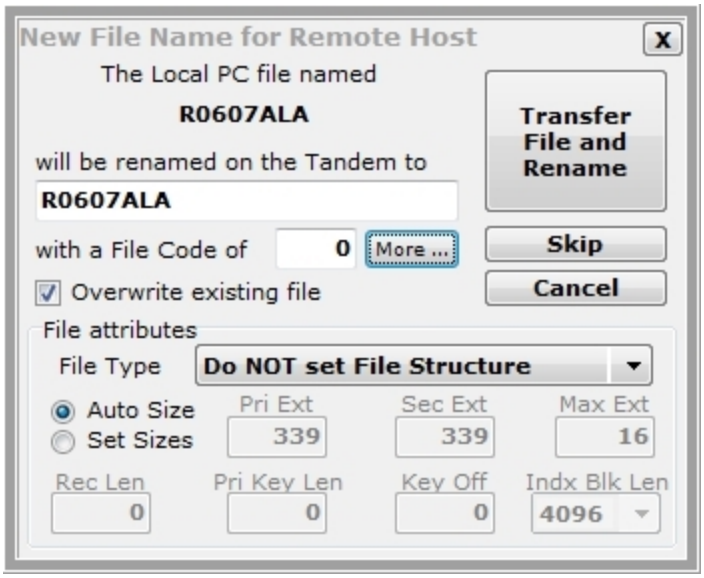
If automatic transfer mode is enabled and the file extension is not found in either ASCII or BINARY, the following logic applies:

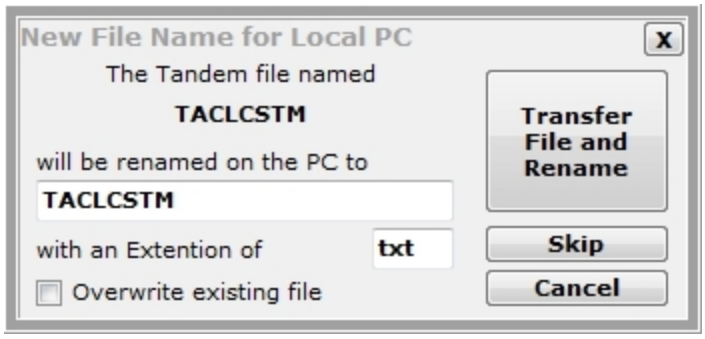
Location	Action
PC	read the first 10 lines of the file and if all the characters are printable - transfer in ASCII
NSK	file code 101 (edit file) transfers in ASCII
OSS	ASCII
Windows	ASCII



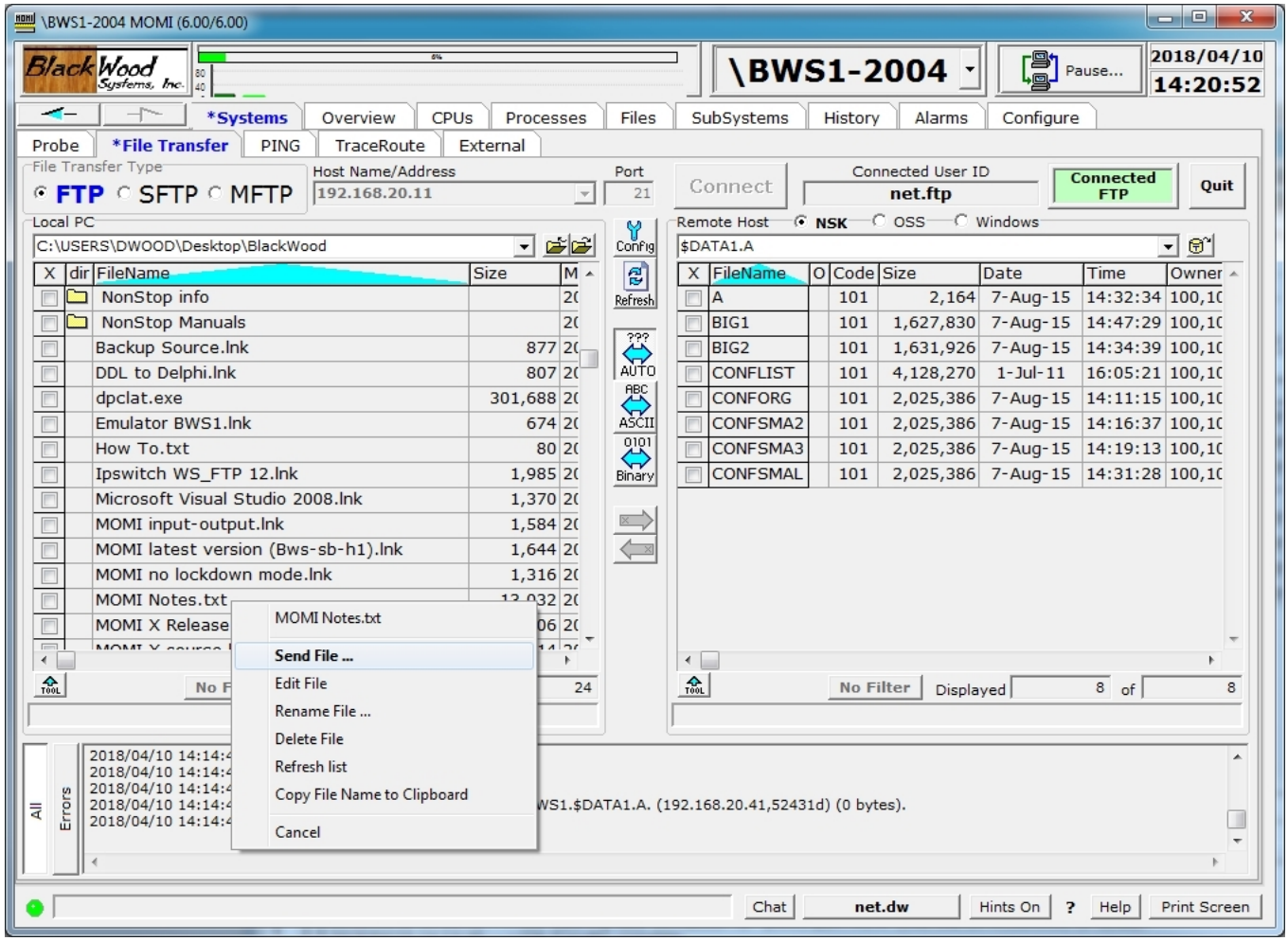
After a drag and drop or selecting a file and pushing a file transfer direction arrow, by default a window appears to set or change transfer attributes. Pushing the More... button expand the pop-up with additional options such as structured file transfers. Available options vary based on the transfer protocol.



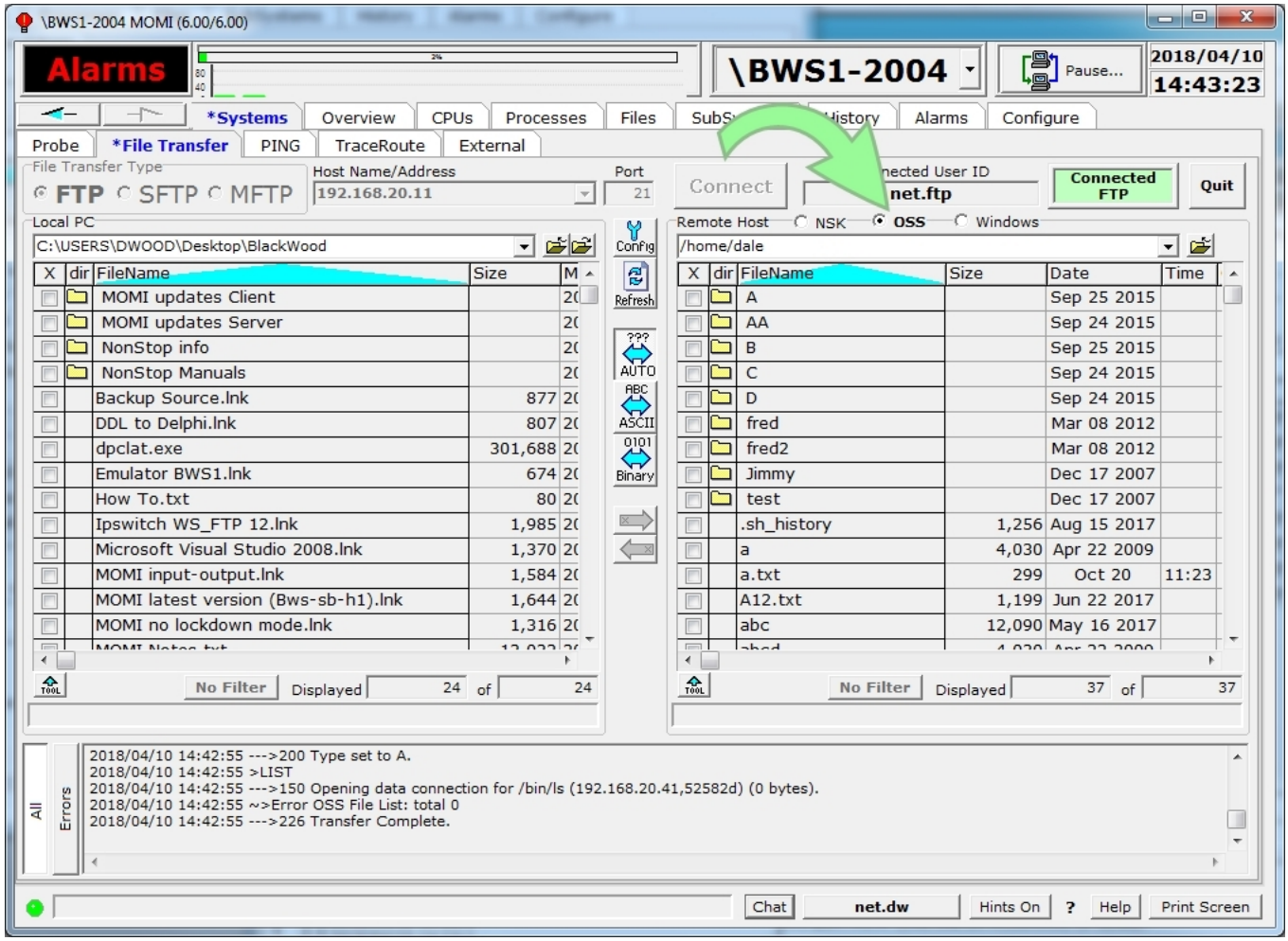




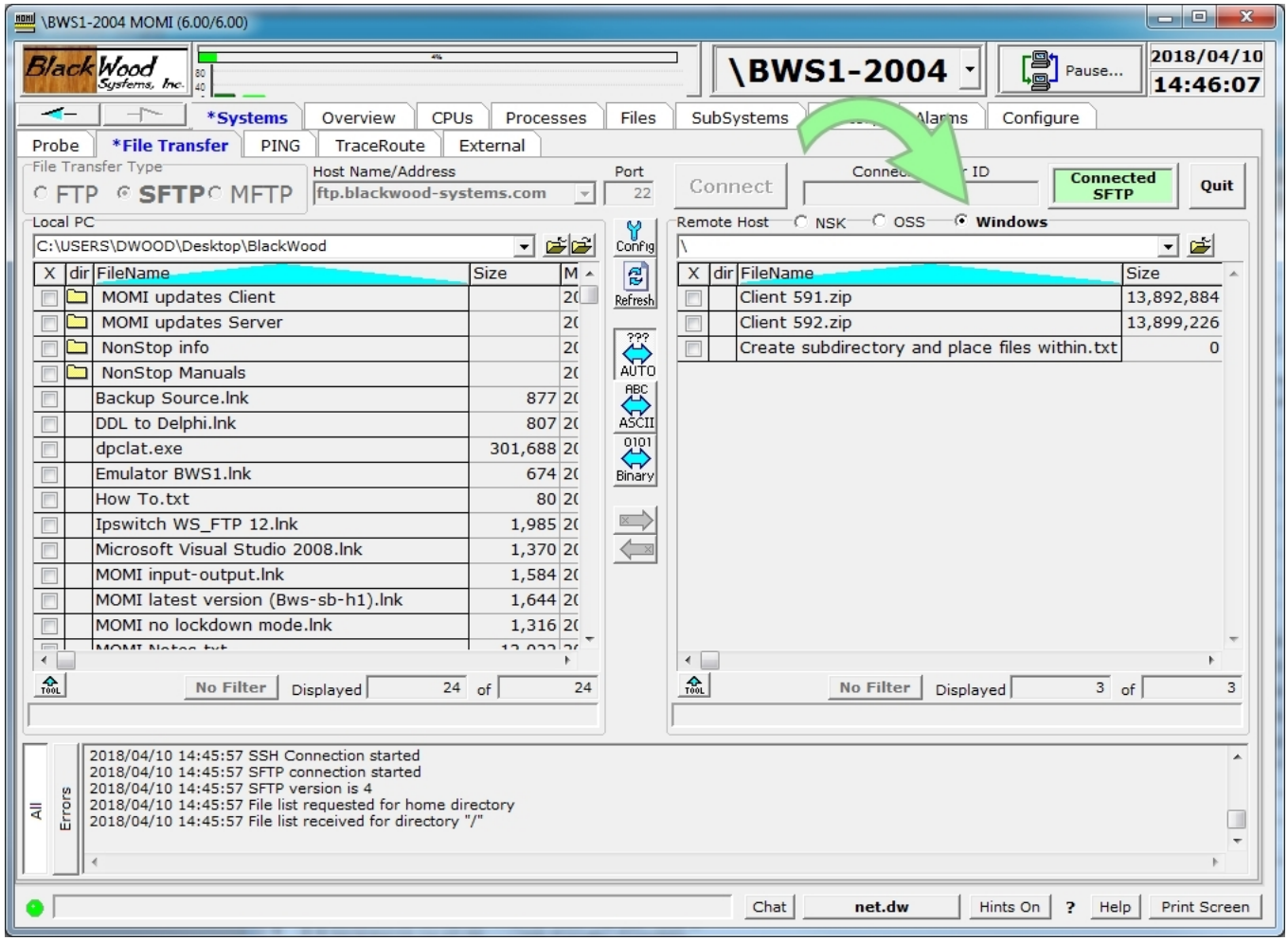
Right-click on a file name to display a context sensitive menu containing operation to transfer, edit, rename or delete. Edit File... causes the file to transfer to the local system, if needed, and then transfer back after the editing is complete. The editor invoked is set within **Config**.



Files on the right hand side may be displayed in the UNIX type format by selecting **OSS**. Folders are displayed for subdirectories (the field is blank for files). Click on a folder to display its contents. Click on the folder with the up arrow (to the right of the current path field) to navigate up one level.



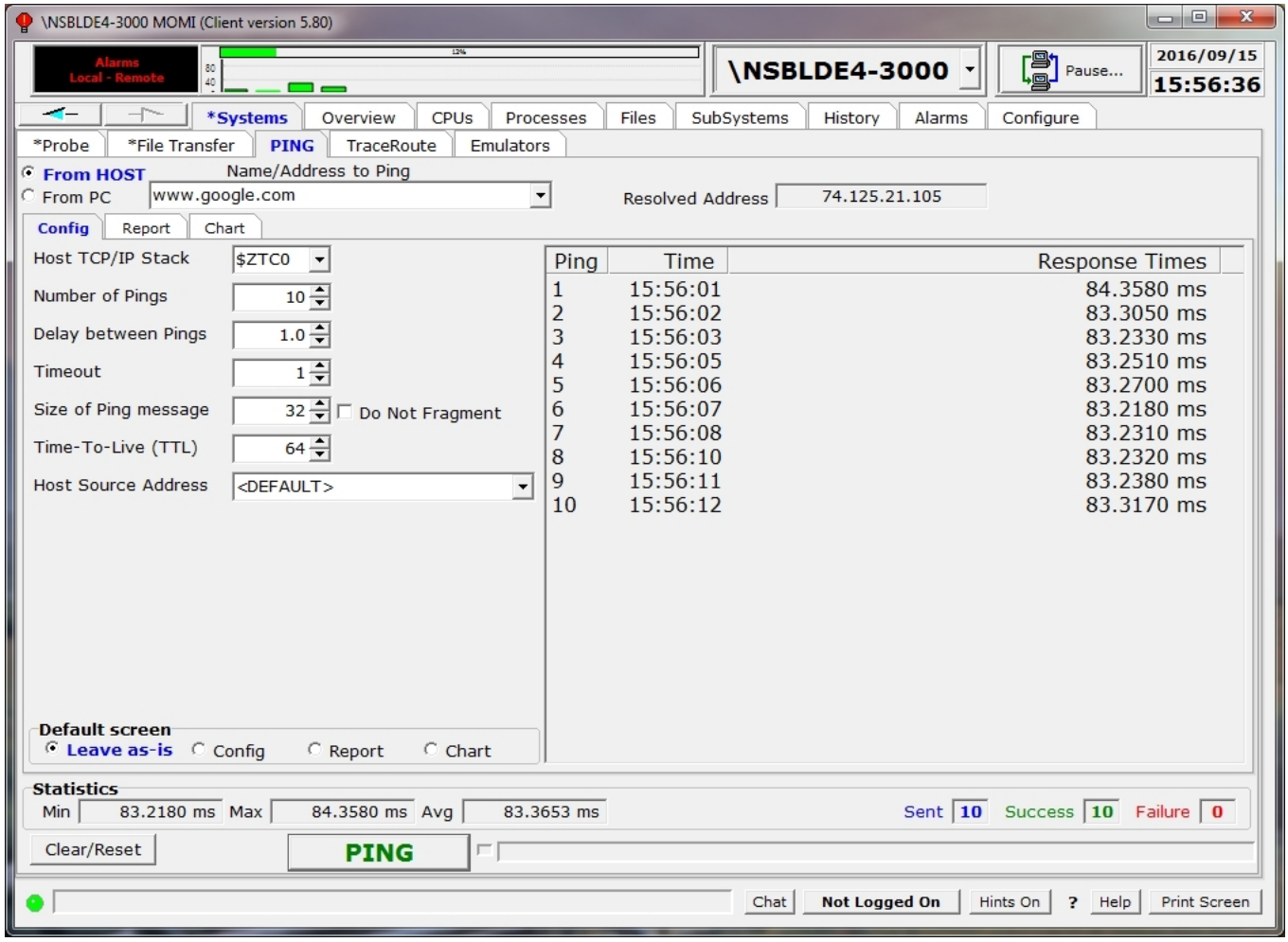
Click on the circle next to the **Windows** label to enable the conventional PC personality. Folders are displayed for subdirectories (the field is blank for files). Click on a folder to display its contents. Click on the folder with the up arrow (to the right of the current path field) to navigate up one level.



Press **Quit** to drop an established session.

Certain file transfer activities while active may temporarily freeze or lock-up other MOMI PC Client screens.

Systems / PING



The PING screen provides the ability to issue a communication test from either the NonStop System or the PC to the specified IP address. The PING may originate either from the currently connected HOST or from the local PC.

PING is the generic name used to describe a simple diagnostic of the communication path between the source (i.e. NonStop System or PC) and destination.

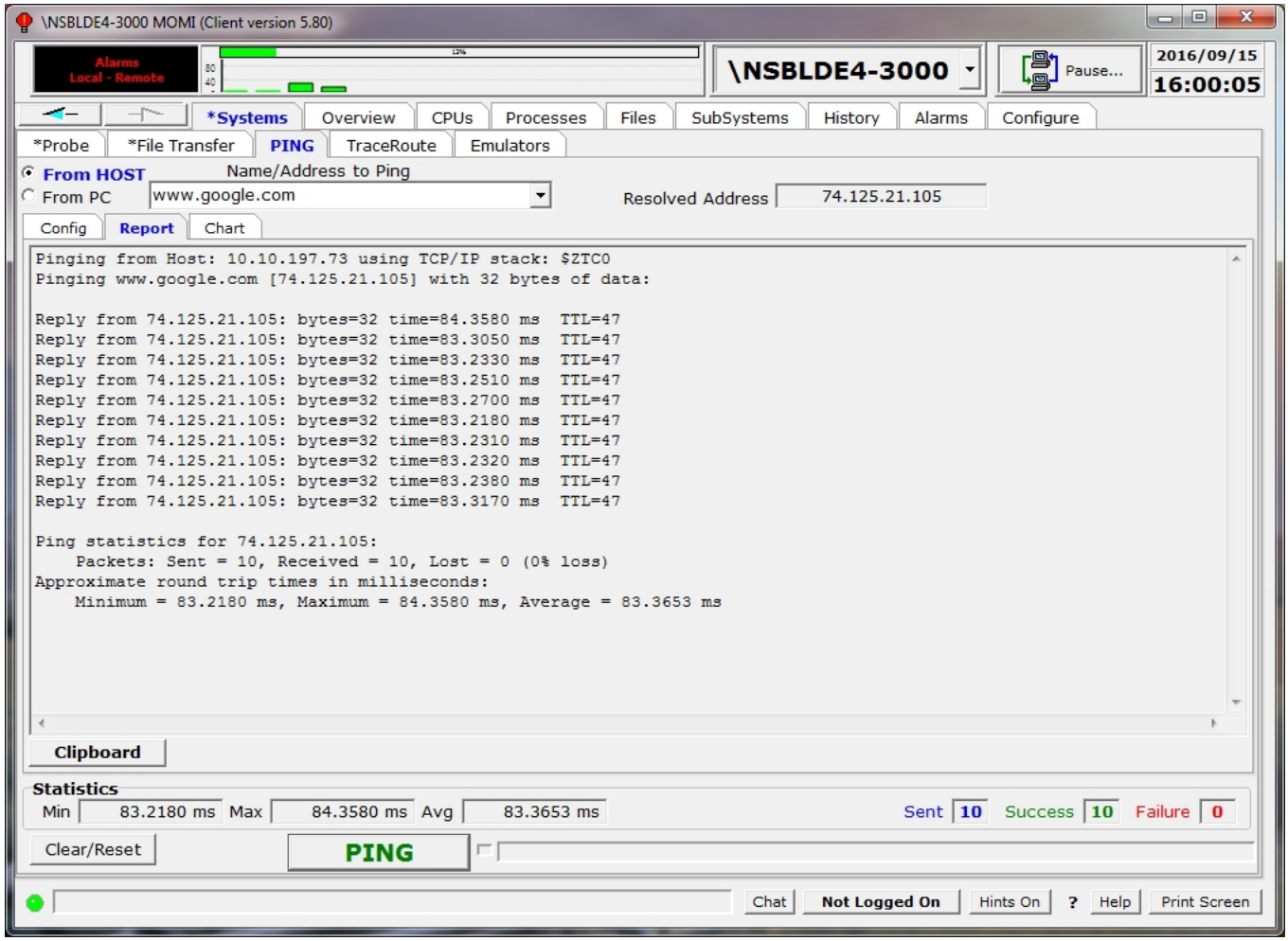
A PING consists of an ICMP echo message that most TCP/IP stacks acknowledge by simply echoing the original request. Note that a) some networks disable the ability to respond to the PING, and b) just because a PING works that does not mean other protocols will work.

The screen is used by selecting either "From Host" or "From PC", entering a destination as an DNS name or IP address and pressing the PING button. Optionally, the number of PING attempts (Count), the time delay in seconds between each attempt, the amount of time to wait for a PING to respond and the TCP/IP Stack to issue the PING through. The center of the screen displays the

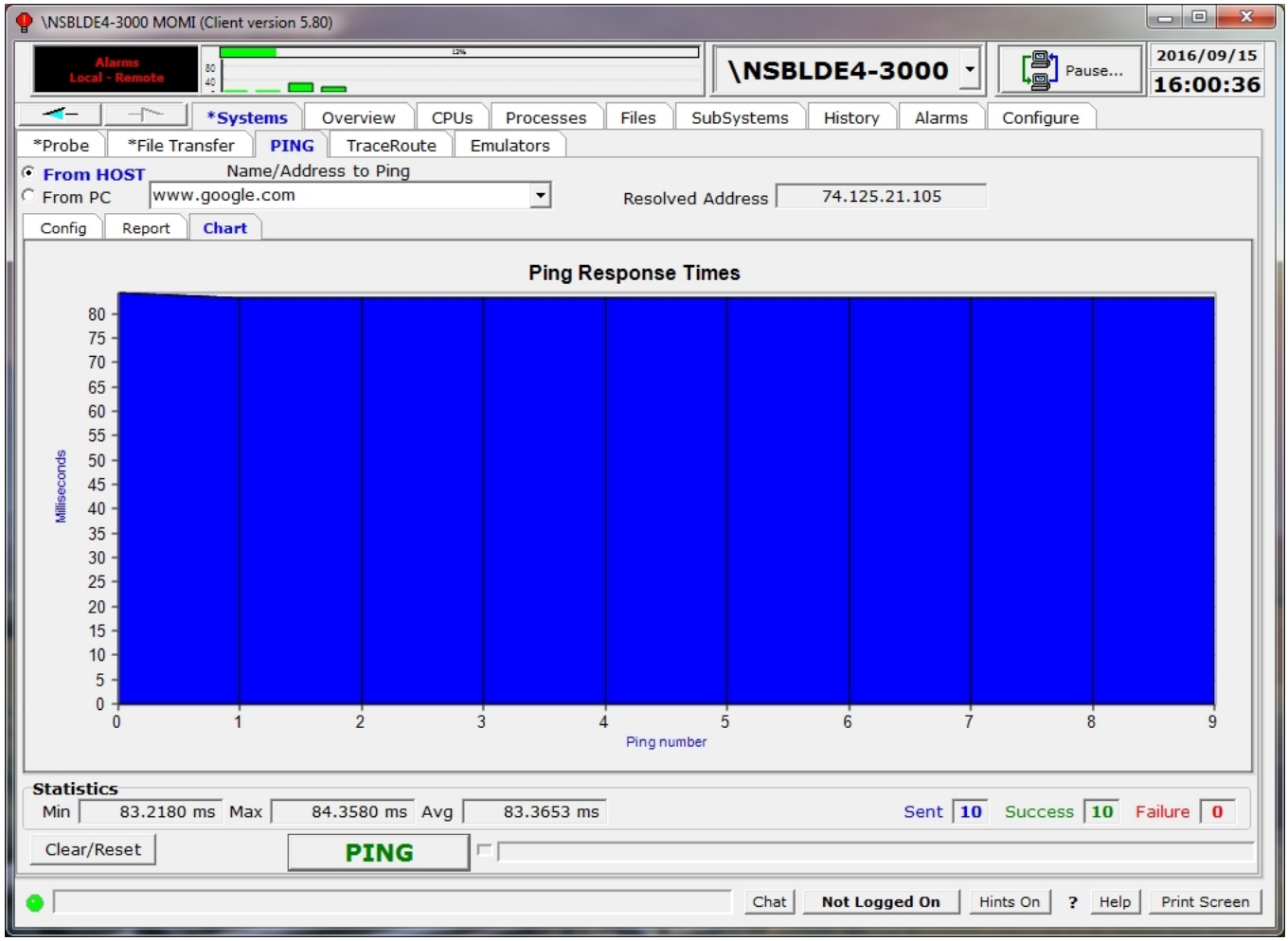
amount of time each response took and a graphical presentation of the response time. General statistics about the number of attempts, successful and failed pings are also displayed.

The ability to PING from the HOST may require the setup of an additional MOMI server to perform the function. See General considerations for additional information.

The Report tab displays the results in a format similar to the Windows PING utility.



The Chart tab display a graphic of the response time.



Possible errors

Failure to open socket - Error 4013

This error may indicate that BWSSG is present and not PROGID'ED to a Super.Group owner.

See BWSSG in the General considerations of Security.

BWSSG and BWMOMI object creation timestamp do not match (update BWSSG)

This error indicates that the BWSSG object was updated when a new version of the server was installed.

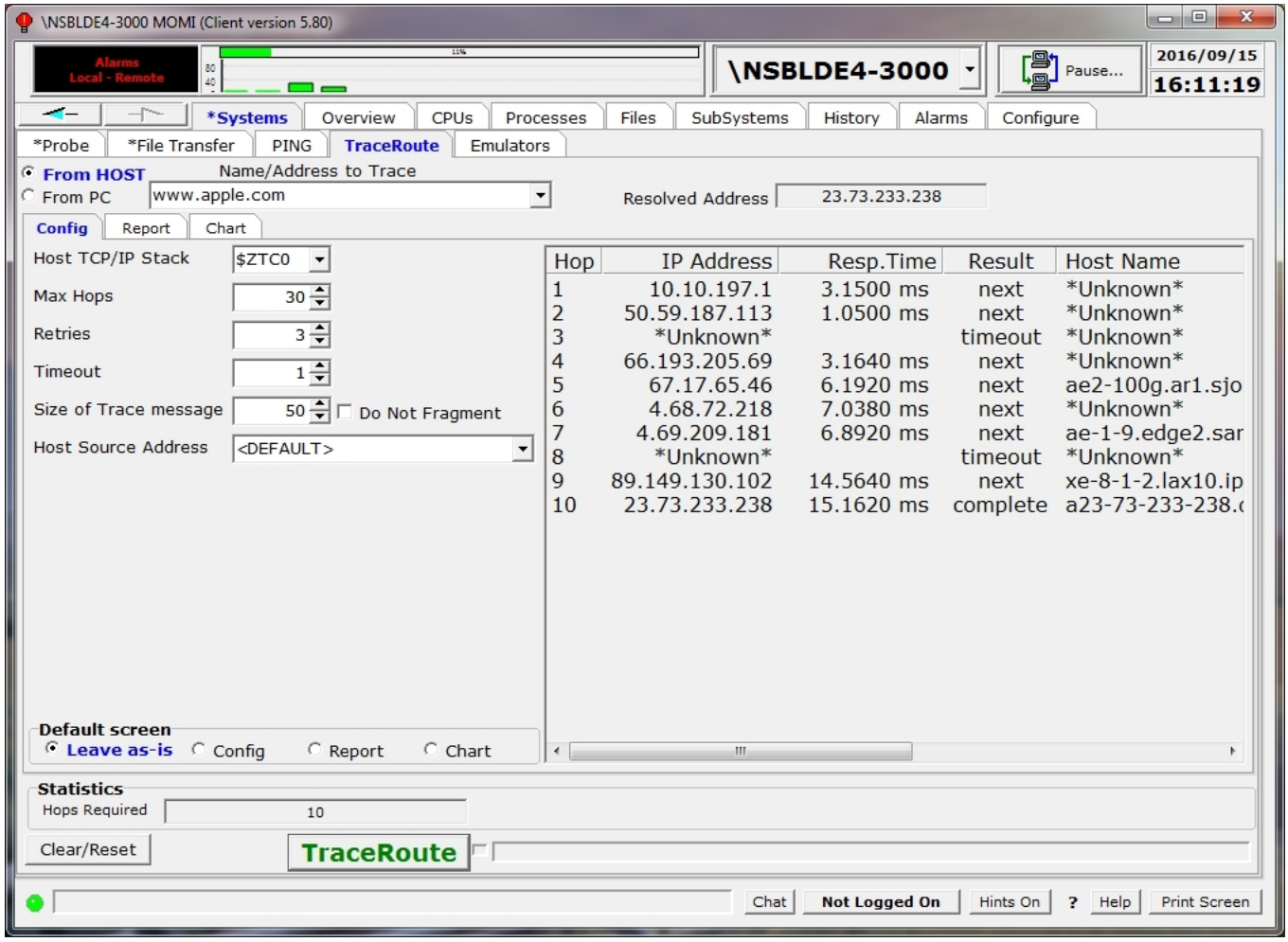
See the steps on how to create or update BWSSG.

Error accessing BWSSG or BWMOMI object file creation timestamp (check file security)

This error indicates that BWSSG is present but cannot be accessed.

Check the file security. This file is located in the same subvolume as the MOMI server.

Systems / TraceRoute



The TraceRoute screen provides the ability to PING all devices in-between another host or device on the TCP/IP network. The PING may originate either from the currently connected HOST or from the local PC.

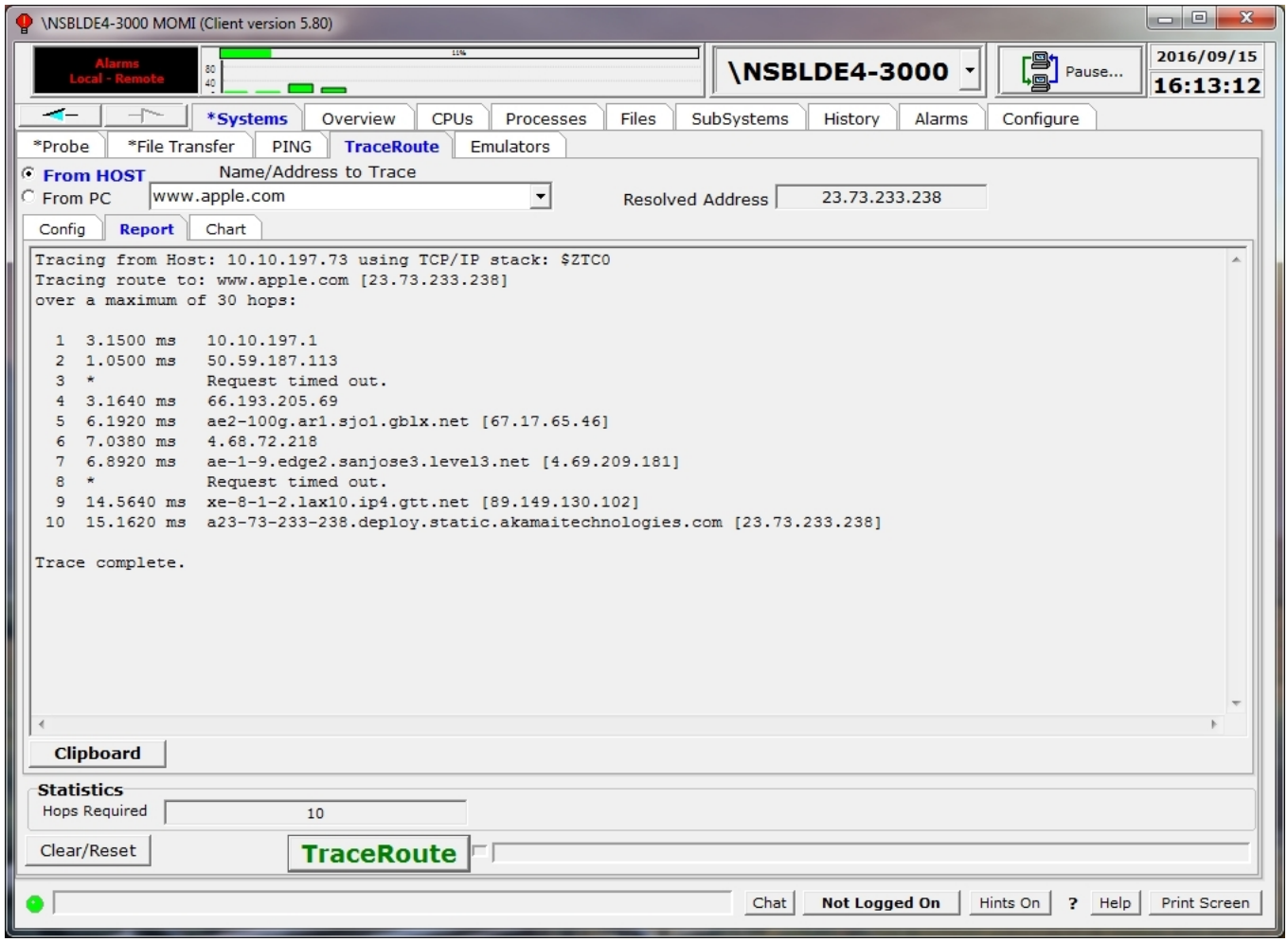
PING is the generic name used to describe a diagnostic to test if communication is possible between the source (i.e. HOST or PC) to the destination. A TraceRoute uses a form of PING where the TTL (Time-To-Live) count starts at 1 and is incremented (to an absolute maximum of 255) allowing the display of each intervening device (usually a router) until the destination is reached. The resultant trace of information is an indication of the number of hops through the devices required to achieve a communications connection end-to-end.

The screen is used by selecting either "From Host" or "From PC", entering a destination as an DNS name or IP address and pressing the TraceRoute button. Optionally, the maximum number of Hops, the retries per device, the amount of time to wait for a response and the TCP/IP Stack to issue the PING through. The center of the screen displays a trace of each device (IP address and

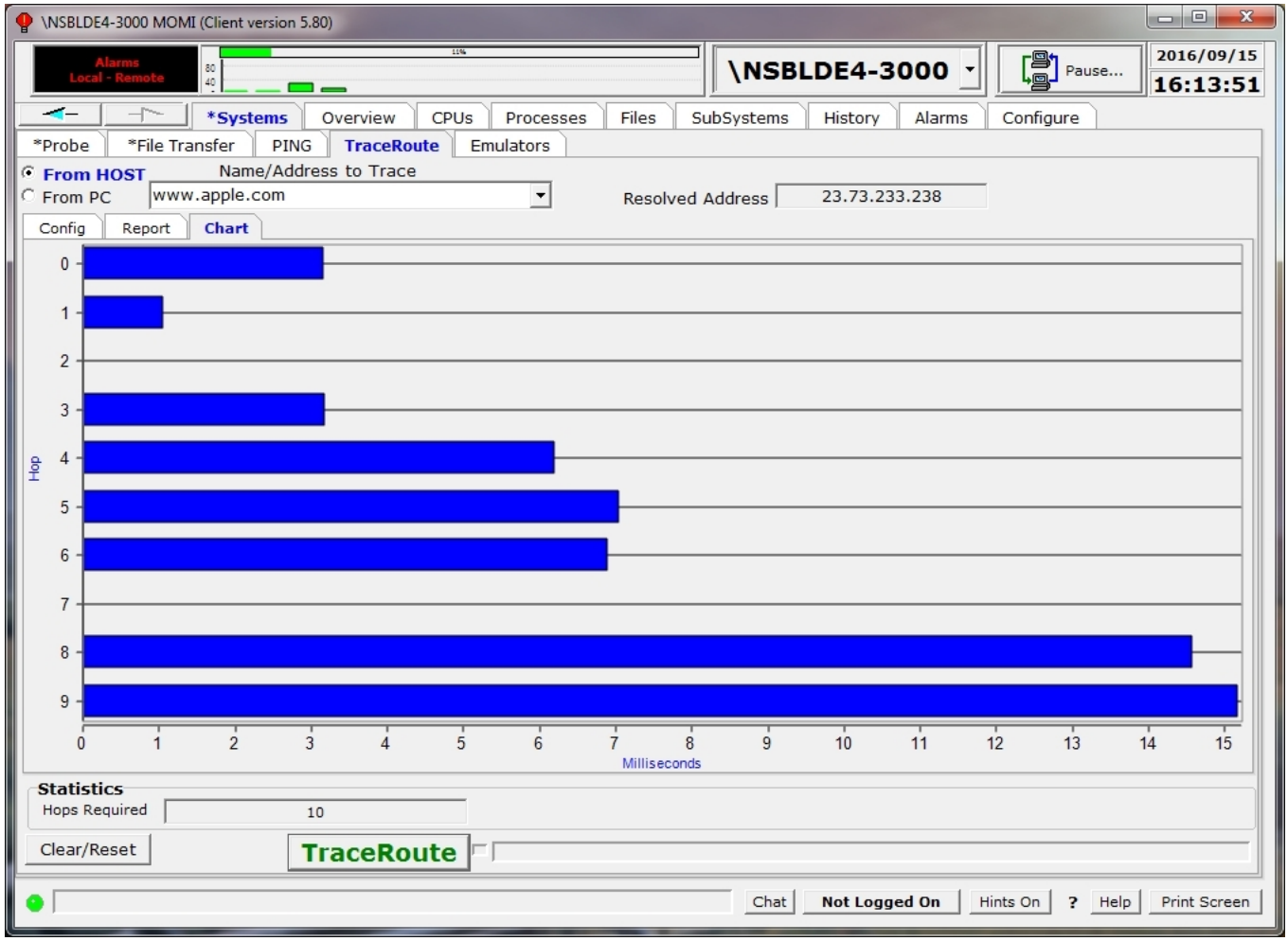
DNS name if available) and the amount of time each response took. The graphic displays the response time.

The ability to PING from the HOST may require the setup of an additional MOMI server to perform the function. See General considerations for additional information.

The Report tab displays the results in a format similar to the Windows Tracert utility.



The Chart tab display a graphic of the PING response time.



Possible errors

Failure to open socket - Error 4013

This error may indicate that BWSSG is present and not PROGID'ED to a Super.Group owner.

See BWSSG in the General considerations of Security.

BWSSG and BWMOMI object creation timestamp do not match (update BWSSG)

This error indicates that the BWSSG object was updated when a new version of the server was installed.

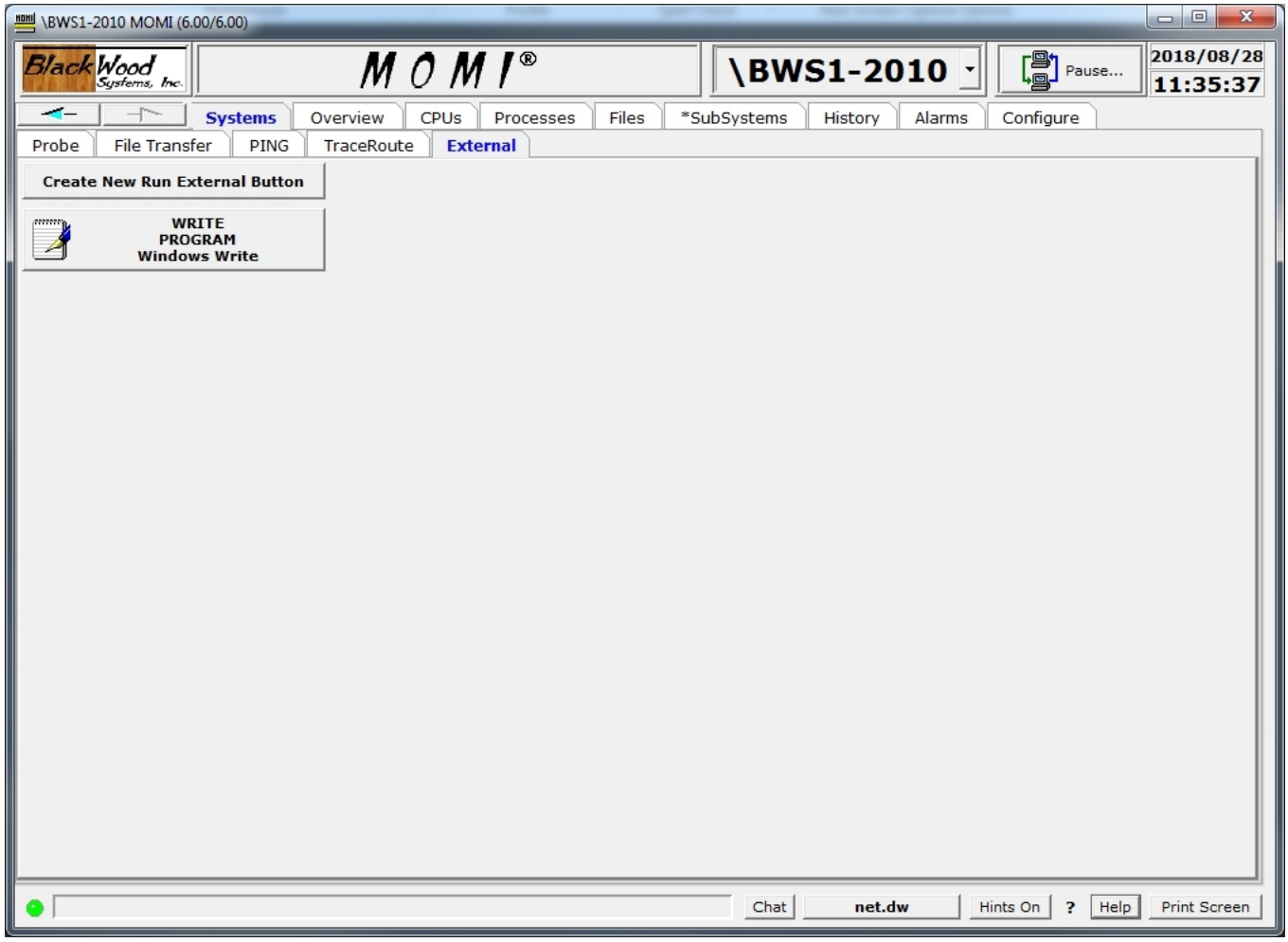
See the steps on how to create or update BWSSG.

Error accessing BWSSG or BWMOMI object file creation timestamp (check file security)

This error indicates that BWSSG is present but cannot be accessed.

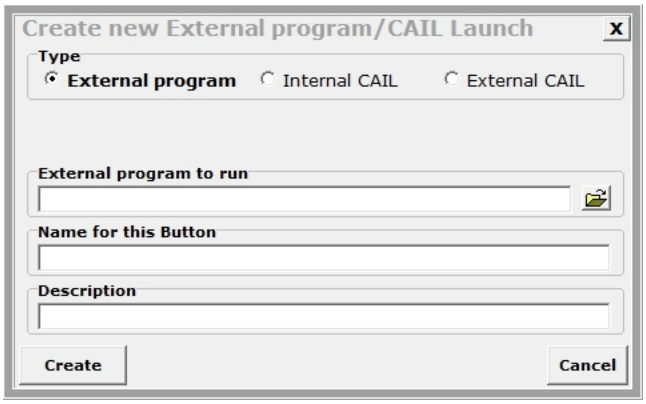
Check the file security. This file is located in the same subvolume as the MOMI server.

Systems / External



The Systems / External screen provides the ability to define and launch a process external to the MOMI Client providing access to other applications.

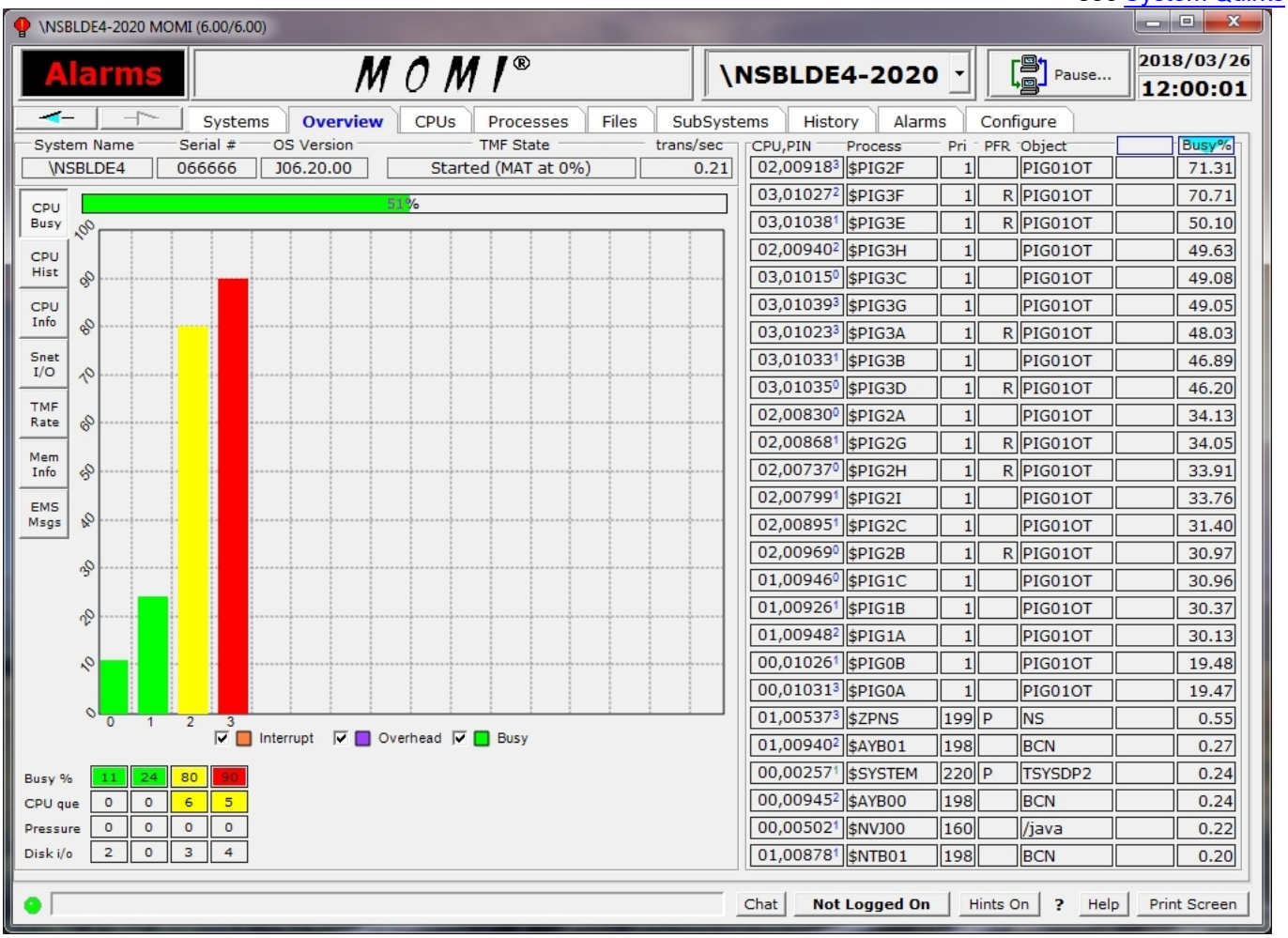
An entry is created pushing **Create New Run External Button**. In the pop-up window enter the requested information and **Create**. The new button created may then be pressed to launch the application. Right-click on the created emulator button to delete.


Create new External program/CAIL Launch
Type
External program
Internal CAIL
External CAIL
External program to run
Name for this Button
Description
Create
Cancel

Overview

Main Overview

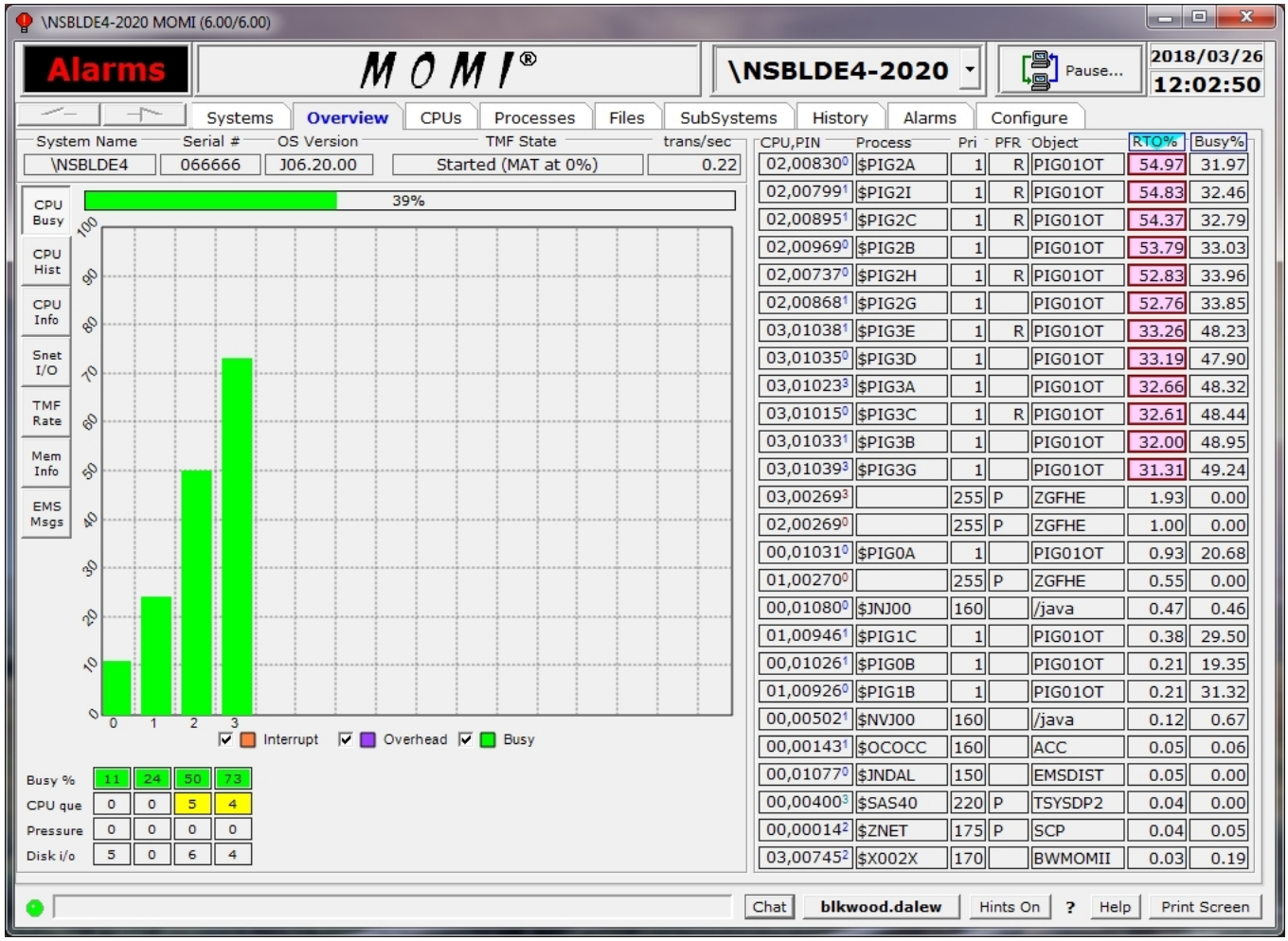
see System Quirks



The Main Overview screen provides the launch point for general information about the health of the System.

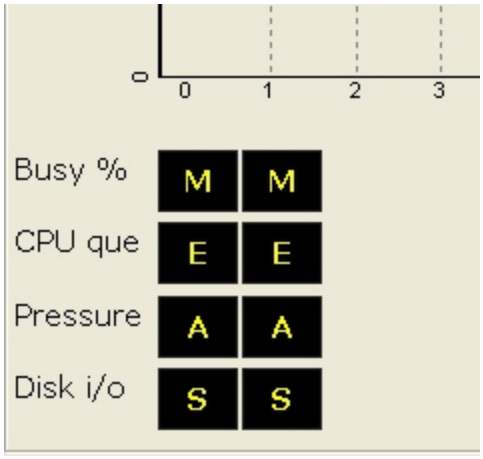
The top left side of the screen contains the System name, System serial number and operating system version. The next two fields provide the state of TMF and the current transactions per second. The TMF State displays the percent utilization of the audit trail (largest value is displayed if TMF is configured with auxiliary audit trails).

The fields in the lower left handle corner Busy %, CPU que, Pressure and Disk I/O display the Overall CPU busy, average CPU queue depth, memory pressure (a value from 0 to 7 where 7 is the highest) and physical disk I/Os per second respectively. Additionally, these fields are used to relate special information as described below.

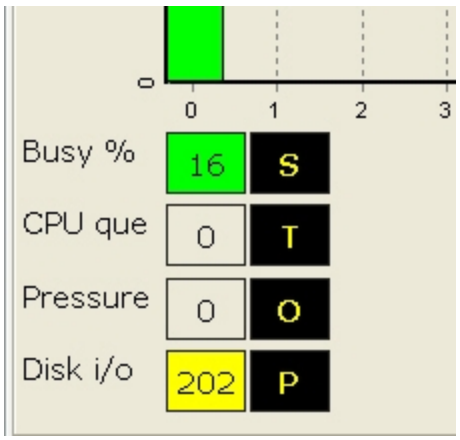


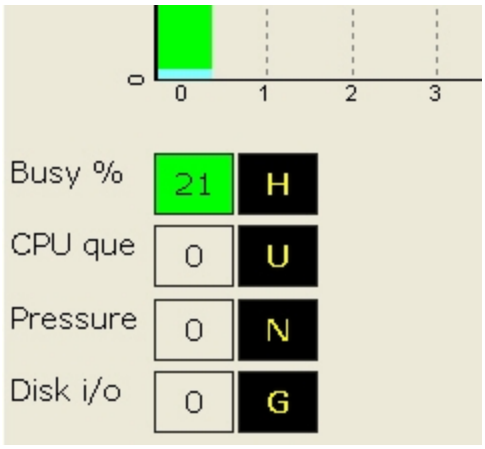
Information for the right side of the screen normally displays process and sorted by process busy percent. Right-click on the heading of the blank column to display process $RECEIVE Queue Length, the process Ready Time Only (RTO is the process Ready Time less Process Busy Time) or blank to restore an empty column. The selection displays and sorts by the chosen value. Click on the Busy% column to sort by process busy percent.

If MEASURE is not started, **MEAS** is reported in each affected processor. Use MEASCOM to start the MEASURE subsystem (for the system or a particular processor):

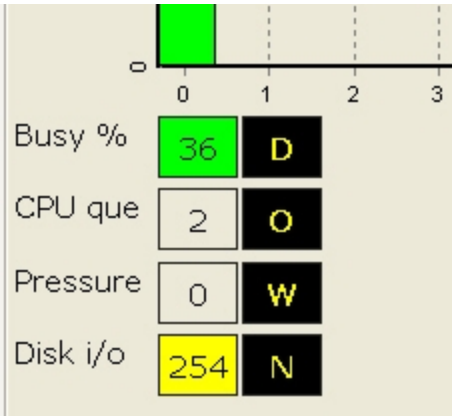


If a MOMI CPU collector process does not report in a timely manner, typically after 30 seconds, **STOP** is reported. MOMI then issues a Guardian stop on the collector process and the process is automatically restarted by MOMI (the automatic restart occurs approximately 1 minute after the **STOP** is displayed). Technically, the processor is marked as **HUNG** but this is usually not visible if the CPU collector process stops as directed:

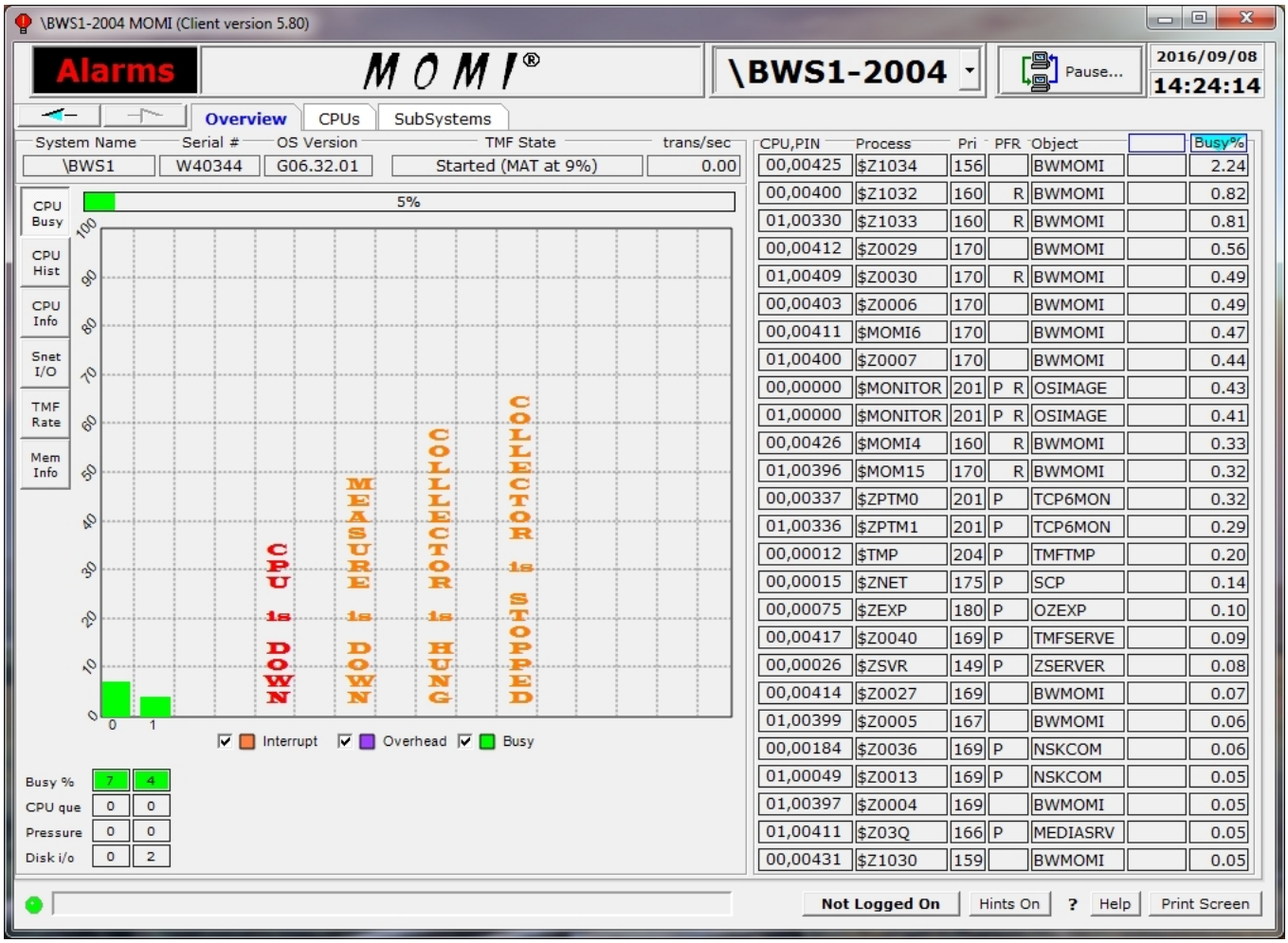




In the event a CPU fails, **DOWN** is reported for the affected CPU. MOMI automatically restarts a collector process approximately 1 minute after the CPU is reloaded:

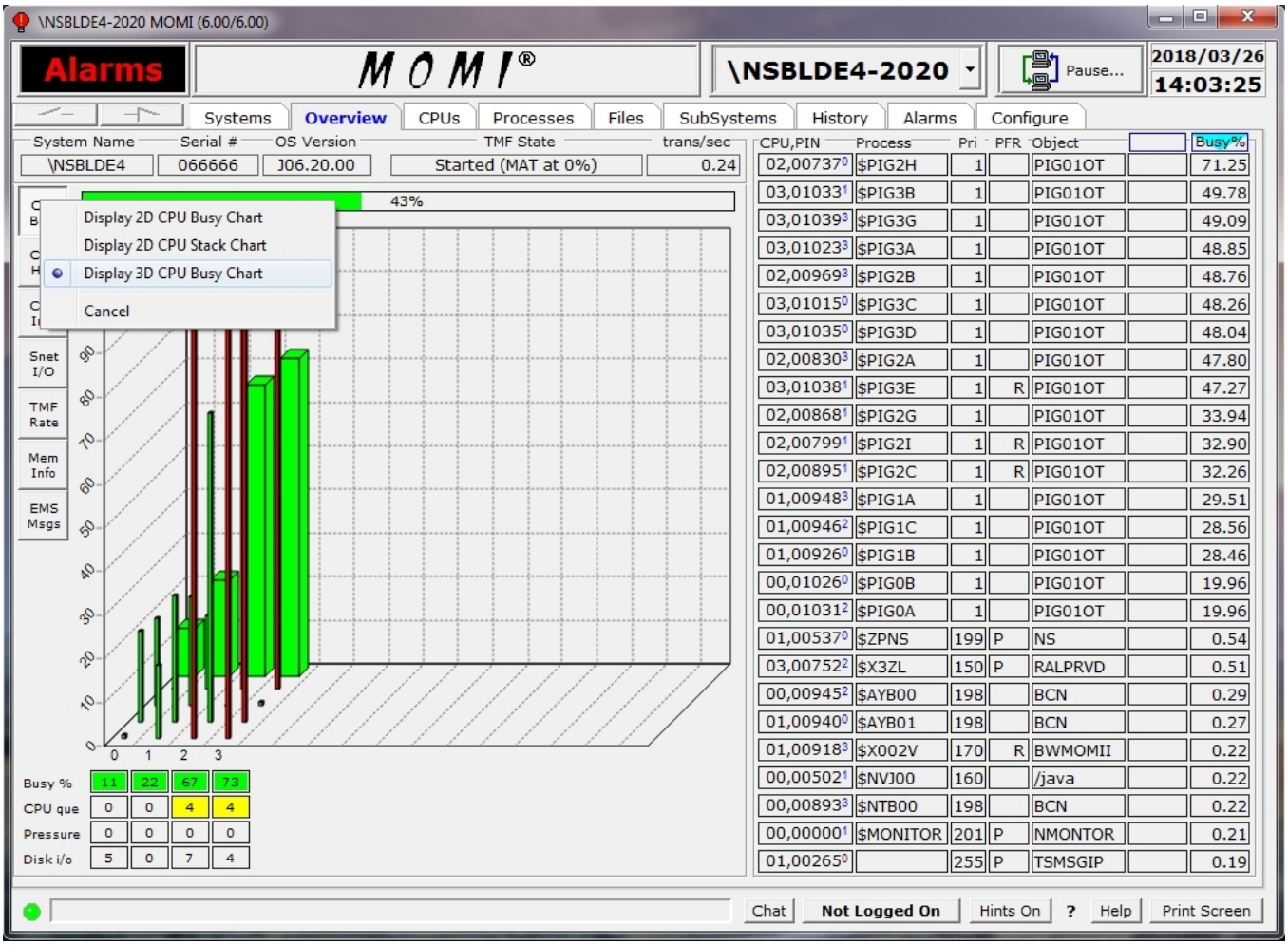


Later versions of MOMI display a more prominent CPU / MOMI status (the text below are samples, real text appears over the affected CPU).



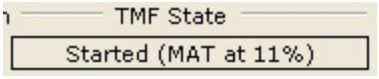
Additional CPU busy information is available for NonStop Blade and later Systems. Right-click on the CPU Busy button on the left hand side and select **Display 3D CPU Busy Chart**. This setting enables the display of individual IPU's (small narrow bar) along with the overall process busy (wide

bar). Note that on the right side of the screen that it is possible for an individual process to be at (or near 100%) and yet the overall CPU busy is nowhere near 100%. A processes percent busy is based on its activity within an individual IPU as it may not span multiple IPUs. So, for example, if a CPU has four IPUs and a single process is running at 100% the total CPU busy reported is 25% (yes, it does take some getting used to).

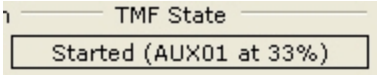


TMF State reports the general status of TMF (Started, Stopped, etc...) along with overflow and other error indicators. The percentage full (i.e. in use) of the largest of the Master (MAT) or Auxiliary (AUXnn) audit trails is also reported in parentheses for most states. Generally speaking, the "worst" state or condition is selected for display.

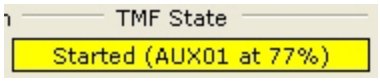
Below are some sample TMF States:



Master audit trail at 11% full



Auxiliary audit trail 1 at 33% full

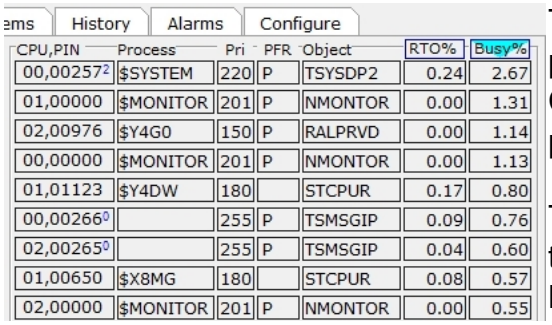


Auxiliary audit trail 1 at 77% full

Auxiliary audit trail 1 cannot rollover (out of space and no overflow available)

Auxiliary audit trail 1 using overflow

Begin transactions disabled within TMF



The right side of the screen shows the busiest or top processes on the entire System. The columns listed are the CPU, PIN [IPU], process name (blank if unnamed), current priority, PFR, Object file name, RTO% and Busy Percent.

The **Busy%** column is the percentage of total CPU busy for the process during the last monitored interval (by default MOMI scans the CPUs every 10 seconds). Note that for multiple IPU CPUs the percent busy represents the single IPU.

The **RTO%** column is a user selectable column that may display nothing, $RECEIVE Queue Time (average queue length), or Ready Time Only.

The screen may be sorted by clicking in either the RTO% or Busy% column.

The **Object** column displays the last portion of a Guardian or OSS executable file name

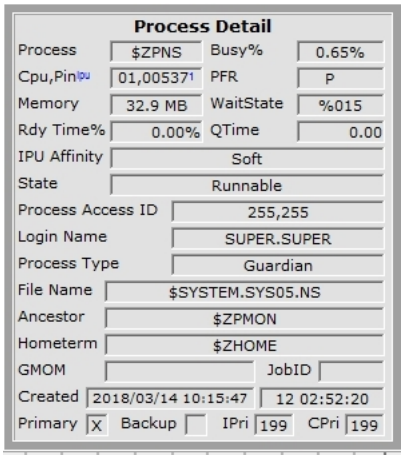
The **PFR** column indicate with single letter descriptors if the process is P - Privileged , F - page Faulting or R - on the Ready list. A process is Privileged if it has been granted authority to run without security limitations (many System processes are Privileged). A process needing memory space is page Faulting during the time it takes to free or swap needed memory pages. A process is on the Ready list when it is waiting for CPU execution. Additionally, a process can be on the Ready list while waiting on a page Fault.

The **Pri** column is the current process priority field. The surrounding box is yellow if the operating system has reduced the process priority and the process is a disk. A red color is used for any other process with a reduced priority. Disk processes will assume the priority of the requesting process while waiting for executing time. It is normal to see a disk process running at different priorities. User processes have their priority lowered by the System if they are determined to be CPU bound (i.e. taking a great deal of CPU resources).

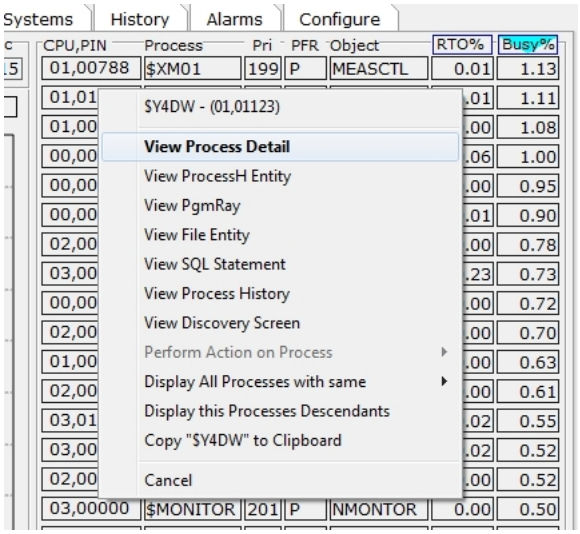
The **Process** column displays the process $Name, a system special name (such as $MONITOR), or nothing to represent an unnamed process.

The **CPU,PIN[IPU]** column shows the processor number, process identification number and optionally the CPU core number (i.e. IPU) where the process is running. The IPU is displayed, by default, if it has been specifically assigned by the operating system or by the user. If the IPU is not displayed it is assumed to be floating or in other words the process is dispatched in any available CPU core.

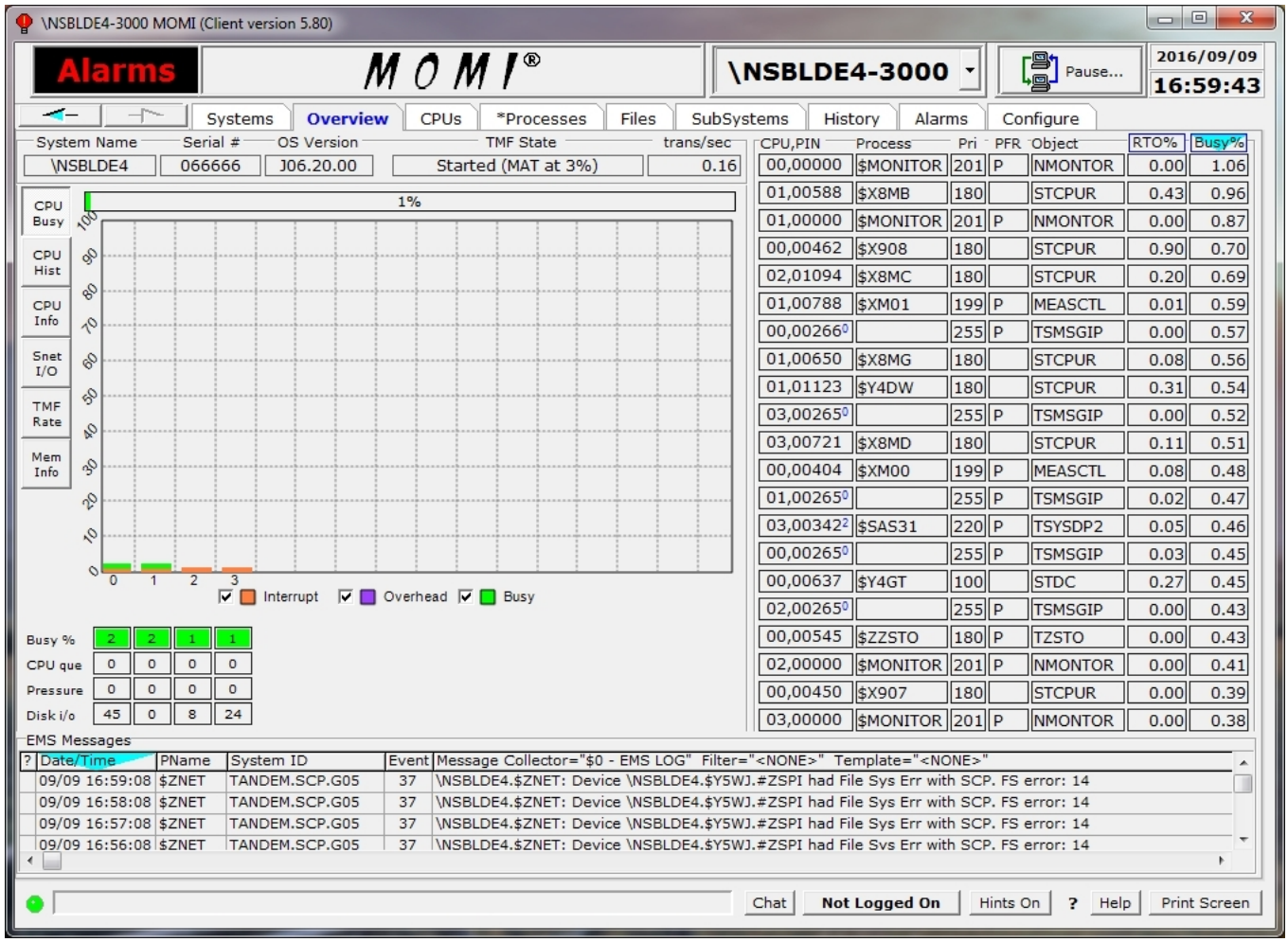
Placing the mouse over a CPU,PIN displays a pop-up window presenting additional information for that process. Below is a sample image:



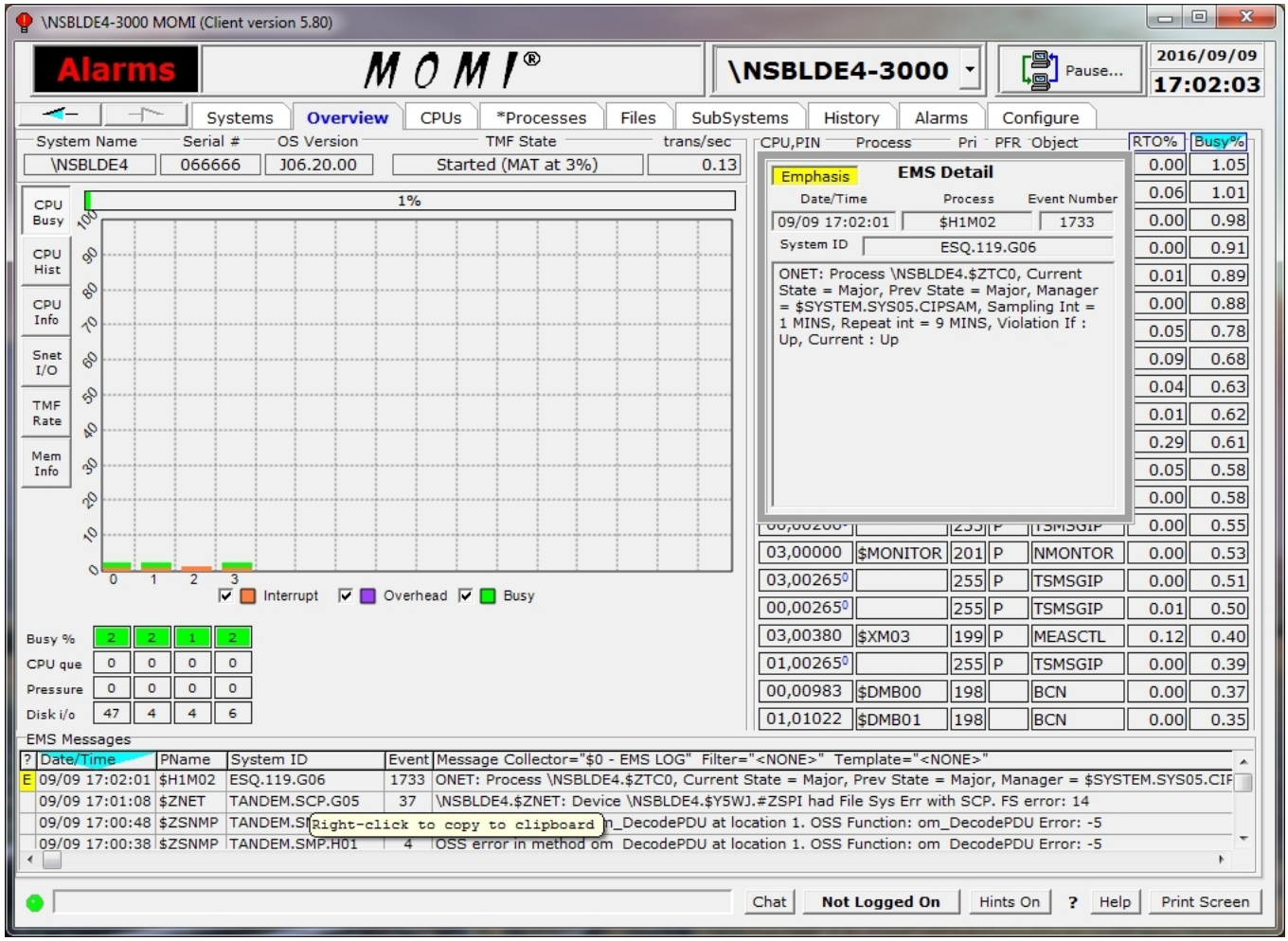
A right click on the CPU,PIN or Process field will bring up a menu allowing the User to select a variety of jumps or actions using the selected process (the actions available vary depending on the process):



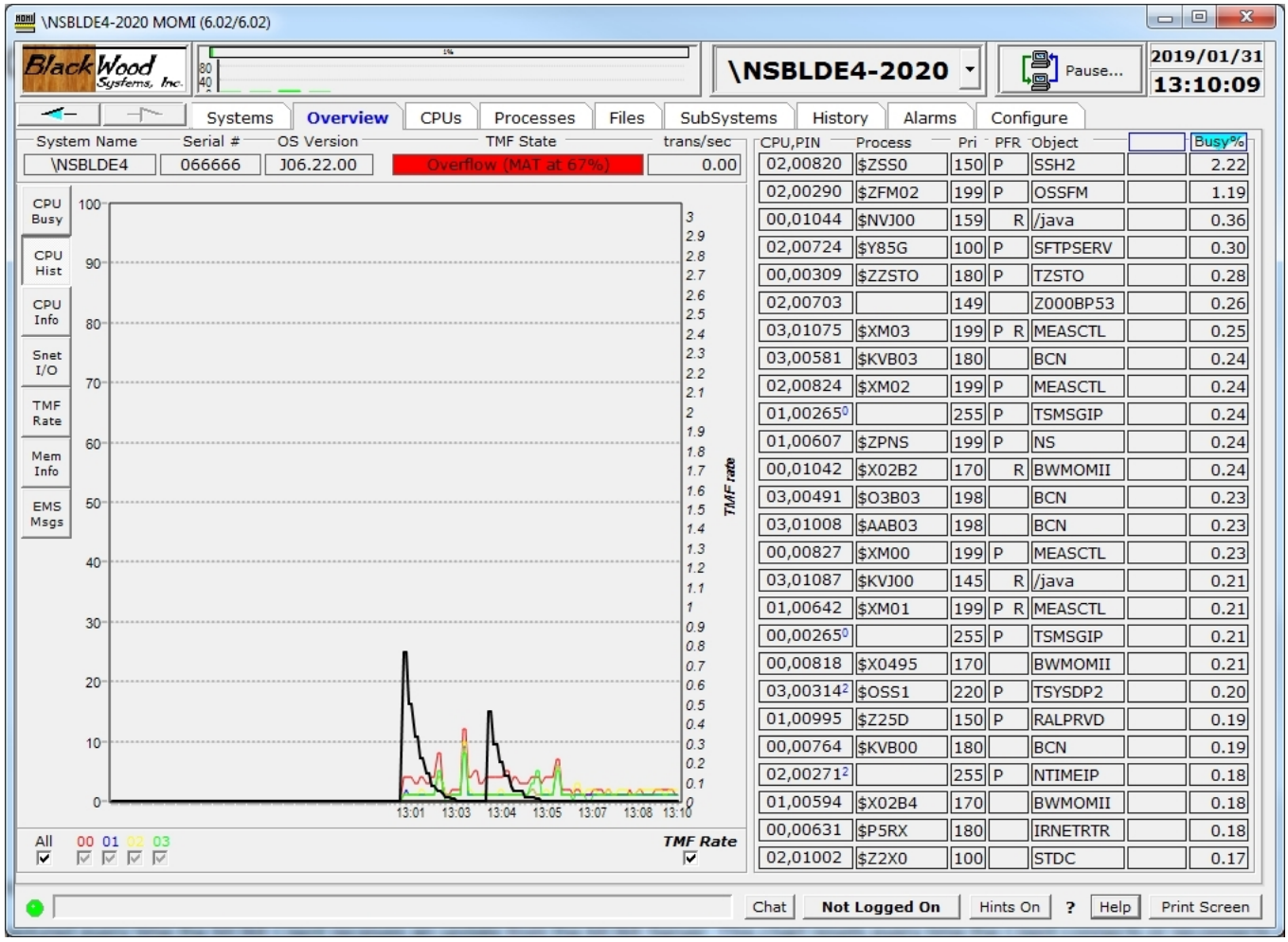
The Overview screen may also be configured, on the page Settings / Single Screen to an alternate format that displays EMS information along the bottom on the screen. Note that in this mode the EMS Msgs button along the left side is not longer present. To change the text size of the EMS messages, right-click on any EMS column heading and select Display Grid Tools... Note that space is limited on this screen so a large font selection will not normally work well.



Placing the mouse over an EMS message display a pop-up detail.


\NSBLDE4-3000 MOMI (Client version 5.80)
Alarms
MOMI®
\NSBLDE4-3000
Pause...
2016/09/09
17:02:03
Systems
Overview
CPUs
*Processes
Files
SubSystems
History
Alarms
Configure
System Name
Serial #
OS Version
TMF State
trans/sec
\NSBLDE4
066666
J06.20.00
Started (MAT at 3%)
0.13
CPU Busy
CPU Hist
CPU Info
Snet I/O
TMF Rate
Mem Info
1%
Interrupt
Overhead
Busy
Busy %
CPU que
Pressure
Disk i/o
Emphasis
EMS Detail
Date/Time
Process
Event Number
09/09 17:02:01
$H1M02
1733
System ID
ESQ.119.G06
ONET: Process \NSBLDE4.$ZTC0, Current State = Major, Prev State = Major, Manager = $SYSTEM.SYS05.CIPSAM, Sampling Int = 1 MINS, Repeat int = 9 MINS, Violation If : Up, Current : Up
CPU,PIN
Process
Pri
PFR
Object
RTO%
Busy%
03,00000 $MONITOR 201 P NMONTOR 0.00 0.53
03,00265 255 P TSMSGIP 0.00 0.51
00,00265 255 P TSMSGIP 0.01 0.50
03,00380 $XM03 199 P MEASCTL 0.12 0.40
01,00265 255 P TSMSGIP 0.00 0.39
00,00983 $DMB00 198 BCN 0.00 0.37
01,01022 $DMB01 198 BCN 0.00 0.35
EMS Messages
Date/Time
PName
System ID
Event
Message Collector="$0 - EMS LOG" Filter="<NONE>" Template="<NONE>"
09/09 17:02:01 $H1M02 ESQ.119.G06 1733 ONET: Process \NSBLDE4.$ZTC0, Current State = Major, Prev State = Major, Manager = $SYSTEM.SYS05.CIP
09/09 17:01:08 $ZNET TANDEM.SCP.G05 37 \NSBLDE4.$ZNET: Device \NSBLDE4.$Y5WJ.#ZSPI had File Sys Err with SCP. FS error: 14
09/09 17:00:48 $ZSNMP TANDEM.SM Right-click to copy to clipboard n_DecodePDU at location 1. OSS Function: om_DecodePDU Error: -5
09/09 17:00:38 $ZSNMP TANDEM.SMP.H01 4 OSS error in method om_DecodePDU at location 1. OSS Function: om_DecodePDU Error: -5
Chat
Not Logged On
Hints On
?
Help
Print Screen

Overview CPU History

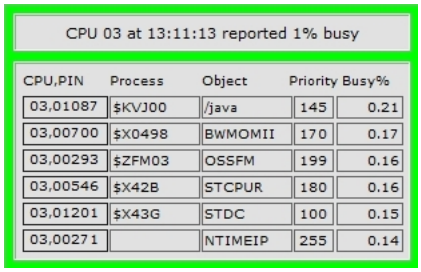


The Overview CPU History screen charts the busy history of the CPU's on the System with the TMF transaction rate. The last 200 values are retained.

Data for this chart is captured every time the MOMI Client receives an update from the MOMI Server. This chart resets every time the Client connects or reconnects unless the system is being probed (from any sub-screen) on Systems / Probe or Expand / Diagram.

Selection boxes near the bottom of the screen may be unchecked to removed CPUs or the TMF Rate from the display.

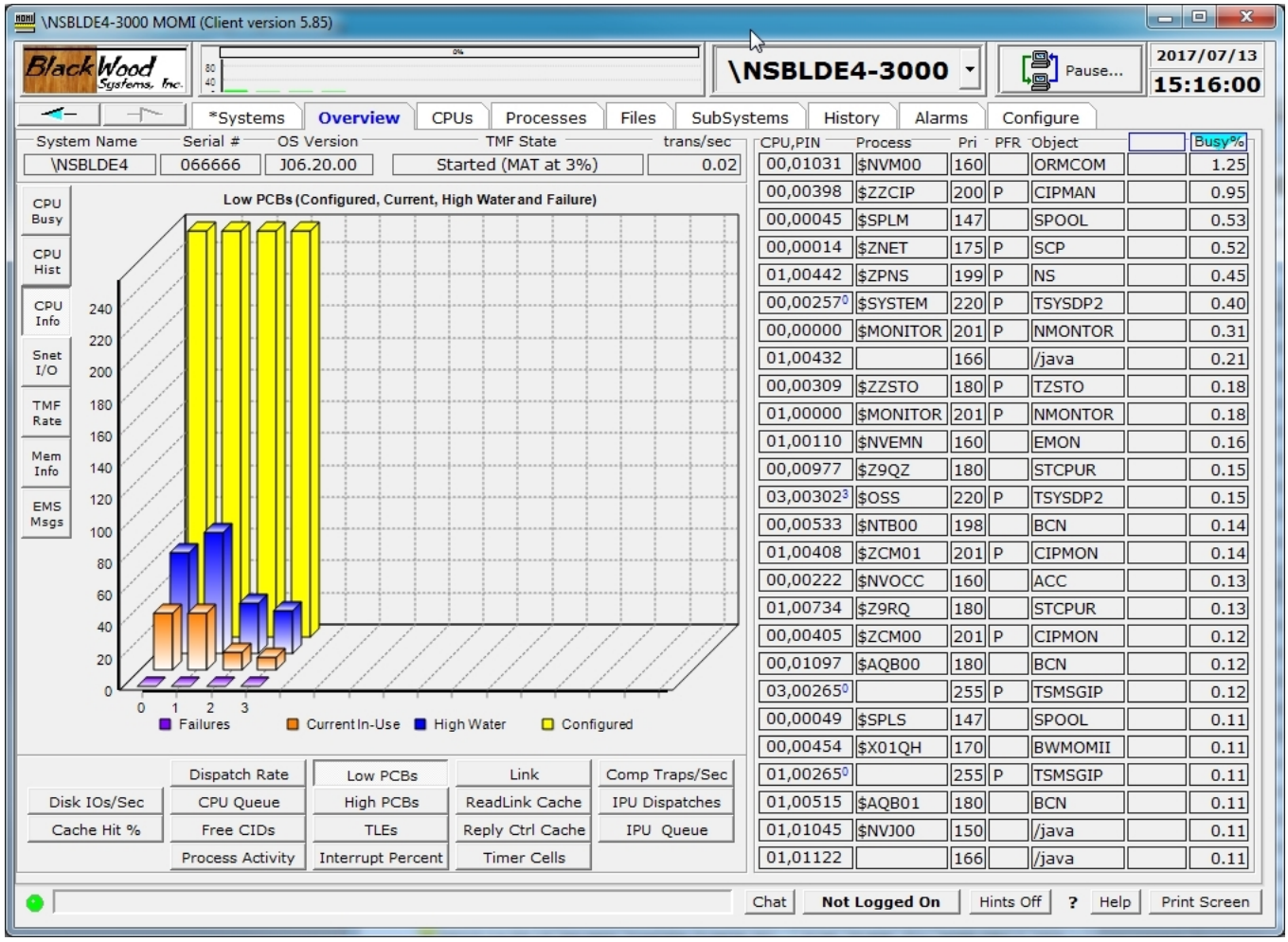
Place the mouse over a chart line and click to display the top few processes at that point in time.



CPU 03 at 13:11:13 reported 1% busy

CPU,PIN	Process	Object	Priority	Busy%
03,01087	$KVJ00	/java	145	0.21
03,00700	$X0498	BWMOMII	170	0.17
03,00293	$ZFM03	OSSFM	199	0.16
03,00546	$X42B	STCPUR	180	0.16
03,01201	$X43G	STDC	100	0.15
03,00271		NTIMEIP	255	0.14

Overview CPU Info



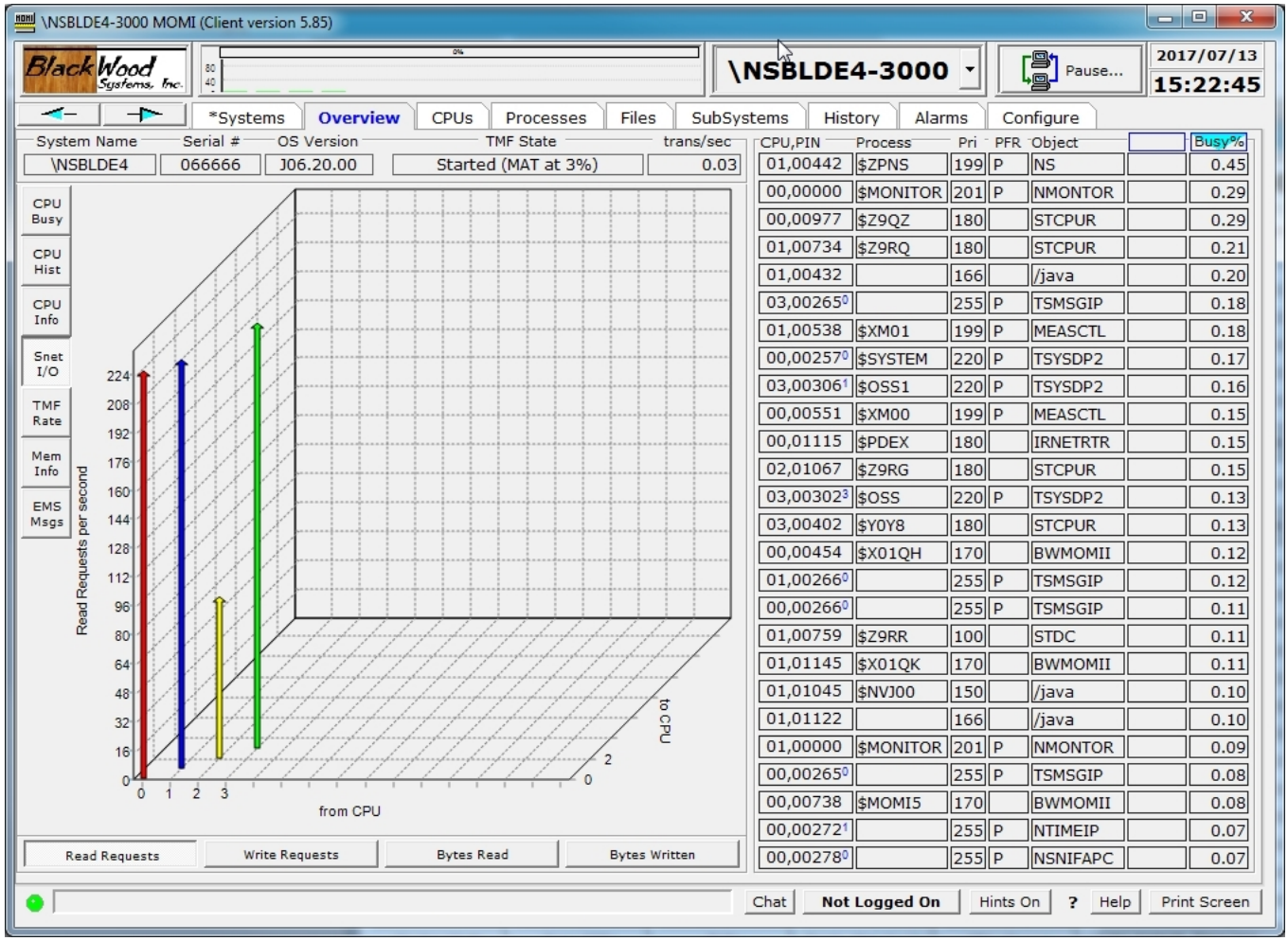
The Overview CPU Info charts various values pertaining to a CPU.

Place the mouse over the Bar to see the actual value. Several bars are hot spots jump for additional information about a particular value. For example, pressing the **Comp Traps/Sec** button and clicking on a bar will jump to the **Processes / Process Entity** screen and configure a process entity measurement to display the highest generators of Comp Traps (**Computational Traps**[1]) in the particular CPU.

---

[1]Also known as Computational Traps is the situation when memory is accessed and the data is not "well aligned". Well aligned data is located at a multiple of the size of the field. For example, a 32-bit value is four bytes, so the address of the value should be evenly divisible by four otherwise a Comp Trap occurs on access. A Comp Trap indicates that the system had to perform extra work in order to process the exception. Generally speaking, 16-bit programs always cause Comp Traps and 32-bit programs ideally have zero to very few Comp Traps.

Overview Snet I/O

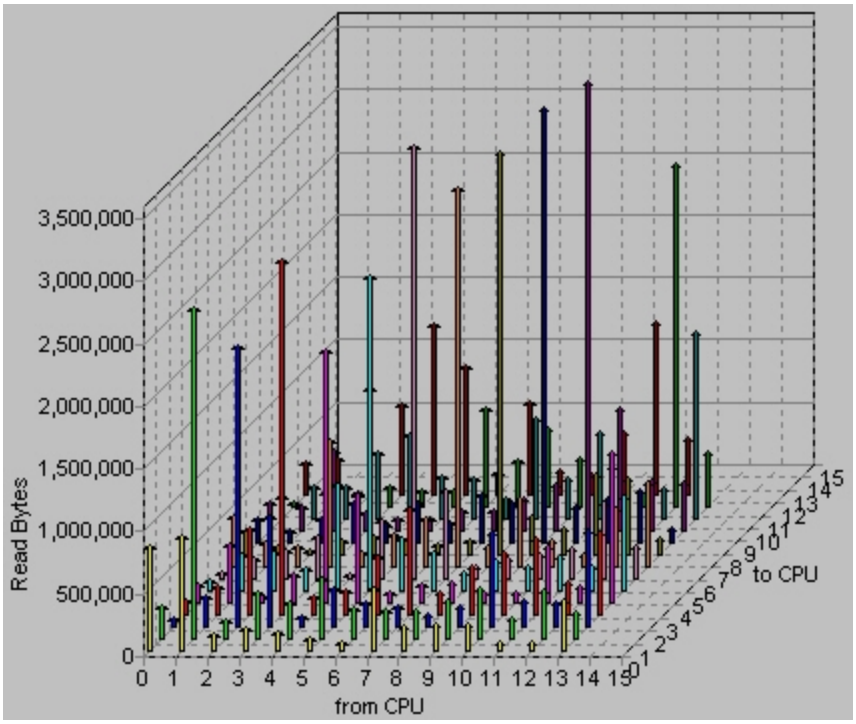


The Overview Snet I/O screen displays the traffic in and out of the ServerNet interconnects at the CPU level. Certain traffic is counted twice, so the chart is just a general idea of the traffic rate and distribution.

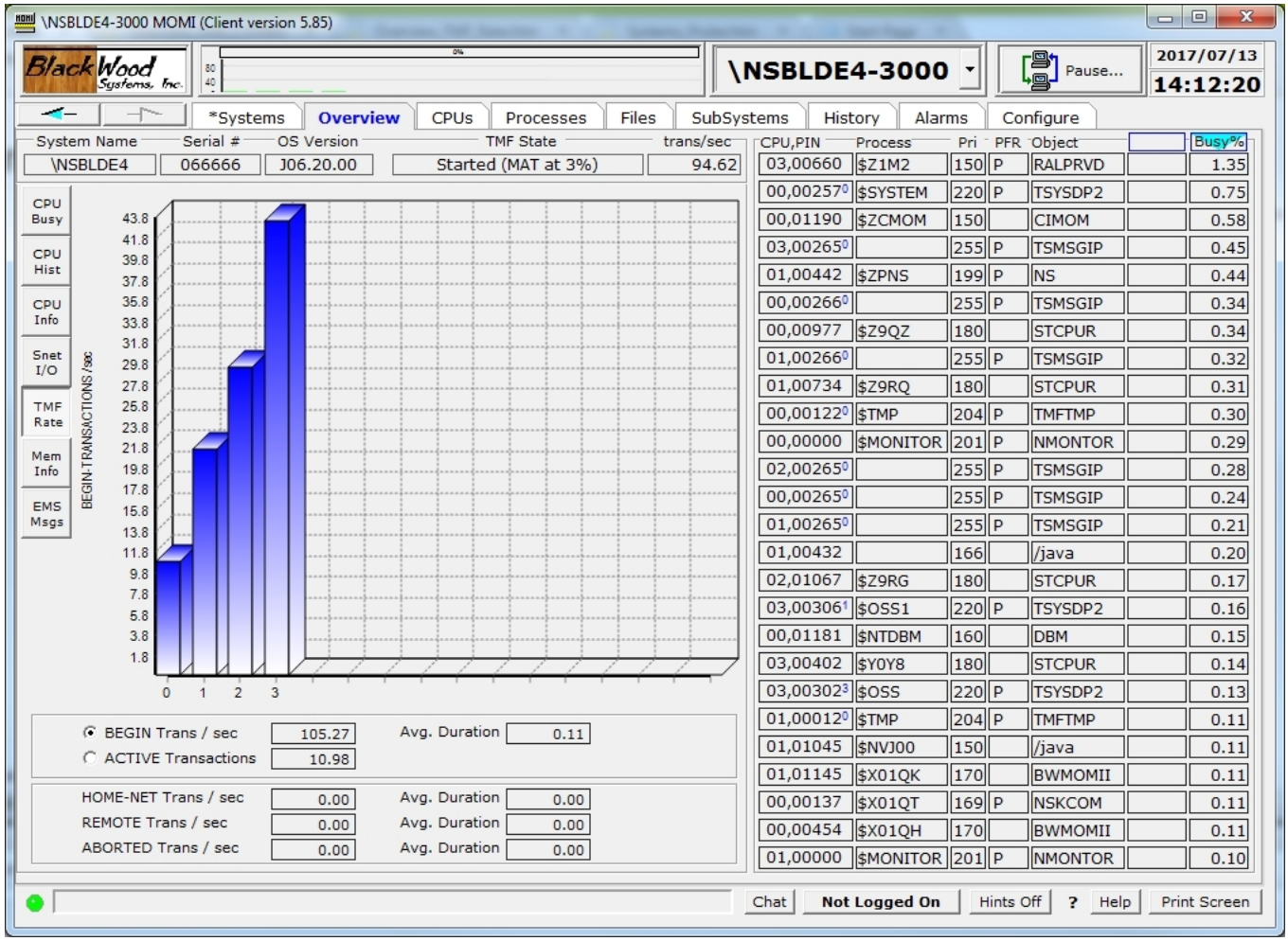
The 3D chart presents data transfers from all CPUs to all CPUs. On older ServerNet systems the chart appears more 3-dimensional than on newer systems. Newer systems seem to only display a 3-dimensional effect under very heavy load.

The chart presents the Read / Write Requests and the Read / Write Byte rates.

Below is a portion of the same screen from an older system showing the 3-dimensional affect.


Read Bytes
3,500,000
3,000,000
2,500,000
2,000,000
1,500,000
1,000,000
500,000
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
from CPU
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
to CPU

Overview TMF Rate

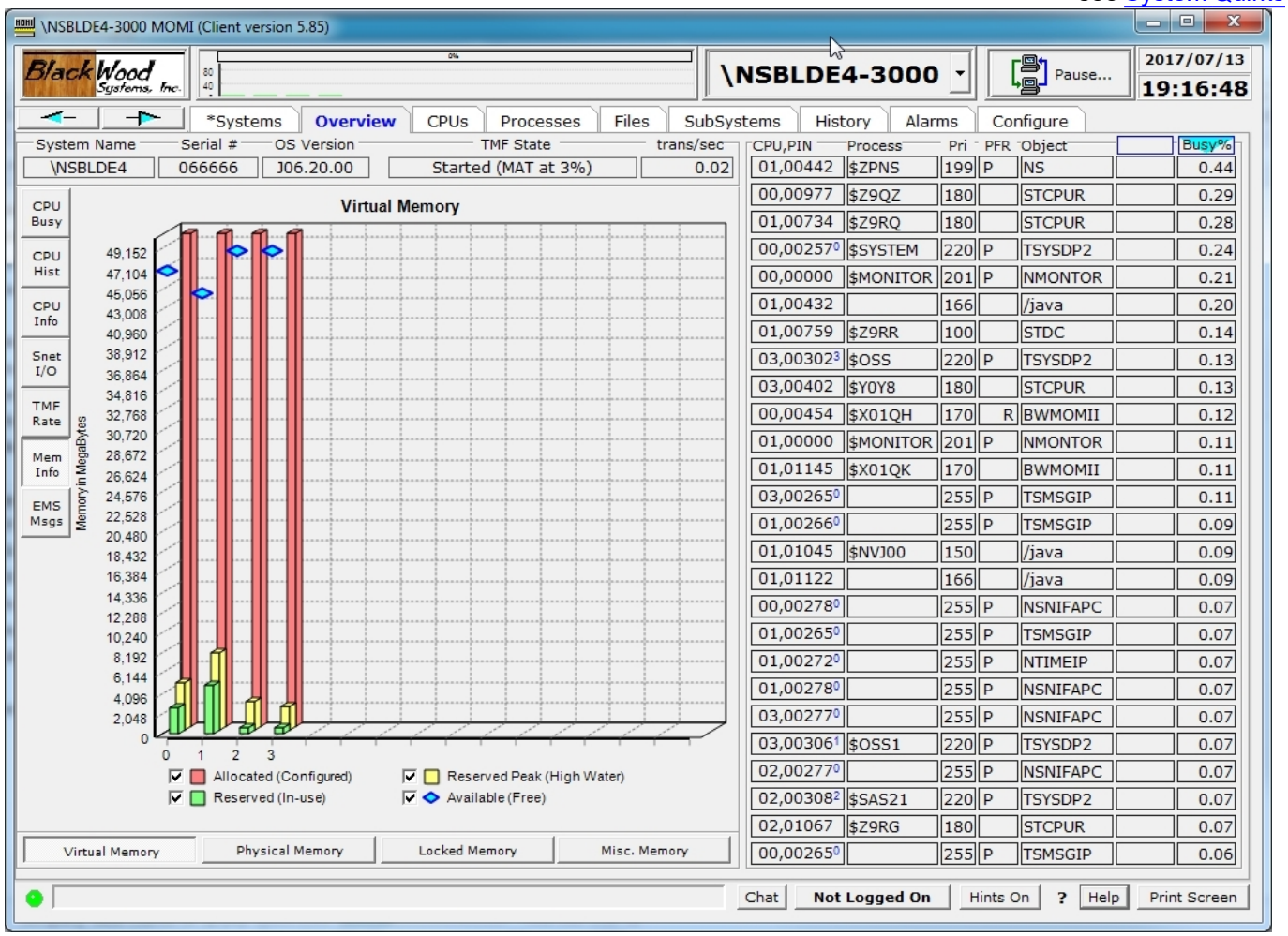


The Overview TMF rate chart presents the TMF transaction rate at a per CPU level.

Two options are available. Plotting data at **BEGIN** shows the count of transactions that were started. Plotting data at **ACTIVE** shows the number of transactions that were started but not yet completed. **ACTIVE** transactions are an approximation, based on available information from MEASURE.

Overview / Mem Info

see System Quirks



The **Overview / Mem Info** screen presents information and stats about various aspects of CPU memory.

Virtual Memory is the amount of disk space available to extend physical memory when fully consumed. The Operating System encounters a shortage of free memory, it will page out or write to storage portions of memory then release it. When the Operating System detects a process encountered memory paged out, the process is transparently suspended, its memory returned (i.e. read) from storage, and re-activated. This situation is referred to as a page-fault. Typically, a NonStop System would not want to encounter frequent page faults as they result in waits on storage I/O. However, the alternative would be the Operating System expectantly terminating a process or denying a request for memory.

Virtual memory is configured via the utility NSKCOM. A "rule of thumb", the amount of virtual memory is at least 2.5 times physical memory. This is configured on a per CPU basis and generally places the swap file on a storage volume DP2 process also residing in the same CPU.

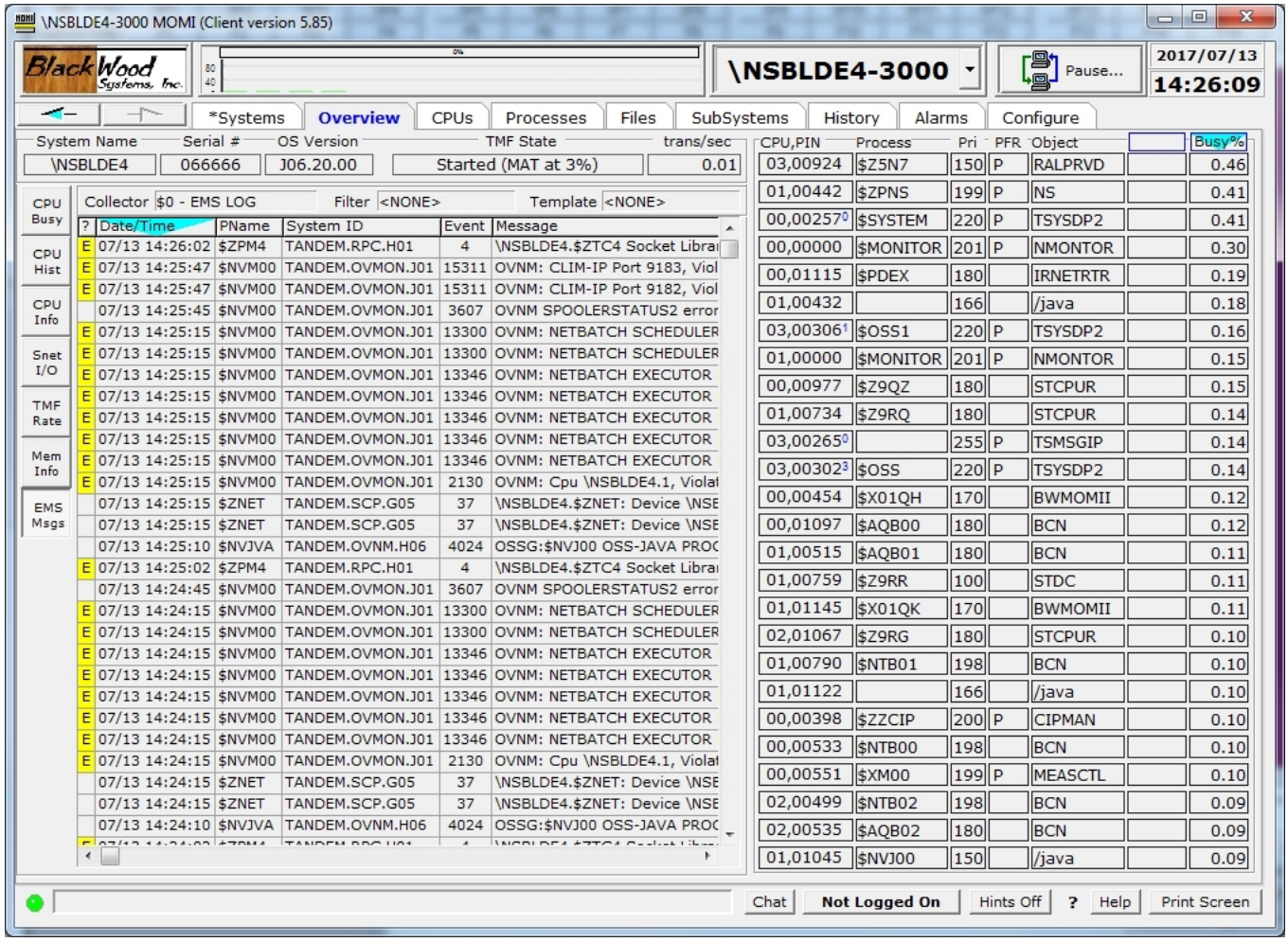
Physical Memory is the memory actually present in the CPU. For example, you have a 32 Gigabyte processor.

Locked Memory is memory that cannot be swapped to disk. Certain operations, such as I/O to/from the CPU, must be physically present for write or read operations. A certain amount of memory is always locked by the operating system at start up and the total mount varies as processes start, stop and I/O activity is performed.

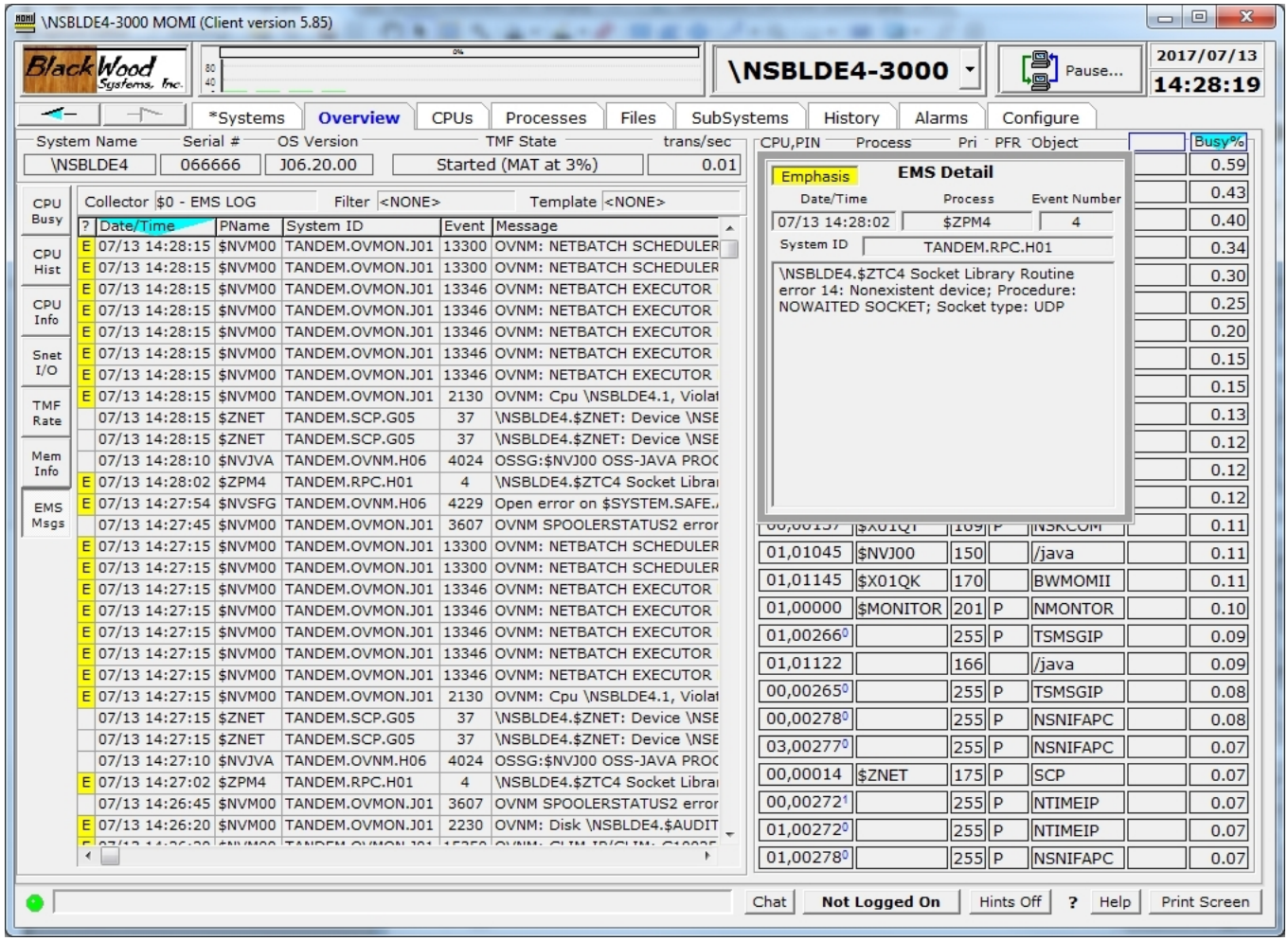
Misc. Memory is a 'catch area' for other useful memory counters.

- Unsponsored Memory is memory that is temporarily not owned by any process but must remain allocated.
- Memory Pressure is a value from 0 to 7 where 0 is low. It is an indicator of the demand for memory. Systems with plenty of memory have a pressure of 0.
- A page fault is the situation where a memory page is needed but not present. This occurs when a process is first started and continues until all needed memory is accessed at least once. Additionally, a page fault can occur when memory that was previously accessed is no longer present, perhaps due to another process needing memory.
- Memory Queue is an indication of the number of processes waiting for memory. Newer systems do not seem to have a general memory queue. The servicing of page faults is now handled within the context of a user process (via system code in the background). On older machines, a memory queue was a second ready list because a process must first have memory accessible before it can wait to execute.

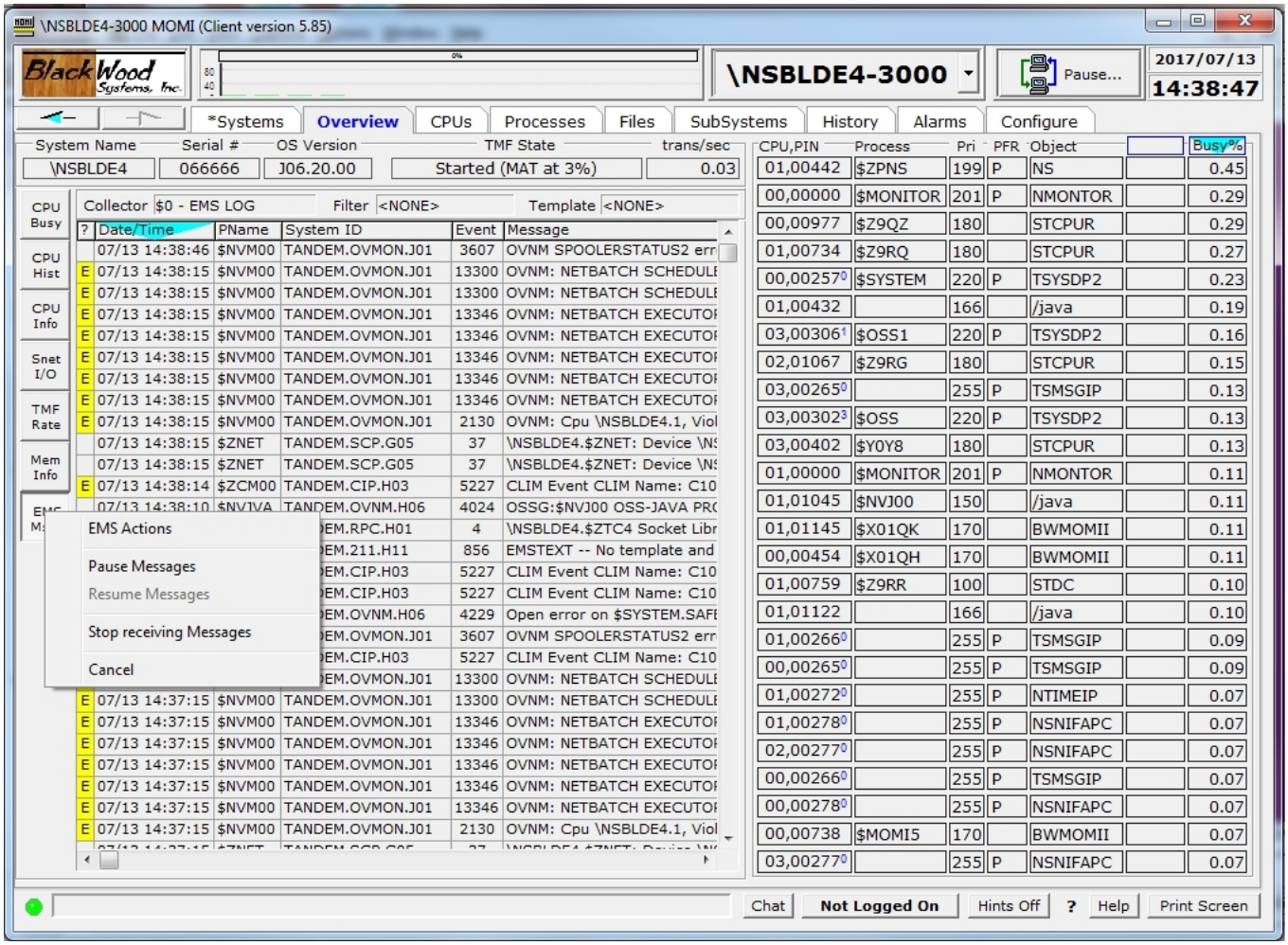
Overview EMS Msgs



The Overview EMS Msgs screen presents EMS log messages from the system log $0.



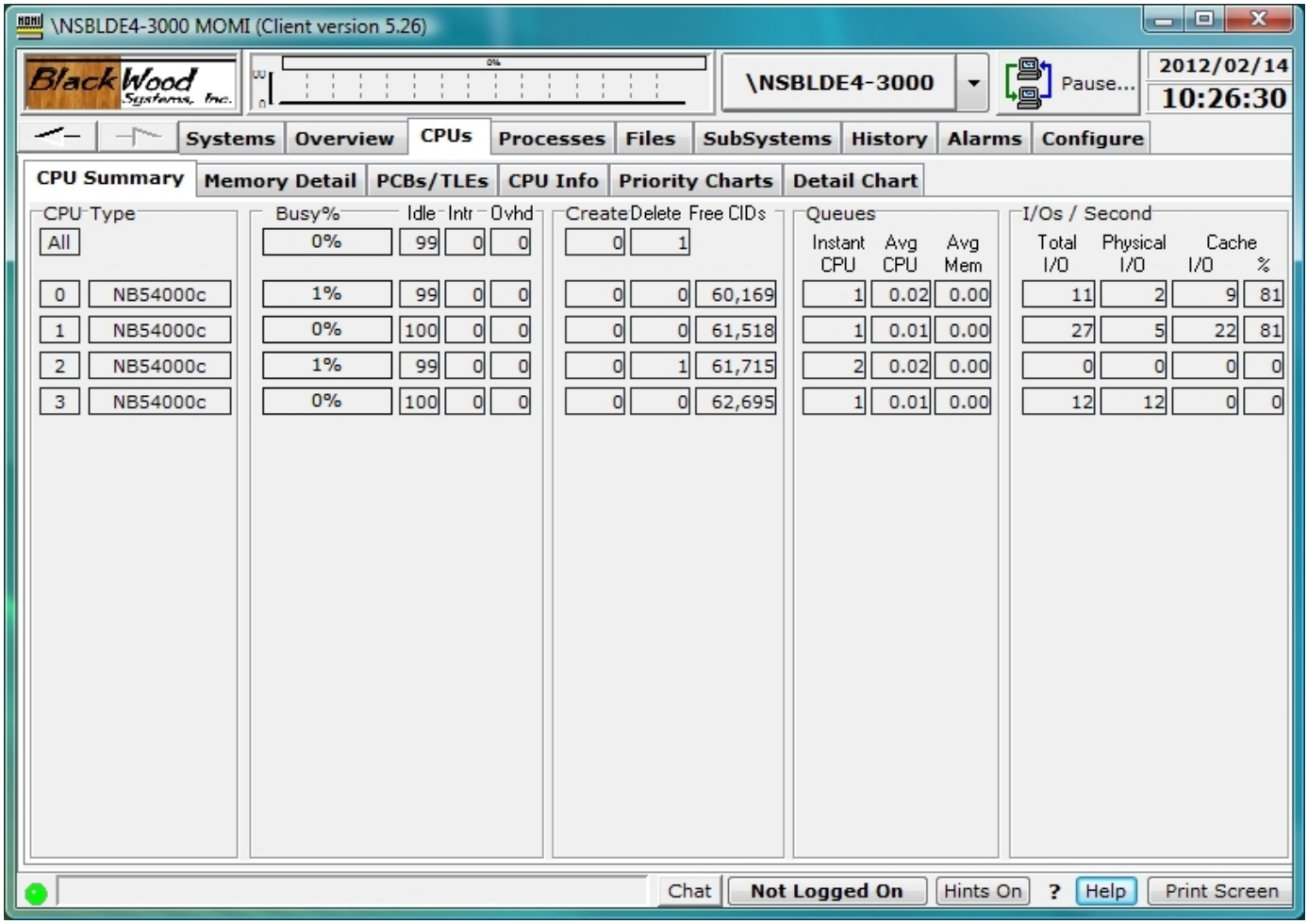
Place the mouse over an EMS line and a pop-up window displays the entire message.



A right-click on the EMS Msgs button allows actions to pause, resume or stop EMS message processing on this screen.

CPUs

CPUs / CPU Summary



The CPUs / CPU Summary screen provides high level detail on the activity of each processor in the System.

Busy % breaks down the total CPU busy value into Idle, Interrupts, and Overhead. Idle is the percent of time the processor did nothing. Interrupts is the percent of time the processor serviced low-level disruptions to normal processing such as an I/O completion. Overhead is the percent of time the processor spent creating or deleting processes.

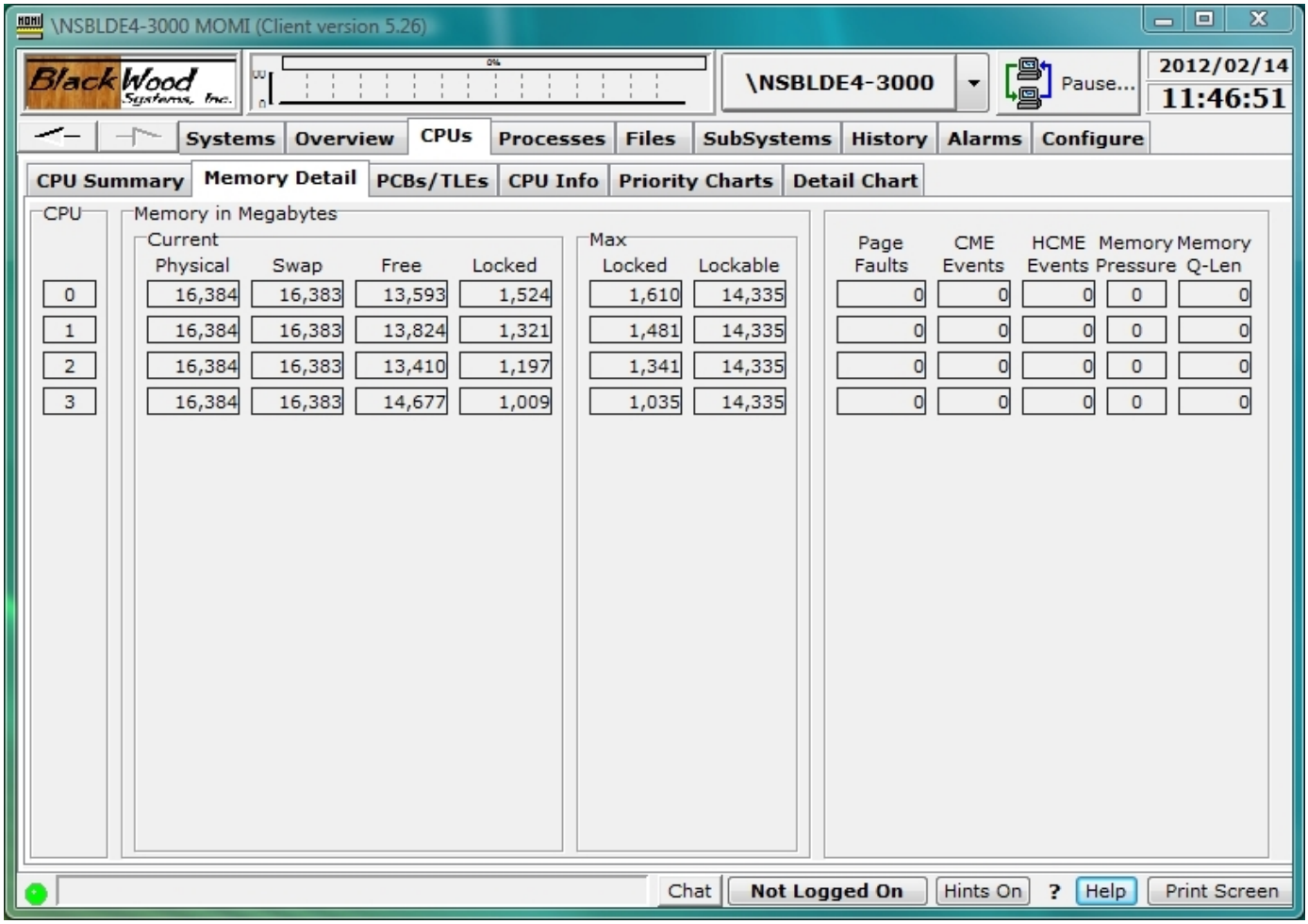
Create and **Delete** show the number of processes new to the system and removed from the system as seen by MOMI. This gives an indication as to the level of process turnover on the System. Typical Guardian-based processes usually start and stay started. A high process turnover may indicate a problem. OSS-type processes usually have a much higher turnover due to the nature of the UNIX-like environment.

Free CIDs are a MEASURE resource used to contain measurement counters. This value should never be at or near zero so that additional or transient measurements may always configure without error.

CPU Queues are reported as Instant, or what is in the queue right now, and Average, which is the queue over the Interval period. The Instant queue is a whole number of processes while the Average is a decimal value. The Average memory queue is only available (i.e. have a non-zero value) on older platforms.

I/Os / Second indicate the gross total disk I/O activity at the CPU level. The physical I/Os plus the I/Os satisfied from cache equal the total I/Os. The percent of cache hits indicates the overall effectiveness of the disk cache. High levels of I/Os with a low cache hit percentage may indicate insufficient cache allocated to disk.

CPUs / Memory Detail



The CPUs / Memory Detail screen provides the detail of the physical memory of each processor.

Physical memory refers to what was installed in the CPU, such as a 64 Gigabyte processor (note that some platforms do not provide all physical memory). **Swap** (swappable) memory is available for virtual memory and could be swapped to disk (a user process would normally occupy virtual memory space). **Free** memory is not allocated. **Locked** memory is forced to remain in the processor and may not be swapped to disk (i.e. virtual memory). **Lockable** memory is the maximum amount that can be locked. Some system processes, control blocks, I/O buffers, and dirty cache are examples of memory usually locked.

The values displayed in **Memory in Megabytes** section are originally reported by the Operating System as a number of memory pages which are then converted to Megabytes as follows:

(nbr-pages * bytes-per-page) / bytes-per-megabyte

Where -

```
bytes-per-page      =    16384
```

```
bytes-per-megabyte = 1048576
```

and may be converted to Gigabytes taking the displayed value and divide it by 1024. The value bytes-per-page is obtained from the operating system.

Page Faults occur when a process needs to access memory but it is not present. The System must either read the memory from disk or create a new page.

CME Events (Correctable Memory Errors) is the situation where a hardware check-sum algorithm indicates a memory bit was in error but the System could recover from the event. CMEs are usually transient in nature. The value reported here comes from the Guardian procedure call PROCESSOR_GETINFOLIST_.

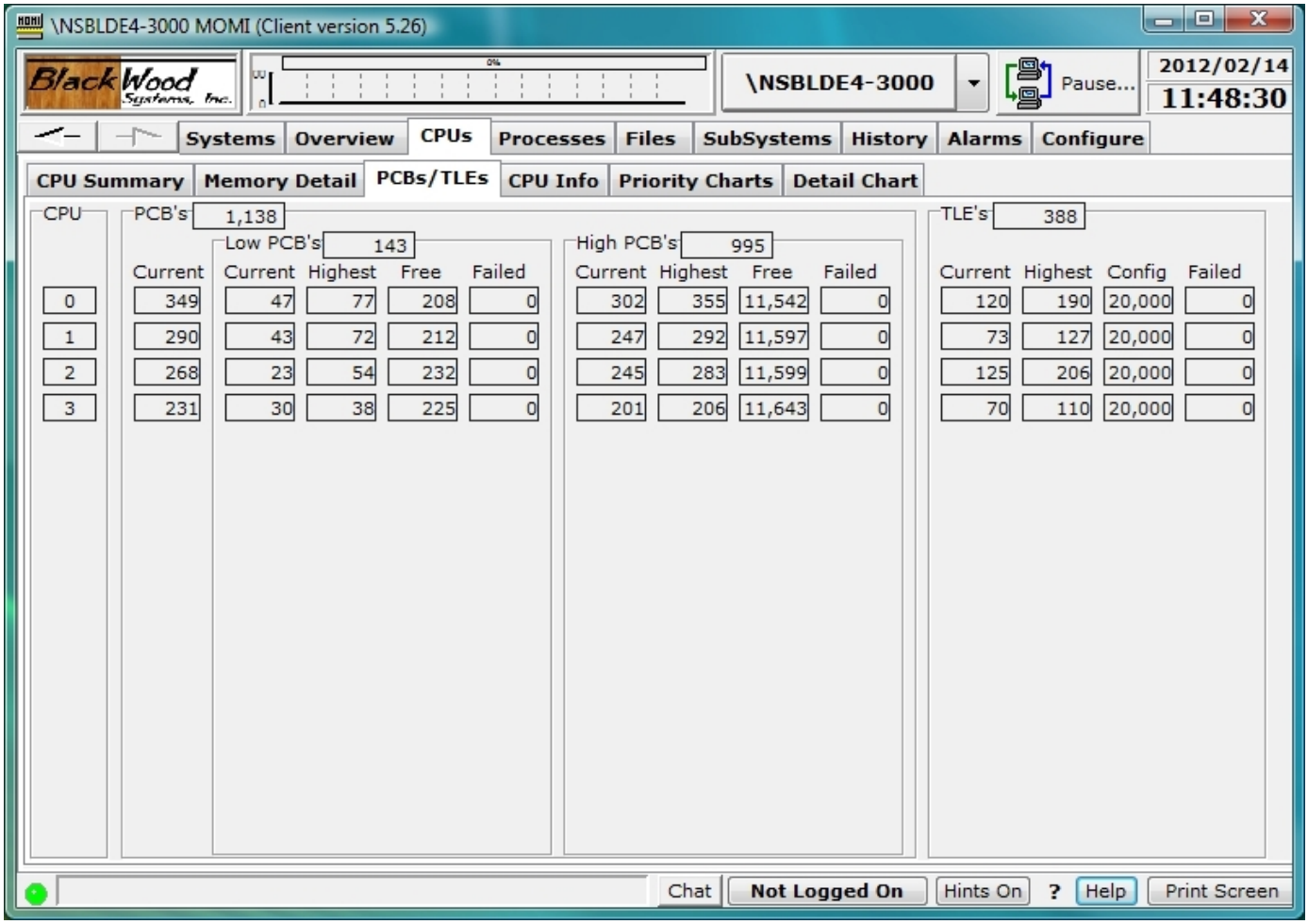
HCME Events (Hard Correctable Memory Errors) is a situation on newer NonStop platforms where a CME memory errors exist but has exceeded an internal threshold triggering for removal the memory page where the error is located. The problem does not necessarily indicate a hardware failure. The value reported here comes from the MEASURE CPU Entity.

UCME Events (Uncorrectable Memory Errors) is the situation on older NonStop platforms where a hardware check-sum algorithmindicates a memory bit(s) was in error but the System could NOT recover from the event. If the UCME located in an unused area of memory is 'locked out' from further usage. If the error is located in a User process the operating system may cause the process to abend. If the error is located in a system process or system data area it may cause the operating system to halt the CPU. The value reported here comes from the MEASURE CPU Entity.

Memory pressure is an indication of the need for memory. This value ranges from 0 to 7, with 0 being no pressure and 7 being extreme pressure. Observation of this value indicates it is an average over time that increases rather rapidly but decreases rather slowly.

Memory queue length is the number of processes waiting for memory. Newer platforms do not see a value in this field because memory service occurs within the context of the process itself.

CPUs / PCBs/TLEs



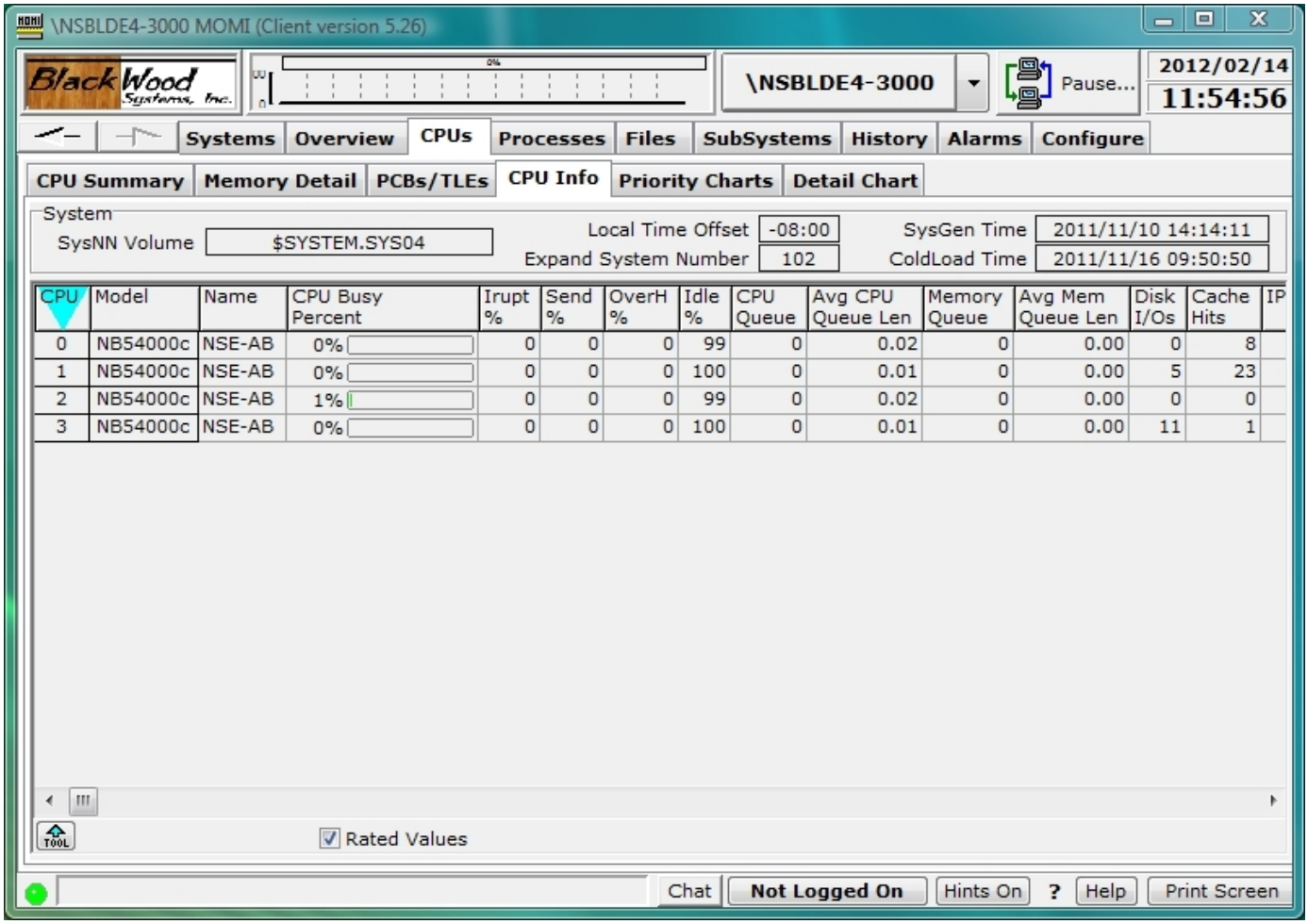
The CPUs / PCBs/TLEs screen provides details on the availability of Process Control Blocks (PCB) and Time List Elements (TLE).

Each running process on the system requires one PCB. Additionally, a process may require one or more TLEs, if it has requested notification of an elapsed amount of time.

A PCB is considered low if it is allocated within the range of 0 to 254 and is considered high if 256 or greater. Older programs that do not have the High Pin enabled flag set within the executable are limited to the low range. PCB 255 is not used.

Any failure to allocate one of these resources shows in RED.

CPUs / CPU Info



The CPUs / CPU Info screen presents an assortment of miscellaneous System and processor information.

SysNN Volume is the location where the Operating System was loaded.

The **Local Time Offset** is a system value used to determine LCT (Local Civil Time) and LST (Local Standard Time). The NonStop System technically maintain internal time as GMT (Greenwich Mean Time) and use the Local Time Offset to present LCT. Internally using GMT provides an easy and safe value for systems networked throughout the world to exchange.

The **SysGen Time** is when the Operating System was generated (via DSM/SCM).

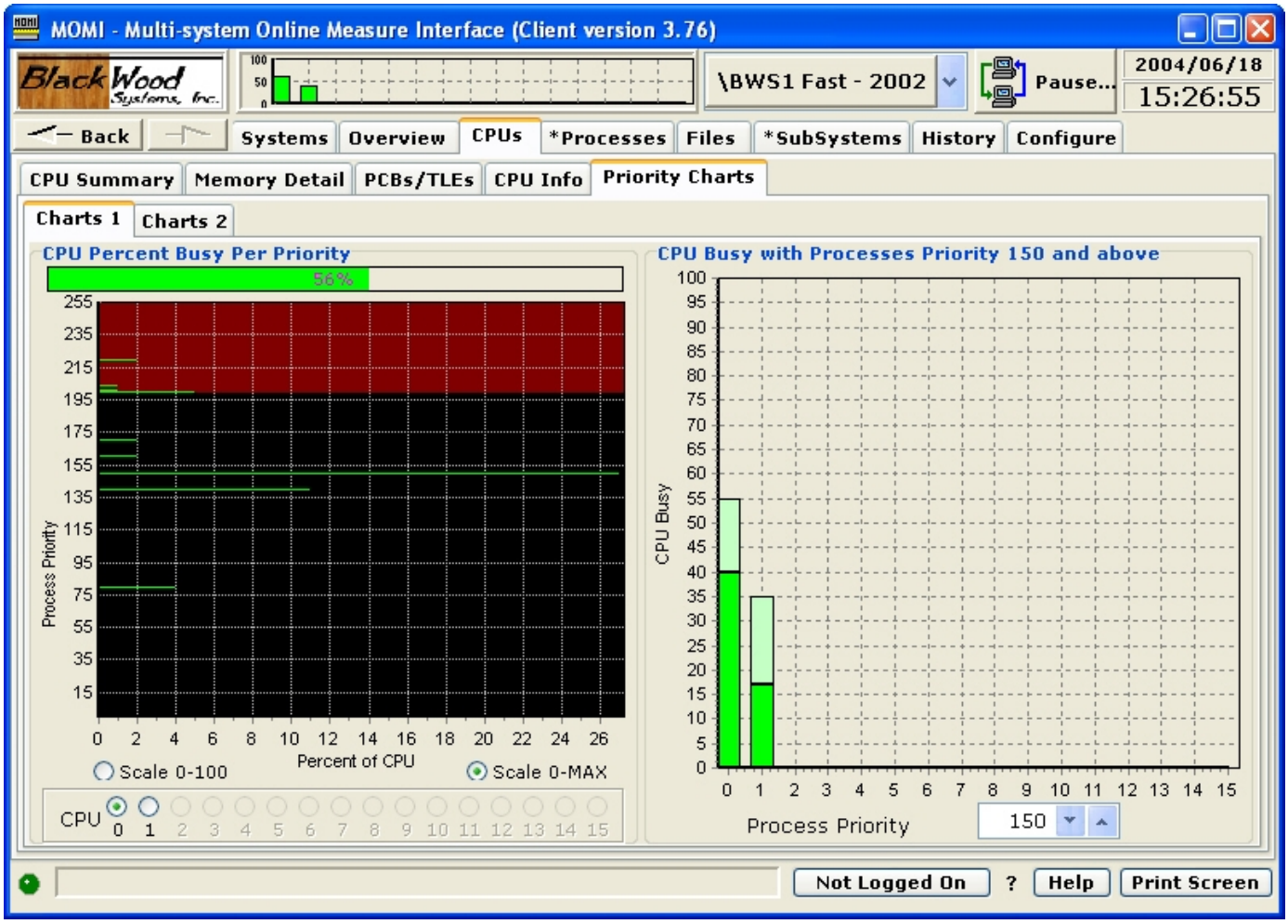
The **Cold Load Time** is when the system was loaded with the Operating System.

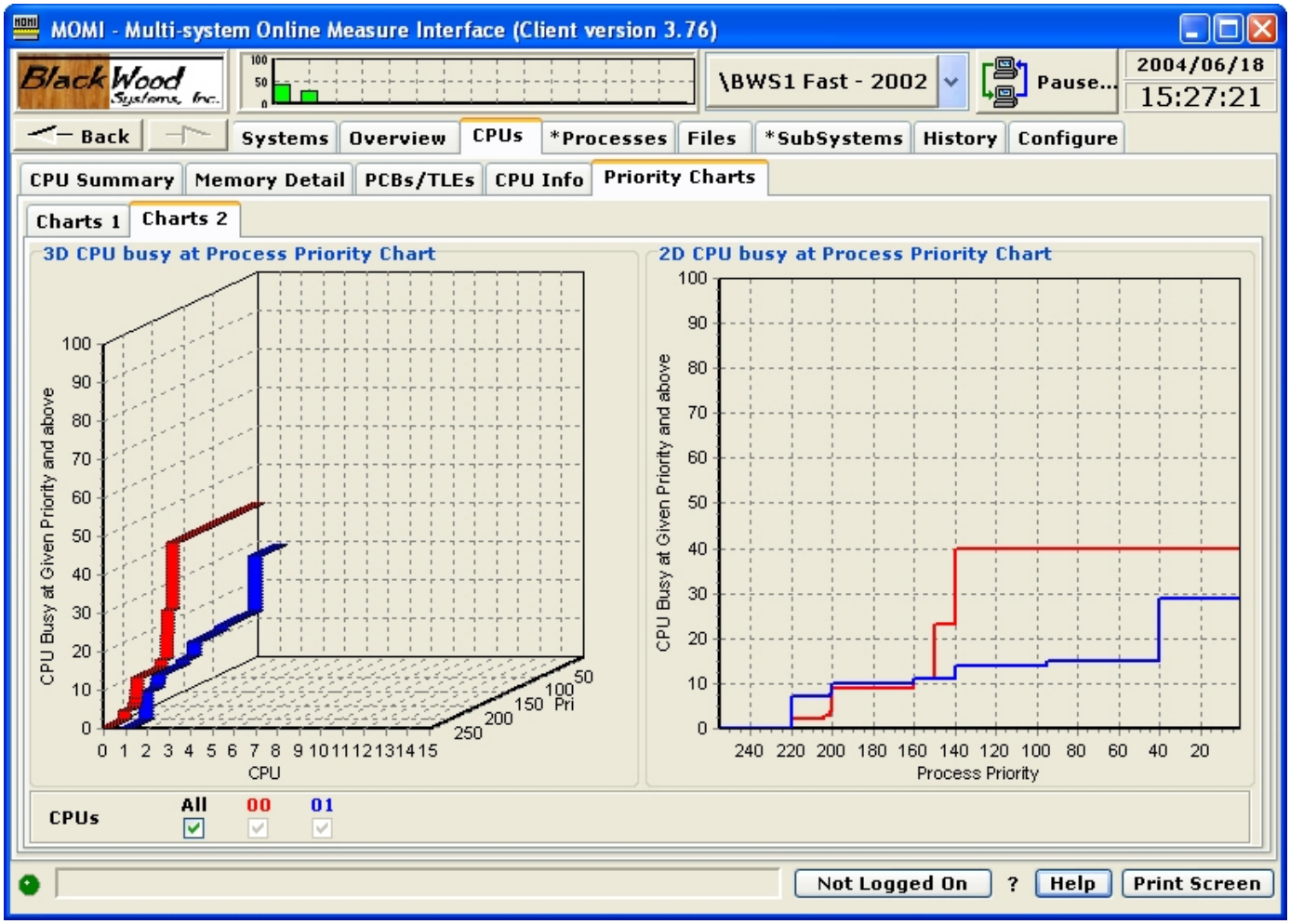
The **SysGen Time** or System Generation time is when the operating system was constructed.

The **Expand System Number** is a unique numeric identifier from 0 to 254 that identifies a System linked on a network.

Other fields of interest are the **System Loads**. This indicates the number of times the operating system was started or loaded from $SYSTEM. **Breakpoints** are the number of stops or holds placed in an executing process by the system debugger Inspect. Older platforms had to configure the maximum number of breakpoints allowed within SYSGEN.

CPUs / Priority Charts / Charts



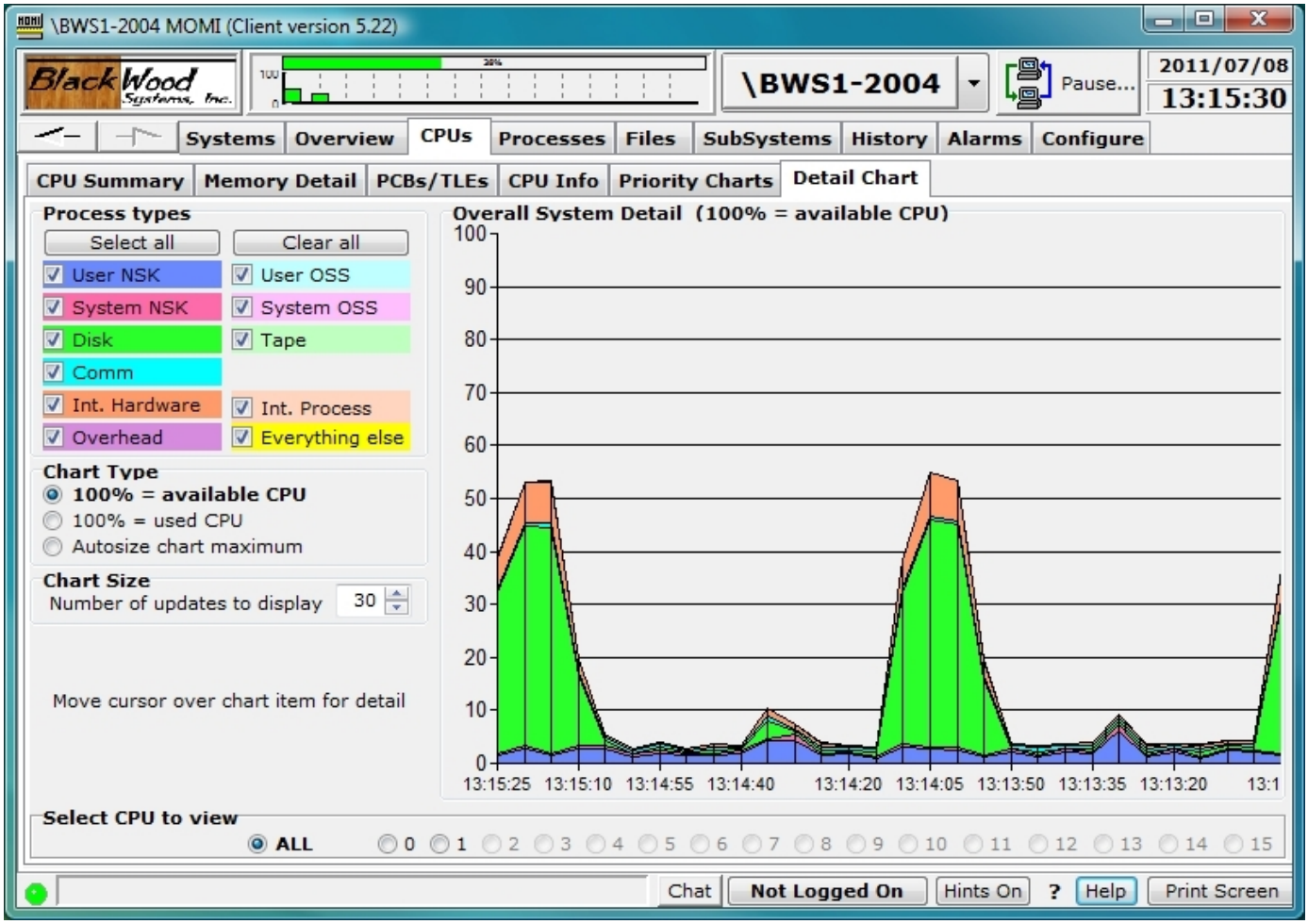


The CPUs / Priority Charts / Charts screens display the relative CPU busy as related to priority.

These screens give several visualizations to the CPU usage. While a CPU could be 100% busy, if 90% was being used at priority 10, the overall CPU health may be good but the processor is currently performing a batch type of operation.

The **Process Priority** up/down selector allows the setting of a threshold in which the lighter color represents a level below the threshold and the darker color of the bar represents the level above the threshold. For example, if the System runs production processes at priority 170 and above, and the indicator is set to 170, the dark portion of the bar presents the CPU usage at and above 170. The lighter color represents the CPU usage below that level. The total bar height still indicates the complete CPU usage.

CPUs / Detail Chart



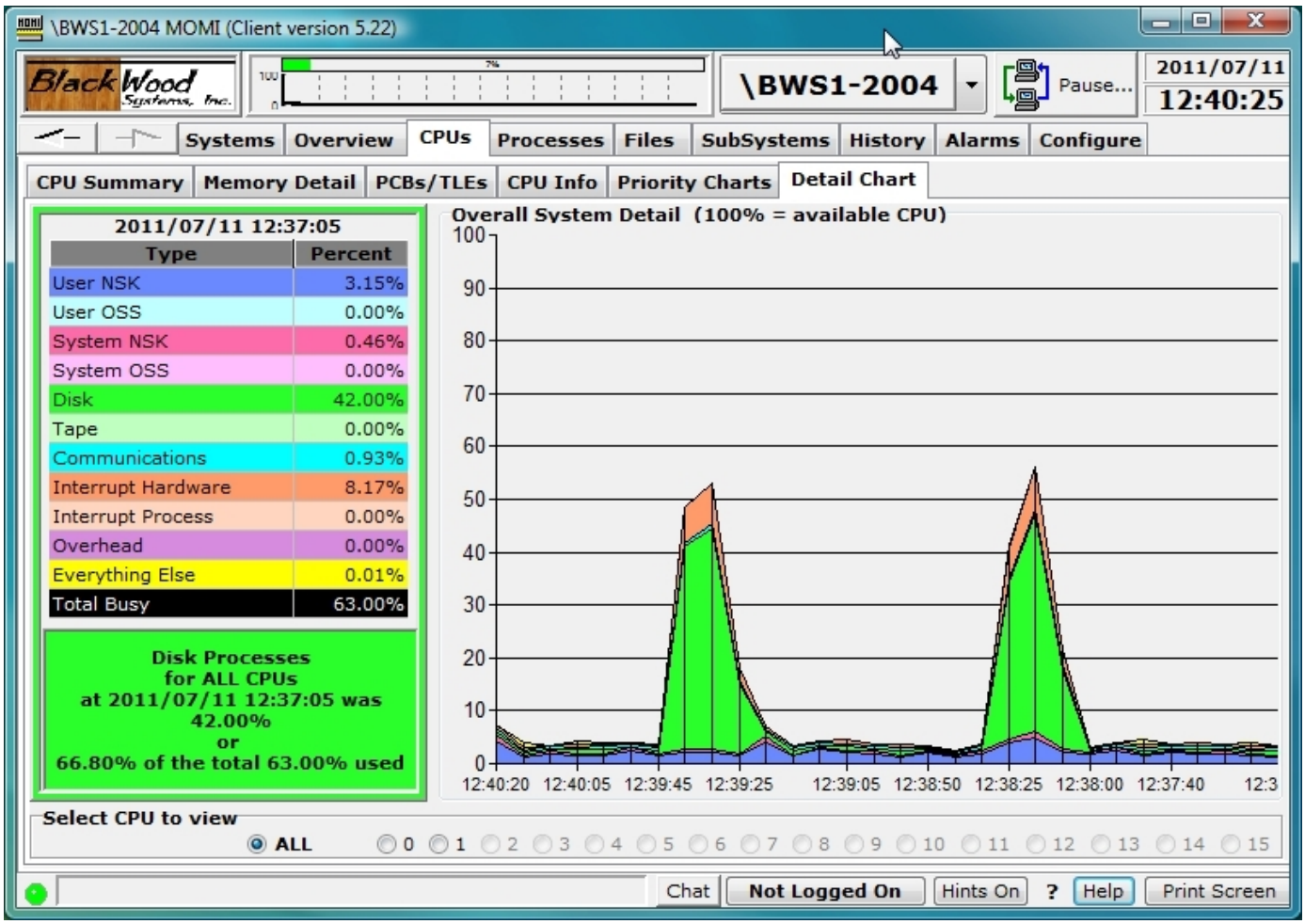
The CPUs / Detail Chart provides moving graph breakdown of CPU utilization by major category such as disk, communications, and user processes.

The following are the field meanings:

Label	Description
User NSK	Guardian processes at or below priority 199
User OSS	OSS processes at or below priority 199
System NSK	Guardian processes above priority 199
System OSS	OSS processes above priority 199
Disk	Disk processes (physical or virtual)
Tape	Tape process
Comm	Communication processes (i.e. LAN, SWAN)
Int.Hardware	CPU interrupt handler routine

Label	Description
Int. Process	System Interrupt Processes (i.e. processes at priority 255)
Overhead	System time for process start-up/shutdown
Everything else	all other processing not within an existing category

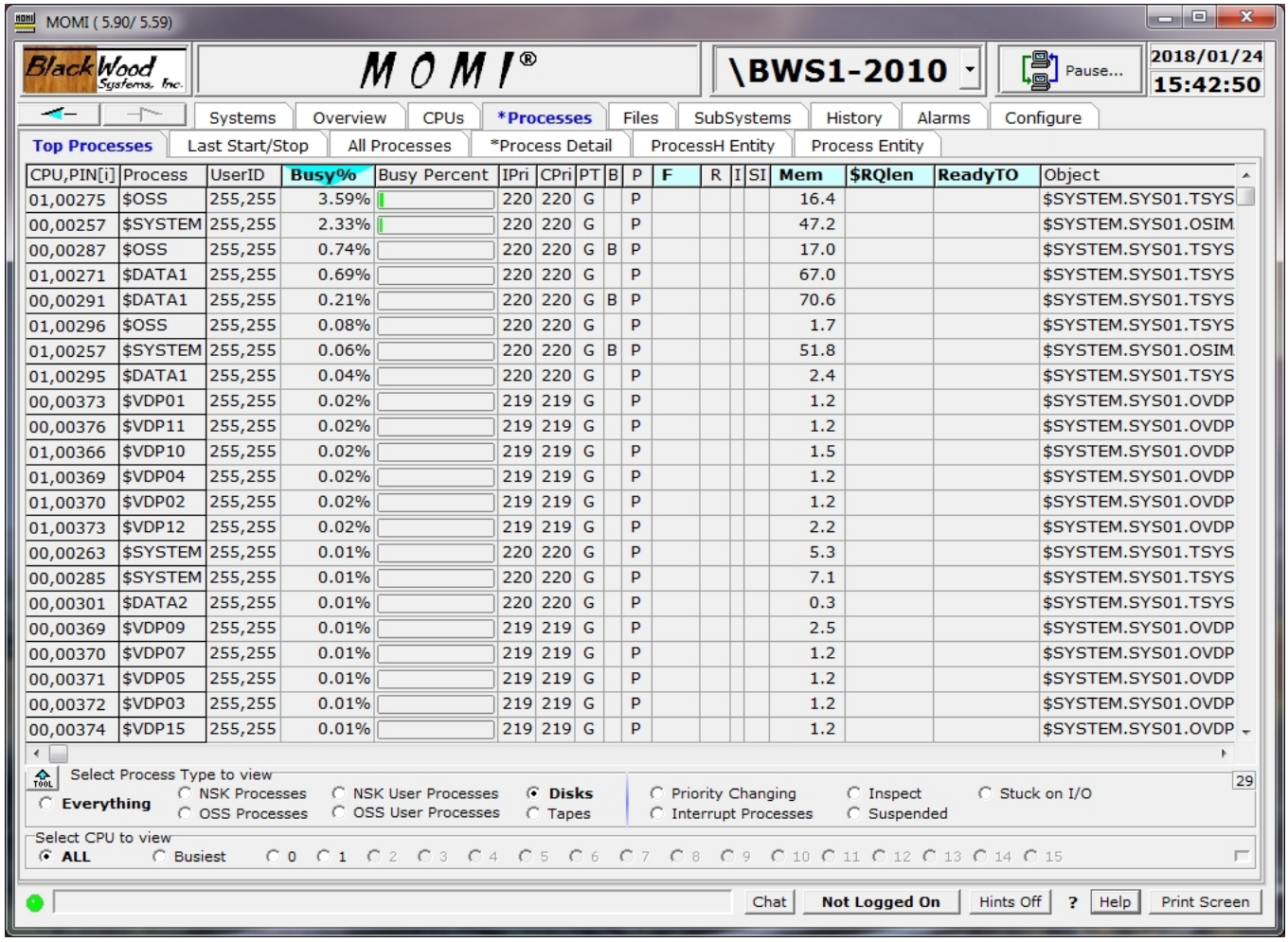
Placing the mouse over the graph displays a pop-up window on the left showing detail for the area under the mouse.



The chart clears and restarts when a different system is selected.

Processes

Processes / Top Processes



The Processes / Top Processes screen presents the top 'hitters' of various system resources. The information presented on this screen is quickly displayed as it is always maintained by MOMI.

The screen is configured by starting at the bottom and working your way up. **Select CPU to view**, **Process Type** and click one of the highlighted headings at the top of the page (**Busy%**, **F**, **Mem**, **$RQlen**, **ReadyTO**).

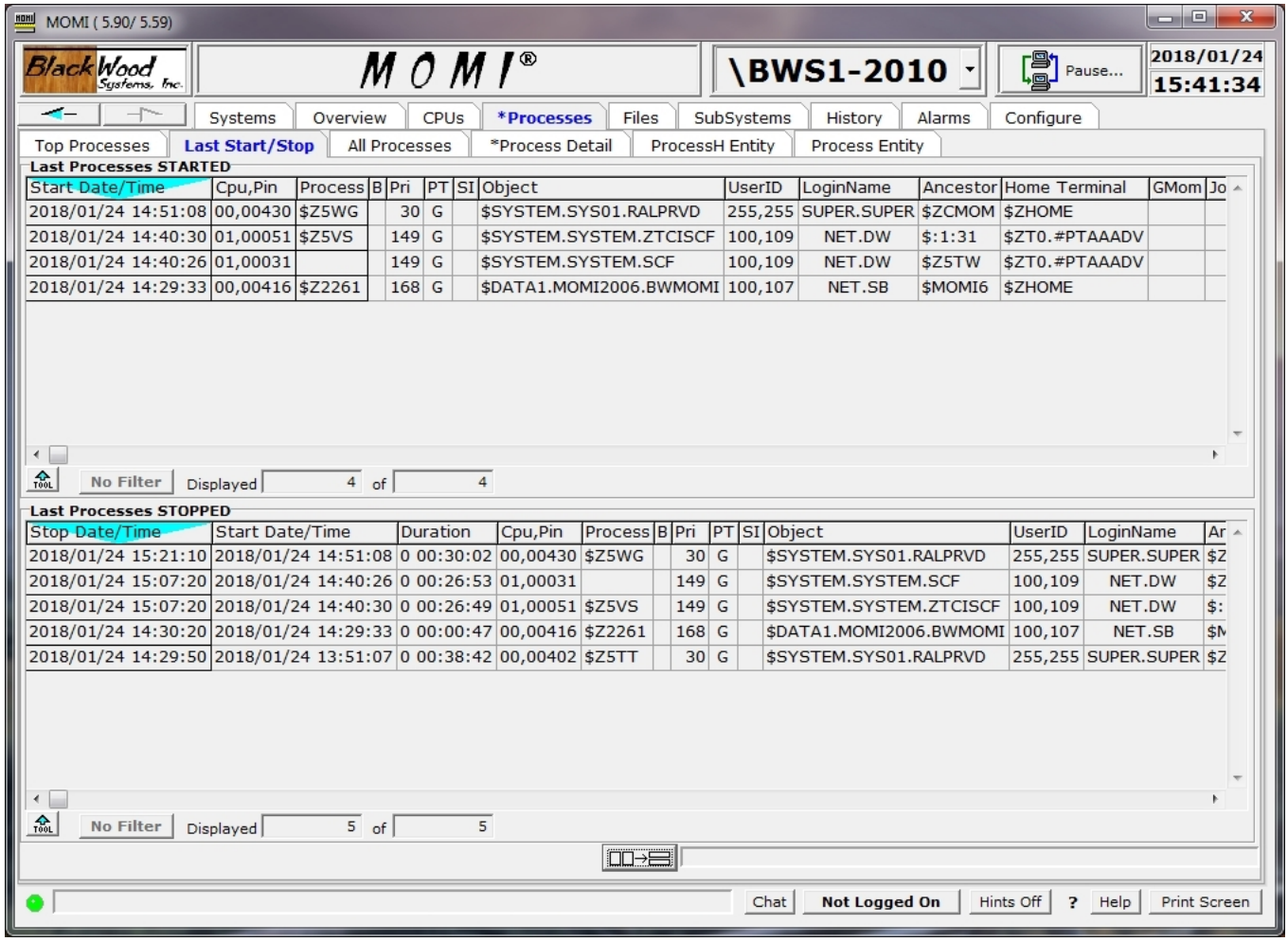
For example, to view the top memory users in CPU 1 of user processes, click CPU to view 1, Process Type to view Process Only and then click the Mem heading.

In the section "Select Process Type to view":

Everything	displays all processes
NSK Processes	all Guardian processes
NSK User Processes	all Guardian processes except system processes

OSS Processes	all OSS processes
OSS User Processes	all OSS processes except system processes
Disks	only disk process
Tapes	only tape process
Priority Changing	processes with priority temporarily lowered by system
Interrupt Processes	TNS/E - TNS/X interrupt processes
Stuck on I/O	processes in wait state of %004 (see IO-STUCK-INTERVAL)
Inspect	any process in Debug / Inspect
Suspended	any process suspended

Processes / Last Start/Stop



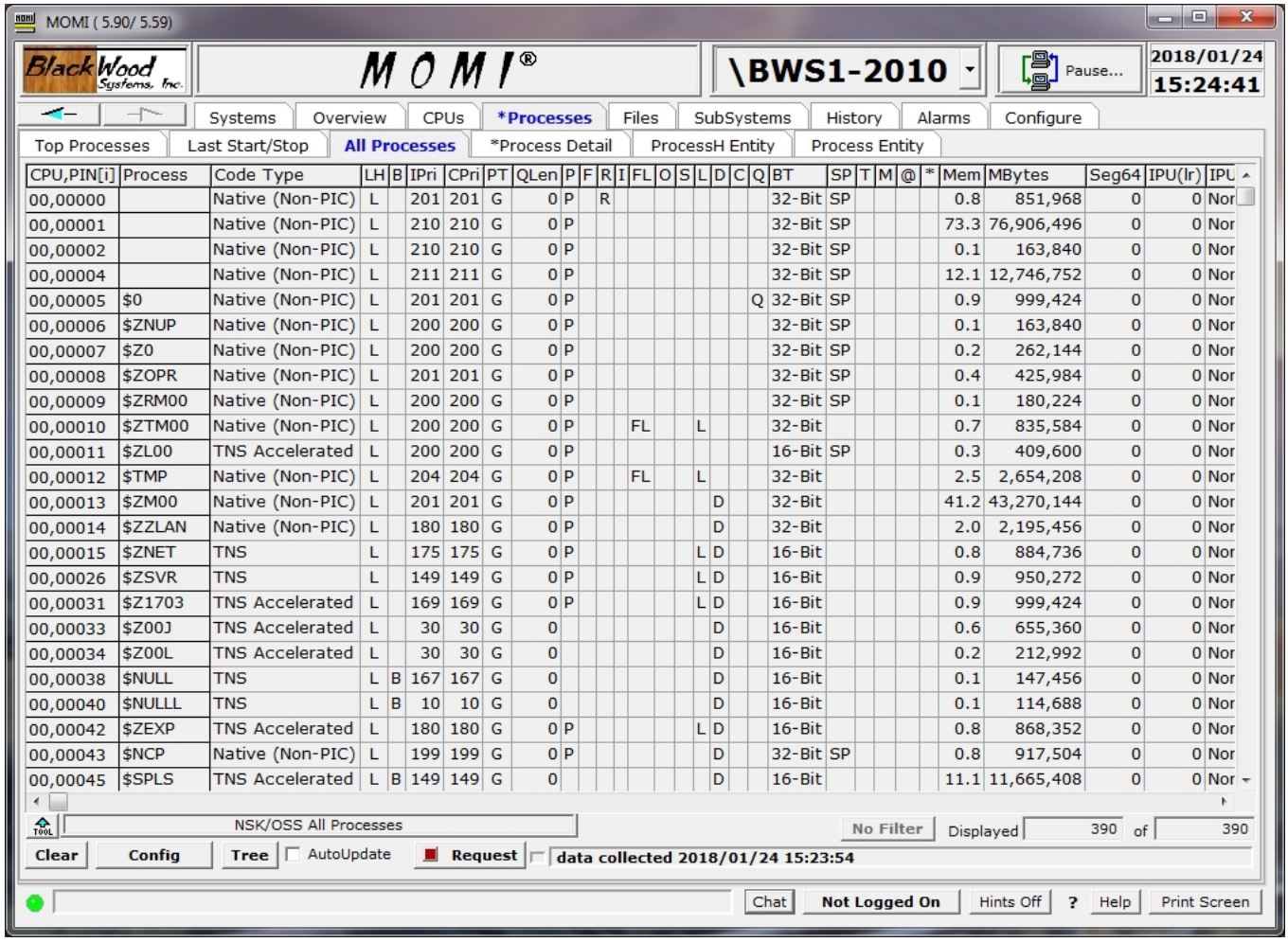
The Processes / Last Start/Stop screen presents the last few processes that were created or terminated on the System.

This view of process turnover is as seen by MOMI. Short lived processes, those running less than the Interval, may not be reported on this screen.

During process creation, there is a short period of time where accurate information is not available and this initial creation period is ignored by MOMI. A process is not considered stopped until it has been completely removed from the System or replaced by another process.

Processes started by MOMI, and any direct descendants are intentionally omitted from this screen.

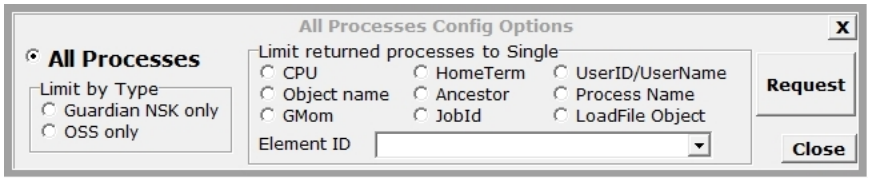
Processes / All Processes



The Processes / All Processes screen presents a comprehensive detail of each process requested. The detail provided is generally everything you wanted to know but were afraid to ask! The information is generally *static* in nature and less performance related.

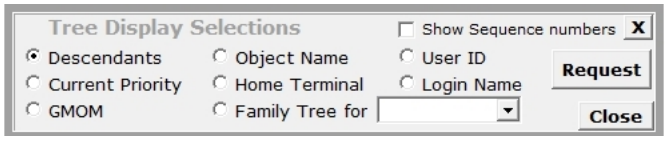
The **AutoUpdate** check box causes automatic request of new data.

By default, pressing the **Request** button will gather and report all processes on the System. While the data can be filtered and sorted as needed, the **Config** button allows the User to qualify the desired data.

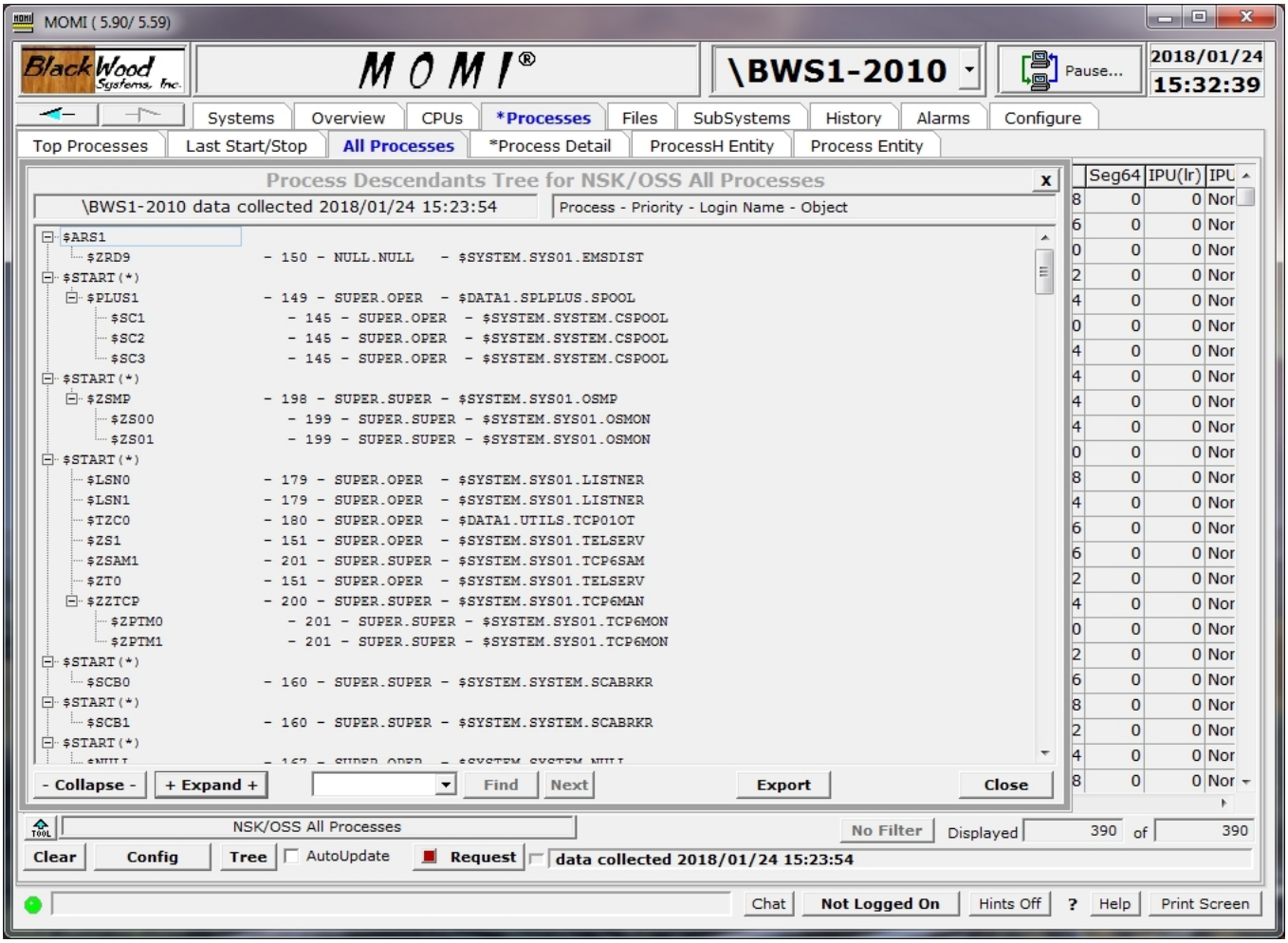


Limits entered on the Config Options pop-up require that the information return match the specified criteria. This reduces the information collected and speeds up the request.

Once data is retrieved, it may also be organized in a Tree View fashion. Pressing the Tree button displays the following:



The organization items allow relationships to be better viewed. For example, displaying data by Descendants presents a hierarchy of the order of process creation.



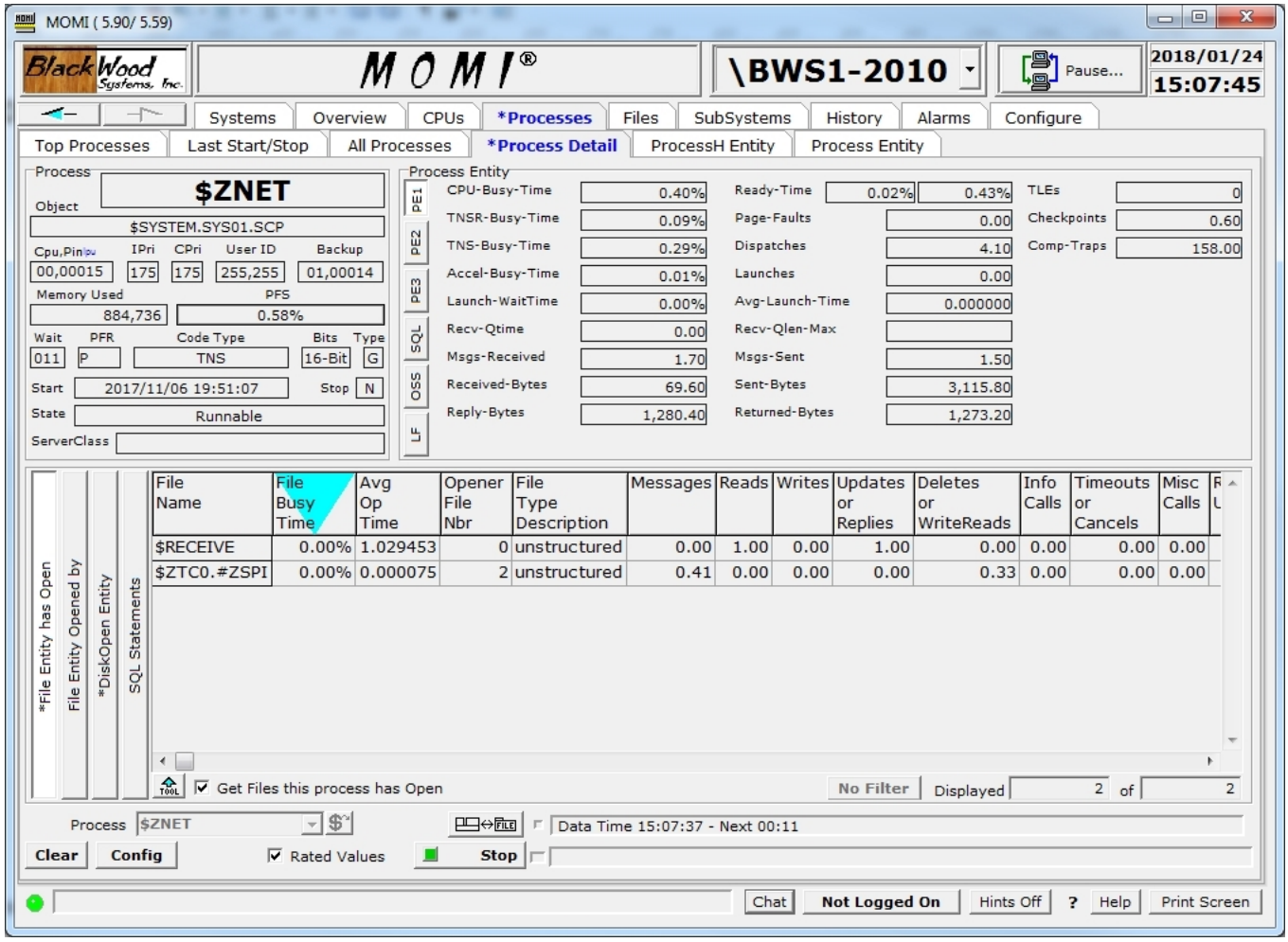
The **Tree View** is based on the information gathered from running processes. Processes referenced in a hierarchy that are no longer active may cause gaps in the Tree View (i.e. a process may refer to an ancestor no longer running).

The Find button allows searching for text within the Tree View. This can be useful on large systems with a great deal of data.

MOMI server version 5.24 and later:

> If a subtype 30 process is suspended or in debug/inspect a device query (where a message is sent) is avoided and error 160 (ZFIL^ERR^INVALIDSTATE) displayed in the **DI Error** column.

Processes / Process Detail



The Processes / Process Detail screen presents a comprehensive look into a process.

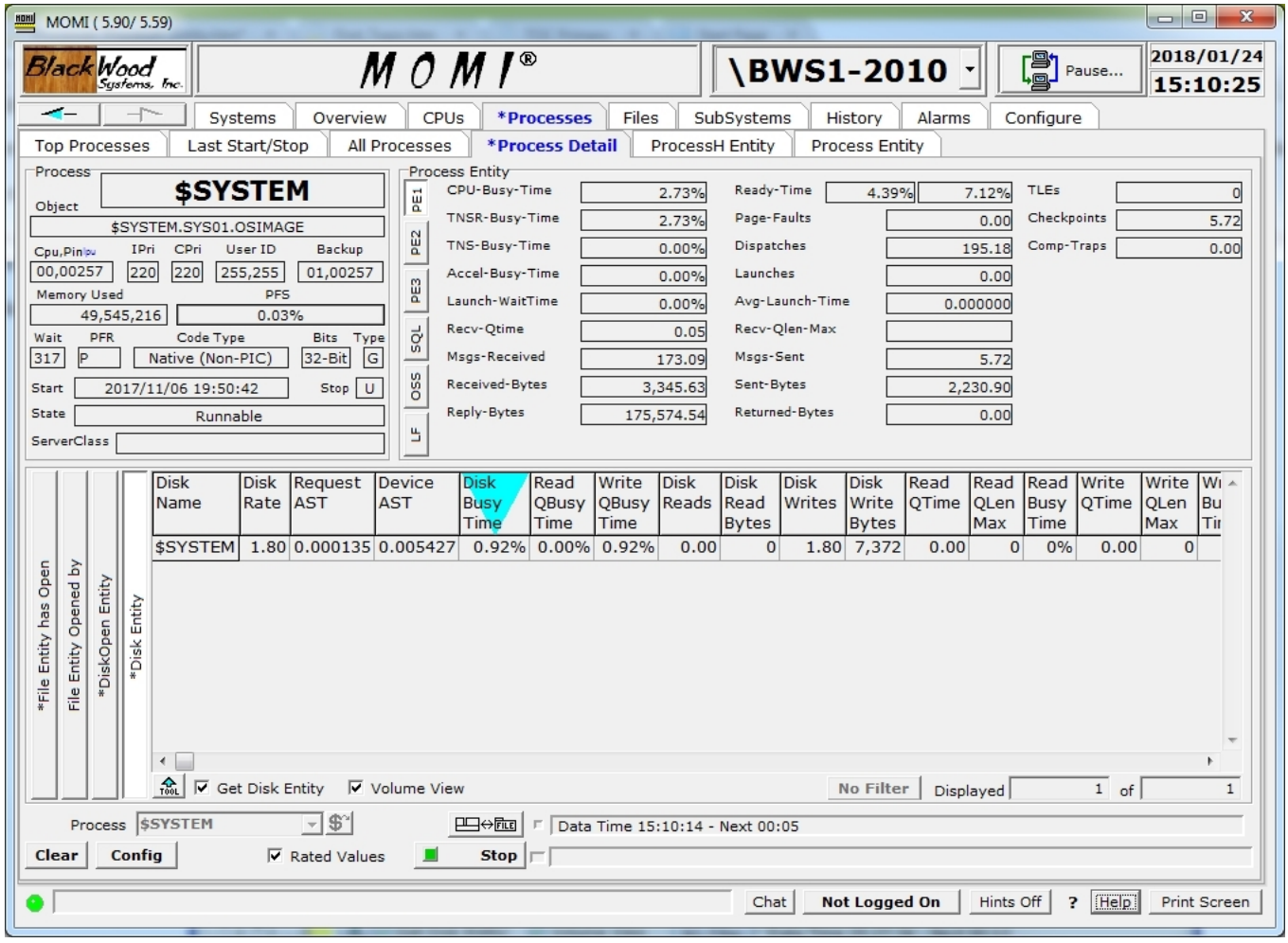
Click on the areas in the image above display additional information.

The screen presents resources consumed, files opened by the process, files opened to the process, and SQL Statements activity. The SQL Statements section is not activated by default.

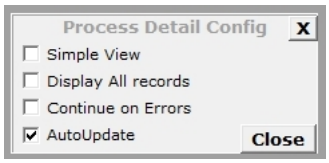
Information on this screen is obtained via Guardian procedure calls and the MEASURE entities of Process, File, DiskOpen, SQL Process and SQL Statement.

Values by default are rated in a "Per Second" mode such as 1.40 Comp-Traps. Uncheck the Rated box causes raw values to display. The window of time for the Process Entity is visible by pushing the Page 3 button. The window of time for the File, DiskOpen and SQL Statements entities are visible by scrolling all the way over to the right.

A Disk process displays similar information except the SQL Statements button is replaced by the Disk Entity.



The operation of the screen may be altered by the **Config** button:



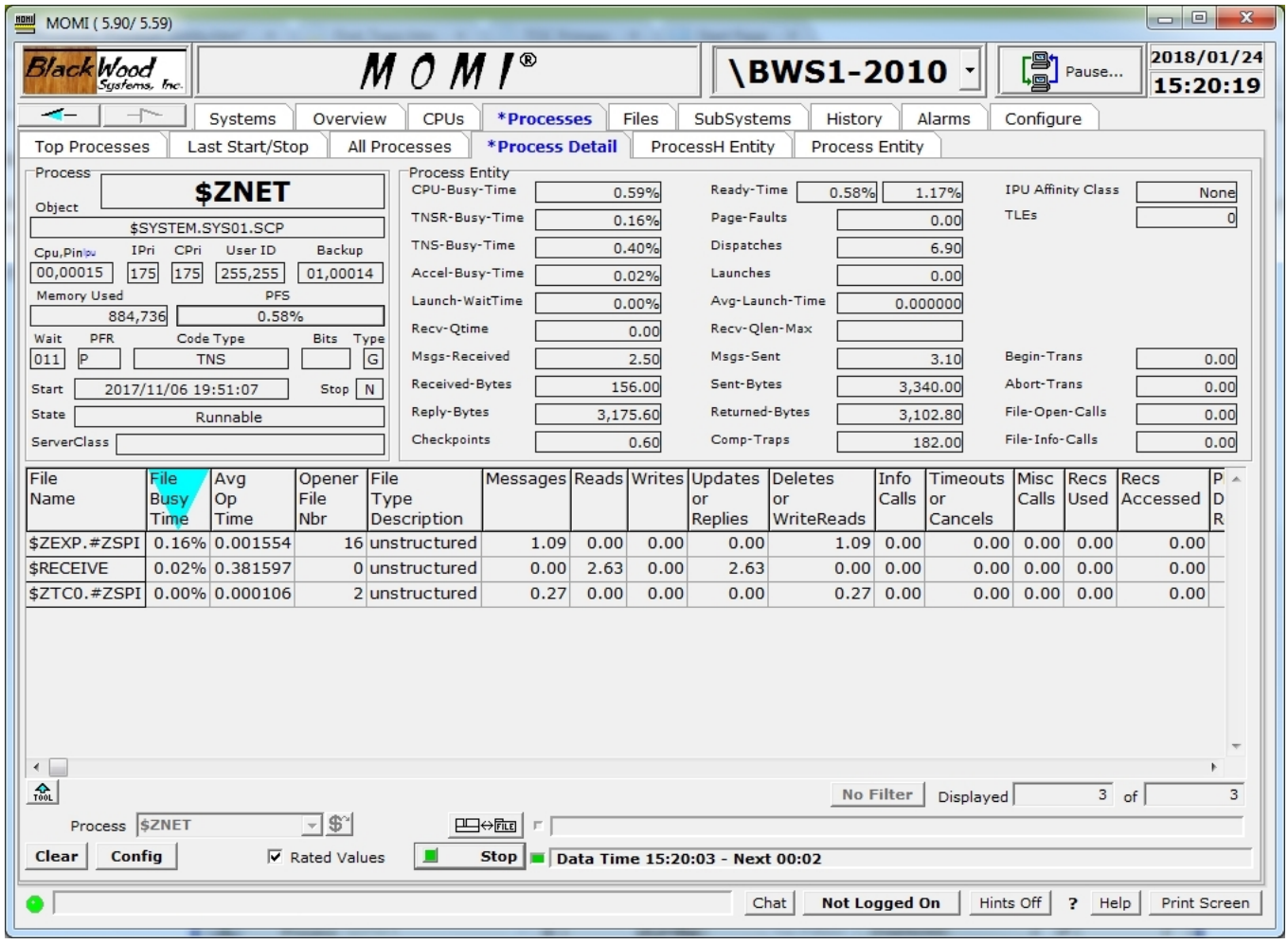
Simple View will reduce or 'down size' the amount of information presented. By Default, records in the File, DiskOpen and SQL Statements do not display records with zero values (i.e. open but not active).

Display All records option allows zero value records to be displayed.

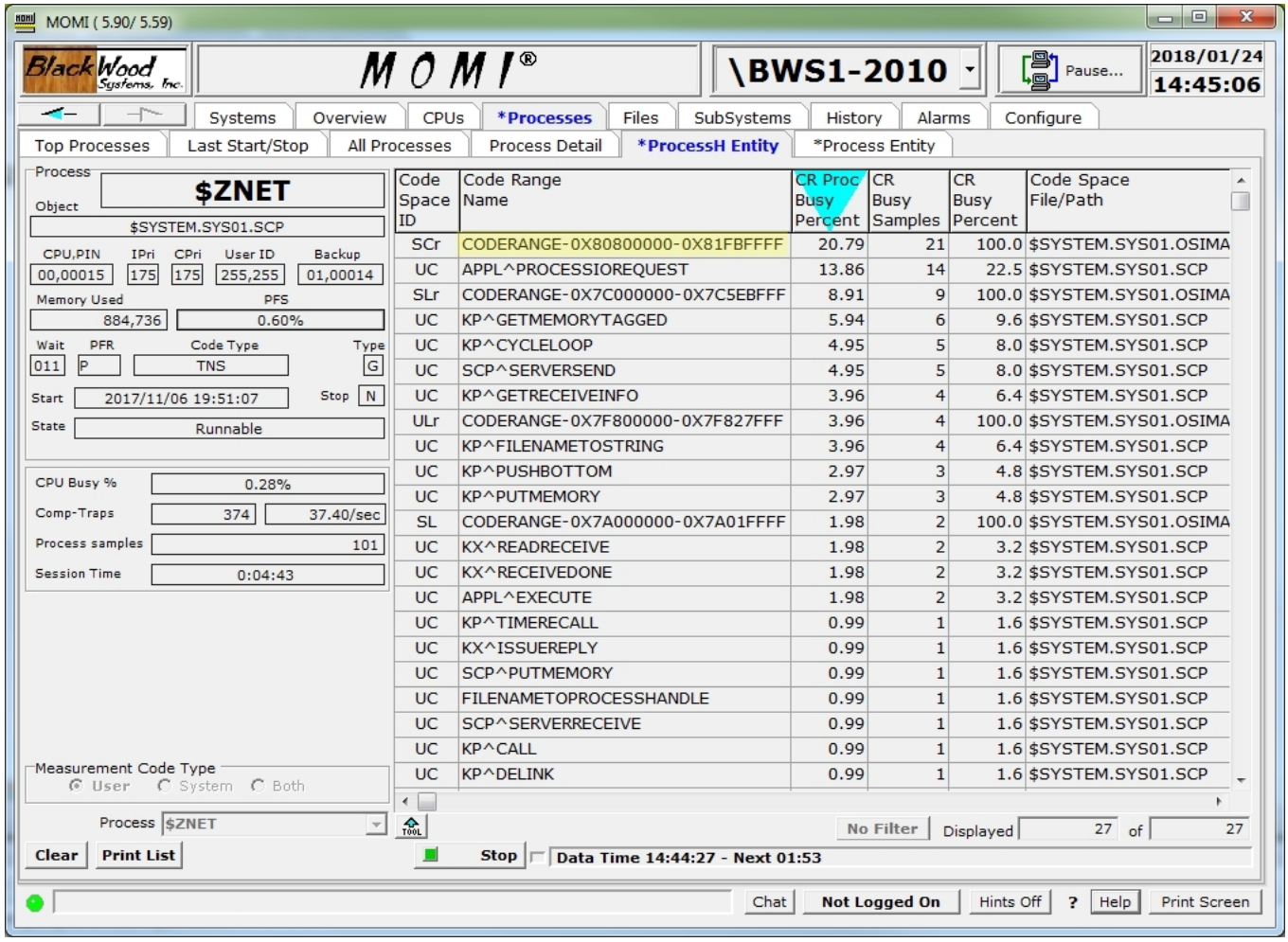
Continue on Errors option causes the screen to continue gathering data even when the process being measured has stopped (i.e. the display of data automatically picks back up again if the process is restarted). This is useful for transient processes or during debugging sessions.

AutoUpdate, checked by default, determines if the screen continues to update after Request is pushed.

Below is a sample screen for the **Simple View** selection:



Processes / ProcessH Entity



The Processes / ProcessH screen presents the locations within the process code where CPU time is spent. This information is obtained from the MEASURE ProcessH entity.

While a process has a busy time, the areas of code where it spends its execution time are of interest to developers and may reveal locations that are not performing efficiently. Generally, only a developer with access to the source code can interpret the meaning of the values given on this screen.

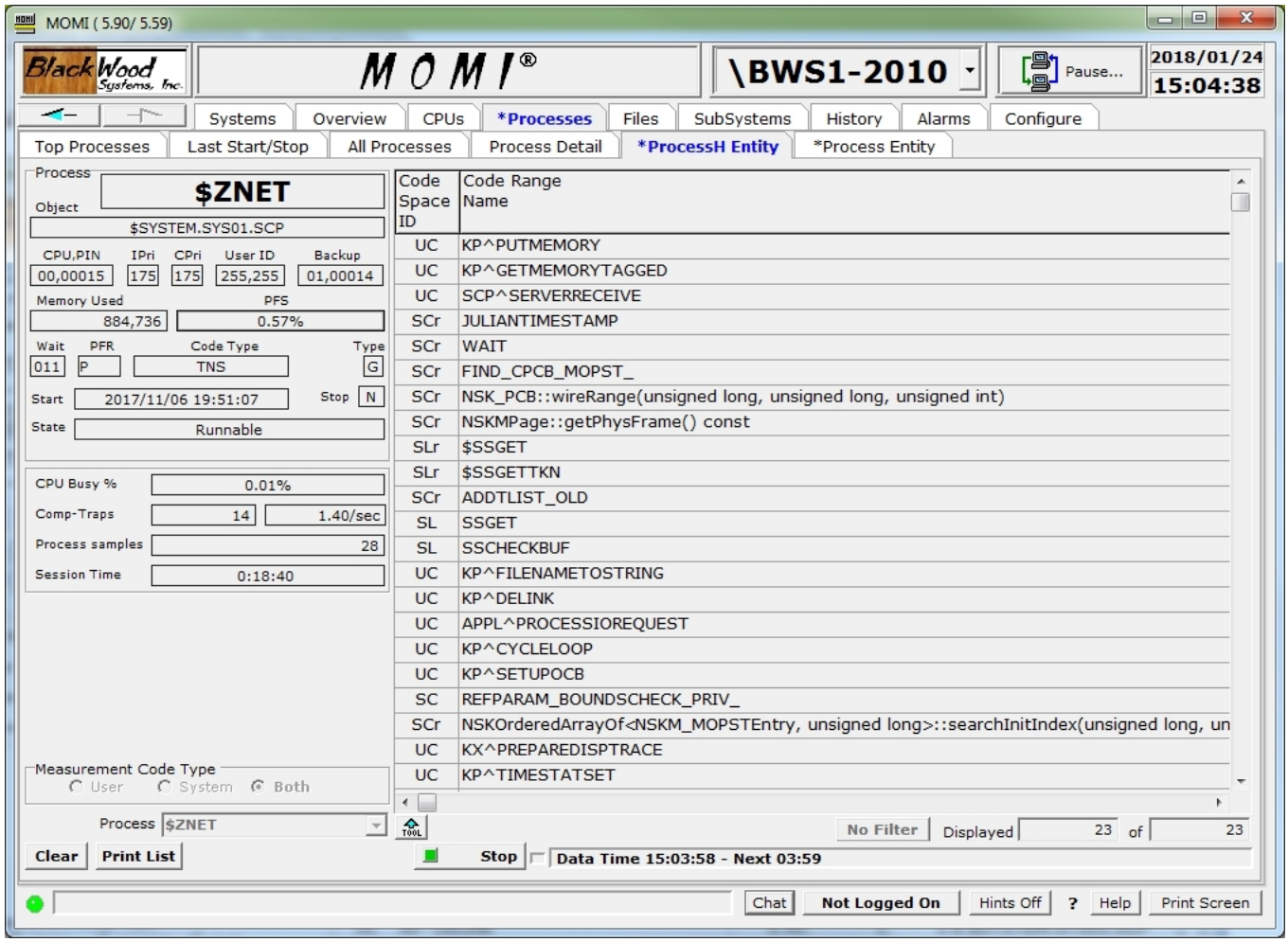
Three types of Measurements are available: **User, System,** or **Both**.

User code generally is the area of the program the developer has actually coded. This includes the main object file plus all loadfiles (i.e. SLRs and DLLs) not located in $SYSTEM.SYSnn or $SYSTEM.ZDLL.

System code is the area spent in the operating system supplied routines and is defined as the loadfiles (i.e. SLRs and DLLs) located in $SYSTEM.SYSnn or $SYSTEM.ZDLL.

Both are a combination of **User** and **System**.

The image below shows some of the additional procedure information obtained with **Both** is selected.



Measuring the System code may be an interesting exercise, but will usually result in reporting locations that don't make much sense. This is due to the fact that documentation on internal operating system routines are not published. On occasion, the code area may have a name that directly relates to a system call initiated by the developer (such as FILE_SETKEY_ (position within an Enscribe file) or malloc (a C memory allocation routine)).

A Code Range with a name of CODERANGE (sample highlighted in yellow above) followed by hexadecimal numbers are locations where time was spent but a procedure (or function) name was not available. More recent releases of MEASURE seem to collect information on the entire process even if only the user area is selected. The Code Range may indicate system code, SRL's or DDL's.

A Code Range is an individual function or procedure within a program. Programs are typically comprised of many functions or procedures Exceptions to this are COBOL and Java which is interpreted code not supported by this MEASURE entity.

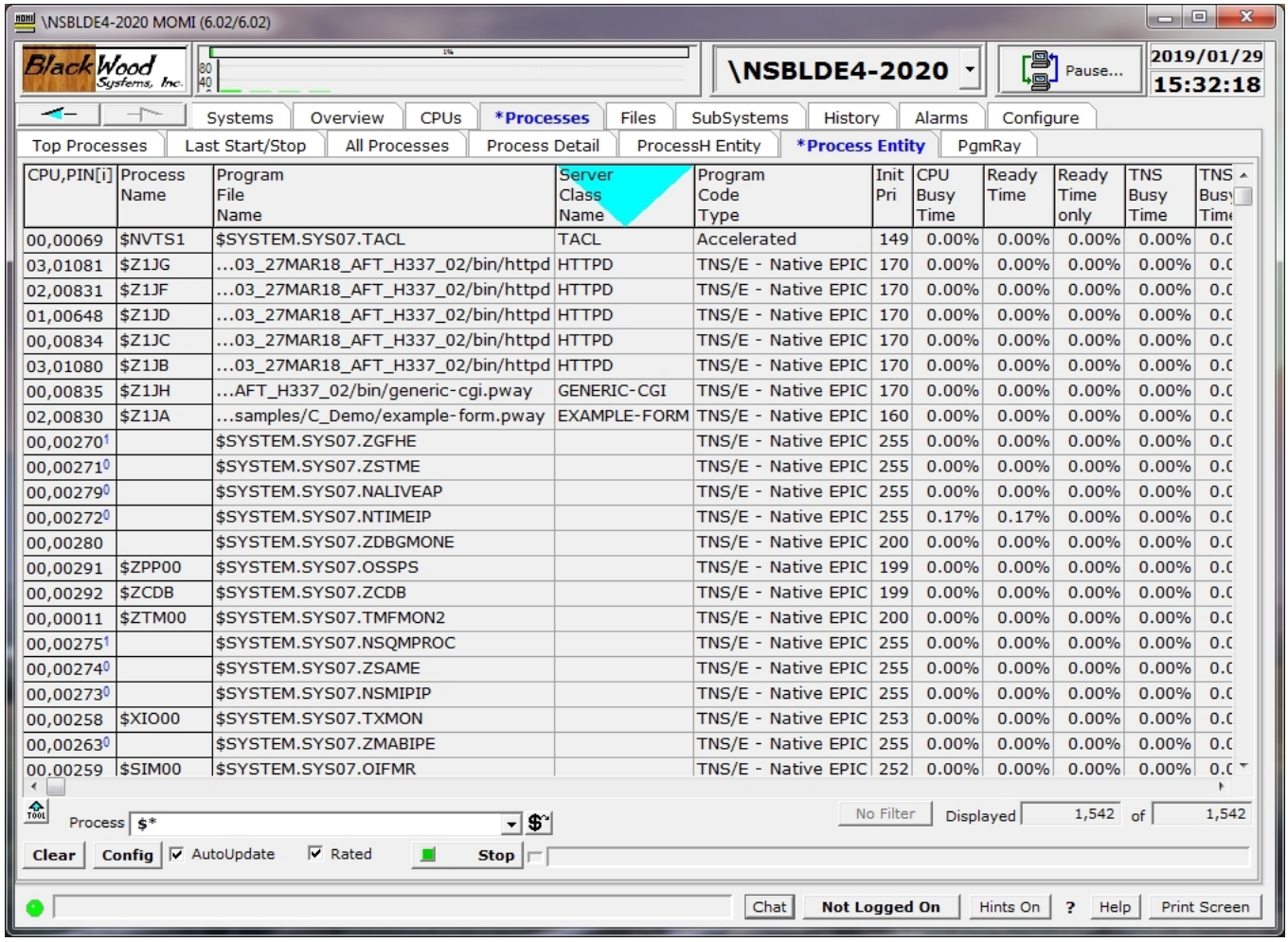
A Code Space is a logical grouping of one or more Code Ranges. This typically could be thought of as all of the Code Ranges within program file comprise a Code Space. However, a process accesses many Code Spaces as each SRL or DLL and the Operating System itself are divided into Code Spaces.

The column CR Proc Busy Percent reports the percentage of the total process busy time the code range was active. This column is the best overall indicator to the activity of a Code Range.

This screen continually gathers data in increasing increments. The initial sample is of a short period of time increasing to longer periods of time. This allows a relatively quick display of initial samples while easing the system load if the measurement is left to run. Internally, MOMI combines data from each measurement and displays the seamless results. Processes that consume very little CPU may require running this measurement for several hours in order to gather sufficient data.

Security Note: The User ID that MOMI operates under must be able to READ the object file(s) of the process under measurement. MEASURE requires this information in order to obtain procedure / function names and code range locations.

Processes / Process Entity



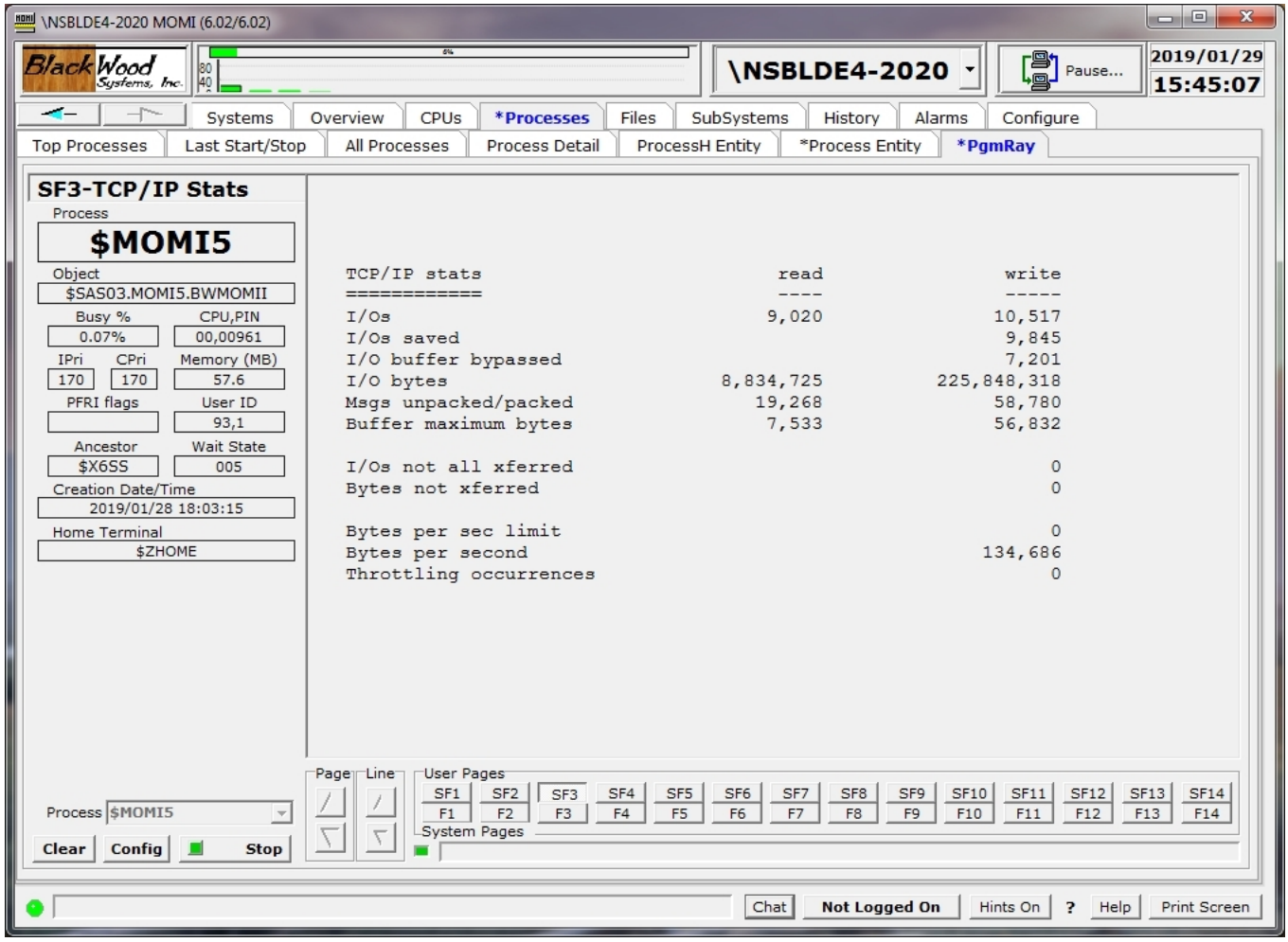
The Processes / Process Entity screen presents the MEASURE Process Entity in tabular format for a number of processes.

This screen differs from the Processes / All Processes screen in that MEASURE is used to obtain the information rather than Guardian procedure calls.

The data retrieved may be limited to a single process or all processes in a CPU.

Values such as **CPU Busy Time** and **Page Faults** by default are rated to result in a percentage of the measurement window or a "Per Second" as appropriate. Uncheck the **Rated** box will cause raw values to display. The measurement window is the last column all the way over to the right.

Processes / PgmRay



The Processes / PgmRay screen presents information using an internal interface into MOMI and other programs written by BlackWood Systems.

The screen is used internally by BlackWood Systems and may be required in troubleshooting certain end-user problems.

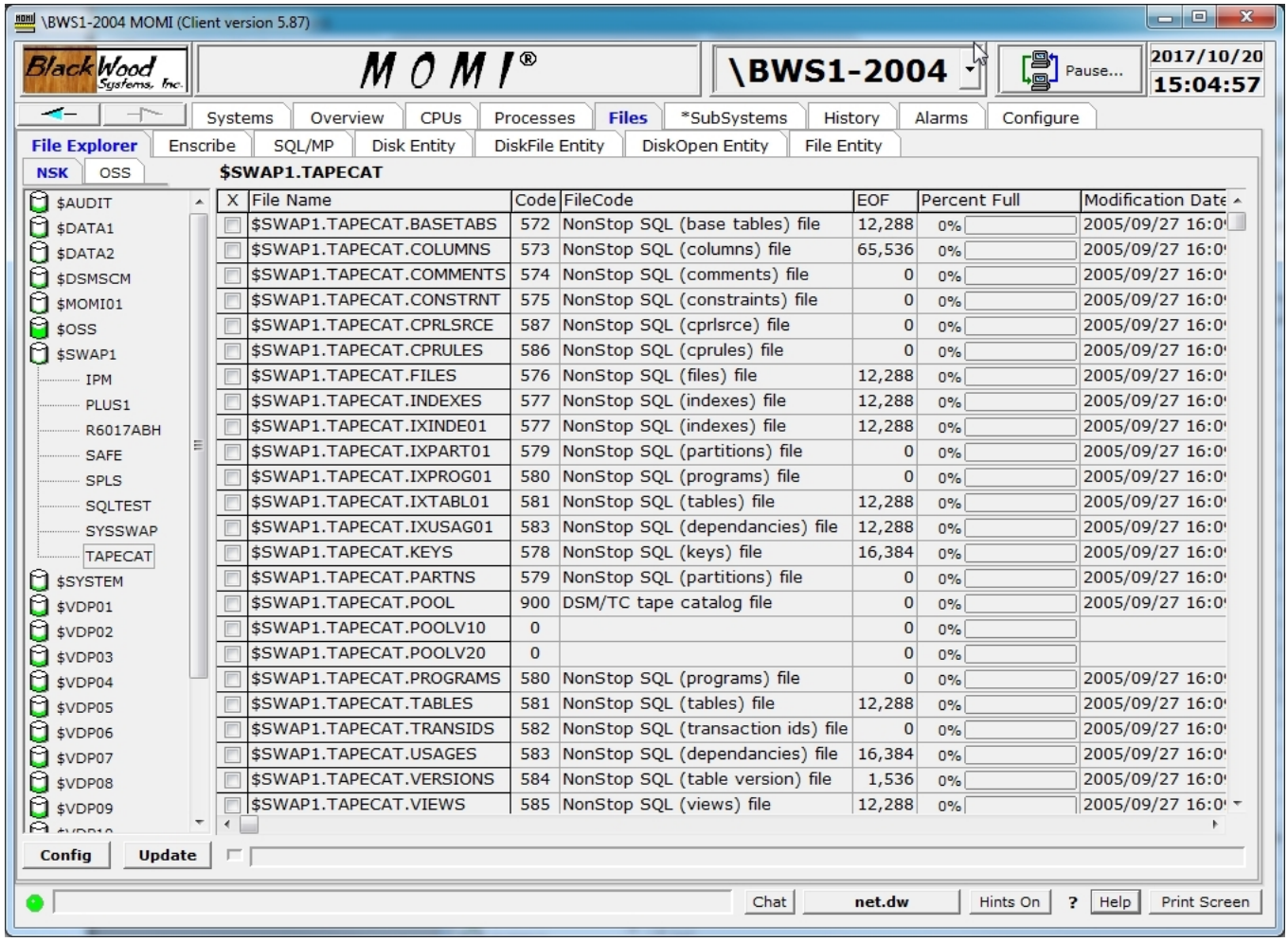
Various internal information such as memory usage, open files, message traffic and task status are available. The function keys F1 through F14 provide generally fixed information. The function keys SF1 through SF14 provide information specific to the program.

This screen is only for use in BlackWood Systems programs and you should not attempt to use it on any other processes. Doing so causes an open to the specified process and query for information that may adversely affect its operation.

This screen is turned off by default and is enabled on the screen Configure / Client / Single Screen under **Optional Screens**.

Files

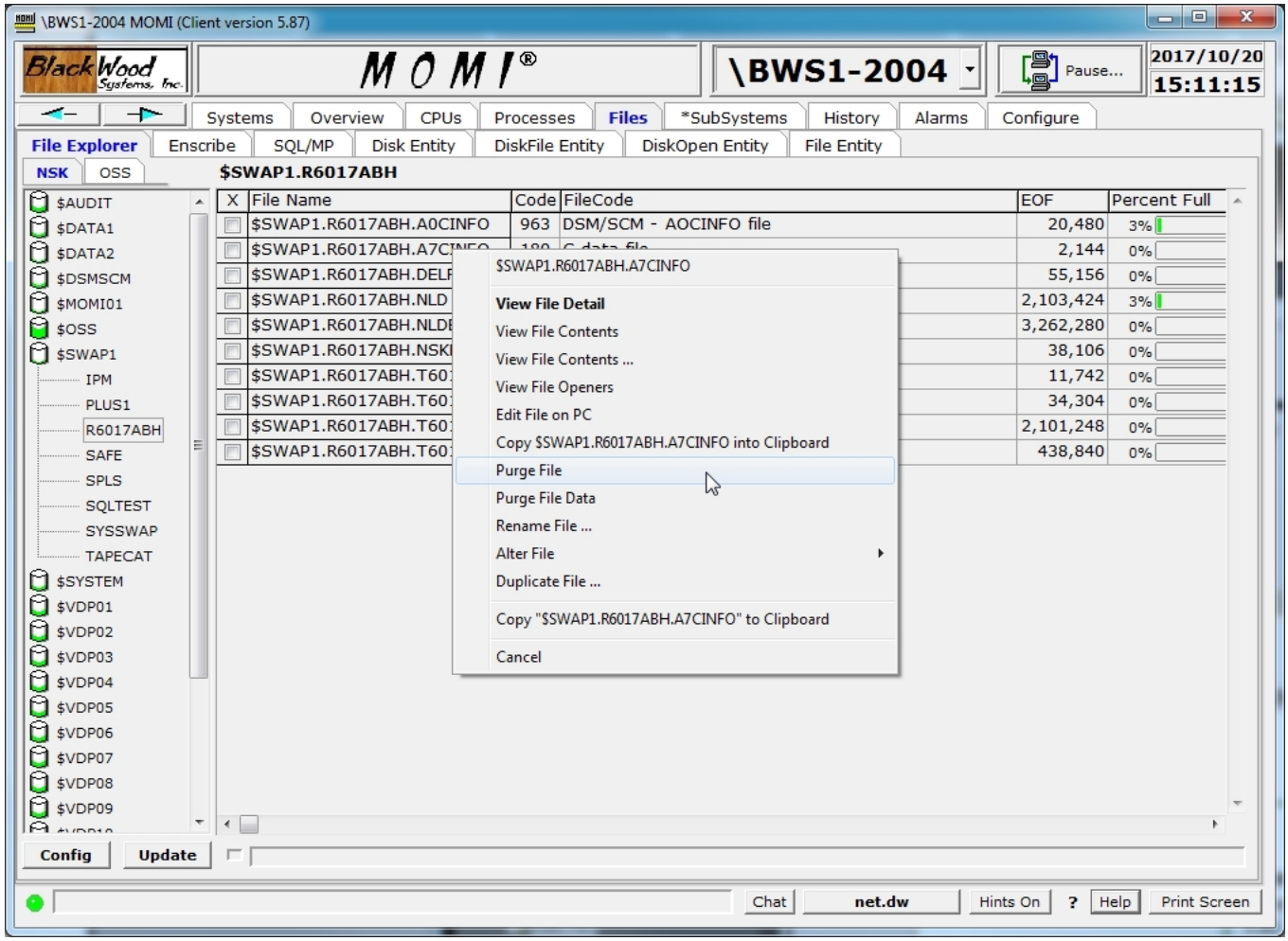
Files / File Explorer



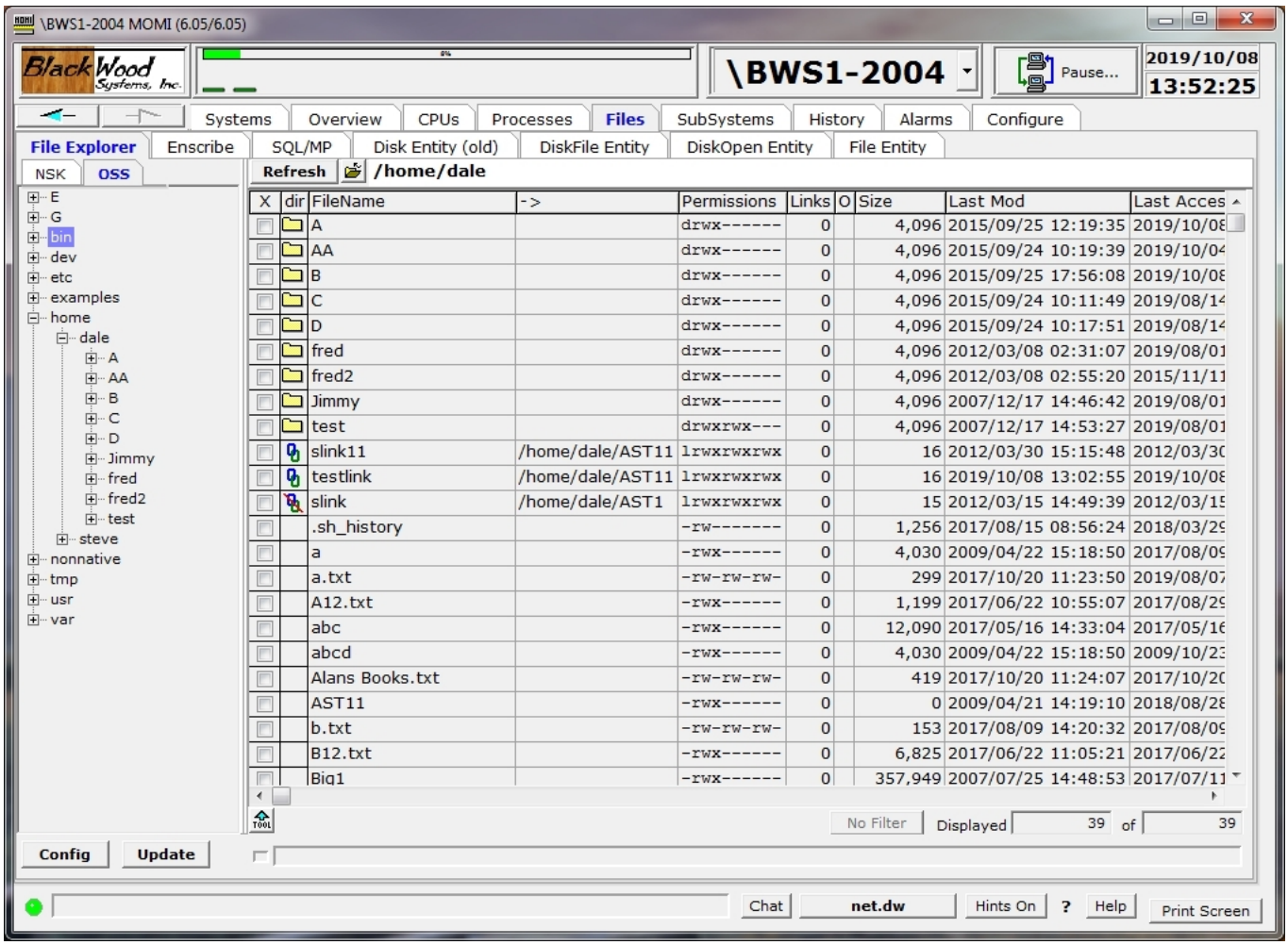
The Files / File Explorer screen provides a powerful high level graphical navigator for NSK (i.e. Guardian) and OSS files. Operations such as drag-and-drop, copy-and-paste and other right-click functions are supported.

On the left side of the screen are listed disk volumes and clicking on a disk volume displays its subvolumes. Clicking on a subvolume displays on the right the files under that subvolume. The disk volume displays in color the approximate percentage full for the disk.

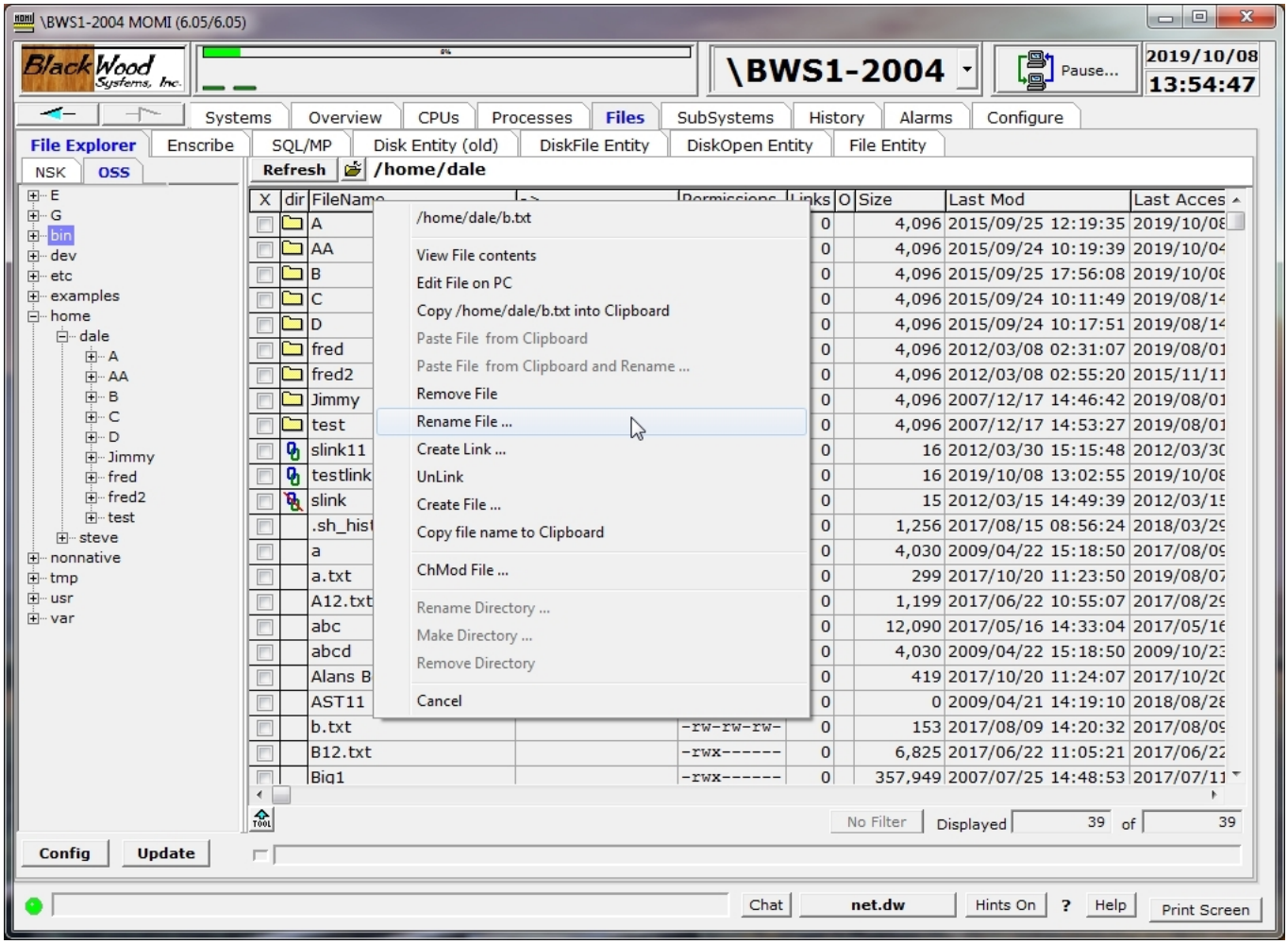
Actions that are available are displayed by right-clicking on a file name.



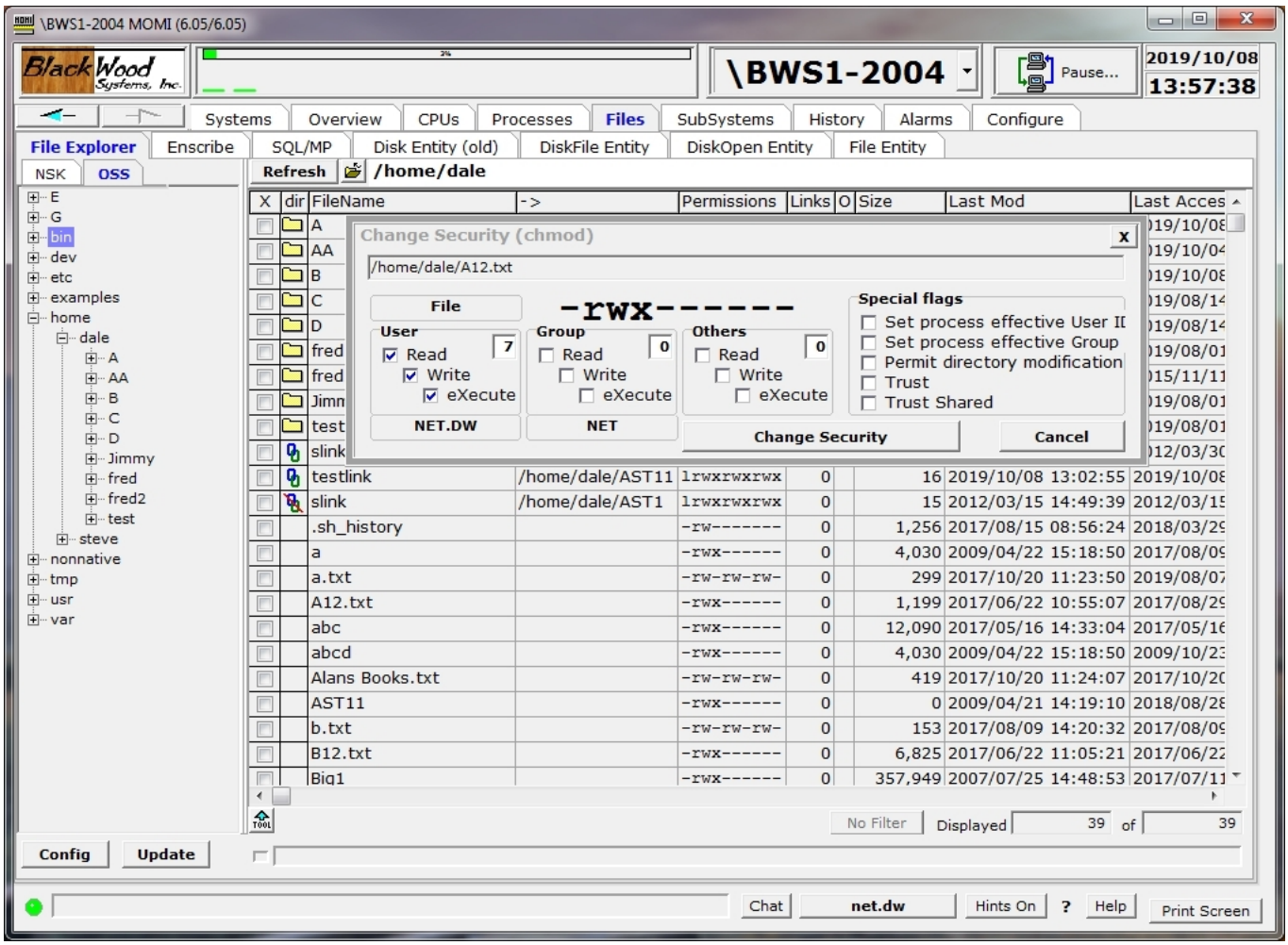
Selecting the OSS tab displays a tree view of the OSS file system on the left and files under the currently selected subdirectory. Subdirectories on the left continue to drill-down if selected.



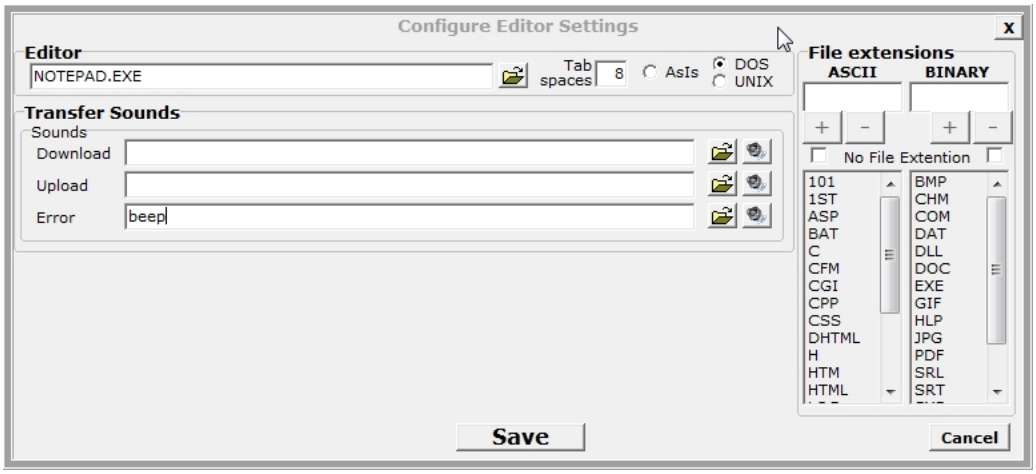
Available actions are displayed by right-clicking on a file name.



Additional pop-up information is displayed when the mouse is placed over a **Permissions** field. A right-click option of **CHMod File...** displays and allows changing of permissions and Special flags.



The Config button displays configuration settings for this screen such as the Windows editor to launch when editing a file and sounds played. The **No File Extension** option determines the treatment of a file with no extension.



The **Editor** field defines the Windows program that is launched when editing a file. Editing a file involves the following steps: 1) the file is transferred from the NonStop System to the PC using an internal transfer method (FTP is not involved), 2) the Editor is launched and, 3) after exiting the Editor the file is transferred back to the NonStop System (if changed).

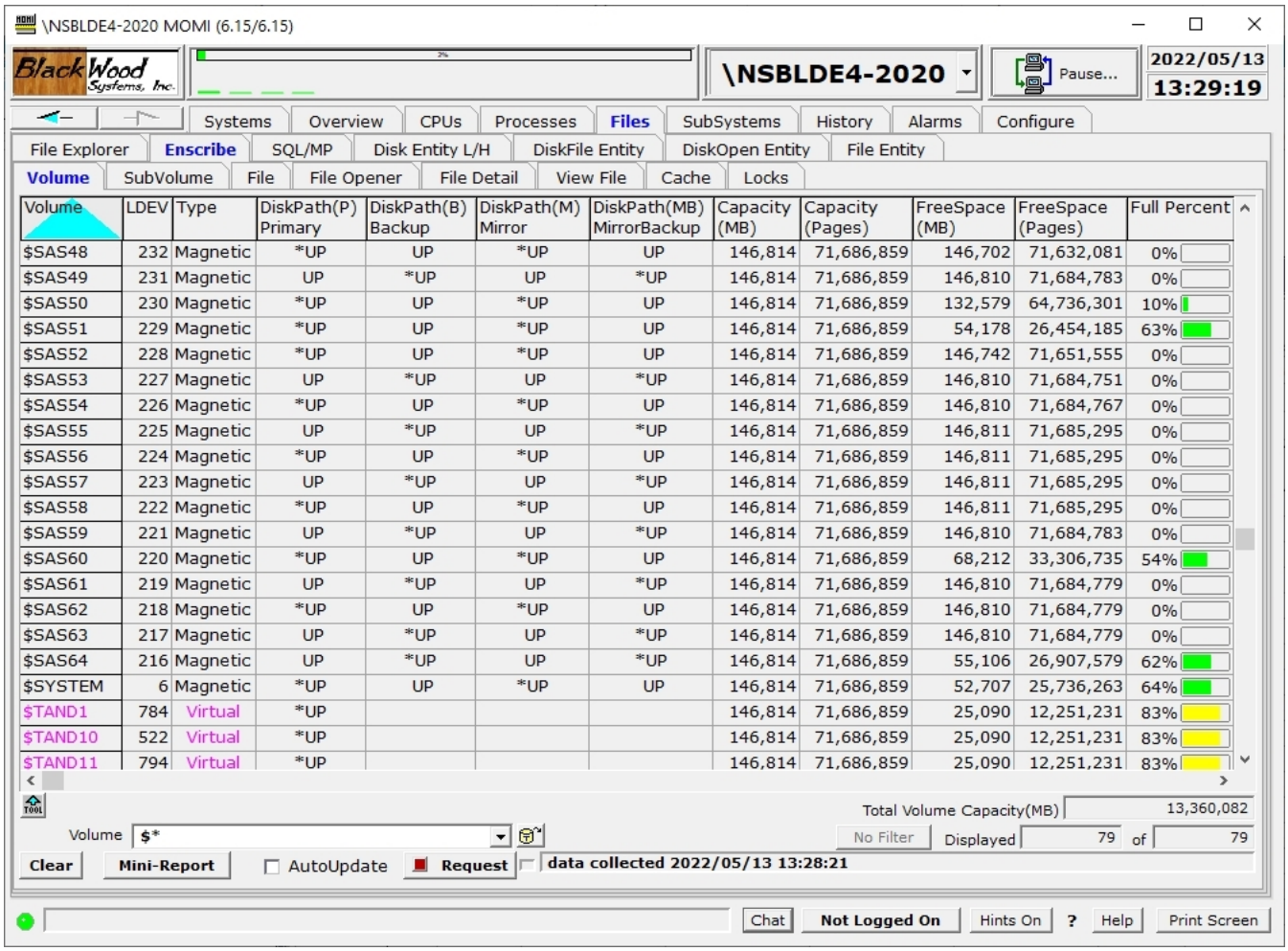
The **DOS** and **UNIX** format selection determines how line termination is formatted for ASCII files. The DOS Format uses a CR/LF and the UNIX Format uses a LF to terminate a line. This selection determines how the file is prepared for the Editor.

Edit On PC Settings selects various sounds played during file transfers and if an error occurs.

The **ASCII File extensions** determine when a file is consider ASCII, or a readable text file, and edit is allowed. Otherwise, the file is considered binary and is not editable.

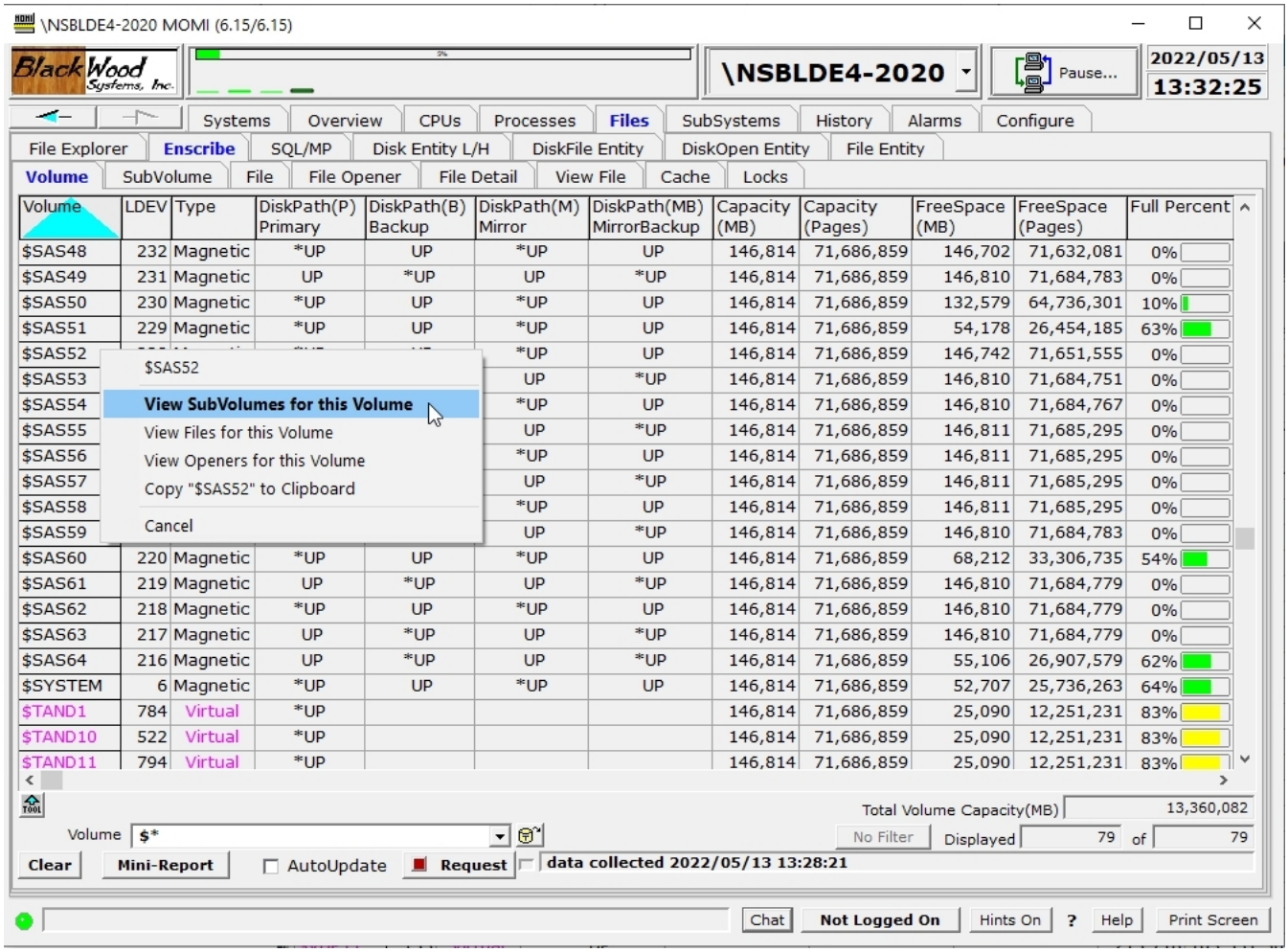
Enscribe

Enscribe / Volume



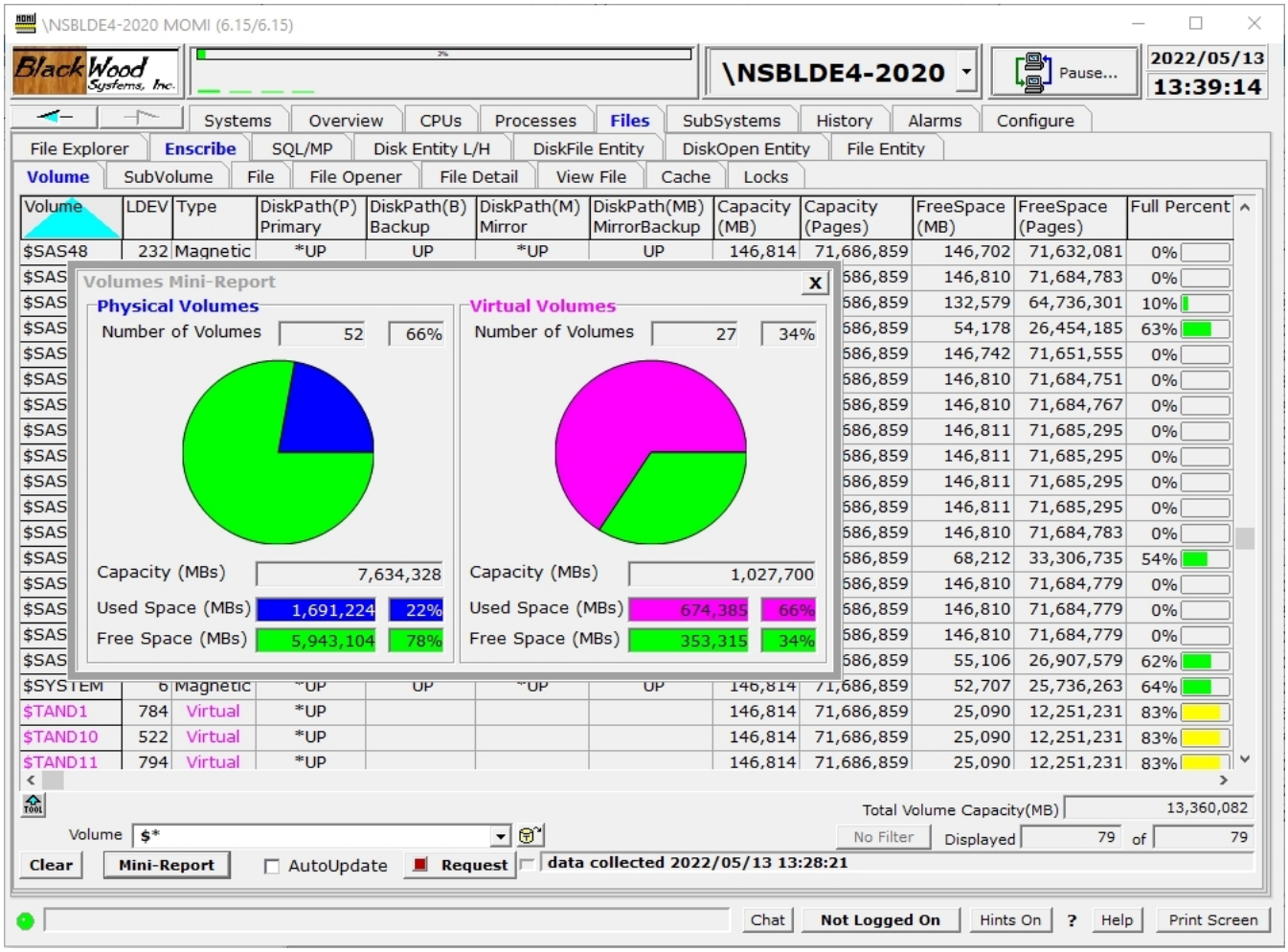
The Enscribe / Volume screen presents the disk present on the System. Drive types are identified as Magnetic (the default), SSD, Virtual and Optical.

Information displayed includes capacity, free space, fragments, if the volume may have TMF audited files, and drive product number. Mirrored volumes display additional information.



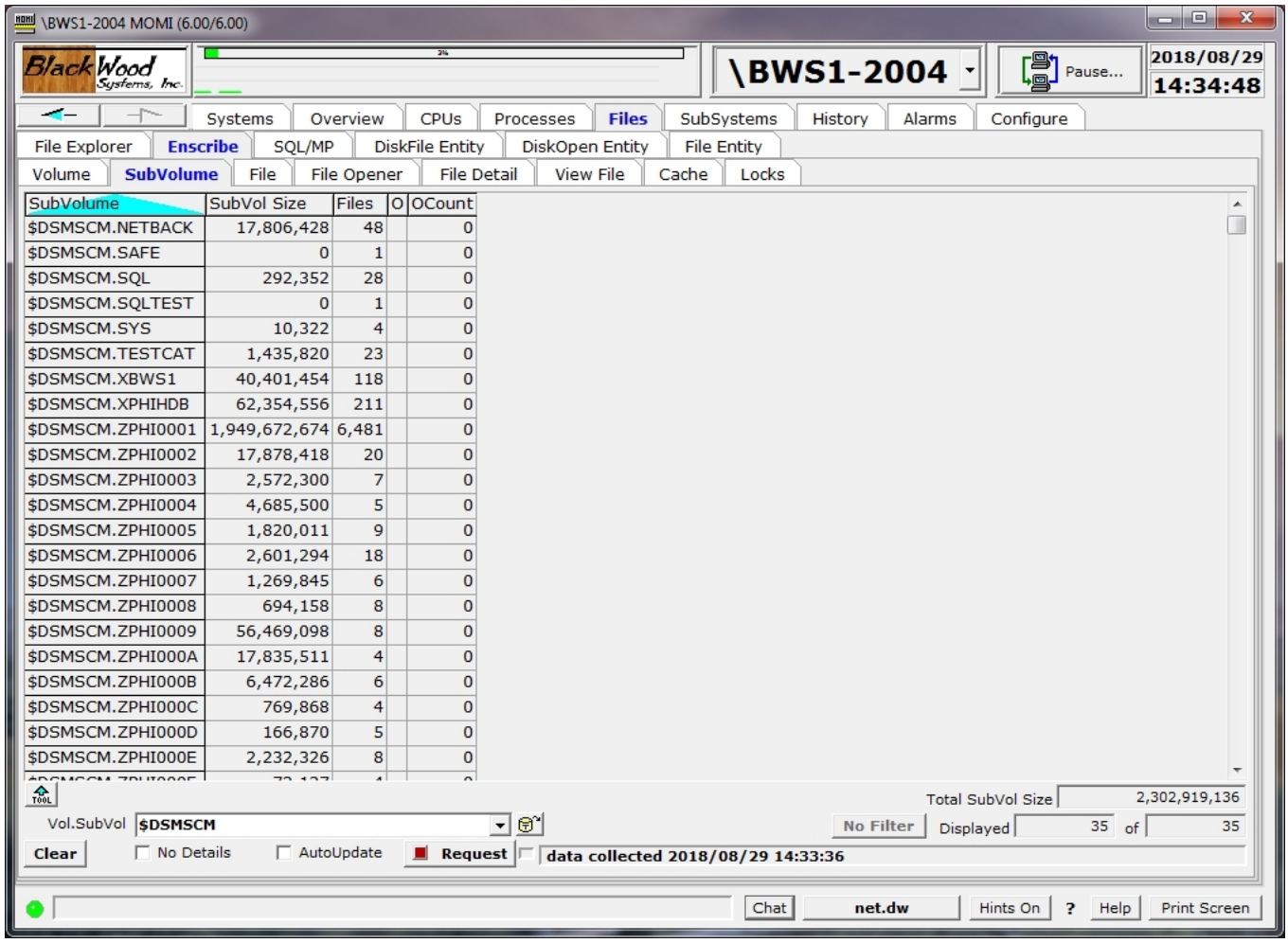
Volume	LDEV	Type	DiskPath(P) Primary	DiskPath(B) Backup	DiskPath(M) Mirror	DiskPath(MB) MirrorBackup	Capacity (MB)	Capacity (Pages)	FreeSpace (MB)	FreeSpace (Pages)	Full Percent
$SAS48	232	Magnetic	*UP	UP	*UP	UP	146,814	71,686,859	146,702	71,632,081	0%
$SAS49	231	Magnetic	UP	*UP	UP	*UP	146,814	71,686,859	146,810	71,684,783	0%
$SAS50	230	Magnetic	*UP	UP	*UP	UP	146,814	71,686,859	132,579	64,736,301	10%
$SAS51	229	Magnetic	*UP	UP	*UP	UP	146,814	71,686,859	54,178	26,454,185	63%
$SAS52					*UP	UP	146,814	71,686,859	146,742	71,651,555	0%
$SAS53					UP	*UP	146,814	71,686,859	146,810	71,684,751	0%
$SAS54					*UP	UP	146,814	71,686,859	146,810	71,684,767	0%
$SAS55					UP	*UP	146,814	71,686,859	146,811	71,685,295	0%
$SAS56					*UP	UP	146,814	71,686,859	146,811	71,685,295	0%
$SAS57					UP	*UP	146,814	71,686,859	146,811	71,685,295	0%
$SAS58					*UP	UP	146,814	71,686,859	146,811	71,685,295	0%
$SAS59					UP	*UP	146,814	71,686,859	146,810	71,684,783	0%
$SAS60	220	Magnetic	*UP	UP	*UP	UP	146,814	71,686,859	68,212	33,306,735	54%
$SAS61	219	Magnetic	UP	*UP	UP	*UP	146,814	71,686,859	146,810	71,684,779	0%
$SAS62	218	Magnetic	*UP	UP	*UP	UP	146,814	71,686,859	146,810	71,684,779	0%
$SAS63	217	Magnetic	UP	*UP	UP	*UP	146,814	71,686,859	146,810	71,684,779	0%
$SAS64	216	Magnetic	UP	*UP	UP	*UP	146,814	71,686,859	55,106	26,907,579	62%
$SYSTEM	6	Magnetic	*UP	UP	*UP	UP	146,814	71,686,859	52,707	25,736,263	64%
$TAND1	784	Virtual	*UP				146,814	71,686,859	25,090	12,251,231	83%
$TAND10	522	Virtual	*UP				146,814	71,686,859	25,090	12,251,231	83%
$TAND11	794	Virtual	*UP				146,814	71,686,859	25,090	12,251,231	83%

The screen supports drill down by right clicking on a drive to bring up a selection of jump options.



Pressing **Mini-Report**, after data is displayed, provides a summery of overall available storage capacity, space used, and space free for the physical and virtual volumes.

Enscribe / SubVolume



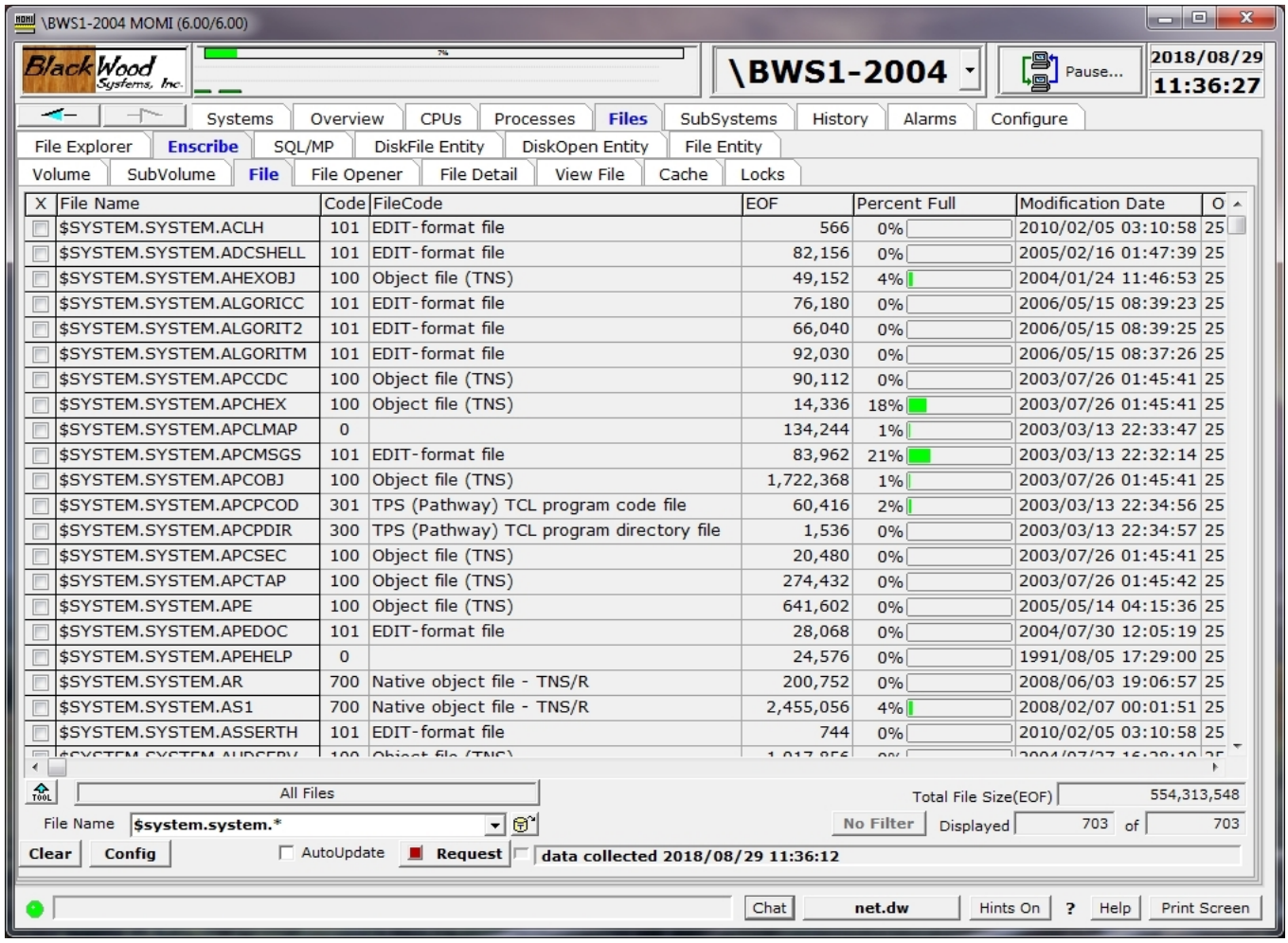
The Enscribe / SubVolume screen presents a list of subvolumes following the Guardian file naming convention.

The total number of bytes of the files comprising the subvolume, number of files present, if any files are open and the count of open files are displayed.

The option **No Details** directs that statistics are not gathered which speeds processing.

Temporary files on a disk are referenced as $vol.#*. The search specification of $*.#* displays all volumes with temporary disk files.

Enscribe / File

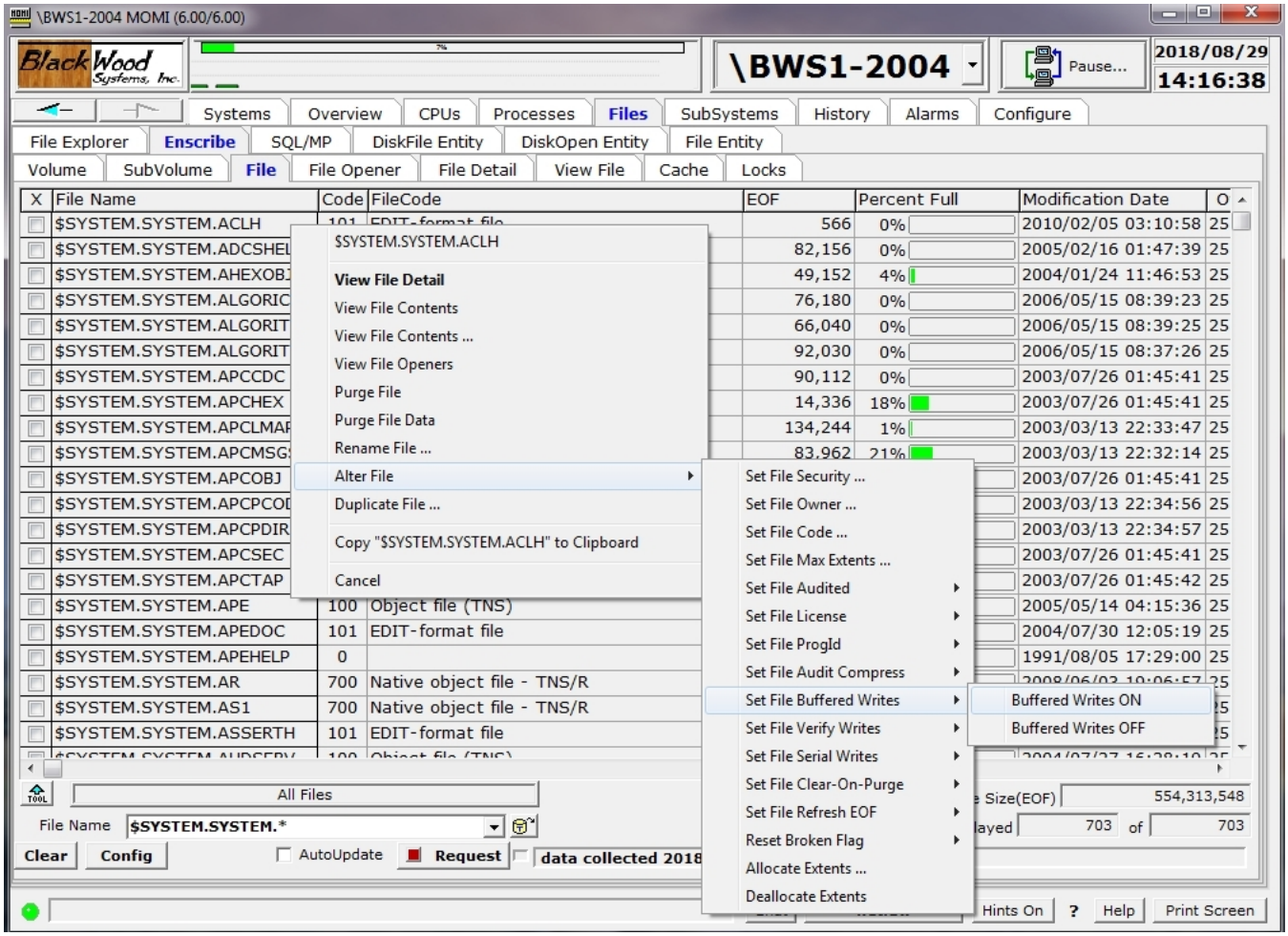


X	File Name	Code	FileCode	EOF	Percent Full	Modification Date	O
	$SYSTEM.SYSTEM.ACLH	101	EDIT-format file	566	0%	2010/02/05 03:10:58	25
	$SYSTEM.SYSTEM.ADCSHELL	101	EDIT-format file	82,156	0%	2005/02/16 01:47:39	25
	$SYSTEM.SYSTEM.AHEXOBJ	100	Object file (TNS)	49,152	4%	2004/01/24 11:46:53	25
	$SYSTEM.SYSTEM.ALGORICC	101	EDIT-format file	76,180	0%	2006/05/15 08:39:23	25
	$SYSTEM.SYSTEM.ALGORIT2	101	EDIT-format file	66,040	0%	2006/05/15 08:39:25	25
	$SYSTEM.SYSTEM.ALGORITM	101	EDIT-format file	92,030	0%	2006/05/15 08:37:26	25
	$SYSTEM.SYSTEM.APCCDC	100	Object file (TNS)	90,112	0%	2003/07/26 01:45:41	25
	$SYSTEM.SYSTEM.APCHEX	100	Object file (TNS)	14,336	18%	2003/07/26 01:45:41	25
	$SYSTEM.SYSTEM.APCLMAP	0		134,244	1%	2003/03/13 22:33:47	25
	$SYSTEM.SYSTEM.APCMSGS	101	EDIT-format file	83,962	21%	2003/03/13 22:32:14	25
	$SYSTEM.SYSTEM.APCOBJ	100	Object file (TNS)	1,722,368	1%	2003/07/26 01:45:41	25
	$SYSTEM.SYSTEM.APCPCOD	301	TPS (Pathway) TCL program code file	60,416	2%	2003/03/13 22:34:56	25
	$SYSTEM.SYSTEM.APCPDIR	300	TPS (Pathway) TCL program directory file	1,536	0%	2003/03/13 22:34:57	25
	$SYSTEM.SYSTEM.APCSEC	100	Object file (TNS)	20,480	0%	2003/07/26 01:45:41	25
	$SYSTEM.SYSTEM.APCTAP	100	Object file (TNS)	274,432	0%	2003/07/26 01:45:42	25
	$SYSTEM.SYSTEM.APE	100	Object file (TNS)	641,602	0%	2005/05/14 04:15:36	25
	$SYSTEM.SYSTEM.APEDOC	101	EDIT-format file	28,068	0%	2004/07/30 12:05:19	25
	$SYSTEM.SYSTEM.APEHELP	0		24,576	0%	1991/08/05 17:29:00	25
	$SYSTEM.SYSTEM.AR	700	Native object file - TNS/R	200,752	0%	2008/06/03 19:06:57	25
	$SYSTEM.SYSTEM.AS1	700	Native object file - TNS/R	2,455,056	4%	2008/02/07 00:01:51	25
	$SYSTEM.SYSTEM.ASSERTH	101	EDIT-format file	744	0%	2010/02/05 03:10:58	25

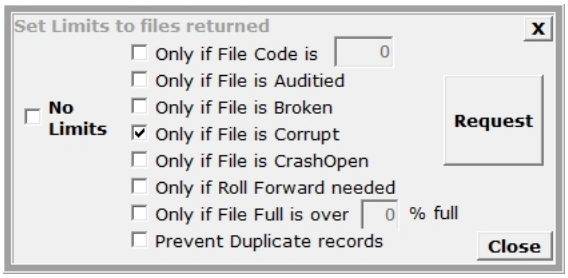
The Enscribe / File screen presents information about disk files.

Files listed may also include OSS files. The path name is listed in a column further to the right on the scroll bar.

Actions available may be taken by right-clicking on the file name.



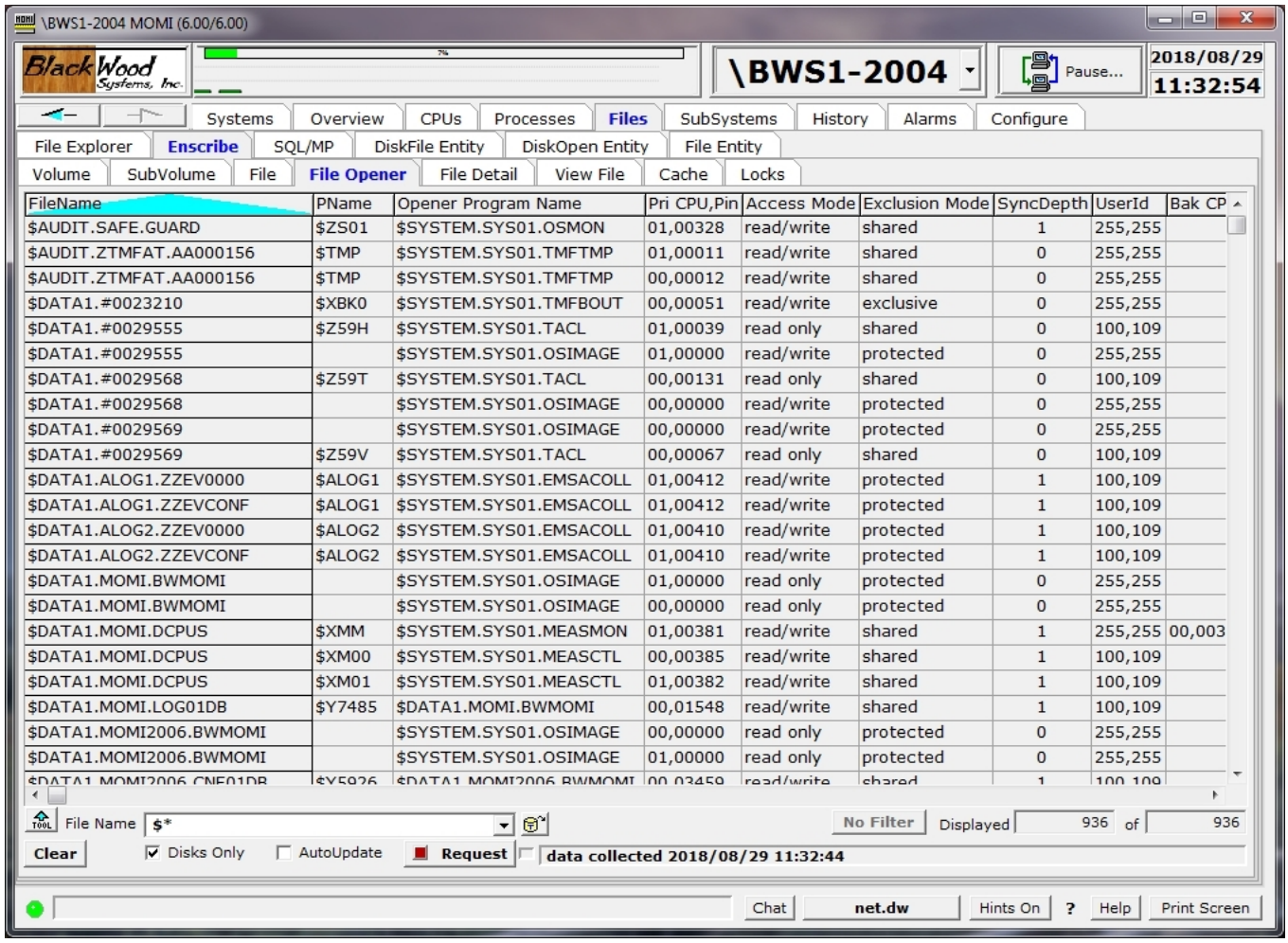
To obtain a limited set of files, the **Config** button provides several options:



The files retrieved are restricted by any option enabled and may result in faster processing as only those records with matching criteria are returned to the Client. The above example requests only files with the Corrupt attribute. If multiple limits are enabled everything checked must match (i.e. evaluates to true).

OSS files have a different structure from Enscribe files. While the underlying Guardian files are displayed by this screen, columns such as "% full" and "maximum extents" may not apply or have strange information displayed.

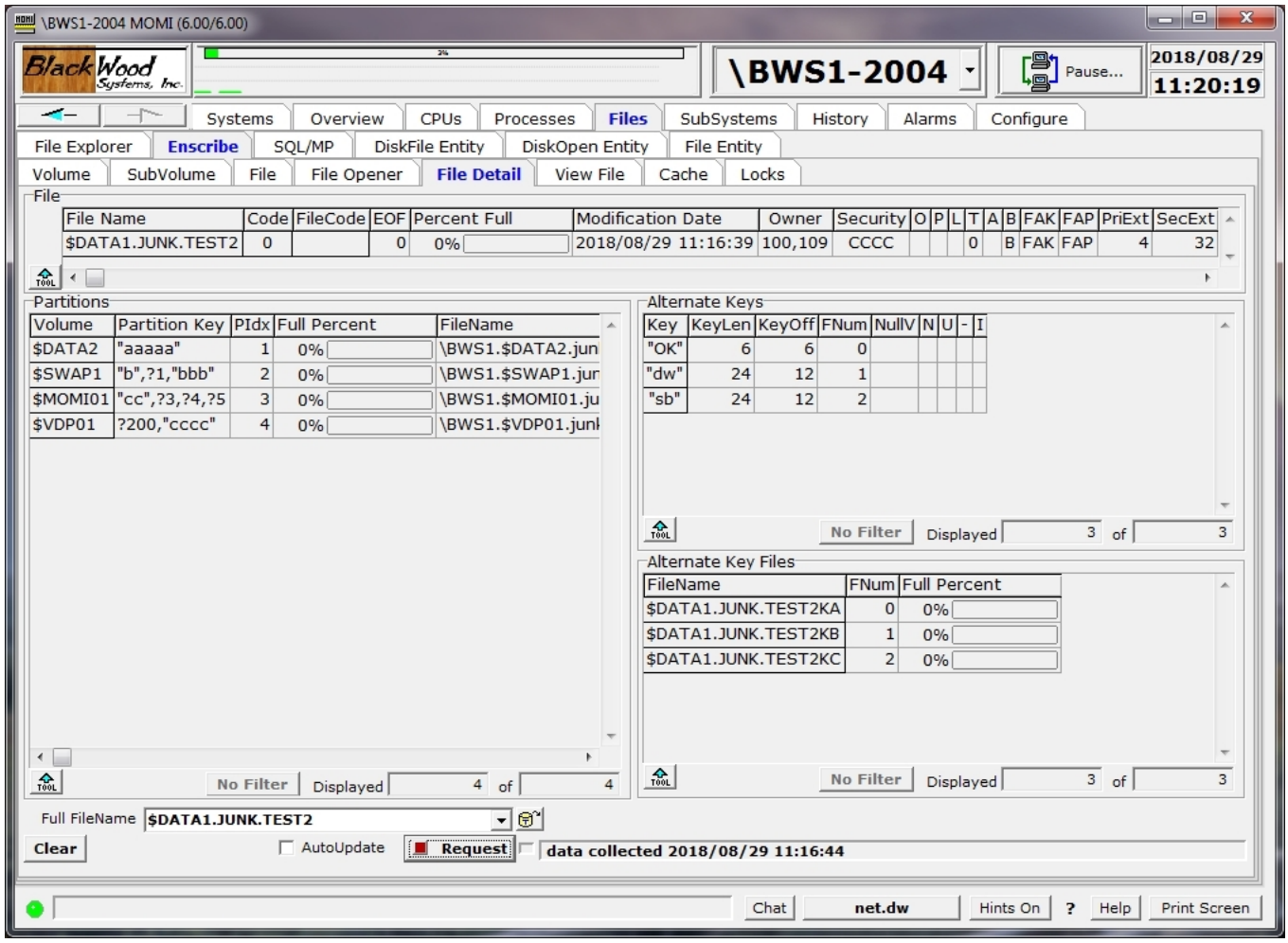
Enscribe / File Opener



The Enscribe / File Opener screen presents a list of 'opens' (the establishment of a communication path) from one entity to another. This is similar to the FUP LISTOPENS command.

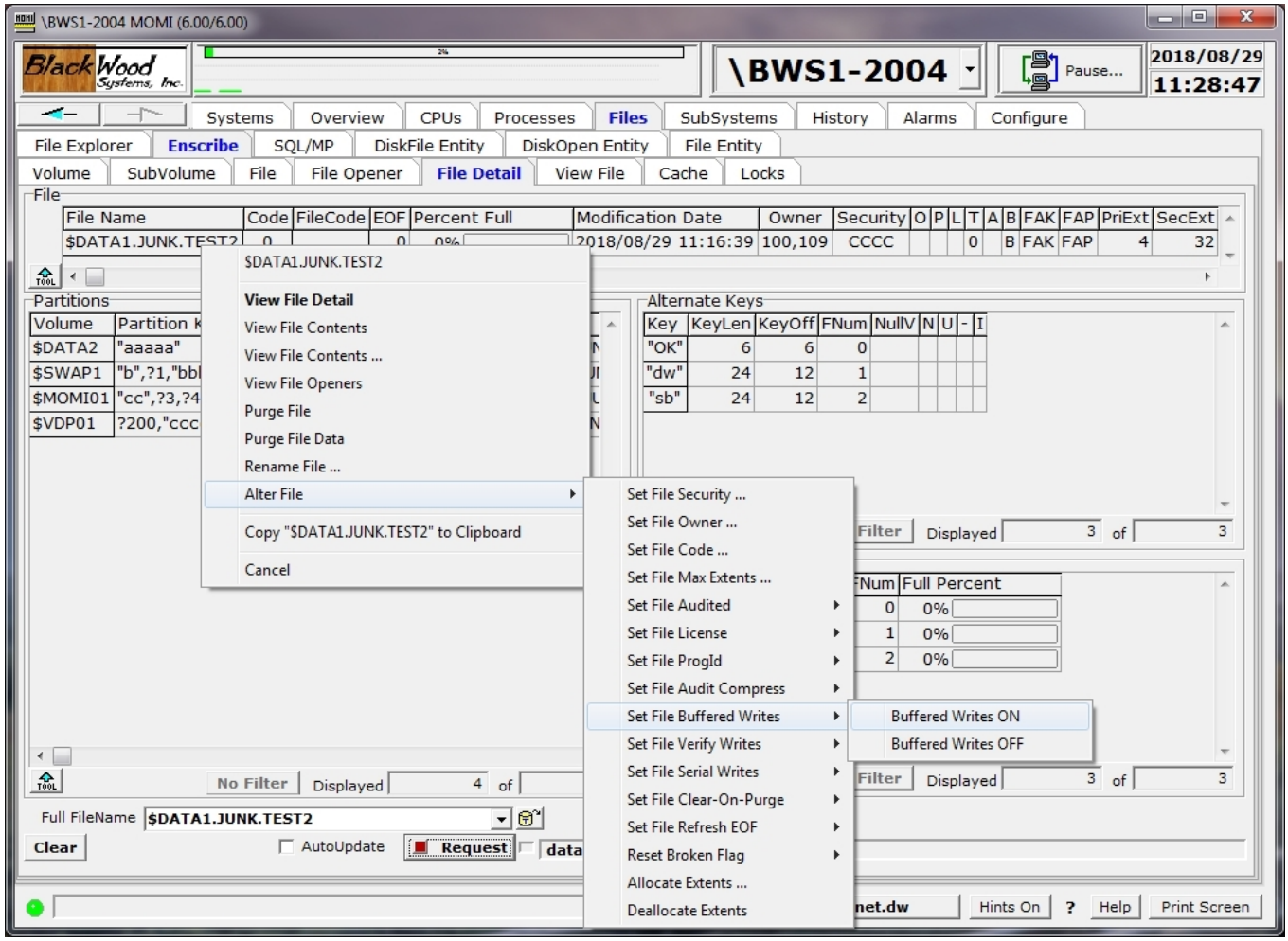
Generally, only System processes such as the disk process provide opener information.

Enscribe / File Detail



The Enscribe / File Detail screen presents details of the Partitions, Alternate Keys and Alternate Key Files associated with an Enscribe file.

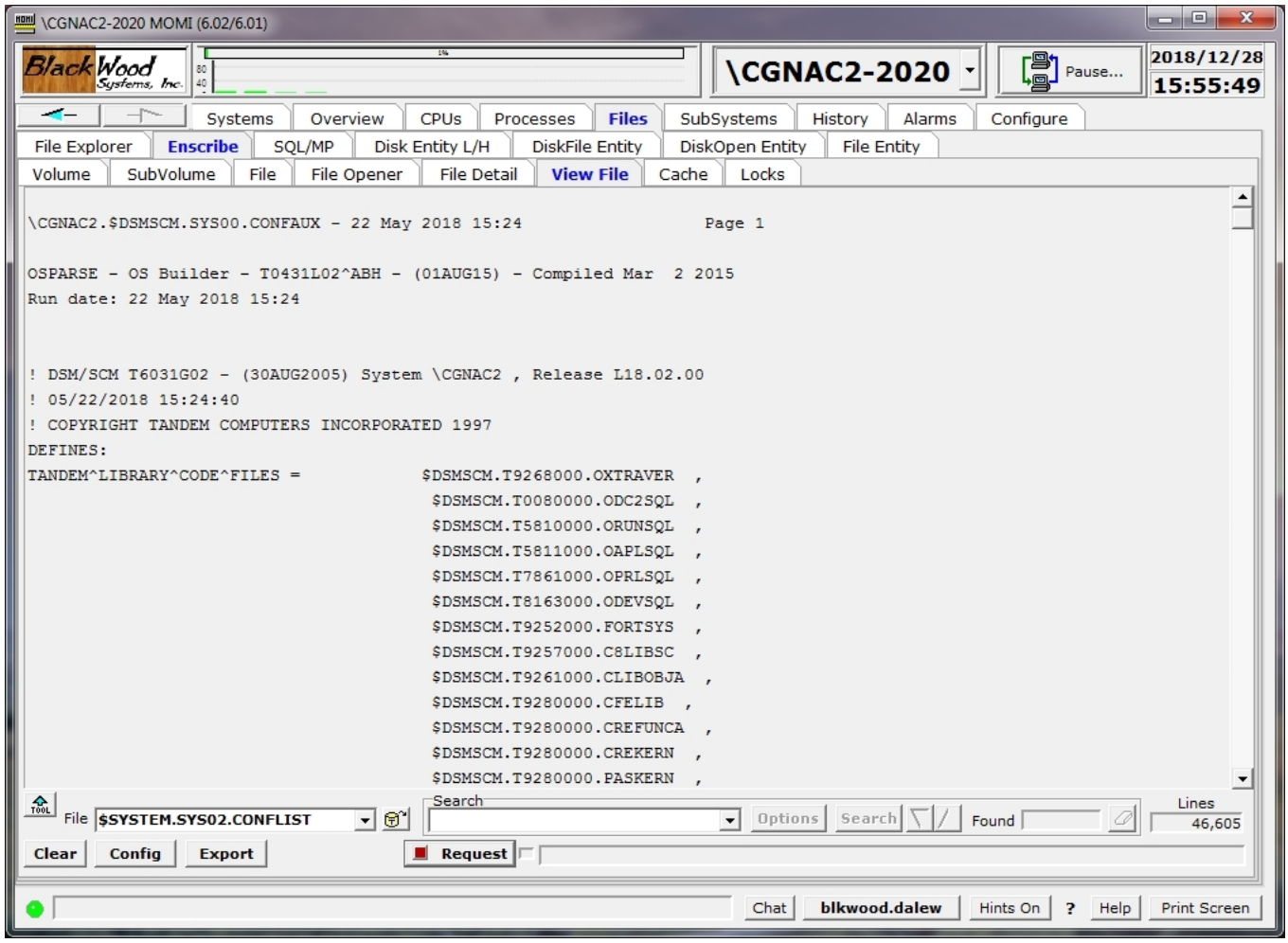
File level detail is the same as presented on the Enscribe / Files screen.



Right click actions are available on the File, Partitions and Alternate Key files.

SQL and OSS files may not display correctly as the Guardian procedures do not always return correct information these types of files.

Enscribe / View File

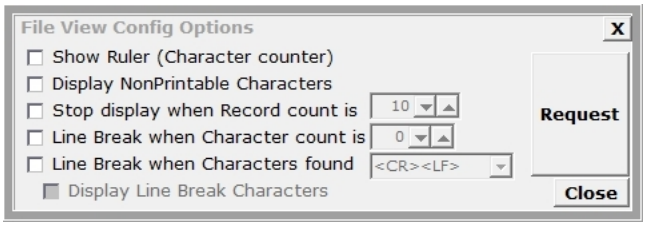


The Enscribe / View File screen allows display of the contents of a disk file. The User must Logon in order to perform action on this screen.

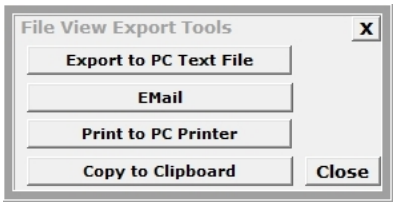
The file selected may either be an edit type or other type of file (such as an object file). By default, null characters are converted to ~ (tilde).

Object files, file codes 100, 700 and 800, have a fixed line length of 132 bytes. Edit files have variable length lines. All other files have a line length determined by the file record size.

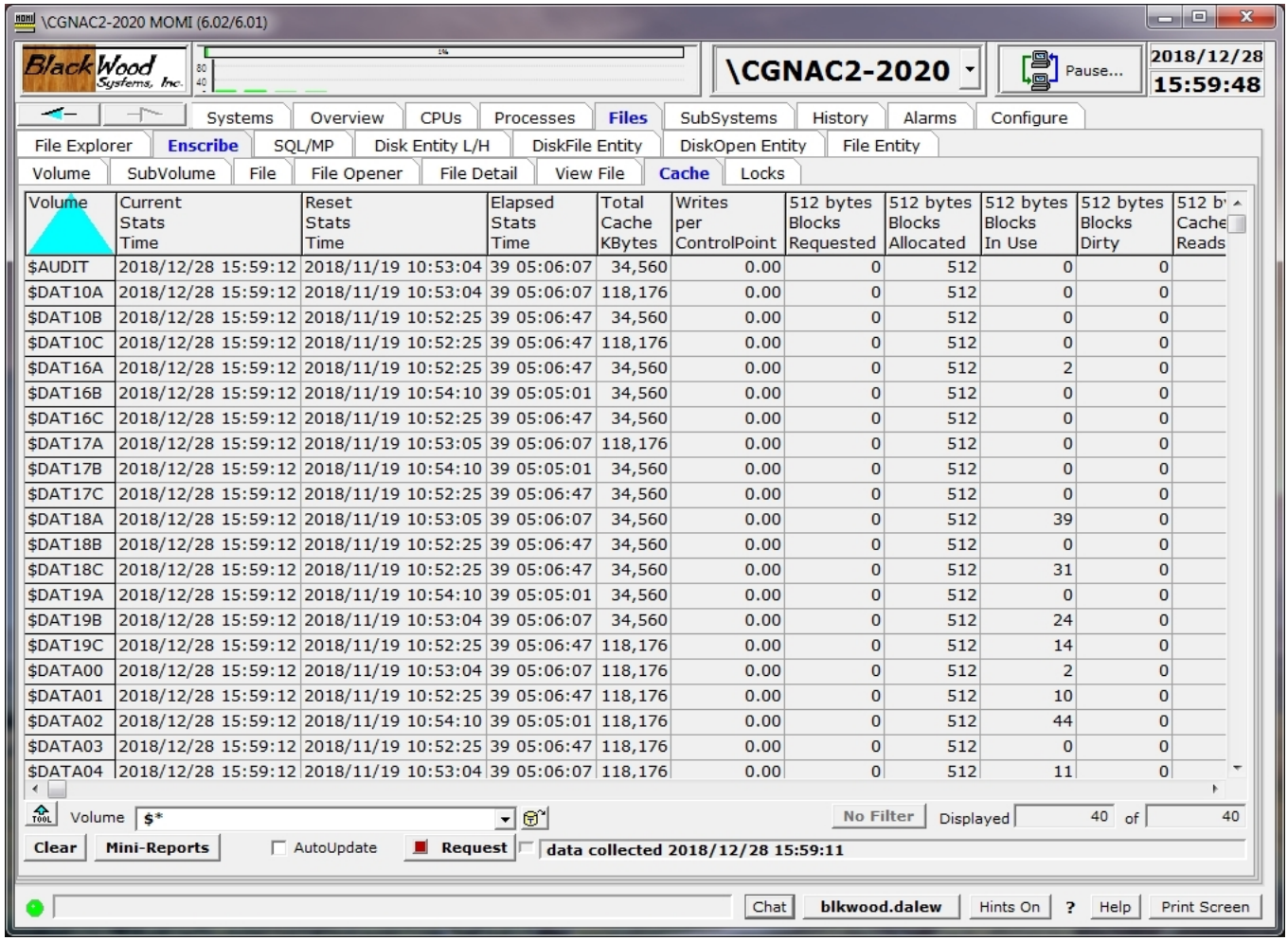
The Config button display a pop-up window to alter screen defaults, such as not converting unprintable ASCII characters to ~ (tilde). Unstructured files with fixed length lines or lines terminated by particular characters (such as CR/LF) may also be processed.



The Export button is used to transfer the displayed information to a file, printer on the local PC or via an Email.



Enscribe / Cache



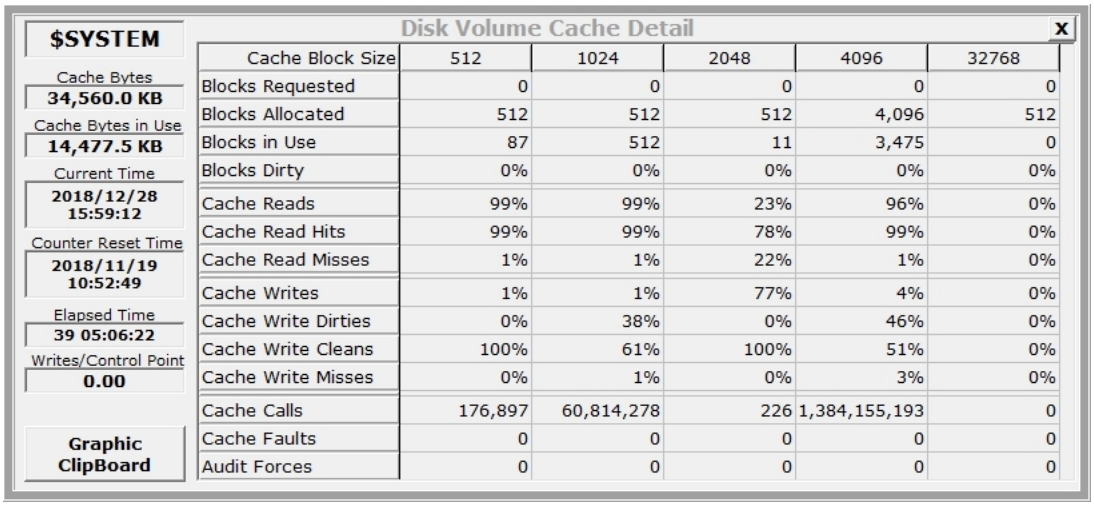
The Enscribe / Cache screen presents the cache allocated and statistics about cache efficiency.

Disk cache is memory allocated in the disk process that is referenced prior to a physical access to disk. Data found in the cache reduces physical disk access and is inherently faster.

Disk cache is divided into 4 groups or block sizes. A block size is specified when a disk file is created. Cache blocks are allocated to each size. The sum of these blocks represents the Total Cache size.

A high percentage of "Cache Read Hits" and "Cache Write Dirties" are desired. Generally, additional cache blocks are allocated (via SCF) until percentages in the 90's are achieved. However, Users should also give attention to the Cache Faults counter. If a processor is running low on memory, the memory manager can take memory allocated to the disk cache. This counter indicates the frequency of this event. Consideration should be given to reducing the amount of disk cache or increasing memory in a processor when cache faults is constantly incrementing.

Clicking on a disk will bring up an SCF-like output containing detail for that volume:



$SYSTEM

Cache Bytes
34,560.0 KB

Cache Bytes in Use
14,477.5 KB

Current Time
2018/12/28 15:59:12

Counter Reset Time
2018/11/19 10:52:49

Elapsed Time
39 05:06:22

Writes/Control Point
0.00

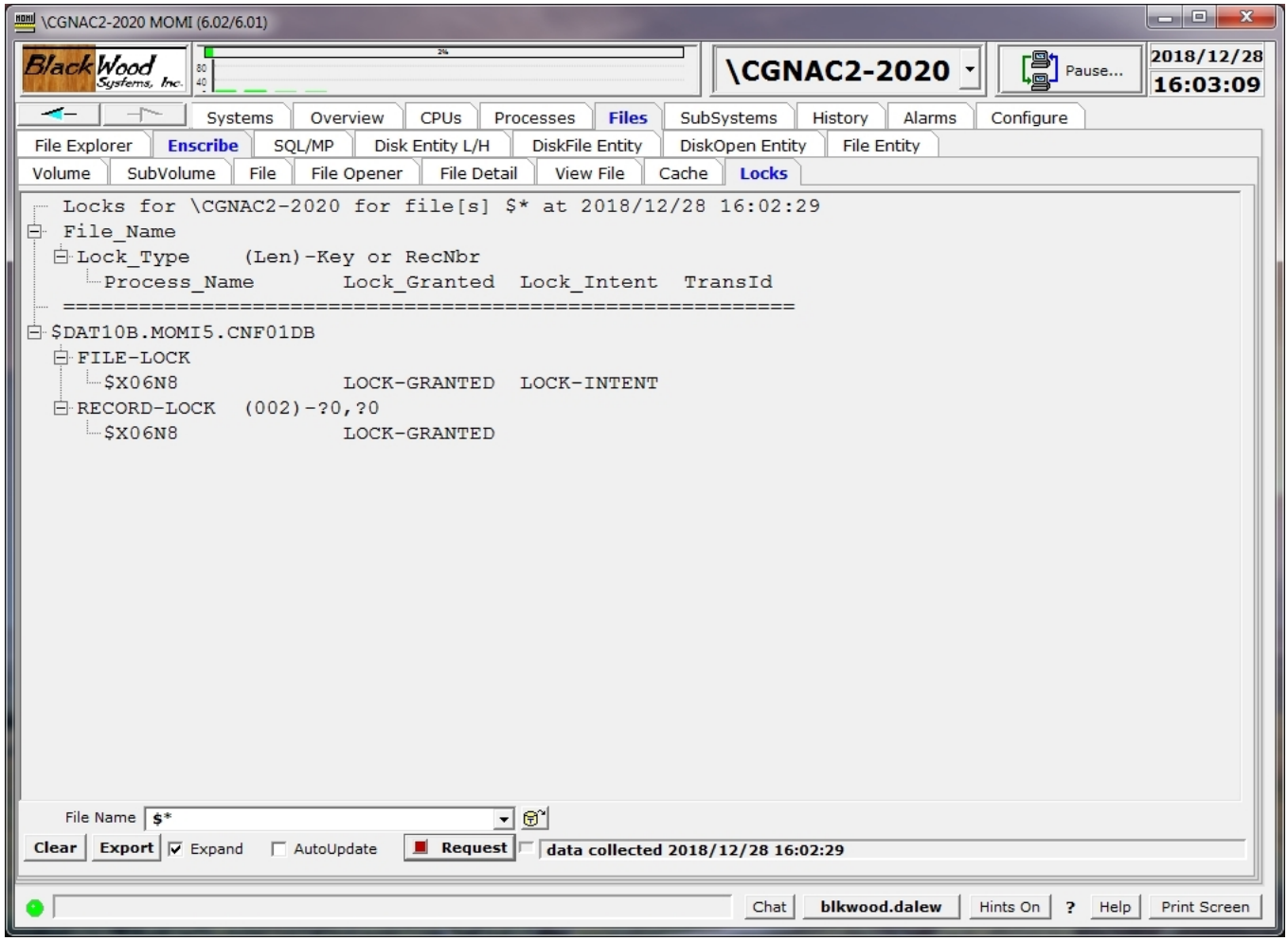
Graphic ClipBoard

Disk Volume Cache Detail

Cache Block Size	512	1024	2048	4096	32768
Blocks Requested	0	0	0	0	0
Blocks Allocated	512	512	512	4,096	512
Blocks in Use	87	512	11	3,475	0
Blocks Dirty	0%	0%	0%	0%	0%
Cache Reads	99%	99%	23%	96%	0%
Cache Read Hits	99%	99%	78%	99%	0%
Cache Read Misses	1%	1%	22%	1%	0%
Cache Writes	1%	1%	77%	4%	0%
Cache Write Dirties	0%	38%	0%	46%	0%
Cache Write Cleans	100%	61%	100%	51%	0%
Cache Write Misses	0%	1%	0%	3%	0%
Cache Calls	176,897	60,814,278	226	1,384,155,193	0
Cache Faults	0	0	0	0	0
Audit Forces	0	0	0	0	0

Enscribe / Locks

see System Quirks



The Enscribe / Locks screen presents file and record lock information similar to the FUP LISTLOCKS command.

This screen only functions on systems running G06.26 or later.

The information is presented in a tree-view format. The first level is the file name, the second level a file level or record level type of lock and the third level the locking process.

The following are descriptions of the columns:

```
File_Name    = File Name that has the Lock
Lock_Type    = Type of Lock (File or Record)
(Len)-Key    = (length of Key)-Key Value for Key-Sequenced files or
RecNbr       = Record Number for non-Key-Sequenced files
Process_Name = Name of Process that requested the Lock or
                 process that initiated the TMF transaction
Lock_Granted = Lock is Granted
Lock_Intent  = Lock Intent requested by DP2
TransId      = Transaction ID (if TMF active on participant)
```

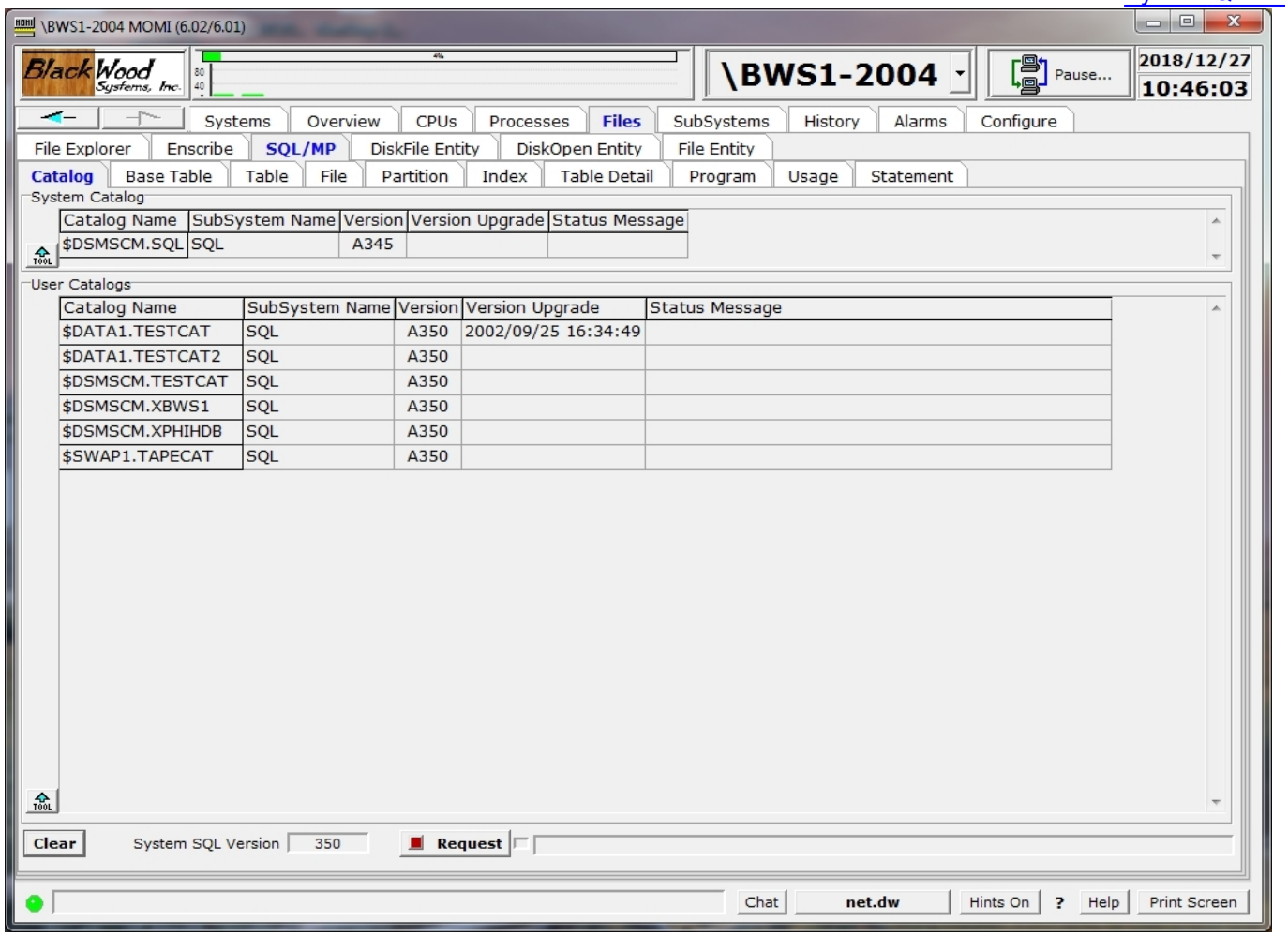
The underlying Guardian procedure to return locking information does not include the requesting process name if a Transaction ID is returned. MOMI queries TMF and requests the process name that initiates the TMF transaction. This is not necessarily the process performing I/O to the listed file.

If a lock cannot be granted, indicated by a space in the Lock_Granted column, then the request is placed in a queue until the lock is granted or the requesting process times out.

SQL/MP

SQL/MP / Catalog

see System Quirks



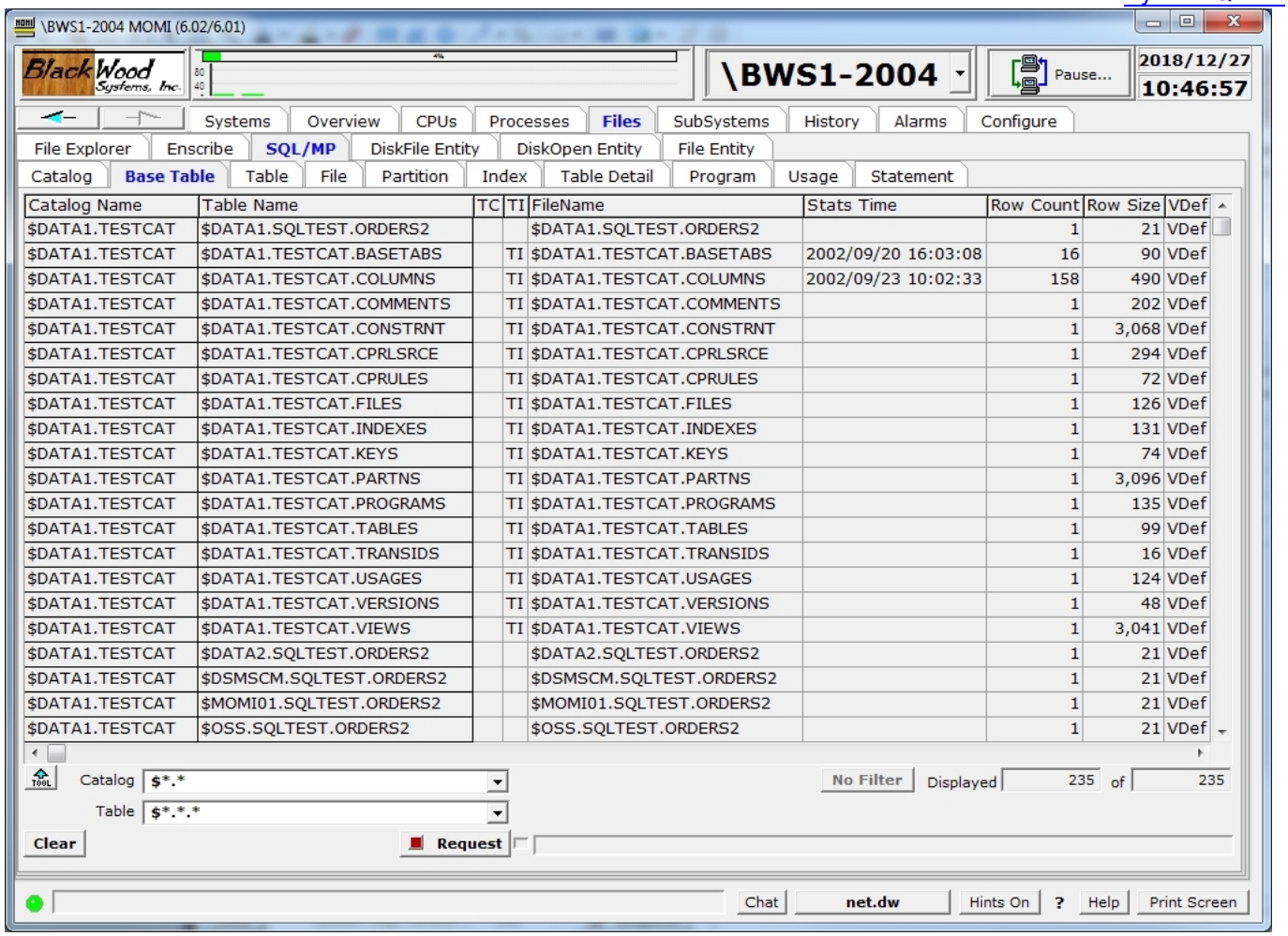
The SQL/MP / Catalog screen presents the SQL/MP Catalog of the System and User Catalogs defined on the system.

The NonStop System itself has a version number for SQL/MP and is shown in the lower left hand corner of the display. The System Catalog and User Catalogs are usually at or below the version of SQL/MP of the system.

The column Status Message displays an error message if a simple query of the listed User Catalogs fails. An error in this column either indicates that MOMI does not have security access to the catalog or that the catalog is not present. Any error message in this column indicates that MOMI will encounter errors on other SQL/MP screens.

SQL/MP / Base Table

see System Quirks

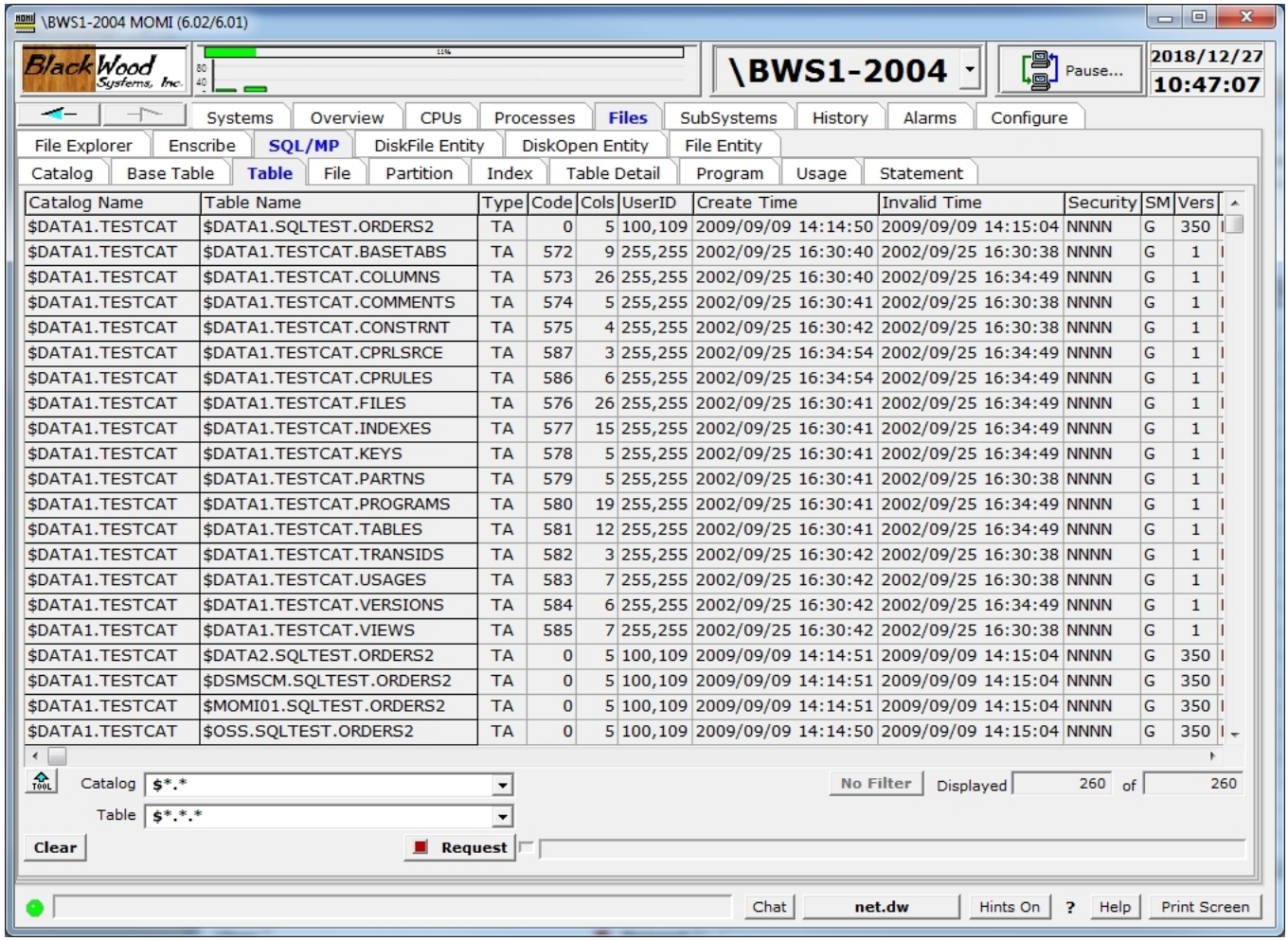


The SQL/MP / Base Table screen presents information about base tables.

Base tables are physically present and are not a logical 'view' of the data. SQL tables are made up of rows and columns where a row can be thought of as a record and a column is a field in a record.

SQL/MP / Table

see System Quirks

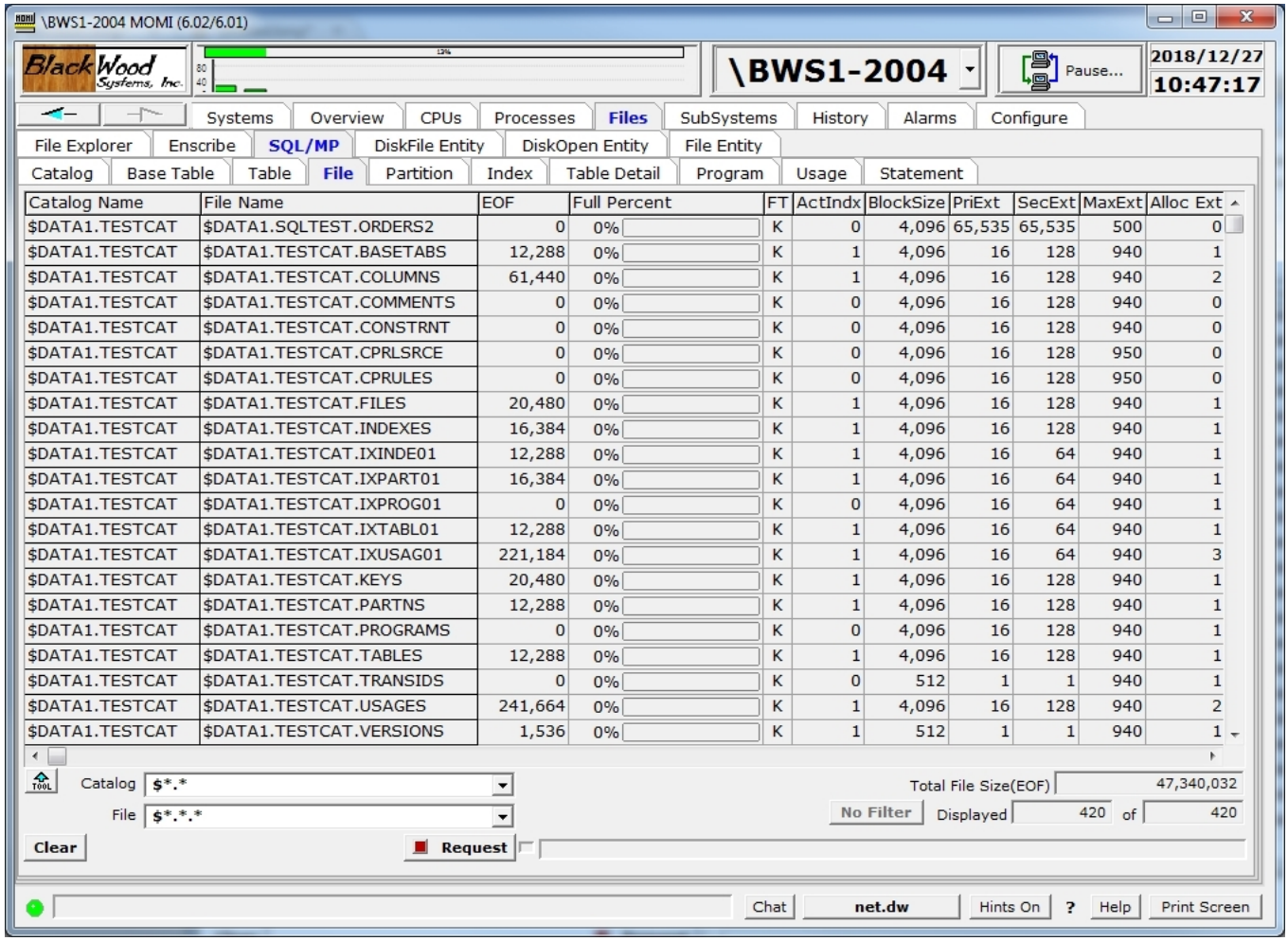


The SQL/MP / Table screen presents the TABLES table which is a catalog table that contains information about tables, views, and collations.

The TABLES table contains a row for each table, view, and collation in the catalog, including itself and other catalog tables.

SQL/MP / File

see System Quirks



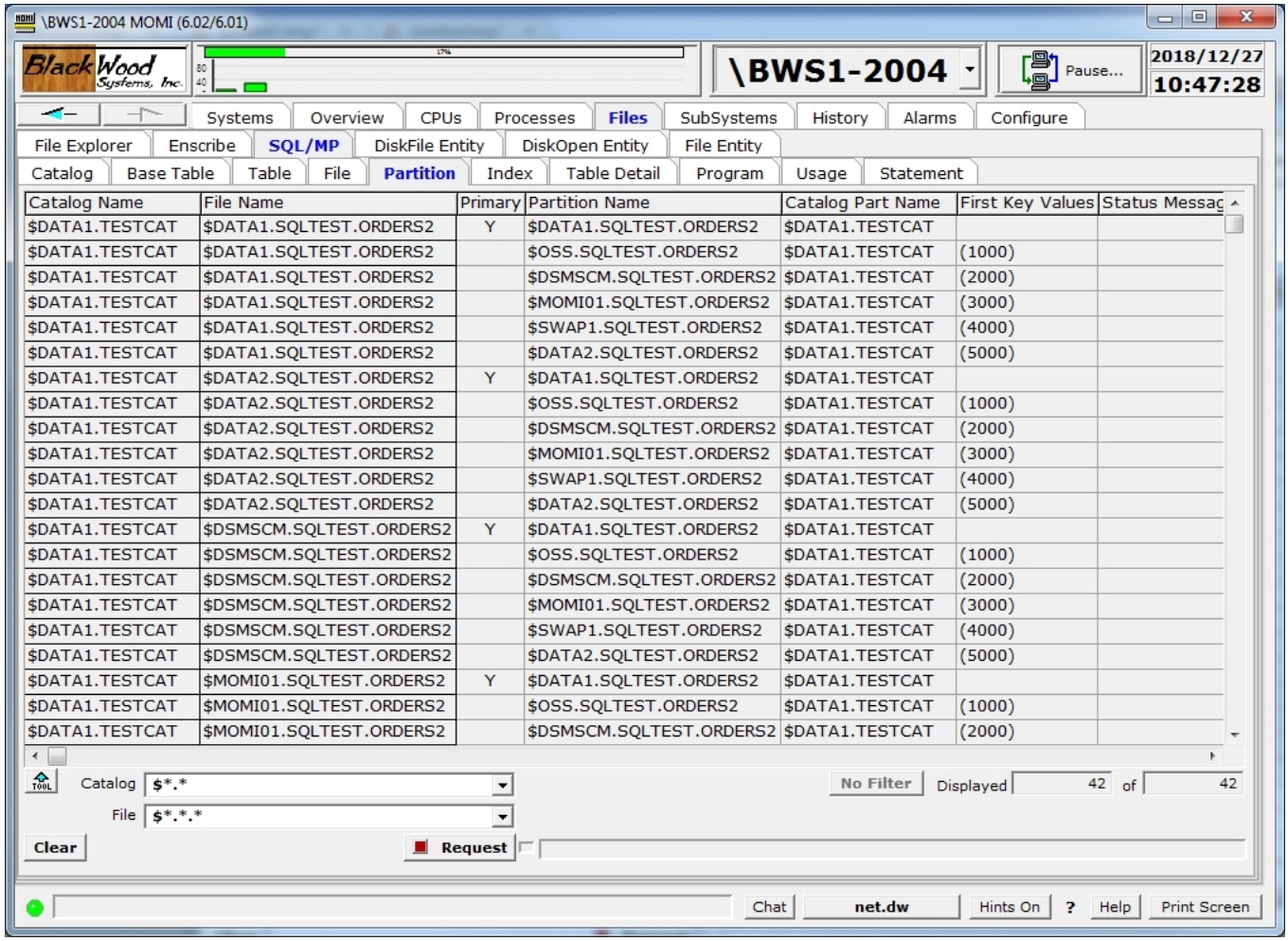
The SQL/MP / File screen presents attributes of files that contain tables and indexes.

A file is the physical entity where data or descriptive information is stored.

The Actual Index (ActIndx) column displays the number of index levels in the file at the time of the screen display. This value does not come from the catalog. A high number of index levels can affect System performance. Sorting the screen by this column allows index levels to be quickly checked.

SQL/MP / Partition

see System Quirks

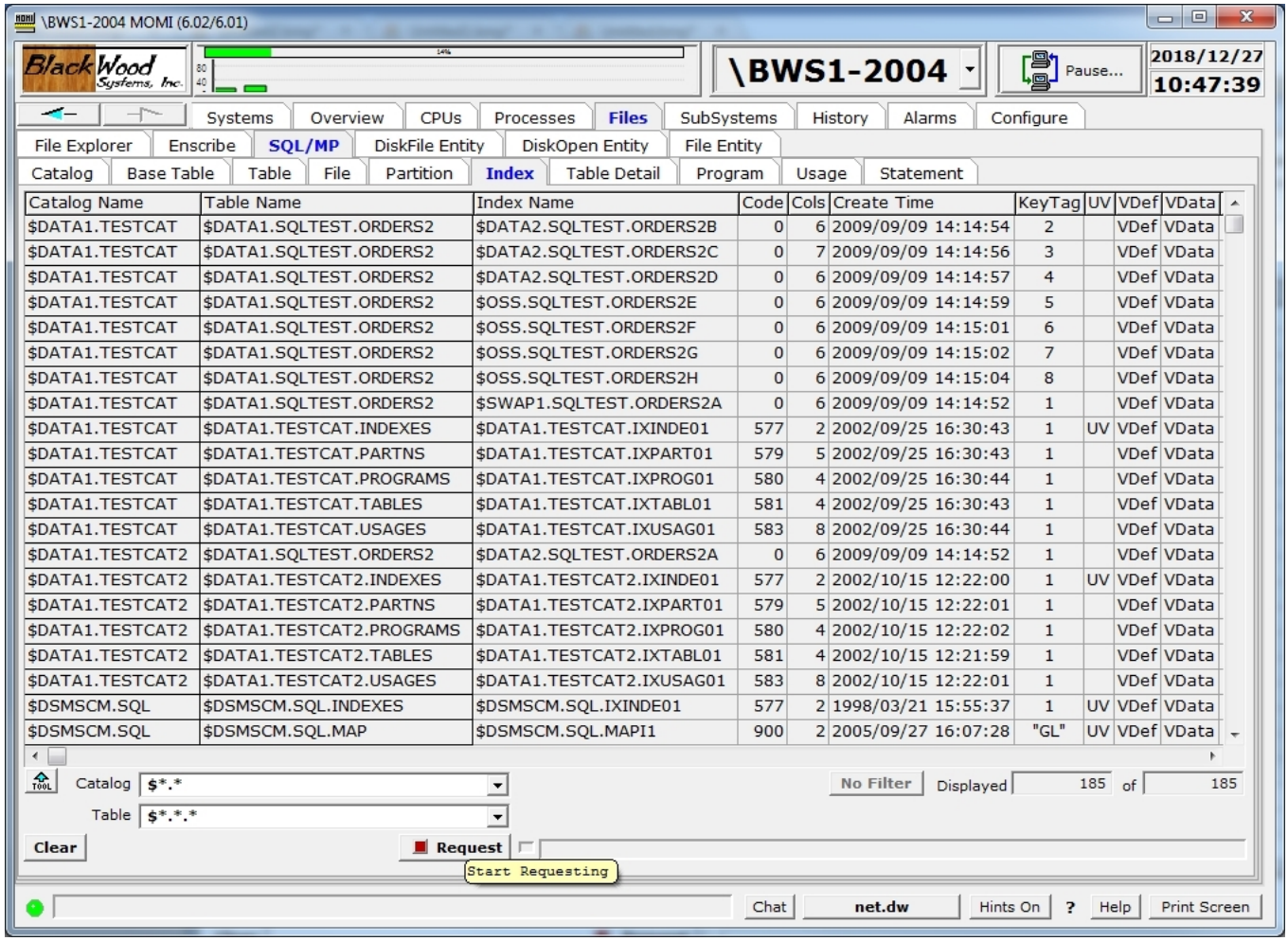


The SQL/MP / Partition screen presents information about how large logical files are split across multiple smaller physical files.

As files grow in either physical size or number of accesses, partitions are used to spread the load across multiple disk drives. Multiple drives provide more physical space and/or better access time by overlapping I/O across more hardware.

SQL/MP / Index

see System Quirks

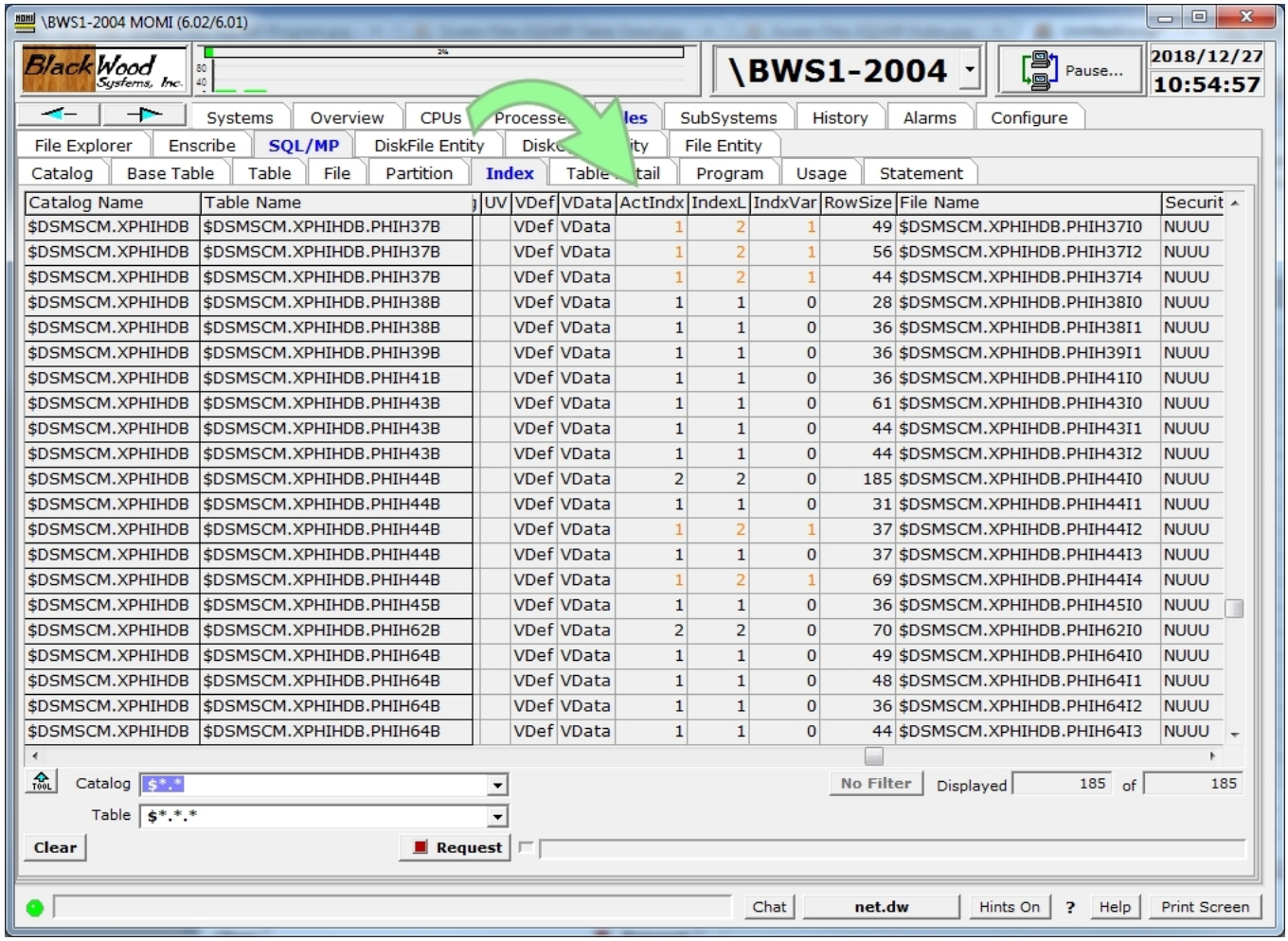


Catalog Name	Table Name	Index Name	Code	Cols	Create Time	KeyTag	UV	VDef	VData
$DATA1.TESTCAT	$DATA1.SQLTEST.ORDERS2	$DATA2.SQLTEST.ORDERS2B	0	6	2009/09/09 14:14:54	2		VDef	VData
$DATA1.TESTCAT	$DATA1.SQLTEST.ORDERS2	$DATA2.SQLTEST.ORDERS2C	0	7	2009/09/09 14:14:56	3		VDef	VData
$DATA1.TESTCAT	$DATA1.SQLTEST.ORDERS2	$DATA2.SQLTEST.ORDERS2D	0	6	2009/09/09 14:14:57	4		VDef	VData
$DATA1.TESTCAT	$DATA1.SQLTEST.ORDERS2	$OSS.SQLTEST.ORDERS2E	0	6	2009/09/09 14:14:59	5		VDef	VData
$DATA1.TESTCAT	$DATA1.SQLTEST.ORDERS2	$OSS.SQLTEST.ORDERS2F	0	6	2009/09/09 14:15:01	6		VDef	VData
$DATA1.TESTCAT	$DATA1.SQLTEST.ORDERS2	$OSS.SQLTEST.ORDERS2G	0	6	2009/09/09 14:15:02	7		VDef	VData
$DATA1.TESTCAT	$DATA1.SQLTEST.ORDERS2	$OSS.SQLTEST.ORDERS2H	0	6	2009/09/09 14:15:04	8		VDef	VData
$DATA1.TESTCAT	$DATA1.SQLTEST.ORDERS2	$SWAP1.SQLTEST.ORDERS2A	0	6	2009/09/09 14:14:52	1		VDef	VData
$DATA1.TESTCAT	$DATA1.TESTCAT.INDEXES	$DATA1.TESTCAT.IXINDE01	577	2	2002/09/25 16:30:43	1	UV	VDef	VData
$DATA1.TESTCAT	$DATA1.TESTCAT.PARTNS	$DATA1.TESTCAT.IXPART01	579	5	2002/09/25 16:30:43	1		VDef	VData
$DATA1.TESTCAT	$DATA1.TESTCAT.PROGRAMS	$DATA1.TESTCAT.IXPROG01	580	4	2002/09/25 16:30:44	1		VDef	VData
$DATA1.TESTCAT	$DATA1.TESTCAT.TABLES	$DATA1.TESTCAT.IXTABL01	581	4	2002/09/25 16:30:43	1		VDef	VData
$DATA1.TESTCAT	$DATA1.TESTCAT.USAGES	$DATA1.TESTCAT.IXUSAG01	583	8	2002/09/25 16:30:44	1		VDef	VData
$DATA1.TESTCAT2	$DATA1.SQLTEST.ORDERS2	$DATA2.SQLTEST.ORDERS2A	0	6	2009/09/09 14:14:52	1		VDef	VData
$DATA1.TESTCAT2	$DATA1.TESTCAT2.INDEXES	$DATA1.TESTCAT2.IXINDE01	577	2	2002/10/15 12:22:00	1	UV	VDef	VData
$DATA1.TESTCAT2	$DATA1.TESTCAT2.PARTNS	$DATA1.TESTCAT2.IXPART01	579	5	2002/10/15 12:22:01	1		VDef	VData
$DATA1.TESTCAT2	$DATA1.TESTCAT2.PROGRAMS	$DATA1.TESTCAT2.IXPROG01	580	4	2002/10/15 12:22:02	1		VDef	VData
$DATA1.TESTCAT2	$DATA1.TESTCAT2.TABLES	$DATA1.TESTCAT2.IXTABL01	581	4	2002/10/15 12:21:59	1		VDef	VData
$DATA1.TESTCAT2	$DATA1.TESTCAT2.USAGES	$DATA1.TESTCAT2.IXUSAG01	583	8	2002/10/15 12:22:01	1		VDef	VData
$DSMSCM.SQL	$DSMSCM.SQL.INDEXES	$DSMSCM.SQL.IXINDE01	577	2	1998/03/21 15:55:37	1	UV	VDef	VData
$DSMSCM.SQL	$DSMSCM.SQL.MAP	$DSMSCM.SQL.MAPI1	900	2	2005/09/27 16:07:28	"GL"	UV	VDef	VData

The SQL/MP / Index screen presents information about files used to provide ordered access to data.

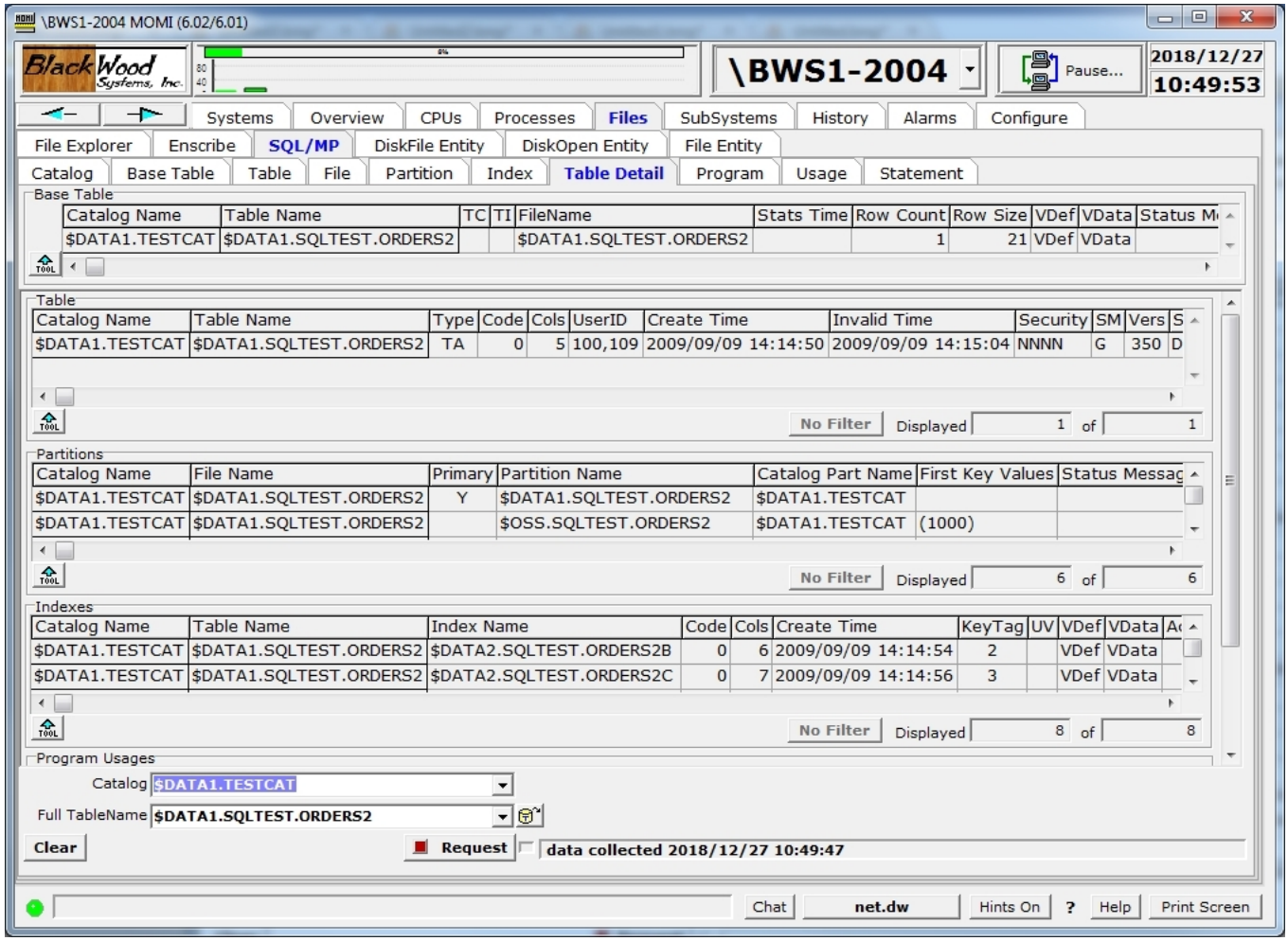
Indexes are comprised of data accessible in an ordered manner. As indexes physically grow in size, the number of levels within the index also increase, which can affect performance. Generally speaking, access to a data record requires the number of index levels plus 1. As such, it is advisable to maintain index levels as low as possible. Note that disk cache generally favors indexes over data so the number of physical drive I/Os is usually reduced by adding cache.

The screen columns ActIndx, IndexL and IndxVar, as shown below, provide the Actual Index levels at the time of screen display, the Index levels recorded in the associated Catalog and the difference. Sorting by and/or comparing these values provides two pieces of information: 1) how many index levels are physically present, and 2) SQL/MP compiles may not be choosing the most optimum access path if the catalog differs greatly from the physical underlying files.



SQL/MP / Table Detail

see System Quirks



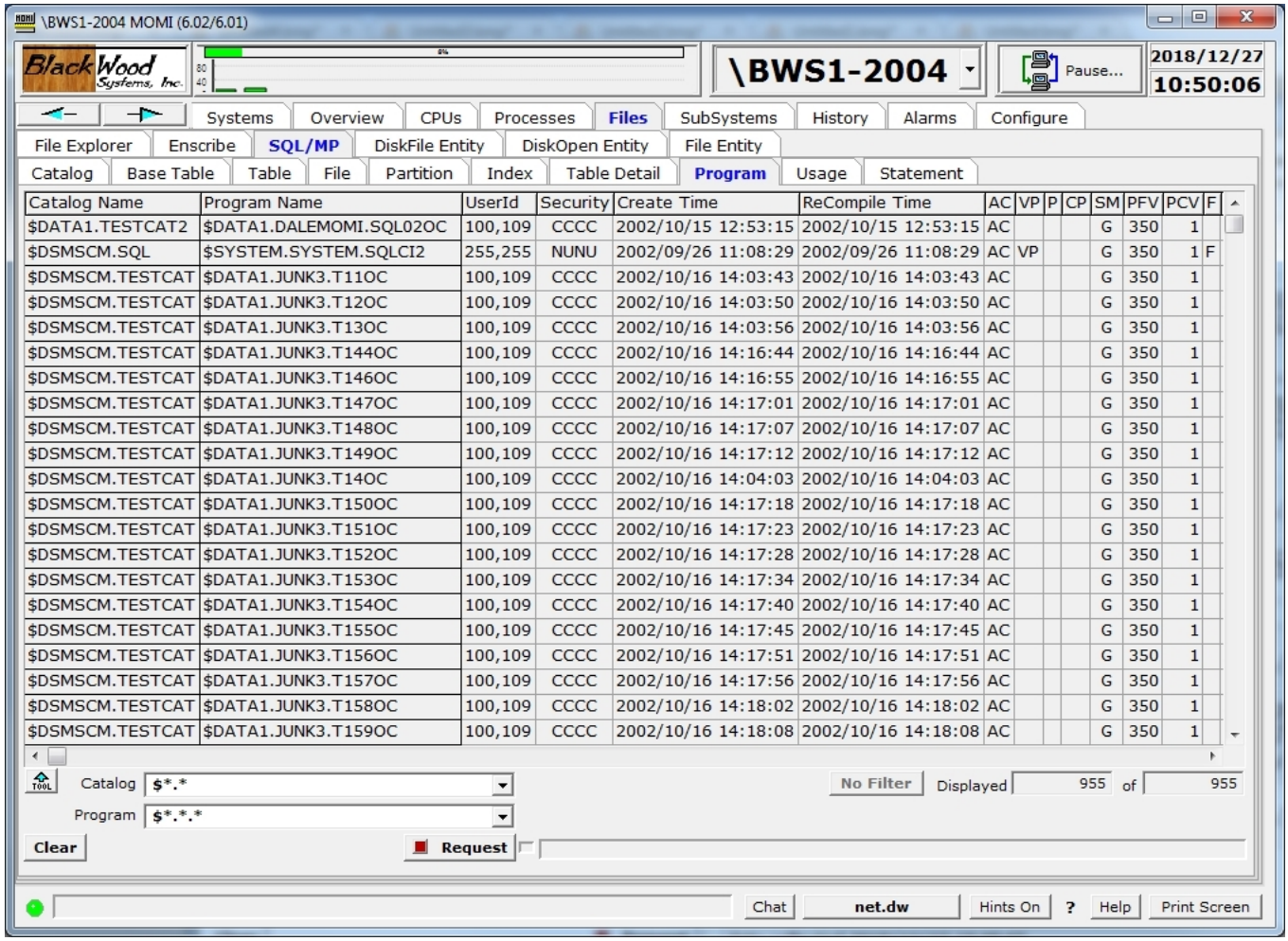
The SQL/MP / Table Detail screen presents the details associated with a Base Table.

All link-able entries to a Base Table are displayed. These include table(s), partition(s), index(es) and program usage(s). A scroll bar is used to display off-screen data.

If a non-Base Table is selected, available information is displayed

SQL/MP / Program

see System Quirks

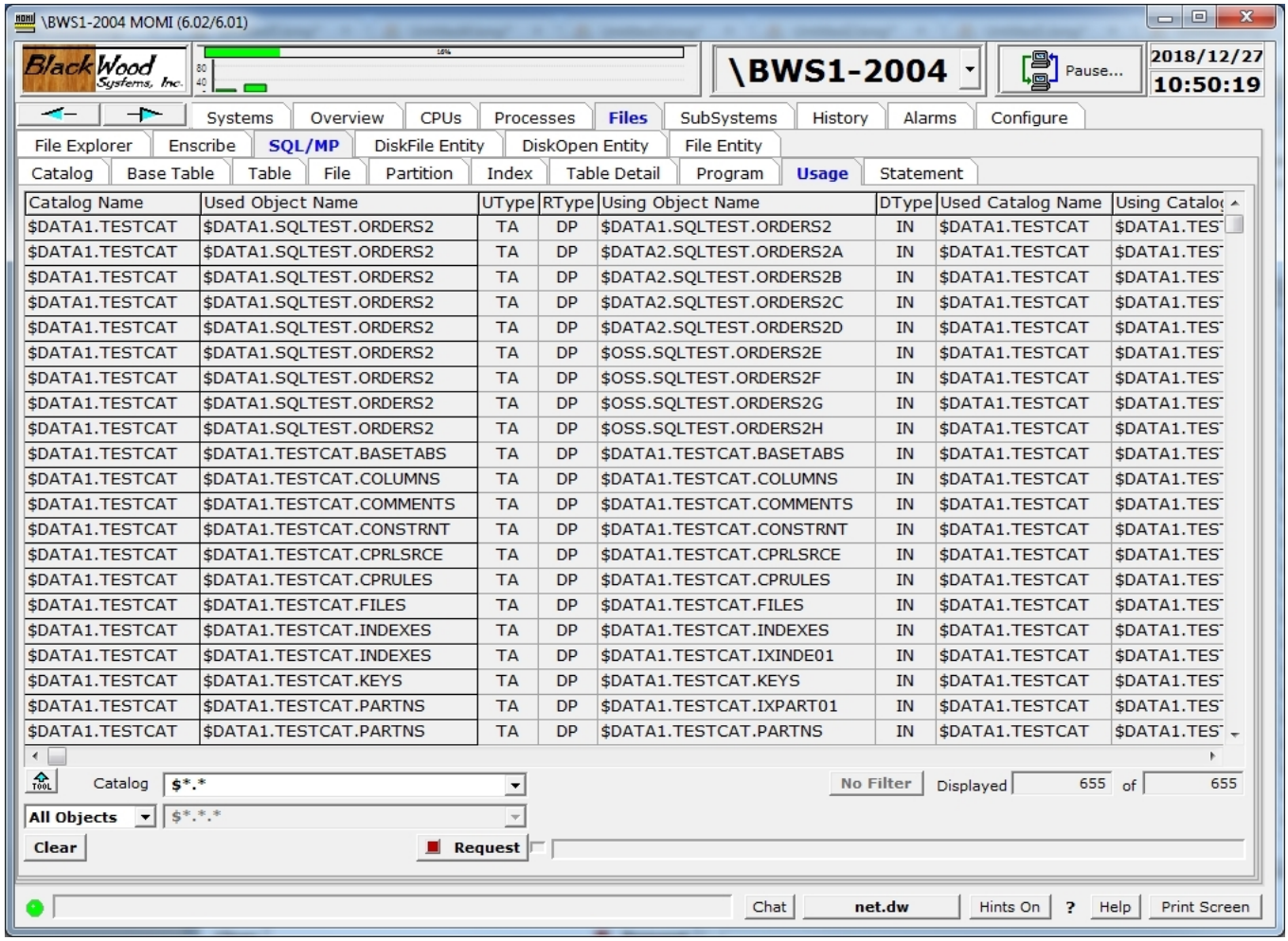


The SQL/MP / Program screen presents programs registered in the SQL/MP catalog.

An SQL-compiled program is by default registered in the catalog. This allows for cross reference and gauging of possible impacts when table changes are made. Not all uses of SQL/MP tables are registered in the catalog.

SQL/MP / Usage

see System Quirks



The SQL/MP / Usage screen presents relationships within the SQL/MP environment.

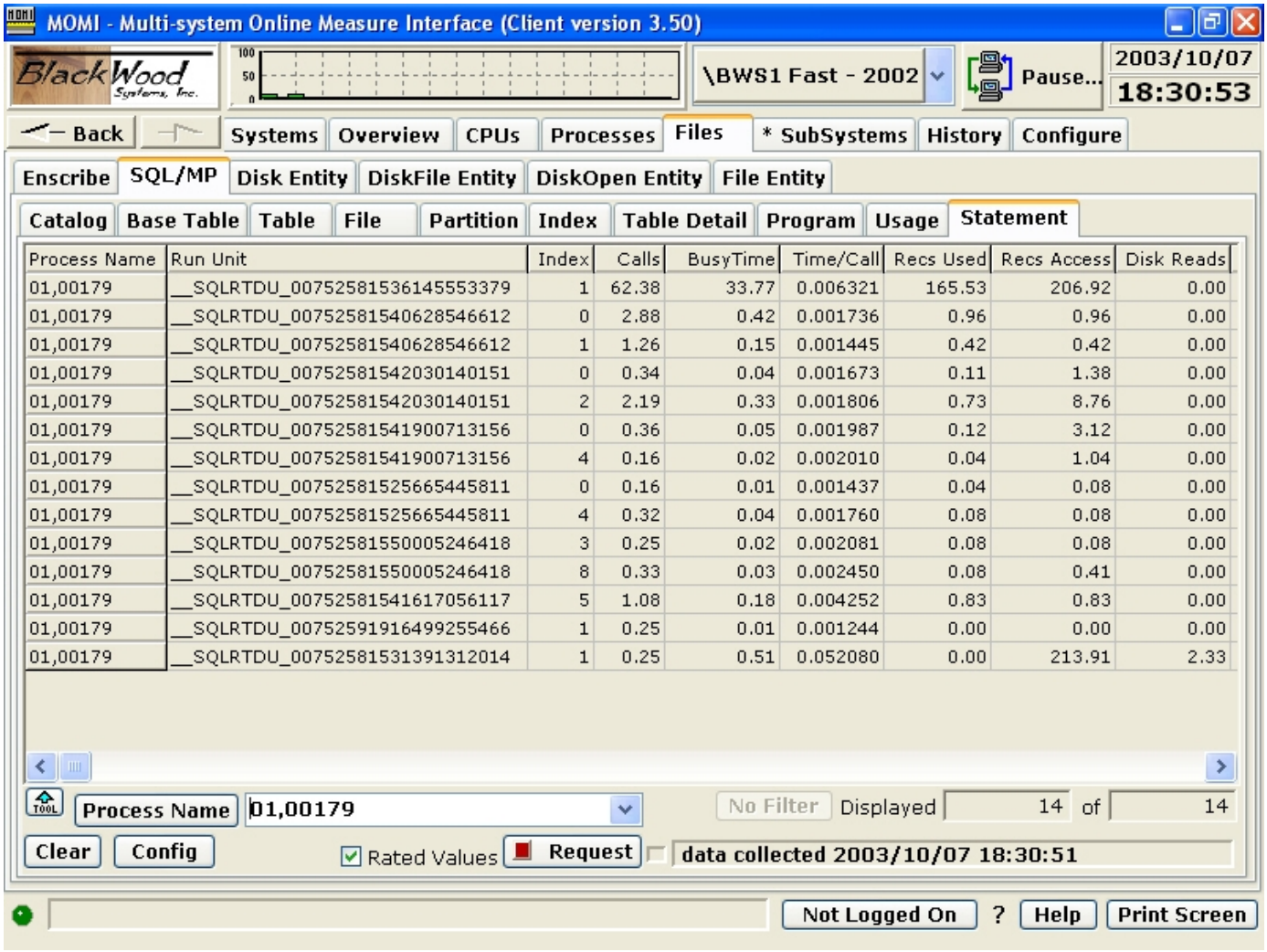
The USAGES table is a catalog table that keeps records of dependencies between objects and between programs and objects.

Relationships recorded in the USAGES table:

- Program uses view, table, or index
- View uses view or table
- Index uses table
- Table, view, index, or program uses collation

SQL/MP / Statement

see System Quirks



Process Name	Run Unit	Index	Calls	BusyTime	Time/Call	Recs Used	Recs Access	Disk Reads
01,00179	__SQLRTDU_00752581536145553379	1	62.38	33.77	0.006321	165.53	206.92	0.00
01,00179	__SQLRTDU_00752581540628546612	0	2.88	0.42	0.001736	0.96	0.96	0.00
01,00179	__SQLRTDU_00752581540628546612	1	1.26	0.15	0.001445	0.42	0.42	0.00
01,00179	__SQLRTDU_00752581542030140151	0	0.34	0.04	0.001673	0.11	1.38	0.00
01,00179	__SQLRTDU_00752581542030140151	2	2.19	0.33	0.001806	0.73	8.76	0.00
01,00179	__SQLRTDU_00752581541900713156	0	0.36	0.05	0.001987	0.12	3.12	0.00
01,00179	__SQLRTDU_00752581541900713156	4	0.16	0.02	0.002010	0.04	1.04	0.00
01,00179	__SQLRTDU_00752581525665445811	0	0.16	0.01	0.001437	0.04	0.08	0.00
01,00179	__SQLRTDU_00752581525665445811	4	0.32	0.04	0.001760	0.08	0.08	0.00
01,00179	__SQLRTDU_00752581550005246418	3	0.25	0.02	0.002081	0.08	0.08	0.00
01,00179	__SQLRTDU_00752581550005246418	8	0.33	0.03	0.002450	0.08	0.41	0.00
01,00179	__SQLRTDU_00752581541617056117	5	1.08	0.18	0.004252	0.83	0.83	0.00
01,00179	__SQLRTDU_00752591916499255466	1	0.25	0.01	0.001244	0.00	0.00	0.00
01,00179	__SQLRTDU_00752581531391312014	1	0.25	0.51	0.052080	0.00	213.91	2.33

The SQL/MP / Statement screen provides a way to drill-down further within a process that uses SQL to see the level of activity performed by individual SQL statements. This information is obtained from MEASURE via the SQLSTMT entity.

The SQL Statement entity describes the cost in terms of time and I/O associated with individual SQL statements in a process. The statements are identified by an Index (0,1,2,3, etc...) obtained from a listing during SQL compilation (specify the SQLMAP option to SQLCOMP).

To use this screen, find a busy SQL application process on your system, perhaps from the Overview screen. Enter the process name (or CPU,PIN) in the Process Name field and push request. After 30 seconds or so, if the process is active and using SQL/MP statements, the results are displayed. The screen continues to update with progressively longer update intervals.

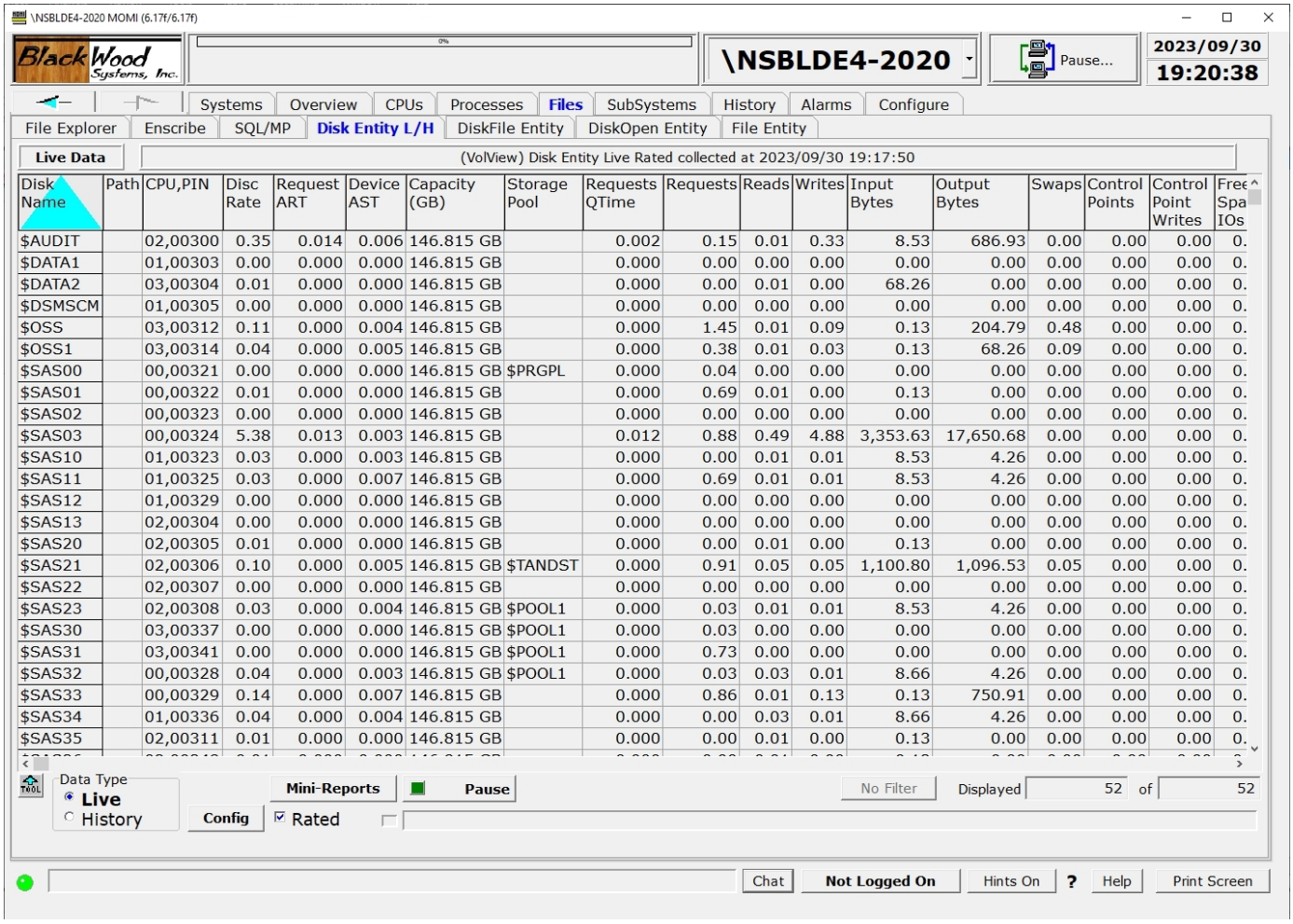
Generally, in order to interpret the output for this screen, you need to work with your DBA and/or application developer as the SQL Statement index is obtained during program compilation. Even with the statement index, usually only the application developer and/or DBA could determine if the

SQL statement is executing as intended (knowledge of the data layout, meaning of the data and accesses available into the data really help).

If you are dealing with SQL/MX, things get a little strange as SQL/MX has the requirement that an SQL Statement measurement must be in place prior to starting the program. This is really backward (and does not make much sense to us) but in theory allows reduced measurement overhead. SQL/MX really did things differently from a historical NonStop perspective. Unfortunately, it may be better to a) run MEASURE manually to collect the data OR b) manually start an external measurement and then go into this screen within MOMI. In the latter case, the measurement that MOMI launches is internally "latching into" an existing measurement which allows data to be obtained.

This screen is disabled by default. See the CONFMOMI keyword DISABLE-MEAS-SQLSTMT for additional information.

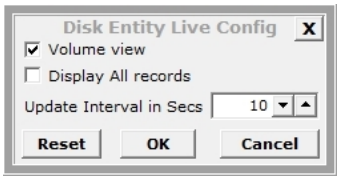
Disk Entity L/H



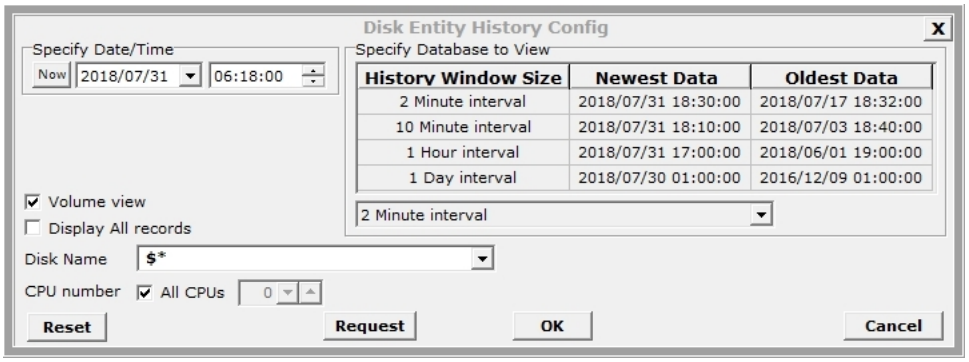
The Files / Disk Entity LH screen presents the MEASURE entity DISC. Data is available live or via history.

The MEASURE entity DISC provides information about the overall disk volume. A mirrored drive would have four paths, Primary, Backup, Mirror and Mirror Backup. By default, MOMI combines these paths into a single line of information referred to as Volume View.

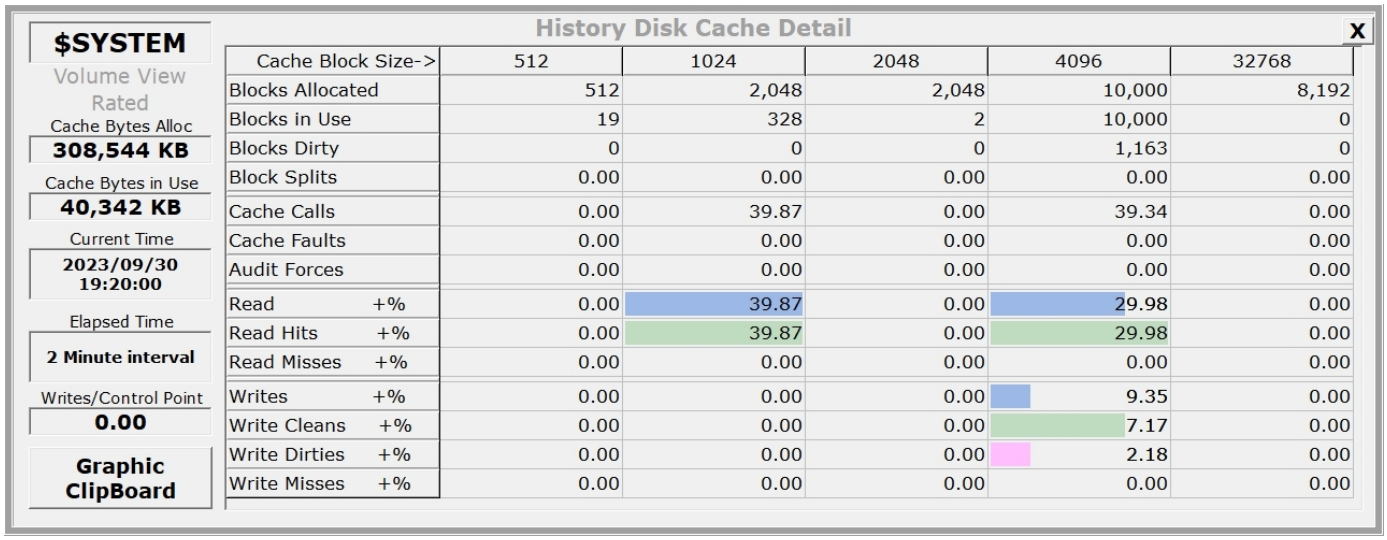
The Data Type field selects either Live (the default) or History. The Config button displays available options which are different depending on the Data Type selection.



Config for Live allows the selection of Volume view, if records with all zero values are displayed, and the screen update interval.



Config for History determines the starting date/time, selection of Volume view, history file where data is retrieved, if records with all zero values are displayed and filtering by Disk and/or CPU.



Clicking on a Disk Name displays Cache Detail showing all cache sizes and cache specific counts. The color bars, on lines with +%, represent percent of the total for that particular operation. Place the mouse over the cell and the tool-tip displays the underlying value.

Notes

Volume view

Volume view combines values from all four disk paths (P, B, M, MB) in an attempt to represent the overall activity on the logical disk drive. When Volume view is NOT selected, MEASURE data is displayed as is, with the exception of derived values.

Output bytes represents the bytes written to one disk volume of the mirrored set. For example, Input Bytes and Output Bytes would equal each other if 4096 bytes were written and immediately read (technically, 8192 bytes were written as the data is copied to each physical disk of the mirrored set).

Derived values

The following values are derived:

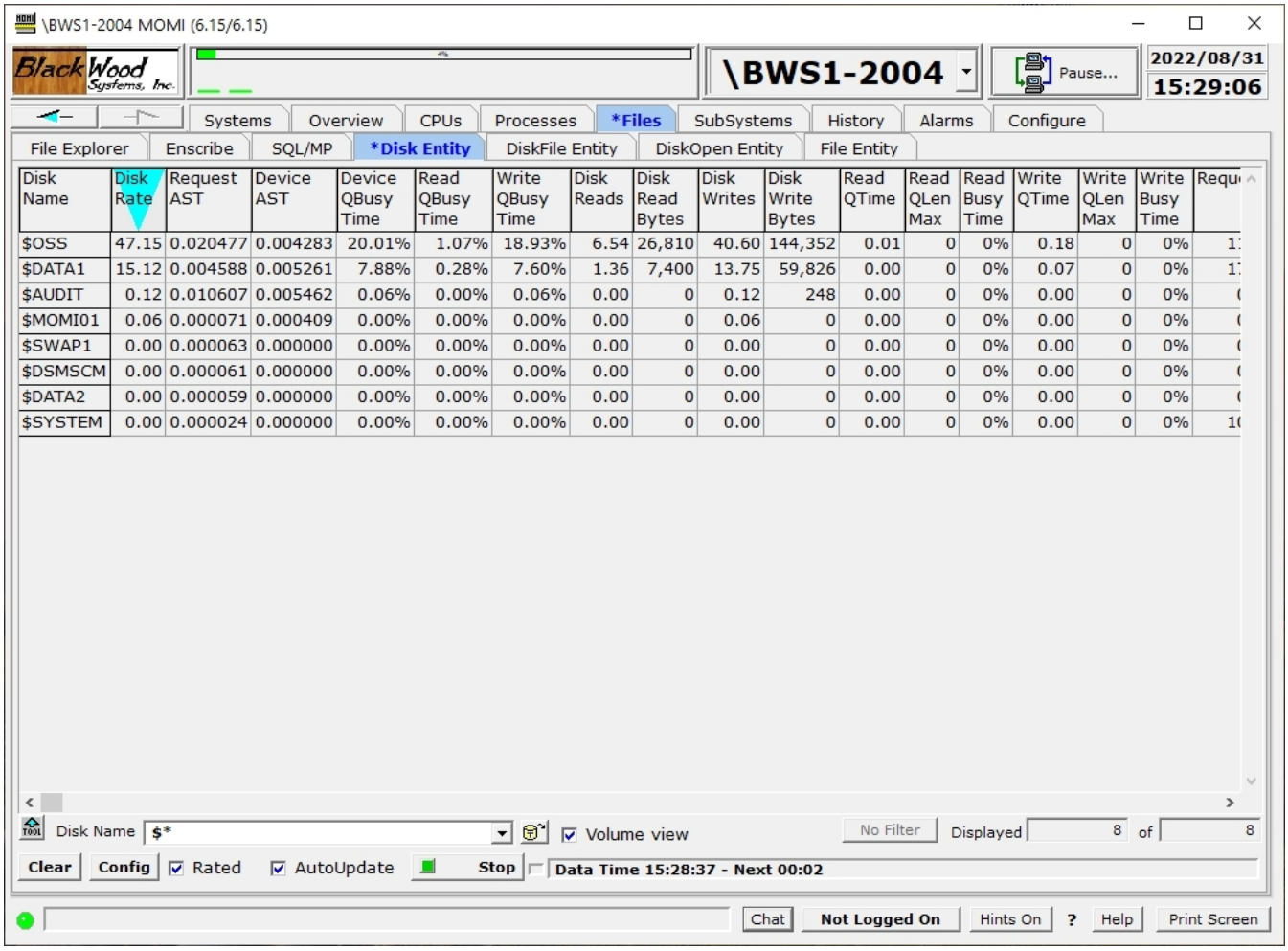
```
disc_rate

      reads + writes
```

Cache Detail

MEASURE does not directly provide all the values displayed and the following values are derived (and may be changed as we learn more about how to obtain them):

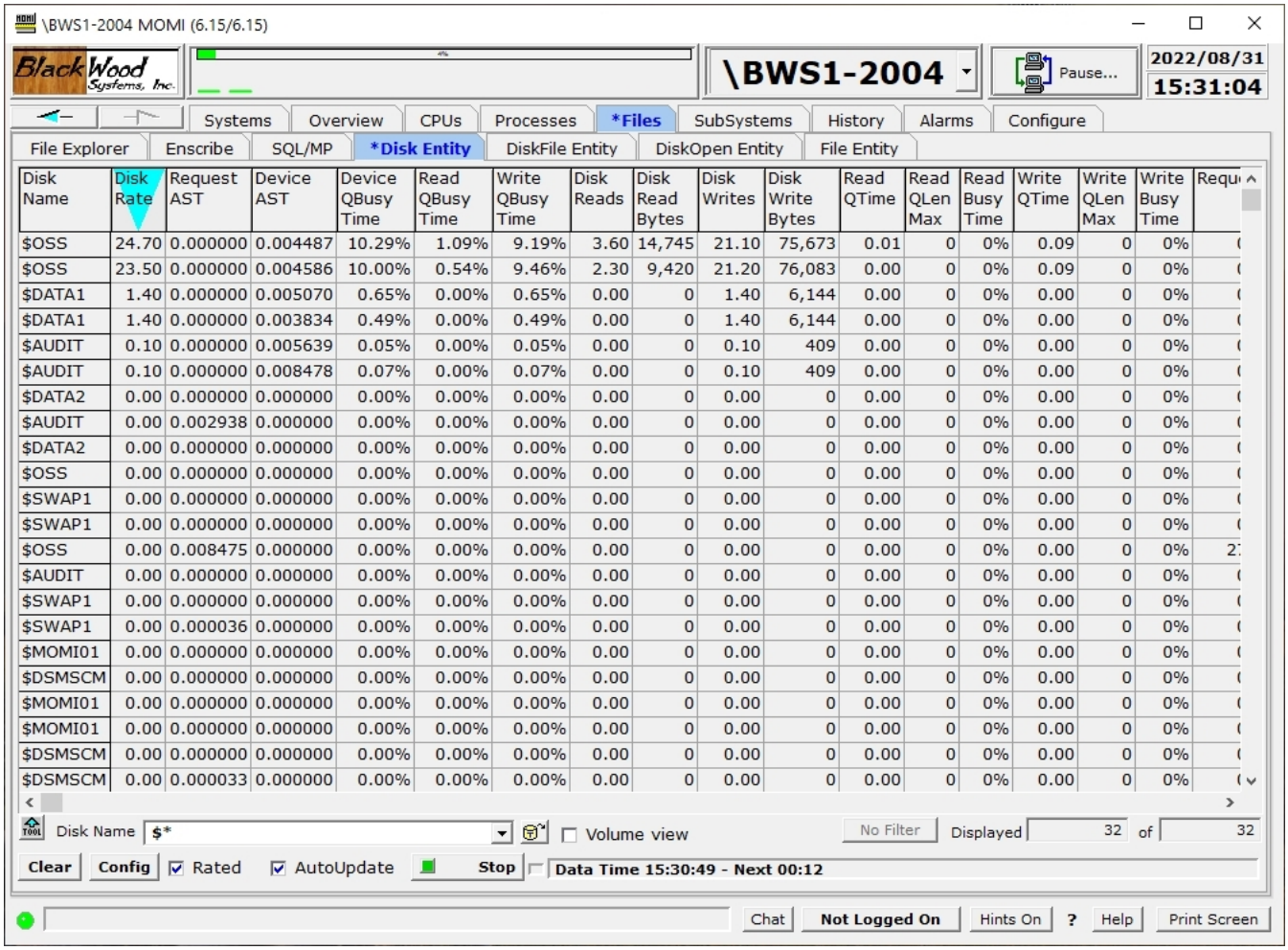
```
cache_calls

       hits
     + misses


cache_writes

      cache_write_cleans
    + cache_write_dirties
    + cache_write_misses


cache_reads

      cache_calls
    - cache_writes


cache_read_hits

      cache_reads
    - misses
```

Files / Disk Entity



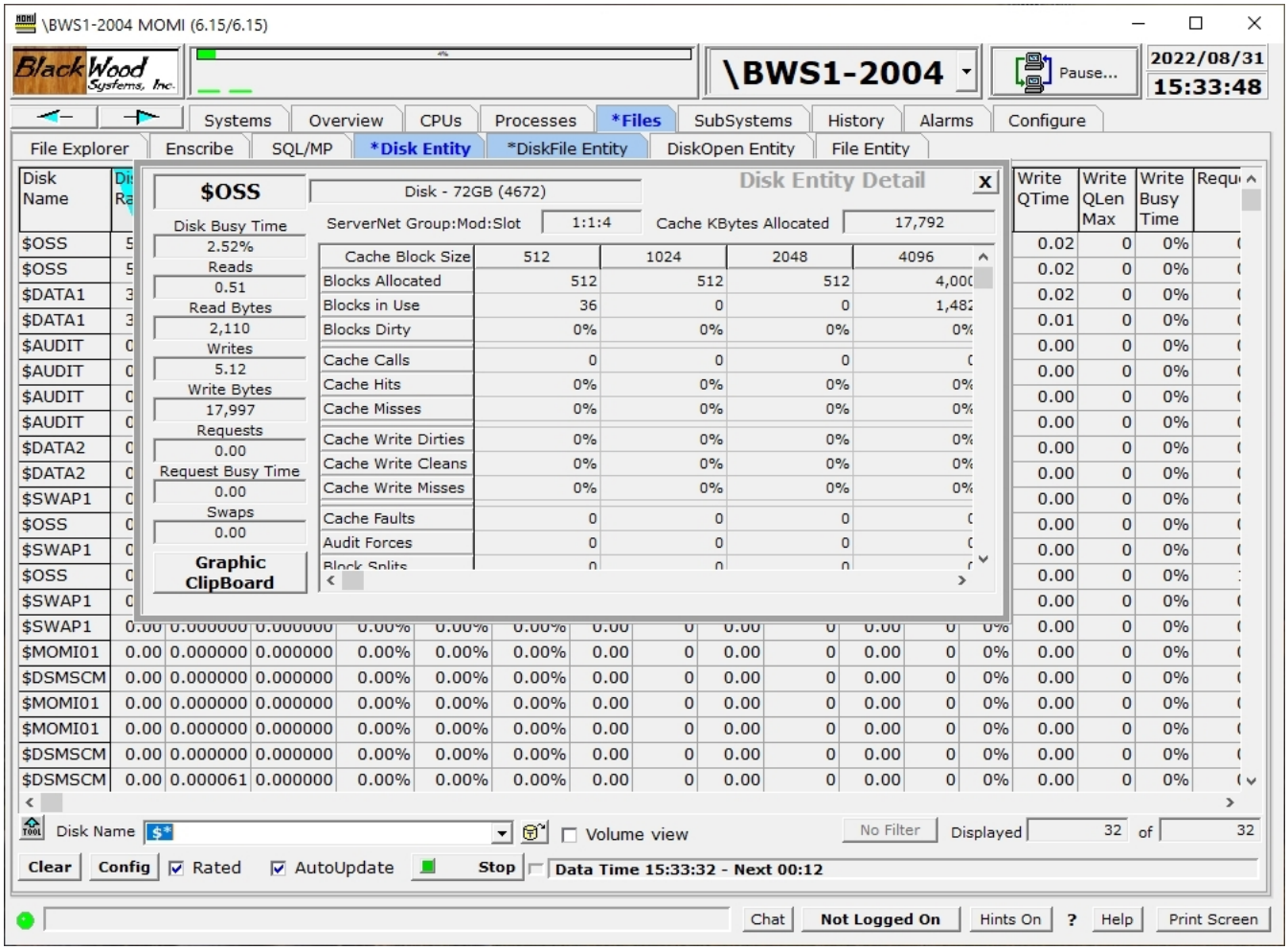
The Files / Disk Entity screen presents the MEASURE Disk Entity which reports statistics at a physical disk volume level.

The Disk Entity provides information on all four possible disk paths (primary, primary backup, mirror and mirror backup) for both the primary and mirror drive. By default, MOMI consolidates this information into a single line of performance information. The combining of this information can result in percentages of usually up to 200% when mirrored drives are present. For example, if the Write-QBusy-Time is 145%, this effectively means that 72.5% of the time of each disk (145 / 2) is being consumed. Uncheck the **Volume view**selection to display information in the original or native format as seen below:



In the non-Volume view, the ServerNet Location (Group:Module:Slot) on newer platforms can be used to distinguish between primary and mirror drives.

Detailed disk cache information for a particular entry may be presented in a consolidated window by right-clicking on a disk name and selecting **View Disk Entity Detail** as seen below. MEASURE does not provide the same level of detail concerning cache as SCF (hence a few screen differences) and not all disk paths return cache information.

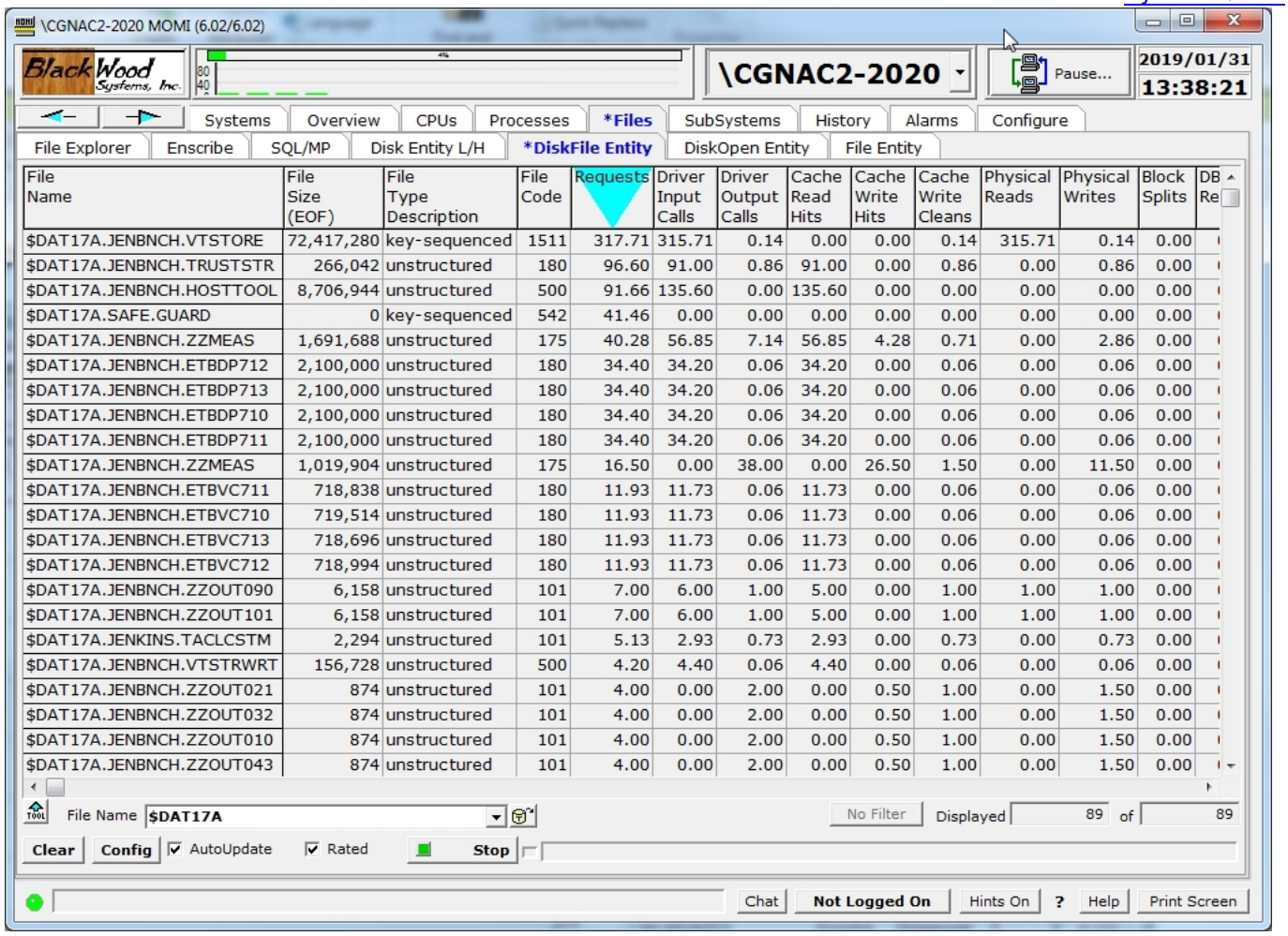


This is an approach to consider when diagnosing disk drive performance issues:

- Drill down starting at the volume level, working down sequentially through the DiskFile, DiskOpen and finally to the process level.
- Starting from the Disk Entity screen, click on a volume and select DiskFile Entity to see all of the active files on the disk.
- On an entry on the DiskFile Entity screen, click to select DiskOpen Entity which displays all of the processes accessing the file.
- On an entry on the DiskOpen Entity screen, click on the **Opener Process Name** to view Process Detail of the opener process of that file (you can also click on the File Name to view the File Entity which provides different I/O counters).

Files / DiskFile Entity

see System Quirks



The Files / DiskFile Entity screen presents information about the MEASURE DiskFile entity.

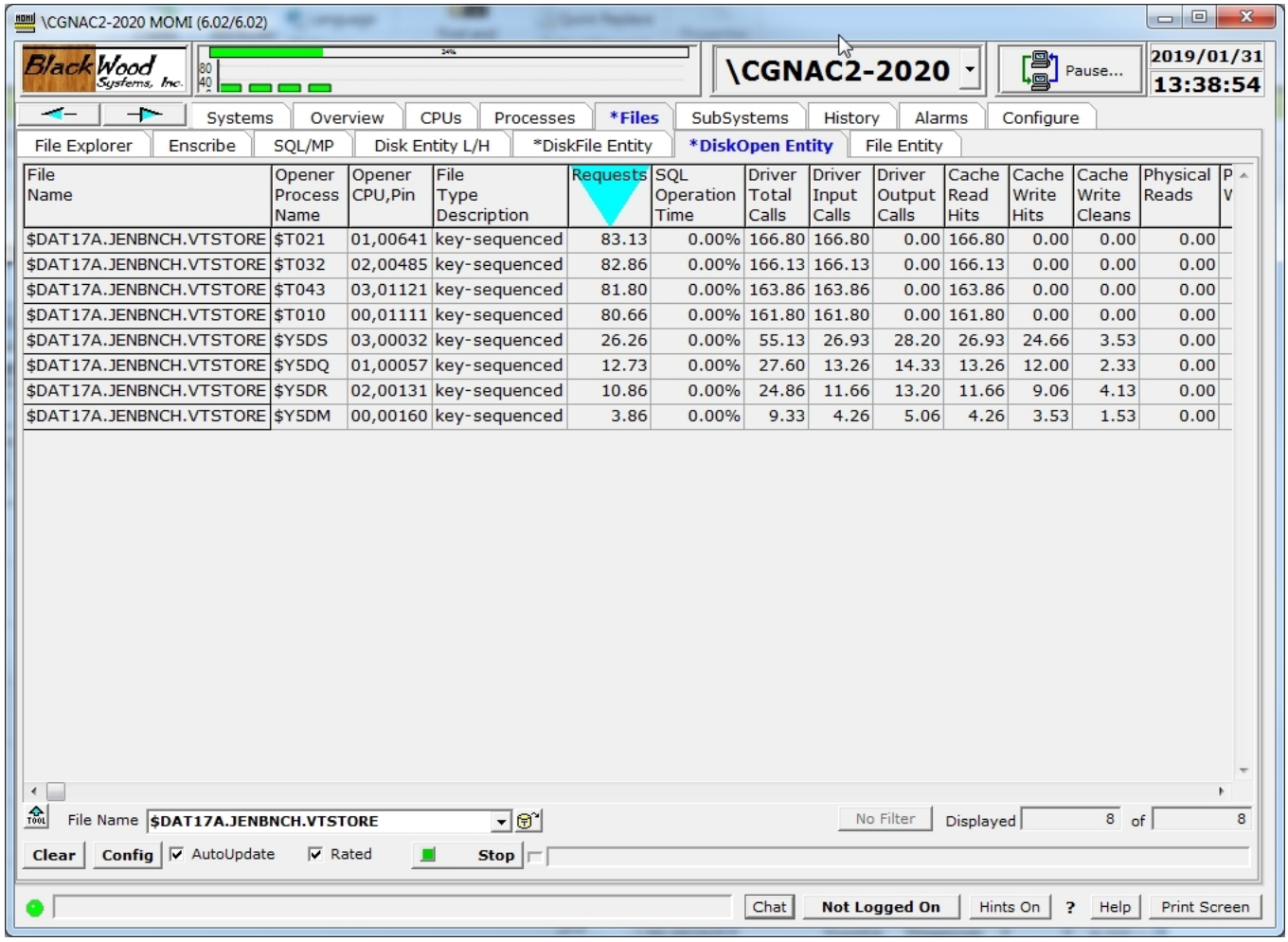
The MEASURE DiskFile entity displays one record for each open (or active) file on a disk volume. This entitycan be considered the next level down from the Disk Entity.

This is an approach to consider when diagnosing disk drive performance issues:

- Drill down starting at the volume level, working down sequentially through the DiskFile, DiskOpen and finally to the process level.
- Starting from the Disk Entity screen, click on a volume and select DiskFile Entity to see all of the active files on the disk.
- On an entry on the DiskFile Entity screen, click to select DiskOpen Entity which displays all of the processes accessing the file.

- On an entry on the DiskOpen Entity screen, click on the **Opener Process Name** to view Process Detail of the opener process of that file (you can also click on the File Name to view the File Entity which provides different I/O counters).

Files / DiskOpen Entity



The Files / DiskOpen Entity screen presents the MEASURE DiskOpen entity.

The MEASURE DiskOpen entity is I/O activity on a file from an opener process perspective. One line is present for every open to a file.

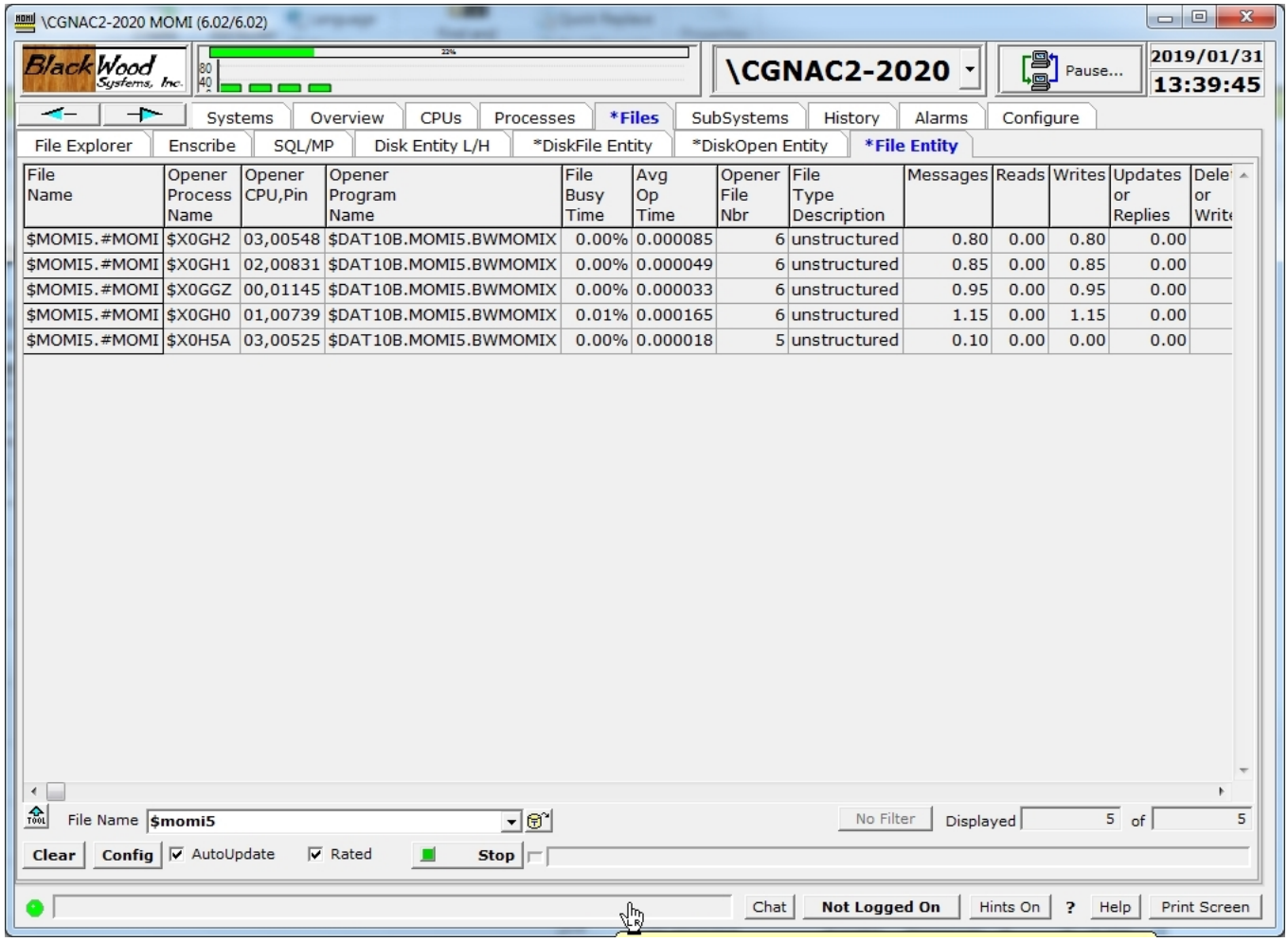
The MEASURE DiskOpen entity does not store the Opener Process Name, only the Opener CPU and Opener PIN. The Opener Process Name column is derived by looking up the current process name at the CPU & PIN currently running at that location.

This is an approach to consider when diagnosing disk drive performance issues:

- Drill down starting at the volume level, working down sequentially through the DiskFile, DiskOpen and finally to the process level.
- Starting from the Disk Entity screen, click on a volume and select DiskFile Entity to see all of the active files on the disk.

- On an entry on the DiskFile Entity screen, click to select DiskOpen Entity which displays all of the processes accessing the file.
- On an entry on the DiskOpen Entity screen, click on the **Opener Process Name** to view Process Detail of the opener process of that file (you can also click on the File Name to view the File Entity which provides different I/O counters).

Files / File Entity



The Files / File Entity screen presents the MEASURE File Entity.

The MEASURE File entity is I/O activity on a file. A file is a communication path from one process to another and is **not** necessarily to a disk file. This entity is unique with its ability to track and report on all sorts of different I/O activity from inter-process, TCP/IP, OSS and process to disk.

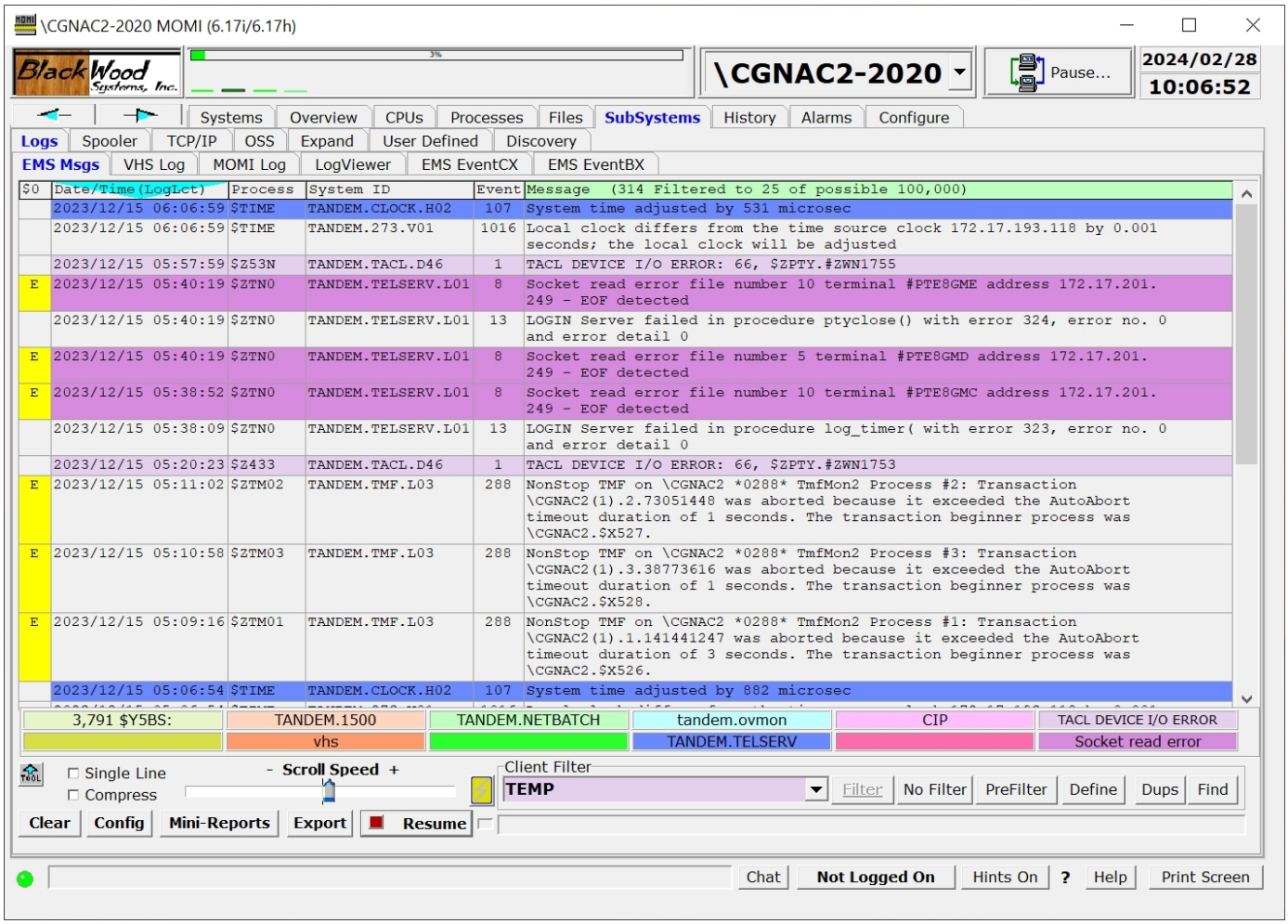
The File entity is measured from the perspective of the opener. A process on a remote system opening a local file cannot be measured on the local system using this entity.

A local process opening a remote file can be measured locally. For example, if you are on system \BWS1 and want to see the file level activity across Expand, you would enter in the File Name field \BWS2.$*.*.* .

SubSystems

Logs

EMS Msgs

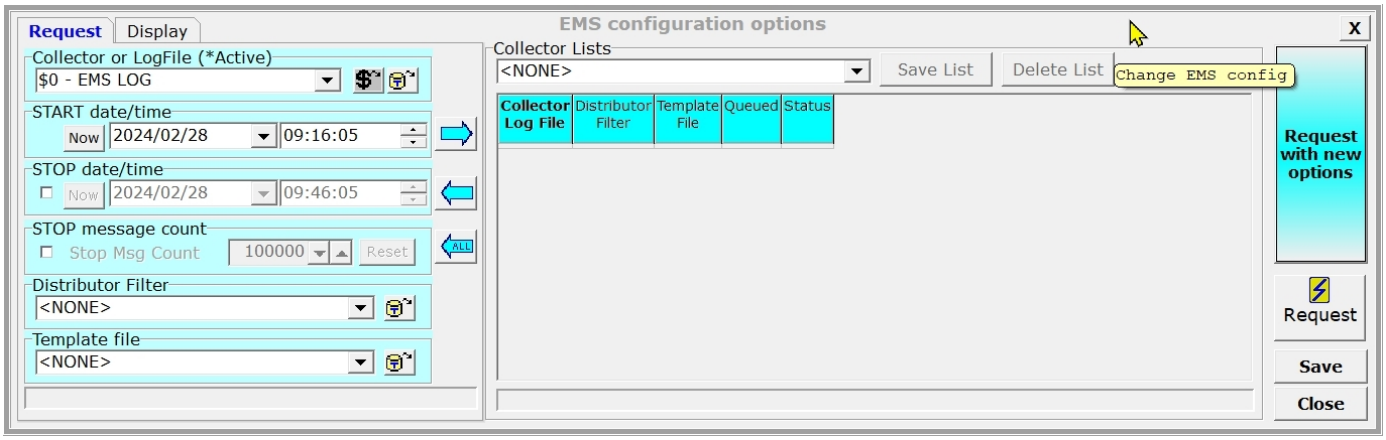


The EMS Msgs screen presents EMS Events logged to one or more EMS collectors or EMS log files. Messages may be filtered (i.e. removed from the screen) on-the-fly and color coded based on content.

EMS messages are used by the system and user programs to communicate and record significant events.

Initially, the screen is blank until the **Request** button is pressed which will by default display EMS messages from $0, the primary system collector.

To view messages from an alternate EMS collector or to change other display options, press the **Config** button. In the pop-up window enter or select the various fields. Press the right pointing arrow to add to the list then press **Request with new options** to begin retrieving data.



Collector or LogFile determines which EMS collector or file to obtain information. The two buttons to the right display a list of EMS collector processes or a file picker keyed to EMS log files (i.e. file code 843).

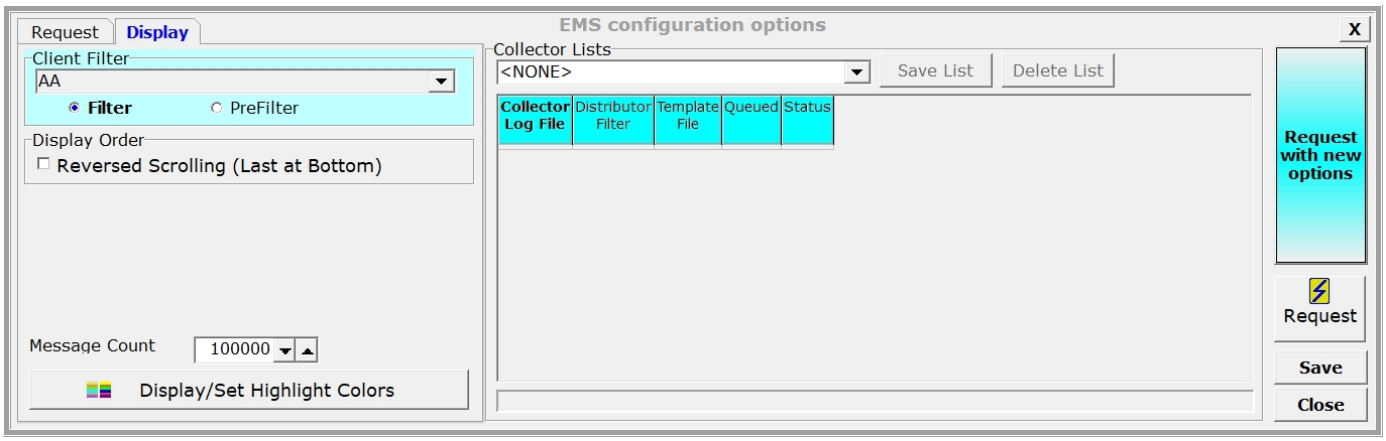
START date/time is the time location to begin retrieving records.

STOP date/time is the time location to stop retrieving records. By default records are read until End of File is reached.

STOP message count determines how many EMS messages are retrieved. The actual number of message returned may exceed this value somewhat due to internal optimizations.

Distributor Filter allows the use of a precompiled EMS filter. A pre-compiled filter is a very efficient means to limit information reported, based on the filter criteria, as this functions as a pre-filter at the distributor level and only provides to MOMI records with matching criteria. This filter is may be used in addition to on-the-fly or client based filtering described later.

Template file allows the definition of templates that are used when converting the EMS messages to readable text. This is equivalent to specifying the TACL DEFINE =_EMS_TEMPLATES.

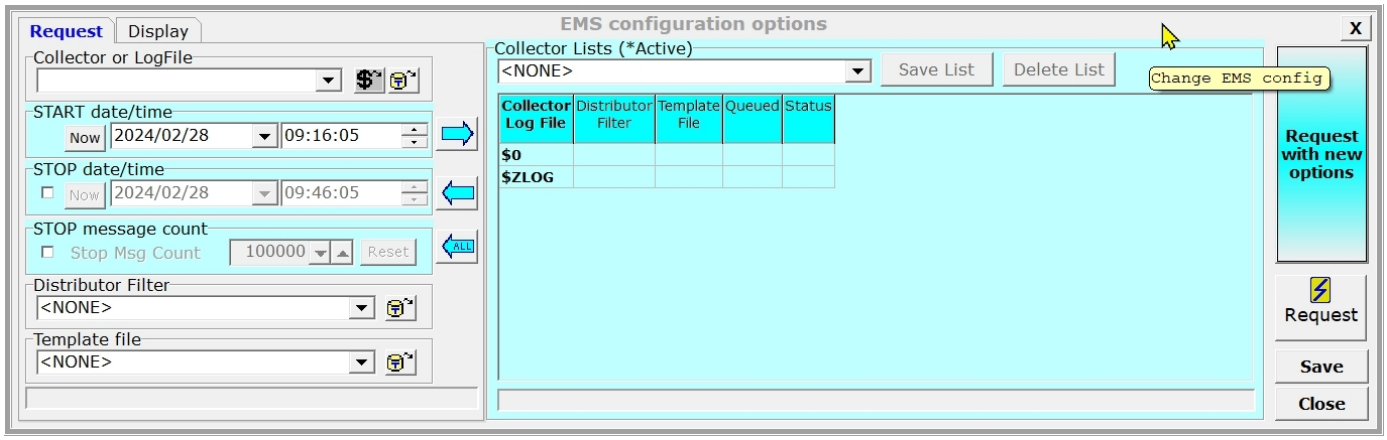


Client Filter allows the selection of a client based filter previously saved. The filter applies to all data displayed on this screen. By default, data matching the **Filter** is hidden from display. The **PreFilter** selection causes data matching the filter to be discarded.

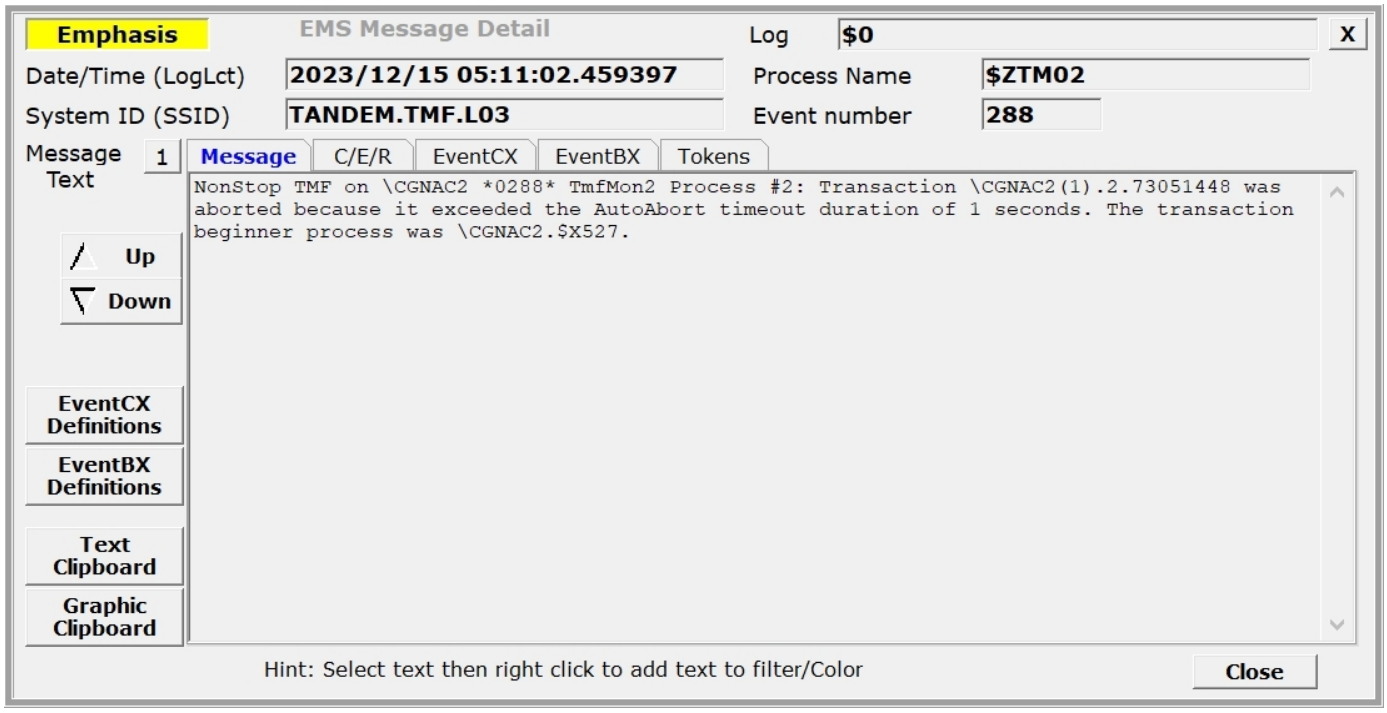
Display Order option of **Reversed Scrolling (Last to Bottom)** causes new messages to initially appear on the bottom of the grid and scroll up. By default, new messages appear at the top of the grid and scroll down.

Message Count determine the maximum number of messages available in the scrollable window area. EMS collectors or EMS log files only provide records from a start time moving forward. No 'backward' ability is available. MOMI allows the user to determine how many EMS messages are loaded with the screen window and to scroll through these messages. When the number of message exceeds the Msg Count, the oldest message is discarded. When filtering messages, described later in this section, the action of **Filter** and **PreFilter** determine if messages are actually present in the screen buffer but hidden (**Filter**) or if messages are actually discarded when the filtering criteria is met (**PreFilter**). Turning a filter off (**No Filter**) causes hidden messages to appear.

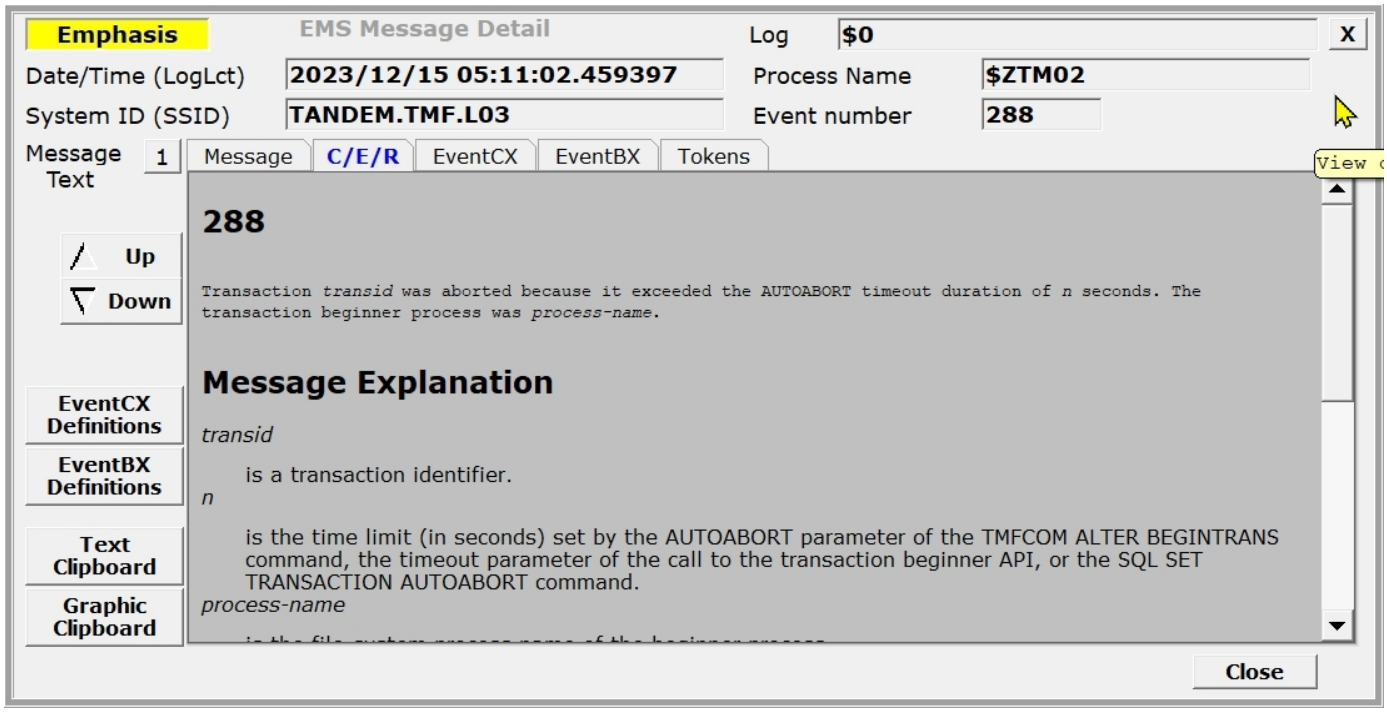
Multiple EMS collectors or log files may be selected. After entering or selecting an EMS collector or log file, press the right array button. The collector and any precompiled EMS filter, now appear in Collector Lists area. Add as many collectors and/or log files as desired. Press **Request with new options** to display EMS records from the entered collectors and/or log files. A single entry may be removed by clicking on it then pressing the left arrow. The left arrow labeled ALL clears the **Collector Lists** area.



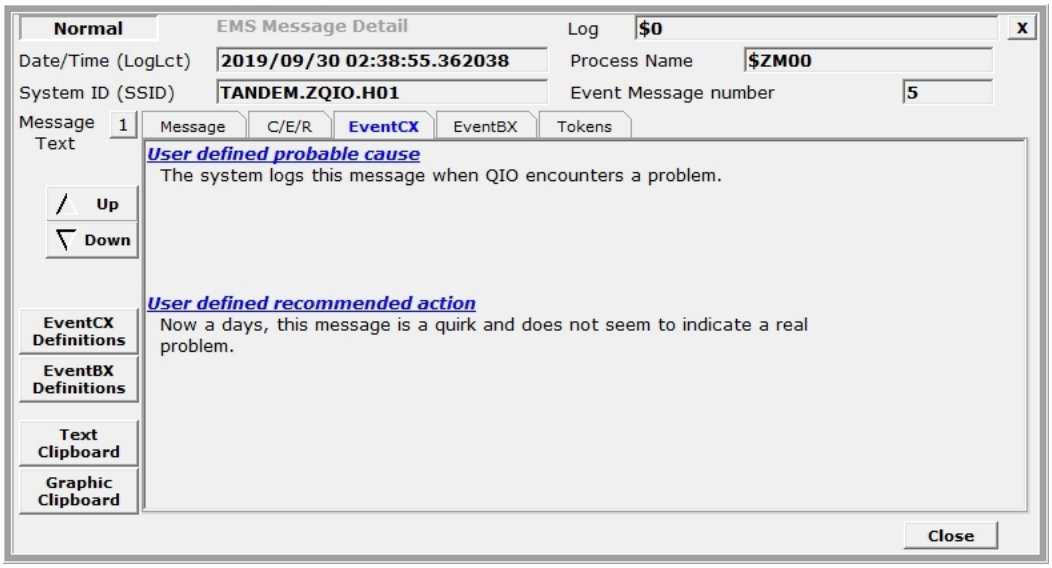
Click on an EMS message to display a pop-up window for additional information. The **Message** tab contains the entire EMS message.



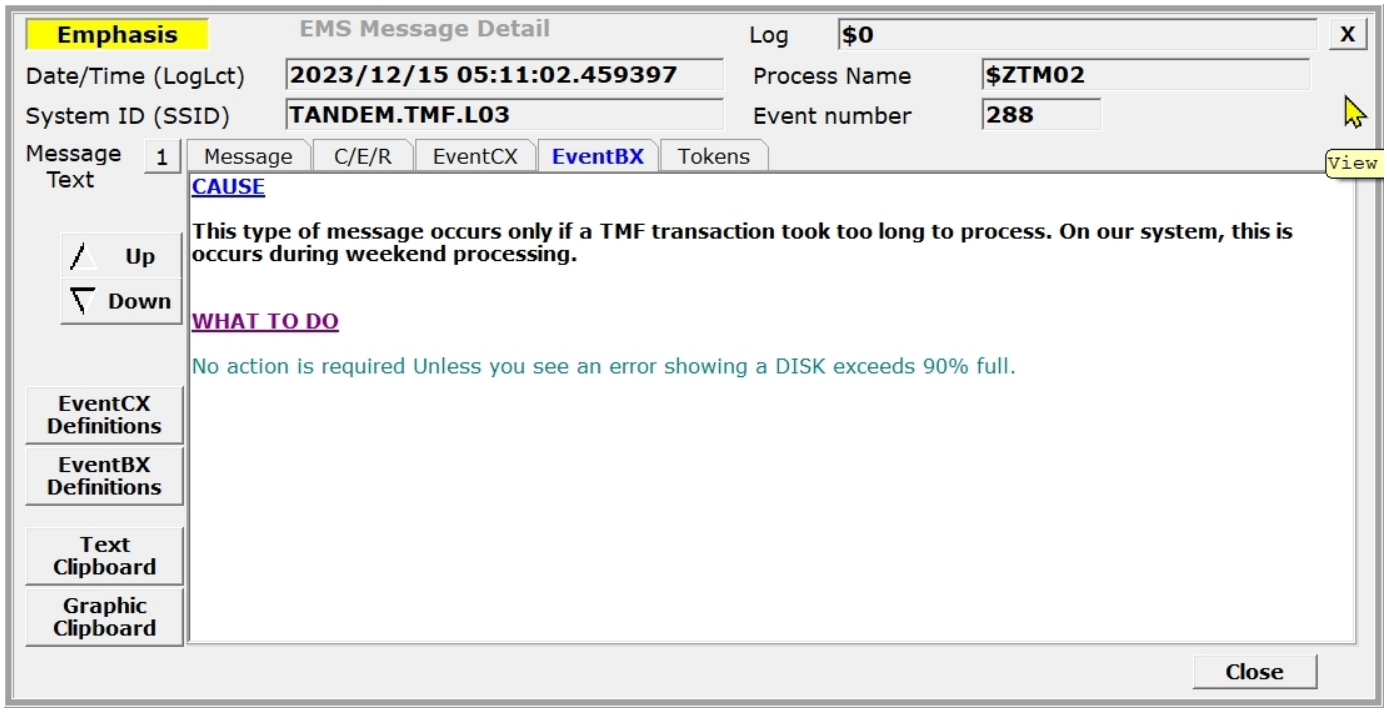
The **C/E/R** (**C**ause / **E**ffect / **R**ecovery) retrieves information from the EVENTTX file which provides detail about many system messages. Usually, it explains what caused the message, possible effects to the system, and makes recovery recommendations (if available). Each release or update to the NonStop O/S may alter these messages.



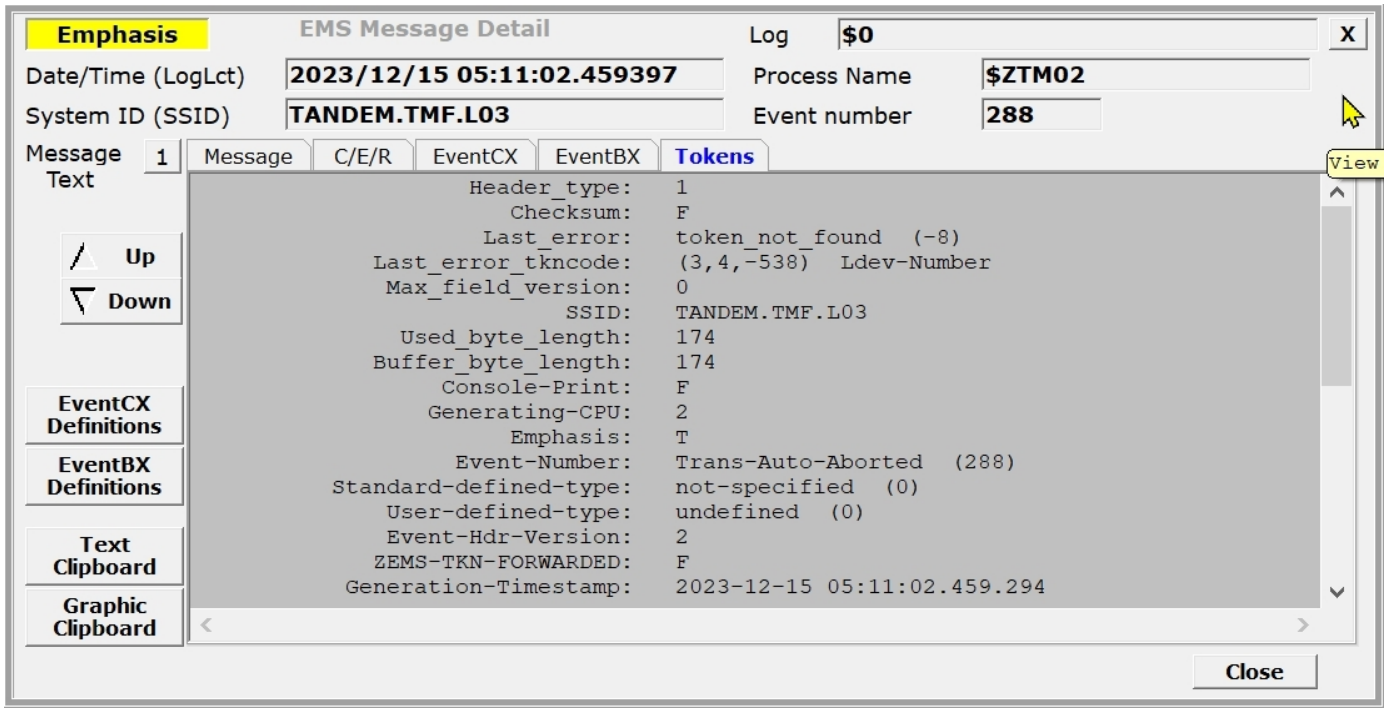
The **EventCX** tab displays any user defined information associated with the EMS message from this system defined file. Note that this file may not be present and is usually **EMPTY**. See EVENTCX for additional information.



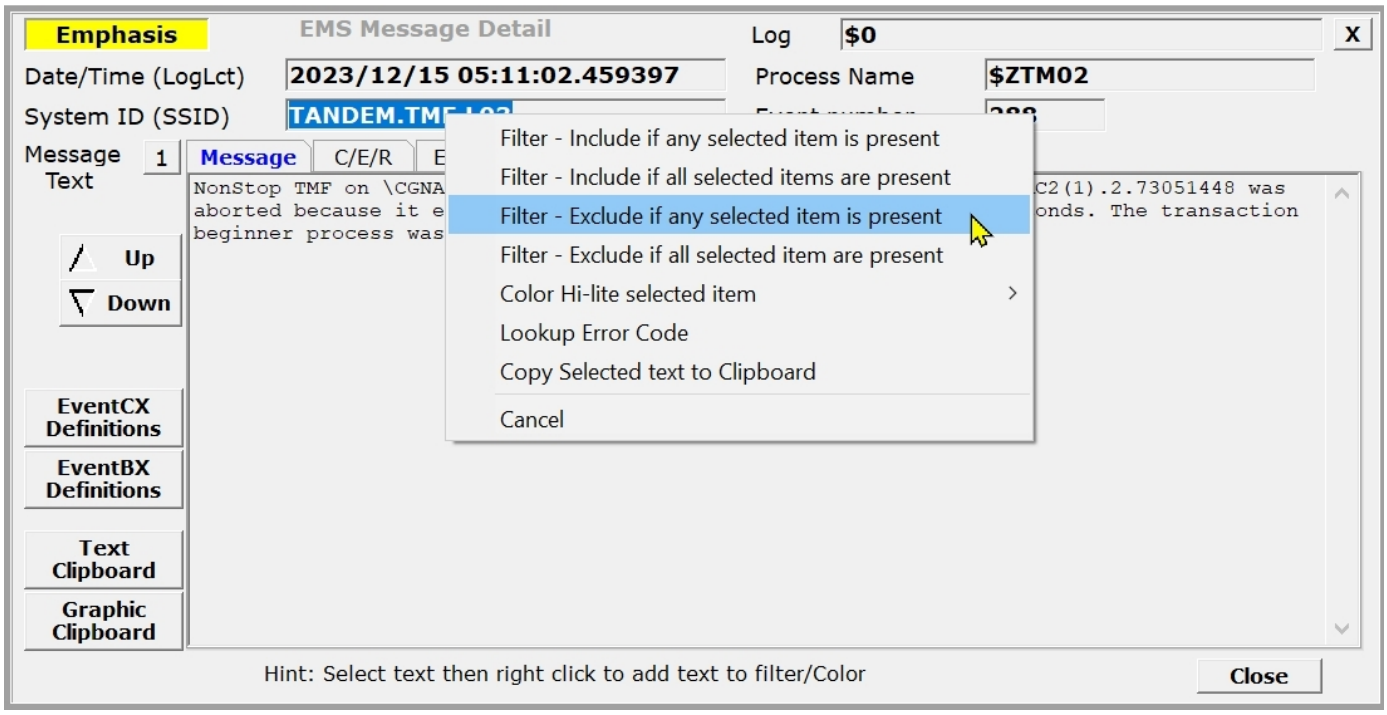
The **EventBX** tab contains any user defined information associated with the EMS message using information defined with MOMI. See EventBX / Define for additional information.



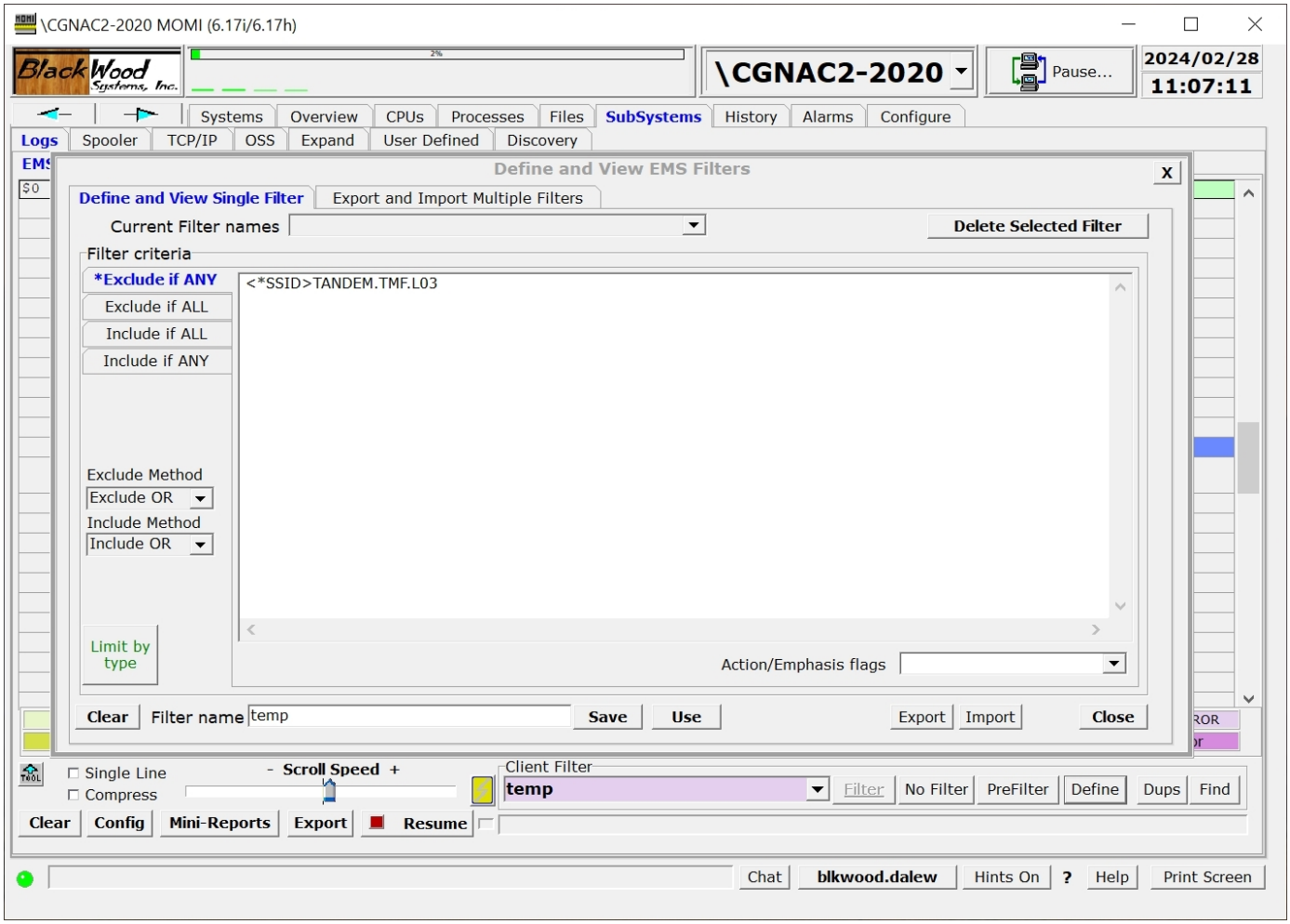
The **Tokens** tab contains all of the EMS tokens comprising the EMS message. Besides providing the raw event, sometimes additional information may be available as formatting templates do not necessarily translate all of the tokens into ASCII text.



Displayed messages may be easily filtered *on-the-fly* to remove unneeded messages and/or to only display particular messages. This is accomplished by clicking on a message to display it in detail (you might need to **Pause** the screen), highlight the text (left-click and hold then drag mouse over the desired text), and then right-click over the selected text. A menu selection provides filtering to include or exclude messages containing the text. *Note that highlighting text then placing the mouse over it does not change the mouse pointer to a hand cursor.*



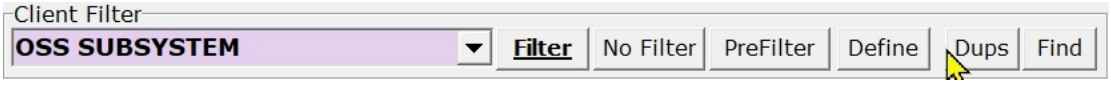
Filters constructed *on-the-fly* may be saved for later reuse. Press the **Define** button to display all of the current filter criteria. Saved filters may be viewed, edited, selected or deleted. Fields searched within the EMS message may be limited by the **Limit by type** button. In the sample image TANDEM.CLOCK is only searched in the SSID portion of the EMS message. If no limit is specified, the search is through the entire EMS message.



Saved filters may be selected from a drop-down box in the **Client Filter** area (or on the **Config** pop-up window). Once a filter is selected, the **Filter**, **No Filter** and **PreFilter** buttons are available.

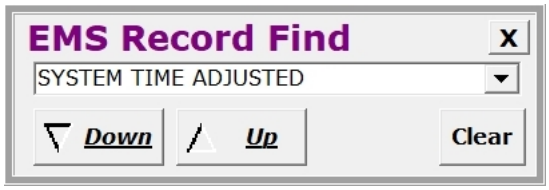
The **Filter** button causes an EMS message matching the current filter to be hidden. EMS messages that are hidden are retained in memory and may be made visible by pressing the **No Filter** button. This allows filters to be applied and cleared without the need to reload from the NonStop Systems. EMS messages, both visible and hidden, are stored in memory until **Message Count** is reached (when a limit is reached the oldest EMS message is discarded to make room for the newest).

The **PreFilter** button causes an EMS message that matches the current filter to be discarded. This means it does not consume any storage and pressing the **No Filter** button cannot make it visible.

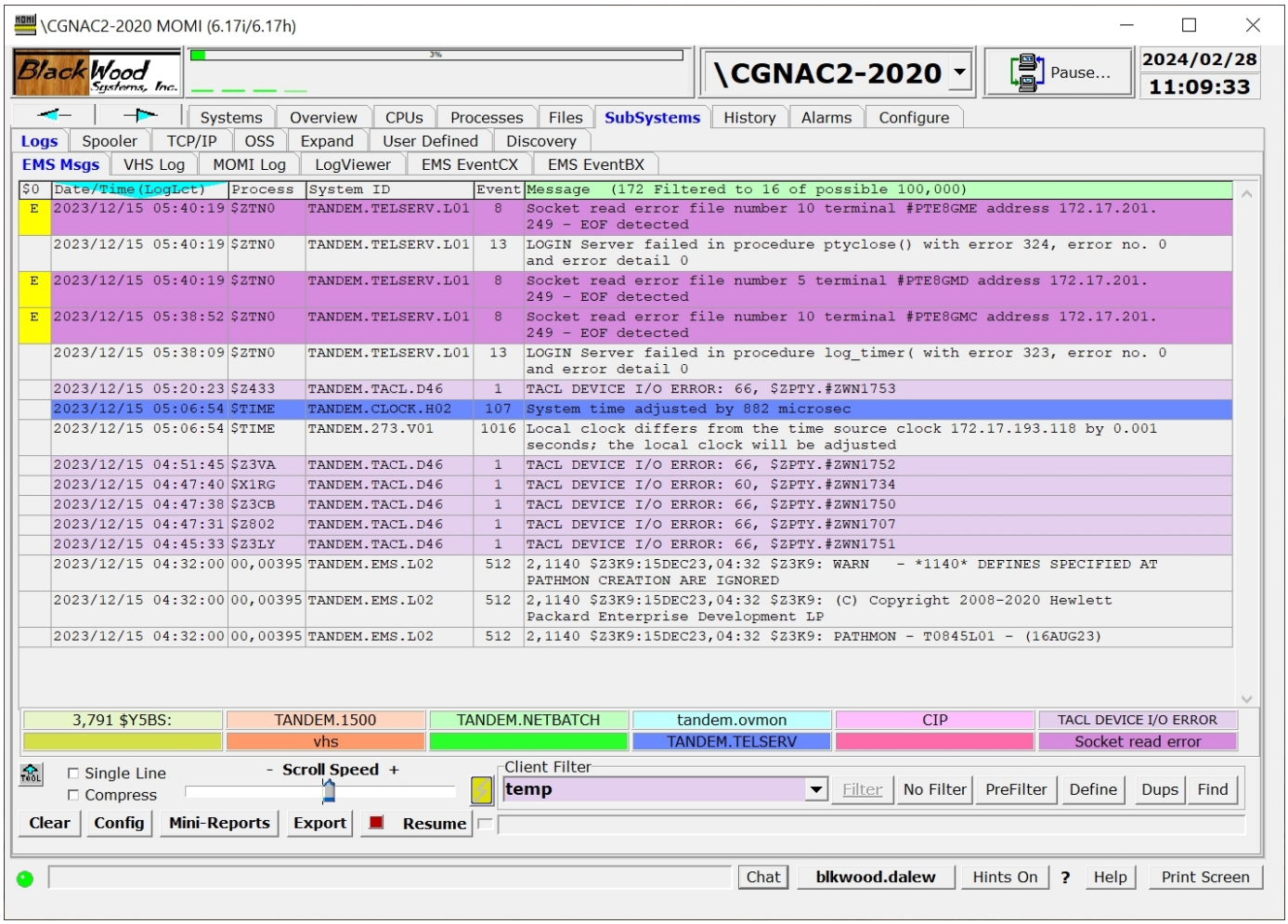


The **Dups** button filters records with the following matching information: Date and Time, Owner, SubSystem, SSID, Text

The **Find** button allows search within the messages stored within the EMS screen buffer with find next and find previous functionality.

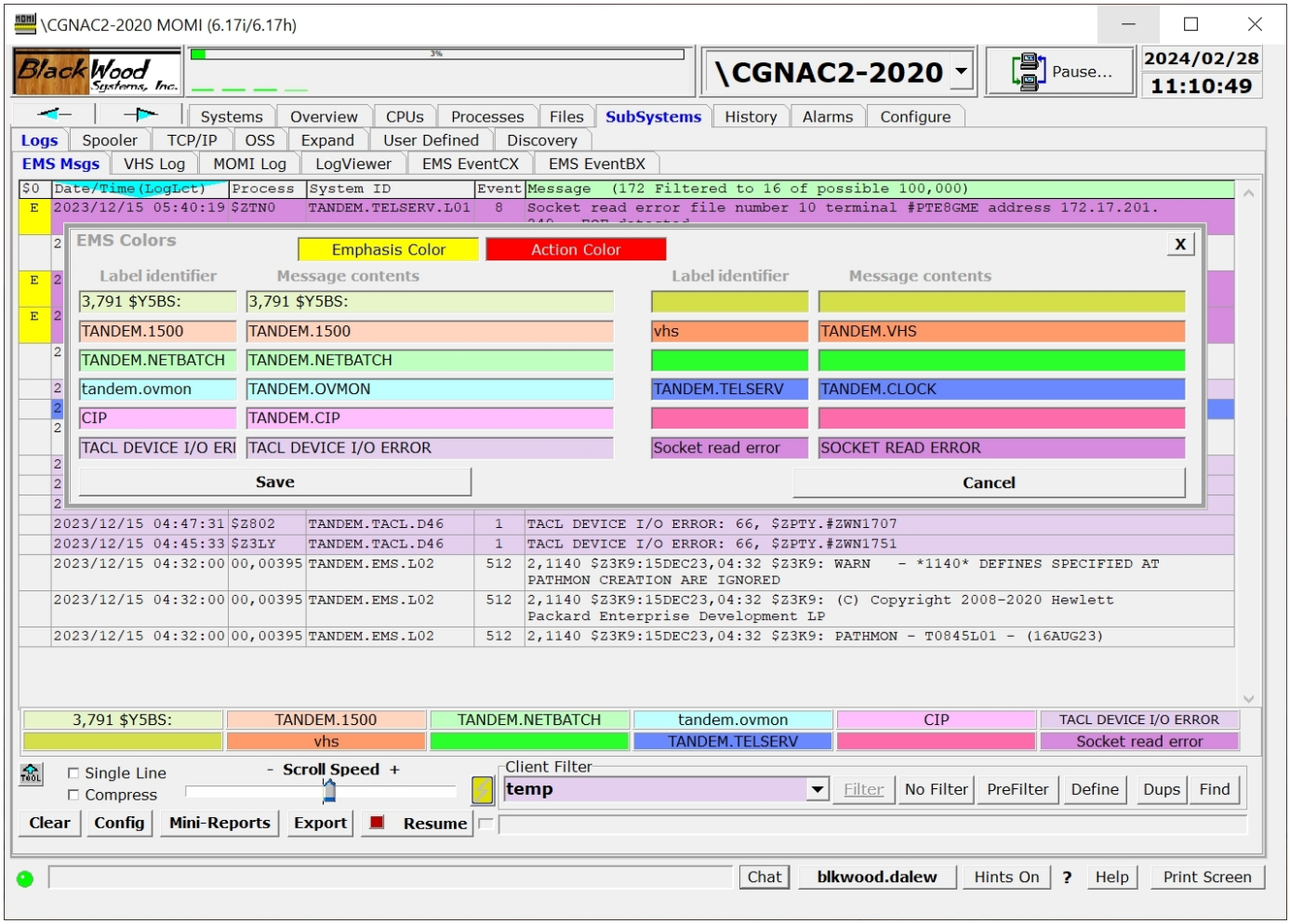


Messages may be color highlighted based on their content.



To activate color highlighting, right-click on the color guides near the bottom of the window and select **Display/Change Color Sections...** Enter the label text and message content text that should cause that color to be displayed. By default, the entered text is considered "one string" for comparison purposes. Multiple strings may be entered by separating them with a **|** (i.e. vertical bar)

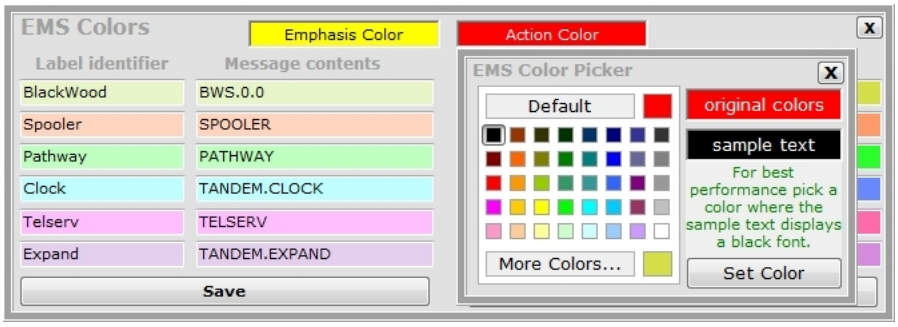
character. Enclose the entered text in quotes (i.e. " ") to retain leading and/or trailing spaces. Press **Save** when complete.



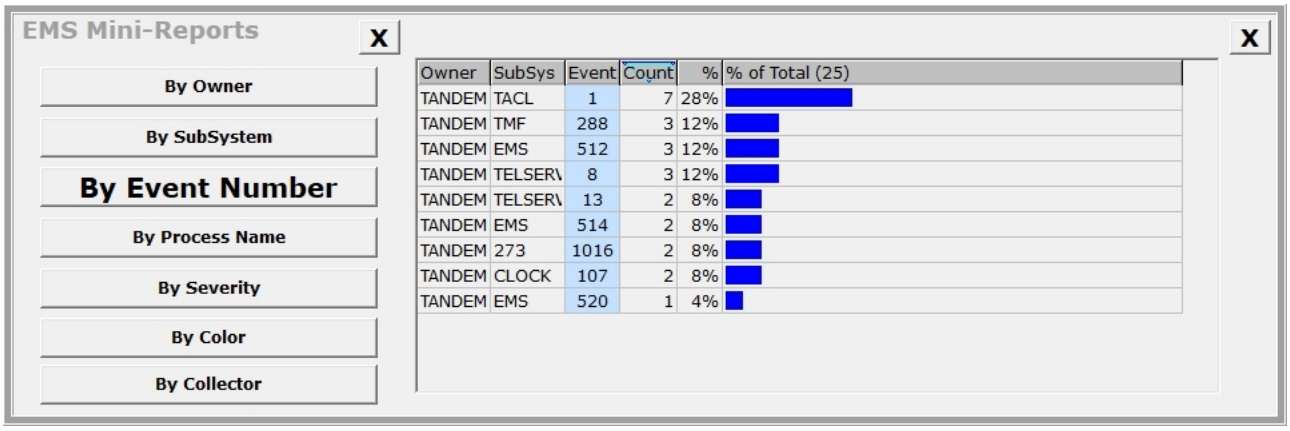
The scanning of text for color determination stops after the first match and is performed in the following order (in the image above starting at top left then down - then to top right then down):

light yellow
light orange
light green
light blue
light red
light purple
bright yellow
bright orange
bright green
bright blue
bright red
bright purple

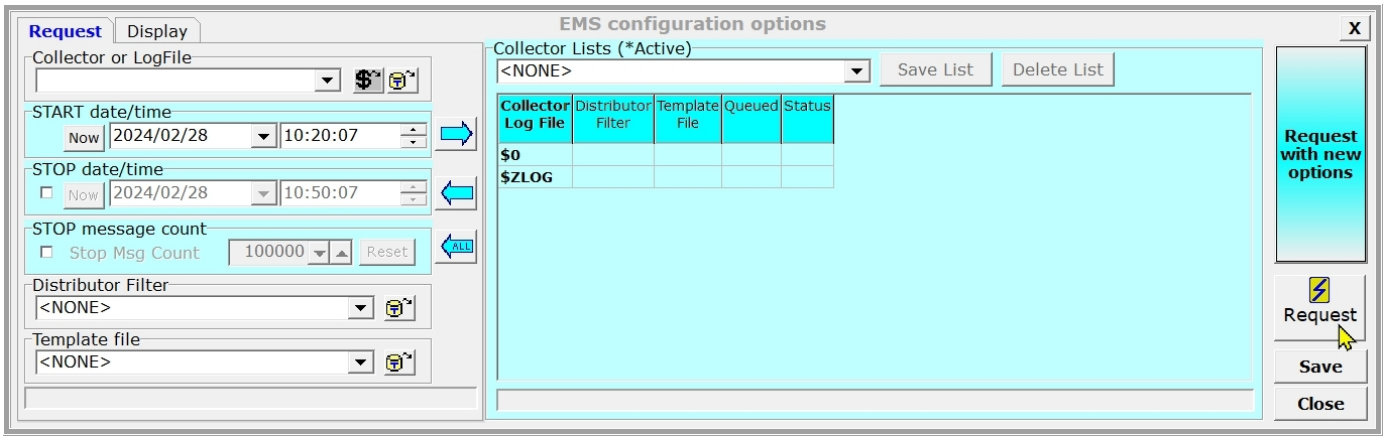
The on screen colors maybe altered by right-clicking on the color then select **Set Color**. In the **More Colors...** select the desired color.

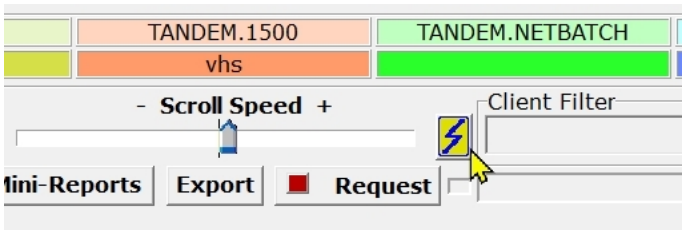


The **Mini-Reports** button displays a pop-up to provide a quick summary of counts by different criteria. The totals are computed based on the messages currently displayed and in the message buffer (i.e. these are the messages you can scroll up/down and see on the screen).

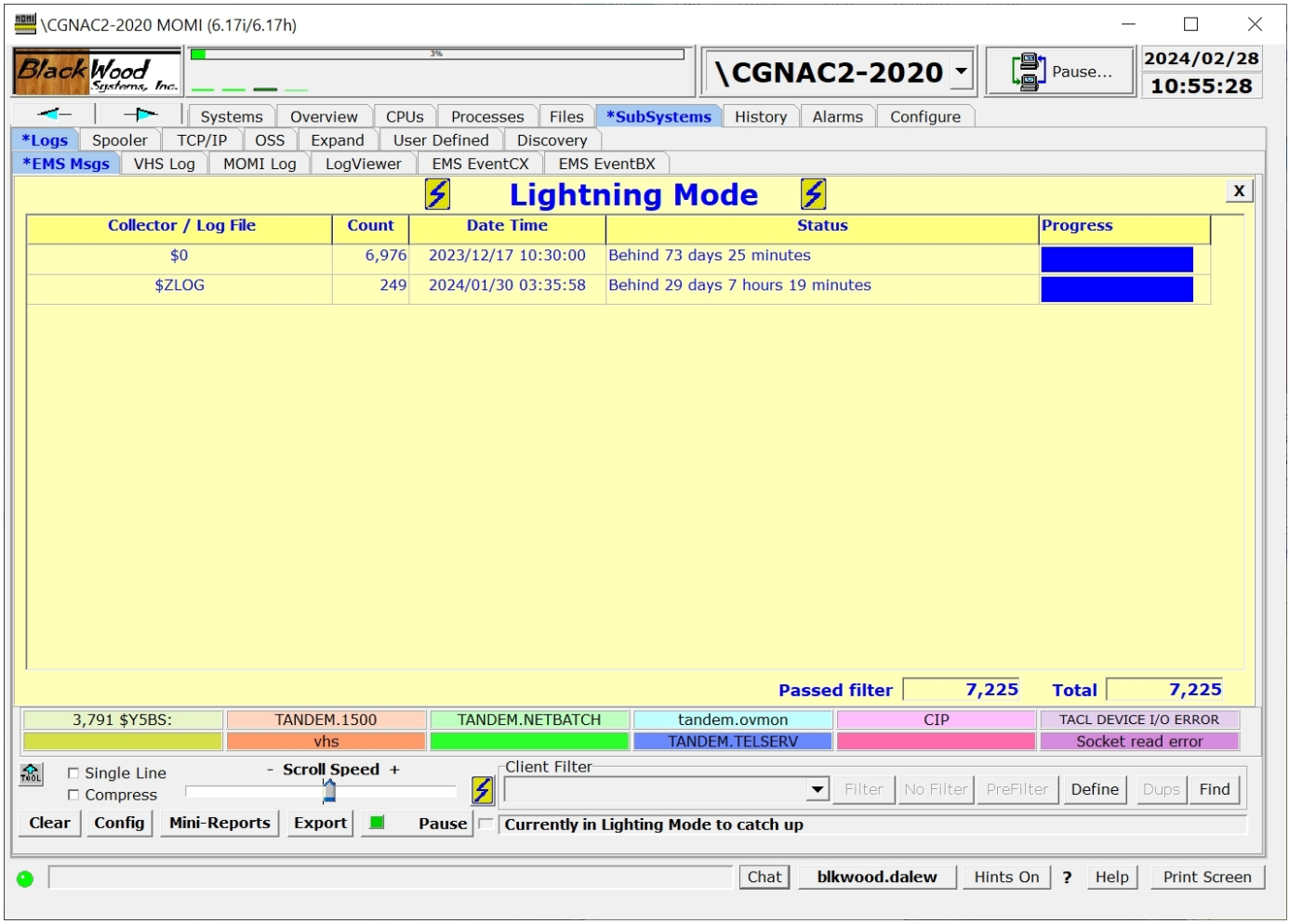


The screen has a Lightning Mode feature that loads EMS messages as quickly as possible. After selecting the desired logs and setting other options, push the **Request** button with the Lightning bolt. Lightning mode may also be entered by pressing the Lightning bolt next to the **Scroll Speed** slider.

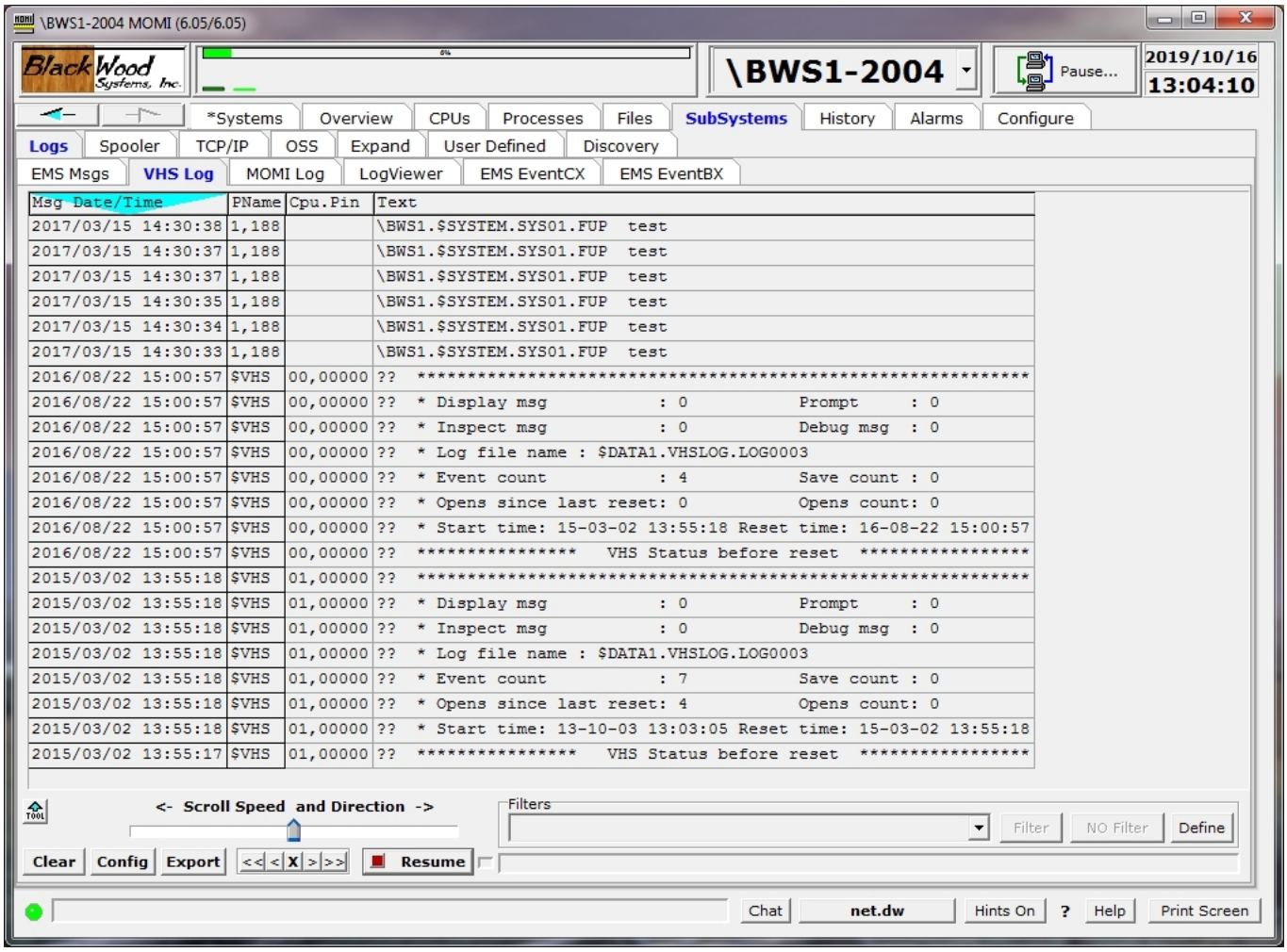




Entering **Lightning Mode** greatly reduces screen updates in order to allow incoming data to flow as quickly as possible. The Lightning Mode status screen remains until all Collector / Log Files reach the current time (or their configured stop time), End of File is reached or the pop-up itself is closed by clicking on the X. No EMS events are lost by entering, exiting, or re-entering Lightning Mode (i.e. you can freely exit and re-enter this mode as needed). The number of EMS events held in memory is still determined by Msg Count previously mentioned.


\CGNAC2-2020 MOMI (6.17i/6.17h)
BlackWood Systems, Inc.
\CGNAC2-2020
Pause...
2024/02/28
10:55:28
Systems
Overview
CPUs
Processes
Files
*SubSystems
History
Alarms
Configure
*Logs
Spooler
TCP/IP
OSS
Expand
User Defined
Discovery
*EMS Msgs
VHS Log
MOMI Log
LogViewer
EMS EventCX
EMS EventBX
Lightning Mode
Collector / Log File
Count
Date Time
Status
Progress
$0
6,976
2023/12/17 10:30:00
Behind 73 days 25 minutes
$ZLOG
249
2024/01/30 03:35:58
Behind 29 days 7 hours 19 minutes
Passed filter
7,225
Total
7,225
3,791 $Y5BS:
TANDEM.1500
TANDEM.NETBATCH
tandem.ovmon
CIP
TACL DEVICE I/O ERROR
vhs
TANDEM.TELSERV
Socket read error
Single Line
Compress
- Scroll Speed +
Client Filter
Filter
No Filter
PreFilter
Define
Dups
Find
Clear
Config
Mini-Reports
Export
Pause
Currently in Lighting Mode to catch up
Chat
blkwood.dalew
Hints On
?
Help
Print Screen

VHS Log



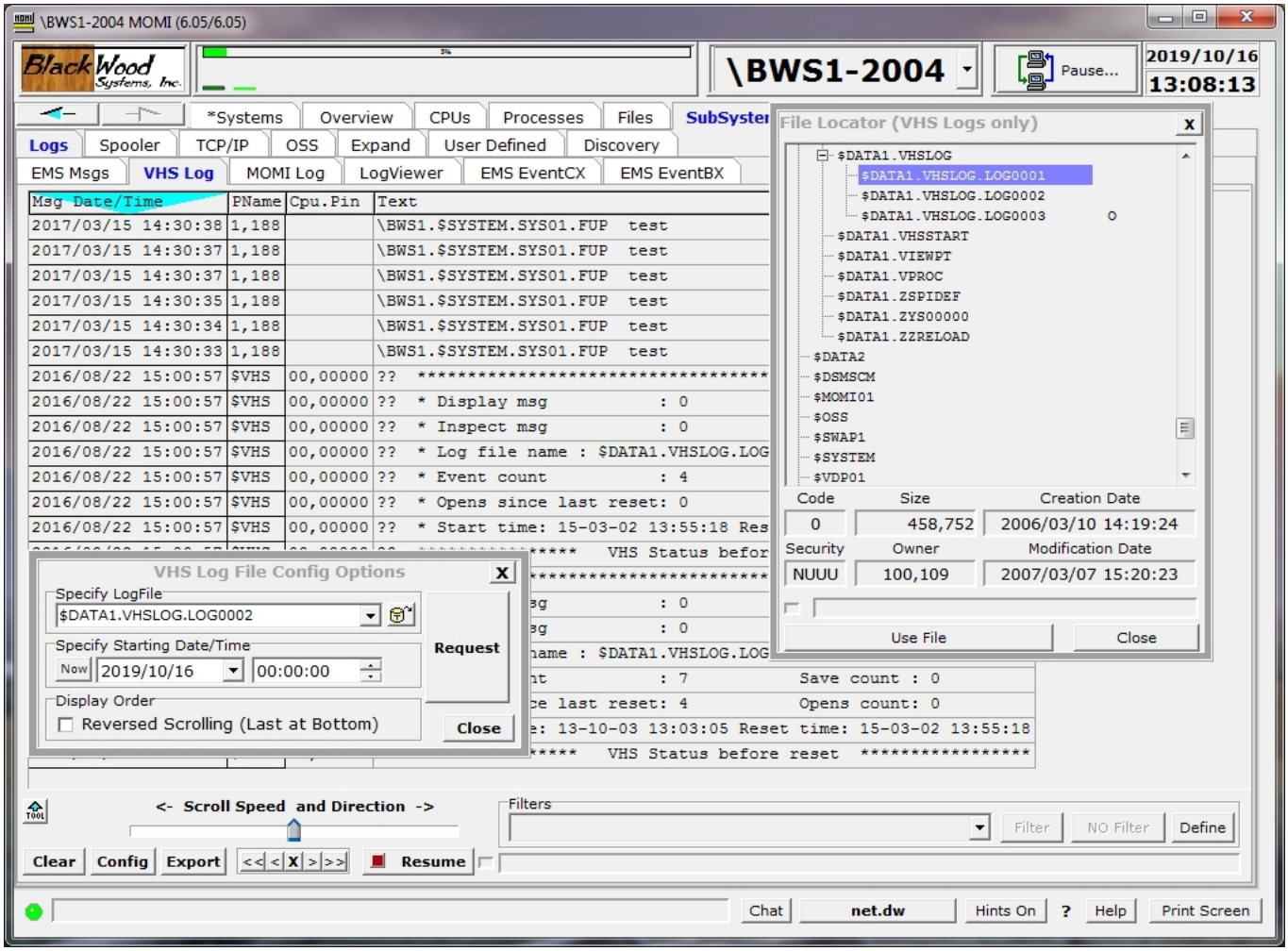
The VHS Log screen presents information collected by a VHS process.

VHS, or the Virtual Hometerm Subsystem, provides a stable location for processes to log message usually related to a problem or status update. Messages received by VHS are logged to a file which may be viewed on this screen. If the screen is going forward in time and reaches the end of file, new messages are automatically displayed shortly after they are written. However, if VHS switches to a new log file, the screen will stop displaying new messages. The user must manually select a new log file in order to continue viewing new messages.

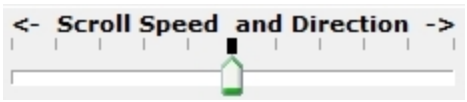
The log file used by VHS does not have a unique file code assigned by default, so It is necessary to logon to MOMI and have the necessary security to view a VHS Log.

To start viewing VHS Log files, press the Config button and Specify a log file. To aid in selecting a log file, press the locator button to display the File Locator pop-up window. Unfortunately, MOMI cannot automatically determine the active LogFile since VHS does not provide such a mechanism.

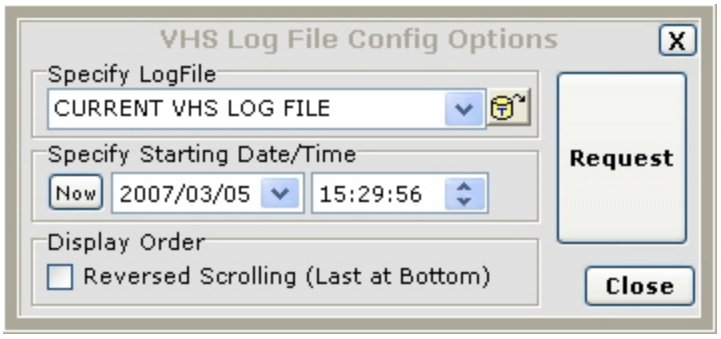
However, using the File Locator, *usually* the file with the highest sequence number is the current file.



Once a file is selected, you may scroll continuously at different speeds with the slider, forward or back one line at a time < >, 10 lines at a time << >> or stop X.

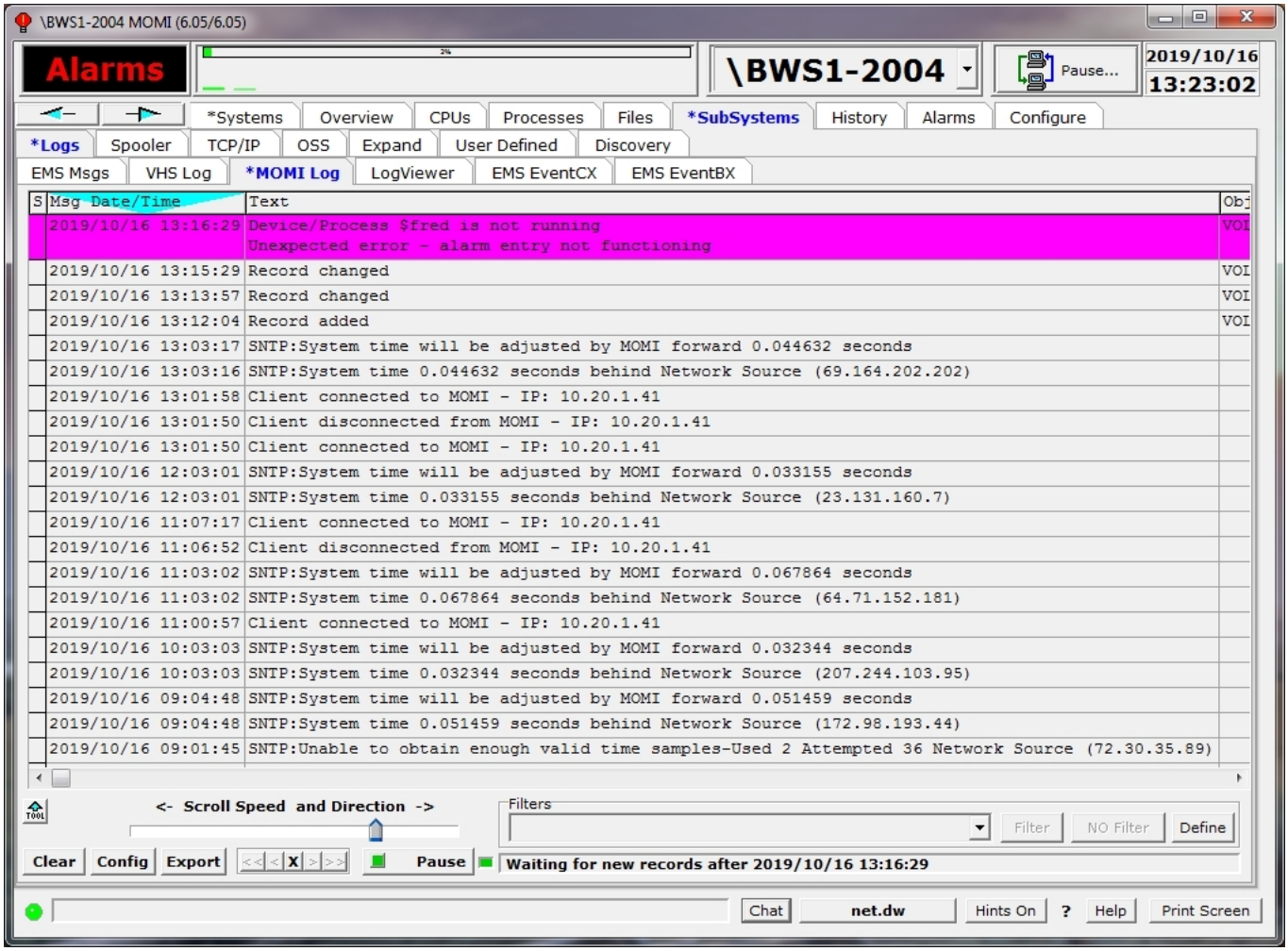


A default location where the File Locator drills-down to may be set with the CONFMOMI keyword SELECTION-DEFAULT-VHS. This keyword provides two functions. First, the File Locator automatically drills down to the subvolume specified (the File Locator may display files other than VHS log files). Second, the special name of "CURRENT VHS LOG FILE" appears and will automatically select the current VHS Log file based on modification date. The automatic selection only occurs when **Request** is pressed.



After selecting the log file, press **Request**. Use the scroll bar to move forward and backward in time through the file.

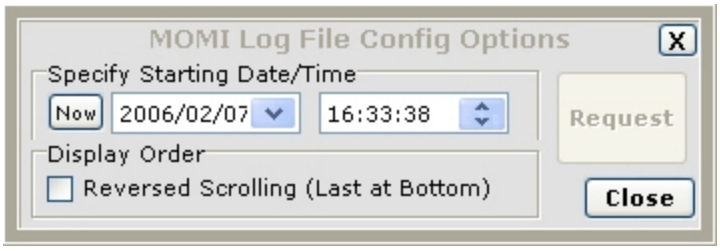
MOMI Log



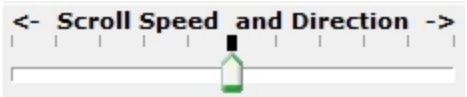
The MOMI Log screen displays records from the MOMI log file, provides an export capability and the definition of an export script. Records in this file are specific to MOMI.

Pressing the Request button, when the screen is clear, causes the client to read and display the last or most recent records in the file. The client then goes into a mode where new records are automatically displayed. Once data is display, used the slider to scroll forward and backward in time through the file.

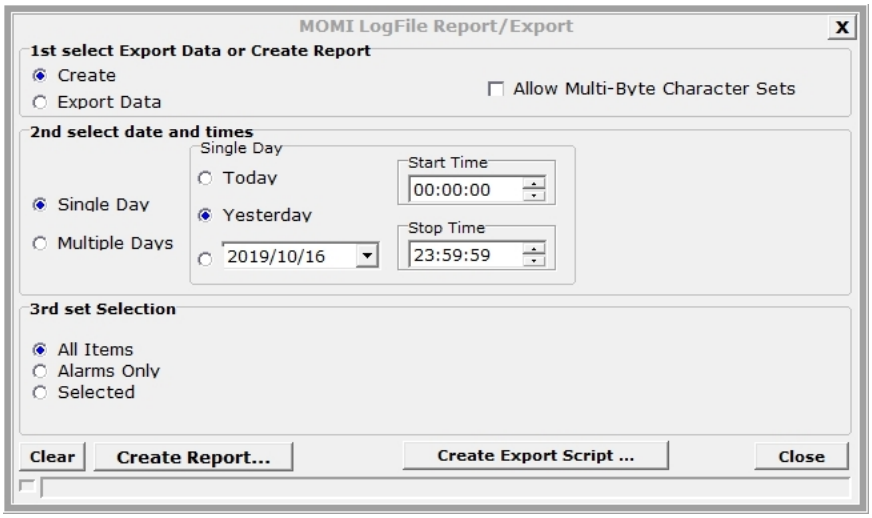
The Config button is used to configure the starting location for record retrieval.



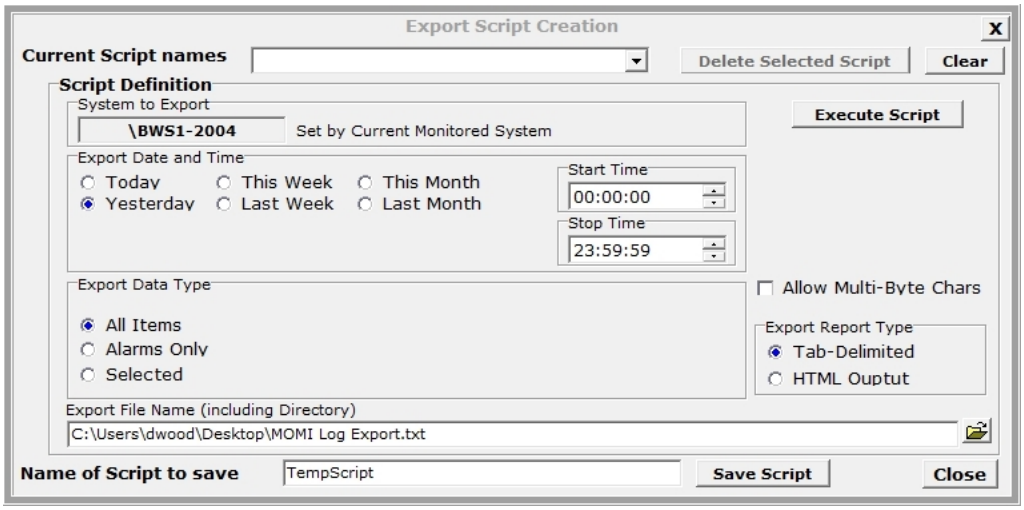
Once a starting time is selected, you may scroll continuously at different speeds with the slider, forward or back one line at a time < >, 10 lines at a time << >> or stop X.



The **Export** button is used to generate a report, export records or create an export script. To create a report, under 1st select Create, under 2nd select a single day or multiple days, and under 3rd select to report on all records or on specific Alarm records. Press Create Report... to begin generation (a pop-up window is displayed to select the file name and location).

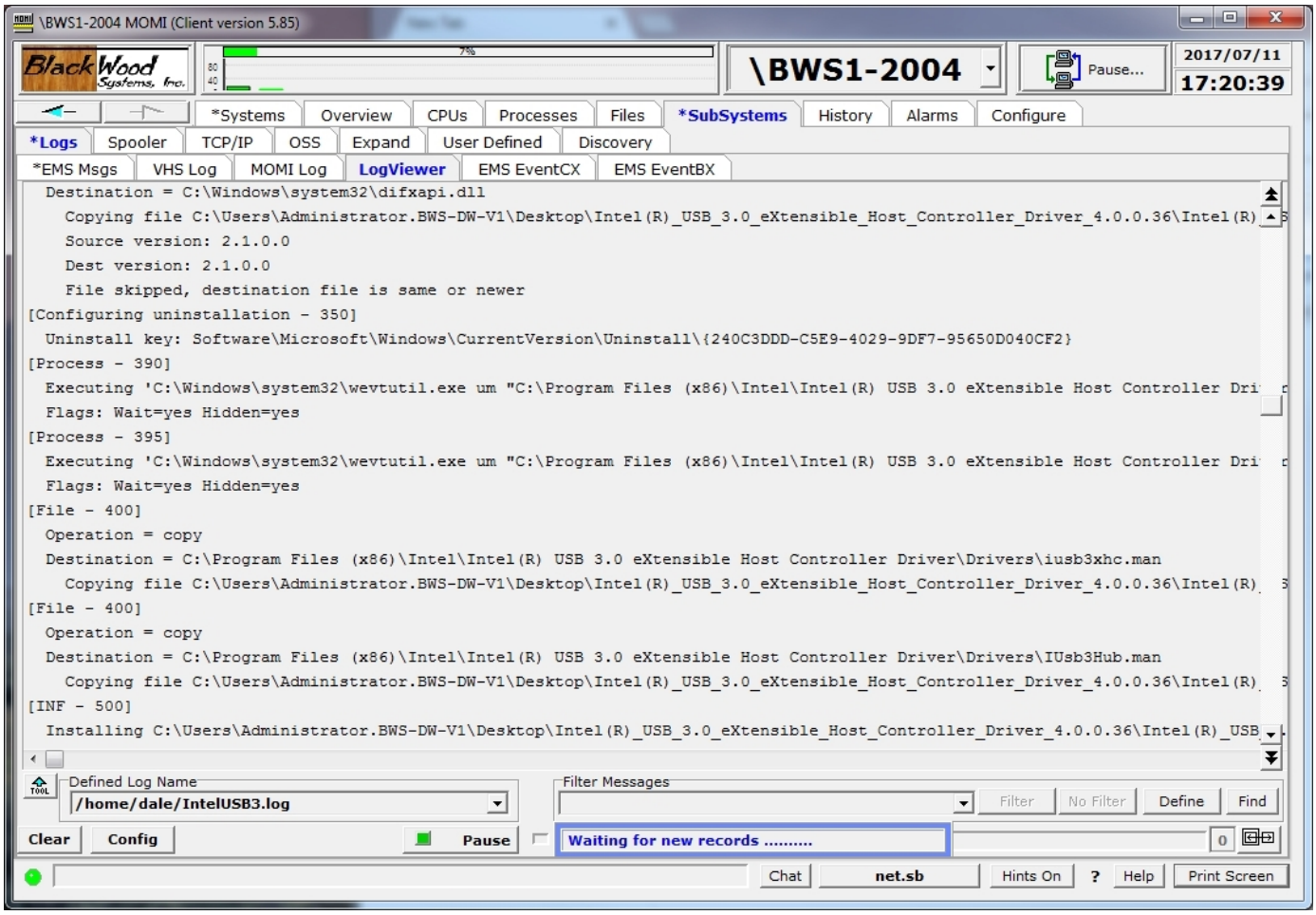


The **Create Export Script** button provides a means to define a script to define an export operation launched via the MOMI PC client run-time parameter SCRIPT-LOG. A script is defined and saved prior to referencing it via the run-time parameter. To place the current date within the Export File Name, in YYYYMMDD format, insert the field <date> (see the tool-tip over this field for an example).

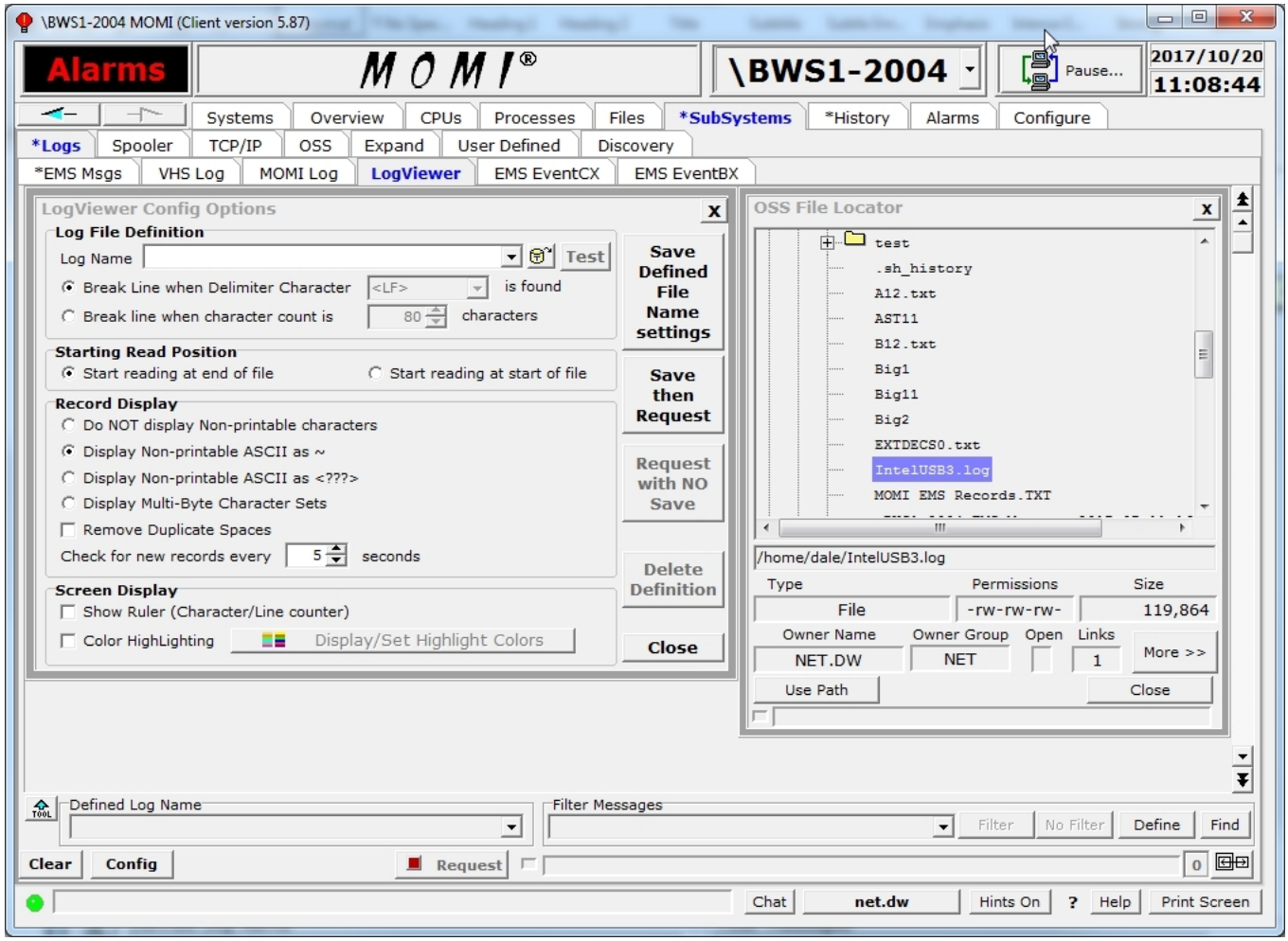


See LOG01DB for additional information.

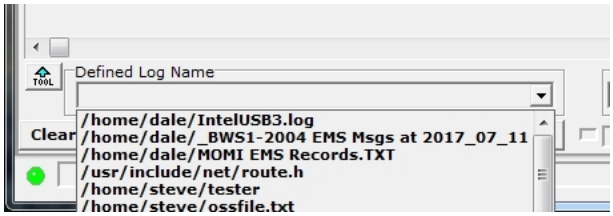
LogViewer



The LogViewer screen displays OSS text files generated by a subsystem. The user selects the file, defines the line terminator characters, and how unprintable characters are handled. When the end of the file is reached, the screen enters a mode where new data appended to the end of file automatically displays.

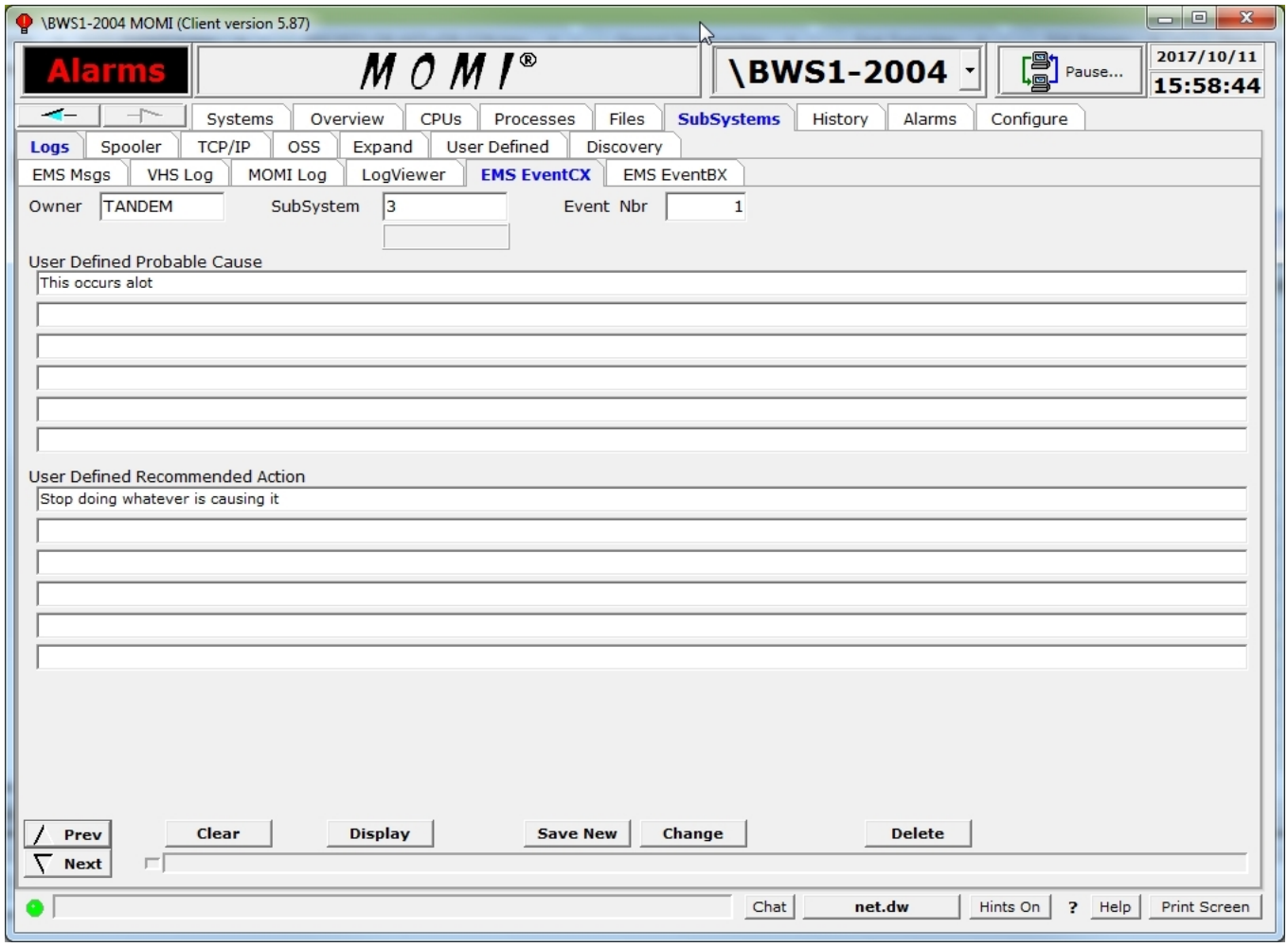


Using the screen for the first time, push the **Config** button. A pop-up options window displays. Next push the file locator button (round disk with T and arrow to the right of **File Name**). Navigate to the desired file and push **Use Path** to load the **File Name** field. The **Starting Read Position** section determines reading from the end of the file or the beginning. The **Record Display** section determines how unprintable characters are handled and frequency of checking for new data once end of file is reached. The **Screen Display** area allows display a ruler at the top of the window area and enabling of text color highlighting based on message content. Once options are set and push **Save then Request** to start data display.



After files are defined, they may be reused by pressing the drop-down arrow on the **Defined Log Name** box.

EMS EVENTCX



The EMS EVENTCX screen allows the creation, display and maintenance of records in the EVENTCX file. This file is defined by Viewpt and by default is **EMPTY**.

The EVENTCX file allows a user defined probable cause and recommended action for EMS events. On the EMS Msgs screen, when an event is displayed in detail, content of this record is available if a matching Owner, SubSystem and Event Number is found.

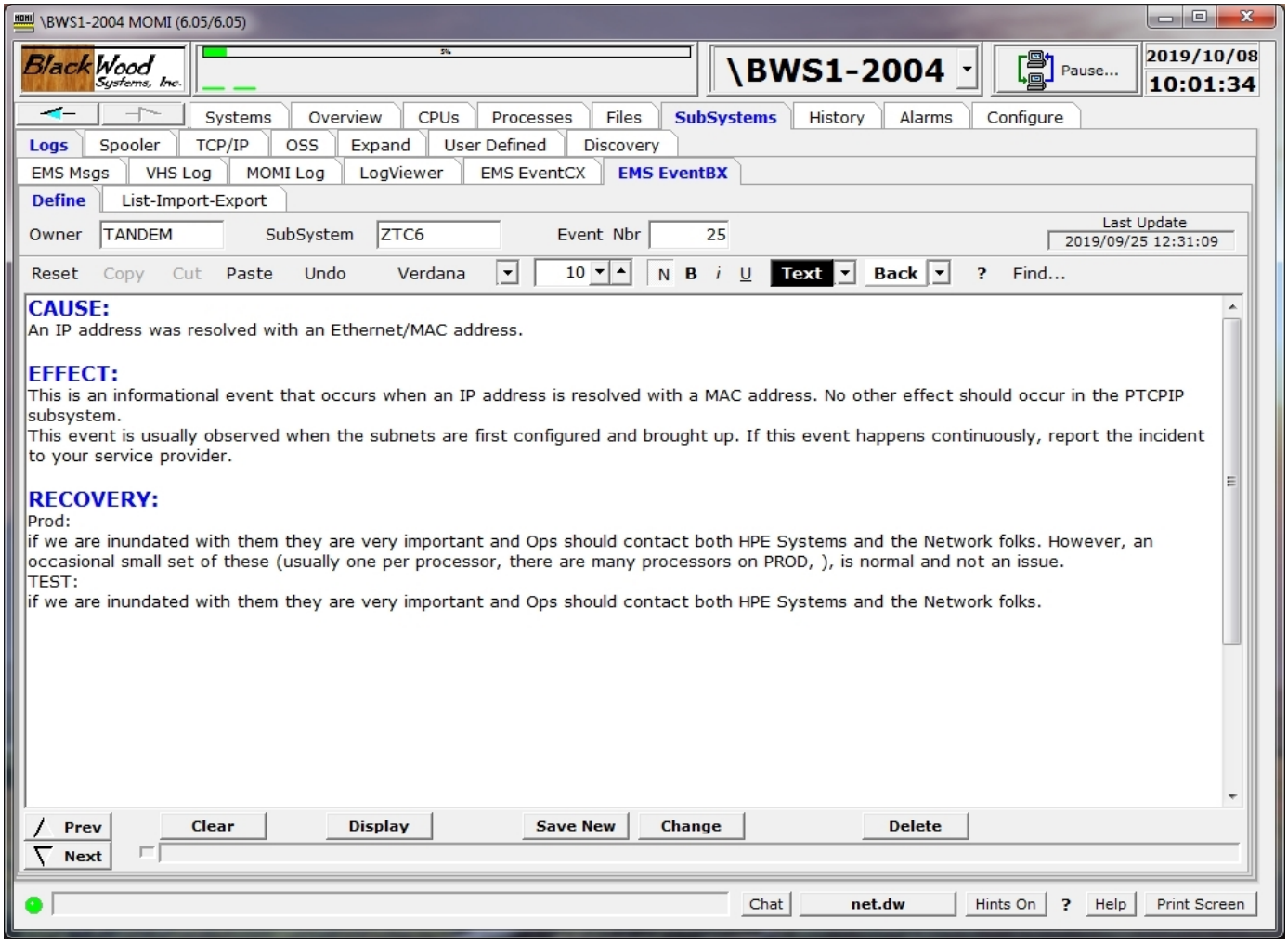
The field layout of this file is not really 'user friendly' as it stores the SubSystem number and not the SubSystem name. For example, Tandem.8 is the equivalent of Tandem.Pathway. MOMI attempts to help in this area as it will automatically convert a SubSystem name into the SubSystem number if it is valid for that Owner. It is necessary that EMS Templates be present (which is built-in for Tandem subsystems) for this conversion to occur.

The **Prev** and **Next** buttons allow scrolling forward and backwards through existing records in the file. Pressing **Next** with a blank screen displays the first record in the file. Pressing **Prev** with a blank screen displays the last record in the file.

See the CONFMOMI keyword EVENTCX for additional information.

EMS EventBX

EMS EventBX / Define



The EMS EventBX / Define screen provides for the creation and maintenance of user supplied information associated with an EMS message. The information is keyed by an EMS SSID (also known as the Owner.Subsystem) and event number.

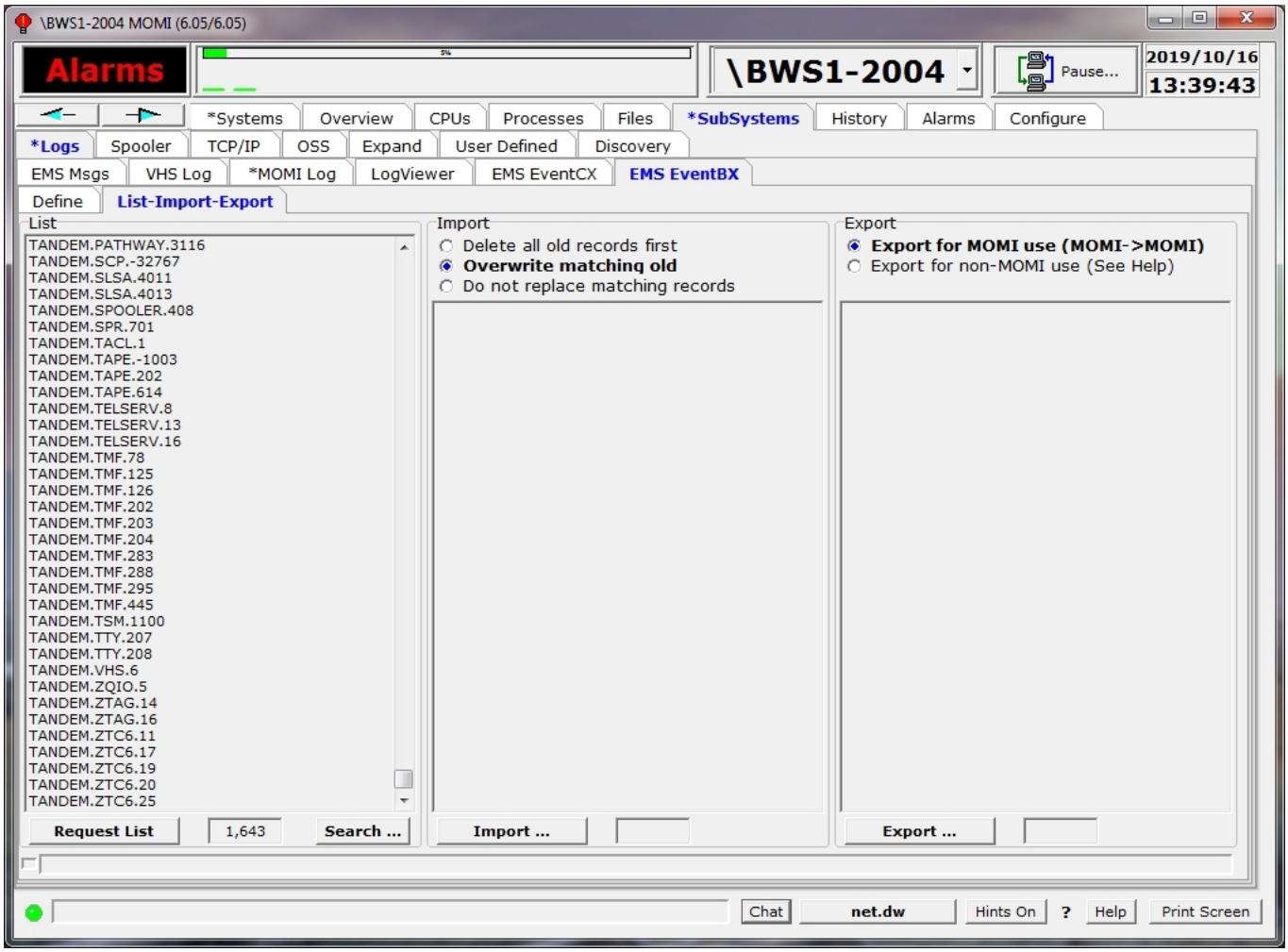
Information entered here is available on the EMS Msgs detail pop-up and could be considered a super-set of the EventCX file functionality.

EMS records by themselves may not contain enough information for an operator or system administrator to determine an action. This screen provides a method for storing additional details and/or actions associated with an EMS message. The main entry field is free form, virtually unlimited in size and available to all users connected to the same system.

To create records, enter the EMS **Owner**, **SubSystem** and **Event Nbr**, desired text, and press **Save New**. Existing entries may be displayed, altered or deleted. Note that the first three fields entered compose the primary key for the record.

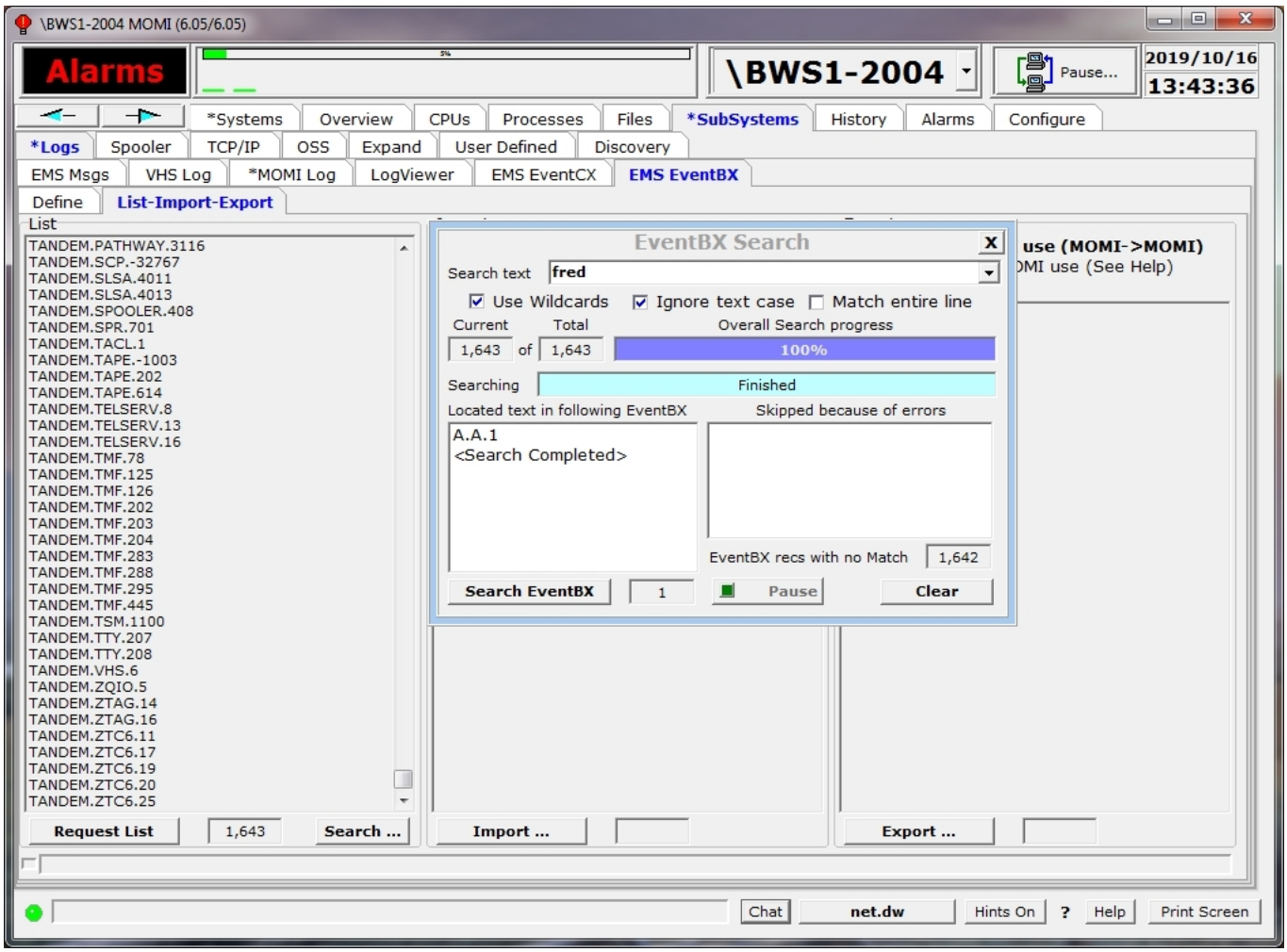
The tool bar across of the text entry area is used to format text such as selecting different fonts or choosing bold, underline and italic to selected text. Information may also be copied and pasted from external sources.

EMS EventBX / List-Import-Export



The EMS EventBX / List-Import-Export screen provides for the listing of existing definitions, searching, import, and export.

A list of current definitions displaying the Owner.SubSystem.Event Nbr is obtained by pushing **Request List**. Clicking an item in the list jumps to the Define screen and displays the record.



The search function scans all existing records for the text specified. Enter the **Search text** then press **Search EventBX**. Clicking on the an entry in the **Located...** column jumps to the record.

To copy EventBX data from one MOMI environment to another, under **Export...**, select **Export for MOMI Use (MOMI -> MOMI)**. Connect to another MOMI environment and then select **Import**. Options are provided to completely replace any existing definitions, just overwrite existing definitions (which leaves existing non-matching data alone) or don't replace any existing definitions. Note that the exact same MOMI PC Client version must be used for both export and import.

Export or Import of non-formatted records uses an ASCII text file comprising one line per record with the following format:

Owner <ETB> **SubSystem** <ETB> **Event Nbr** <ETB> **Text** <CRLF>

Where -

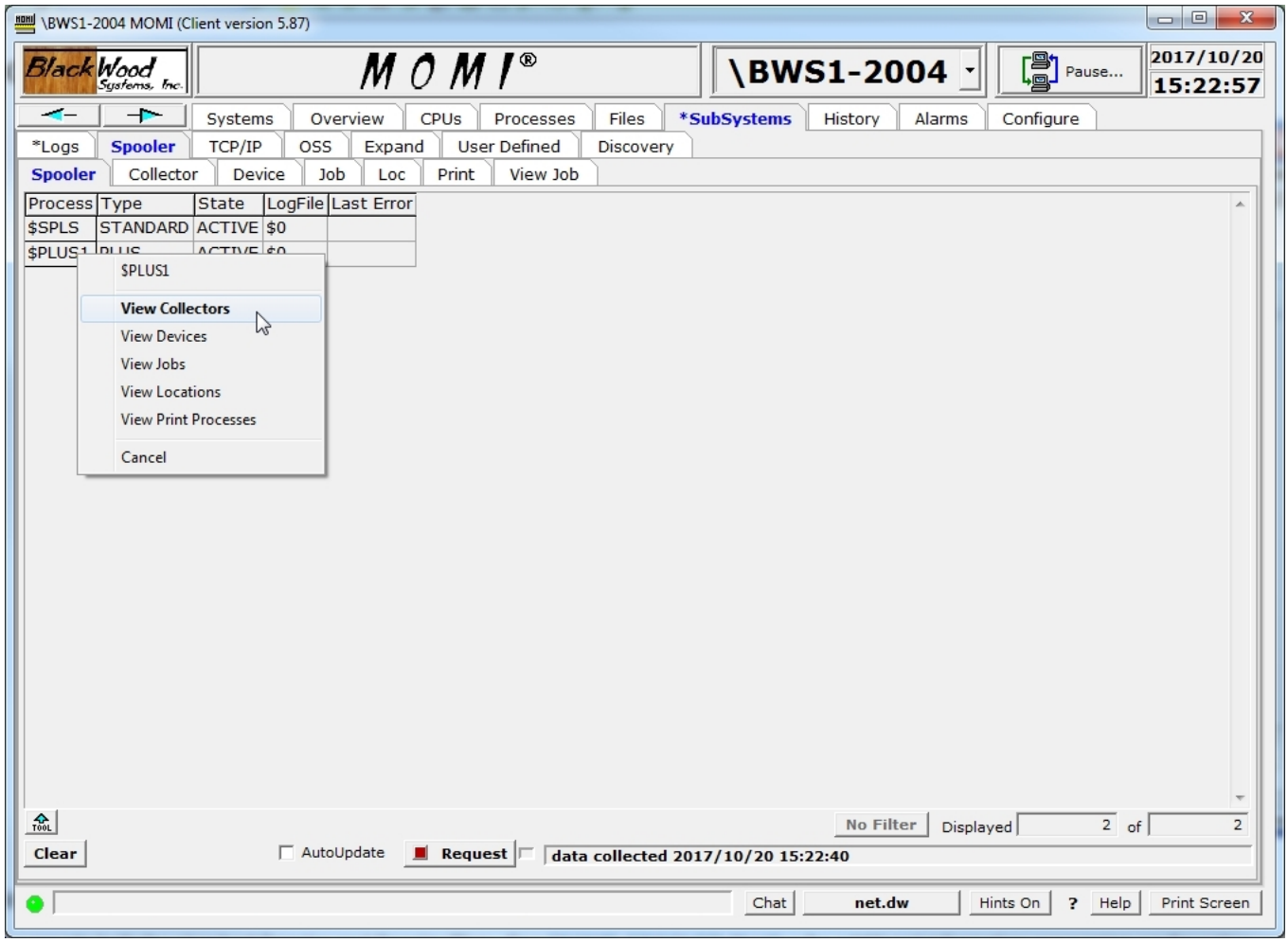
Owner is the overall creator of the record (i.e. Tandem, BWS, etc...)

SubSystem is the group of software generating the record (i.e. Pathway, Storage, etc...)
Event Nbr is the numeric value of the message
Text is the line-by-line of information
any embedded CRLF within **Text** is converted to an ETX with a value of 3

ETB is the ASCII control character with the value 23
CRLF are the ASCII control characters carriage return line feed with the values of 13 + 10

Spool

Spool / Spooler

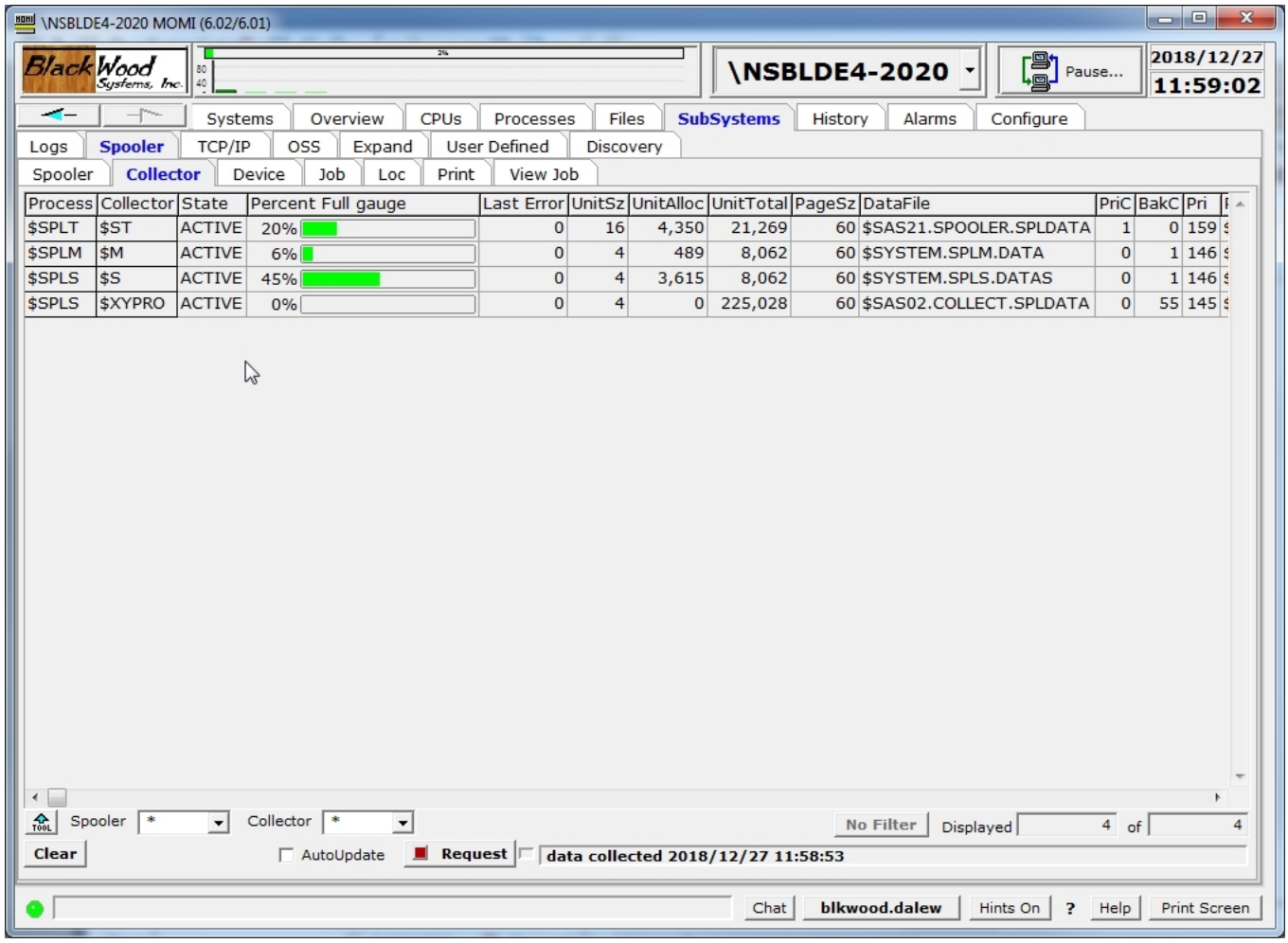


The Spool / Spooler screen presents the Spoolers present on the System and their state.

MOMI supports both the standard Spooler and Spooler Plus. See FNAME-SPOOLER-PLUS-n for additional information concerning configuration considerations for Spooler Plus.

The processes listed on this screen are the supervisor, or controlling process, for a Spooler. ;A Spooler is generally comprised of a single supervisor and then one or more of the following: collector, device, location and print process.

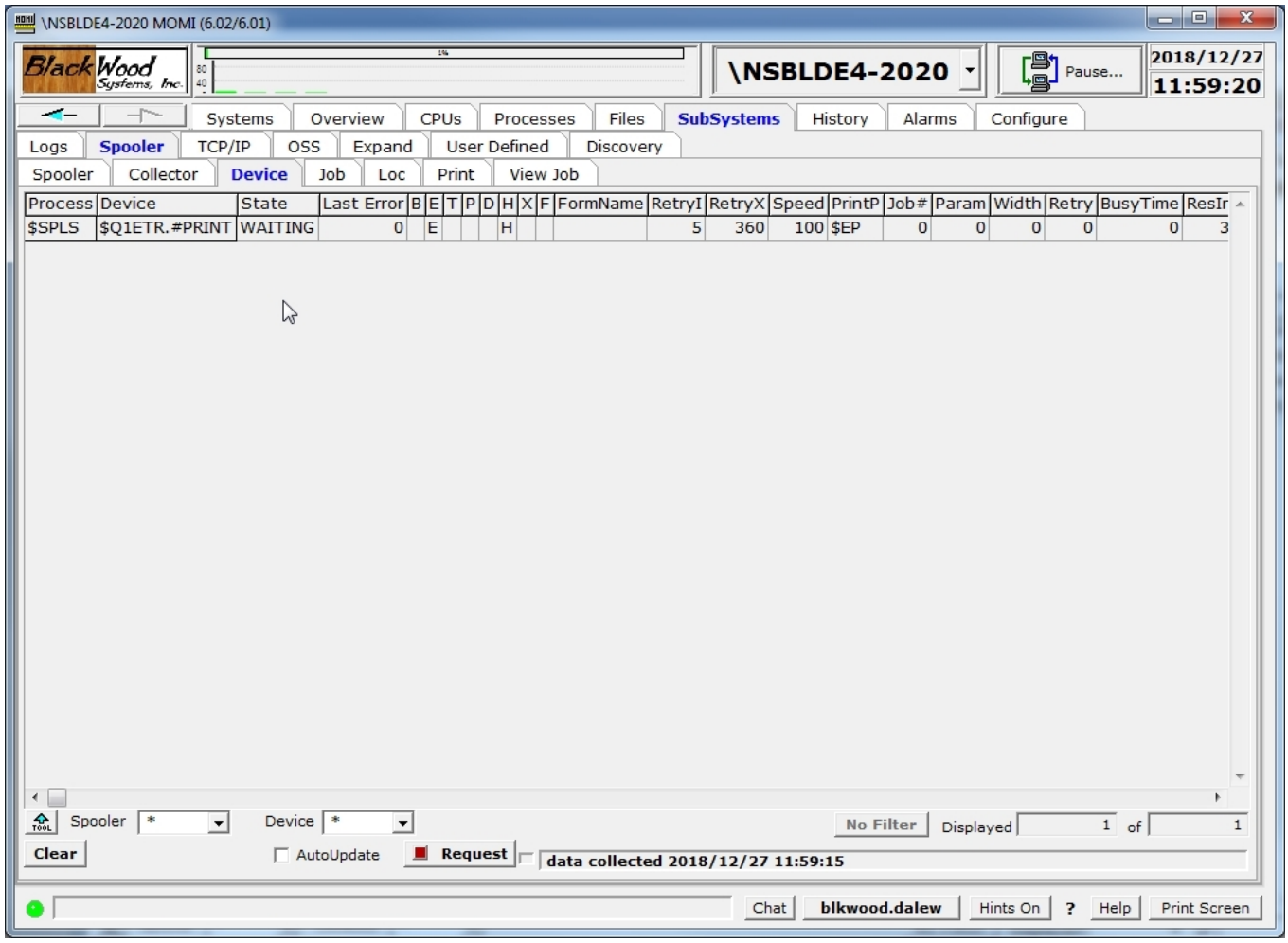
Spool / Collector



The Spool / Collector screen presents the state and status of the Spooler collector processes.

A Spooler collector process is a system location where process output is directed. This output is usually formatted in a manner for printing. The collector process has an underlying data file where space for each output (i.e. job) is allocated. A space available in the data file is organized in units where a unit is equal to 1024 bytes.

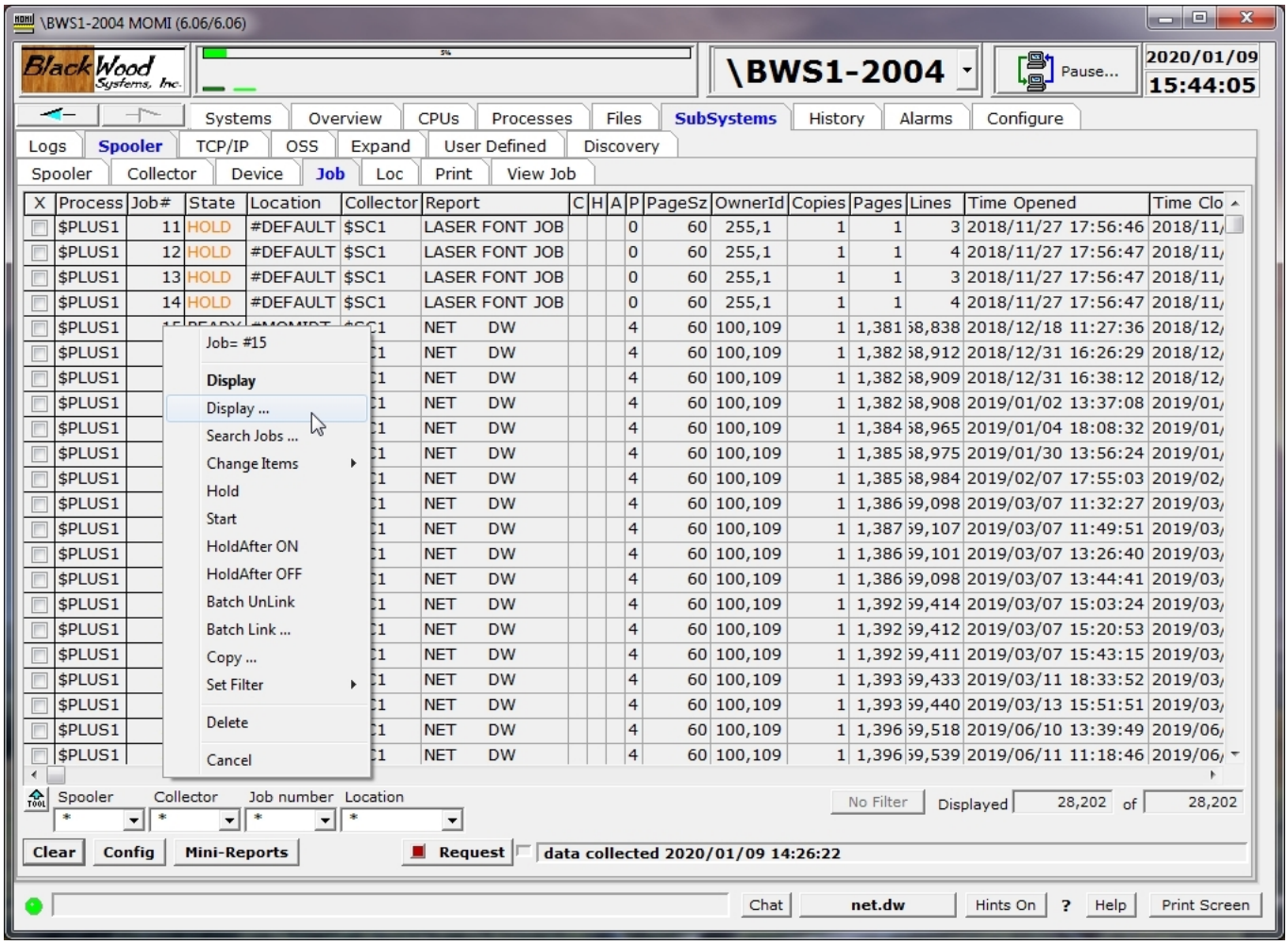
Spool / Device



The Spool / Device screen presents the status of the devices that handle generation of hard-copy from the Spooler.

Each hard-copy device in the Spooler is represented and associated with a print process. The print process is an actual process running on the NonStop System.

Spool / Job

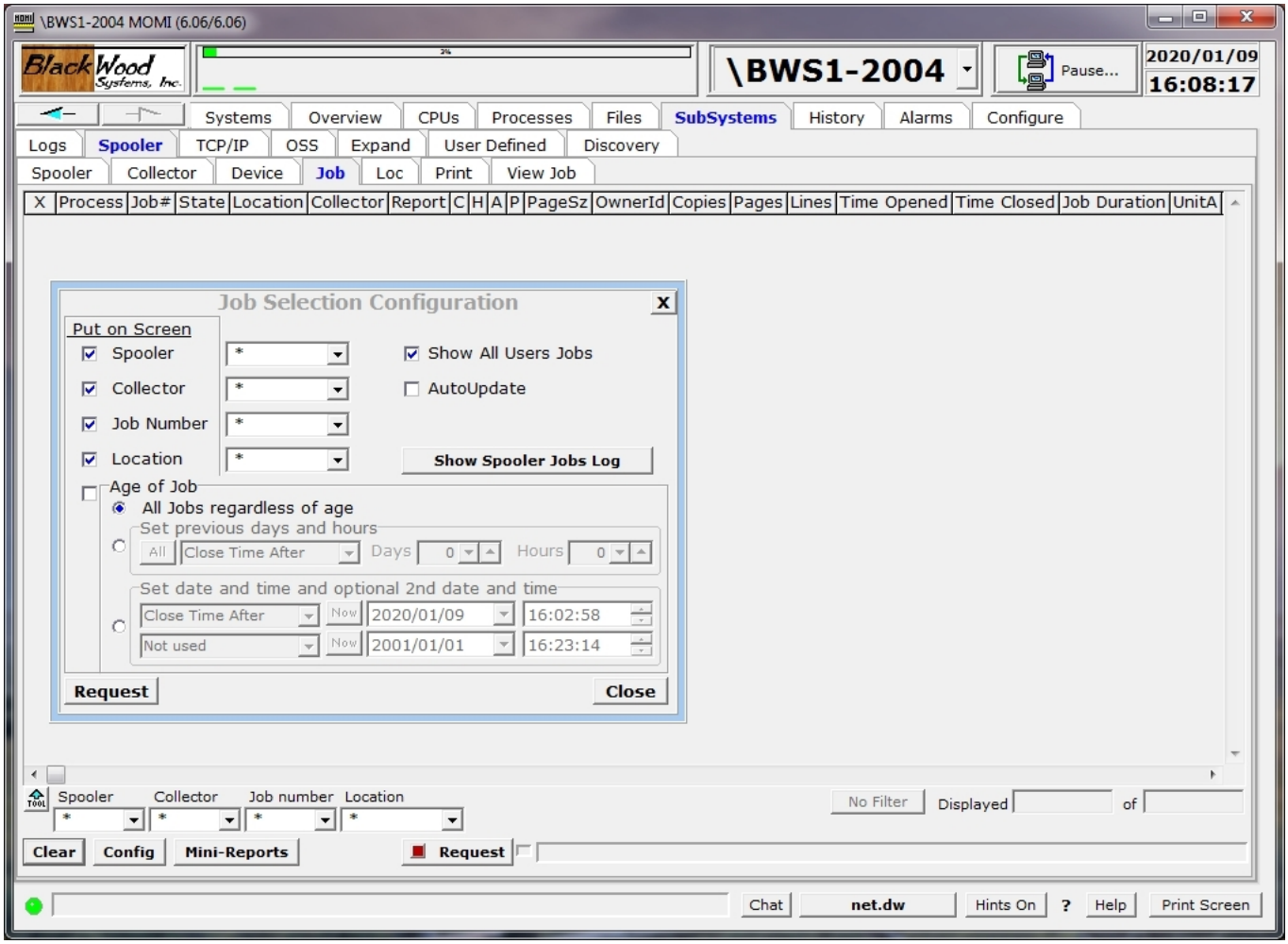


The Spool / Job screen presents the status and state of data waiting in the Spooler.

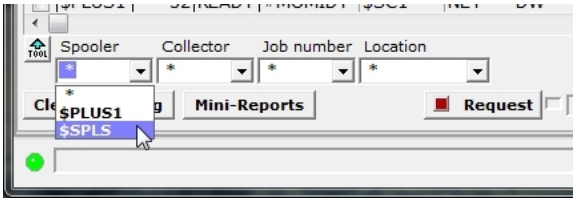
Data is organized in the Spooler in units known as jobs. Generally, a job represents the output of a single process sending data to the Spooler.

Jobs are displayed when the User presses the Request button. In the logged off state, all jobs are displayed. If a User is logged on, by default, only the user's jobs are displayed. Checking the **Show All Users Jobs** box under Config displays all jobs regardless of owner.

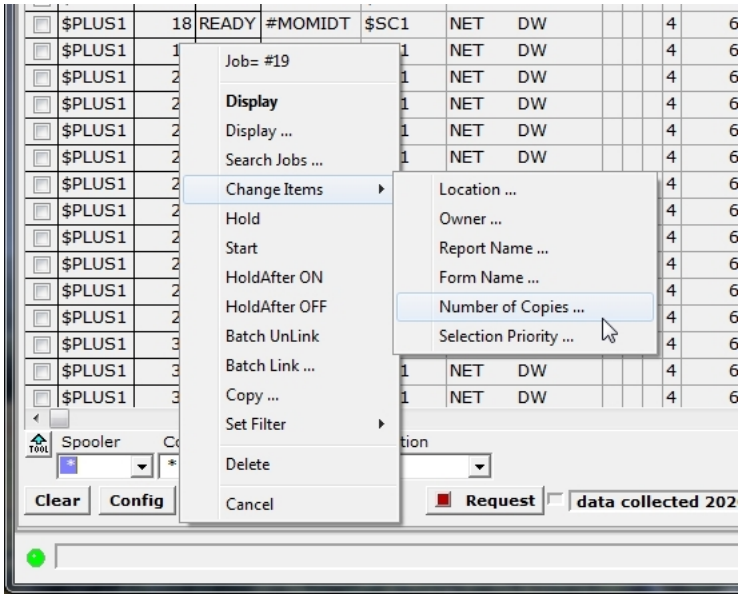
The Config button is used to determine what selections are available along the bottom of the screen by checking the box under **Put on Screen**



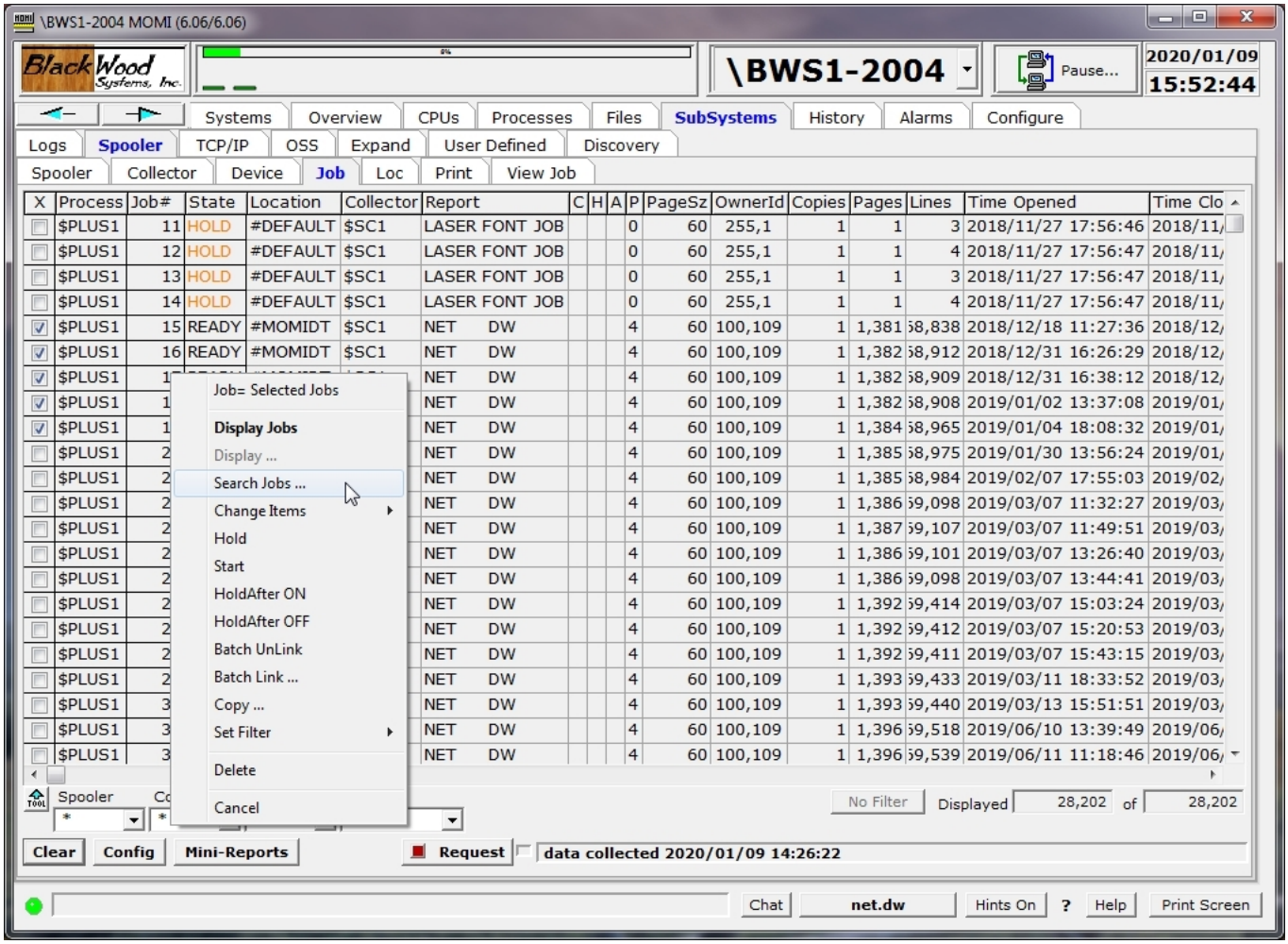
Display of Jobs may also be limited by selecting a particular Spooler and / or Collector in which to retrieve jobs.



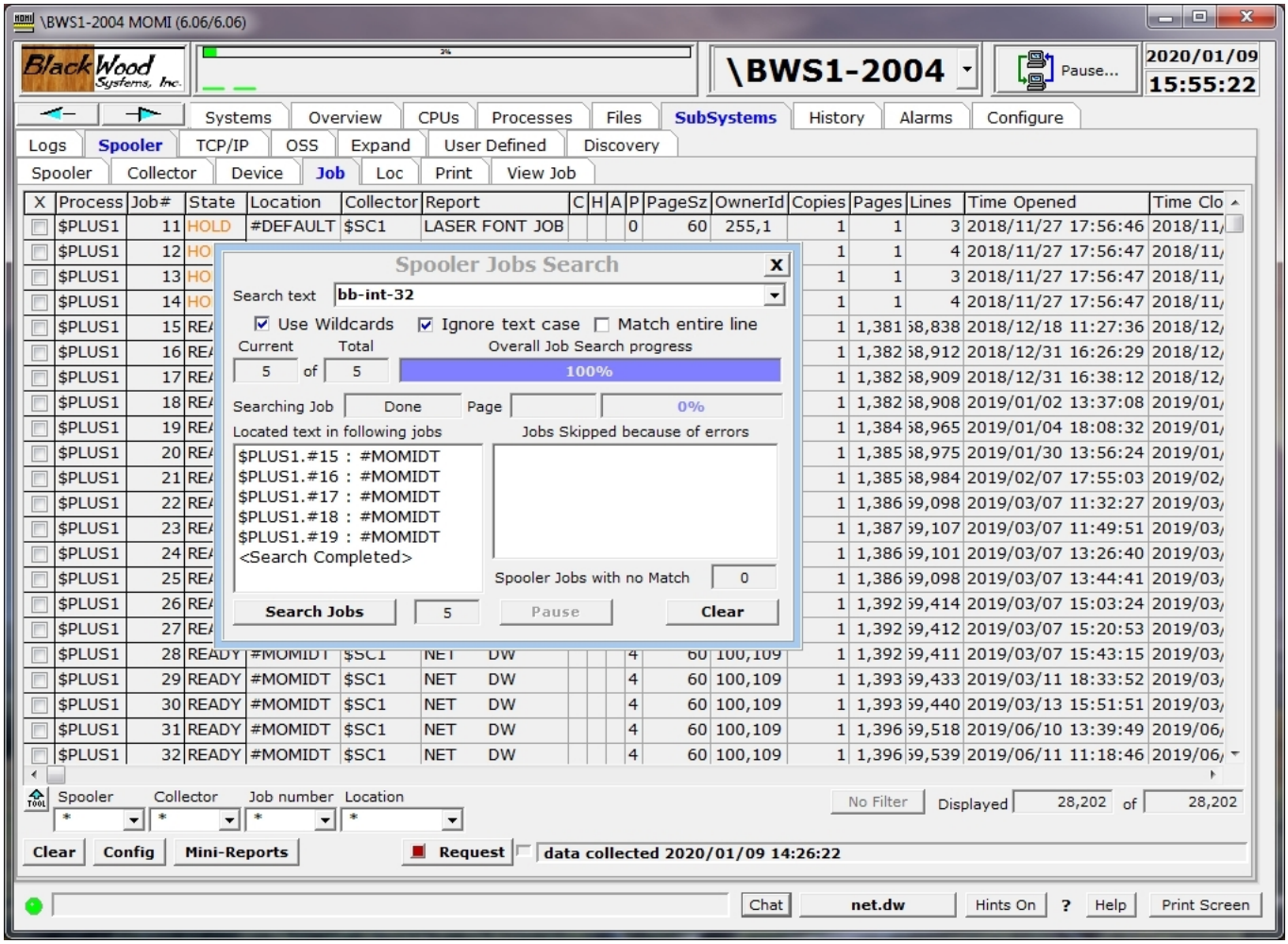
Right click on a job brings up a menu of actions. Actions, such as altering the Location or viewing the contents of a job, require the User to be logged on as the job owner. The Spooler subsystem, **not MOMI**, gives any user logged on under a "Super Group" the right to perform an action regardless job ownership.



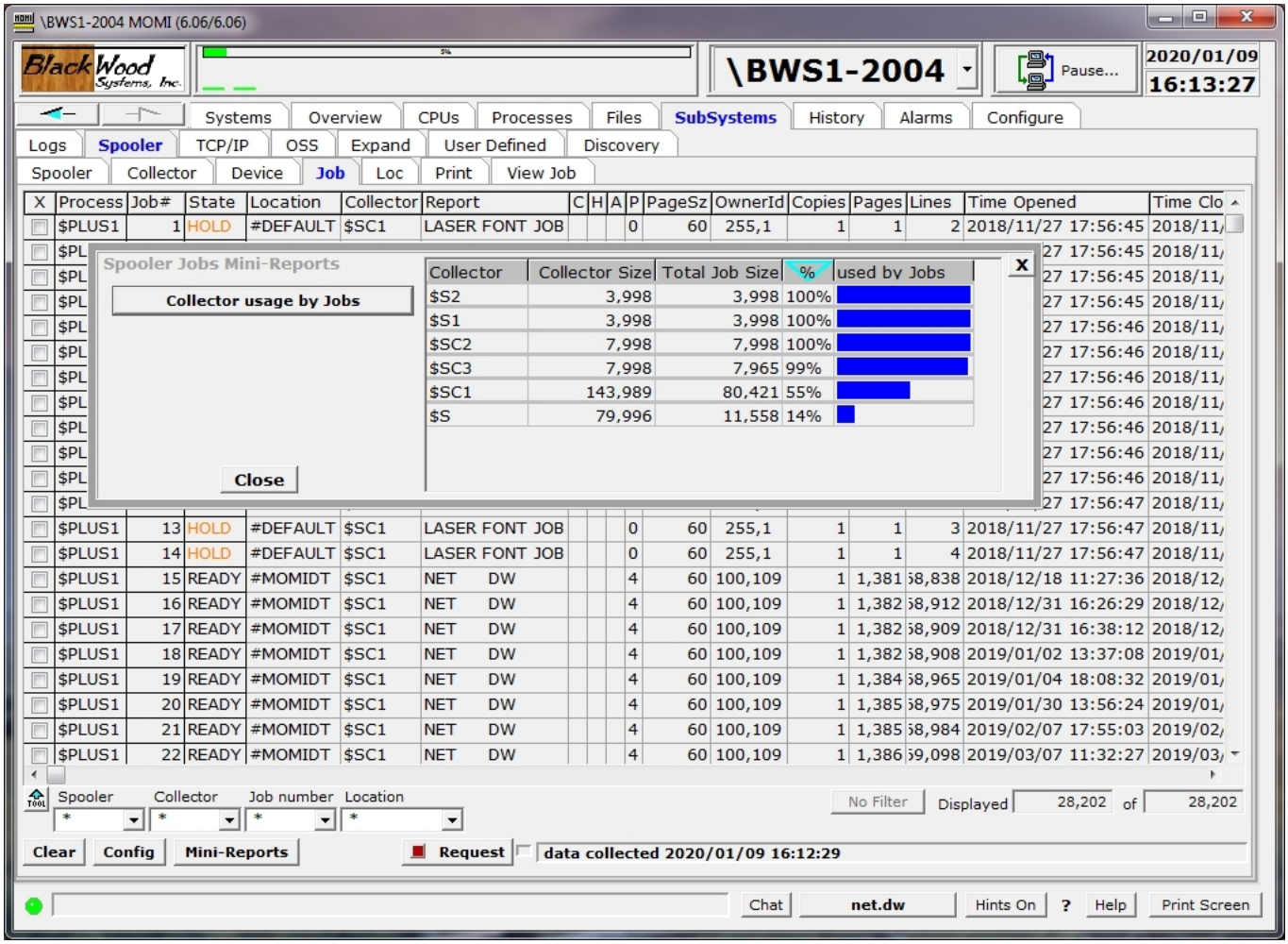
The **Search Jobs...** selection provides the function of scanning through selected jobs looking for desired text. In the first column, left-click to check one or more jobs then right-click over one of the jobs and select Search Jobs...



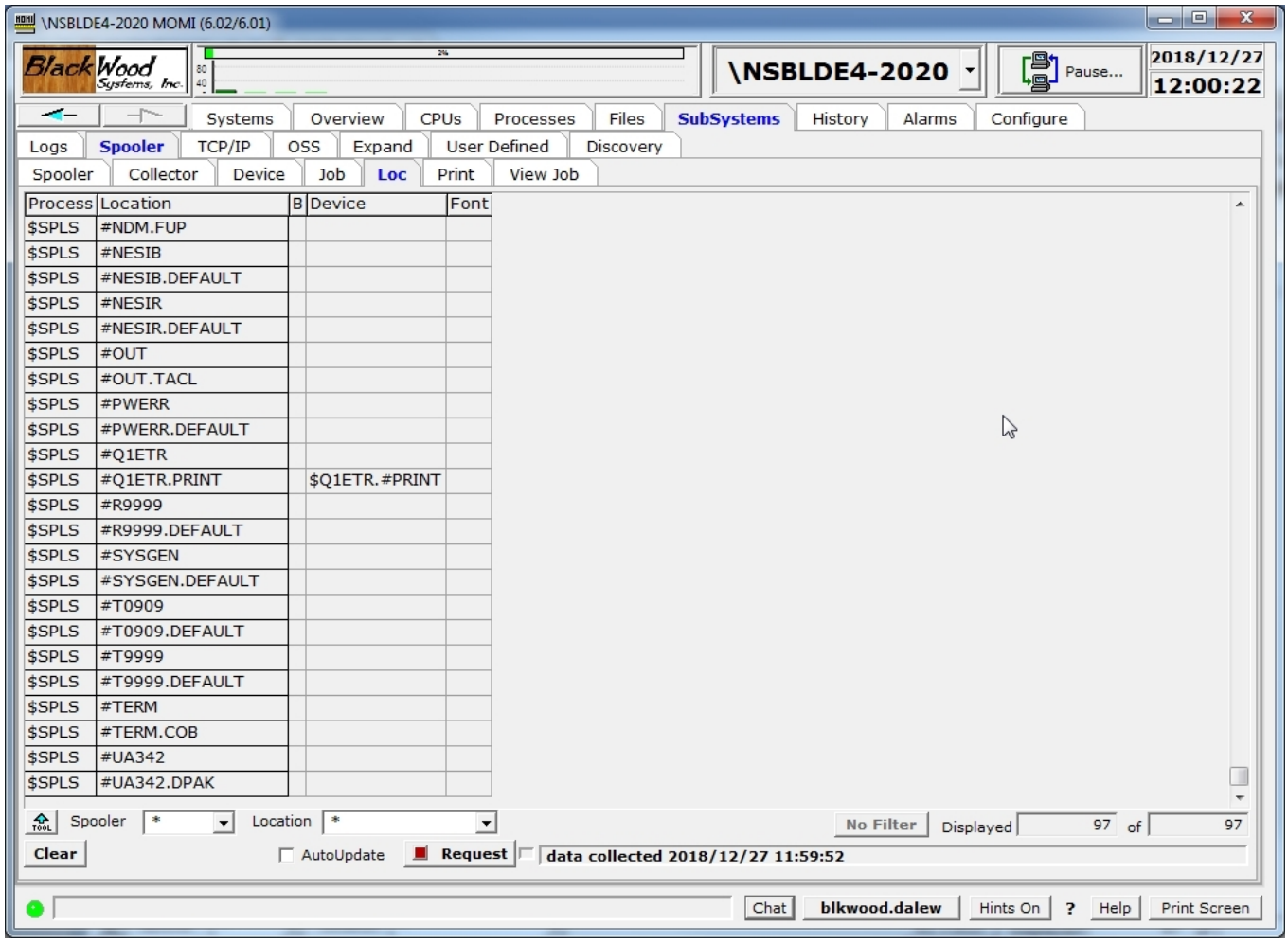
In the pop-up window, enter the text to find and enable any desired options. Press **Search Jobs** and the client will scan through all selected jobs displaying the results. Click on a line in the search results to display the job contents in View Job.



The Mini-Reports button displays a pop-up window that provides various statistics about the currently displayed jobs.


\BWS1-2004 MOMI (6.06/6.06)
BlackWood Systems, Inc.
\BWS1-2004
Pause...
2020/01/09
16:13:27
Systems
Overview
CPUs
Processes
Files
SubSystems
History
Alarms
Configure
Logs
Spooler
TCP/IP
OSS
Expand
User Defined
Discovery
Spooler
Collector
Device
Job
Loc
Print
View Job
Spooler Jobs Mini-Reports
Collector usage by Jobs
Close
Collector
Collector Size
Total Job Size
%
used by Jobs
$S2 3,998 3,998 100%
$S1 3,998 3,998 100%
$SC2 7,998 7,998 100%
$SC3 7,998 7,965 99%
$SC1 143,989 80,421 55%
$S 79,996 11,558 14%
Spooler
Collector
Job number
Location
No Filter
Displayed 28,202 of 28,202
Clear
Config
Mini-Reports
Request
data collected 2020/01/09 16:12:29
Chat
net.dw
Hints On
Help
Print Screen

Spool / Loc

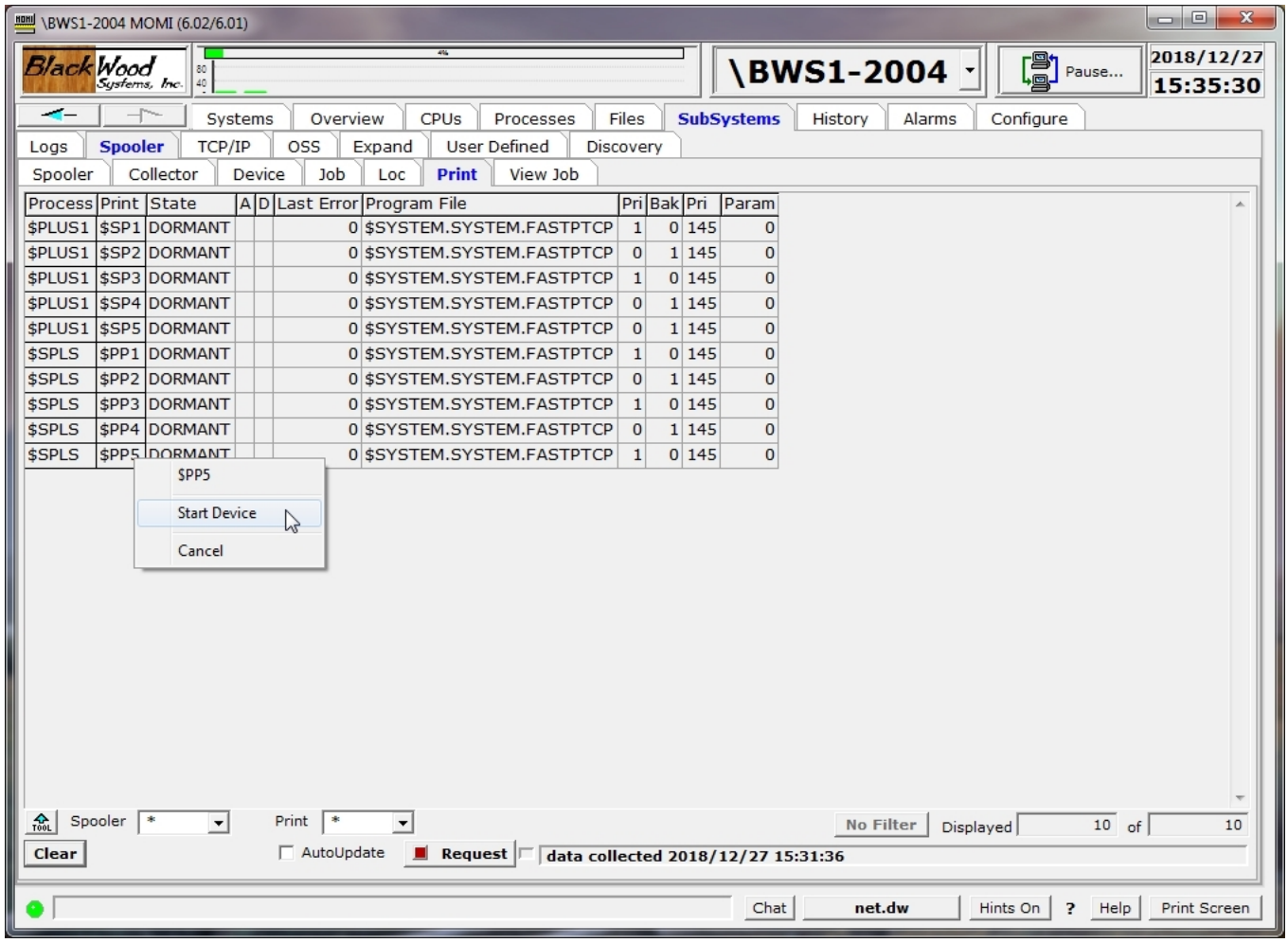


The Spool / Loc screen presents information about the locations in the Spooler.

A location defines the device to receive a job. Locations have attributes associated with them, such as font. If no Device is defined, the job waits in the Spooler.

Locations are defined as associated with a Device and optionally a Font. Locations are statically and dynamically created. Static locations are created when the Spooler is configured. Dynamic locations are created when a process opens a Spool Collector and outputs to a location not statically defined. Dynamic locations generally cause jobs to sit in the Ready state until they are either deleted or their location is changed to a Static, which in turn causes the Job to print.

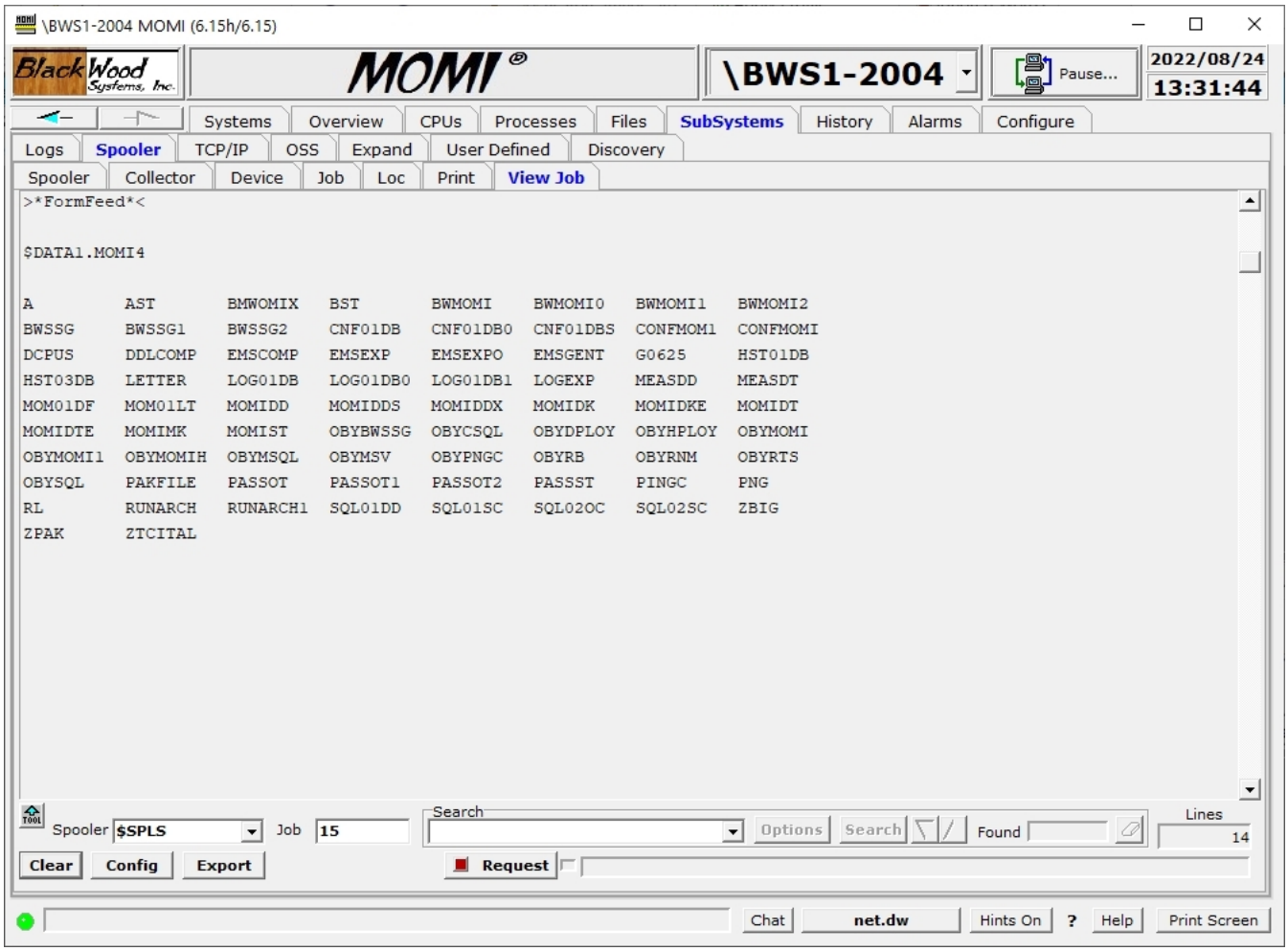
Spool / Print



The Spool / Print screen presents the status of the Spool Print processes.

A Print process handles the actual transfer of data from the Spooler to the hard-copy device or to another Spool Collector.

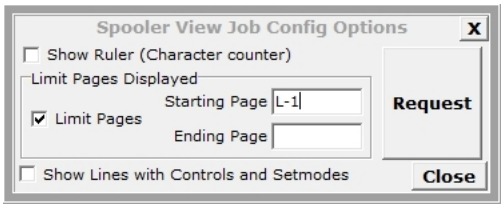
Spool / View Job



The Spool / View Job screen displays the contents of a Spooler Job.

The User must be logged on to view the contents of a job.

Pressing the Config button brings up a pop-up window. This window can be used before a job is requested to limit the number of pages downloaded. This is particularly useful for large jobs where only a few pages need to be examined. The characters F and L represent First page and Last page with an optional + / - followed by a number. In the screen below, only the last page of the job is retrieved and displayed.

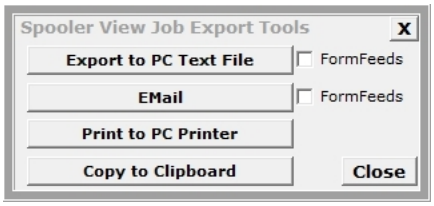


After a job is displayed, press the Tool button in the lower left hand corner bring up a window to change font size.

In the displayed text, special lines indicate various Spooler operations rather than a line of data. Below are examples:

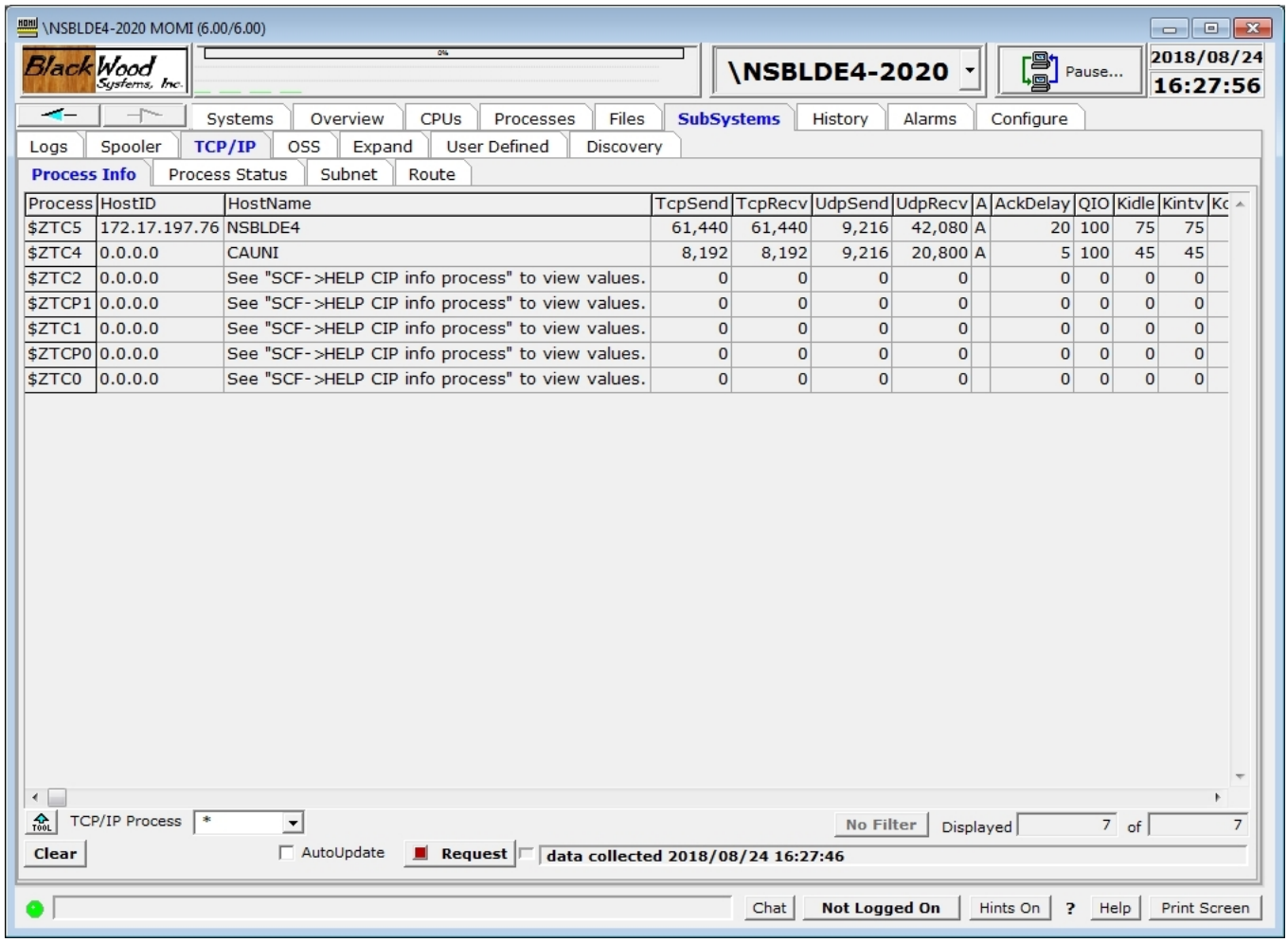
```
>*Controlbuf*<
>*Setmode*  Function #####  Param1 #####  Param2 #####<
>*Control*  Operation <?INT16>  Param <?INT16><
>*FormFeed*<
```

The Export Tools button displays a window to allow the displayed job to 1) export to a text file on the PC, 2) sent to a PC printer, or 3) loaded into an Email (MOMI invokes the default email client configured for the PC).



TCP/IP

TCP/IP / Process Info



The TCP/IP / Process Info screen presents the TCP/IP processes or protocol stacks running on the System.

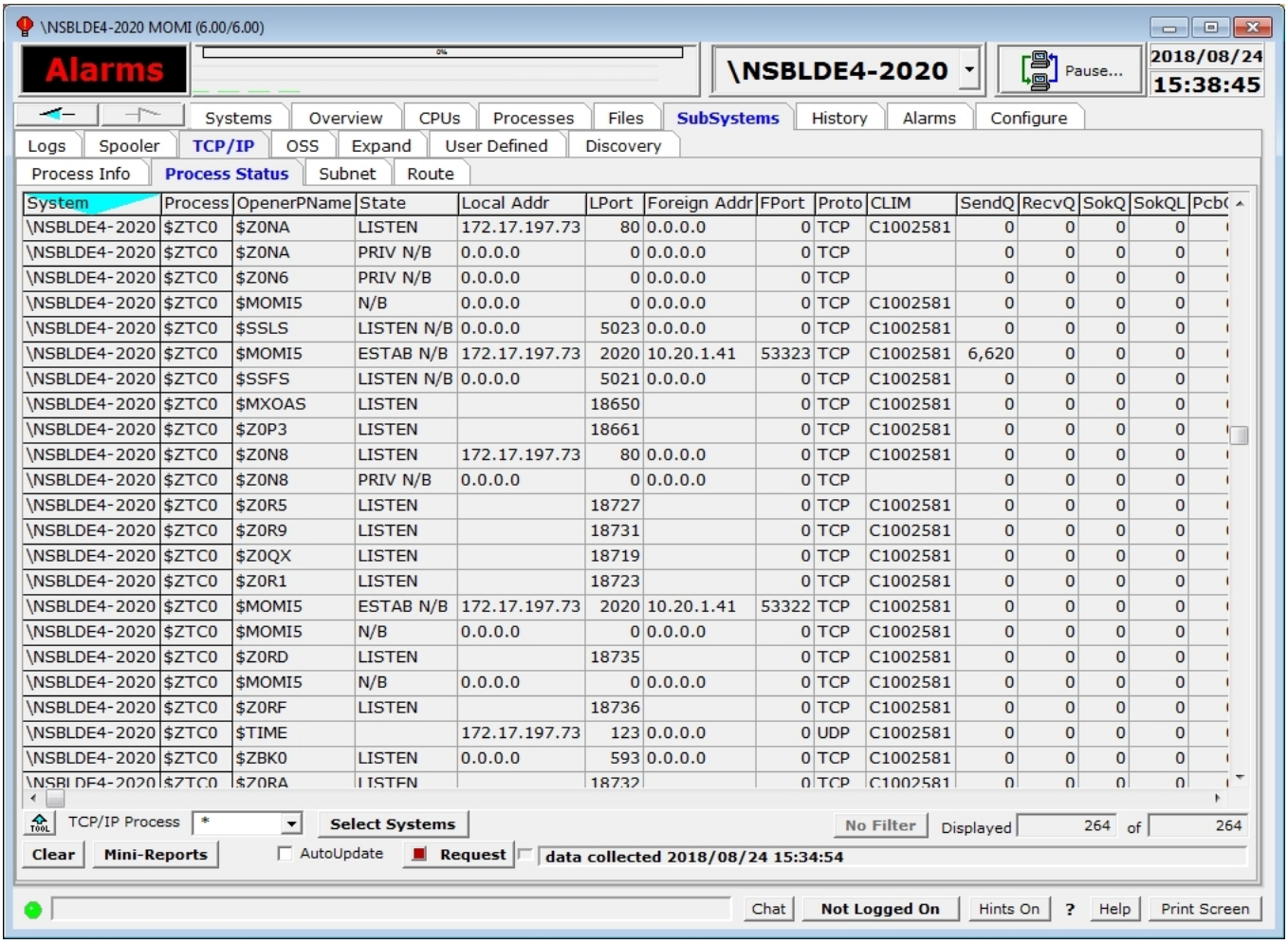
The TCP/IP process is the IOP (Input/Output Process) responsible for handling the TCP/IP traffic for a communication interface.

S-Series systems always have two TCP/IP processes used in support of the system console. These are $ZTCP0 and $ZTCP1. Most systems have a $ZTC0 TCP/IP process running, which is the default selection for socket programs.

At a TACL prompt, the default TCP/IP process is changed by using a DEFINE. Below is an example which sets the default process to $ZSAM1 and then starts an FTP:

```
1> ADD DEFINE =TCPIP^PROCESS^NAME, FILE $ZSAM1
2> TRACER www.hpe.com
```

TCP/IP / Process Status



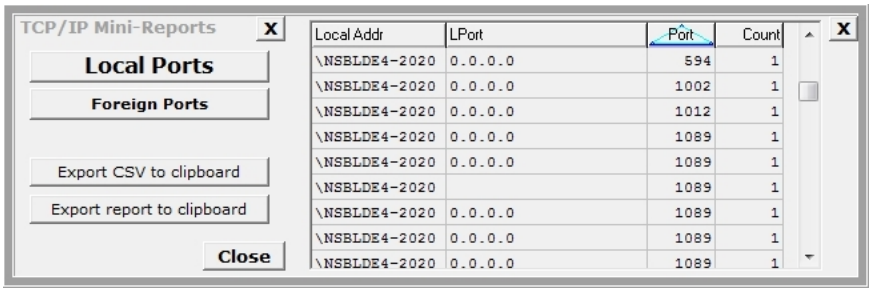
System	Process	OpenerPName	State	Local Addr	LPort	Foreign Addr	FPort	Proto	CLIM	SendQ	RecvQ	SokQ	SokQL	PcbC
\NSBLDE4-2020	$ZTC0	$Z0NA	LISTEN	172.17.197.73	80	0.0.0.0	0	TCP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$Z0NA	PRIV N/B	0.0.0.0	0	0.0.0.0	0	TCP		0	0	0	0	
\NSBLDE4-2020	$ZTC0	$Z0N6	PRIV N/B	0.0.0.0	0	0.0.0.0	0	TCP		0	0	0	0	
\NSBLDE4-2020	$ZTC0	$MOMI5	N/B	0.0.0.0	0	0.0.0.0	0	TCP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$SSLS	LISTEN N/B	0.0.0.0	5023	0.0.0.0	0	TCP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$MOMI5	ESTAB N/B	172.17.197.73	2020	10.20.1.41	53323	TCP	C1002581	6,620	0	0	0	
\NSBLDE4-2020	$ZTC0	$SSFS	LISTEN N/B	0.0.0.0	5021	0.0.0.0	0	TCP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$MXOAS	LISTEN		18650		0	TCP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$Z0P3	LISTEN		18661		0	TCP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$Z0N8	LISTEN	172.17.197.73	80	0.0.0.0	0	TCP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$Z0N8	PRIV N/B	0.0.0.0	0	0.0.0.0	0	TCP		0	0	0	0	
\NSBLDE4-2020	$ZTC0	$Z0R5	LISTEN		18727		0	TCP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$Z0R9	LISTEN		18731		0	TCP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$Z0QX	LISTEN		18719		0	TCP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$Z0R1	LISTEN		18723		0	TCP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$MOMI5	ESTAB N/B	172.17.197.73	2020	10.20.1.41	53322	TCP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$MOMI5	N/B	0.0.0.0	0	0.0.0.0	0	TCP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$Z0RD	LISTEN		18735		0	TCP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$MOMI5	N/B	0.0.0.0	0	0.0.0.0	0	TCP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$Z0RF	LISTEN		18736		0	TCP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$TIME		172.17.197.73	123	0.0.0.0	0	UDP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$ZBK0	LISTEN	0.0.0.0	593	0.0.0.0	0	TCP	C1002581	0	0	0	0	
\NSBLDE4-2020	$ZTC0	$Z0RA	LISTEN		18732		0	TCP	C1002581	0	0	0	0	

The TCP/IP / Process Status screen presents all of the active sockets. CLIM and VIO devices are displayed.

A socket is the term for a TCP/IP communications connection. The states of connection displayed include Listen (waiting for a connection), Established (where communication is taking place) or Closing (where the communications connection is terminating).

Established TCP level sockets will have an LPort (local port) that is the same as the listening port but will have a randomly chosen FPort (foreign port). The combination of the local address, local port, foreign address and foreign port make a socket unique to the TCP/IP stack.

The Mini-Reports button brings up a pop-up screen that provides quick statistics for the count of local and foreign ports based on the address.



The column State has the following meaning when the Proto is TCP (other states may be present - unknown states displayed in hexadecimal as ?%Hnnnn):

CLOSING
: waiting for a terminate connection request acknowledgment from the remote site.

CLOSE-WAIT
: waiting for a terminate connection request from the local user.

ESTAB
ESTAB N/B (N/B = Non-Blocking)
: connection is open and the user can send and receive data. This is the normal state for data transfer.

FIN-WAIT-1
: waiting for a terminate connection request from the remote TCP site or if waiting for acknowledgment of the terminate connection request that the process has sent previously.

FIN-WAIT-2
: waiting for a termination of data to be received after having sent a FIN (termination of data being sent).

LISTEN
LISTEN N/B (N/B = Non-Blocking)
: waiting for a connection request from any remote TCP site.

LAST-ACK
: waiting for acknowledgment of the terminate connection request previously sent to the remote site (which includes an acknowledgment of its terminate connection request).

N/B
: a Non-Blocking socket.

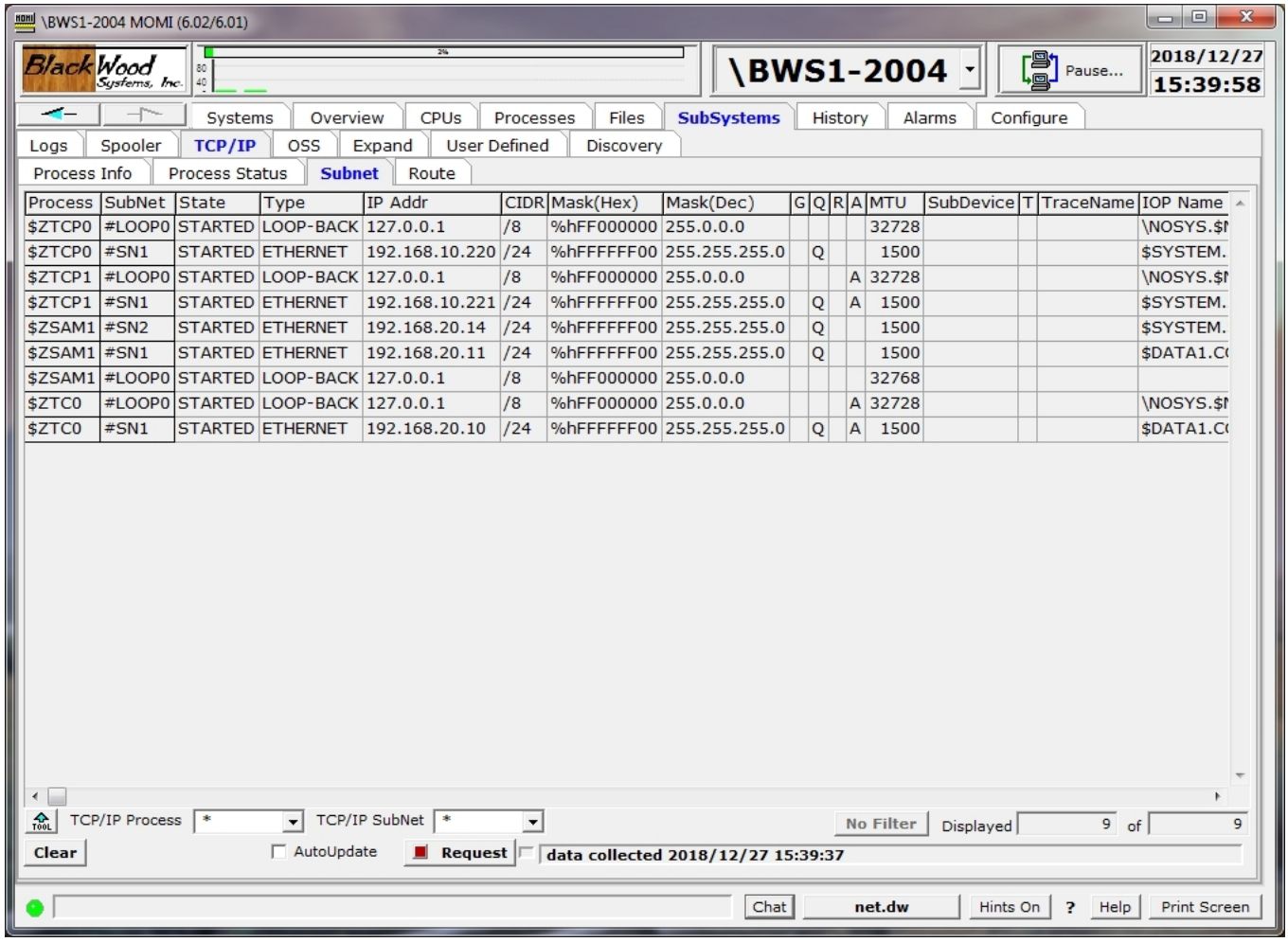
PRIV N/B
: a Privileged Non-Blocking socket.

SYN-RCVD
: waiting for an acknowledgment of a SYN-ACK sent in response to a SYN.

SYN-SENT
: waiting for a SYN-ACK after having sent a SYN.

TIME-WAIT
: waiting for sufficient time to pass (about two round trips) to be sure that stray packets are flushed from the network.

TCP/IP / Subnet



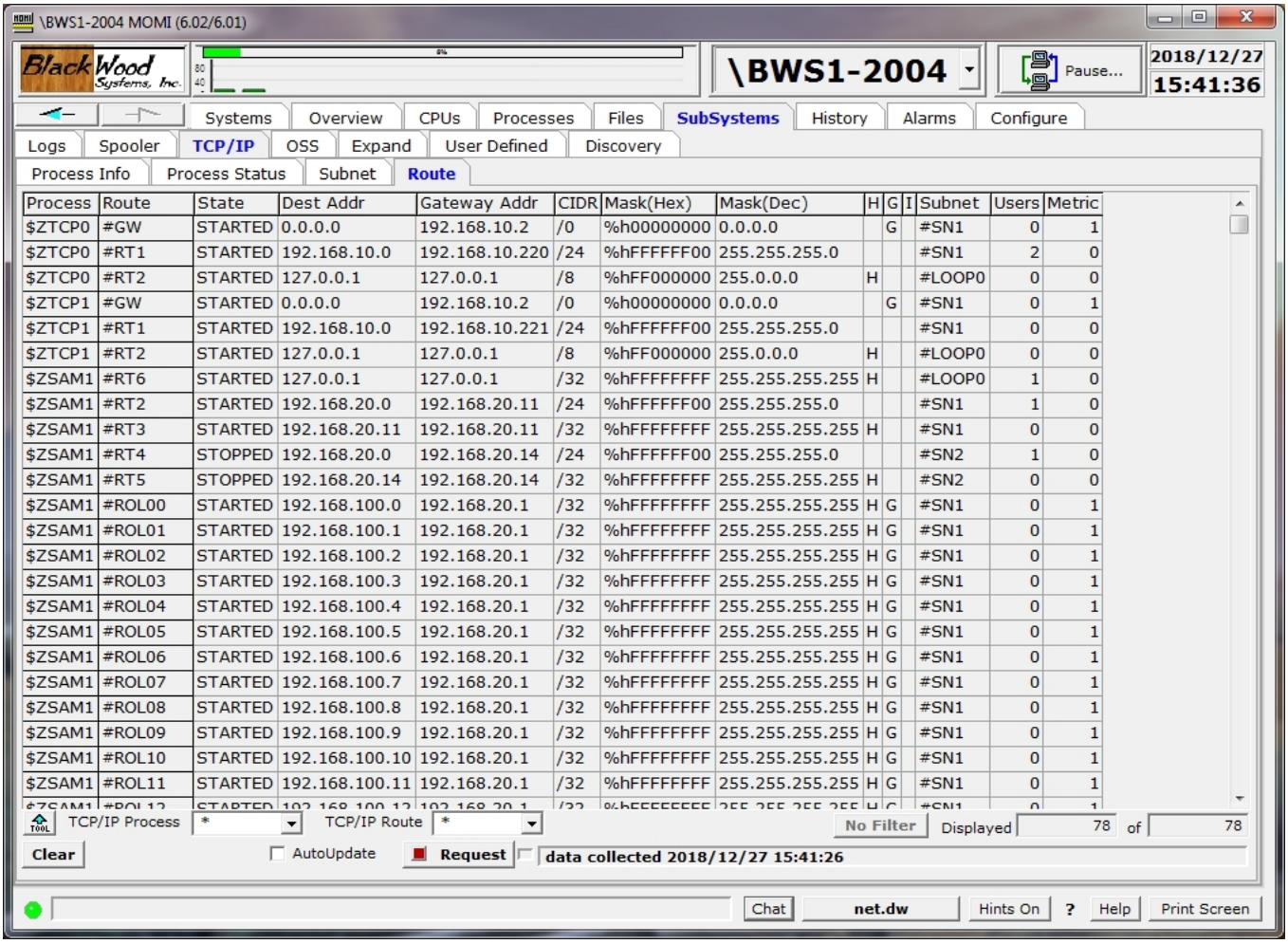
The TCP/IP / Subnet screen presents the status of the connection between the protocol stack and the physical LAN interfaces.

A Subnet is associated with an address and 1) an IOP (Input/Output Process) on older platforms, or 2) a LIF (Logical InterFace) on newer platforms.

The Subnet Mask determines when routing of outgoing messages requires the usage of a default gateway.

The TCP/IP address of 127.0.0.1 is a built-in address used for loop-back testing and is an address that does not go anywhere but is still valid.

TCP/IP / Route

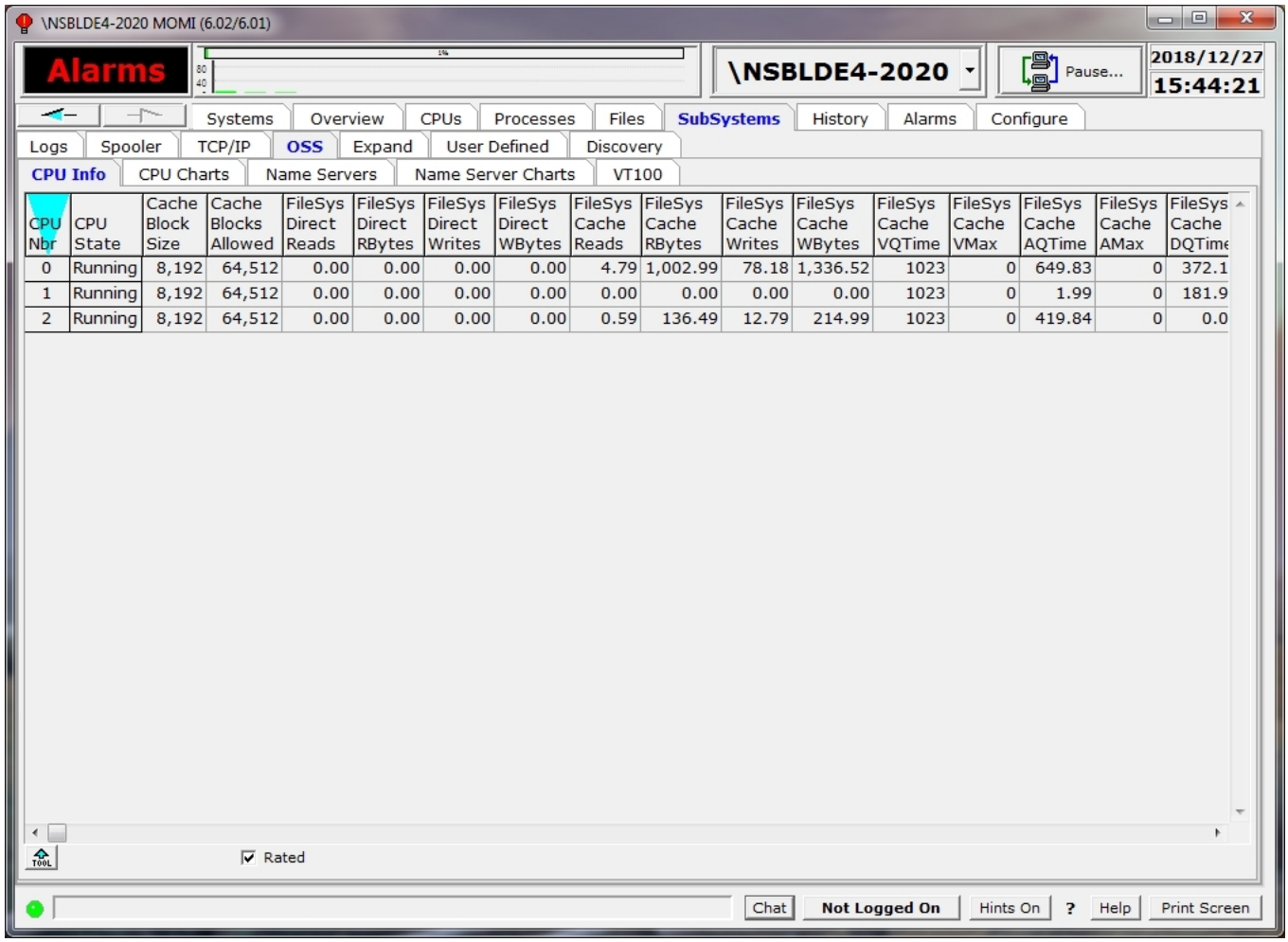


The TCP/IP / Route screen presents the paths used in determining where data is sent if the destination is not on the local Subnet.

TCP/IP routing generally comes into play if the destination is not local. Data is sent to a default gateway (usually a router) so that it may be forwarded either to the destination or to another router in order to reach the final destination.

OSS

OSS / CPU Info

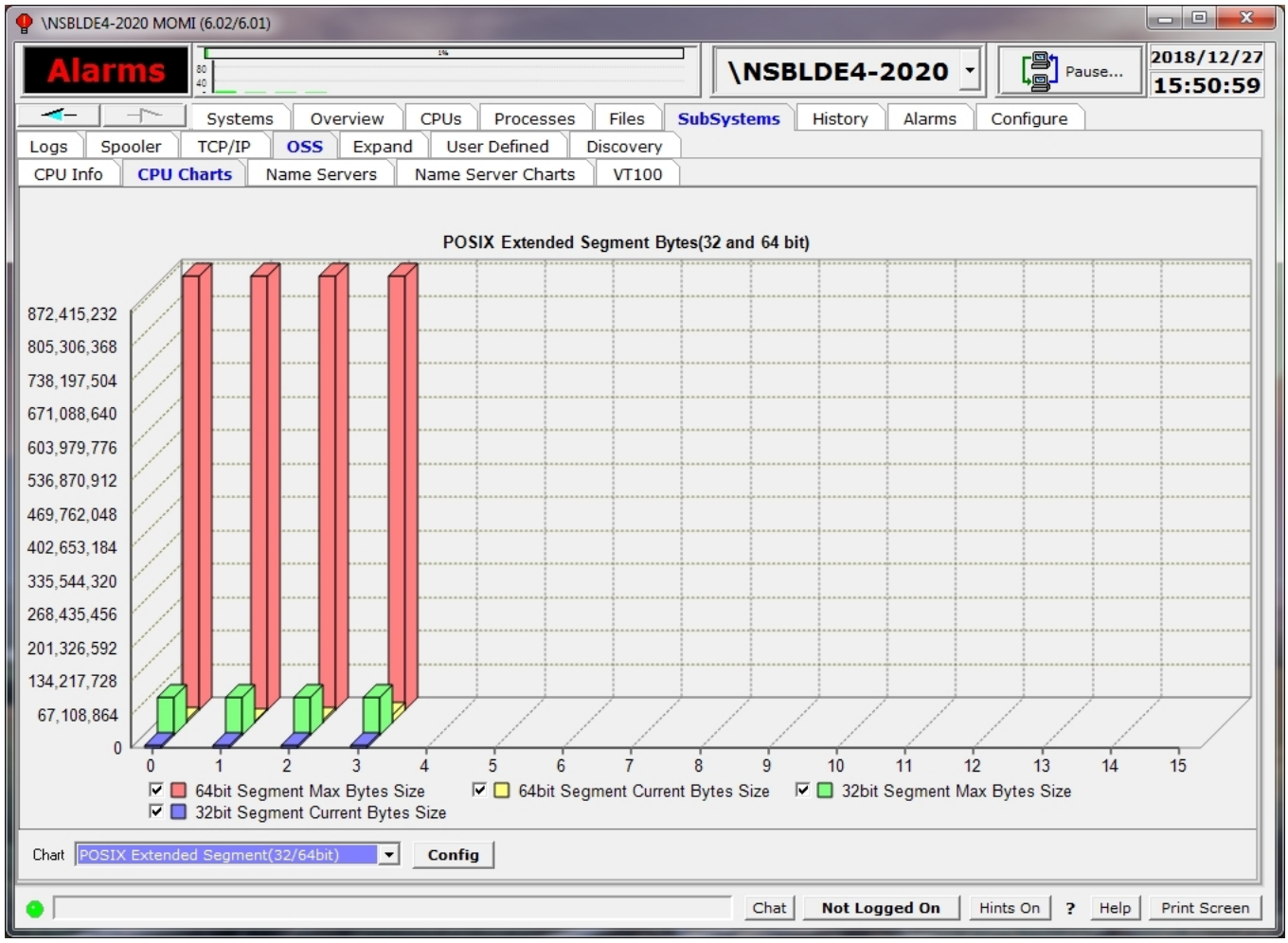


CPU Nbr	CPU State	Cache Block Size	Cache Blocks Allowed	FileSys Direct Reads	FileSys Direct RBytes	FileSys Direct Writes	FileSys Direct WBytes	FileSys Cache Reads	FileSys Cache RBytes	FileSys Cache Writes	FileSys Cache WBytes	FileSys Cache VQTime	FileSys Cache VMax	FileSys Cache AQTime	FileSys Cache AMax	FileSys Cache DQTime
0	Running	8,192	64,512	0.00	0.00	0.00	0.00	4.79	1,002.99	78.18	1,336.52	1023	0	649.83	0	372.1
1	Running	8,192	64,512	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1023	0	1.99	0	181.9
2	Running	8,192	64,512	0.00	0.00	0.00	0.00	0.59	136.49	12.79	214.99	1023	0	419.84	0	0.0

The OSS / CPU Info screen presents the MEASURE OSS CPU entity.

The MEASURE OSS CPU entity provides insight into the various counters concerning OSS at the CPU level. This primarily deals with file cache blocks, local CPU-to-CPU I/O activity and remote CPU-to-CPU I/O activity.

OSS / CPU Charts

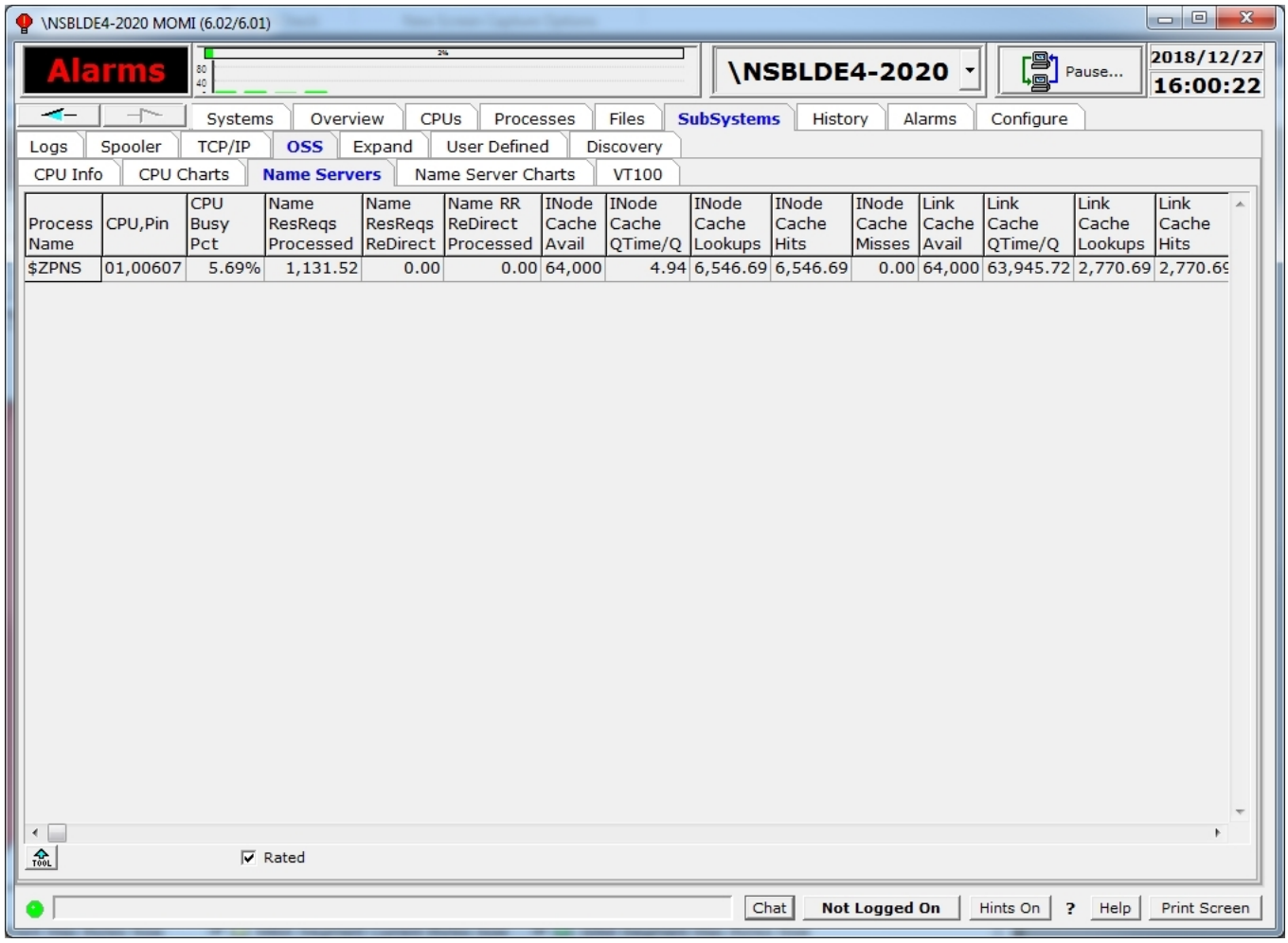


The OSS / CPU Charts screen presents selected data from the OSS / CPU Info screen in graphical form. The information displayed is obtained from the MEASURE OSS CPU entity.

Select the desired data in the drop-down box to display the graph. The Config button allows the definition of predefined buttons on the screen for fast access to graphs.

The MEASURE OSS CPU entity provides insight into the various counters concerning OSS at the CPU level. This primarily deals with file cache blocks, local CPU-to-CPU I/O activity and remote CPU-to-CPU I/O activity.

OSS / Name Servers

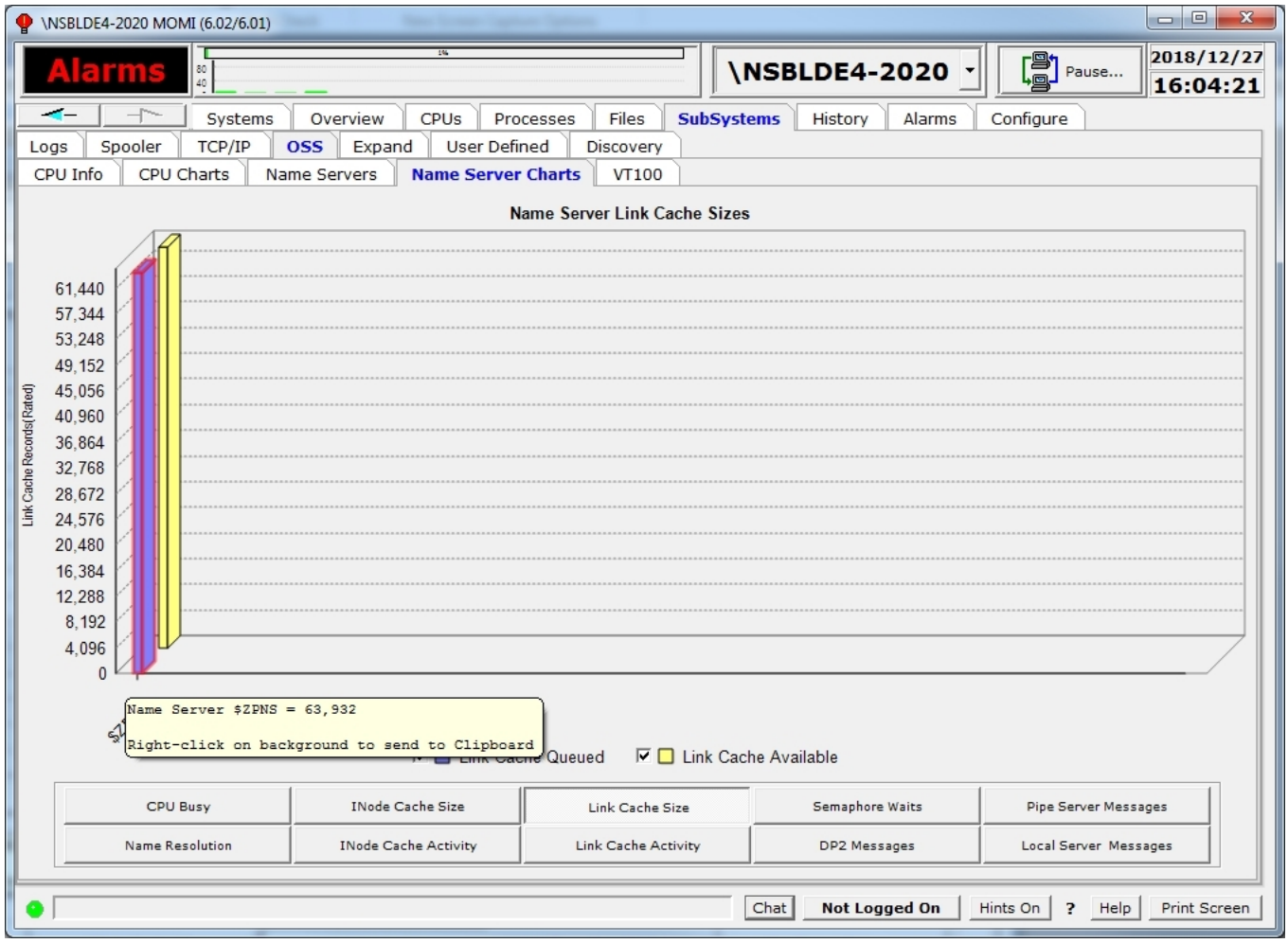


The OSS / Name Server Charts screen presents the MEASURE OSS Name Server entity.

The OSS Name Server is responsible for translating OSS path names into Guardian $volume.subvol.file type names and vice versa. The Name Server also provides the isolation of physical disk drives from the unified path names. Path names do not generally represent physical volumes except when representing Guardian format file names.

A system running OSS has at least one Name Server running, but usually one Name Server per CPU.

OSS / Name Server Charts



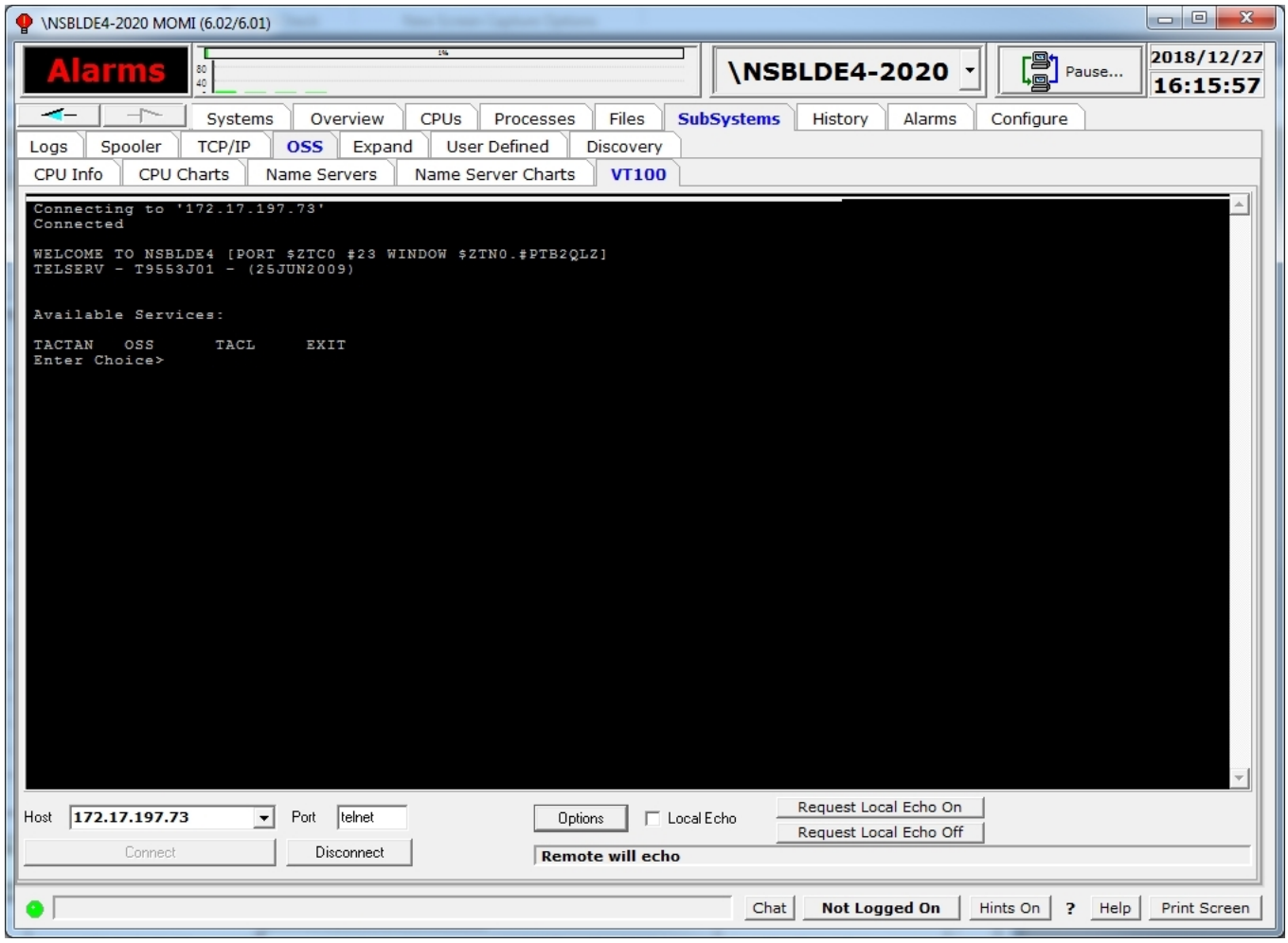
The OSS / Name Server Charts screen presents selected data from the Name Server screen in graphical form. The information displayed is obtained from the MEASURE OSS Name Server entity.

Press the desired button to display the graph.

The OSS Name Server is responsible for translating OSS path names into Guardian $volume.subvol.file type names and vice versa. The Name Server also provides the isolation of physical disk drives from the unified path names. Path names do not generally represent physical volumes except when representing Guardian format file names.

A system running OSS has at least one Name Server running, but usually one Name Server per CPU.

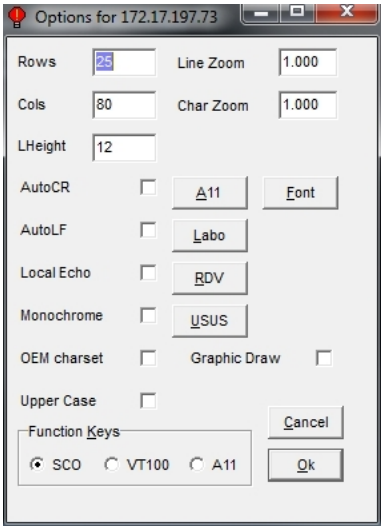
OSS / VT100



The OSS / VT100 screen provides a simple emulator useful in the OSS environment.

Enter the Host TCP/IP address (or Name) and press Connect.

Various options can be set by pressing the Options button and these are saved in the MOMI.INI file. Buttons in the center of the pop-up provide commonly used defaults. Selecting the VT100 Function Keys are recommended for OSS.


Options for 172.17.197.73
Rows
25
Line Zoom
1.000
Cols
80
Char Zoom
1.000
LHeight
12
AutoCR
A11
Font
AutoLF
Labo
Local Echo
RDV
Monochrome
USUS
OEM charset
Graphic Draw
Upper Case
Cancel
Function Keys
SCO
VT100
A11
Ok

Expand

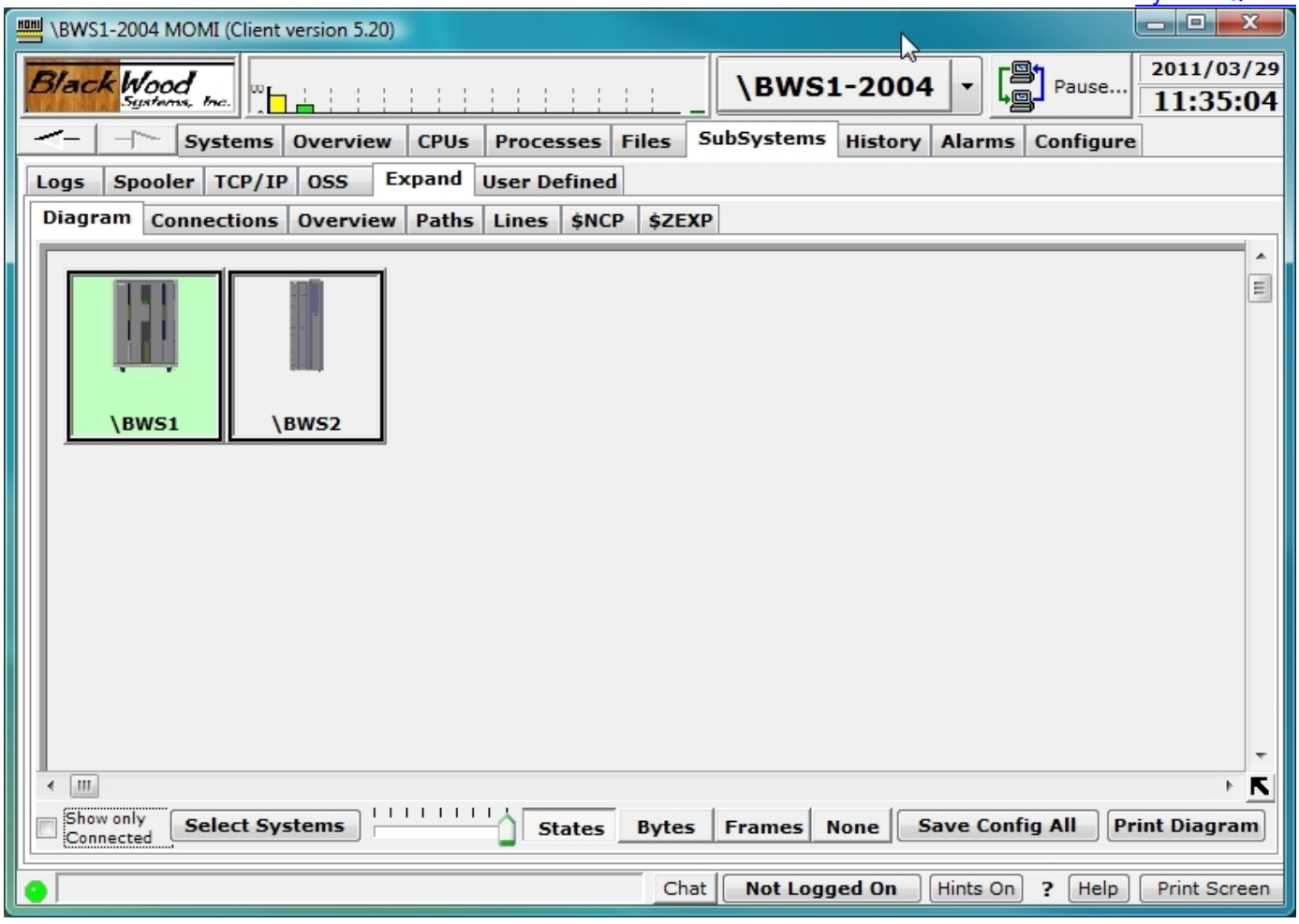
Overview

The Expand subsystem provides information on the status of the communication interconnects linking systems within a network.

All screens under this tab are automatically updated either by timer operation or by events as seen by MOMI. Certain fields such as a time stamp in the stats output are only updated when other values within the stats record change. In other words, stats records are not updated just to update the time stamp, but are updated when 'real' stats values change.

Expand / Diagram

see System Quirks

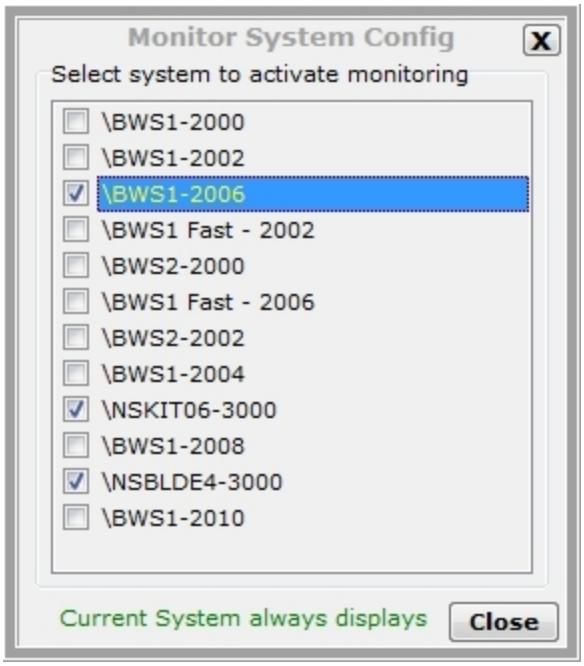


The Expand / Diagram screen provides a pictorial view of the Expand network.

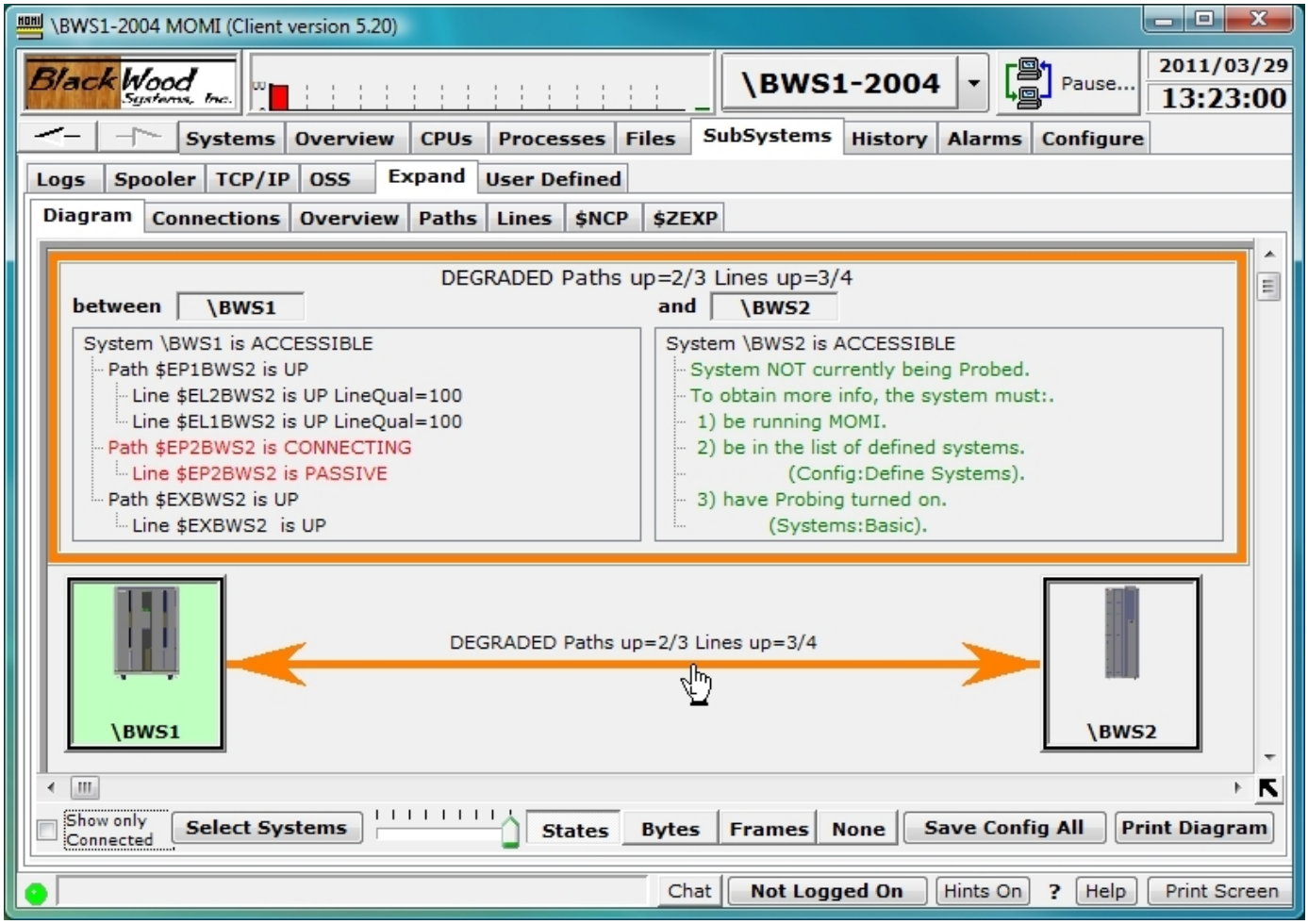
Expand is the name of the subsystem and system software that provides transparent communication between two or more NonStop Systems.

When first started, the nodes or systems on the network are placed right next to each other. Place the mouse within the system box in the 'unused' area and press and hold the mouse to drag the Systems to desired locations. Press the **Save Config All** button to store the screen layout (the locations are not automatically saved).

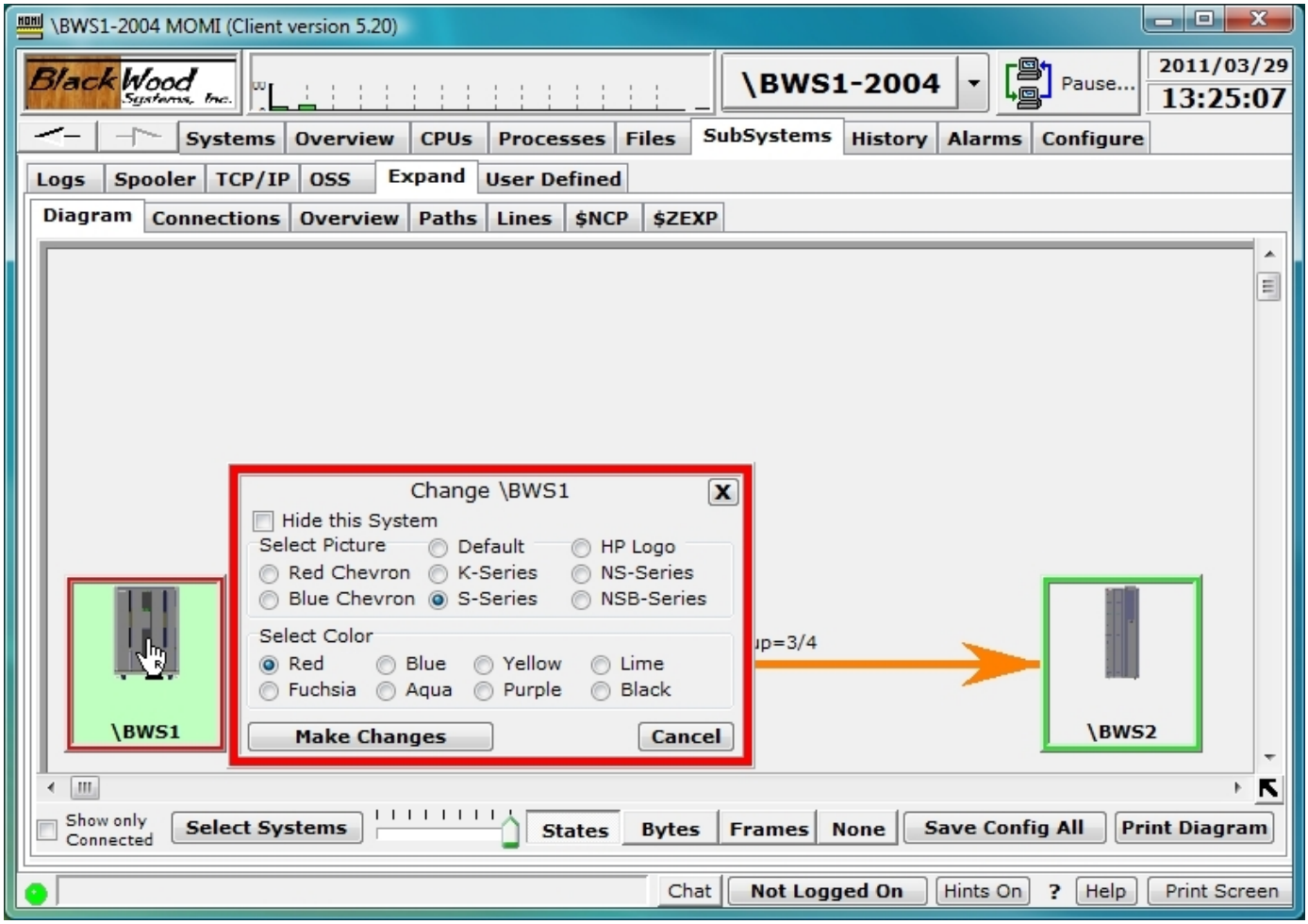
The systems displayed are selected by pressing the button **Select Systems** and choosing systems from the displayed list. The systems available are defined on the screen Define / Systems. Note that the current system selected for the overall client will always display.



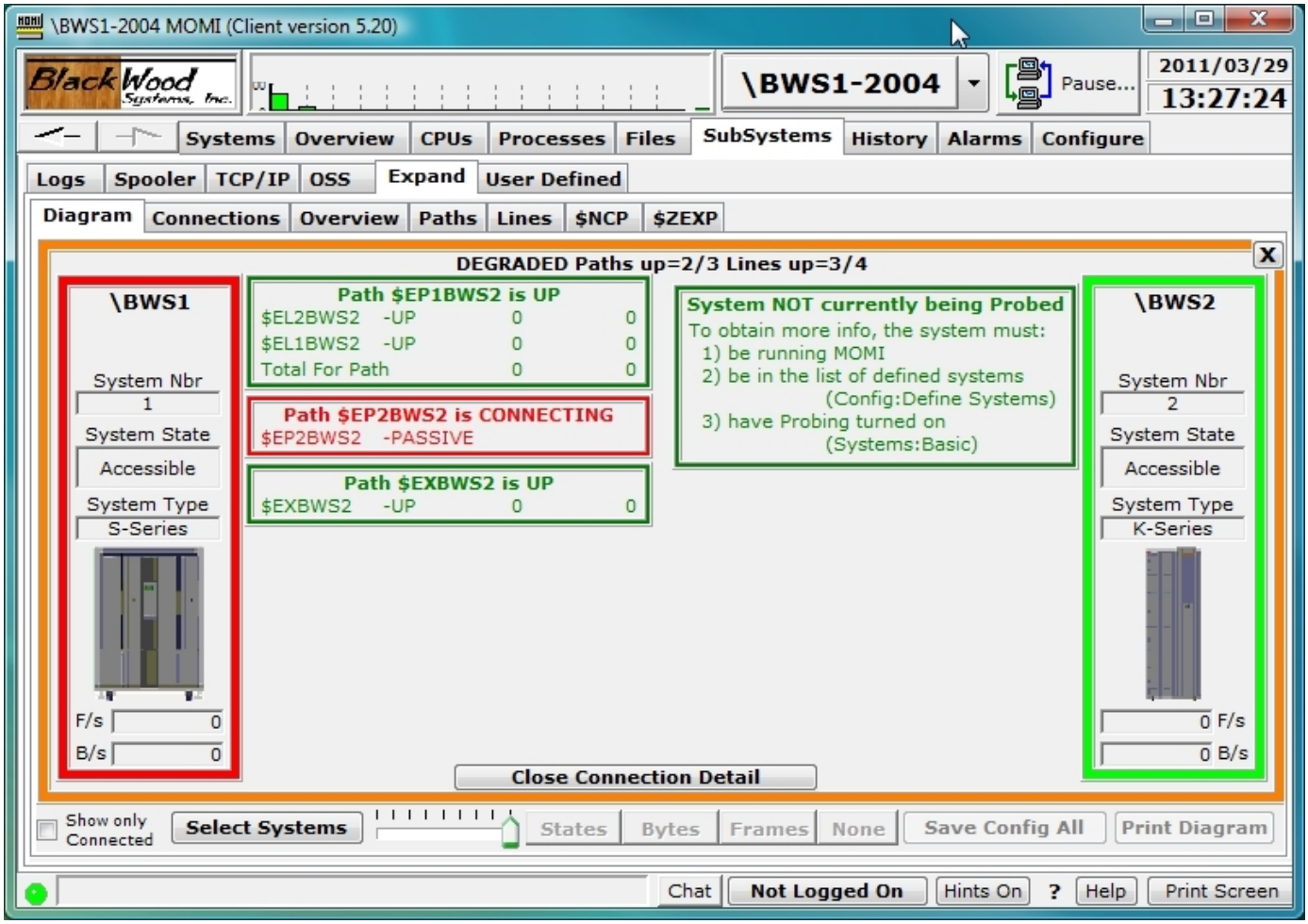
Place the mouse over a connection line displays additional detail about the underlying paths and lines.



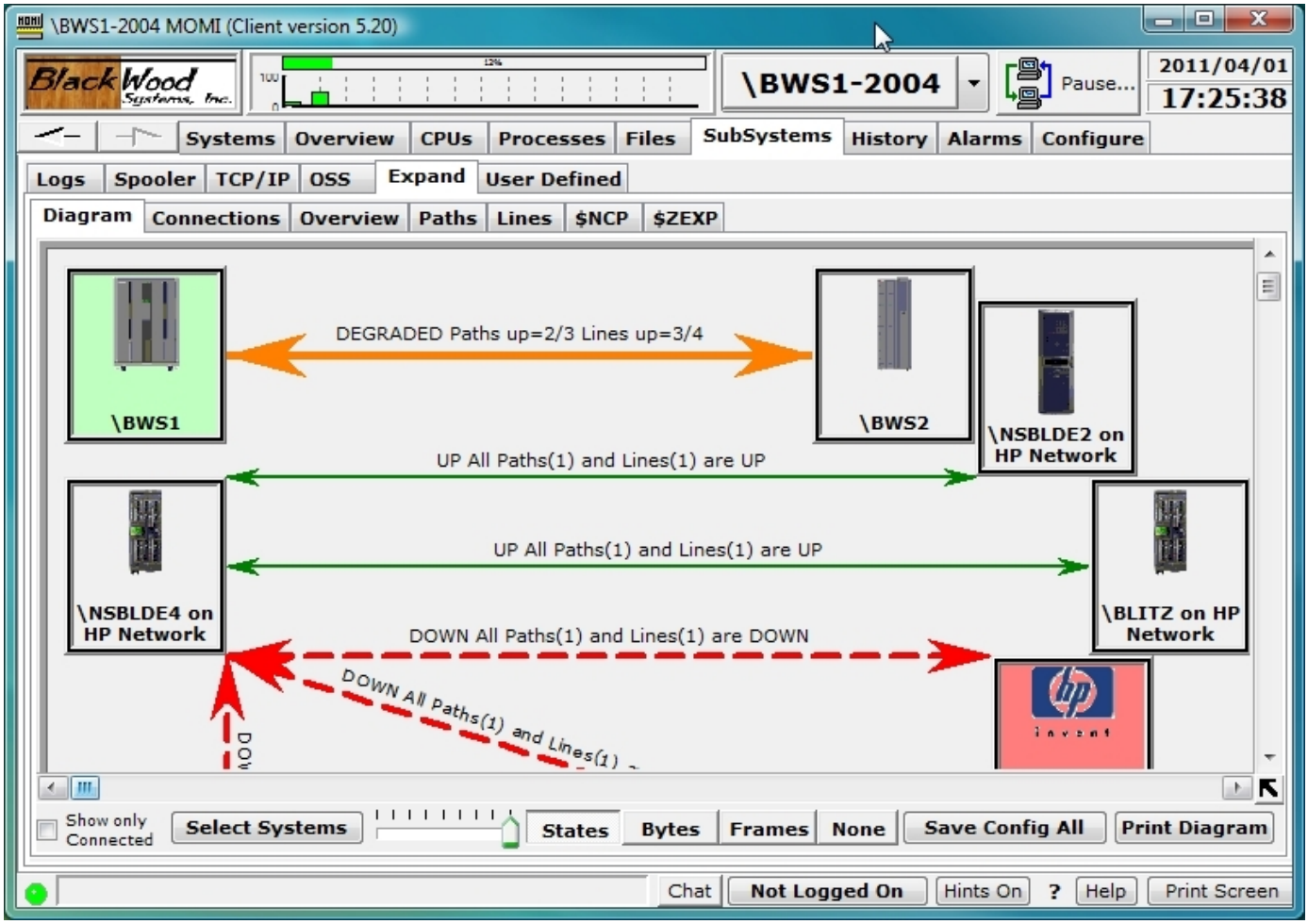
The boxes representing each System on the network may be configured by right-clicking over the image in the center of the box. Various visual attributes such as the system image and color of the box frame may be changed. The option **Hide this System** is used to prevent display on this screen of undesired Systems in the network. Systems are restored on the Expand / Connections screen.



Place the mouse over a connection between two Systems and left-click brings up a detailed diagram of the connections, states, frame rate and byte rate (if available) between the two nodes.

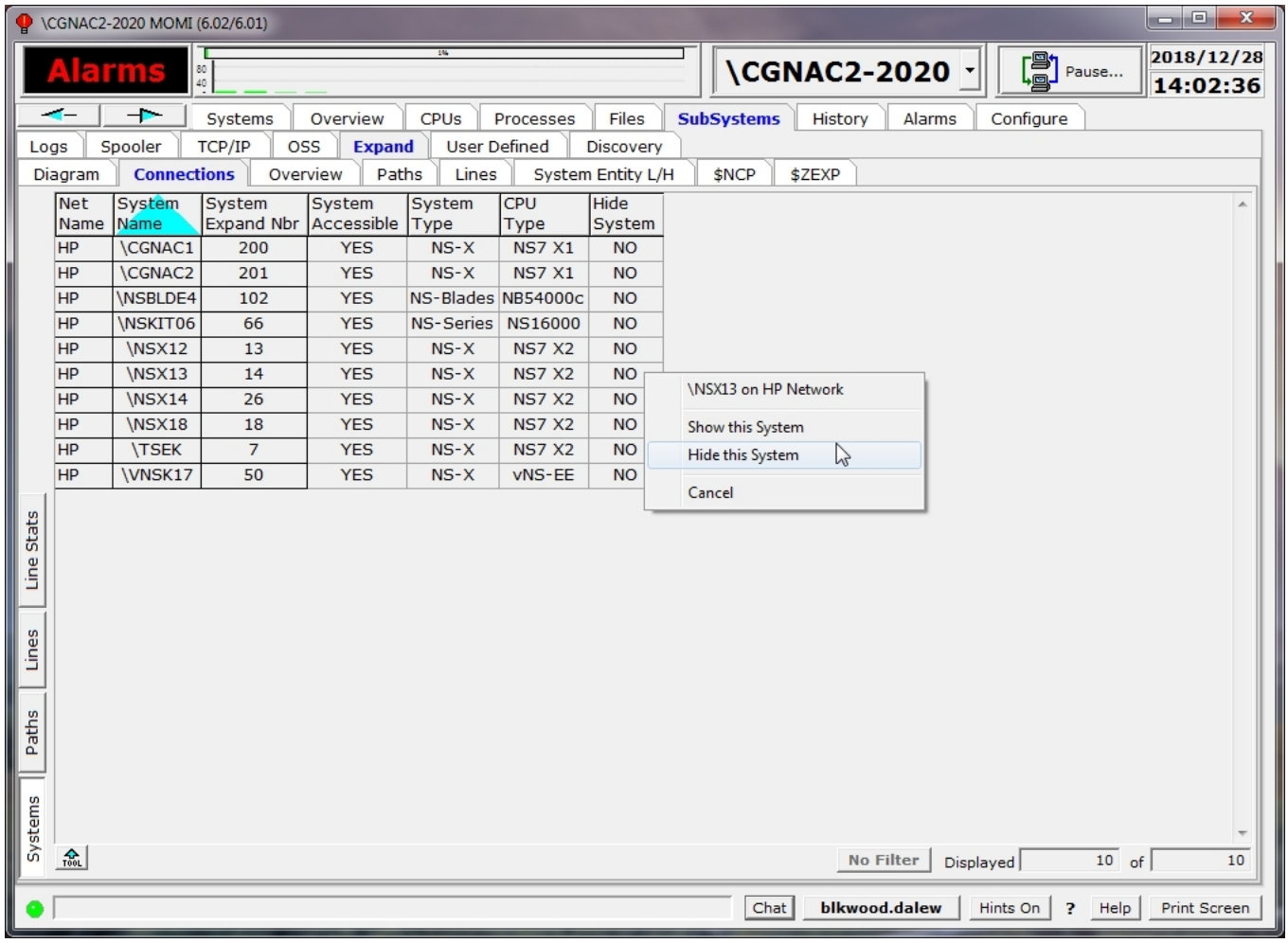


If a MOMI client connects to multiple systems, and the multiple systems are interconnected using different Expand networks, it is possible group them separately by using the CONFMOMI keyword EXPAND-NETWORK-NAME. Below is a diagram showing two networks. The first network comprised of \BWS1 and \BWS2 is unnamed. The second network is called ITUG2005. Notice the network name appears in the box with each system.



To obtain the maximum amount of detail, MOMI must be installed on each node.

Expand / Connections

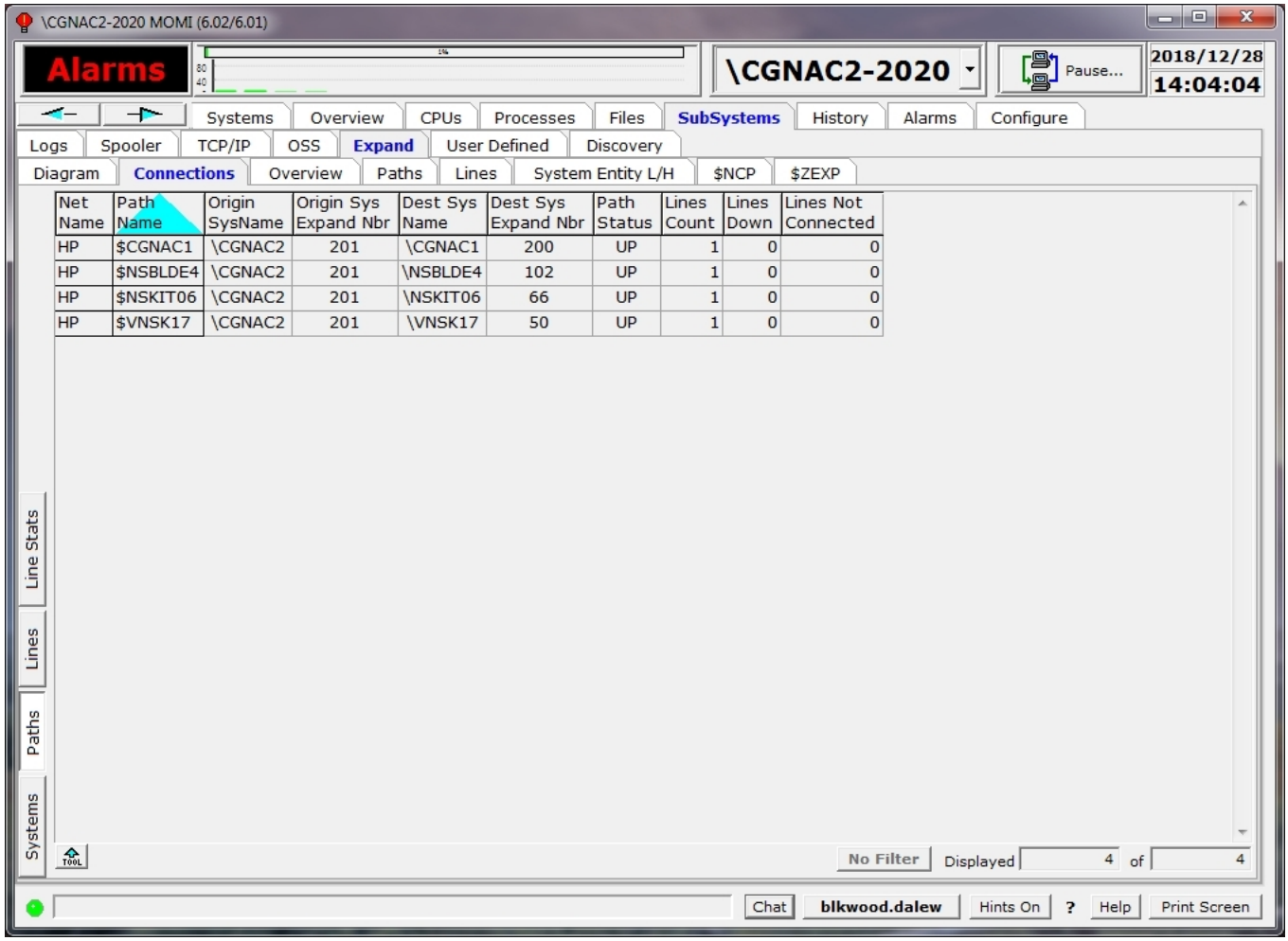


The Expand / Connections screen displays the details, or internal values, used to construct the Expand / Diagram screen. The values present are only a subset of information available on other screens and may change from time-to-time.

Tabs along the left hand side of the screen select different categories of information. The **Systems** tab displays all of the known Expand nodes in the network.

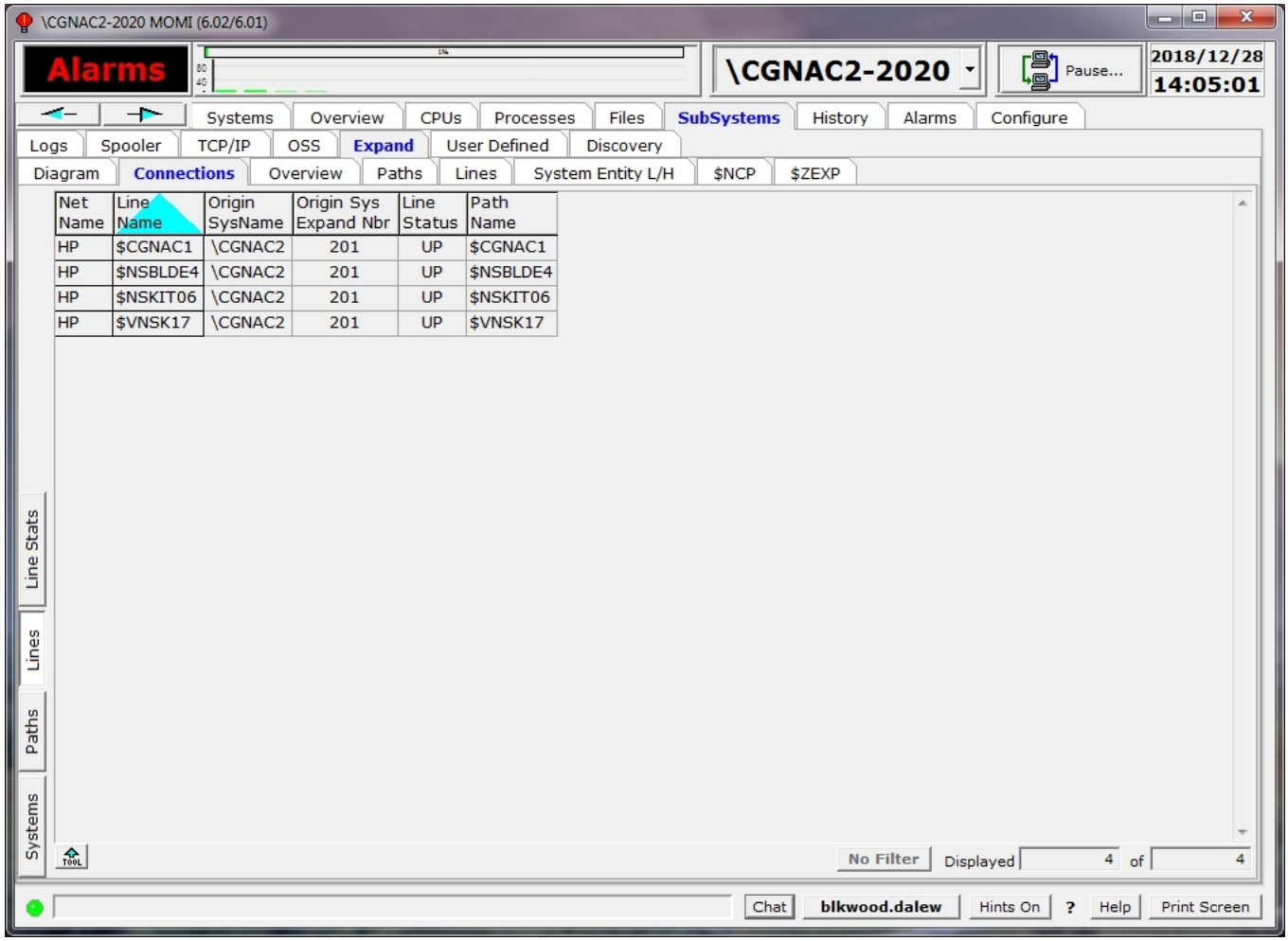
Systems may be hidden from view on the Expand / Diagram page by a right click over the value in the **Hide Systems** column.

The **Paths** tab displays information about each known path.

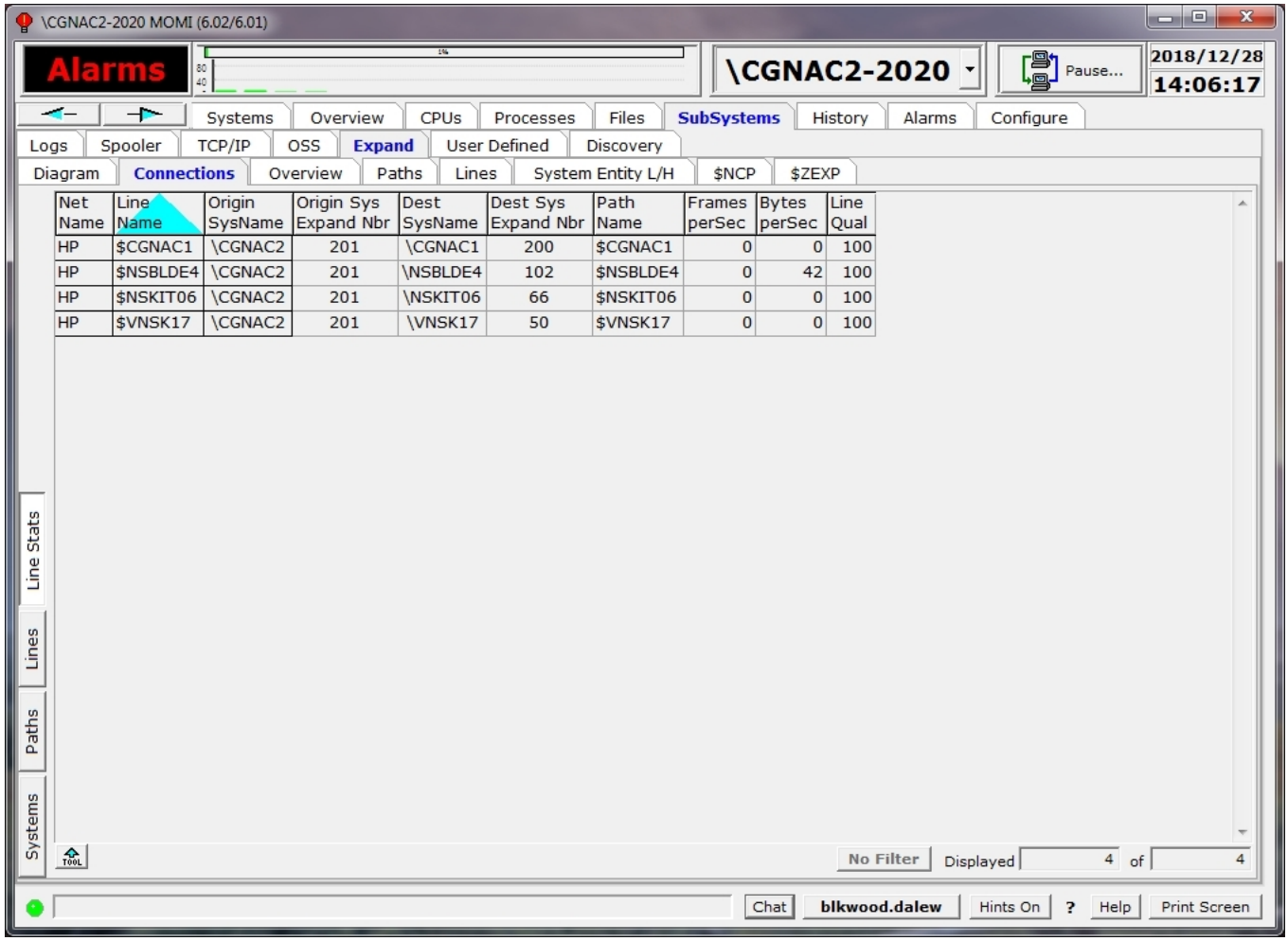


Net Name	Path Name	Origin SysName	Origin Sys Expand Nbr	Dest Sys Name	Dest Sys Expand Nbr	Path Status	Lines Count	Lines Down	Lines Not Connected
HP	$CGNAC1	\CGNAC2	201	\CGNAC1	200	UP	1	0	0
HP	$NSBLDE4	\CGNAC2	201	\NSBLDE4	102	UP	1	0	0
HP	$NSKIT06	\CGNAC2	201	\NSKIT06	66	UP	1	0	0
HP	$VNSK17	\CGNAC2	201	\VNSK17	50	UP	1	0	0

The **Lines** tab displays information about each know Line.



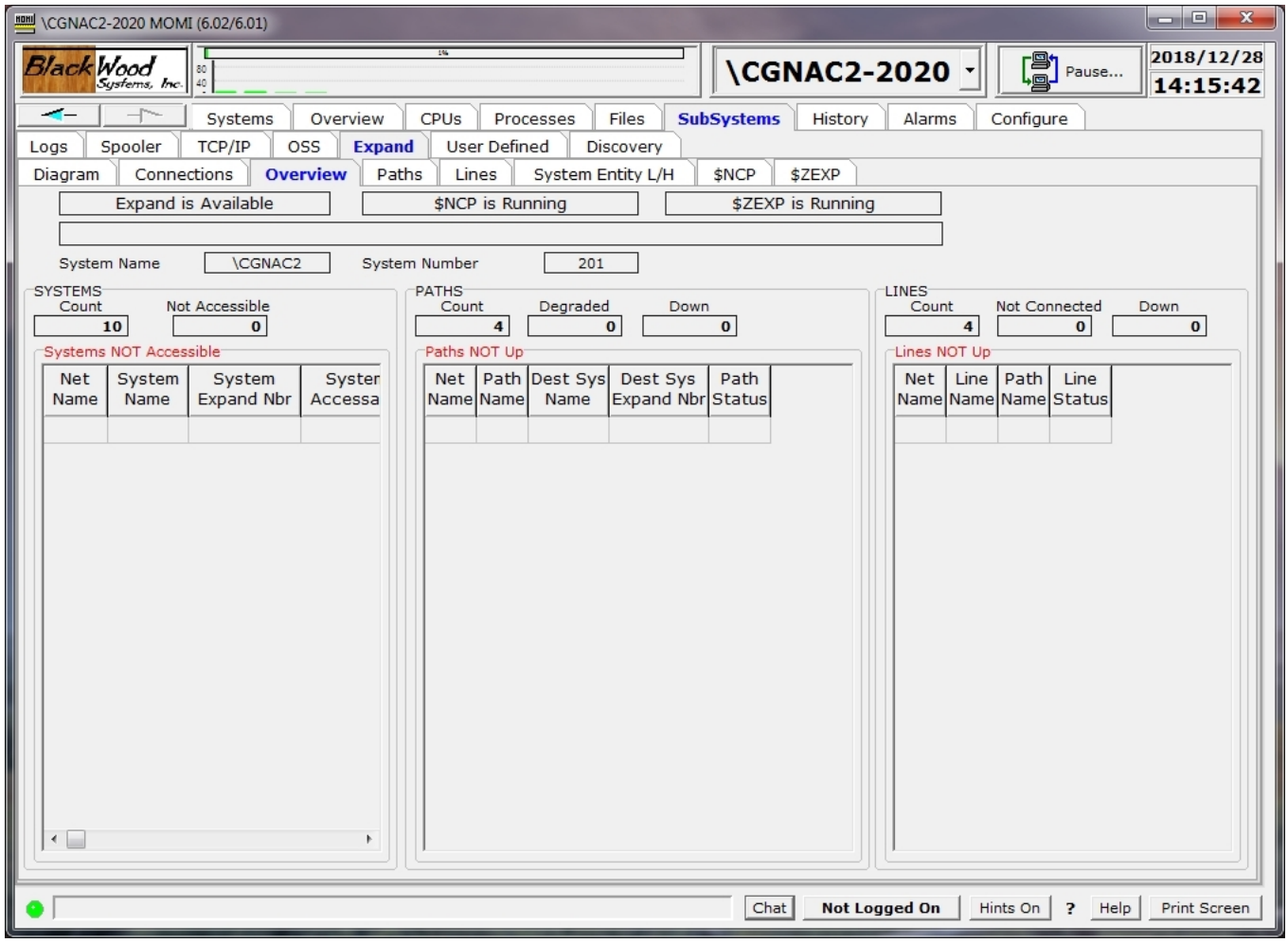
The **Line Stats** tab displays information about each known line. Frames and/or Bytes per second values are not available on all Expand line types.



This screen may show systems no longer connected to the Expand network. Formerly present systems continue to be reported by Expand until they are explicitly removed from network routing. Below are some useful SCF commands in this area:

info process $NCP,systems	list all known systems
delete entry $NCP.*	remove from network routing any system not directly connected
delete entry $NCP.\sys	remove from network routing a specific system (\sys)
info process $NCP,pathsets	displays pathmap and line information

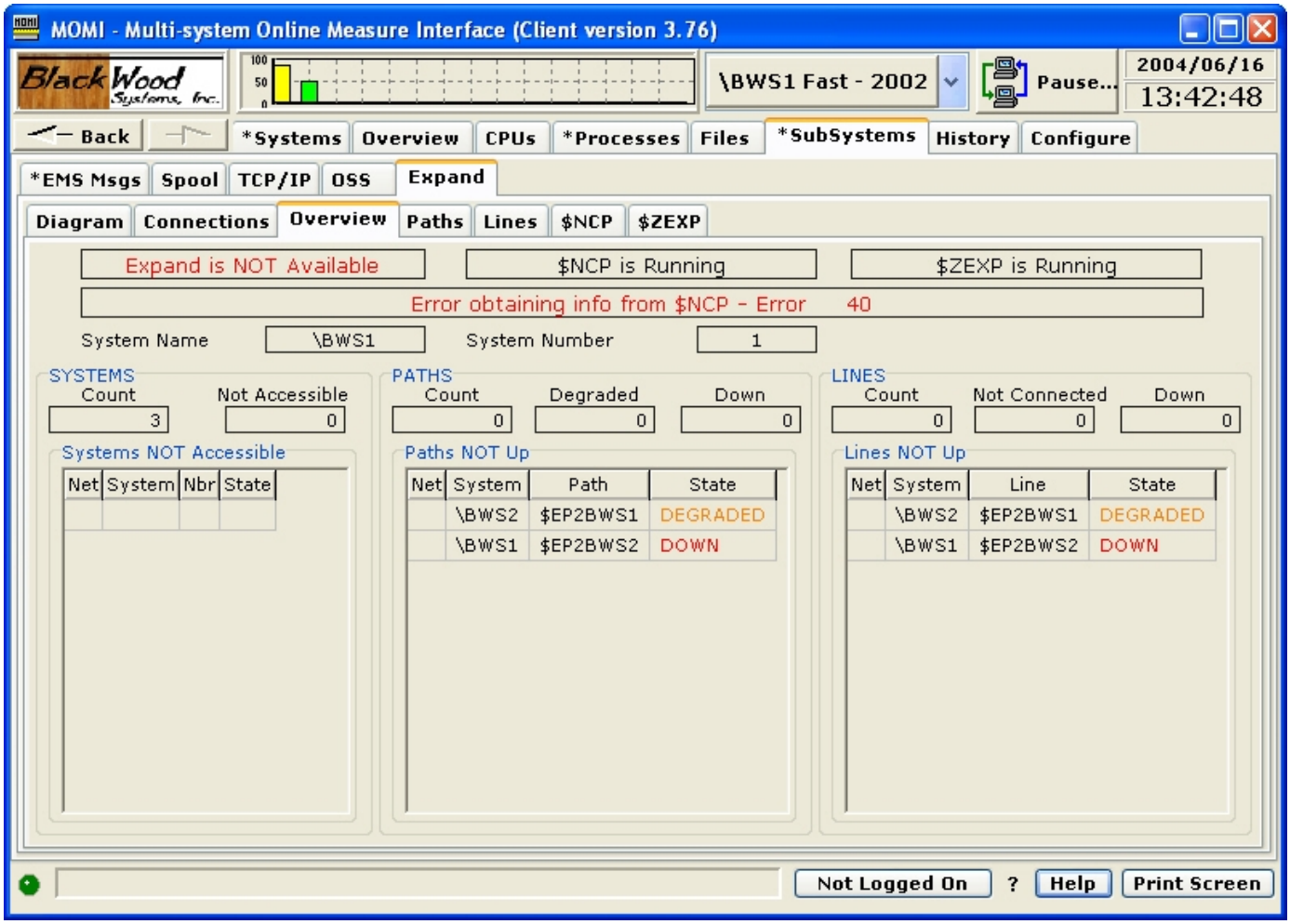
Expand / Overview



The Expand / Overview screen presents a general view of the status of the Expand subsystem.

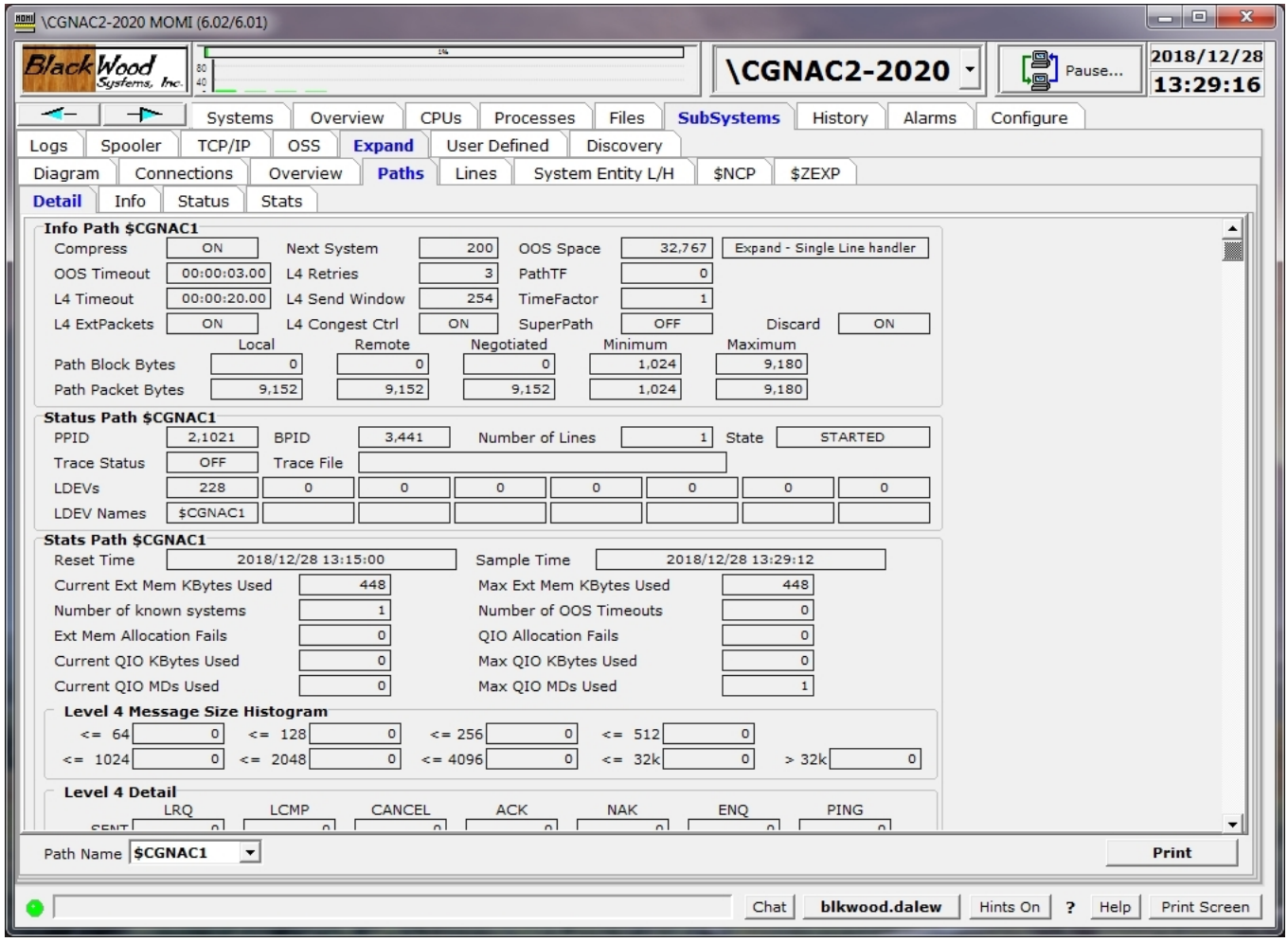
The top portion of the screen shows the result of a general determination of Expand availability by checking the existence of mandatory processes $NCP and $ZEXP . The lower portion of the screen shows any System not accessible via Expand, and any path or line not in a 'good' state. The middle portion of the screen reflects counts for these problem areas.

In the event of a severe error, an additional error message is displayed.


MOMI - Multi-system Online Measure Interface (Client version 3.76)
BlackWood Systems, Inc.
\BWS1 Fast - 2002
Pause...
2004/06/16
13:42:48
Back
*Systems
Overview
CPUs
*Processes
Files
*SubSystems
History
Configure
*EMS Msgs
Spool
TCP/IP
OSS
Expand
Diagram
Connections
Overview
Paths
Lines
$NCP
$ZEXP
Expand is NOT Available
$NCP is Running
$ZEXP is Running
Error obtaining info from $NCP - Error 40
System Name
\BWS1
System Number
1
SYSTEMS
Count
3
Not Accessible
0
Systems NOT Accessible
Net
System
Nbr
State
PATHS
Count
0
Degraded
0
Down
0
Paths NOT Up
Net
System
Path
State
\BWS2
$EP2BWS1
DEGRADED
\BWS1
$EP2BWS2
DOWN
LINES
Count
0
Not Connected
0
Down
0
Lines NOT Up
Net
System
Line
State
\BWS2
$EP2BWS1
DEGRADED
\BWS1
$EP2BWS2
DOWN
Not Logged On
?
Help
Print Screen

Paths

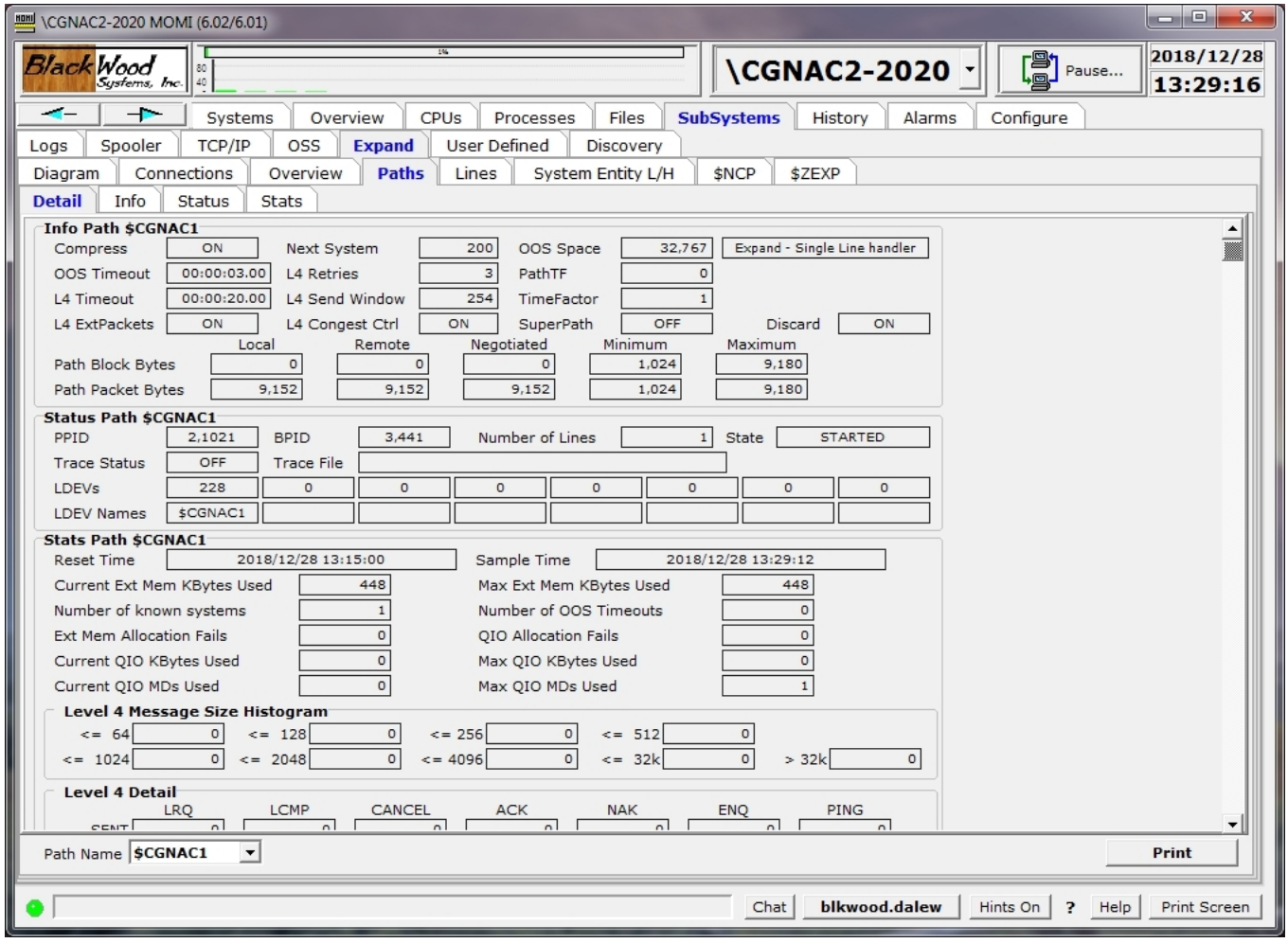
Paths / Detail



The Paths / Detail screen presents an SCF-type view of the Info, Status and Stats for an Expand Path.

The purpose of this format is to provide the User a familiar layout showing all possible data for a given path. Use the scroll bar on the right to view all of the available information.

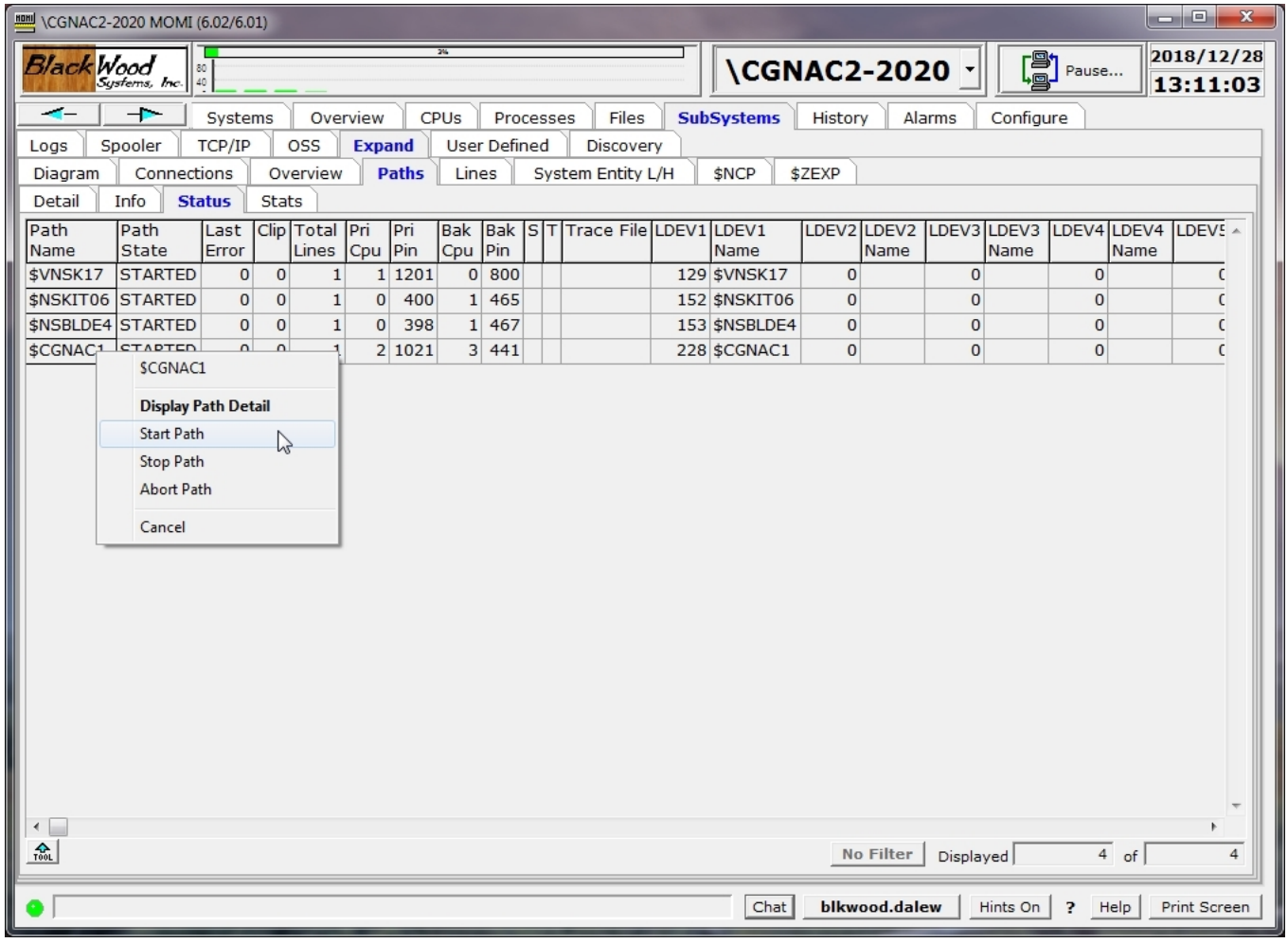
Paths / Info



The Paths / Info screen presents the configuration information for Expand Paths.

Various fields concerning Path Block Bytes and Path Packet Bytes used for Expand over IP are updated by Expand after the line is connected with the remote system and reflects final negotiated values.

Paths / Status

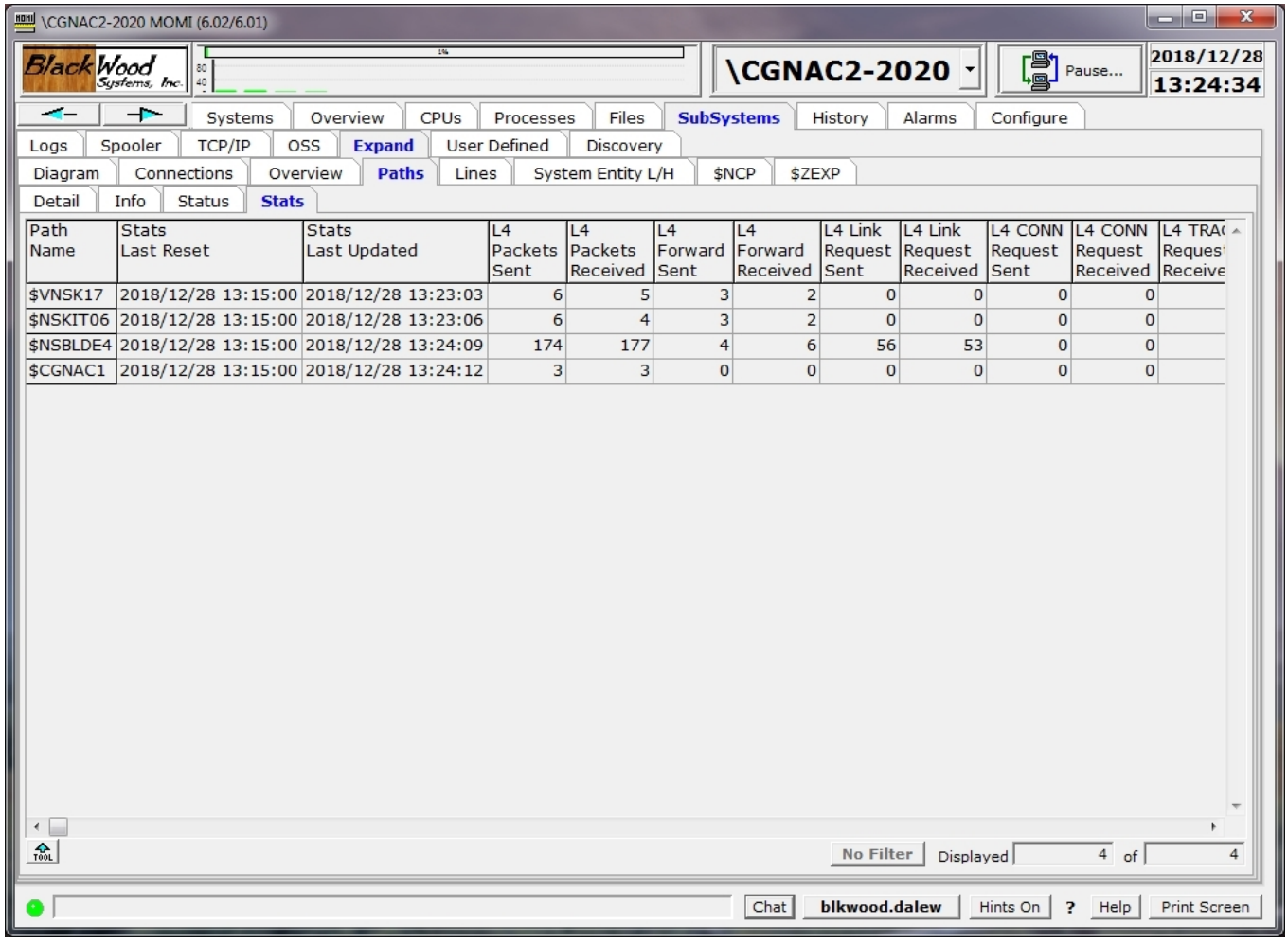


The Paths / Status screen presents the state and status of Expand Paths.

A path is made up of one or more physical lines. The fields LDEV1 - LDEV8 list these lines.

Right-click on a Path Name allows the jump to display the Paths / Detail screen or perform actions on the path.

Paths / Stats



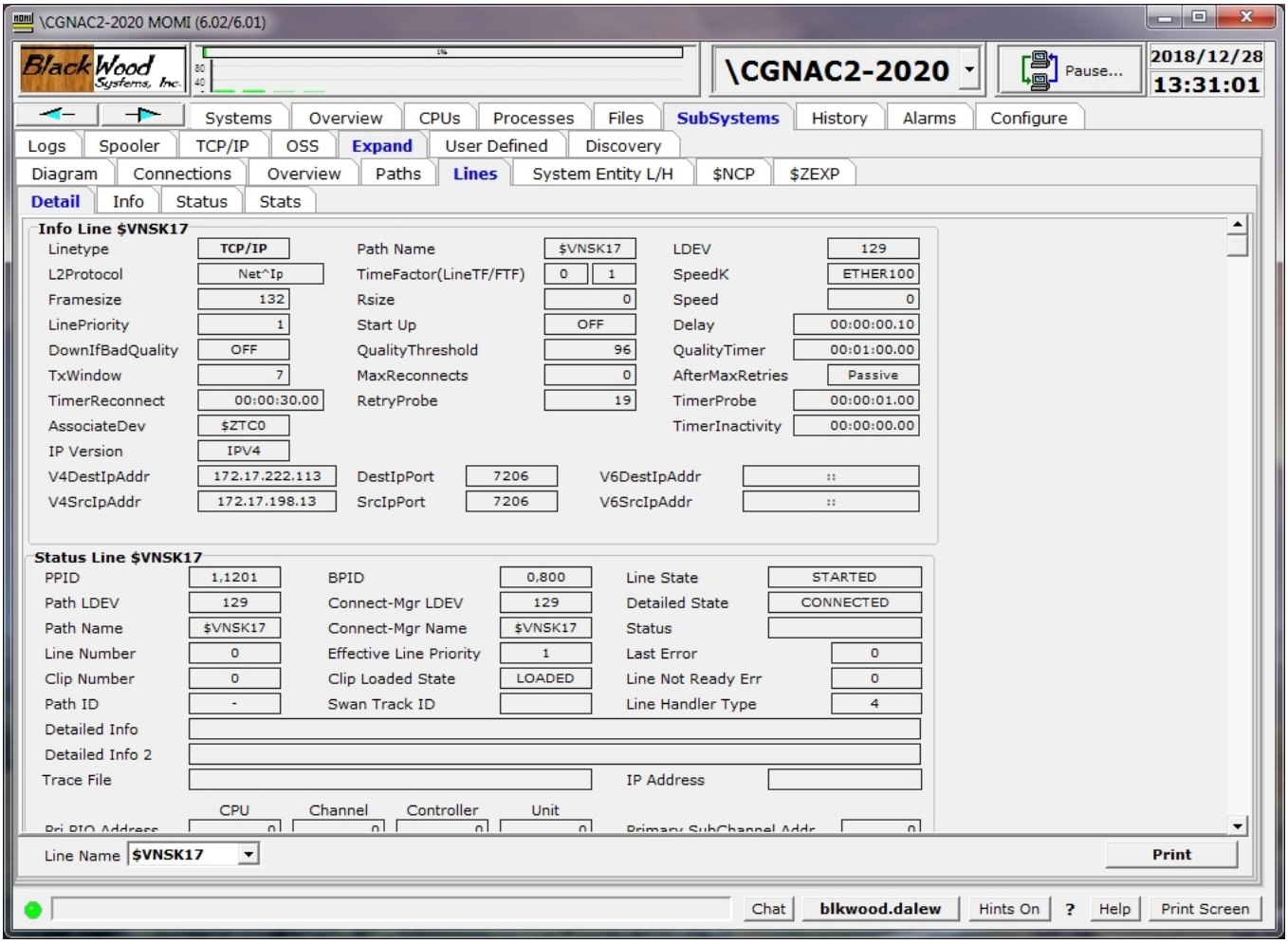
The Paths / Stats screen presents statistical information about an Expand Path.

An Expand Path is made up of one or more physical lines. The statistics represent the total of the underlying lines.

Some older versions of Expand do not report all fields displayed. QIO memory information is not reported by all Expand Path types.

Lines

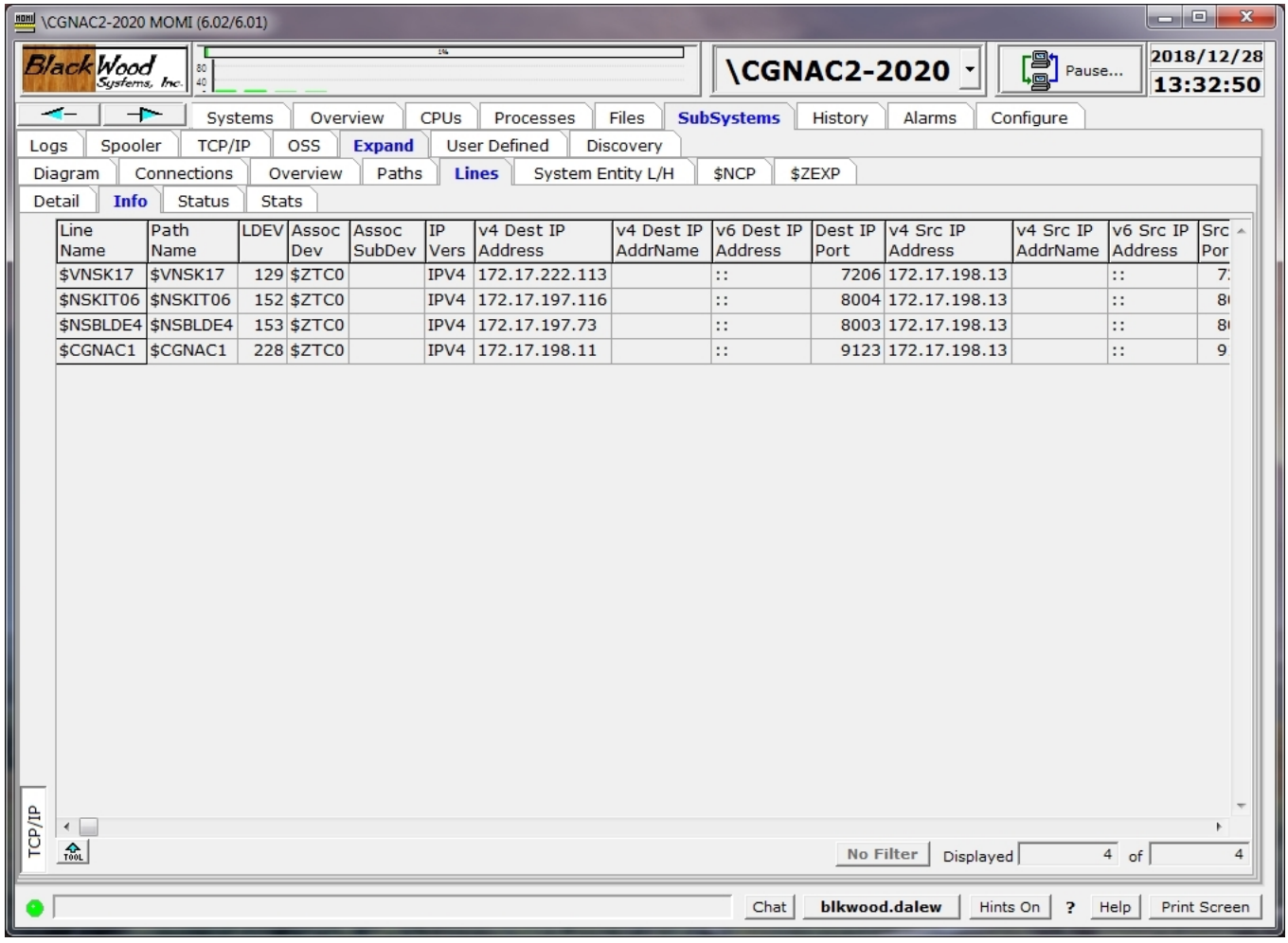
Lines / Detail



The Lines / Detail screen presents an SCF-type view of the Info, Status and Stats for an Expand Line.

The purpose of this format is to provide the User a familiar layout showing all possible data for a given line. Use the scroll bar on the right to view all of the available information.

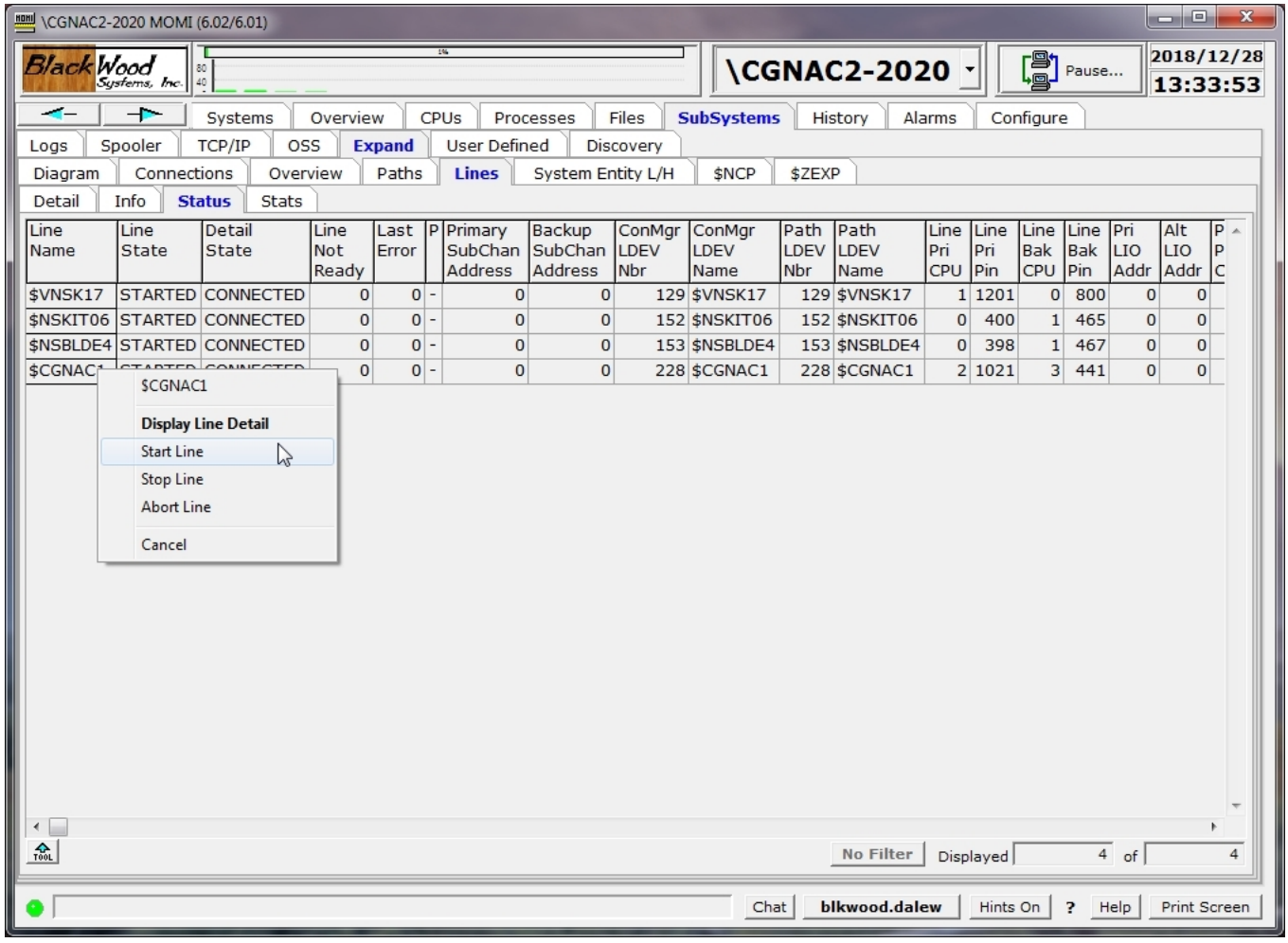
Lines / Info



The Lines / Info screen presents configuration information for Expand Lines.

Various Expand Line types report different information. Each line type is listed under tabs along the left side of the screen.

Lines / Status

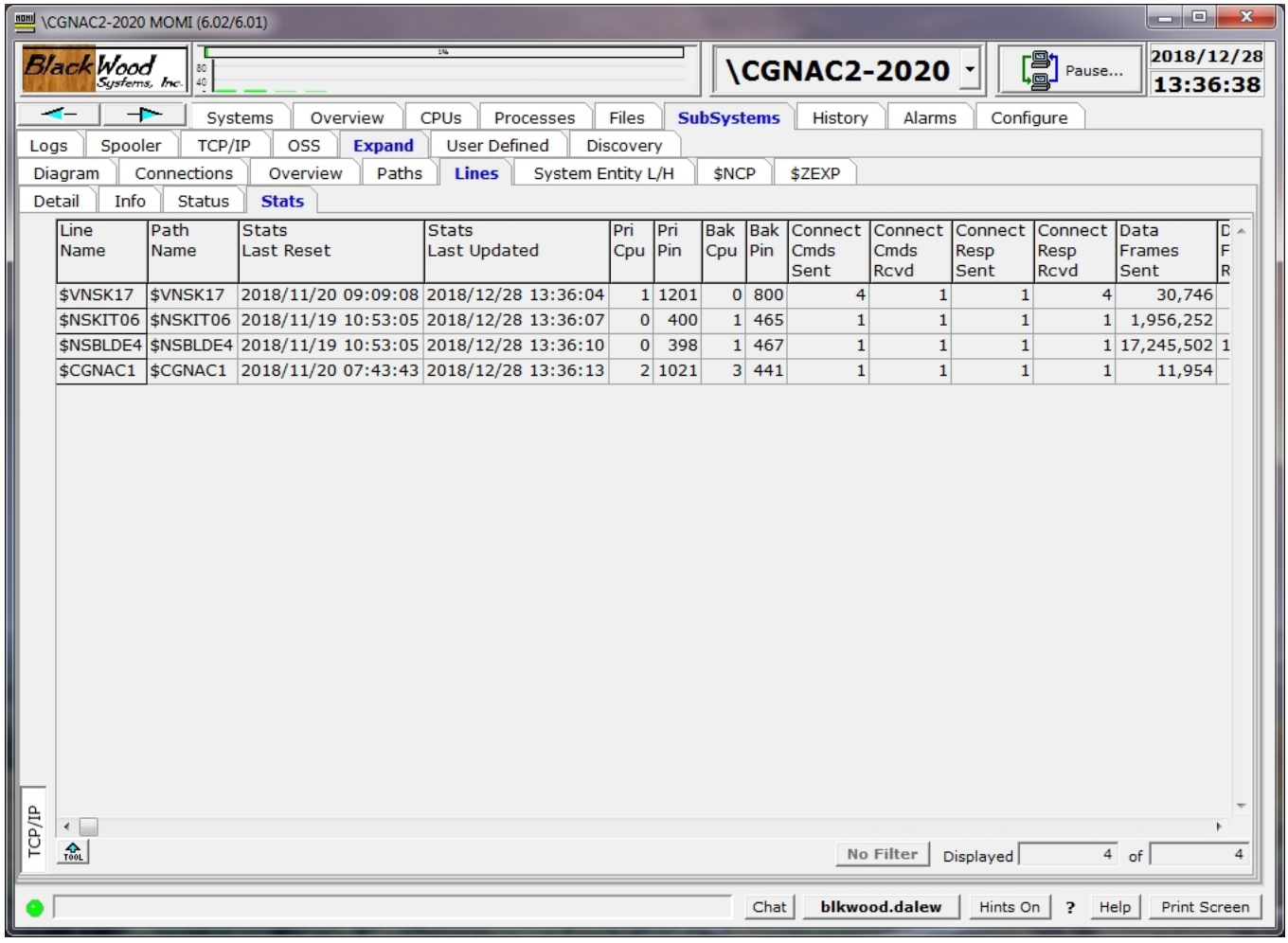


The Lines / Status screen presents status and additional configuration information about Expand Lines.

Not all configuration fields are used by all Expand Line types. The Detail Info fields off the screen (scroll to display) supply additional information under error conditions.

Right-click on a Line Name allows a jump to display the Lines / Detail screen or to perform actions.

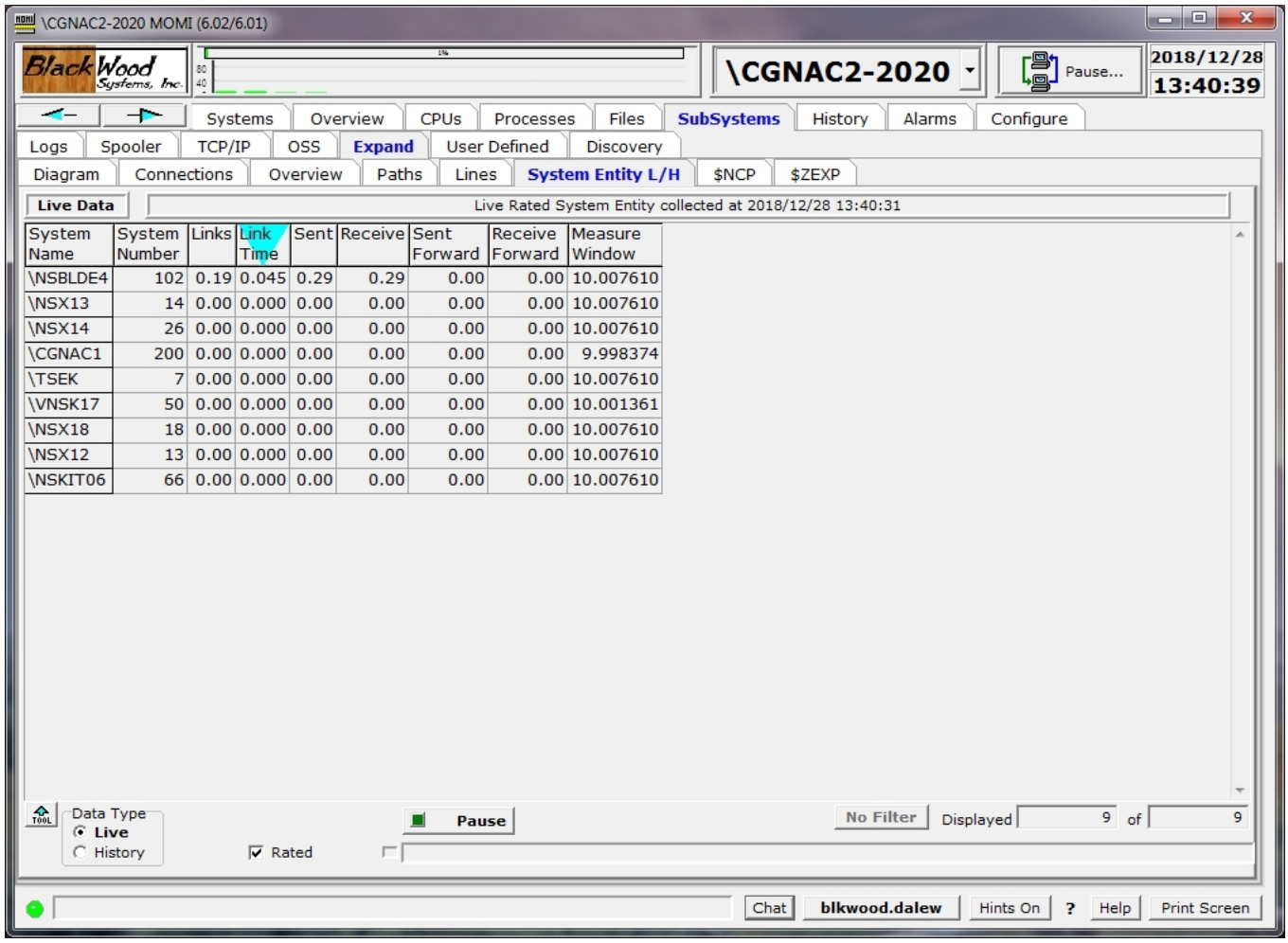
Lines / Stats



The Lines / Stats screen presents statistical information about an Expand Line.

Various Expand Line types report different information. Each line type is listed under tabs along the left side of the screen.

Expand / System Entity

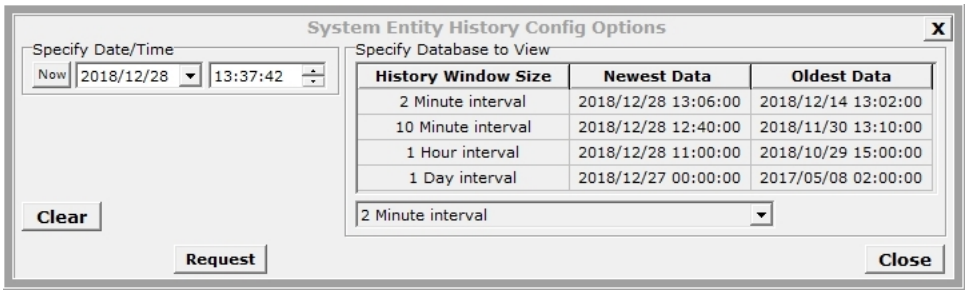


System Name	System Number	Links	Link Time	Sent	Receive	Sent Forward	Receive Forward	Measure Window
\NSBLDE4	102	0.19	0.045	0.29	0.29	0.00	0.00	10.007610
\NSX13	14	0.00	0.000	0.00	0.00	0.00	0.00	10.007610
\NSX14	26	0.00	0.000	0.00	0.00	0.00	0.00	10.007610
\CGNAC1	200	0.00	0.000	0.00	0.00	0.00	0.00	9.998374
\TSEK	7	0.00	0.000	0.00	0.00	0.00	0.00	10.007610
\VNSK17	50	0.00	0.000	0.00	0.00	0.00	0.00	10.001361
\NSX18	18	0.00	0.000	0.00	0.00	0.00	0.00	10.007610
\NSX12	13	0.00	0.000	0.00	0.00	0.00	0.00	10.007610
\NSKIT06	66	0.00	0.000	0.00	0.00	0.00	0.00	10.007610

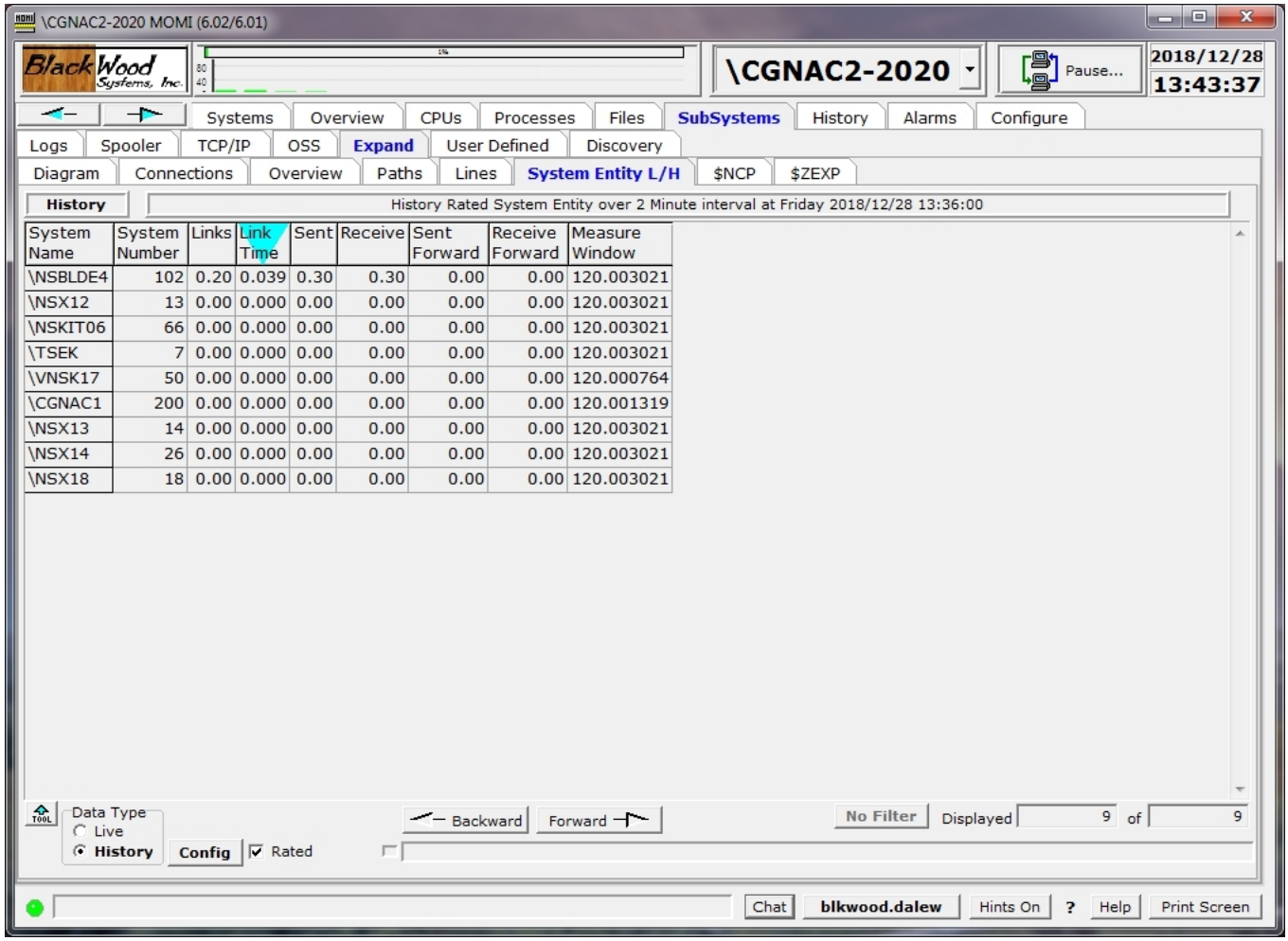
The Expand / System Entity screen displays the MEASURE entity System for all known network nodes. By default, the data is displayed live, collected every Interval, or may be displayed from MOMI history.

When first entering the screen the data is displayed Live and automatically updated.

History, or prior information, may be displayed by pressing the **History** button.



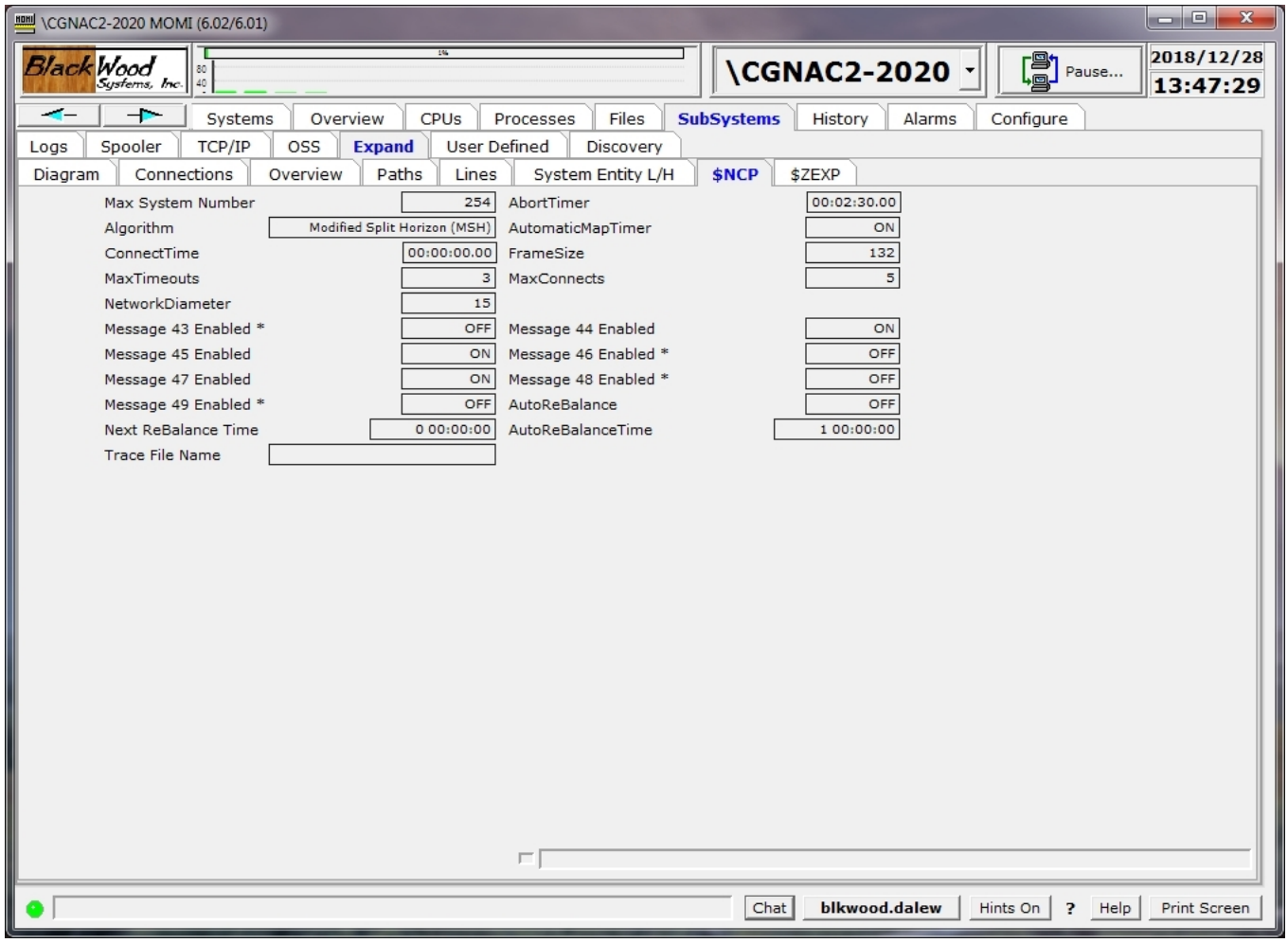
Select the desired start date/time, history file and push **Request**.



System Name	System Number	Links	Link Time	Sent	Receive	Sent Forward	Receive Forward	Measure Window
\NSBLDE4	102	0.20	0.039	0.30	0.30	0.00	0.00	120.003021
\NSX12	13	0.00	0.000	0.00	0.00	0.00	0.00	120.003021
\NSKIT06	66	0.00	0.000	0.00	0.00	0.00	0.00	120.003021
\TSEK	7	0.00	0.000	0.00	0.00	0.00	0.00	120.003021
\VNSK17	50	0.00	0.000	0.00	0.00	0.00	0.00	120.000764
\CGNAC1	200	0.00	0.000	0.00	0.00	0.00	0.00	120.001319
\NSX13	14	0.00	0.000	0.00	0.00	0.00	0.00	120.003021
\NSX14	26	0.00	0.000	0.00	0.00	0.00	0.00	120.003021
\NSX18	18	0.00	0.000	0.00	0.00	0.00	0.00	120.003021

The history mode of the screen shows additional **forward / backward** buttons to allow stepping through the history records.

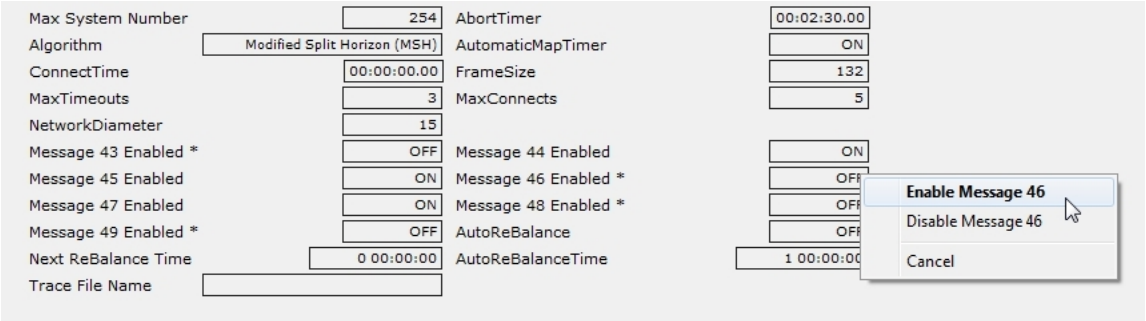
Expand / $NCP



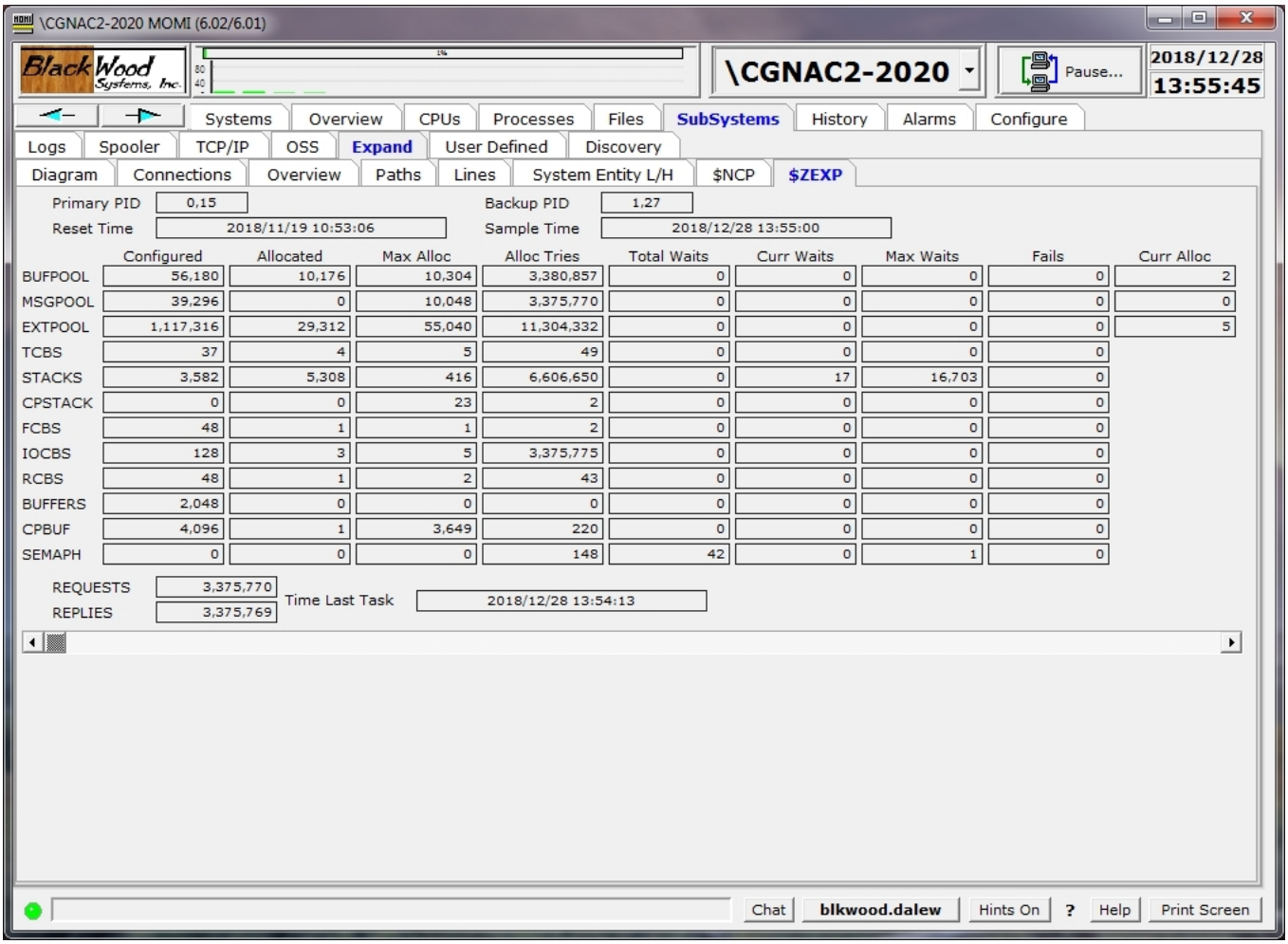
The Expand / $NCP screen presents information about the Network Control Process.

The NCP process is the main 'traffic cop' for the Expand subsystem. While actual 'user data' does not flow through this process, it does determine which Paths are used for transport and it maintains routes and the list of systems on the network.

Certain messages are logged to $0 as the result of changes in the Expand network, such as paths/lines up/down and processors in remote nodes up/down. These messages can be turned on or off. Right-click on fields marked with an * to perform the action. Changes may not be visible right away.



Expand / $ZEXP



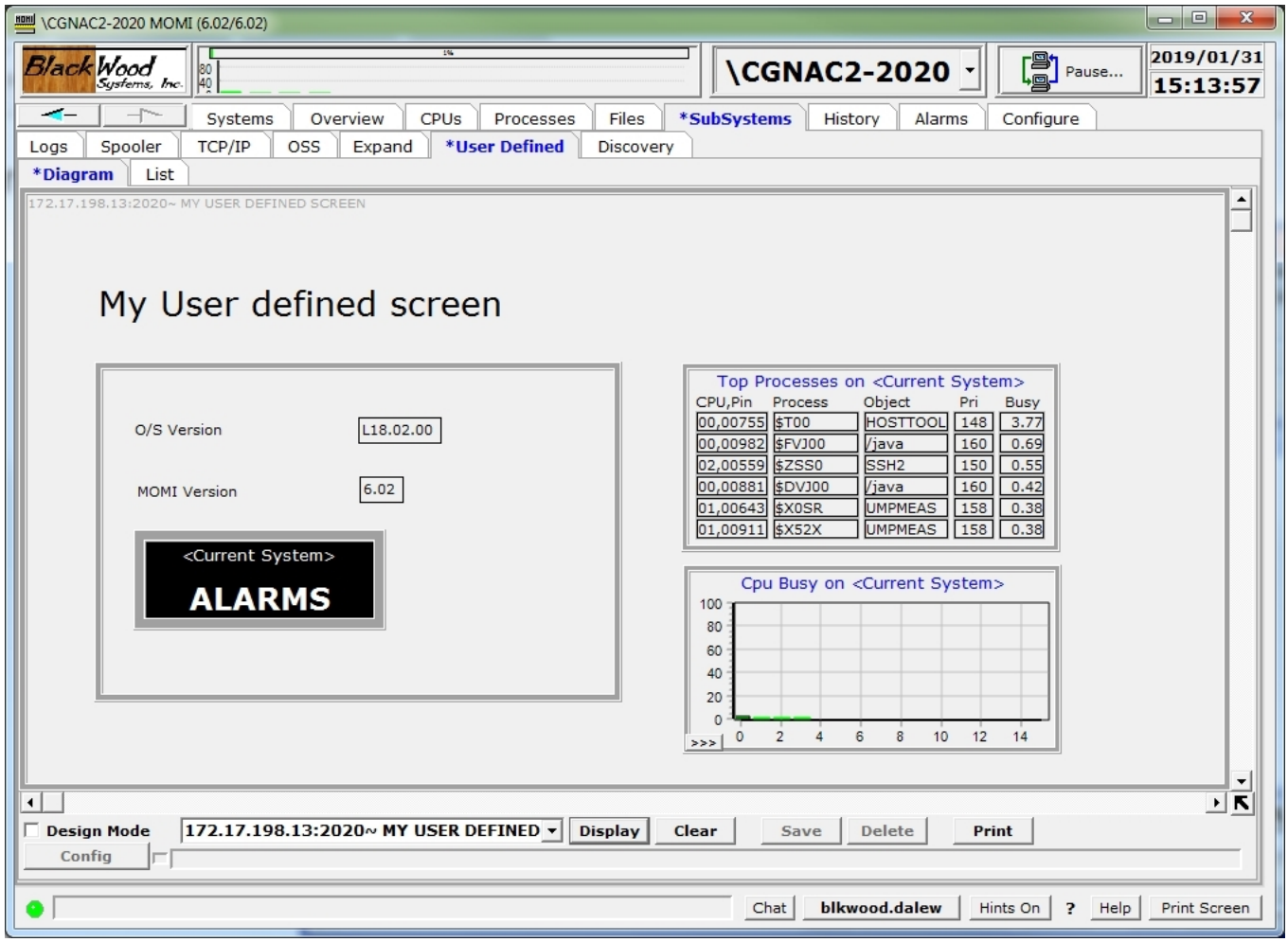
The Expand / $ZEXP screen presents various statistics about the Expand Manager Process. This information is the equivalent of the command "SCF stats process $ZEXP".

The Expand Manager Process supervises the Expand Path and Line handler processes and must be running in order for Expand to be active. Information requests through SCF and the SPI interface are routed through this process.

Unfortunately, the meaning of the values displayed on this screen are not known. The information presented is undocumented and nothing is available (that we can find) to interpret these values. We have included this screen because the information looked interesting.

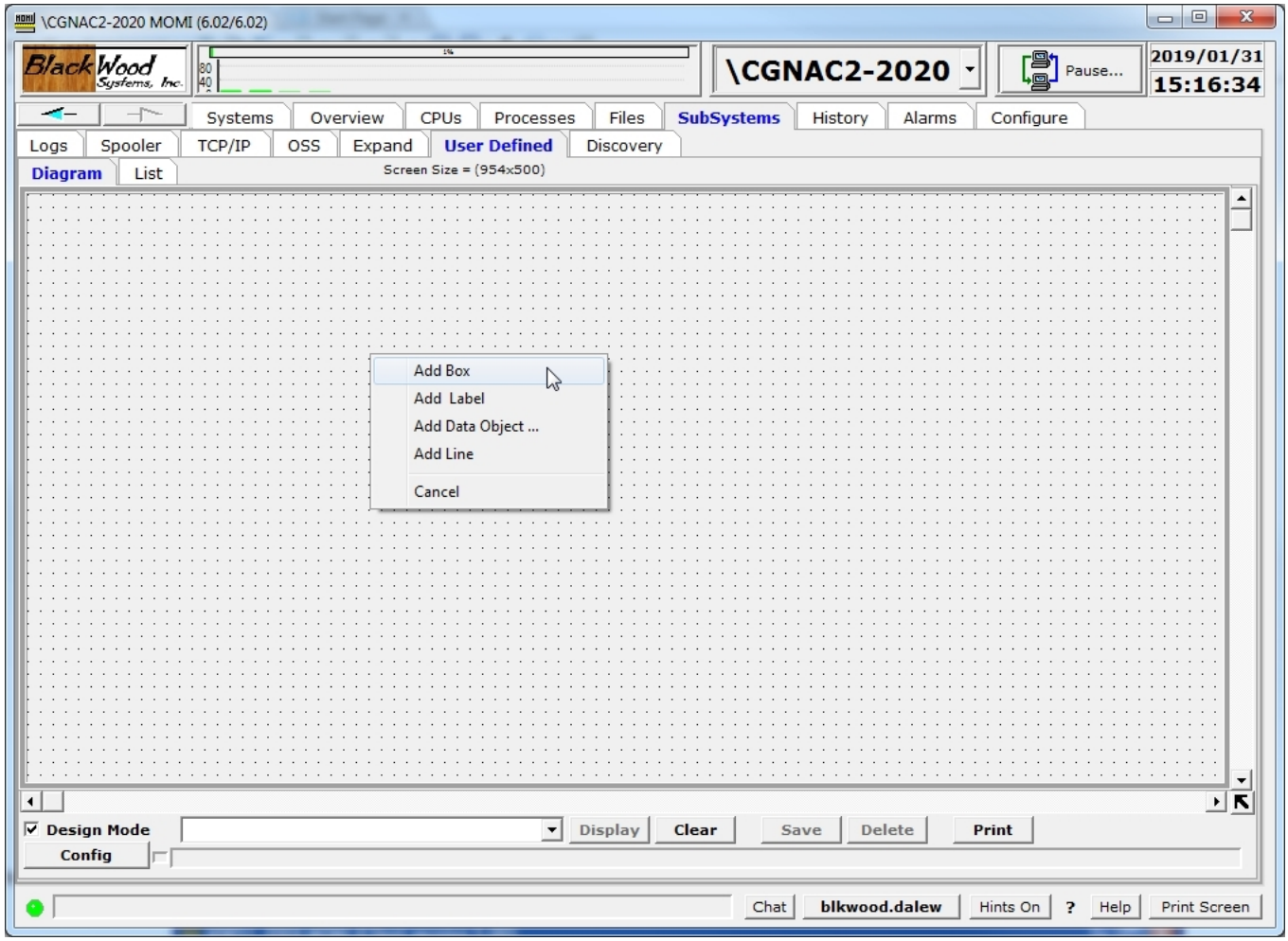
User Defined

User Defined / Diagram

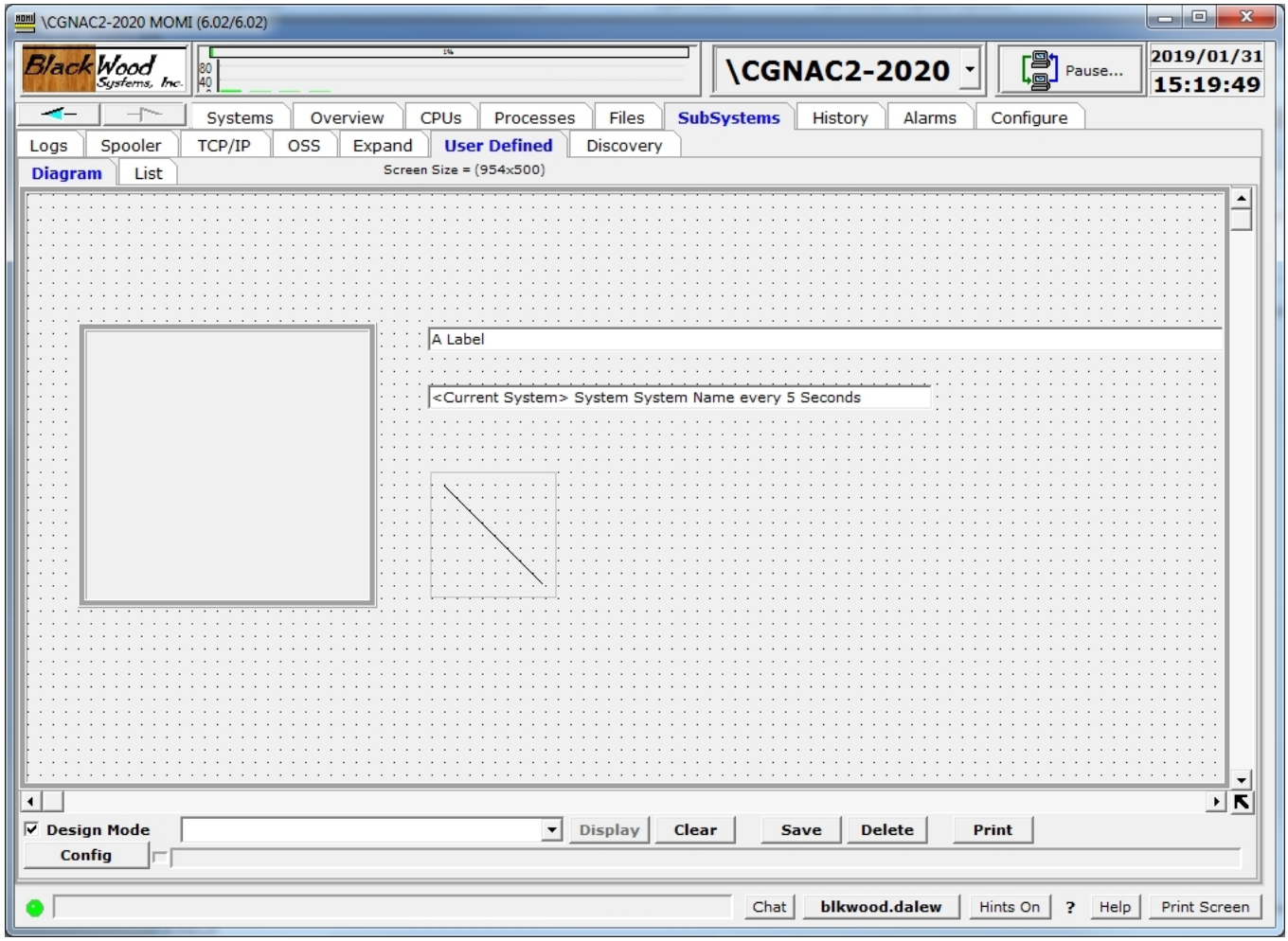


The User Defined / Diagram page allows the user to construct, in a some-what free-form manner, a page of items to display. The items on-screen automatically update. The User Defined page, once created, may be saved for later use and may be displayed within the Diagram page or in a window independent of the main MOMI PC Client screen. The MOMI PC Client must be running as the independent windows are still created by it.

The Diagram page starts 'empty'. The user clicks the box next to **Design Mode** to allow entry / definition of items on screen. Design mode is also used to change existing items. User Define screens are saved under the current User ID. Uncheck **Design Mode** to activate the screen. It may take several seconds for all fields to populate.

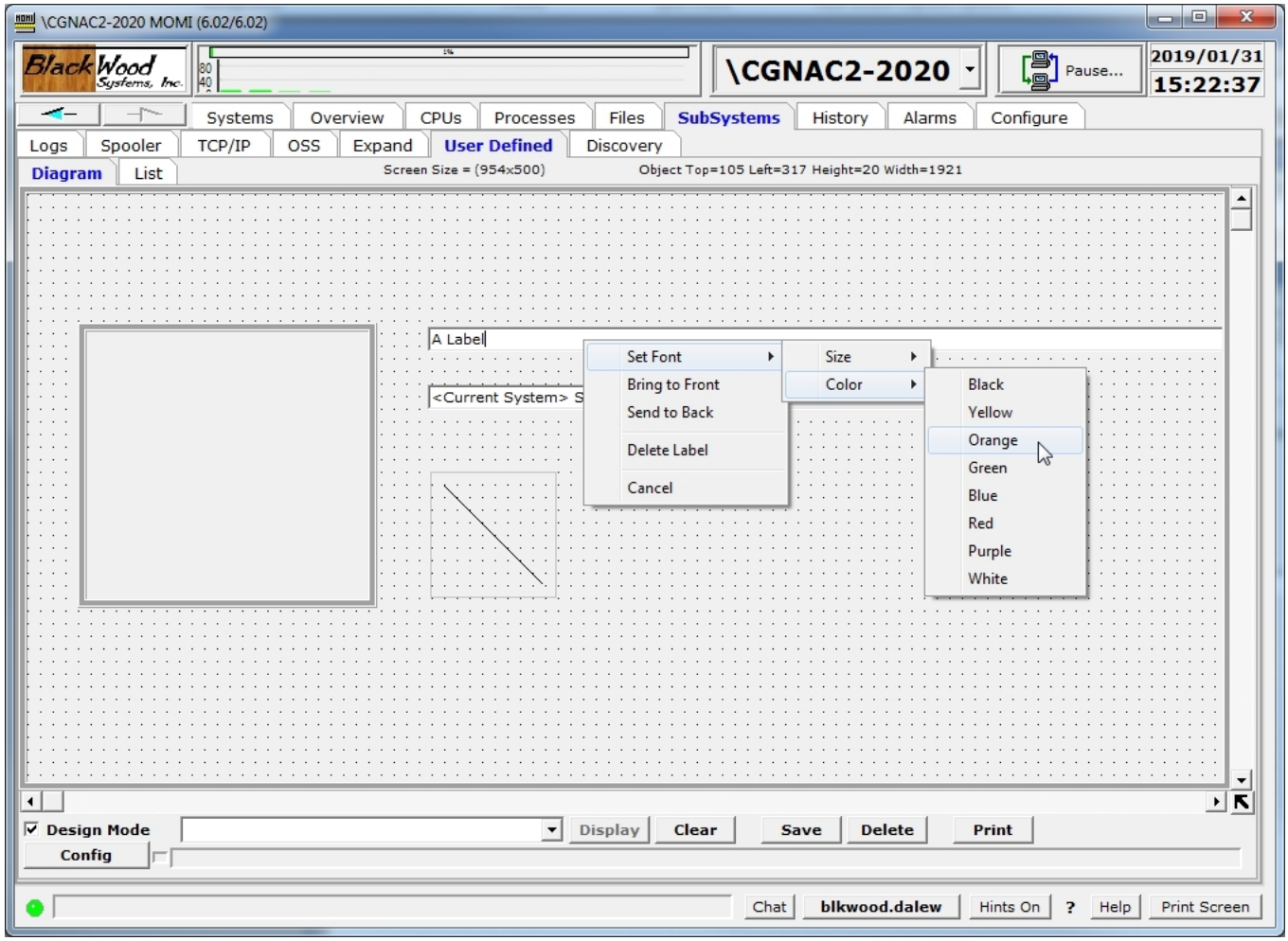


The following picture shows a box, label, data object and line added to the screen.

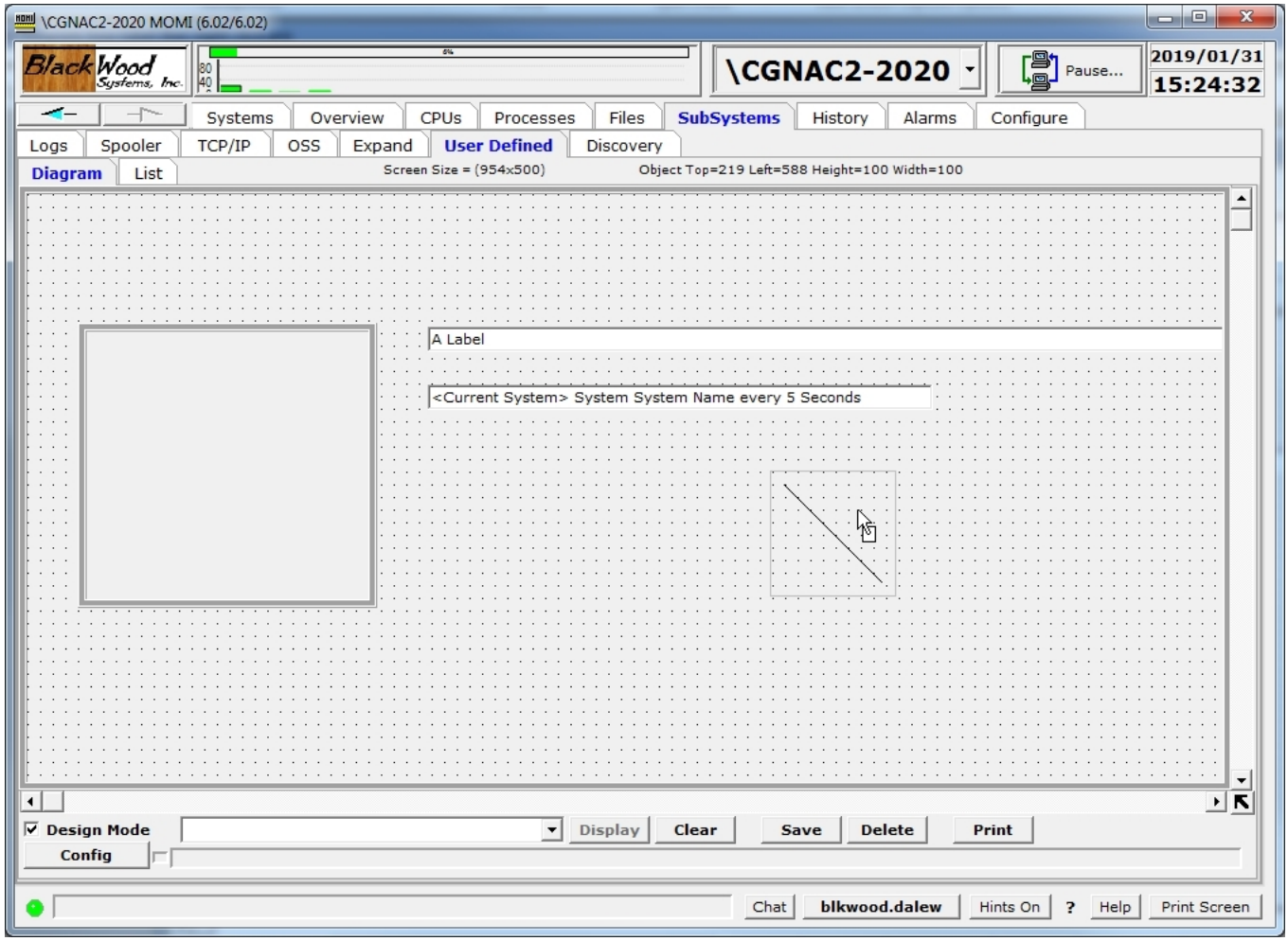


The box is an object used to group or gather items. A label allows the entry to user defined text to provide fixed text information. The data object provides many items for the user to choose from such as CPU, Process and System entities. The line provides an additional graphic for the user to separate or perhaps indicate data flows.

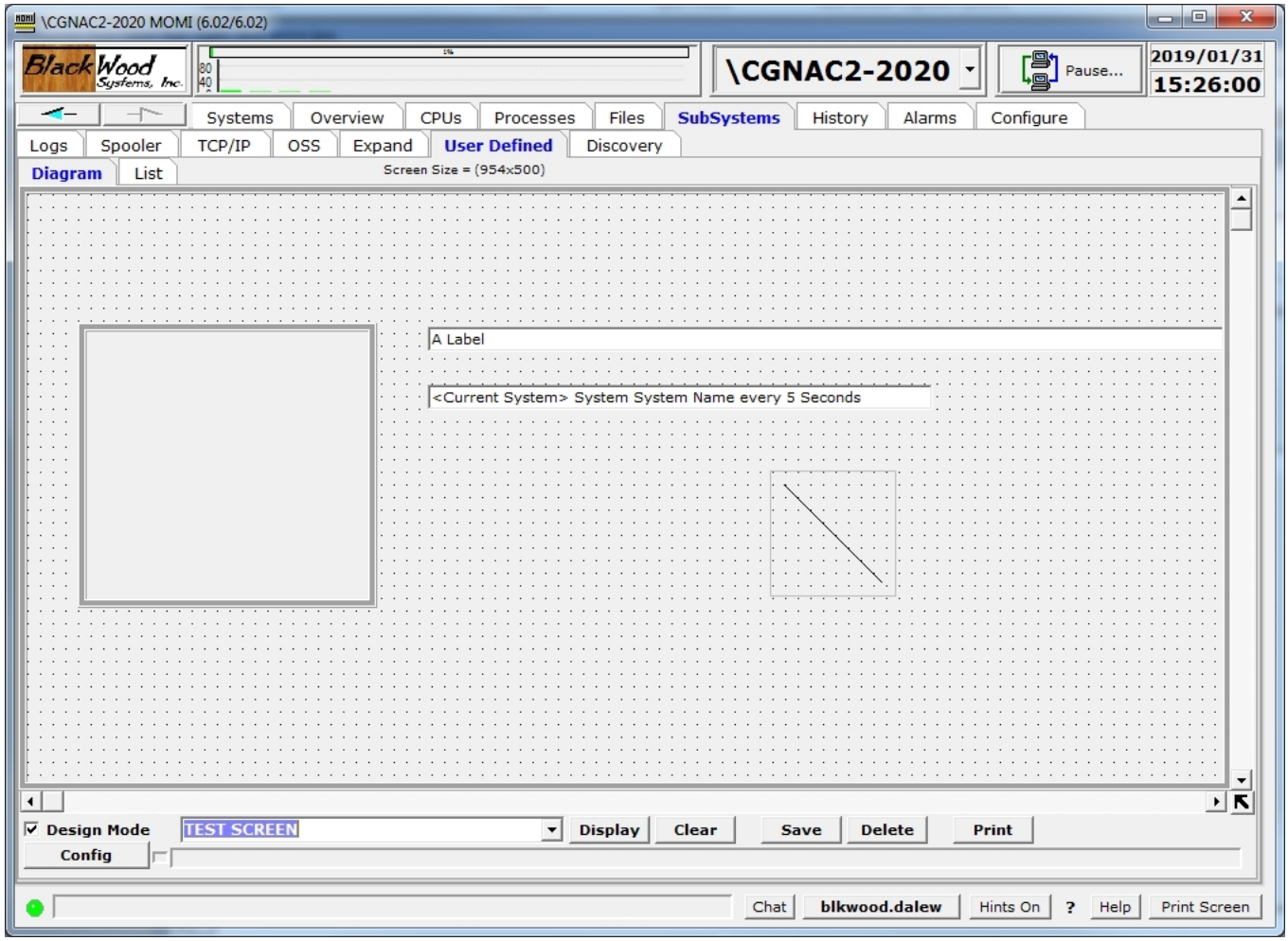
Once an item is on screen, such as the label, right-click over the item to bring up available properties. For the label, the font size and color may be changed. Many items may also be brought forward and placed on top of other items (i.e. Bring to Front) to moved behind other items (i.e. Send to Back).



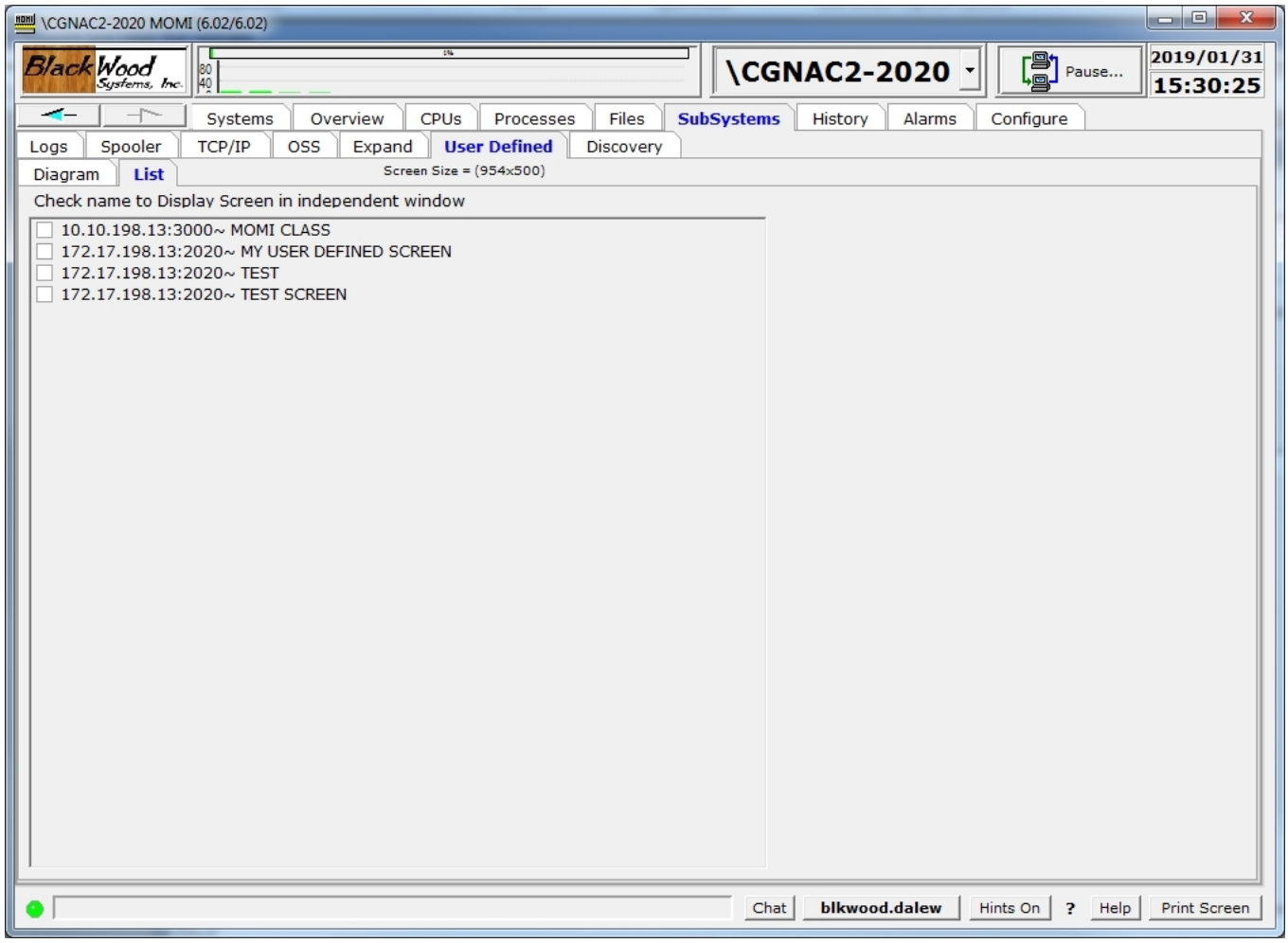
Items may also be moved by placing the mouse over the item, the actual position varies, and left-clicking the mouse and holding while moving to the new location. The line is made to go in different directions by grabbing the size of the box and dragging it larger or smaller.



To save a User Defined screen, enter a name in the field next to **Design Mode** and push **Save**. Existing screens are displayed by using the drop-down box at the end of the field then push **Display**.


\CGNAC2-2020 MOMI (6.02/6.02)
BlackWood Systems, Inc.
\CGNAC2-2020
Pause...
2019/01/31
15:26:00
Systems
Overview
CPUs
Processes
Files
SubSystems
History
Alarms
Configure
Logs
Spooler
TCP/IP
OSS
Expand
User Defined
Discovery
Diagram
List
Screen Size = (954x500)
A Label
<Current System> System System Name every 5 Seconds
Design Mode
TEST SCREEN
Display
Clear
Save
Delete
Print
Config
Chat
blkwood.dalew
Hints On
?
Help
Print Screen

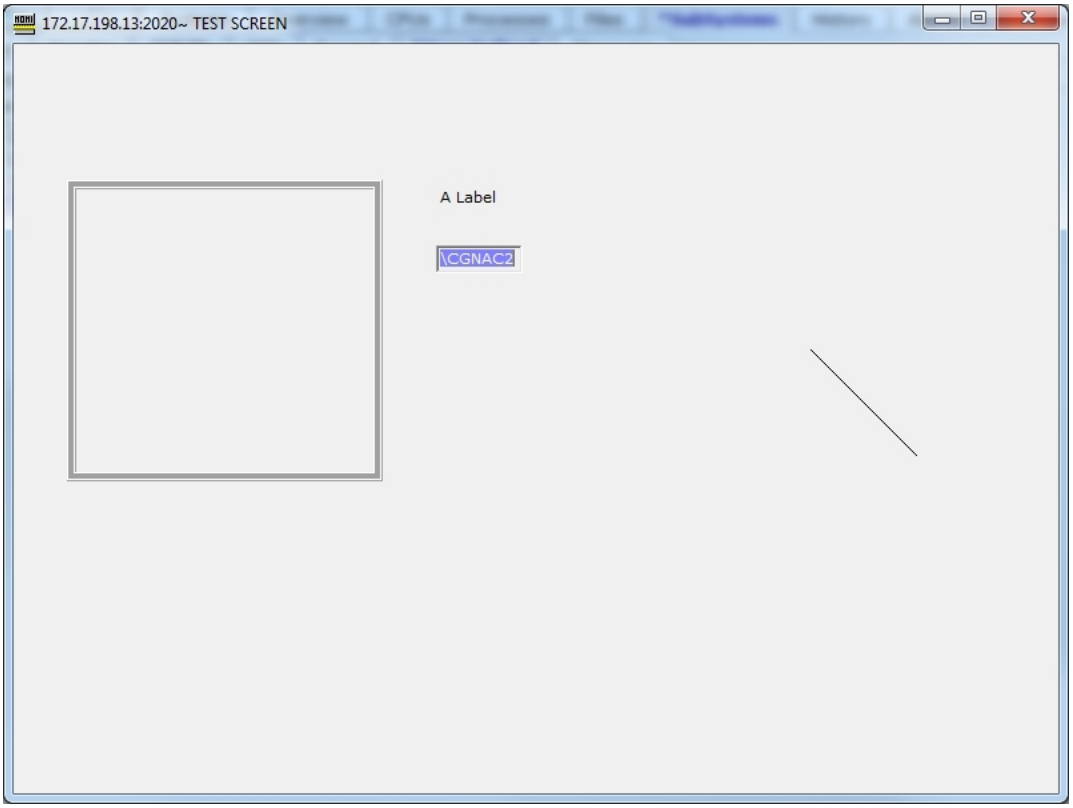
User Defined / List



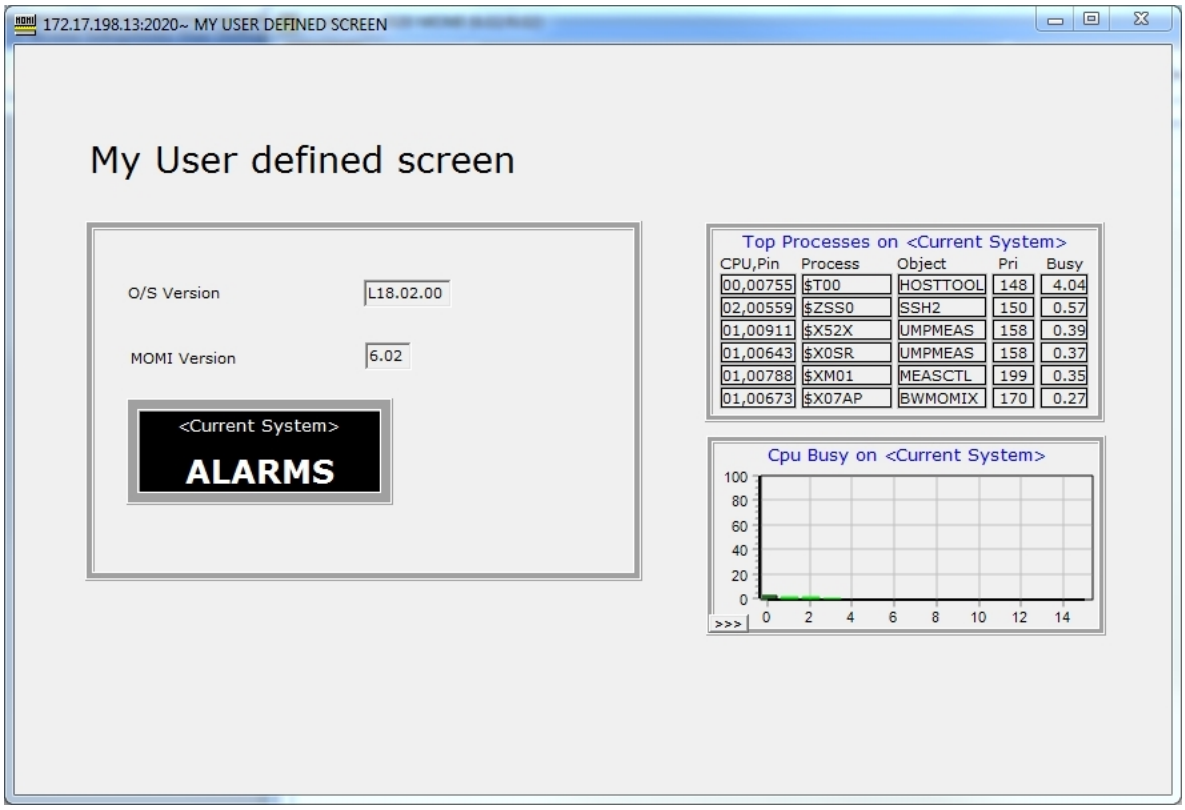
The User Defined / List screen displays screens previously defined and saved and allows them to be displayed as independent windows.

Screen definitions are physically saved on the NonStop System (i.e. not on the PC) and are organized by logged on user.

An independent window appears outside of the MOMI PC Client however the MOMI PC Client must be running for the independent window to display. A window or windows are checked and display separately from the MOMI PC Client on the PC monitor. The following window is a simple display.

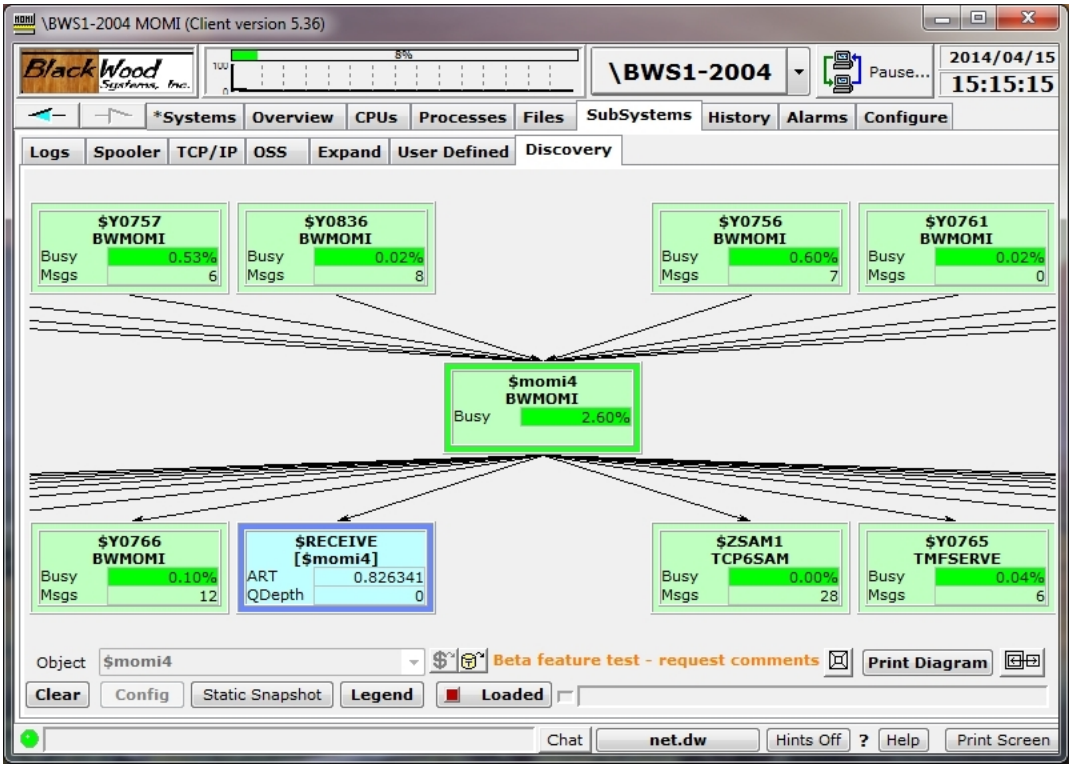


The following window displays additional items.



Discovery

(please see screen limitations below)

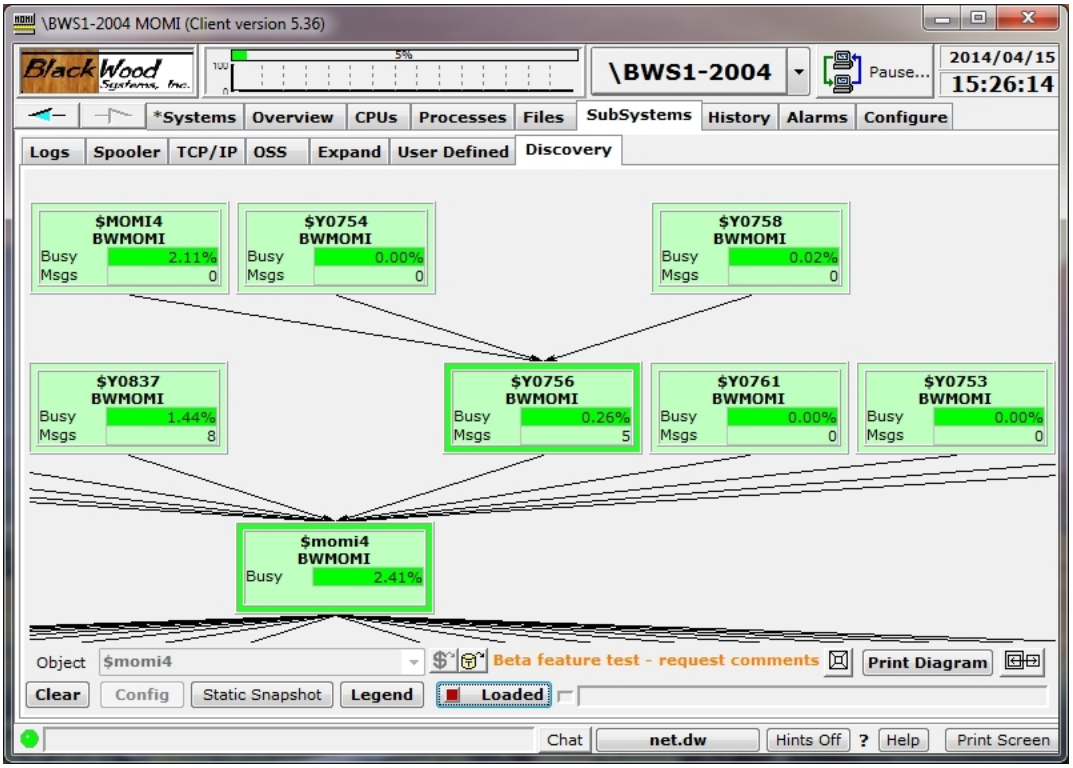


The Discovery screen displays a pictorial representation of what processes communicate "into" and "out of" the specified process.

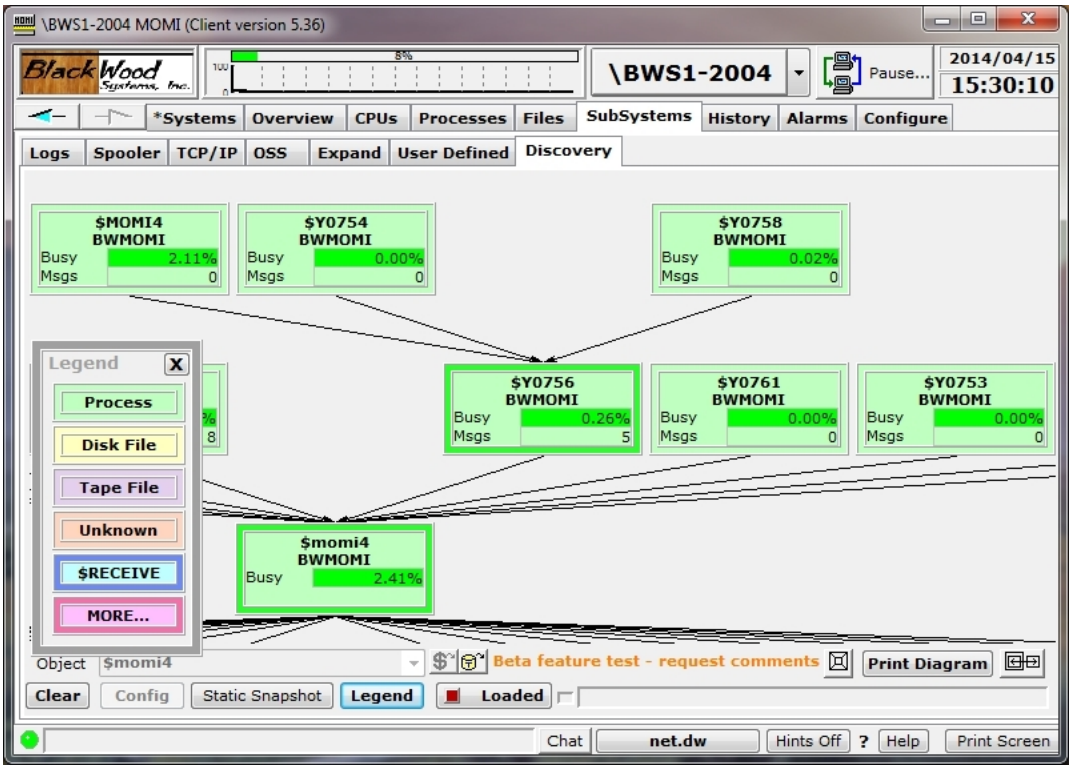
The screen starts empty. A process name is entered, or selected using a picker, and then press **Request**. The entered process is displayed in the middle of the screen. After a short measurement interval processes communicating to the specified process are displayed along the top of the screen. Processes that the selected process is communicating to are displayed along the bottom of the screen.

Place the mouse on the screen background, left-click and hold to drag the screen (i.e. move it around).

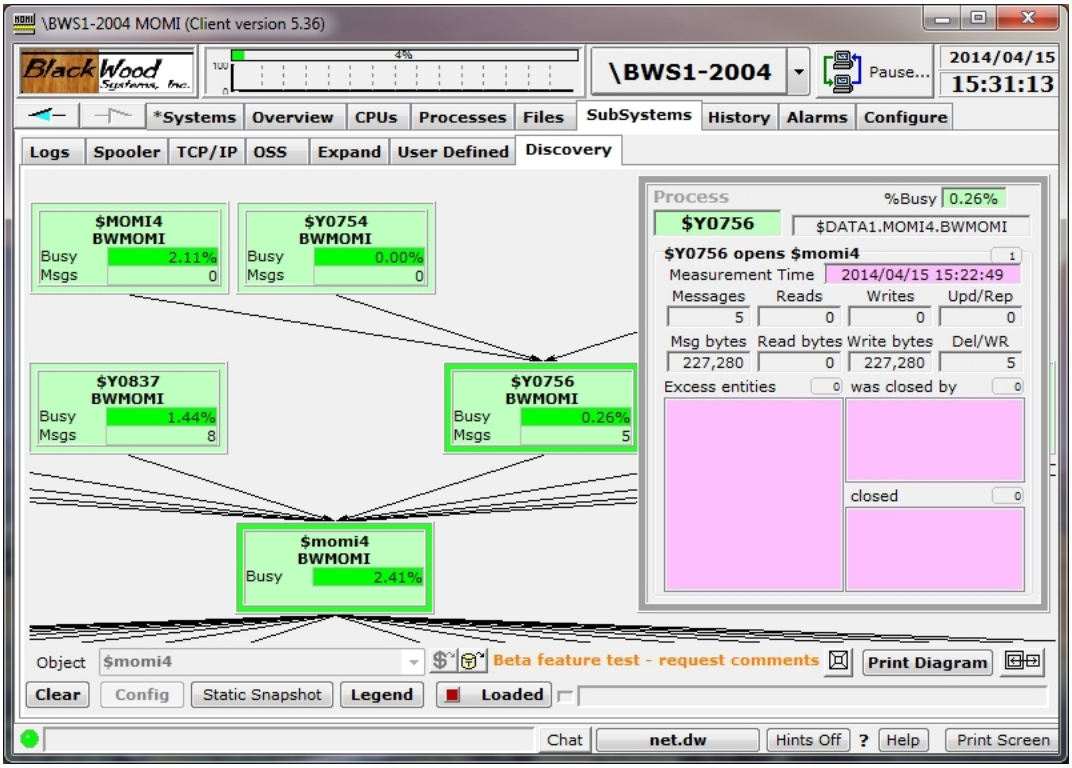
Place the mouse over an entry, right-click and select previous or next to display additional information. In the image below previous was selected.



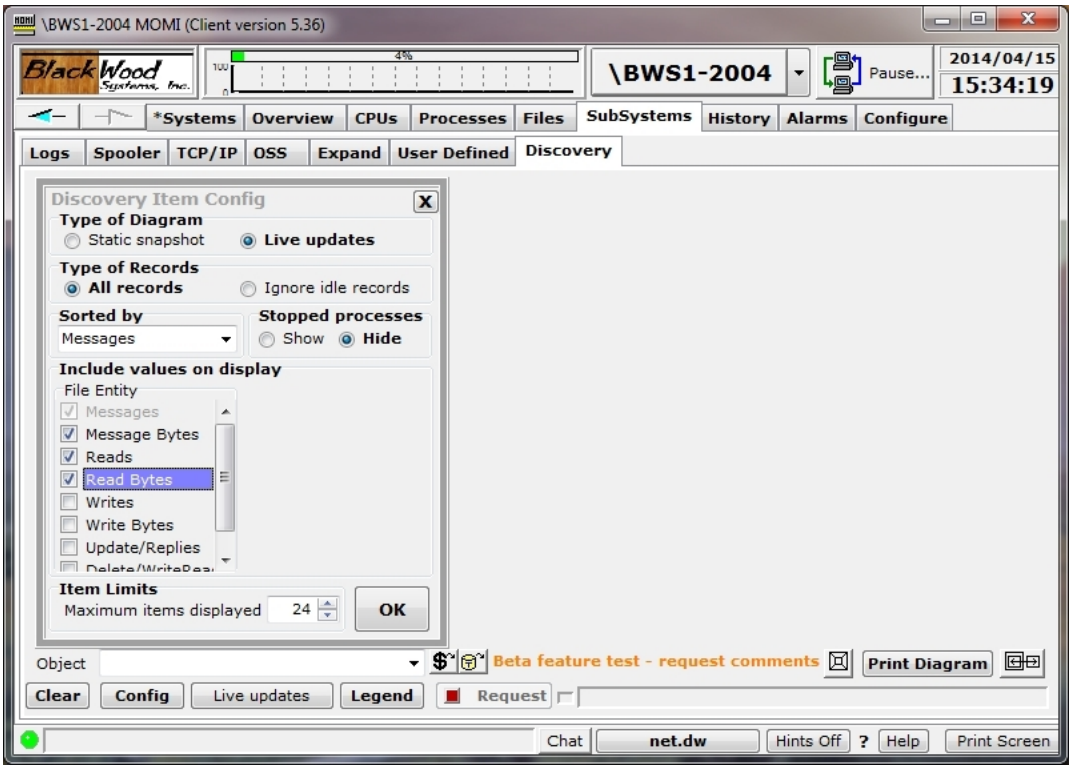
Colors are used to distinguish various process types. Press the Legend button for process types.

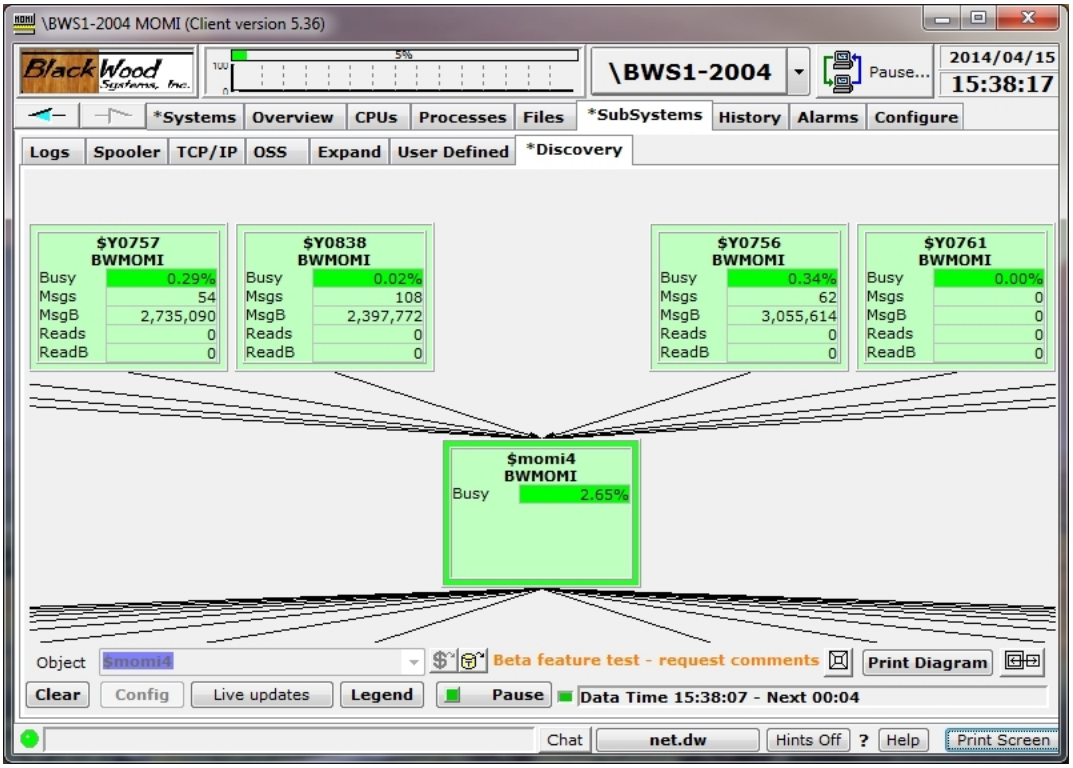


Place the mouse over an entry to display additional information.



Additional information fields about a process, live updates (of data fields) or a static snapshot are determined by pressing the **Config** button.





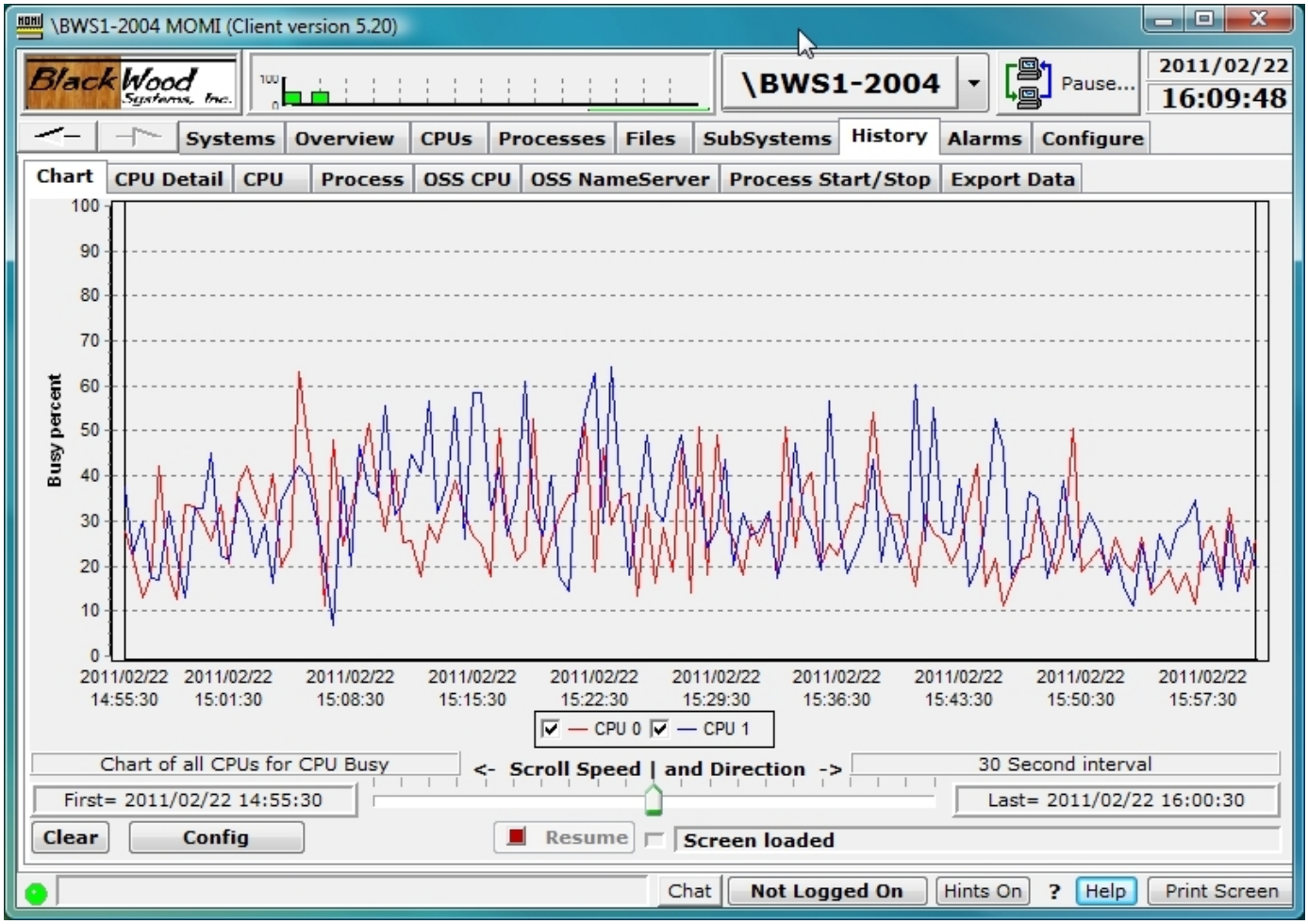
Screen limitations

This screen uses the MEASURE File entity primarily to obtain its information and to determine relationships. This works 'best' if direct I/O (i.e. open/write/read/etc...) is involved. However, if a process uses Pathsend for its I/O the LINKMON or ROUT processes effectively masks the real underlying I/O.

Unnamed processes also restrict the amount of information available from MEASURE.

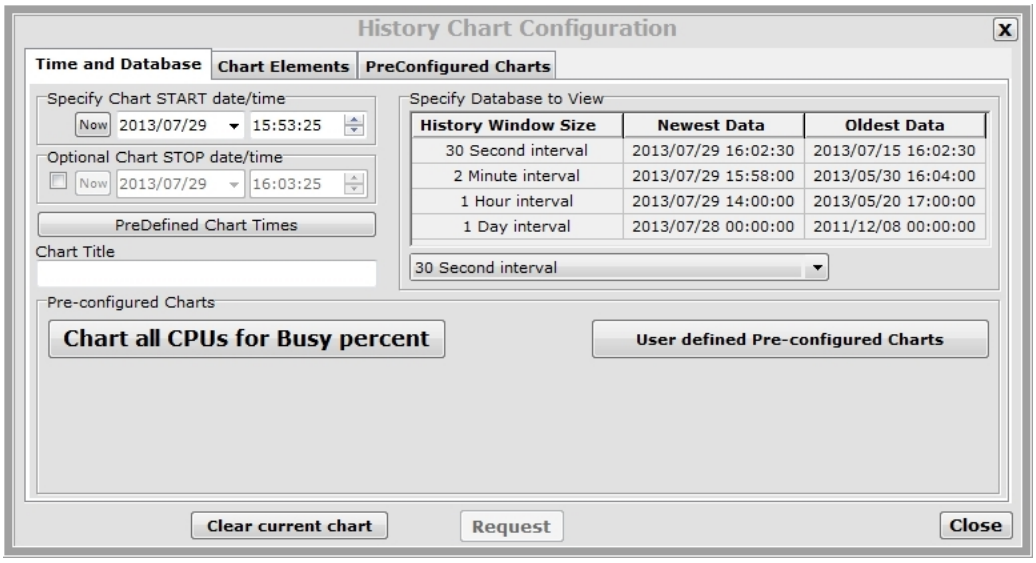
History

History / Chart

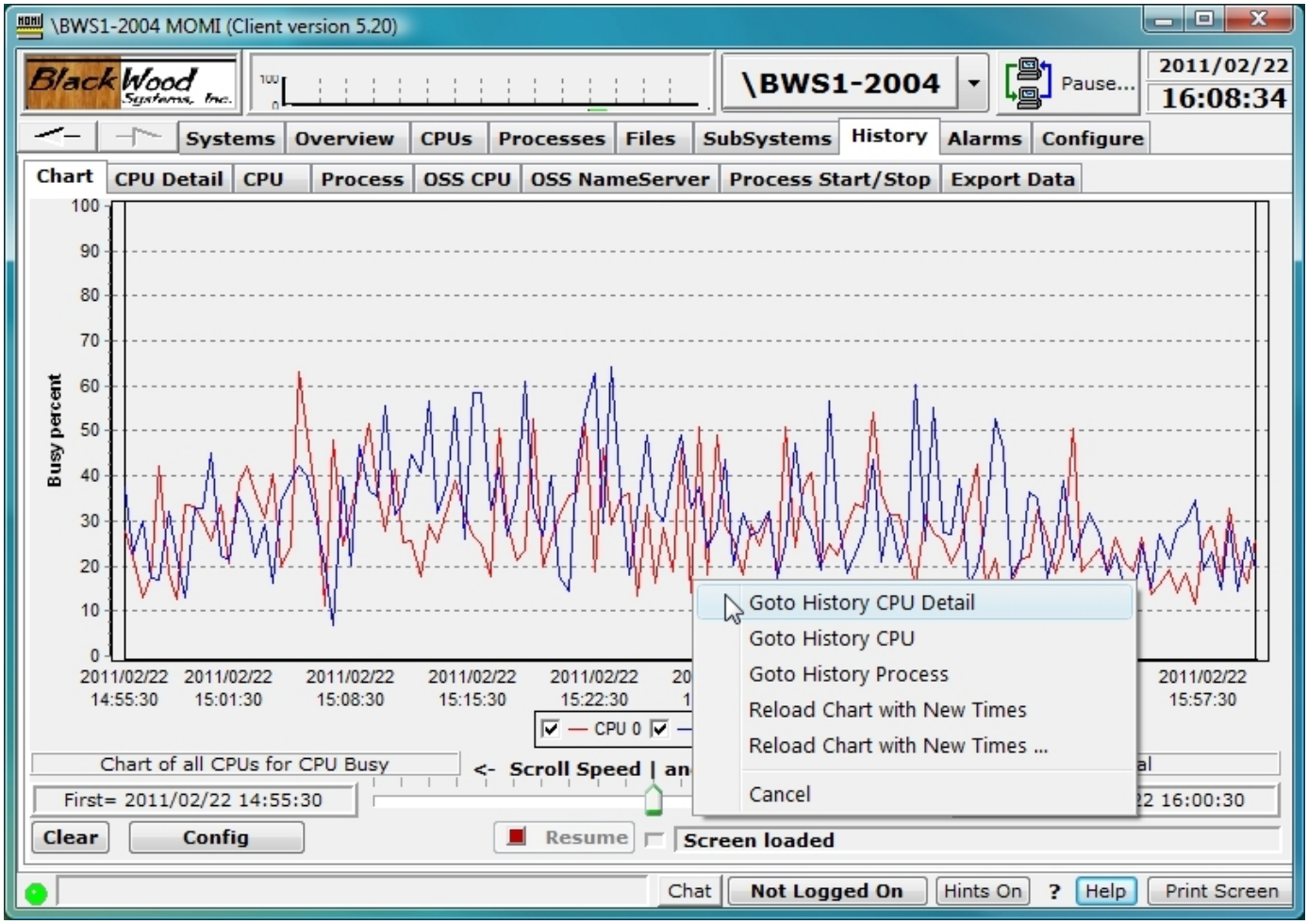


The History / Chart screen graphs various data from the MOMI history files. History capture must be enabled in order for this screen to function.

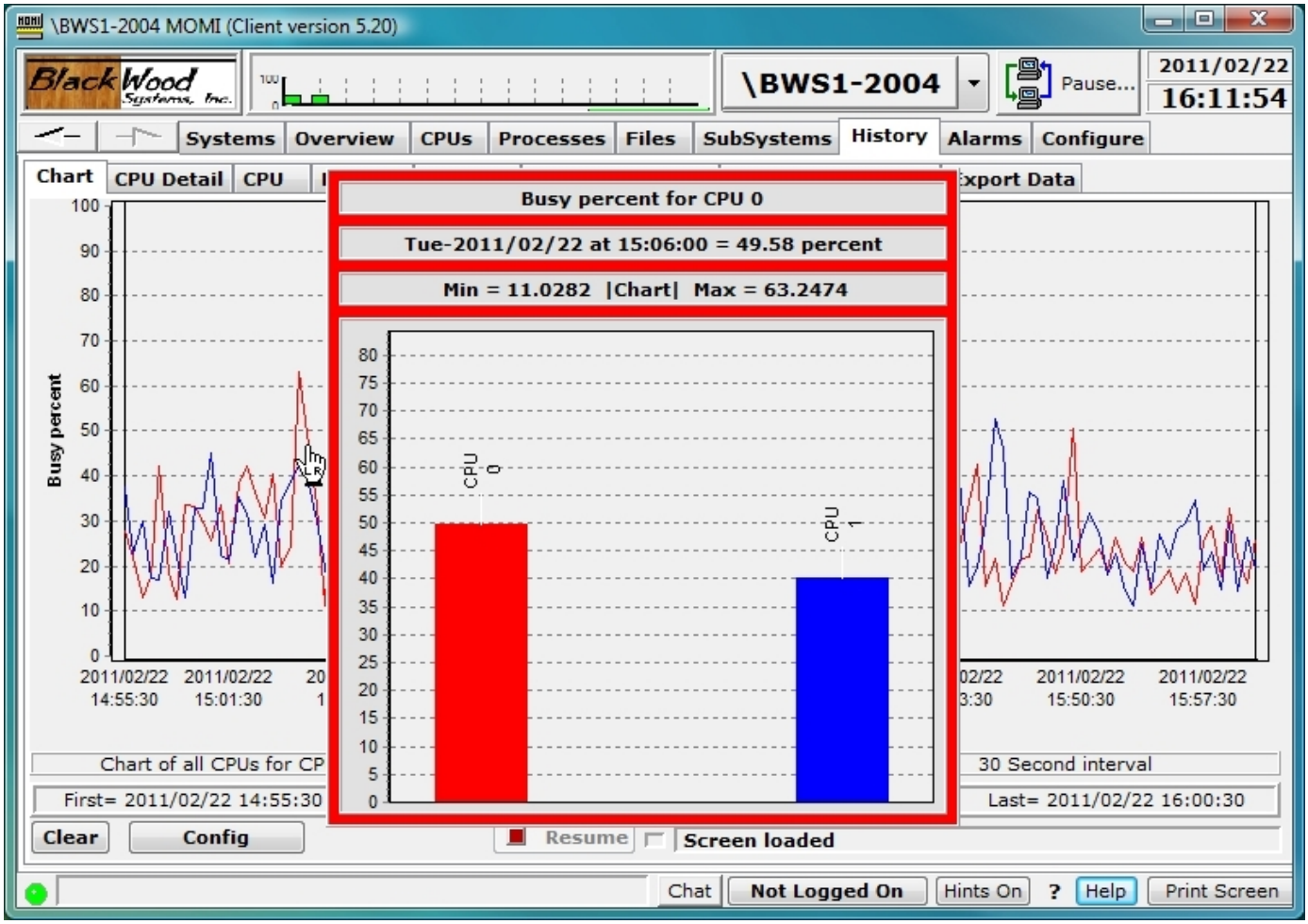
Activity on this screen is started by pressing the **Config** button and selecting a predefined, previously saved or defining data to graph. The pop-up window allows selection of database, start/stop time and item(s) to display. Up to 16 different history items may be selected. When an **Element Type** is selected, the available items appear in the **Element Sub Type** along with other fields such as a CPU number or process name. For example, selecting a Process **Element Type**e requires a process name (or cpu,pin for unnamed processes). The predefined charts **Chart all CPUs for Busy percent** or previously saved user chart are also available.



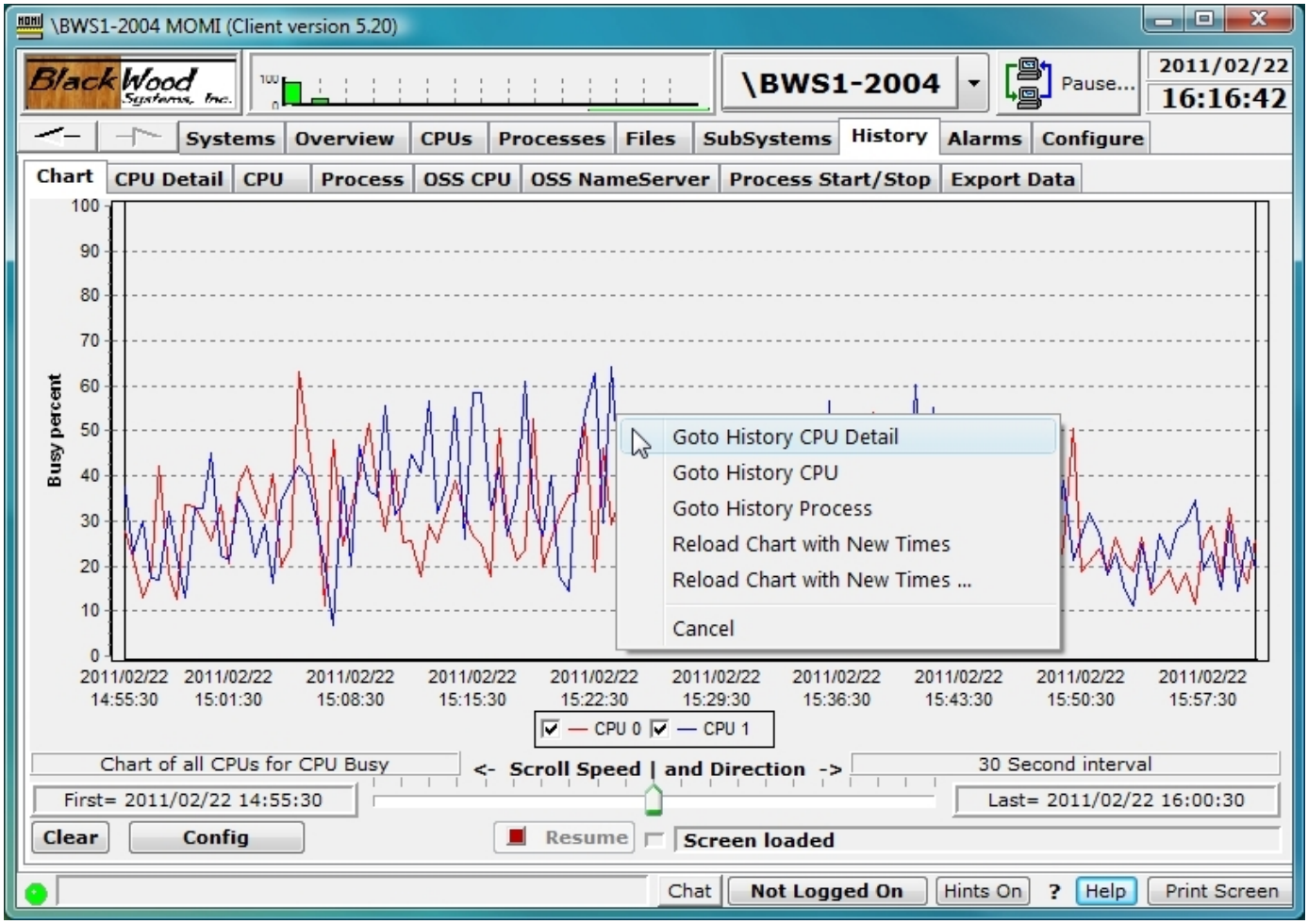
Once a graph is displayed additional information and drill down options are available by placing the mouse over a chart line or point.



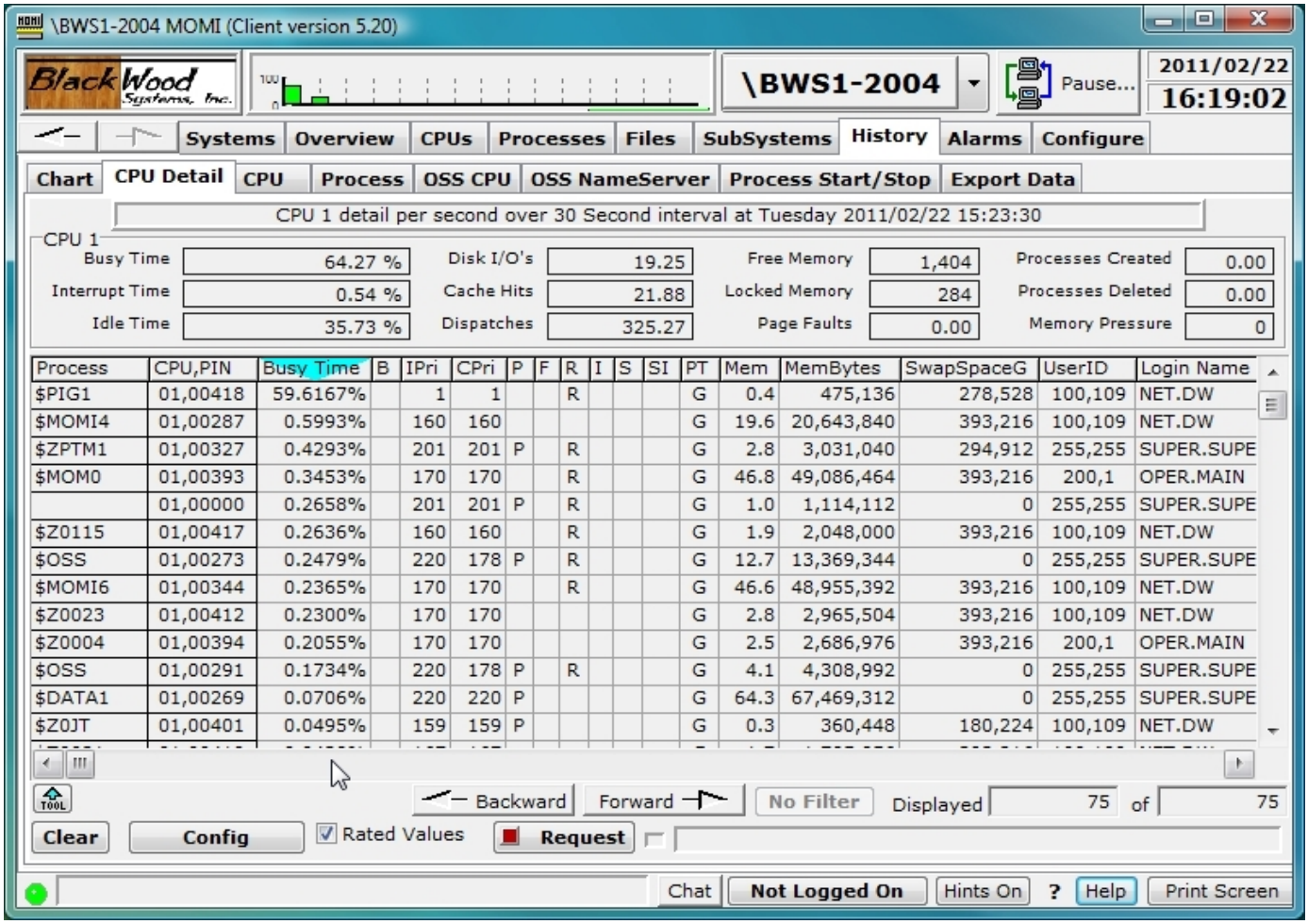
Clicking on a chart line will display the date / time, exact value, minimum, maximum values and other information as available.



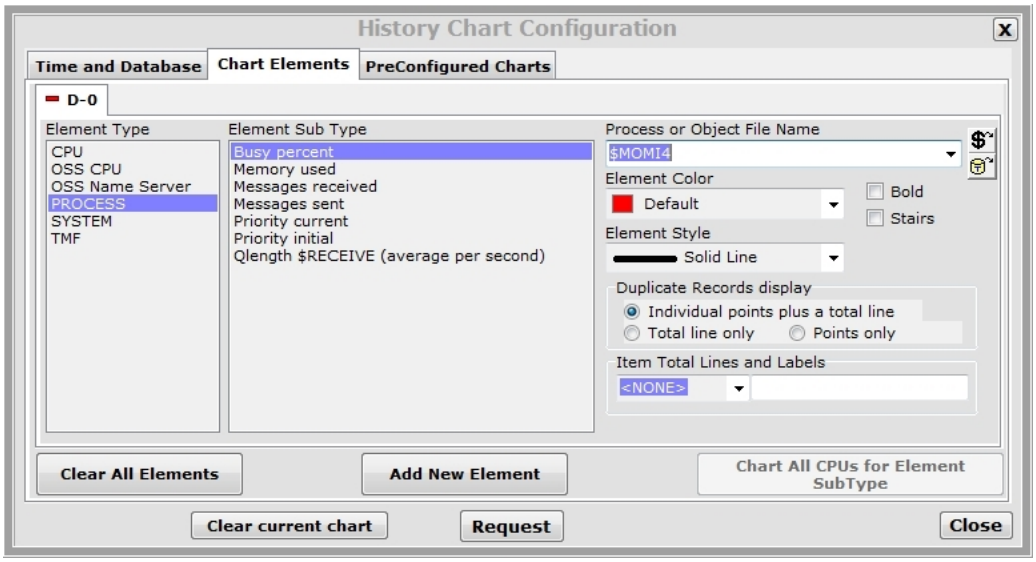
Additional data may also be displayed in some charts. For example, a right-click over a CPU busy line displays menu that allows display of History CPU Detail.



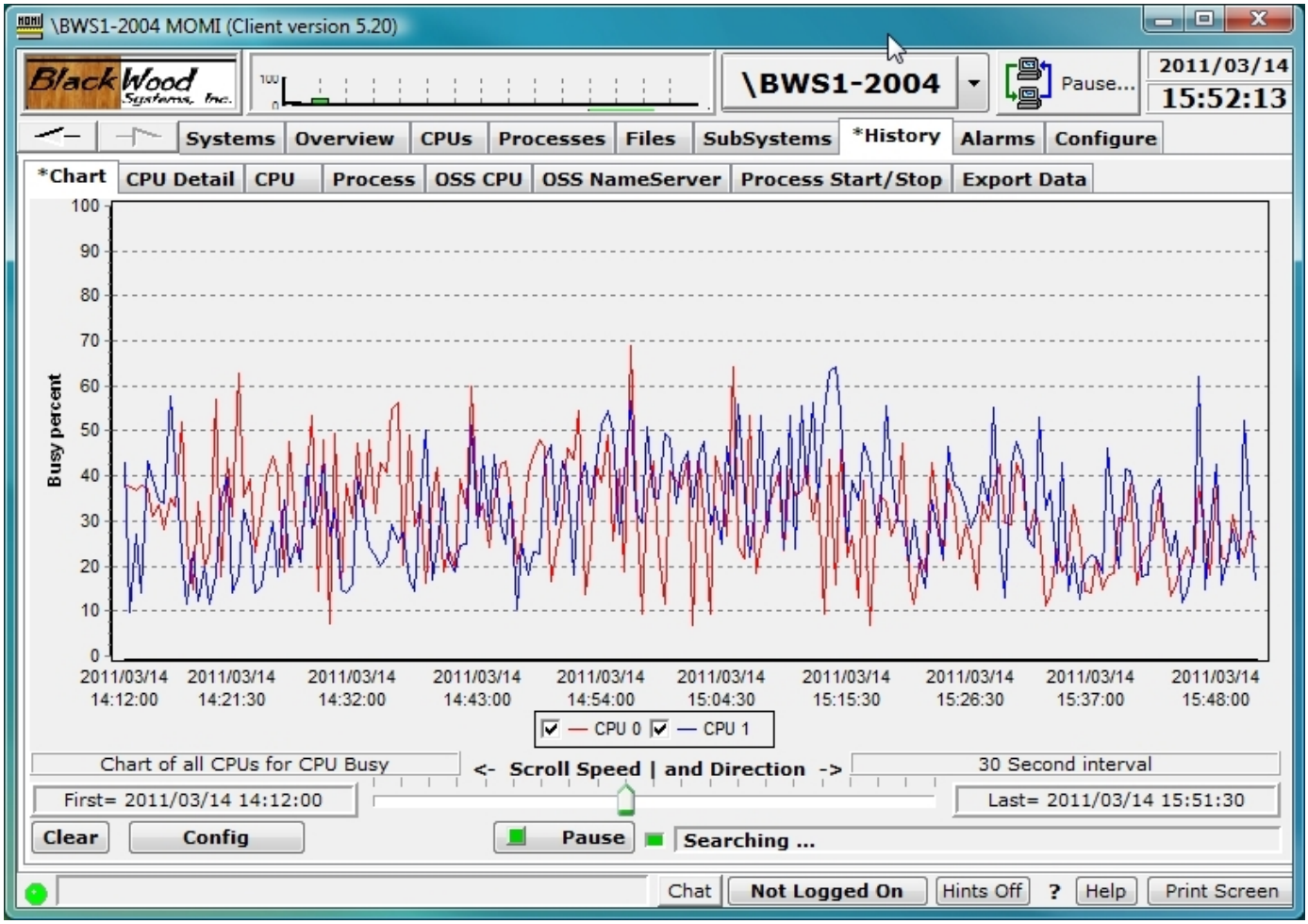
Selecting Goto History CPU Detail jumps to the History / CPU Detail screen for the chart line CPU and processes captured at that point in time. The **Backward** and **Forward** buttons allow stepping through history intervals.



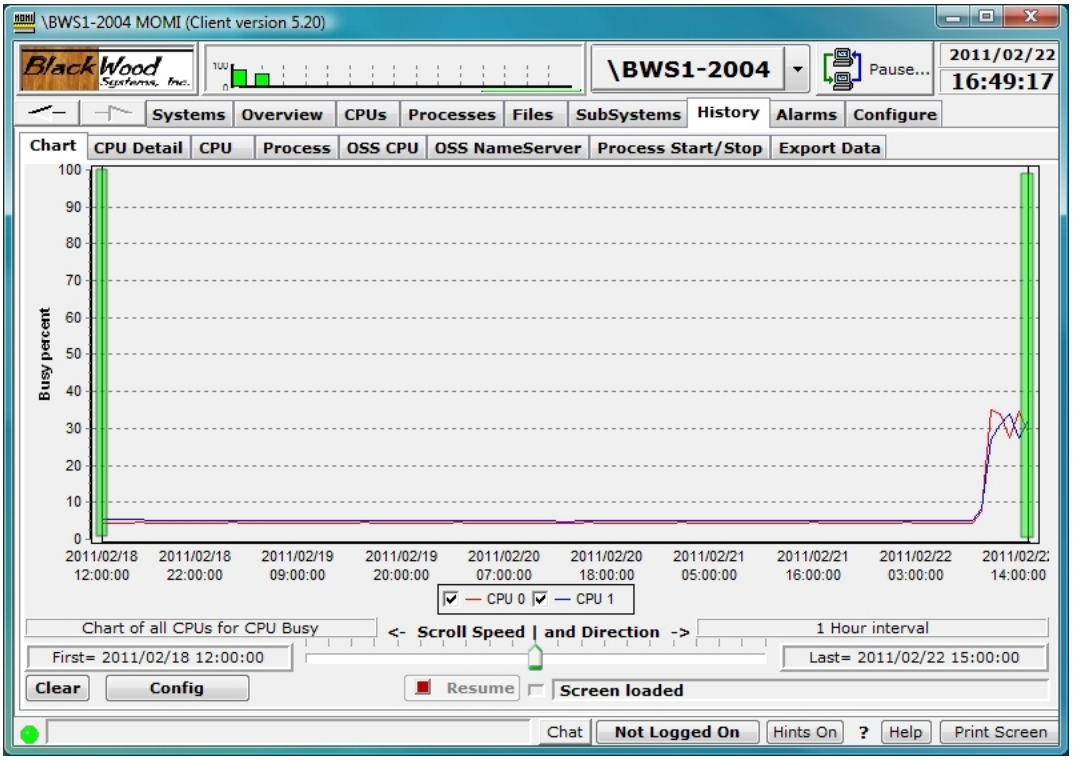
Charts of user defined entities are defined in **Config / Chart Elements**. In the image below, the CPU **Busy percent** for $MOMI4 is selected. Additional chart elements are defined by pressing **Add New Element**.



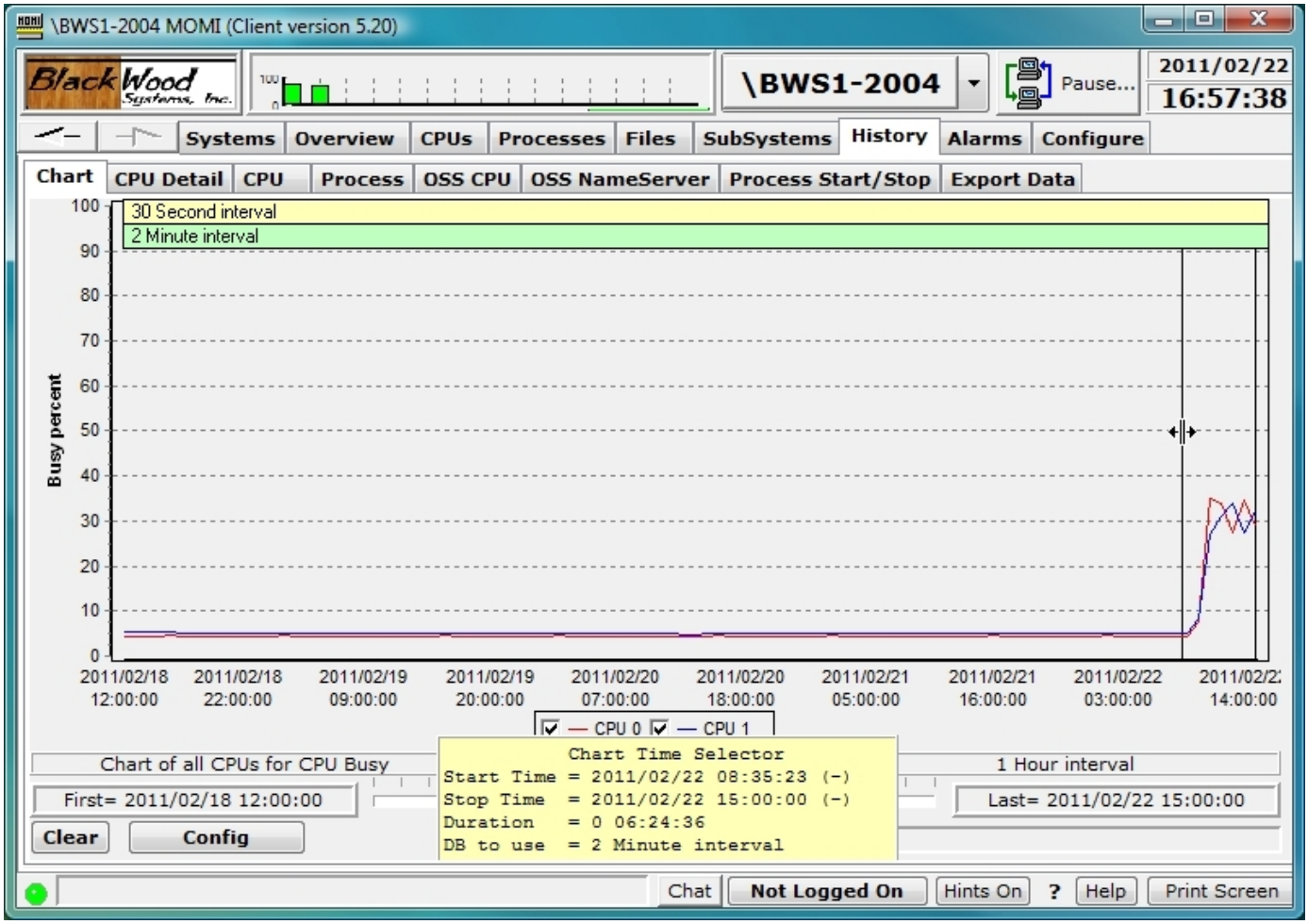
The scrolling feature allows the User to easily scroll forward and backward in time and visually search for unexpected activities. Additionally, when scrolling forward in time, the chart becomes 'live' and displays new data as it arrives.



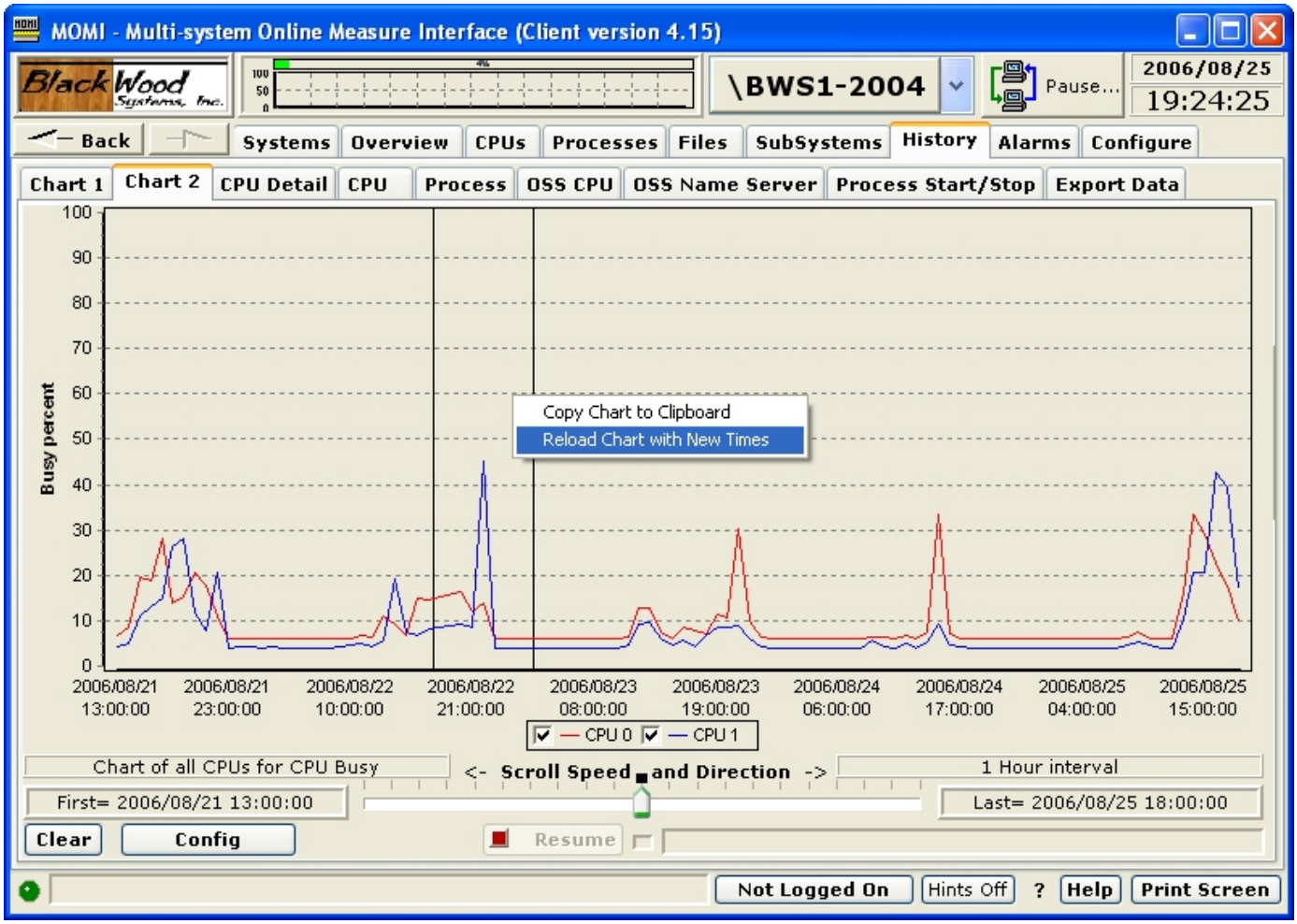
History drill-down is aided by two bars (highlighted in green) on the left and right hand sides of the chart. They provide a graphical means to select a select a window of time based on the current data displayed. In the image below, history at 1 hour is displayed.



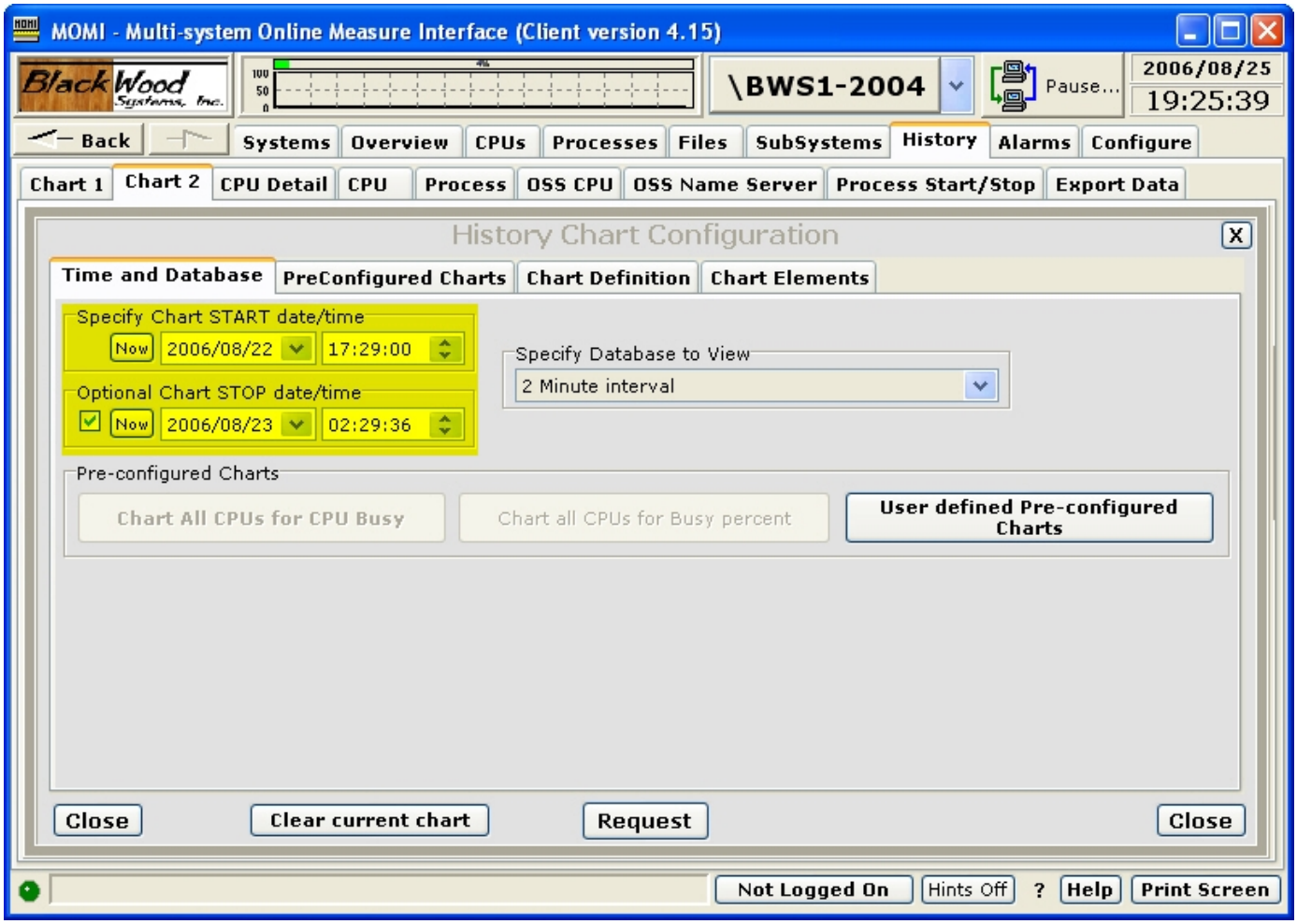
Left-click and hold on a line to move it left and right. A pop-up tool tip appears in the bottom center to show the current time values with the history bar selection. In this example, the scroll bars are moved to just either side of a 'bump' in the history chart graph data.



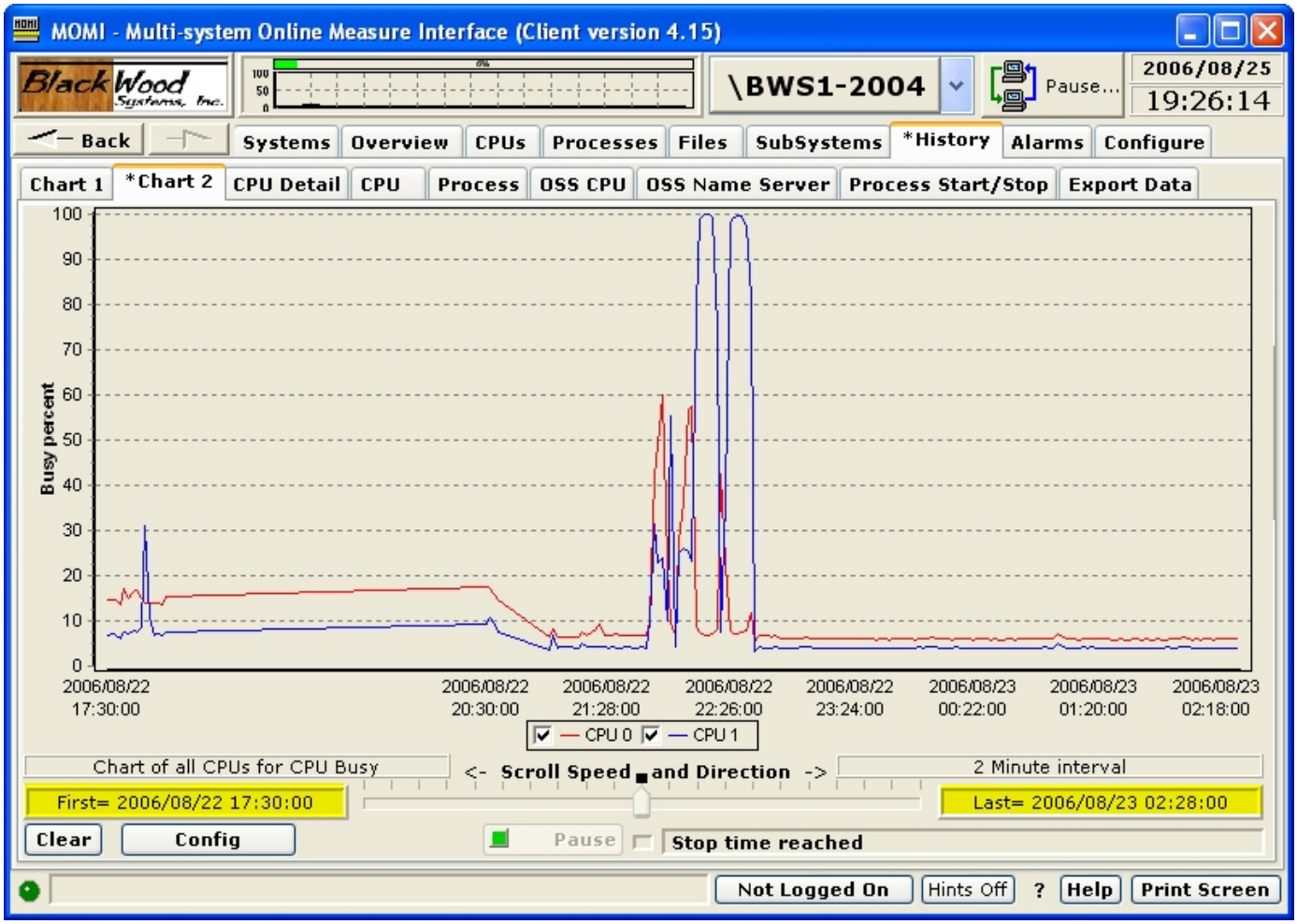
After the bars are positioned, right click on the chart area and select **Reload Chart with New Times**.



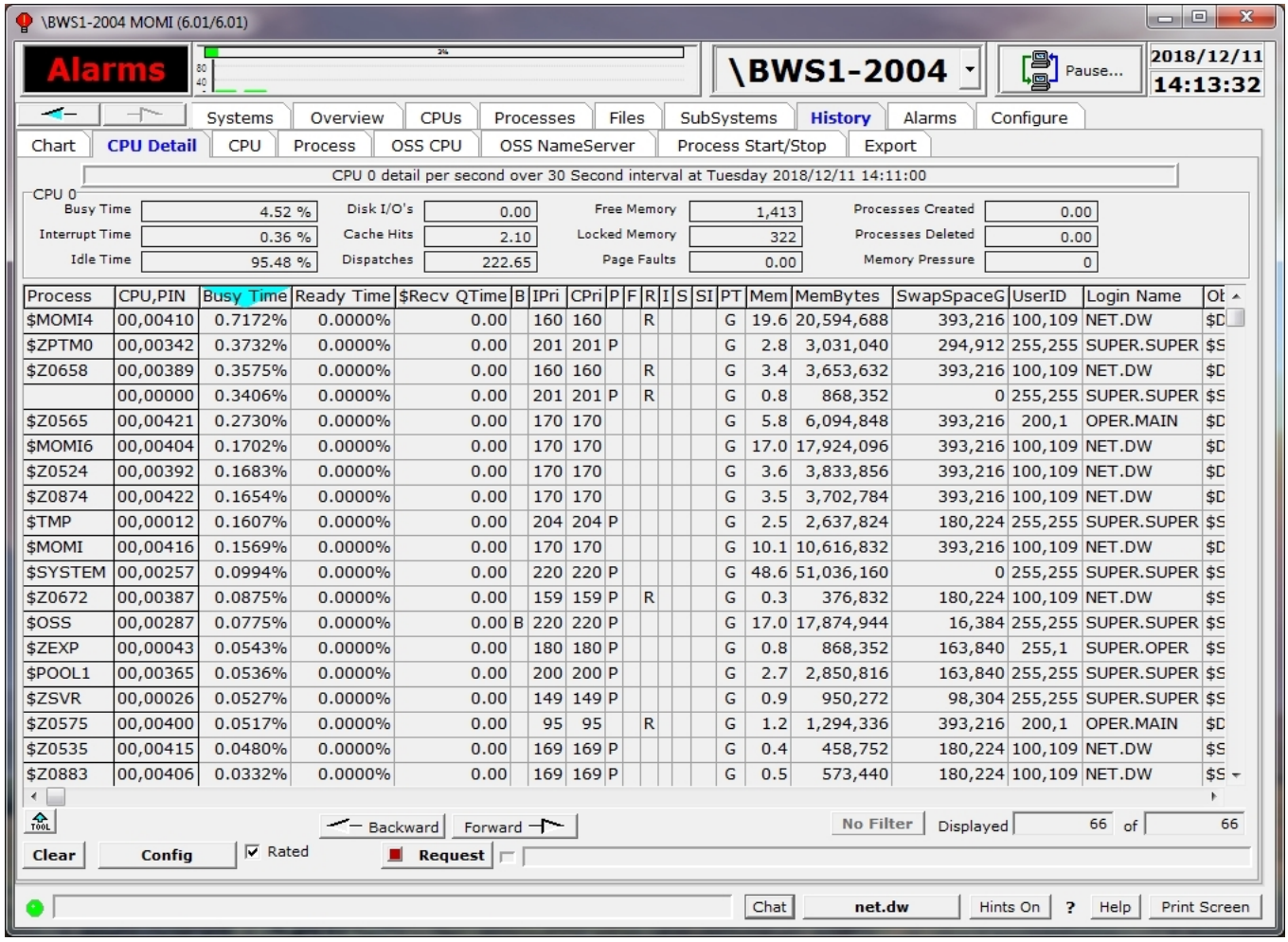
If **Reload Chart with New Times...** is selected, the History Chart Configuration pop-up is displayed. Note that the Chart START and STOP times are preloaded with the values displayed when the history bars were positioned. Additionally, the History database was pre-selected to the next higher resolution. Press **Request** to reload the chart based on the new time values and selected database.



The screen is reloaded with the times selected (perhaps adjusted to the capture interval).

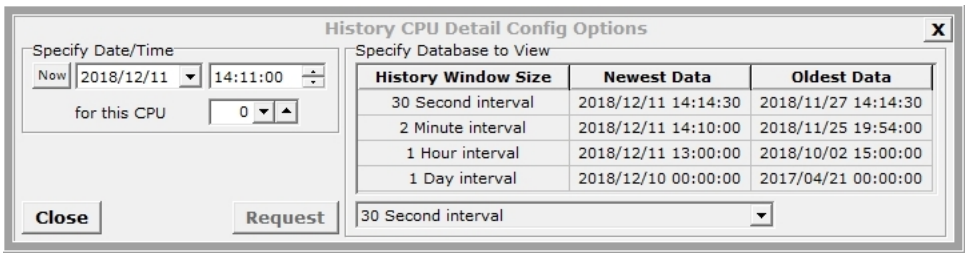

MOMI - Multi-system Online Measure Interface (Client version 4.15)
BlackWood Systems, Inc.
\BWS1-2004
Pause...
2006/08/25
19:26:14
Back
Systems
Overview
CPUs
Processes
Files
SubSystems
*History
Alarms
Configure
Chart 1
*Chart 2
CPU Detail
CPU
Process
OSS CPU
OSS Name Server
Process Start/Stop
Export Data
Busy percent
100
90
80
70
60
50
40
30
20
10
0
2006/08/22 17:30:00
2006/08/22 20:30:00
2006/08/22 21:28:00
2006/08/22 22:26:00
2006/08/22 23:24:00
2006/08/23 00:22:00
2006/08/23 01:20:00
2006/08/23 02:18:00
CPU 0
CPU 1
Chart of all CPUs for CPU Busy
<- Scroll Speed and Direction ->
2 Minute interval
First= 2006/08/22 17:30:00
Last= 2006/08/23 02:28:00
Clear
Config
Pause
Stop time reached
Not Logged On
Hints Off
?
Help
Print Screen

History / CPU Detail

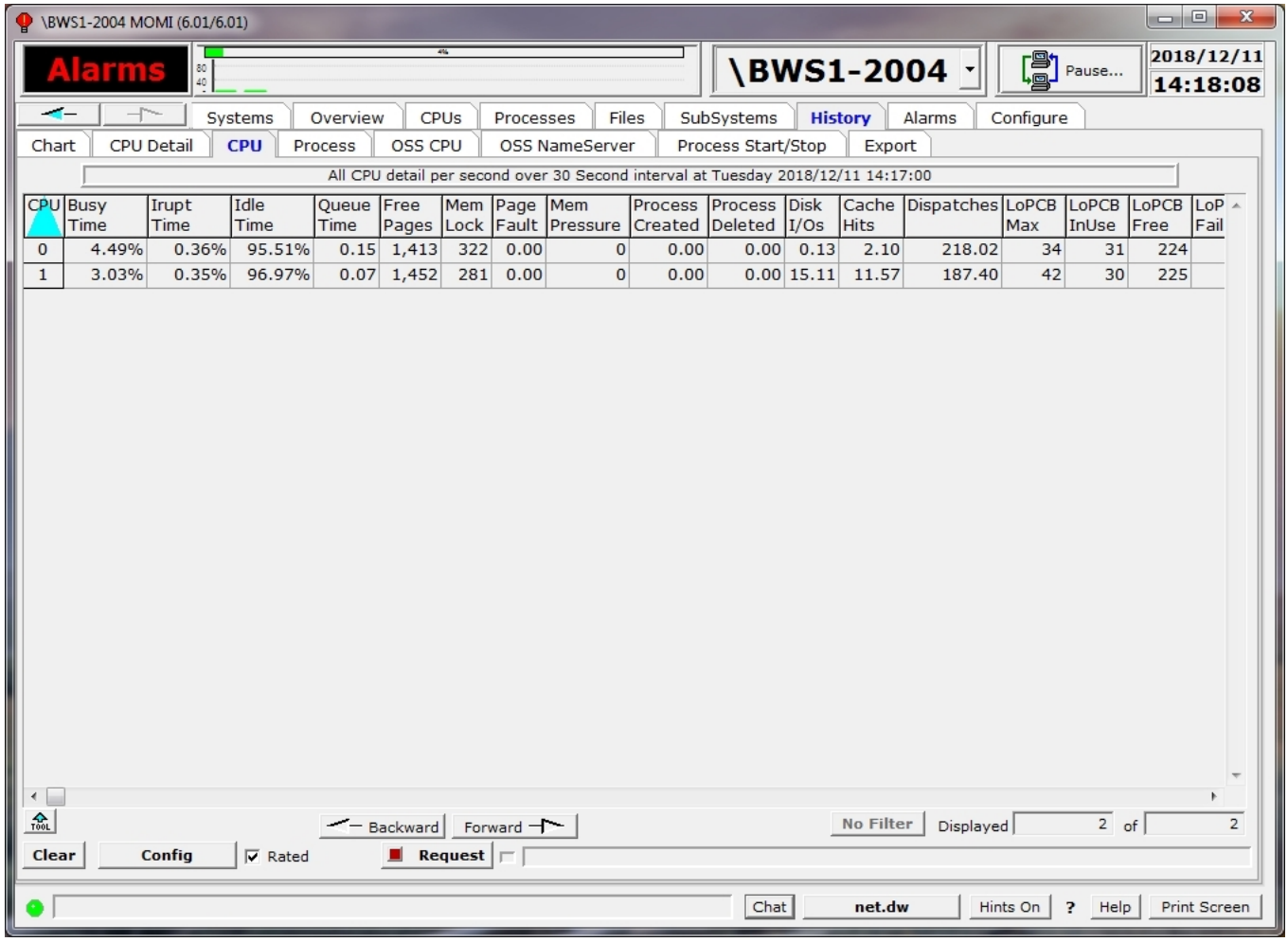


The History / CPU Detail screen presents details of a CPU and the processes in that CPU at a given point in time. Once data is displayed, scrolling forward and backward based on interval in time may be performed.

The Config button displays a pop-up window allowing the specific Date/Time, CPU and Database to retrieve the data.

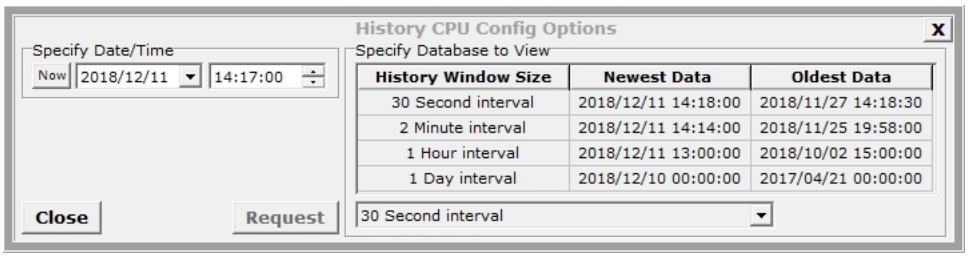


History / CPU

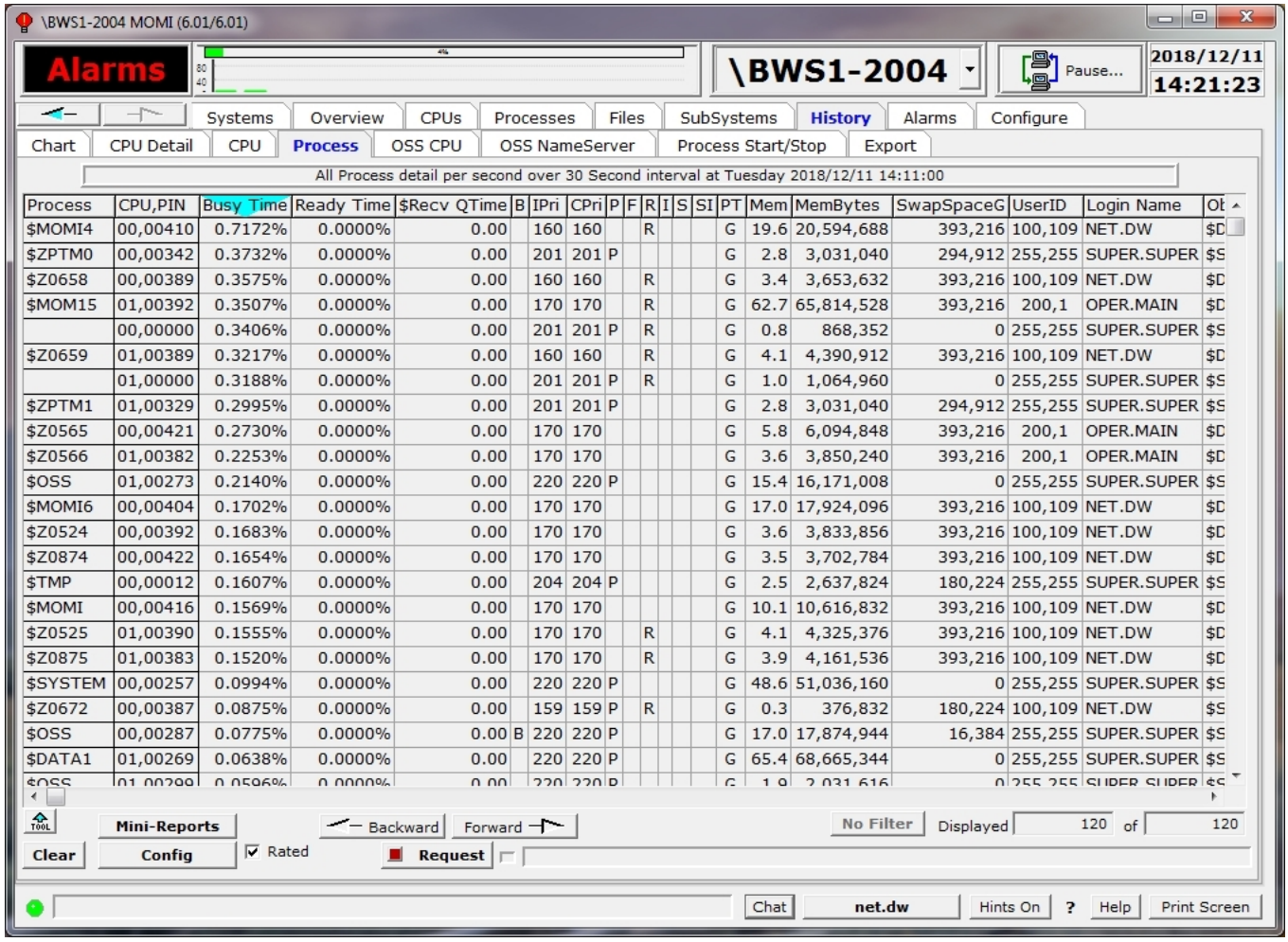


The History / CPU screen presents all CPUs at a given point in time. Once data is displayed, scrolling forward and backward based on interval in time may be performed.

The Config button displays a pop-up window allowing the specific Date/Time and Database to retrieve the data.

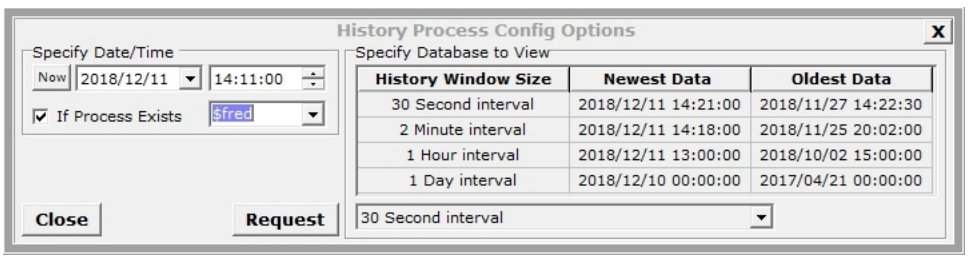


History / Process

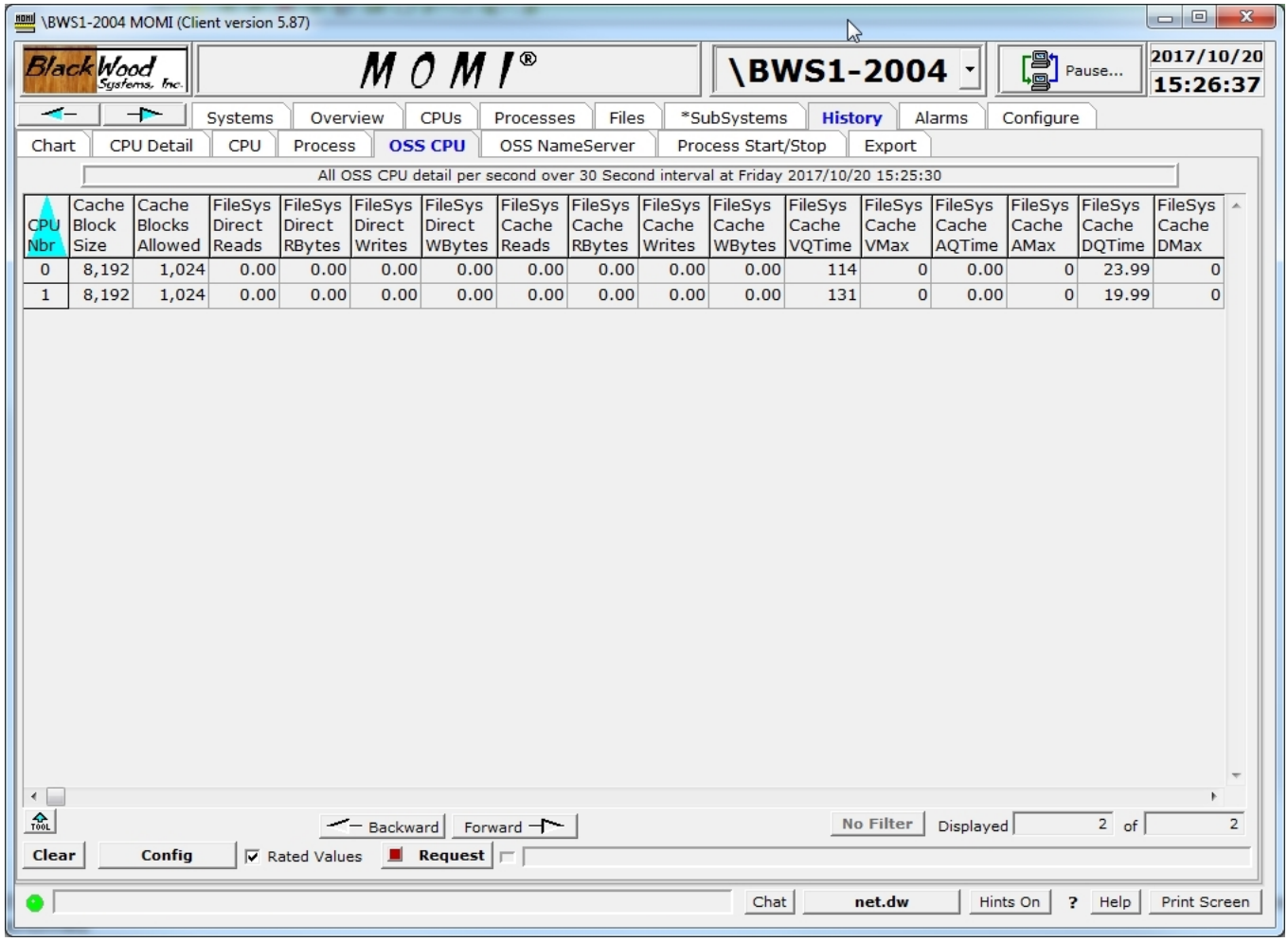


The History / Process screen presents all processes at a given point in time. Once data is displayed, scrolling forward and backward based on interval in time may be performed.

The Config button displays a pop-up window allowing the specific Date/Time and Database to retrieve the data. A search by process feature is also present to limit the returned data to a single process. This is enabled by checking **If Process Exists**.

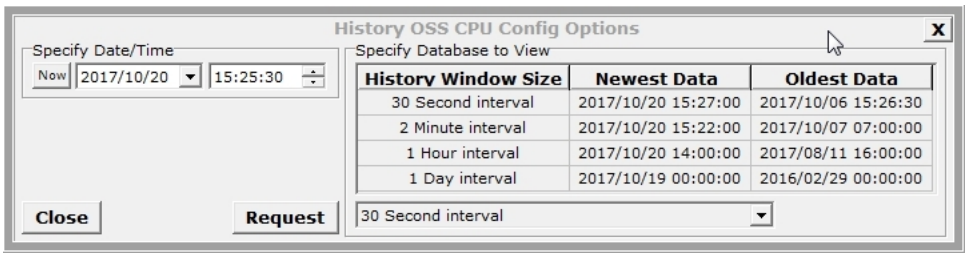


History / OSS CPU

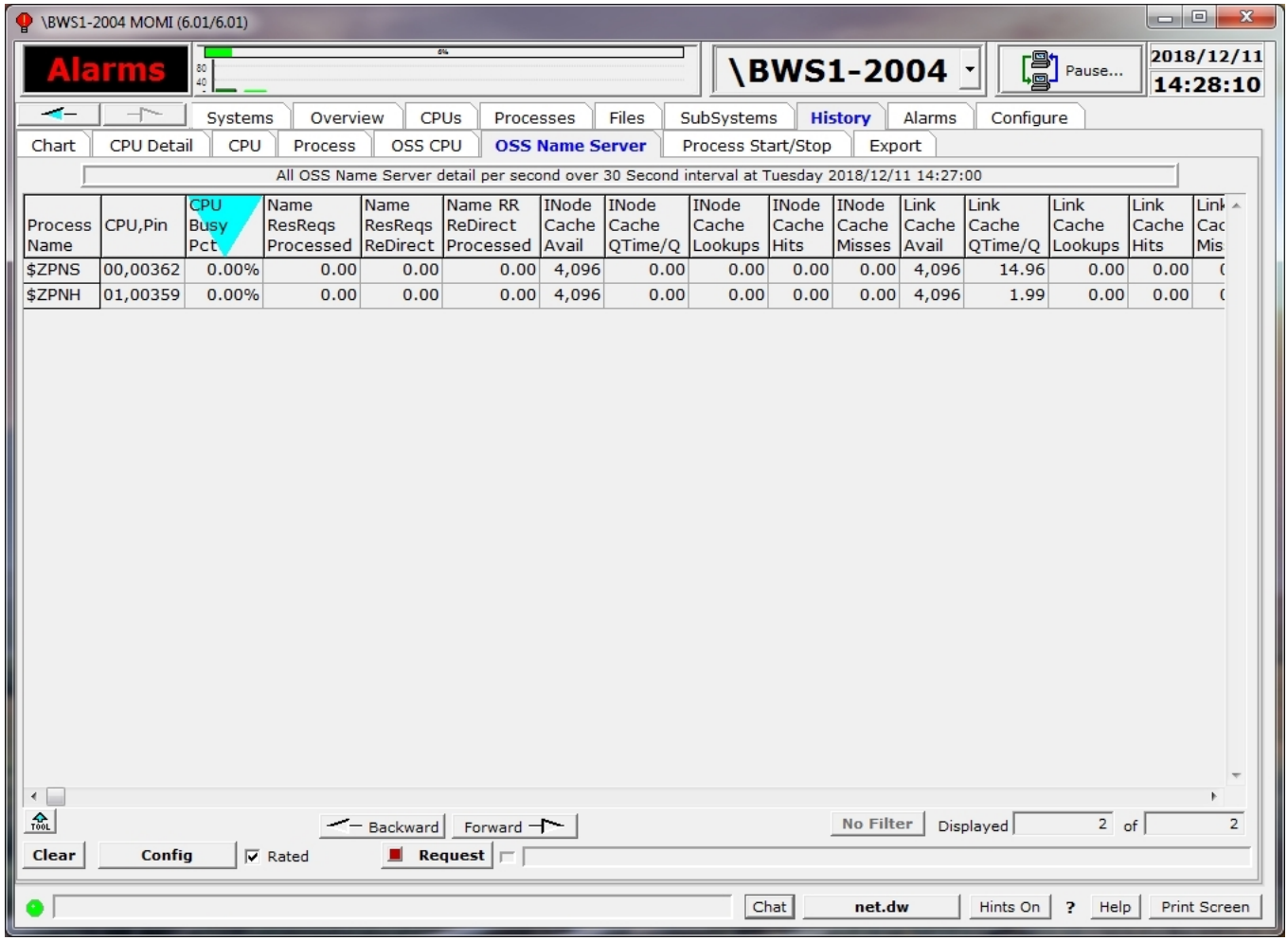


The History / OSS CPU screen presents the MEASURE OSS CPU entity for all processors at a given point in time. Once data is displayed, scrolling forward and backward based on interval in time may be performed.

The **Config** button displays a pop-up window allowing the specific Date/Time and Database to retrieve data.



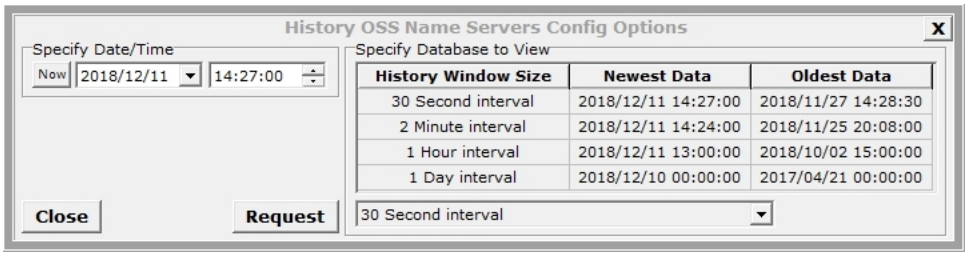
History / OSS Name Server



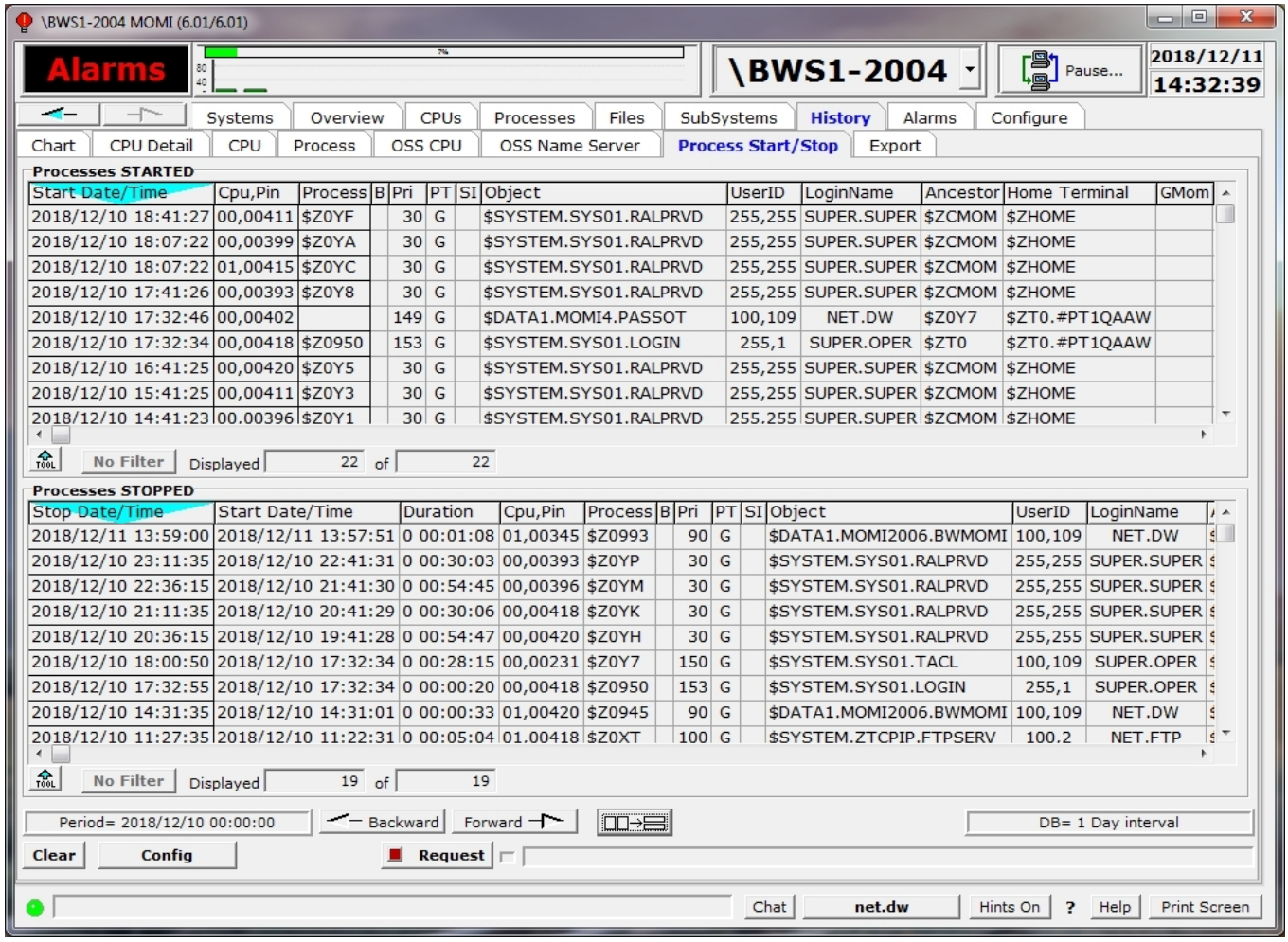
The History / OSS Name Server screen presents the MEASURE OSS Name Server entity for all name servers running on the System. Once data is displayed, scrolling forward and backward based on interval in time may be performed.

Name Servers running, but unused, may not have MEASURE records reported.

The Config button displays a pop-up window allowing the specific Date/Time and Database to retrieve the data.



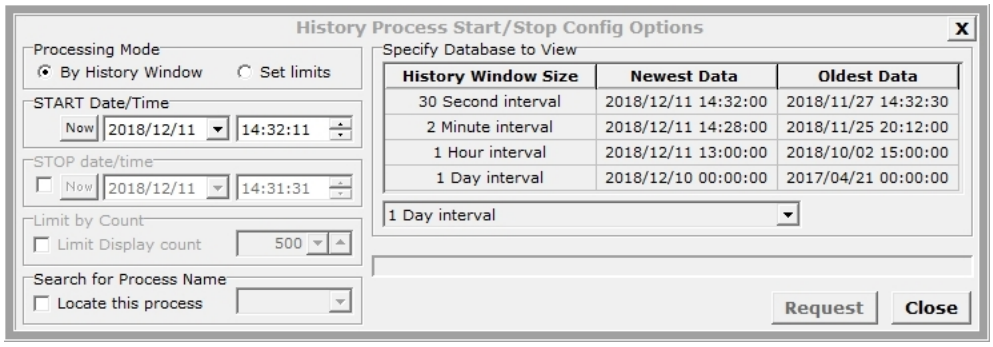
History / Process Start/Stop



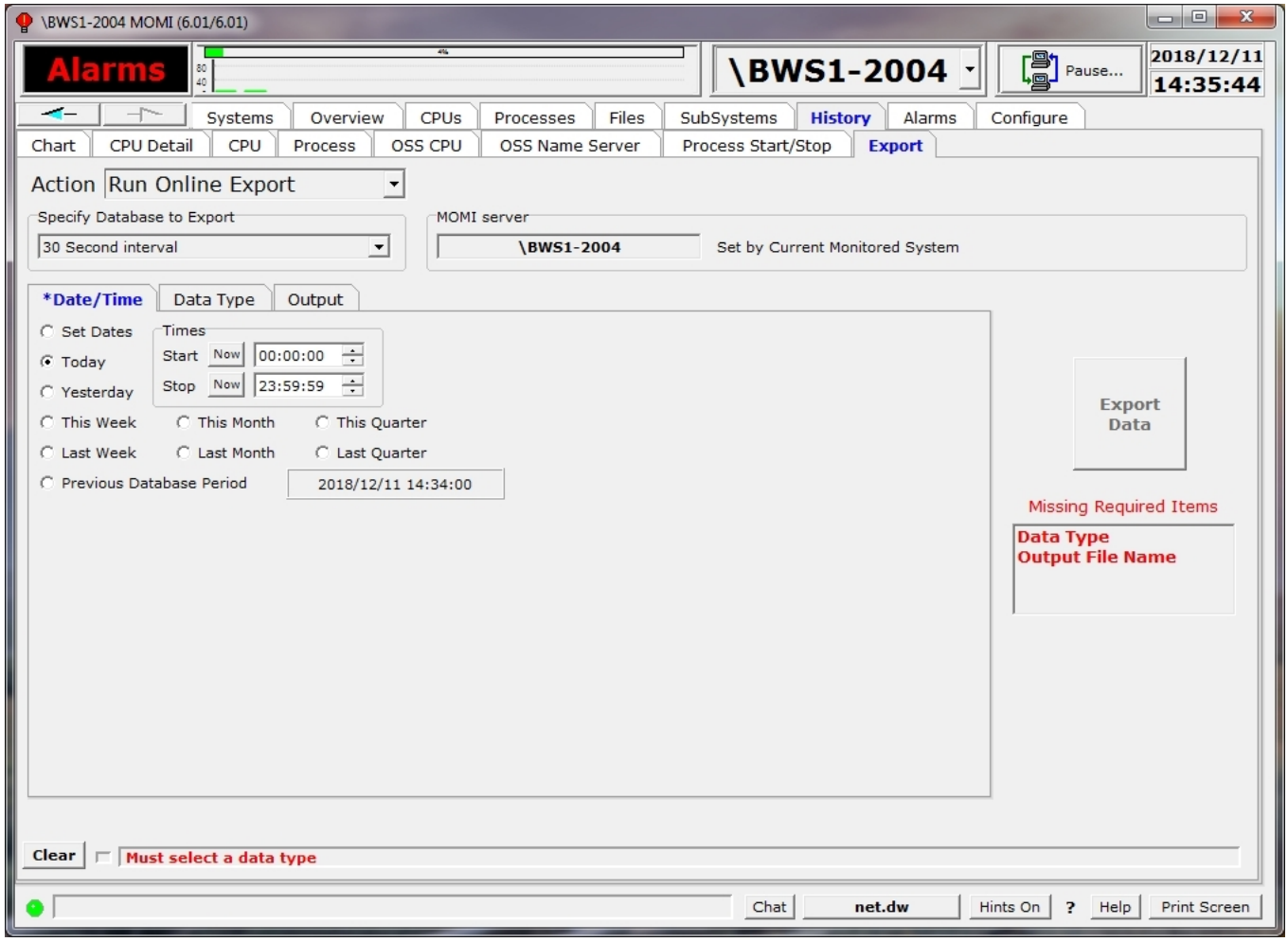
The History / Process Start/Stop screen presents processes started and stopped from a given point in time. Once data is displayed, scrolling forward and backward based on interval in time may be performed.

The screen is similar to the Processes / Last Start/Stop screen except that the underlying information is derived from the MOMI history file. Searching for start/stop records through MOMI history may take some time.

The Config button displays a pop-up window allowing the specific Date/Time and Database to retrieve the data.


History Process Start/Stop Config Options
Processing Mode
By History Window
Set limits
START Date/Time
Now
2018/12/11
14:32:11
STOP date/time
Now
2018/12/11
14:31:31
Limit by Count
Limit Display count
500
Search for Process Name
Locate this process
Specify Database to View
History Window Size
Newest Data
Oldest Data
30 Second interval
2018/12/11 14:32:00
2018/11/27 14:32:30
2 Minute interval
2018/12/11 14:28:00
2018/11/25 20:12:00
1 Hour interval
2018/12/11 13:00:00
2018/10/02 15:00:00
1 Day interval
2018/12/10 00:00:00
2017/04/21 00:00:00
1 Day interval
Request
Close

History / Export Data



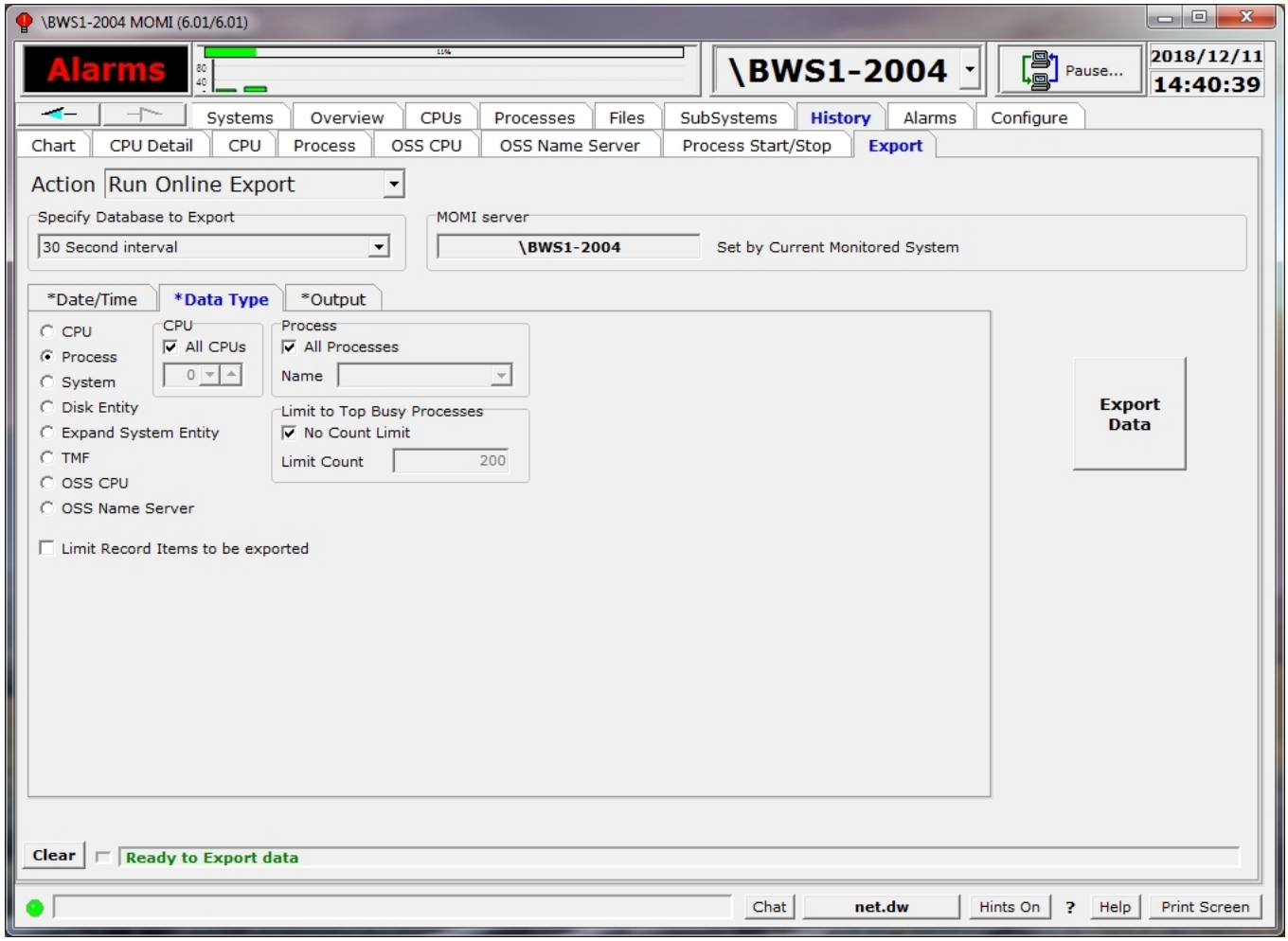
The History / Export Data screen allows the creation of raw history of either delimited separated fields or HTML. The delimiter character, TAB by default, may be changed on Configure / Client / General.

The export function provides a means to obtain raw MOMI data so that the User may generate desired reports or otherwise format data to meet their requirements. The column order of data exported is fixed, however the columns exported are selectable. New information provided with an entity is generally placed on the end of the export line.

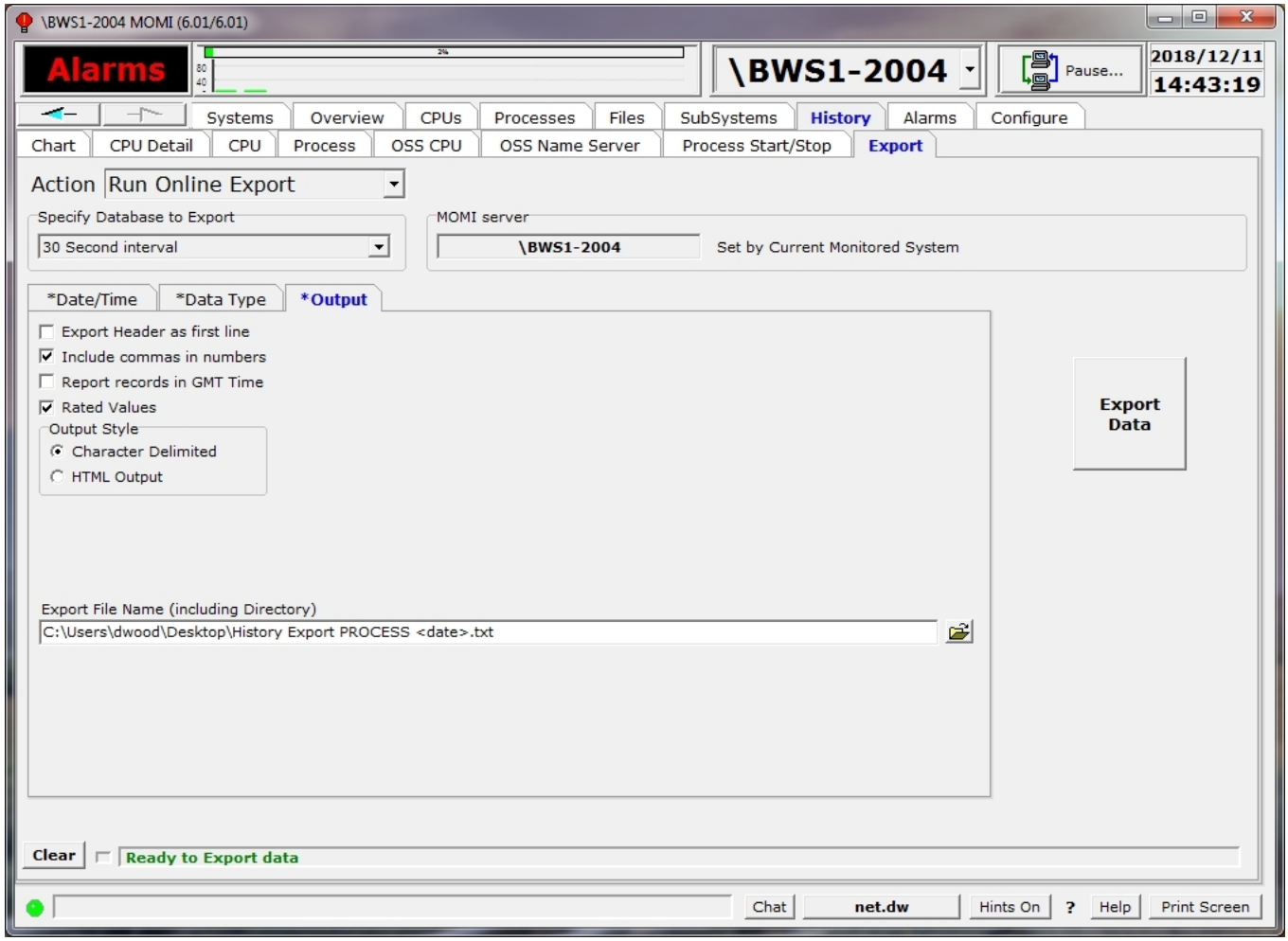
History export may return a great deal of data. A fast connection to the NonStop System is recommended.

The **Action** drop down determines if the export is immediate (Run Online Report), or a saved script (Create Script Export), or when a client is launched (Create Runtime export).

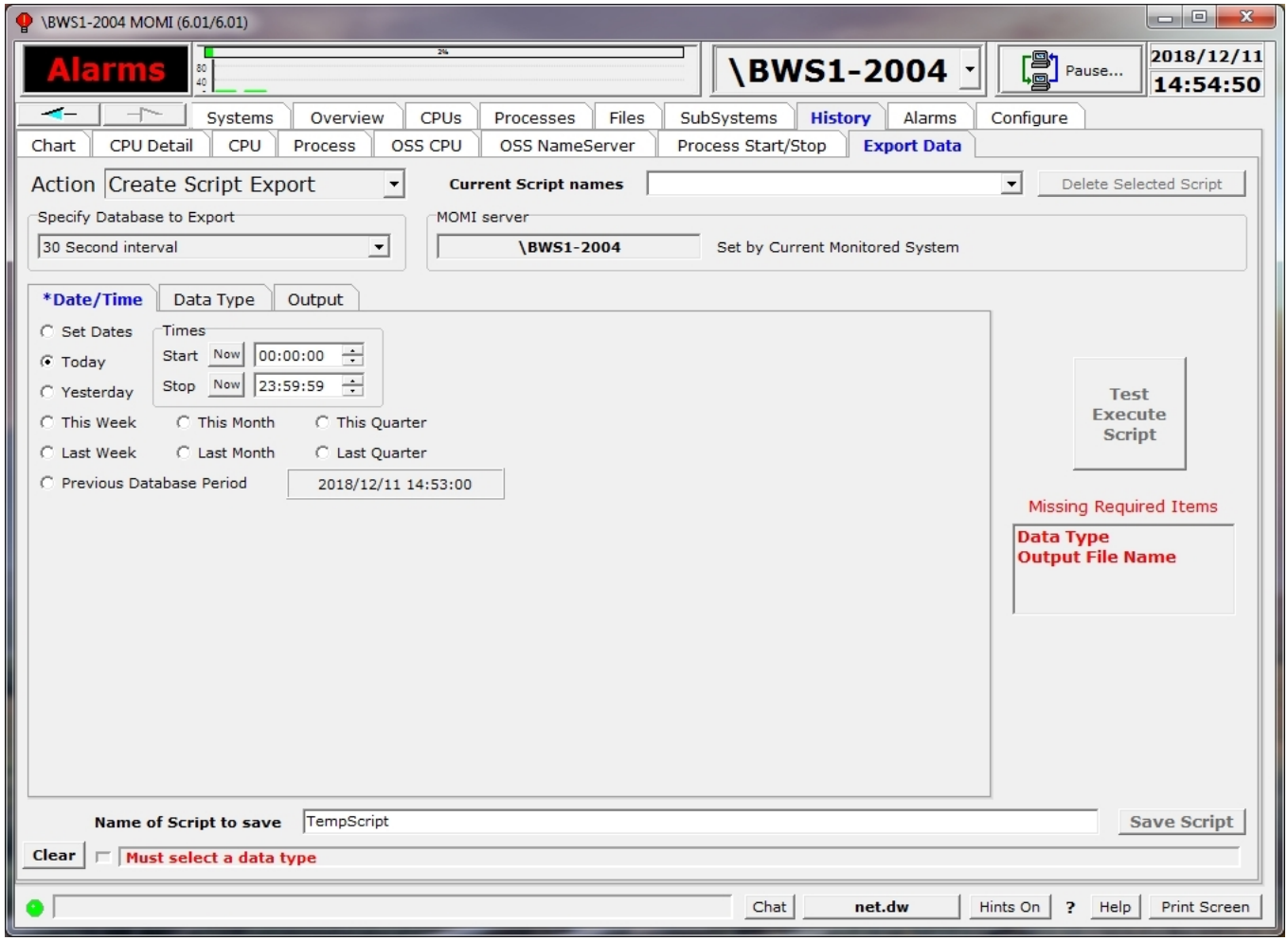
In all Actions, three tabs (Date/Time, Data Type, Output) are used to configure the export. The **Date/Time** tab (shown above) is used to select the time range and the database.



The **Data Type** tab determines the entity type of data to select. Some selections provide limits such as CPU or a process name.



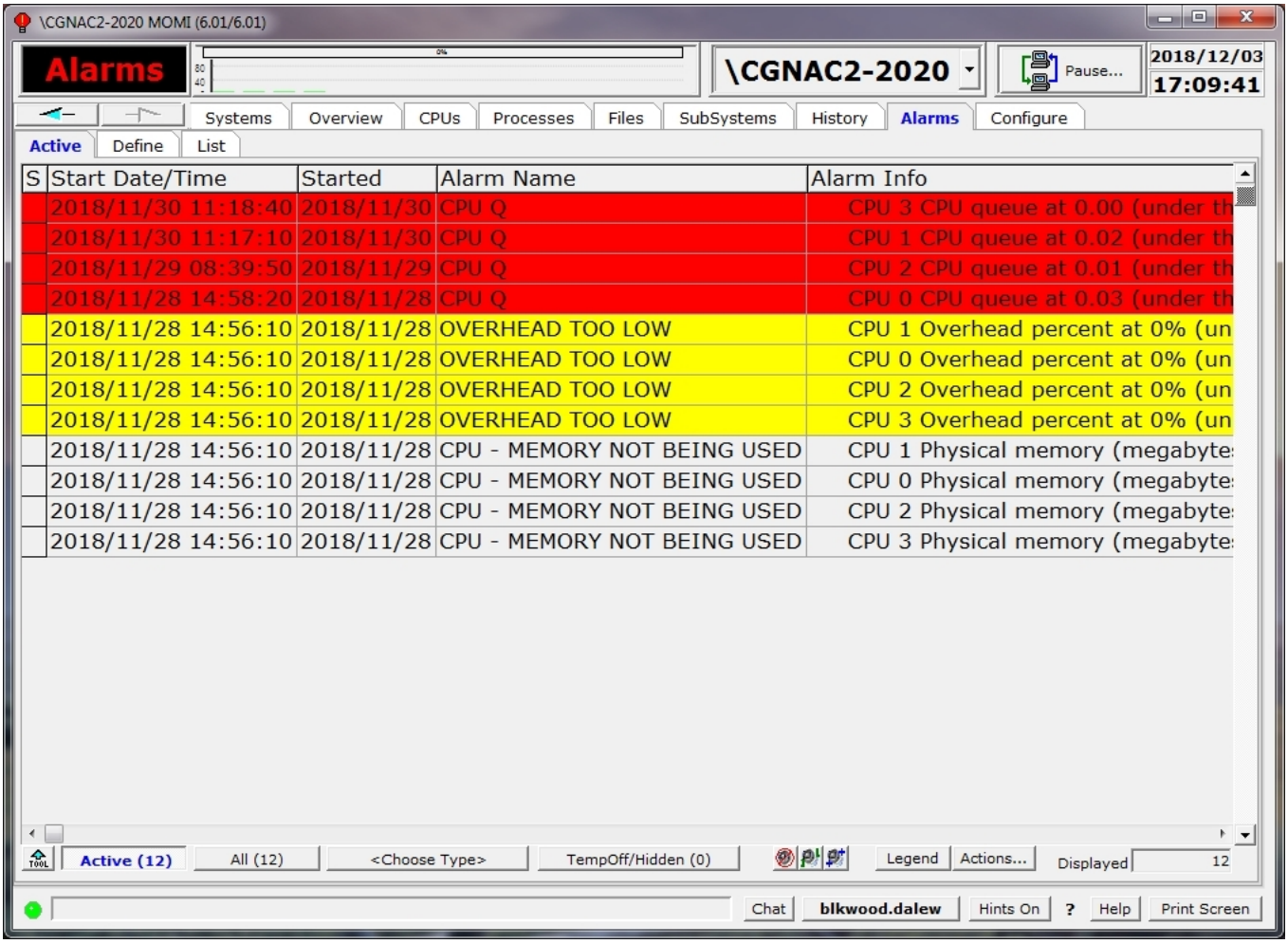
The **Output** tab sets options such as headers, GMT time (LCT is the default), rated, style and the output file name.



An Export Script may be created to provide a means to generate export based on predefined settings. The **Action** of **Create Script Export** allows settings to be saved in the MOMI.INI file under a name and easily reused. The **Action** of **Create Runtime Export** displays the required settings as a series of runtime parameters to that may be given to the MOMI PC Client at start-up to launch the desired export. All other aspects of configuration under the **Date/time**, **Data type** and **Output** tabs function as described previously.

Alarms

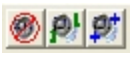
Alarms / Active



The Alarms / Active screen presents a list of alarms currently triggered or have met the required threshold criteria.

Alarms are initially create on the Alarm / Define screen. If the defined threshold is reached, and an on-screen action is requested, the alarm appears on this screen.

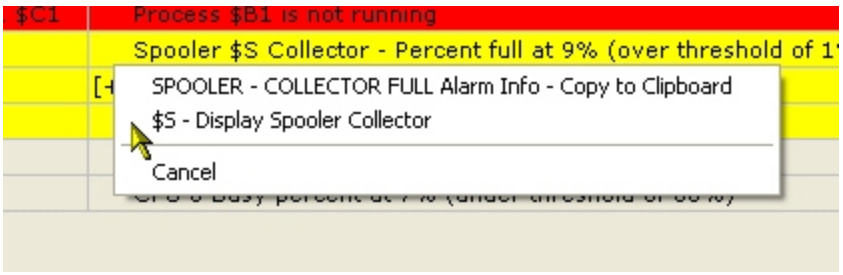
The buttons **Active**, **All**, **Only** and **TempOff/Hidden** control the information displayed in the main portion of the screen. The **Active** button displays only alarms that are triggered or have met their criteria. The **All** button displays every active alarm. The **Only** button displays a particular selected type of alarm after right-clicking over the button and choosing an alarm type.

 These image buttons control sounds associated with alarms. In order from left to right, they stop current sounds, replay last sound and replay all sounds.

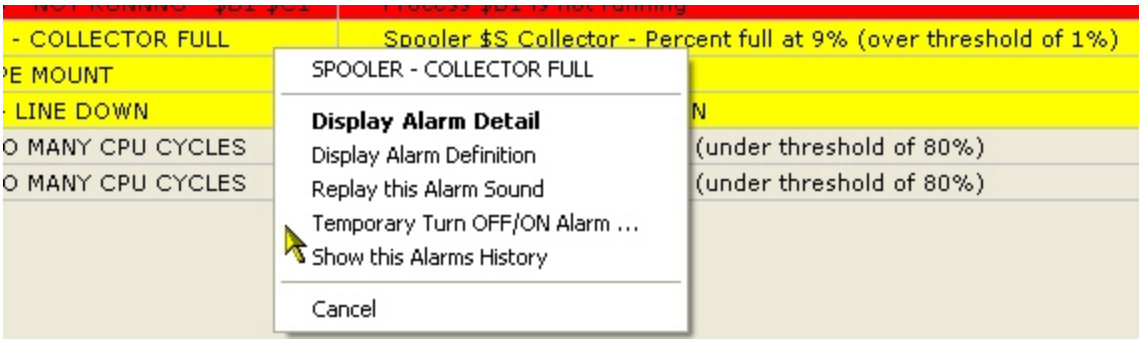
The **Legend** button displays a description of the colors used on screen. The primary alarm colors used are normal, yellow, red and purple (or really red) in order of least to greatest concern. Purple

alarms are reserved for internal MOMI use to alert the user that an alarm configuration error was detected or to many alarms were generated (see ALARMS-MAXIMUM-PER-DEFINITION).

Right-clicking over the Alarm Info provides a jump, if one is available, to a MOMI screen that applies to the Alarm. In the image below, jump for a Spooler Collector percent full is to the screen Subsystems / Spool / Collector.



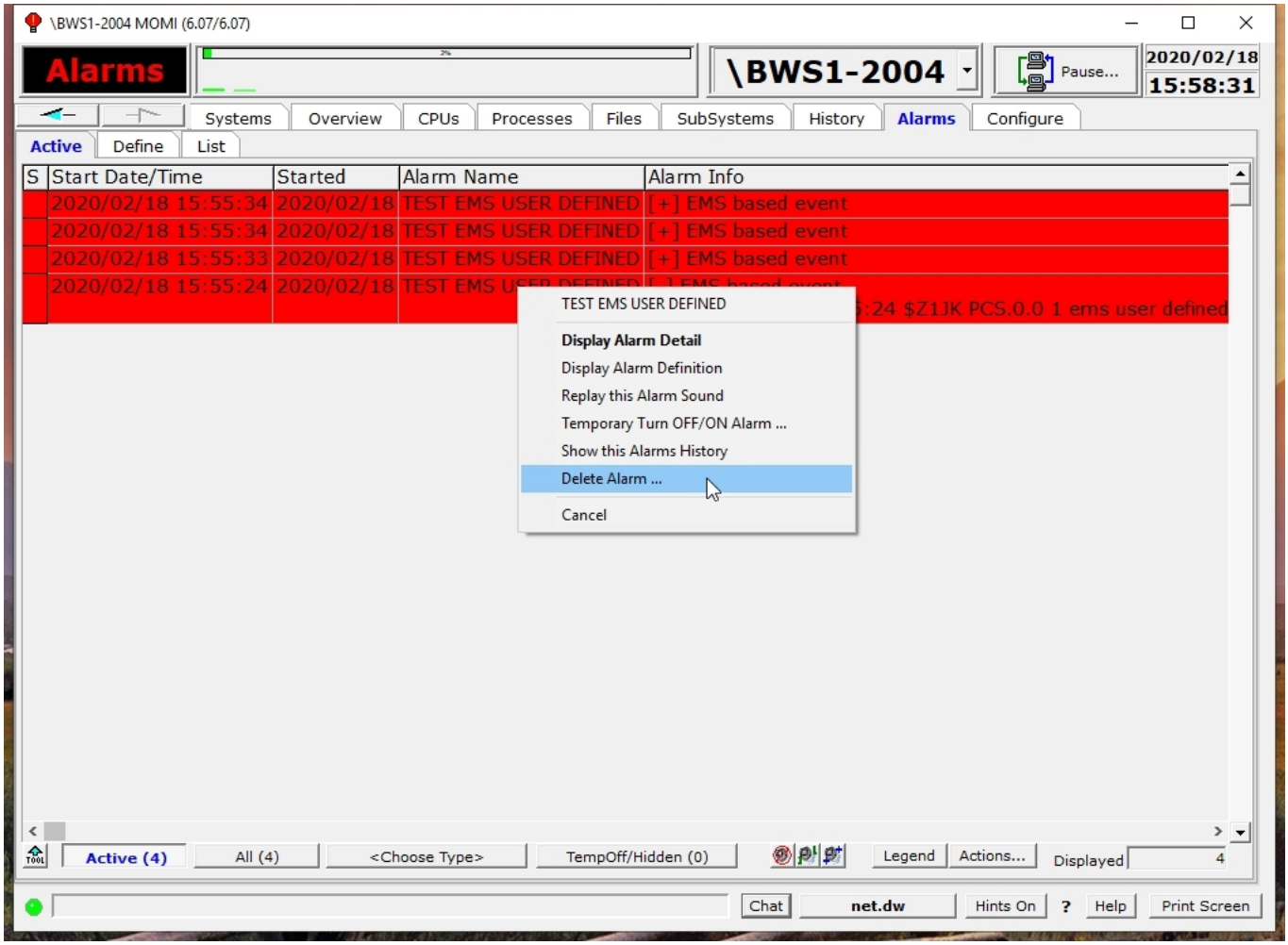
Right-clicking over the Alarm Name to display a list of options.



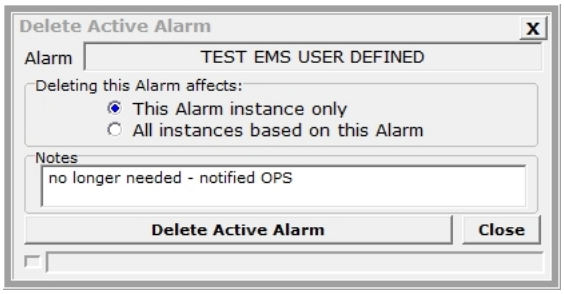
Jumping to the Define / Alarm screen allows the user to quickly access the threshold for an alarm.

Jumping to **Show this Alarms History** (i.e. LogFile) screen displays creates a temporary filter on this screen to show only the items associated with the selected alarm. LogFile records are only generated if the action is configured to do so.

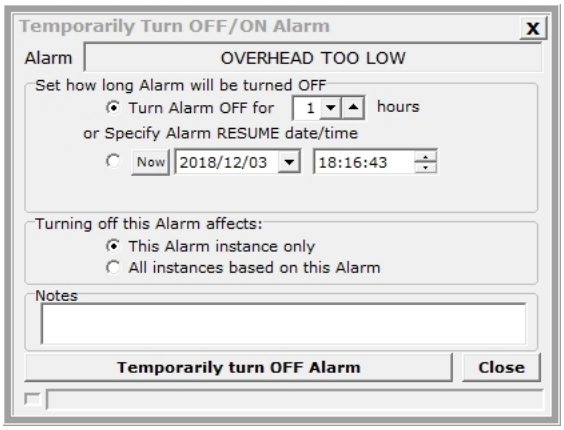
MOMI alarms display until either the situation that caused the alarm is corrected, the alarm is temporarily turned off, or the alarm is deleted. The alarm delete function is only available for **EMS / User defined** alarms. Right click on the alarm to access the delete option.



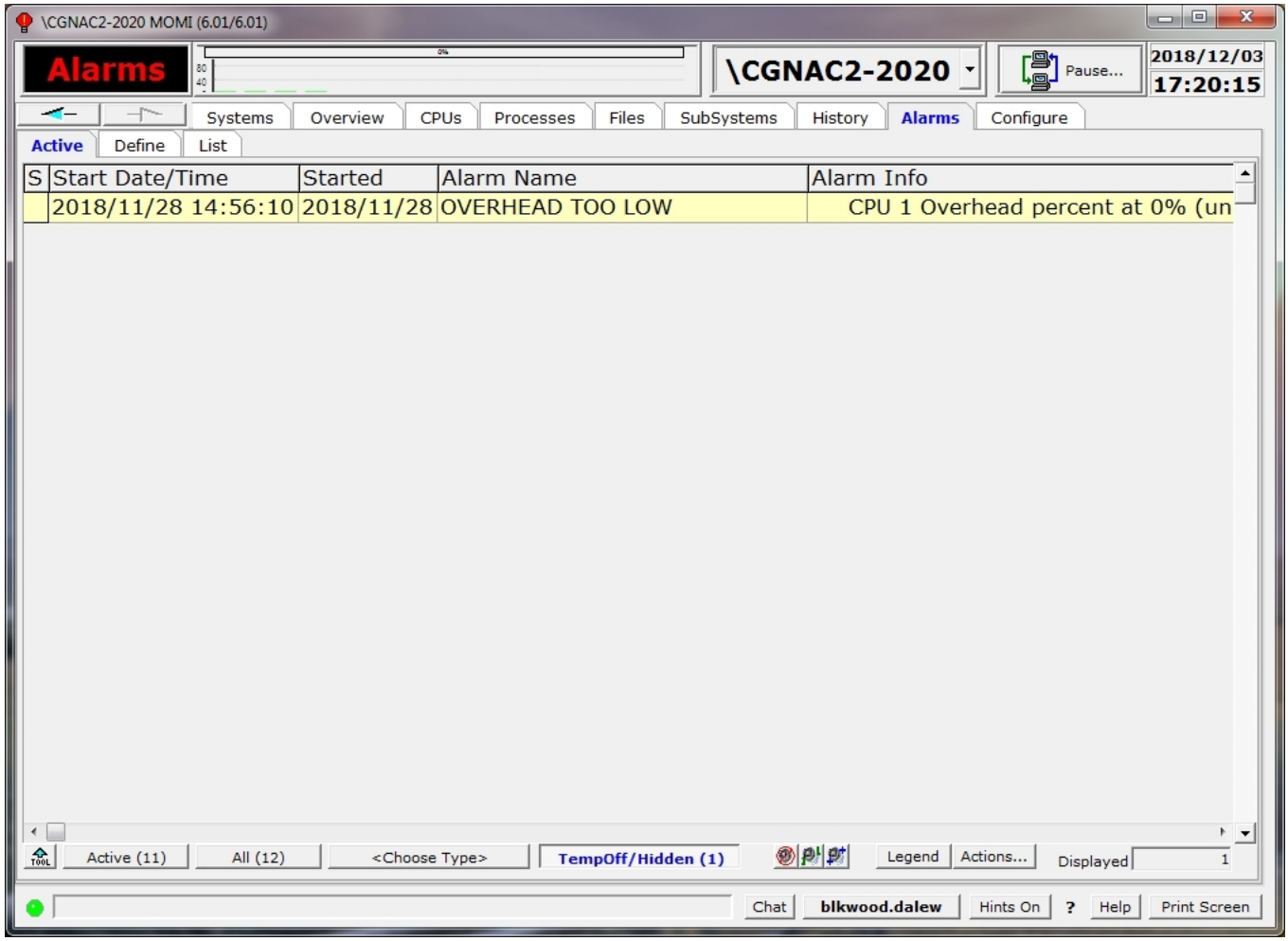
The single instance (i.e. one clicked) or all alarms associated with the definition may be deleted.



Temporary Turn OFF/ON Alarm allows a secured user the ability remove an active alarm from the screen. This feature provides an ability to remove an alarm when it cannot be resolved for a period of time.



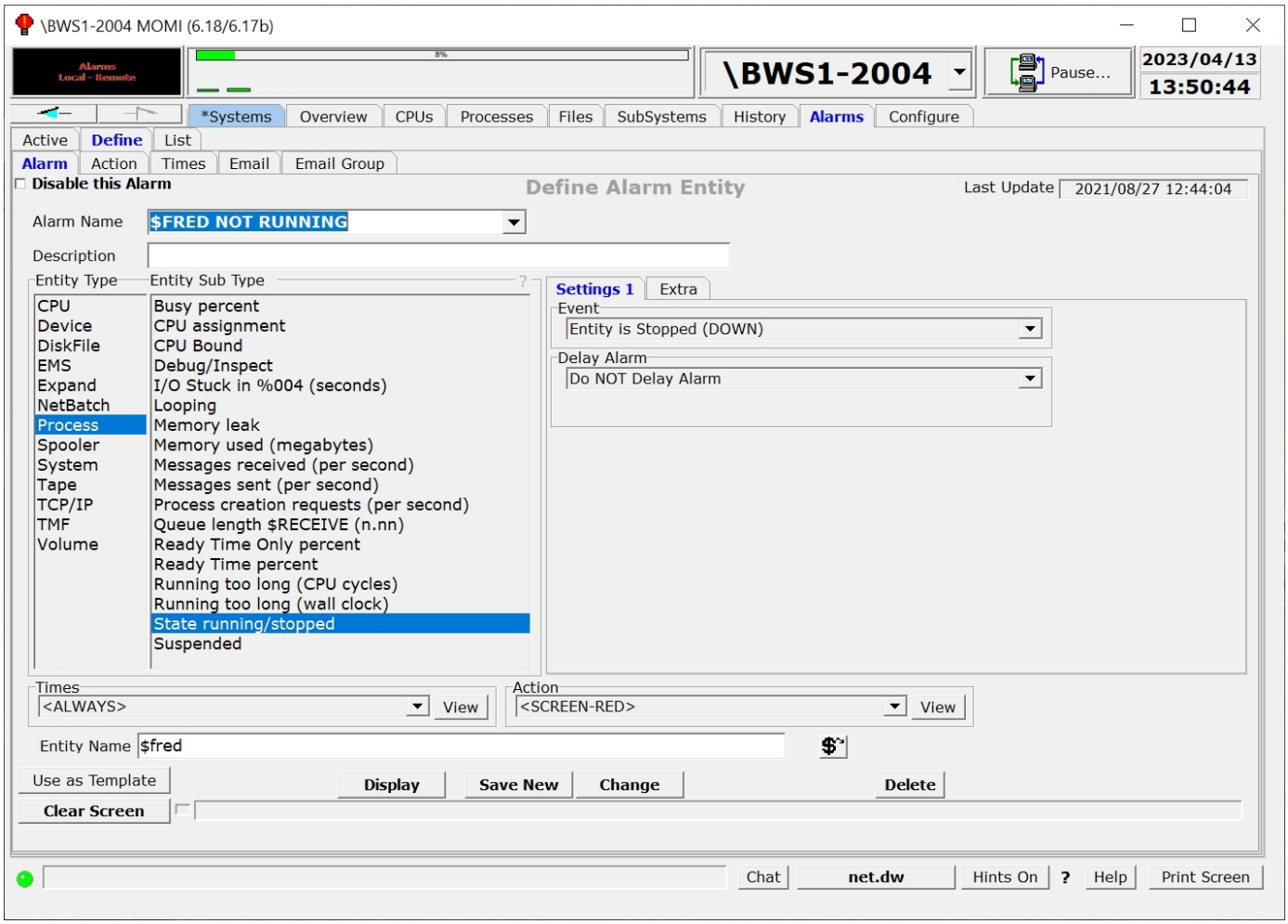
By default the screen display only active alarms. Temporarily turned off alarms or otherwise hidden alarms may also be displayed along with all alarms. The screen below displays Temporarily turned off alarms.



Pressing the **Actions...** button allows a secured user the ability suspend or resume all alarm processing. Any active alarm is removed if processing is suspended. This feature is used, for example, to prevent alarm processing during a maintenance window. This function may be used in conjunction with the CONFMOMI keyword ALARM-SUSPEND-DELAY that suspends alarm processing at the start up of MOMI.

Define

Define / Alarm



The Define / Alarm screen allows the creation and maintenance of alarms and the trigger threshold (s).

Disable this Alarm provides the ability to turn-off (or prevent the processing) of the record without deleting it.

Alarm Name is used to give the alarm a unique identification. This name appears on the Alarms / Active screen and should provide a general idea of what the alarm represents. Existing alarms are selected in the drop-down box.

Alarm Name may contain the characters [ ] to enclose text that will not display on the Alarms / Active screen. This may be used to cause the Alarm Name to appear more generic if needed. Examples:

Alarm Name	What displays on Active Alarm
PROCESS - NOT RUNNING $FRED	PROCESS - NOT RUNNING $FRED

```
PROCESS - NOT RUNNING [$FRED]          PROCESS - NOT RUNNING
```

Description provides free form text to explain the nature of the definition.

Entity Type column provides a list of general categories available. A selection is this column determines the contents of the **Entity Sub Type** column.

Entity Sub Type column is a list of the alarm-able items under an Entity. Selecting an Entity changes the items available under Entity Sub Type. A selection in this column determines the contents of events just to the right. Events are specific to each Sub Type. Place the mouse over events for additional information.

Under the **Extra** tab is the ability to associate sounds with an alarm or define a dependency.

A **Sound** may be played when an alarm is triggered. The available options are 1) speak the alarm, 2) a default or preselected message, 3) a user specified sound or 4) speak a user specified text string.

If this alarm would be redundant or not needed should another alarm already be triggered the **Alarm Dependency** may specify this relationship. Select in the **Alarm Name** drop down box the alarm that if active should prevent this alarm from triggering. For example, suppose $FRED is started by $PETE. If $PETE is down showing an alarm about $FRED does not really help. As shown above, specify the alarm name for $PETE as a dependency to prevent the display of an alarm for $FRED.

Match Alarm Name Only - compare only the Alarm Name.

Match Alarm Name and Entity Name - compare both the Alarm Name and Entity Name. If the alarm uses a wild-card for the Entity Name this option requires that both alarms are triggering off the same resultant name. An Entity Name is the particular item displayed in the alarm message such as $FRED, $LINE1, CPU 2, etc...).

The **Times** drop-down box selects when the alarm should be monitored. The default of <ALWAYS> means continuous monitoring.

The **Action** drop-down box selects what should occur in the event an alarm occurs. The default of <SCREEN-RED> means the Alarm appears on the Alarms / Active screen colored red.

Entity Name is used to enter CPU numbers, file names, process names, etc... The available options are displayed in the fields tool-tip.

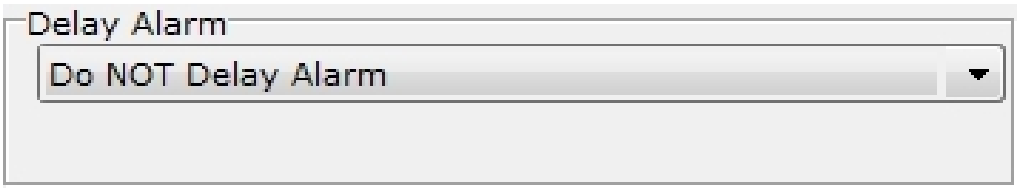
An update (**Change**) or **Disable this Alarm** clears any of its alarms currently active.

Detailed Information

Alarm attributes

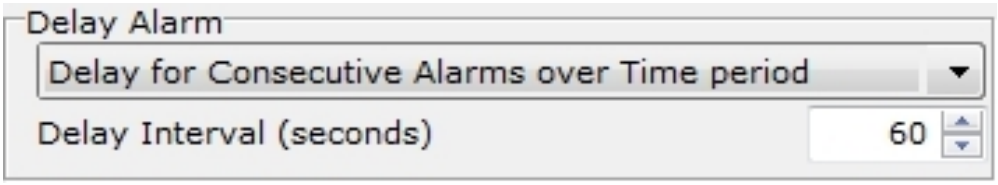
Delay Alarm

The **Delay Alarm** attribute determines if an alarm is displayed immediately or delayed by a period of time with a certain condition.



Do NOT Delay Alarm

Alarm is triggered immediately.



Delay for Consecutive Alarms over Time period

When the alarm criteria is reached, it must continue in that state for the Delay Interval before MOMI will trigger the alarm. Any drop outside the criteria resets the alarm computation or clears an existing alarm (if the thresholds were previously met).

Delay for Alarm Averaged over Time period

When the alarm criteria is reached, it must average in that state for the Delay Interval before MOMI will trigger the alarm. Any drop outside the "averaged" criteria resets the alarm computation or clears an existing alarm.

DiskFile - Percent full consider partitions(s)

This alarm uses the following file structure specific logic to examine and determine a file full condition:

Keyed-sequenced

All partitions are examined individually.

(data is placed in any partition as specified by the partition starting key)

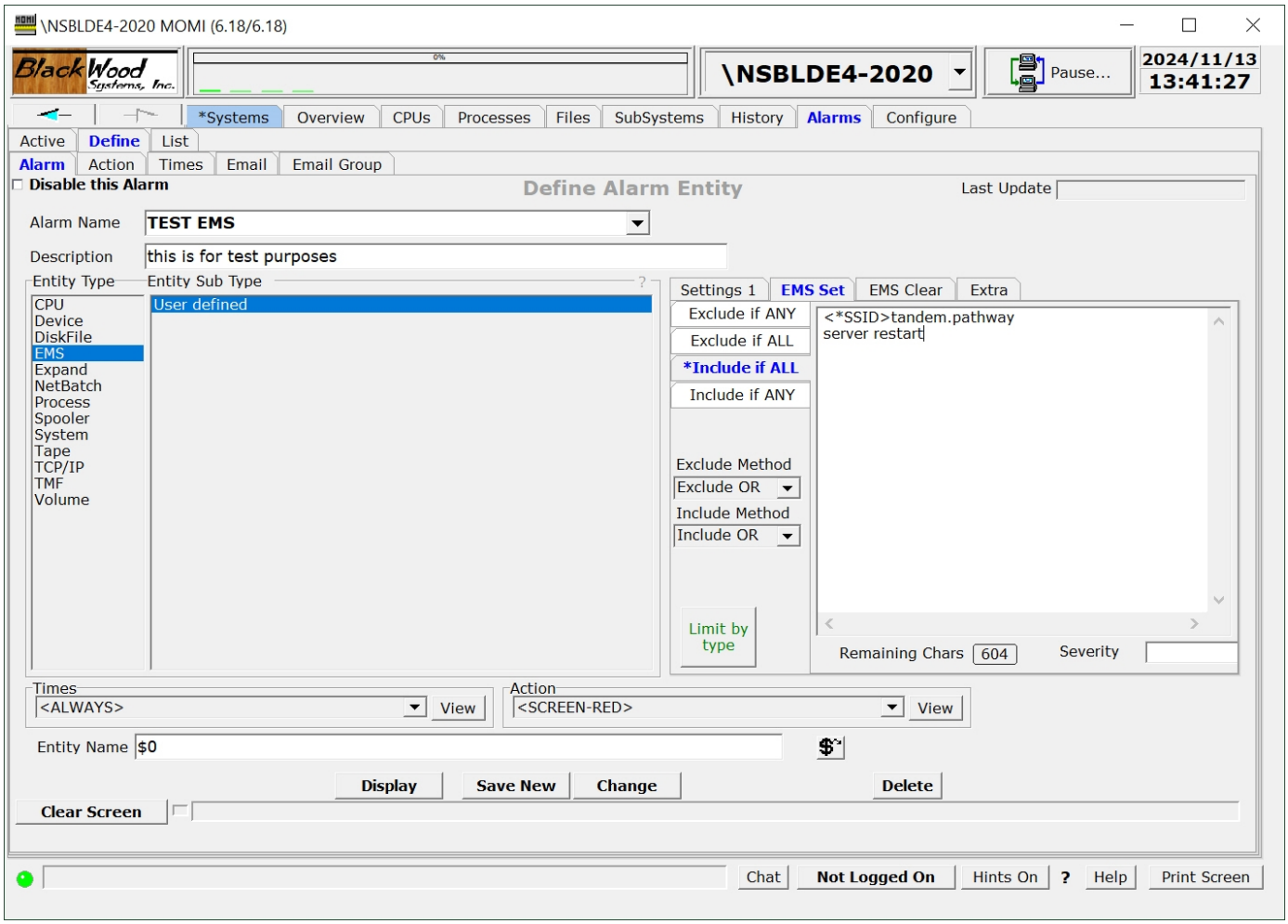
Entry-sequenced
Relative
Unstructured

Only the last partition is examined.
Secondary partitions are skipped.

(partitions are extensions of the overall file so data is placed at the specified location (Unstructured, Relative) or appended to the end (Entry-Sequenced, Relative))

EMS / User Defined

(updated for server version 6.17 or later)



The alarm **EMS / User defined** provides a method to generate an alarm based on the content of an EMS event. The EMS event may be located in $0 or in another EMS collector that MOMI can access.

An alarm is triggered, and potentially cleared, by defining matching text within a series of logical conditions. This criteria is entered into the text area in the selections of **Include if ANY**, **Include if ALL**, **Exclude if ANY** and **Exclude if ALL**. Each section has its own text area where zero or more lines of text may be entered. A line of text is matched in its entirety. For example, *Fred Smith* is considered as one single entry whereas one line of *Fred* and a second line of *Smith* means that these words may appear anywhere within the EMS message (i.e. don't have to be together). Comparisons are not case sensitive.

Below is the 1) the order of evaluation of the criteria, and 2) the action taken when a match occurs. This order is the same for both **EMS Set** and **EMS Clear**. Overall, **EMS Clear** is examined first then if no match was found **EMS Set** is examined second:

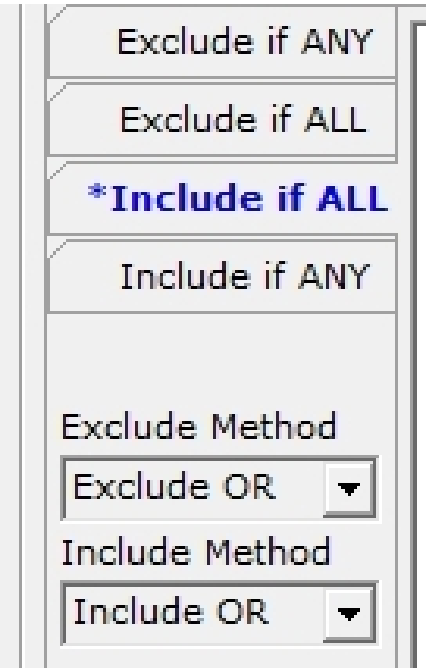
- **Exclude if ANY** means if any single line of text matches.
- **Exclude if ALL** means if ALL lines of text matches.

    *if a match occurs with **Exclude** - the EMS event is ignored*

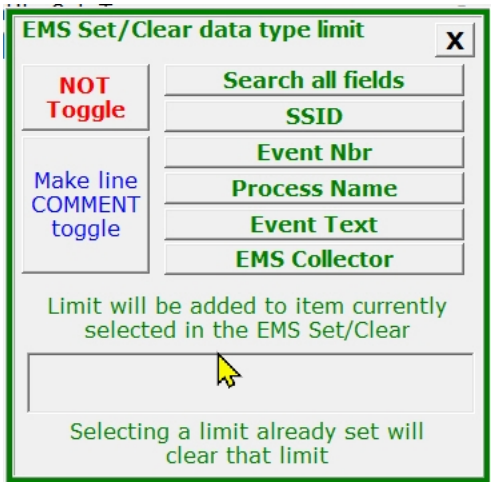
- **Include if ALL** means if all lines of text matches.
- **Include if ANY** means if any single line of text matches.

    *if a match occurs with **Include** - an alarm is generated or cleared*

- when no text matches the EMS event is ignored.

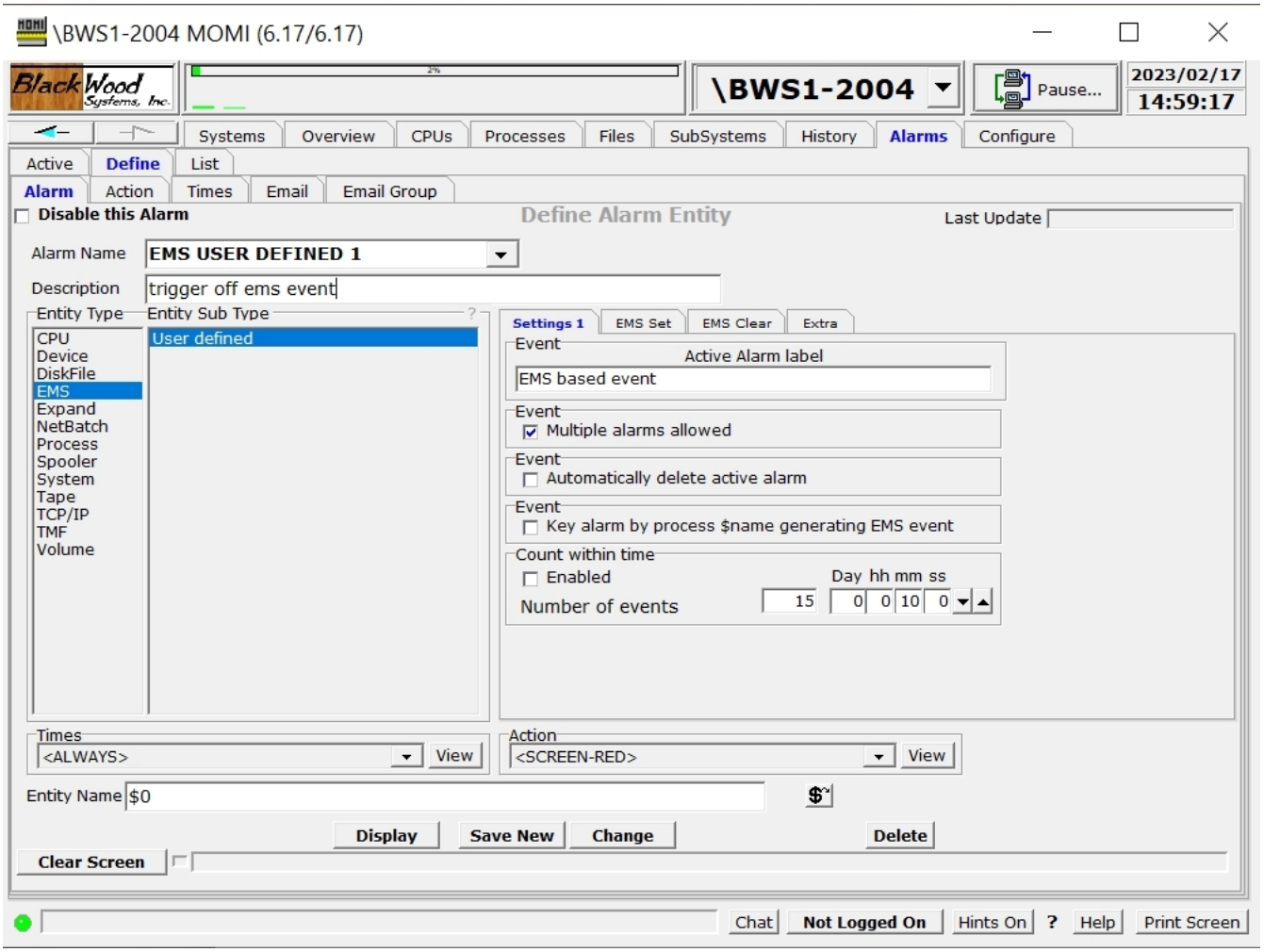


While not specified, prior to MOMI Client and Server version 6.04 an OR was effectively implied for *Excludes* and *Includes*. A text match of **Exclude if Any** OR **Exclude if ALL** caused an EMS message to be ignored. A text match of **Include if ALL** OR **Include if ANY** caused a match which would either generate or clear an alarm. As of MOMI Client and Server version 6.04 and later OR is now displayed plus the new logical operand AND was added. The default remains OR. **Include if ALL** AND **Include if ANY** requires a match of both in order to generate or clear an alarm.



Text entered within **EMS Set** and **EMS Clear** by default searches virtually the entire ASCII translation of the EMS event. To search just a portion of the EMS Message, push the **Limit By type** button and select which portion of the message to examine. If any *<tag(s)>* are present, they must precede any other text on that line.

EMS messages with an SSID of BWS, which is used by MOMI, are ignored. This prevents a potential "looping" situation where an Alarm triggered by an EMS message, causes an EMS message that triggers an alarm, that causes an EMS message that triggers an alarm, and so on...



The tab **Settings 1** contains various options to activate different features of this alarm.

Active Alarm label is text that appears within various messages generated by this alarm.

Multiple alarms allowed or **Multiple active alarms allowed** enables more than one active alarms as a result of this definition. A maximum number of active alarms is imposed to help limit run-away alarm situations and is defined by the CONFMOMI keyword ALARMS-MAXIMUM-PER-DEFINITION.

Automatically delete active alarm enables the generation of, for example emails, as a result of the alarm but then clears the alarm. This option would be used when the display of the alarm on the Alarms / Active screen is not required and manual delete of an alarm will not take place.

Key alarm by process $name generating EMS event causes an alarm to be keyed or associated with the process generating the EMS event. For example, several processes on the system generate an event when it falls behind in processing and a follow-up event when it has caught up. Criteria within **EMS Set** triggers the fall behind alarm and criteria within **EMS Clear** triggers clearing of the alarm. So that an alarm is directly associated with the processing generating the EMS events this option causes the processes $name to be internally linked to the active alarm.

There is no need to specify which processes are monitored. Note that it is required the **EMS Set** and **EMS Clear** contain sufficient criteria (such as SSID and event number) to uniquely filter to the subsystem specific EMS events.

Count within time specifies how many matching **EMS Set** events must occur over a period of time before an alarm occurs. For example, if 15 matching **EMS Set** events occur over a period of 10 minutes the issue is considered sufficient to generate an alarm.

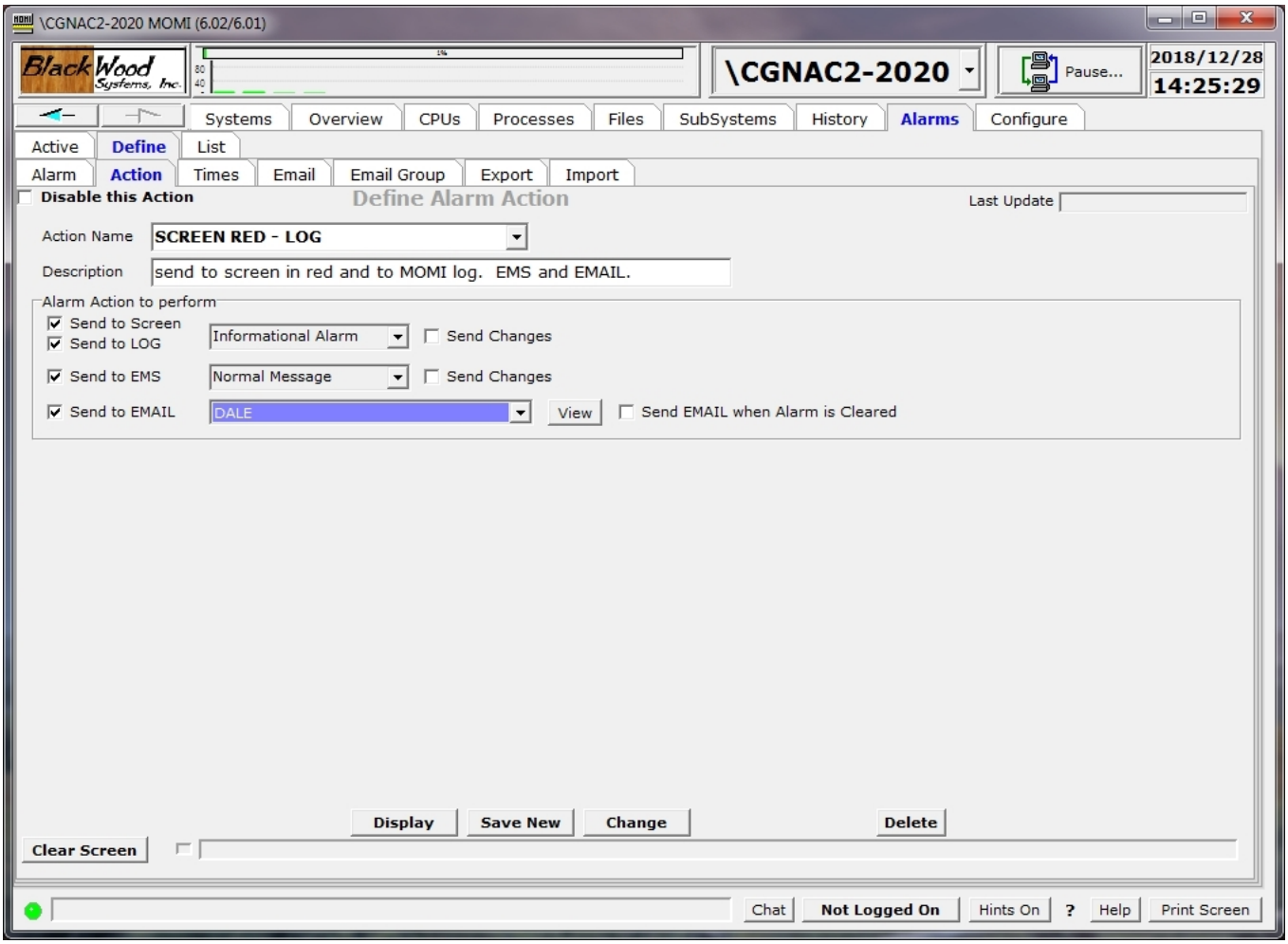
Process - State running / stopped

This alarm sets the variable <%Ancestor%> available on the screen Alarms / Define / Action.

An ancestor is the process that created the process of interest.

MOMI must see the running process once in order to obtain the ancestor information. Otherwise, the variable will return N/A (i.e. Not Available).

Define / Action



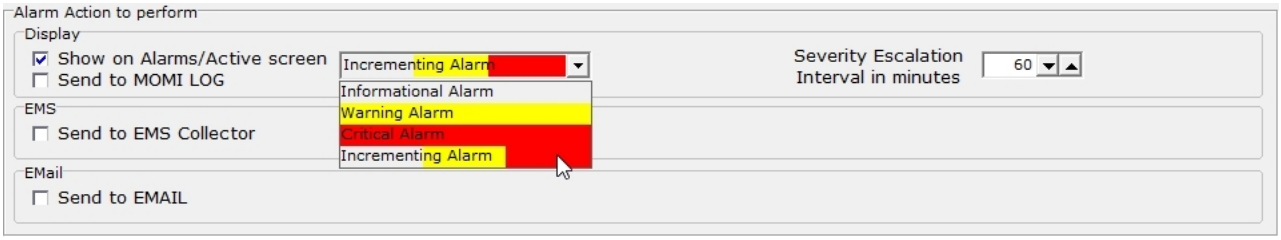
The Define / Action screen allows the creation and maintenance of what happens when an alarm is triggered and may be referenced when defining an alarm.

An Action is a set of notifications that occur when the alarm is triggered. The notification consists of any or all of the following:

- display on the Active Alarm screen
- generate an event to the MOMI LogFile
- generate an EMS message
- generate an Email

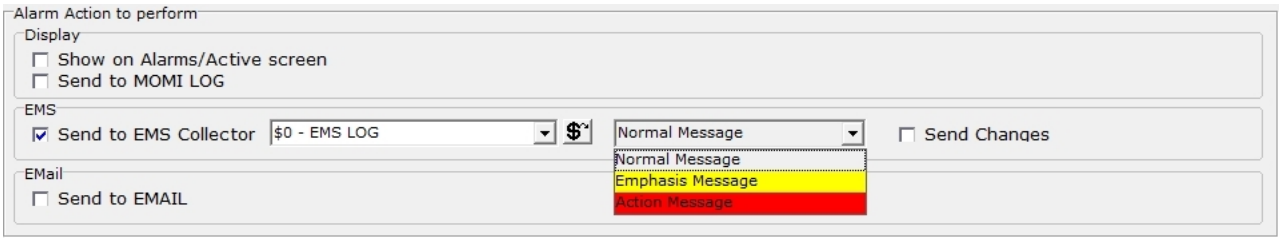
The field **Disable this Action** provides the ability to turn-off (or prevent the processing) of the record without deleting it. However, a disabled Action still results in the default of **Send to Screen** as Red.

A color is associated with Send to Screen and Send to LOG. Red is considered the most severe and Screen the least. An incrementing alarm starts with on screen, changes to yellow and finally to red based on the Escalation Interval.

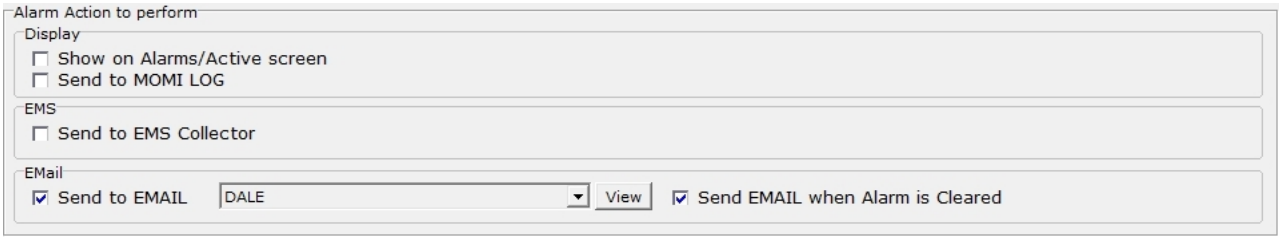


The MOMI LOG is an event file maintained by MOMI. It is accessed on the screen LogFile. Send Changes indicates that any change in the Alarm messages is logged. By default, only the first and last (or clear) are logged.

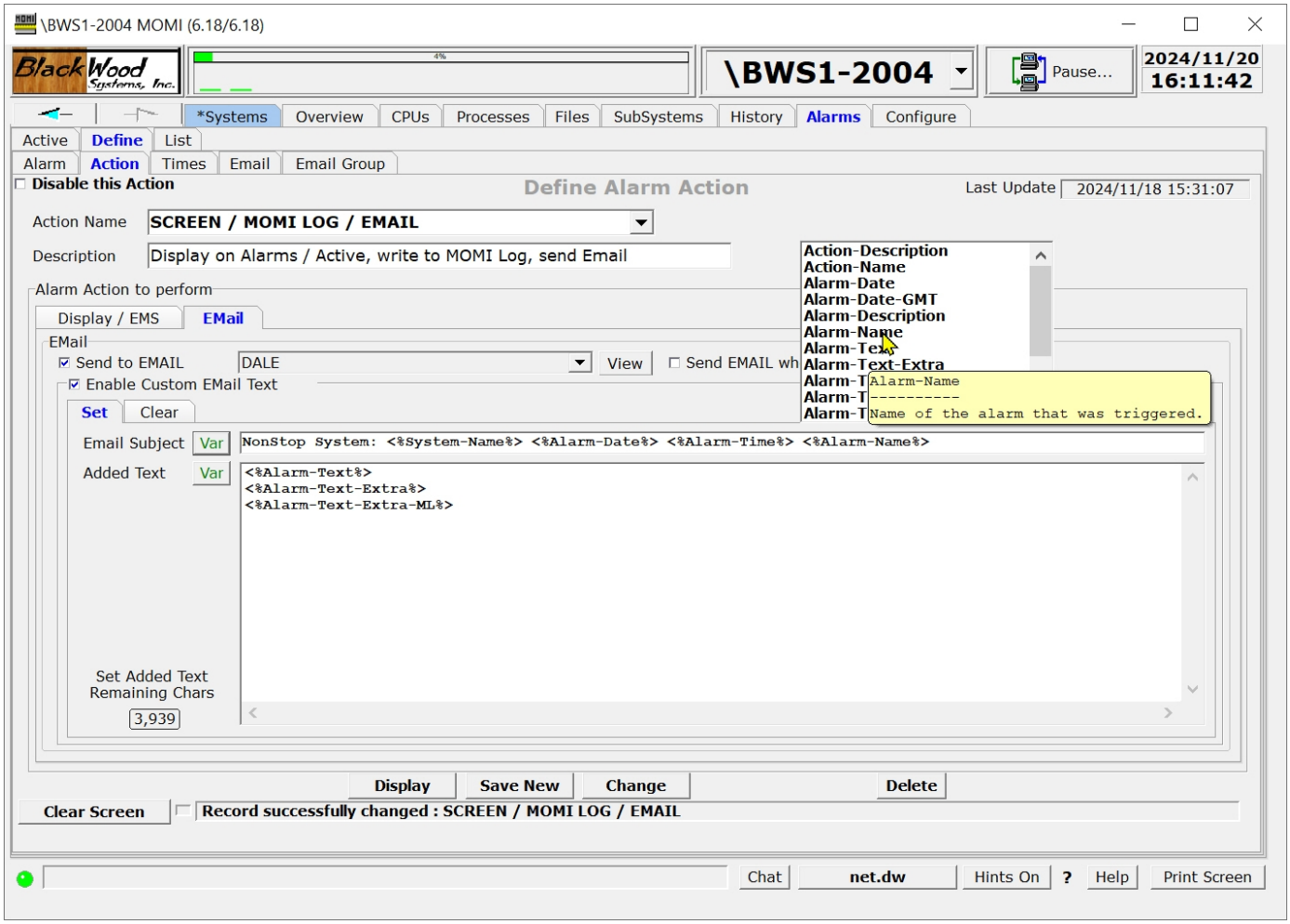
An EMS message may be sent to the $0 collector (default) or an alternate collector. The message has the attributes of Action, Emphasis or Normal as would be shown on the screen EMS Msgs. **Send Changes** indicates that any change in the Alarm messages is also logged. By default, only the first and last (or clear) are logged.



Email may be sent to either an individual address (Email) or a group of addresses (Email_Group). If an alarm is generated, one email is sent. If the option **Send EMAIL when Alarm is Cleared** is checked a second email is sent when the alarm is no longer present.



The default format of the EMS and Email message may be overridden and defined by the user when **Enable Custom...** is checked. Two general formats of alarms are when an alarm is initially triggered (updated), or **Set**, and when an alarm is no longer present, or **Clear**ed. Text entered by the user appears 'as-is' in the mail. Variables, not case sensitive, are identified by names enclosed within <% .. %> and reference information about the alarm, MOMI and the System. Press the **Var** button to display a list of variables, which when selected, inserts at the cursor position. In the drop-down list place the mouse over a variable to display a description of the information it represents.



Variable Descriptions

These variables comprise the basic elements of an active alarm:

- **Alarm-Text**

  The basic text of an alarm such as:

  Process $a is not running

  File $data1.proddb.account is 81% full over threshold of 80%

- **Alarm-Text-Extra**

  A Second line of additional information such as list of processes, object file name, or other information concerning an alarm.

- **Alarm-Text-Extra-ML**

  Some alarms include multiple lines of additional text. For example, the alarm **EMS / User defined** sets this variable to the EMS event that triggered or cleared the alarm (note this duplicates variable *Alarm-Text-Extra*). Place this variable on a line by itself. It is not available for single line fields such as an EMS message or the subject of an alarm.

These variables provide a means to access the three basic elements without reporting duplication. They are not available for single line fields such as an EMS message or the subject of an alarm:

- **Alarm-Text-All-ML**

  All alarm text representing the variables **Alarm-Text**, **Alarm-Text-Extra**, and **Alarm-Text-Extra-ML** generally without duplication. Data for each underlying variable is provided on one or more lines.

- **Alarm-Text-Extra-All-ML**

  All alarm text representing the variables **Alarm-Text-Extra** and **Alarm-Text-Extra-ML** generally without duplication. Data for each underlying variable is provided on one or more lines.

Select variables:

- **Alarm- Date [-GMT]**

  LCT or GMT Date when alarm first occurred - format : yyyy-mm-dd

- **Alarm-Time [-GMT]**

  LCT or GMT Time when alarm first occurred - format: hh:mm:ss

- **Alarm-Name**
- **Alarm-Description**

  Name/Description of alarm as given on the screen Alarms / Define / Alarm.

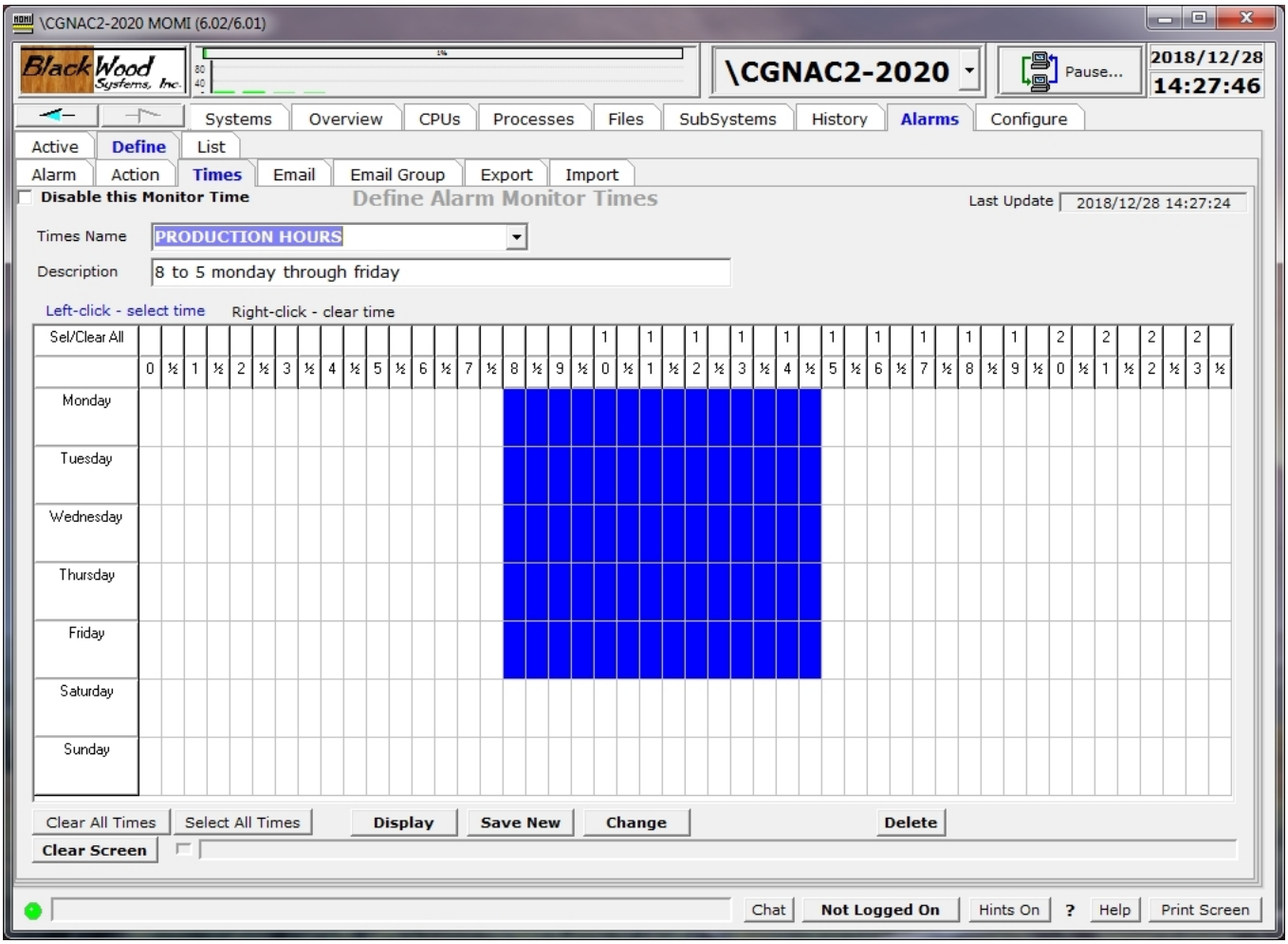
- **Action-Name**
- **Action-Description**

  Name/Description of action given on the screen Alarms / Define / Action.

- **Entity-Name**

Process name, file name, EMS Collector name, etc... of specific item triggering the alarm.

Define / Times



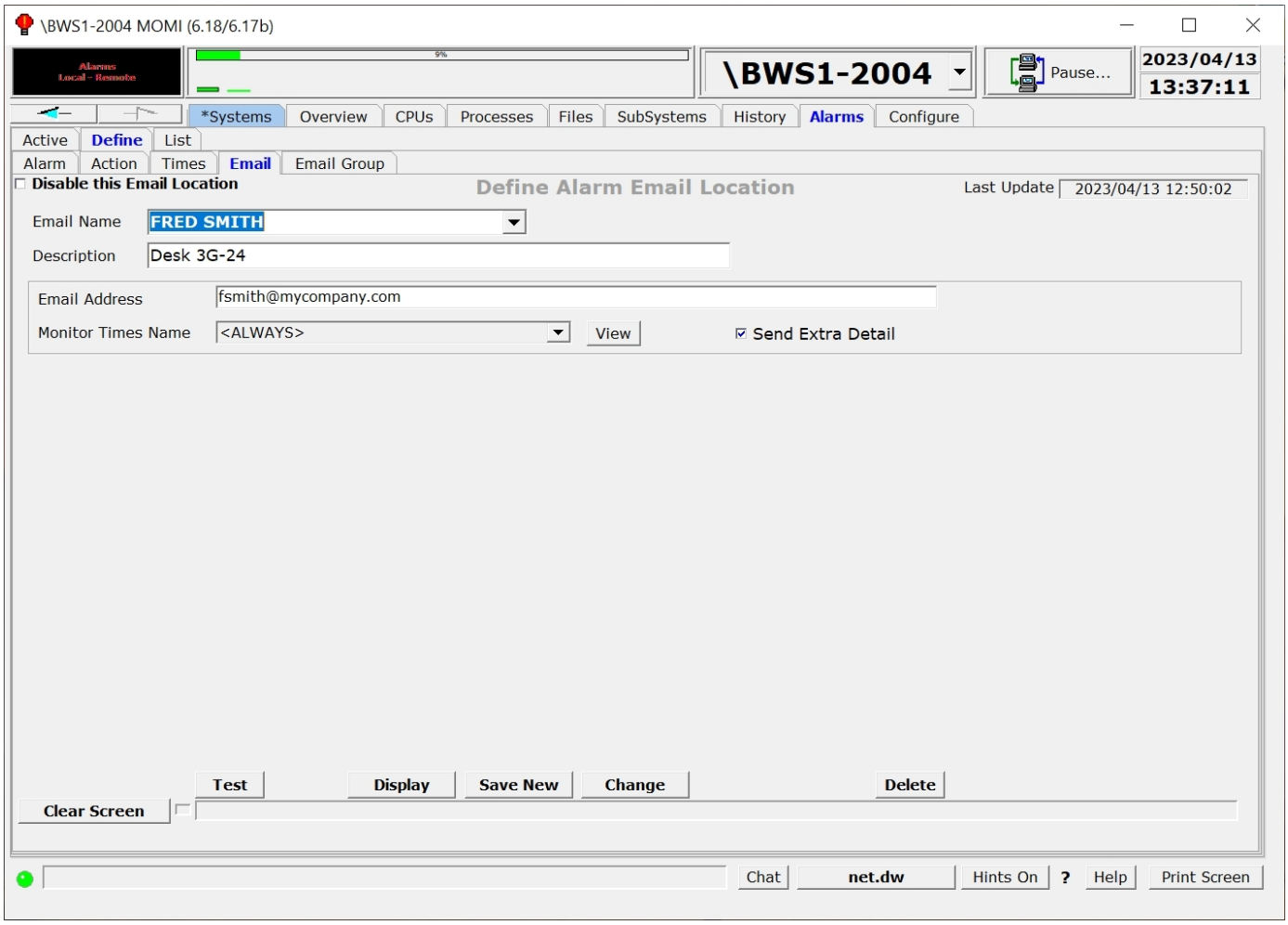
The Define / Times screen allows the creation and maintenance records used to determine periods of time which may be referenced when defining an alarm and email.

The field **Disable this Monitor Time** provides the ability to turn-off (or prevent the processing) of the record without deleting it. However, any object that references a disabled entry results in the default of active all the time.

A Monitor Times record provides 1/2 hour increments Monday through Sunday.

Place the mouse over a desired time period an press the left mouse button. Holding the button down and moving the mouse sets multiple cells (left mouse button sets / right mouse button clears). All cells may be set or cleared by clicking on **Clear All Times** or **Select All Times**.

Define / Email



The Define / Email screen allows the creation and maintenance of Email records which may be referenced when defining an alarm.

Disable this Email Location provides the ability to turn-off (or prevent the processing) of the record without deleting it.

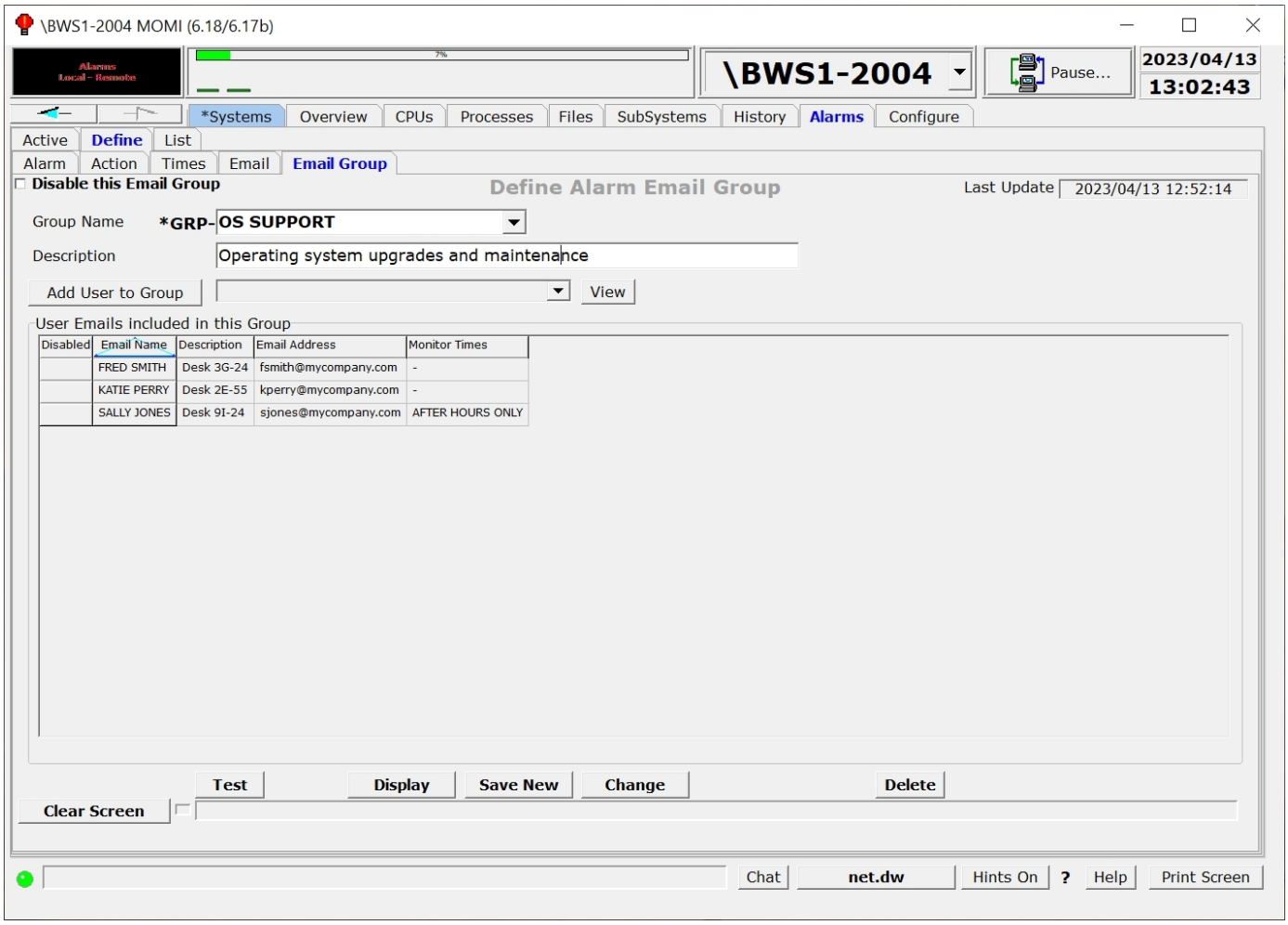
Send Extra Detail causes the inclusion of additional information, if available, within the email for this particular entry.

Monitor Times Name may be used to place a time limitation on emails.

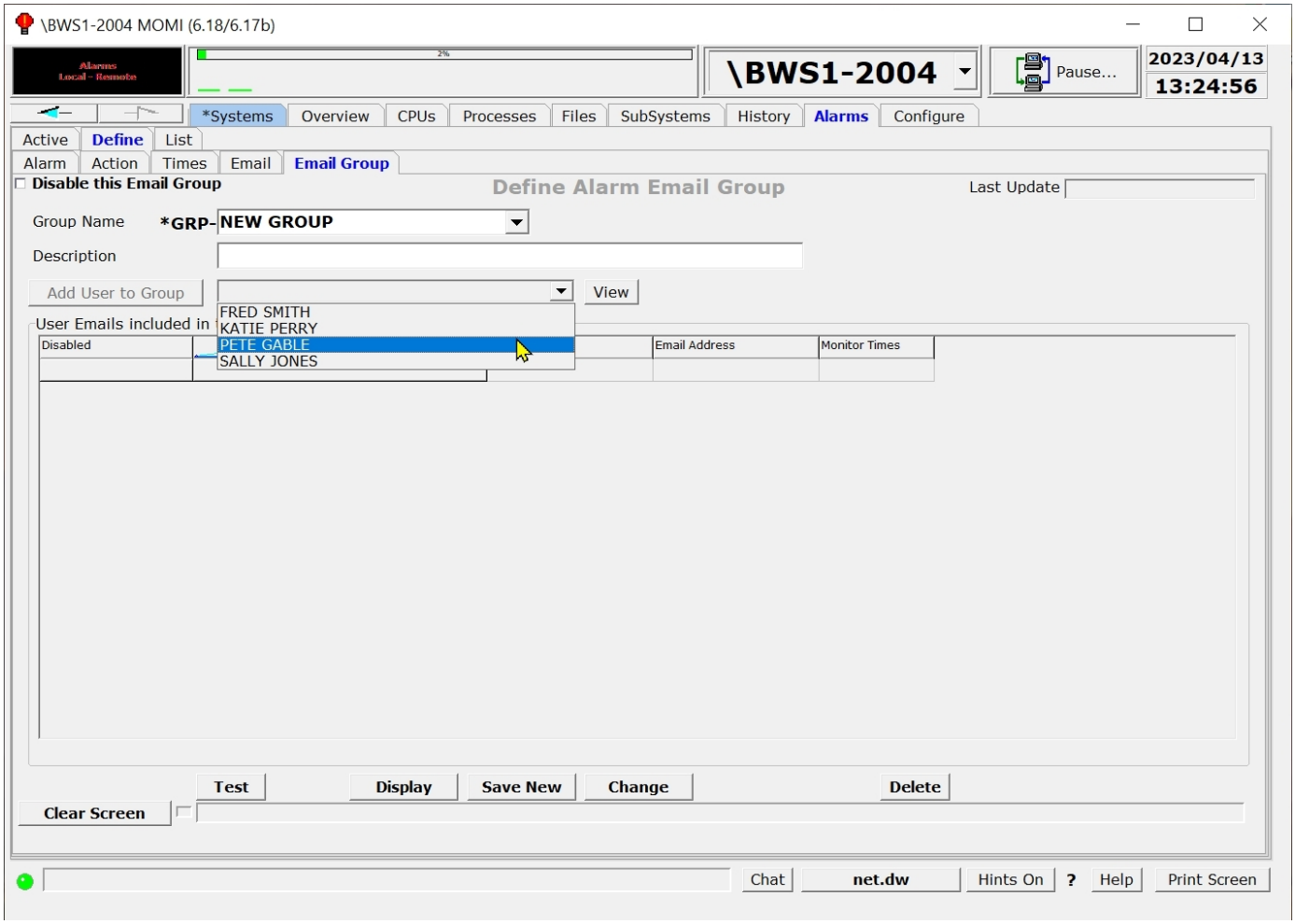
The **Test button** generates a sample email. This helps insure that the information entered is correct and that MOMI is configured properly.

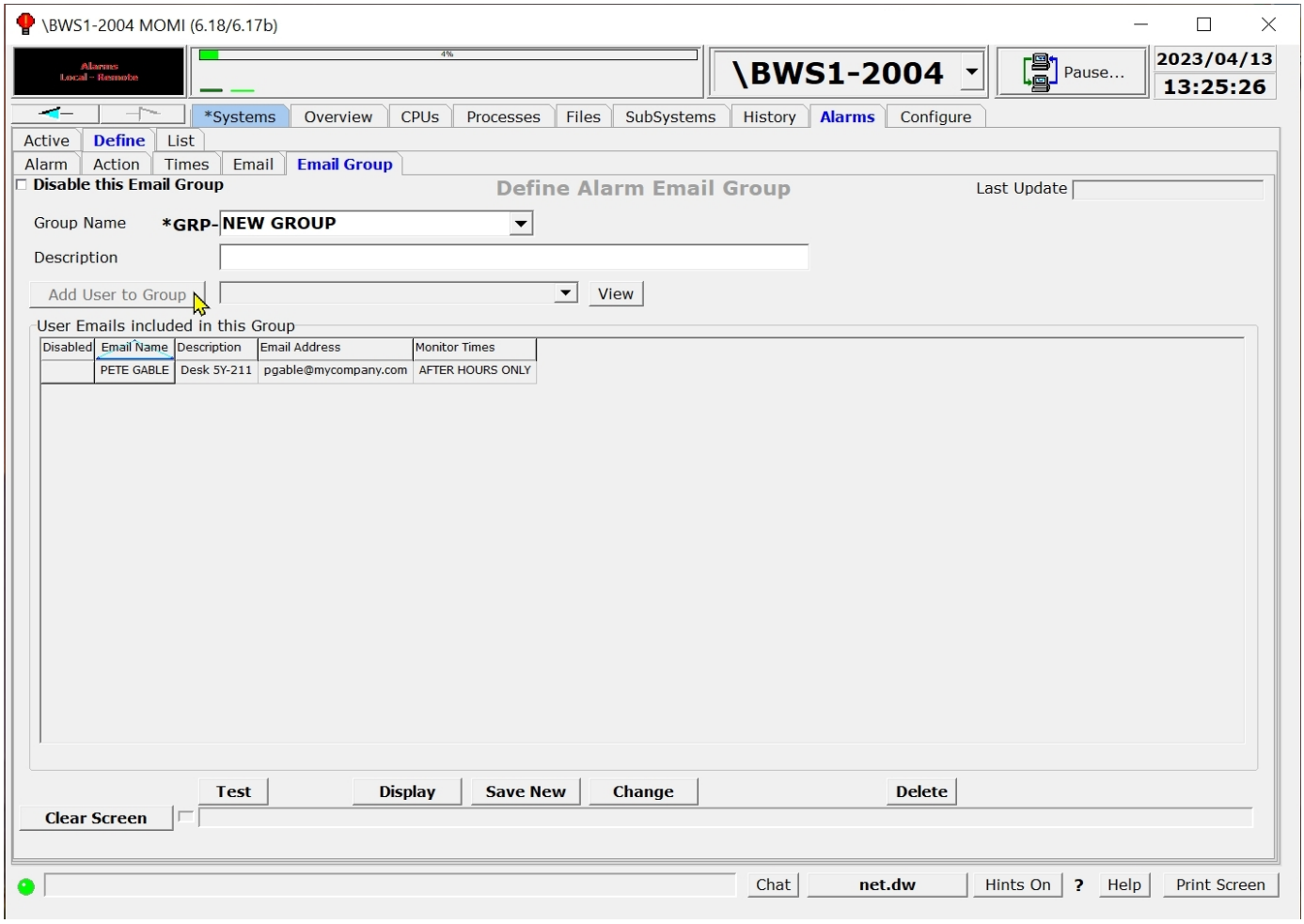
In order to generate email, it is necessary to define several keywords in the CONFMOMI file. See Alarm Overview for more information.

Define / Email Group

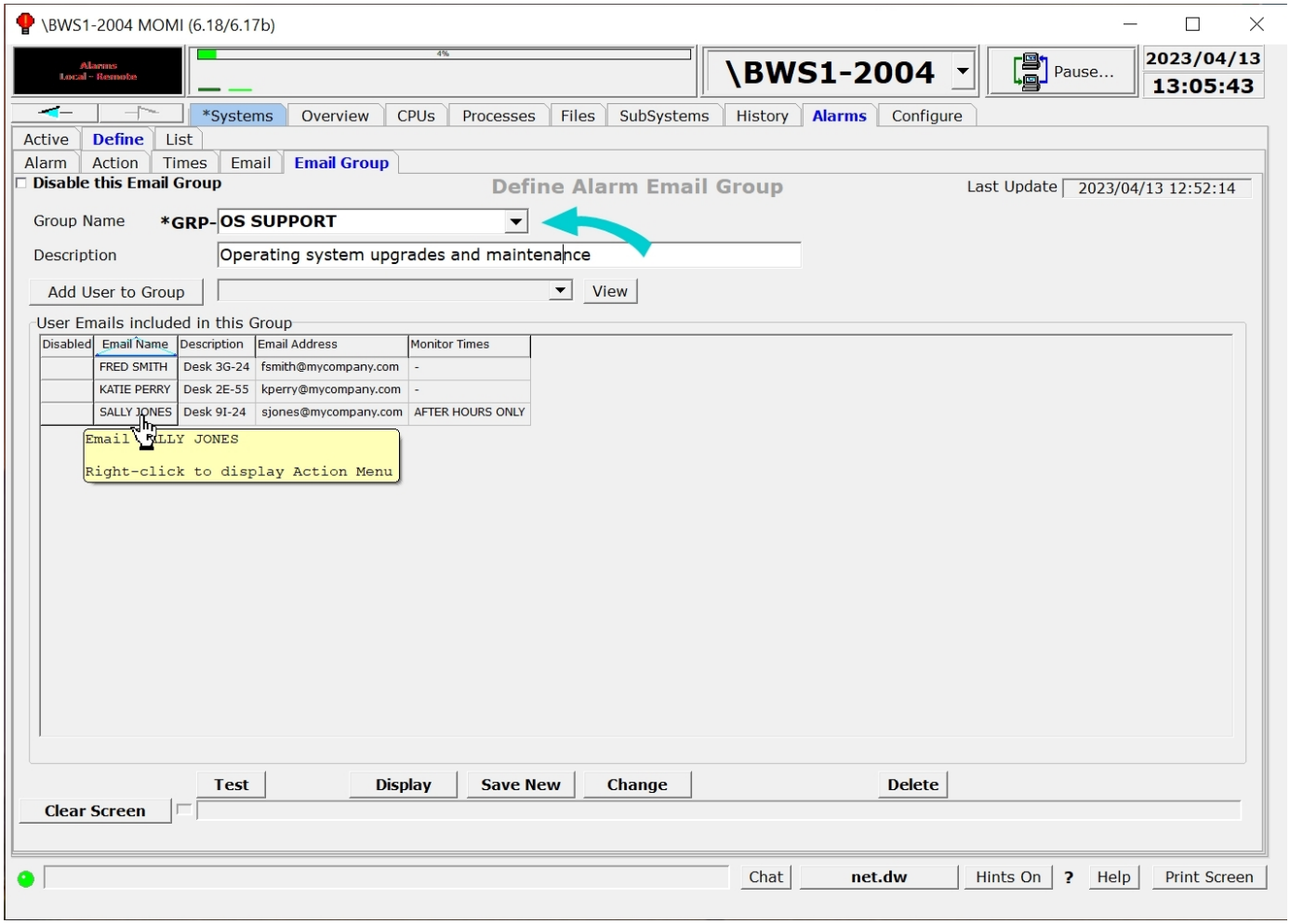


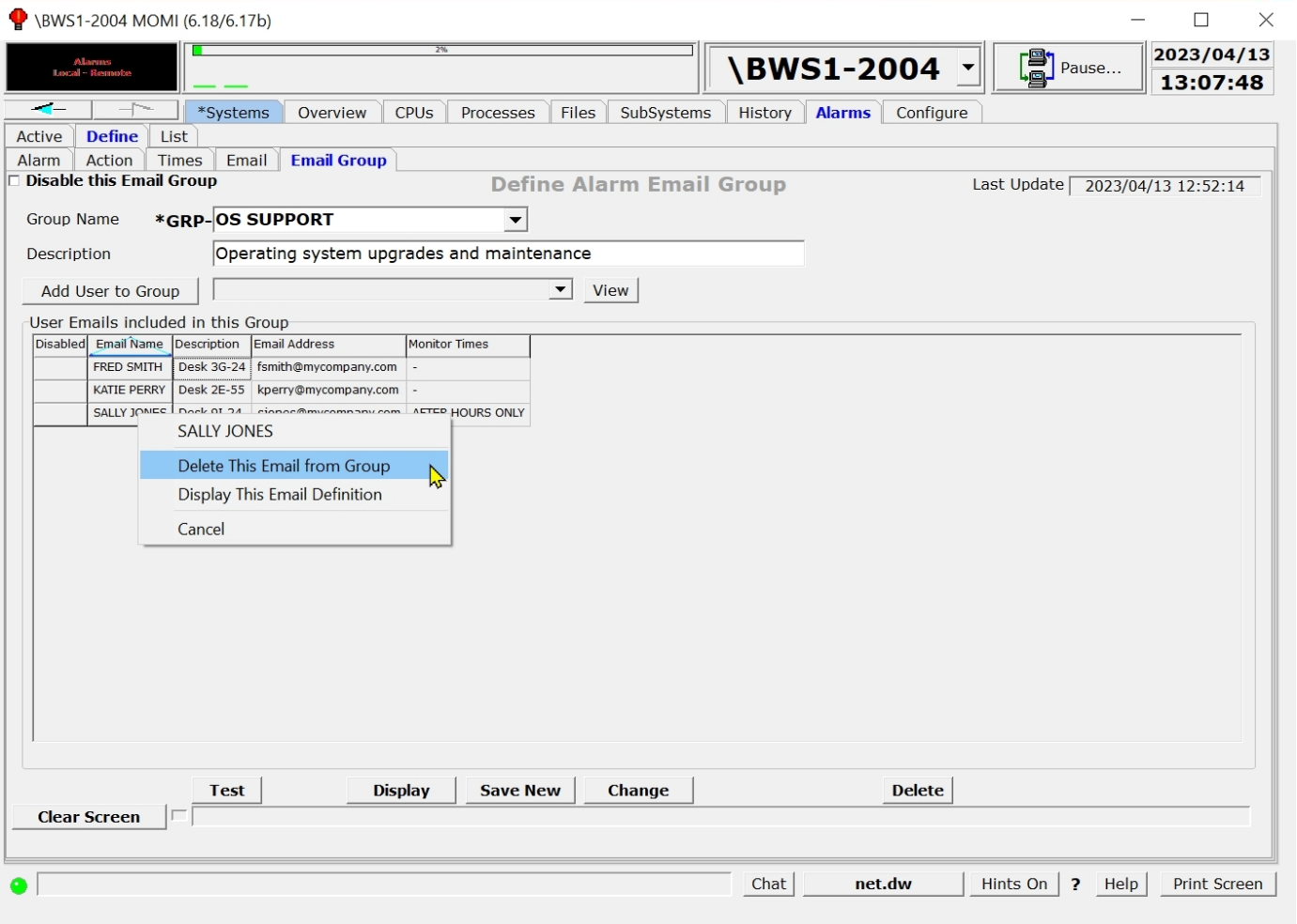
The Define / Email Group screen allows the creation and maintenance of Email group records. An Email Group is a reference to one or more previously created Email definitions.


\BWS1-2004 MOMI (6.18/6.17b)
Alarms
Local - Remote
2%
\BWS1-2004
Pause...
2023/04/13
13:24:56
*Systems
Overview
CPUs
Processes
Files
SubSystems
History
Alarms
Configure
Active
Define
List
Alarm
Action
Times
Email
Email Group
Disable this Email Group
Define Alarm Email Group
Last Update
Group Name
*GRP-
NEW GROUP
Description
Add User to Group
View
FRED SMITH
KATIE PERRY
PETE GABLE
SALLY JONES
User Emails included in
Disabled
Email Address
Monitor Times
Test
Display
Save New
Change
Delete
Clear Screen
Chat
net.dw
Hints On
?
Help
Print Screen



To create a new group, enter a **Group Name** (the *GRP- prefix is fixed), **Description** and then using the drop down box as shown above select an Email then press **Add User to Group**. Repeat for each email needed in this group. Press **Save New** when complete.


\BWS1-2004 MOMI (6.18/6.17b)
\BWS1-2004
Pause...
2023/04/13
13:05:43
*Systems
Overview
CPUs
Processes
Files
SubSystems
History
Alarms
Configure
Active
Define
List
Alarm
Action
Times
Email
Email Group
Disable this Email Group
Define Alarm Email Group
Last Update
2023/04/13 12:52:14
Group Name
*GRP-
OS SUPPORT
Description
Operating system upgrades and maintenance
Add User to Group
View
User Emails included in this Group
Disabled
Email Name
Description
Email Address
Monitor Times
FRED SMITH
Desk 3G-24
fsmith@mycompany.com
-
KATIE PERRY
Desk 2E-55
kperry@mycompany.com
-
SALLY JONES
Desk 9I-24
sjones@mycompany.com
AFTER HOURS ONLY
Email SALLY JONES
Right-click to display Action Menu
Test
Display
Save New
Change
Delete
Clear Screen
Chat
net.dw
Hints On
?
Help
Print Screen



To modify an existing group, select the group in the **Group Name** drop down, make any changes in the **Description**, add any new emails (as previously described), or to remove an email right-click over its **Email Name** and select **Delete This Email from Group**. Press **Change** when complete.

Records defined here do not in-and-of themselves perform any action. But rather are referenced in other definitions.

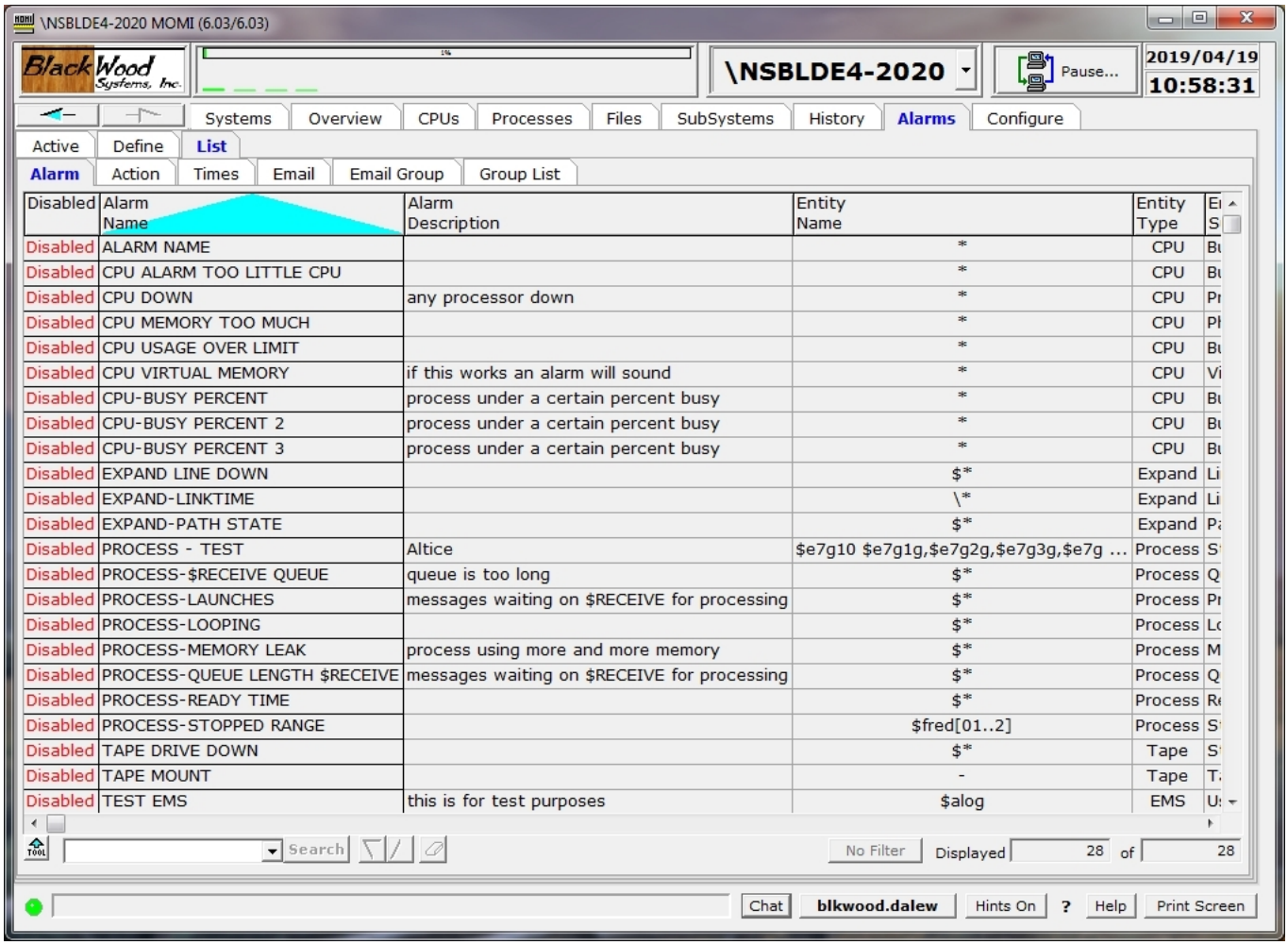
Disable this Email Group provides the ability to turn-off (or prevent the processing) of the record without deleting it.

The **Test** button generates a test email using the settings defined for Alarm Email.

In order to use email, it is necessary to set several keywords in the CONFMOMI file. See the Alarm Overview page for more information.

List

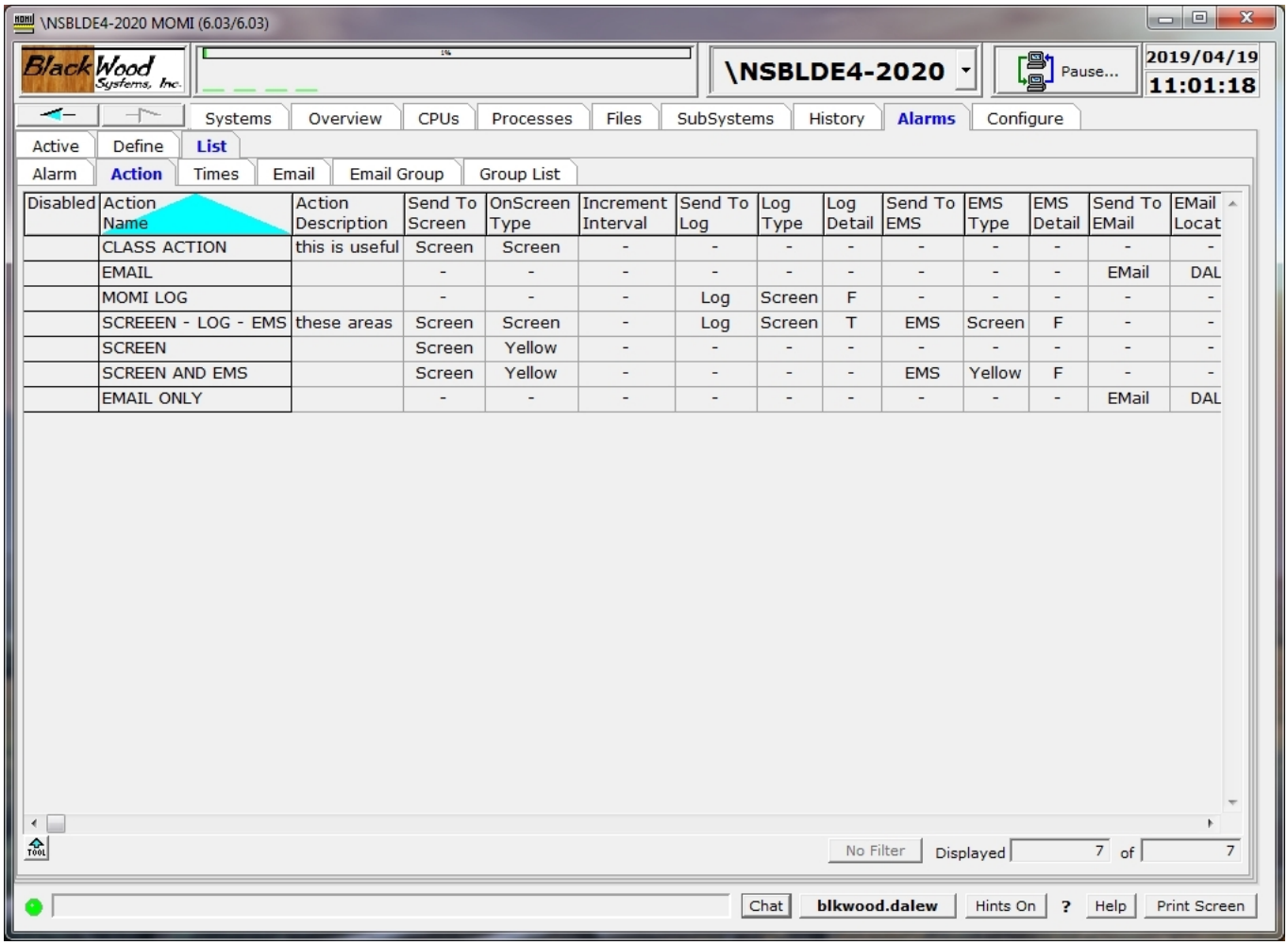
List / Alarm



The List / Alarm screen displays a tabular summary of currently defined records.

A left-click jumps to the Define screen for an Alarm. Several right-click options are available to jump to the Define / Alarm screen, Monitor Times and Action screens. Also, a record may be disabled (which effectively turns it off but does not delete it).

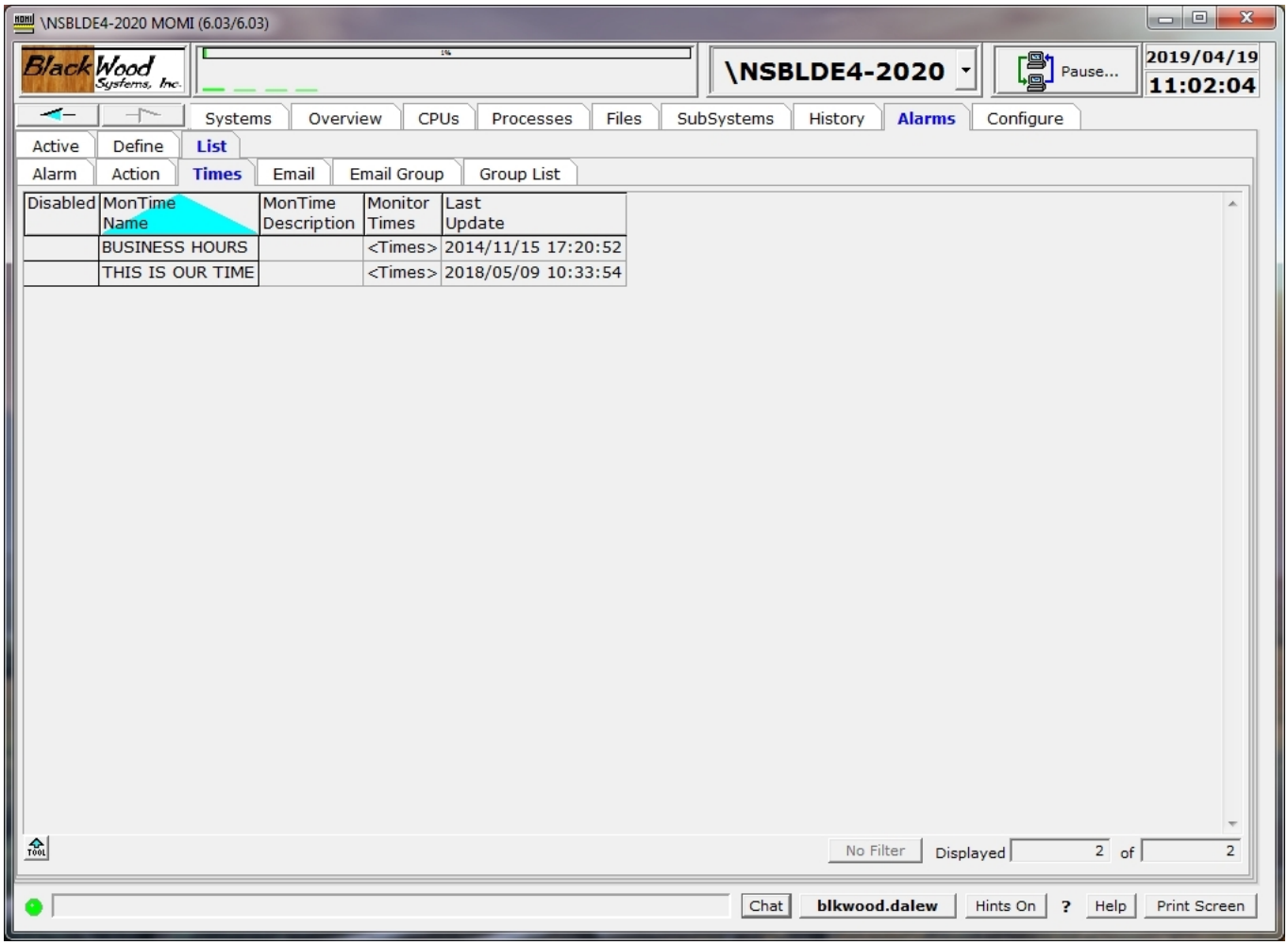
List / Action



The List / Action screen displays a tabular summary of currently defined records.

A left-click jumps to the Define screen for an Action. A right-click is available to disable a record (which effectively turns it off but does not delete it).

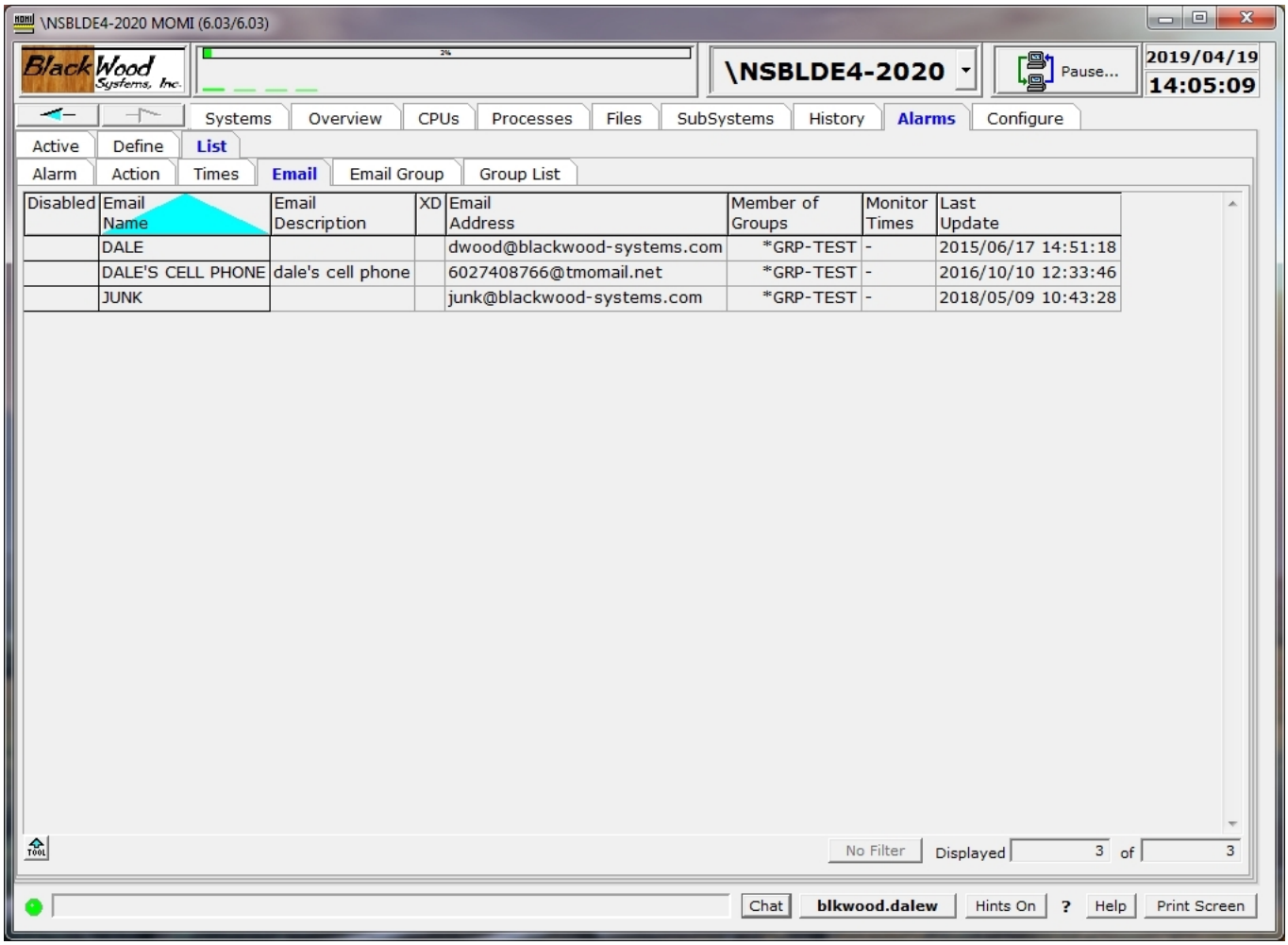
List / Times



The List / Times screen displays a tabular summary of currently defined records.

A left-click jumps to the Define screen for an Times. A right-click is available to disable a record (which effectively turns it off but does not delete it).

List / Email

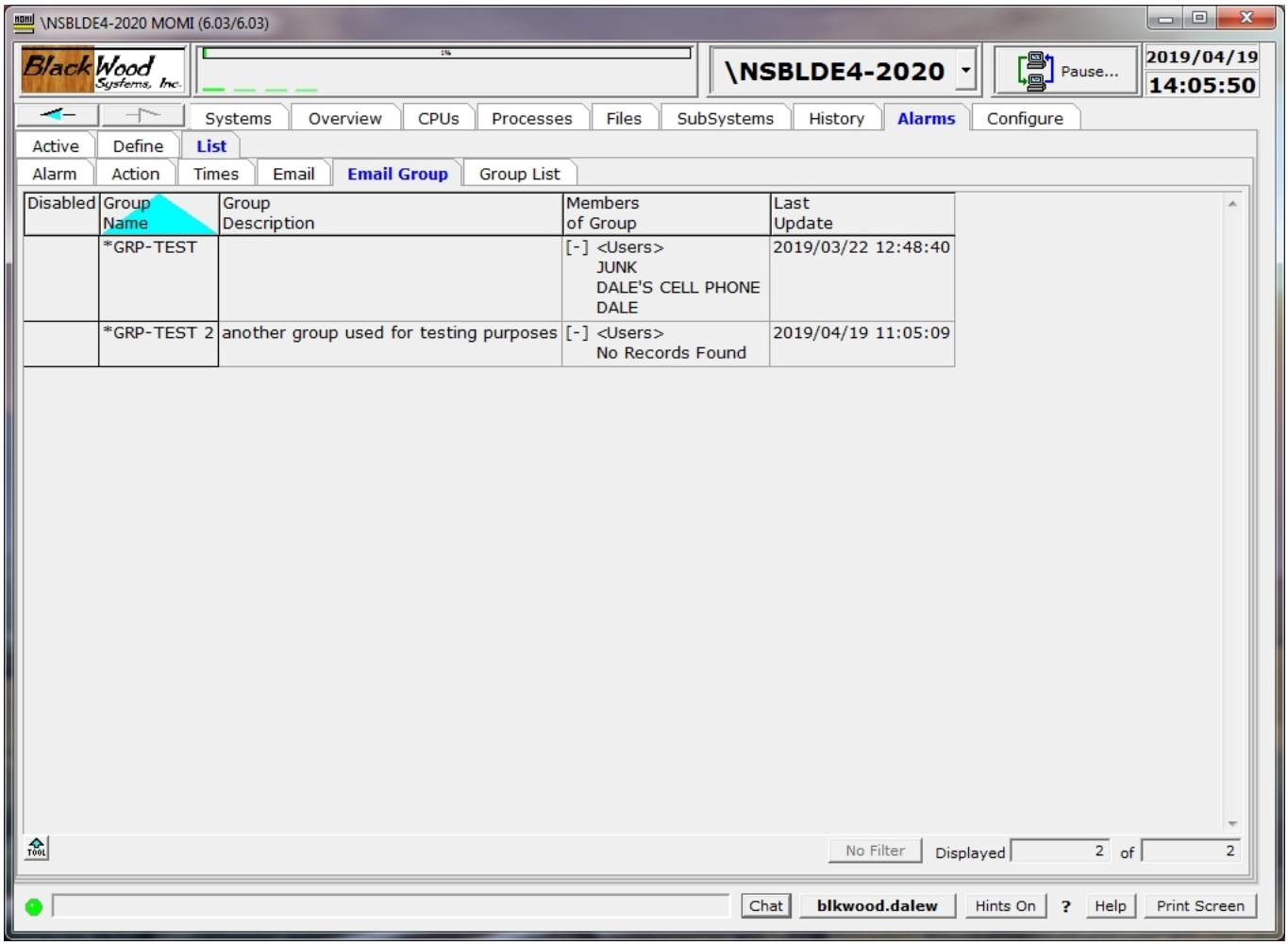


The List / Email screen displays a tabular summary of addresses defined.

A left-click on the Email Name jumps to the Define screen. Click on the [+] to display groups this email is a member.

A right-click is available to disable a record (which effectively turns it off but does not delete it).

List / Email Group

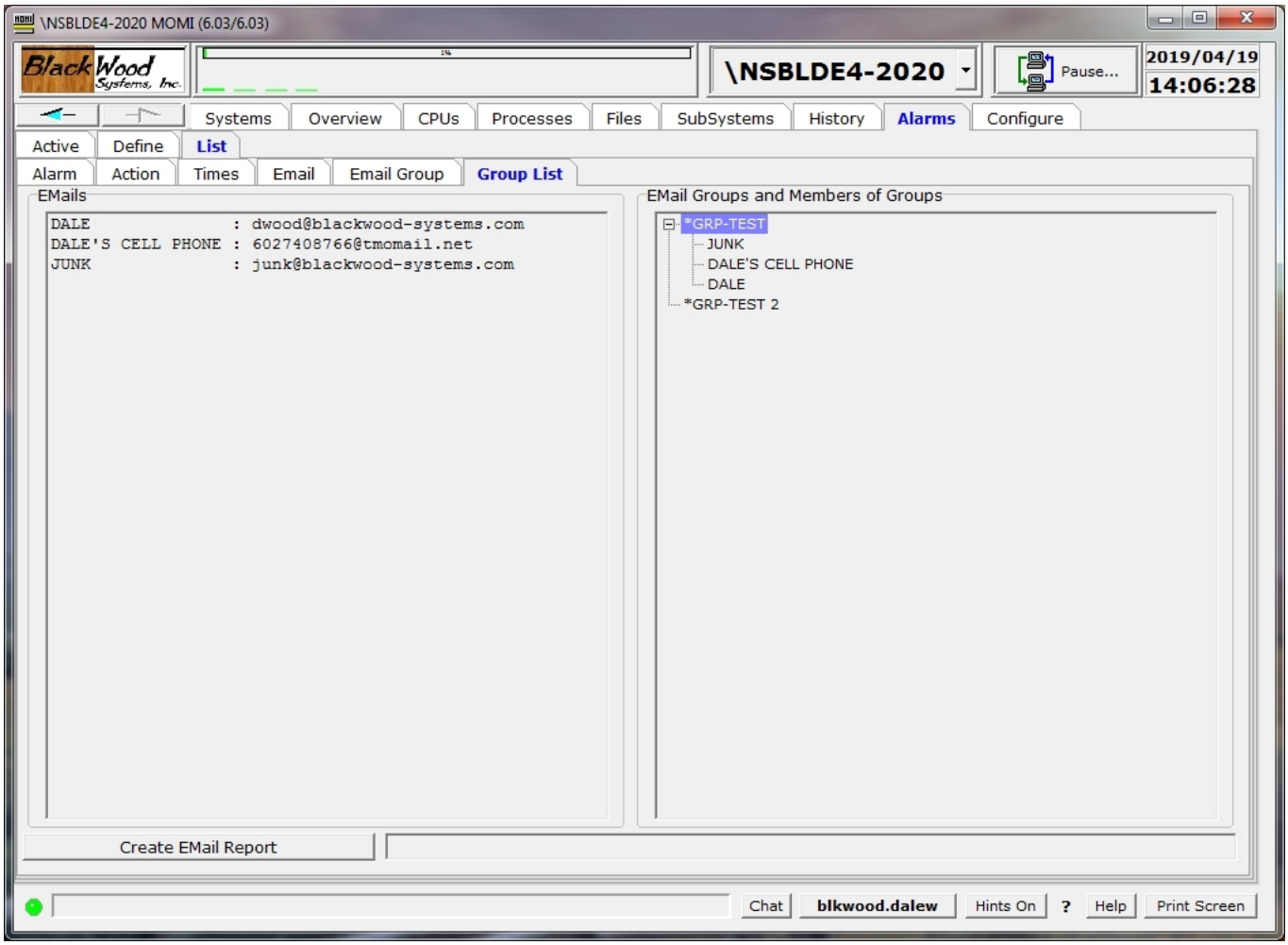


The List / Email Group screen displays a tabular summary of currently defined records.

A left-click on the Group Name jumps to the Define screen for an Email Group. Click on the [+] to display members of the group.

A right-click is available to disable a record (which effectively turns it off but does not delete it).

List / Group List

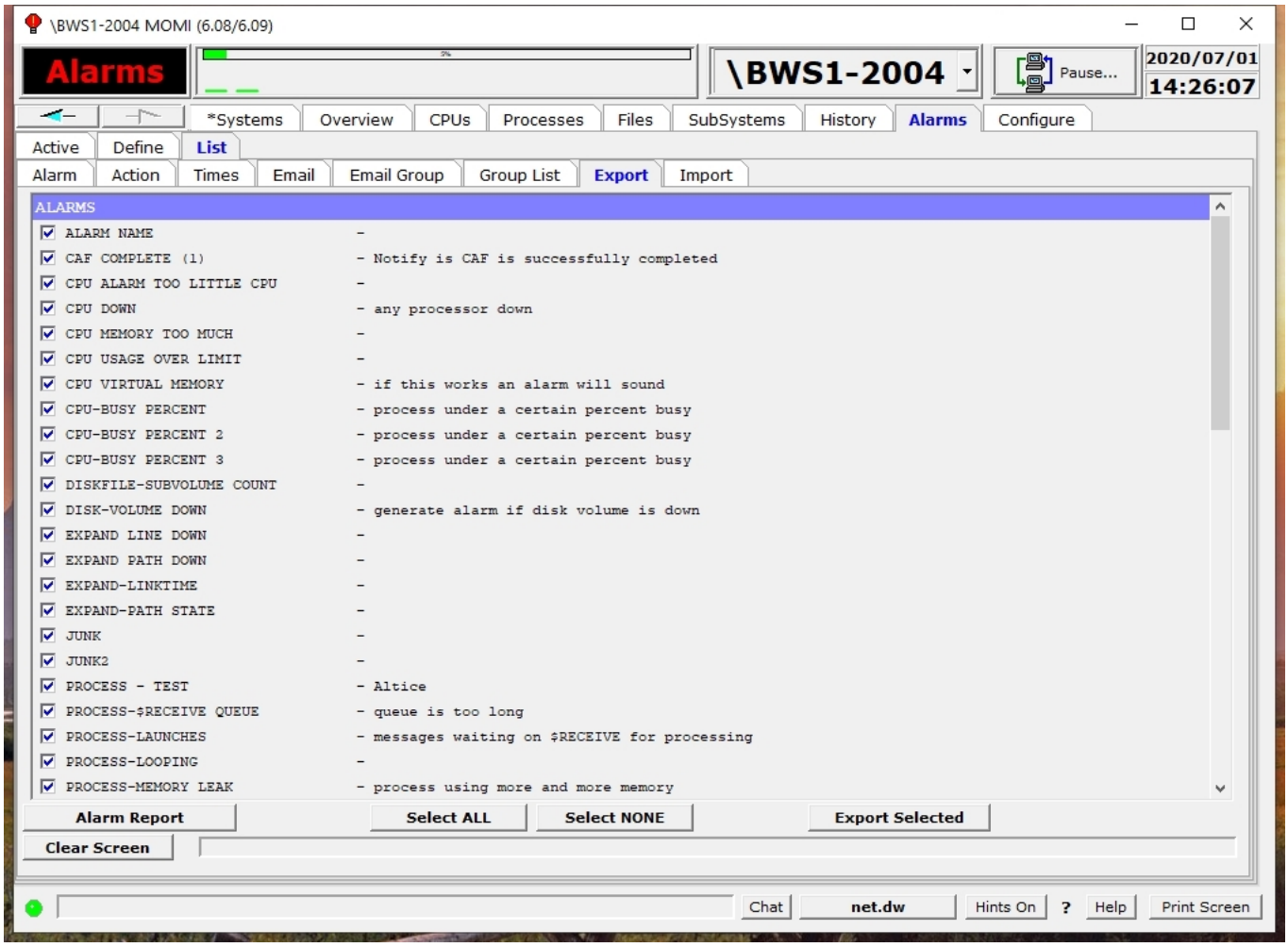


The List / Group List screen displays all individual emails and groups of emails plus allows drag and drop operations to add or remove members.

To add an individual email to a group drag from the left side of the screen to the right and drop on a group of a member of a group. To remove a individual email from a group drag from the right to the left side of the screen and drop over any position on the left.

Groups are initially created on the screen Define / Email Group.

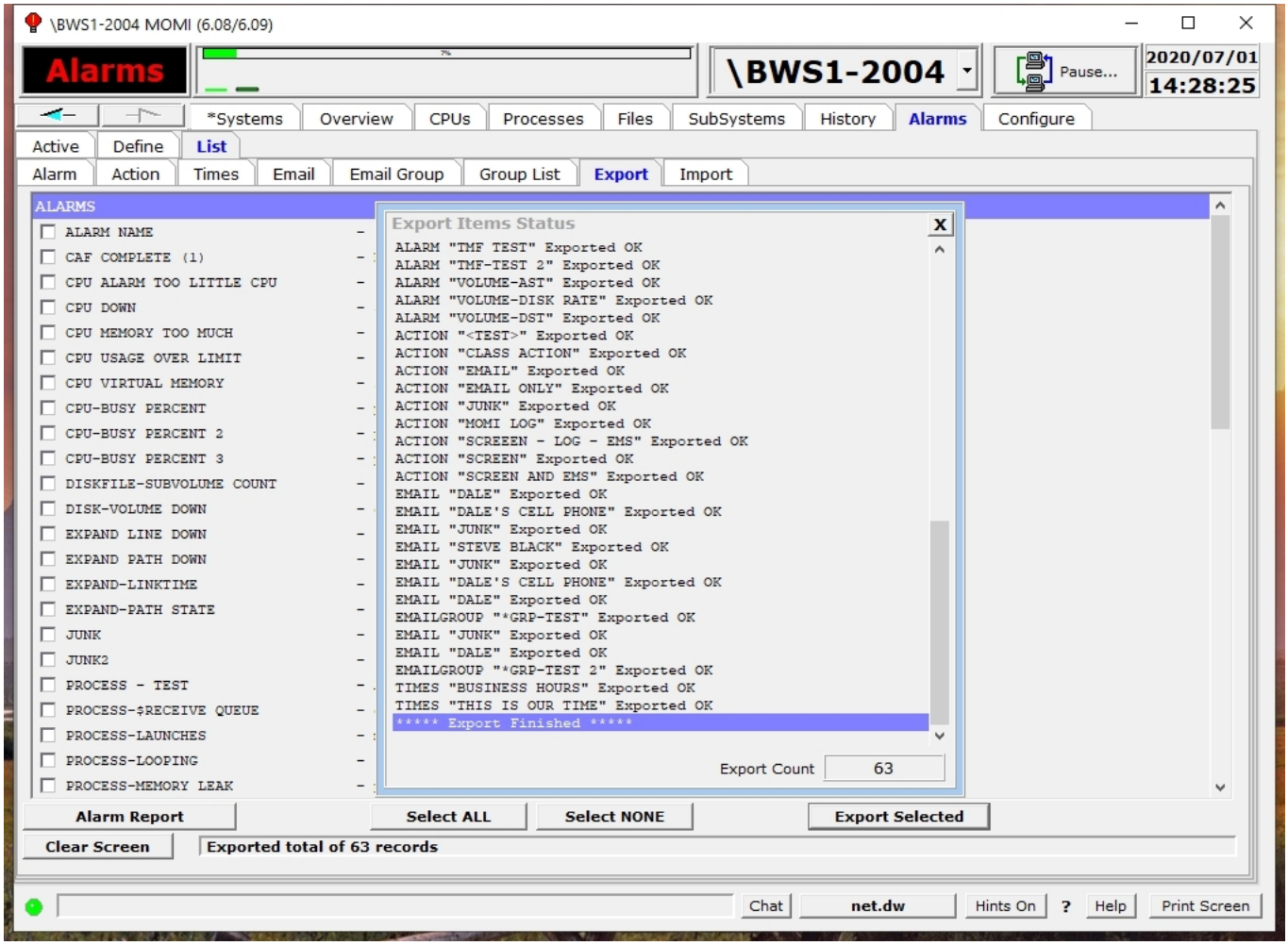
List / Export



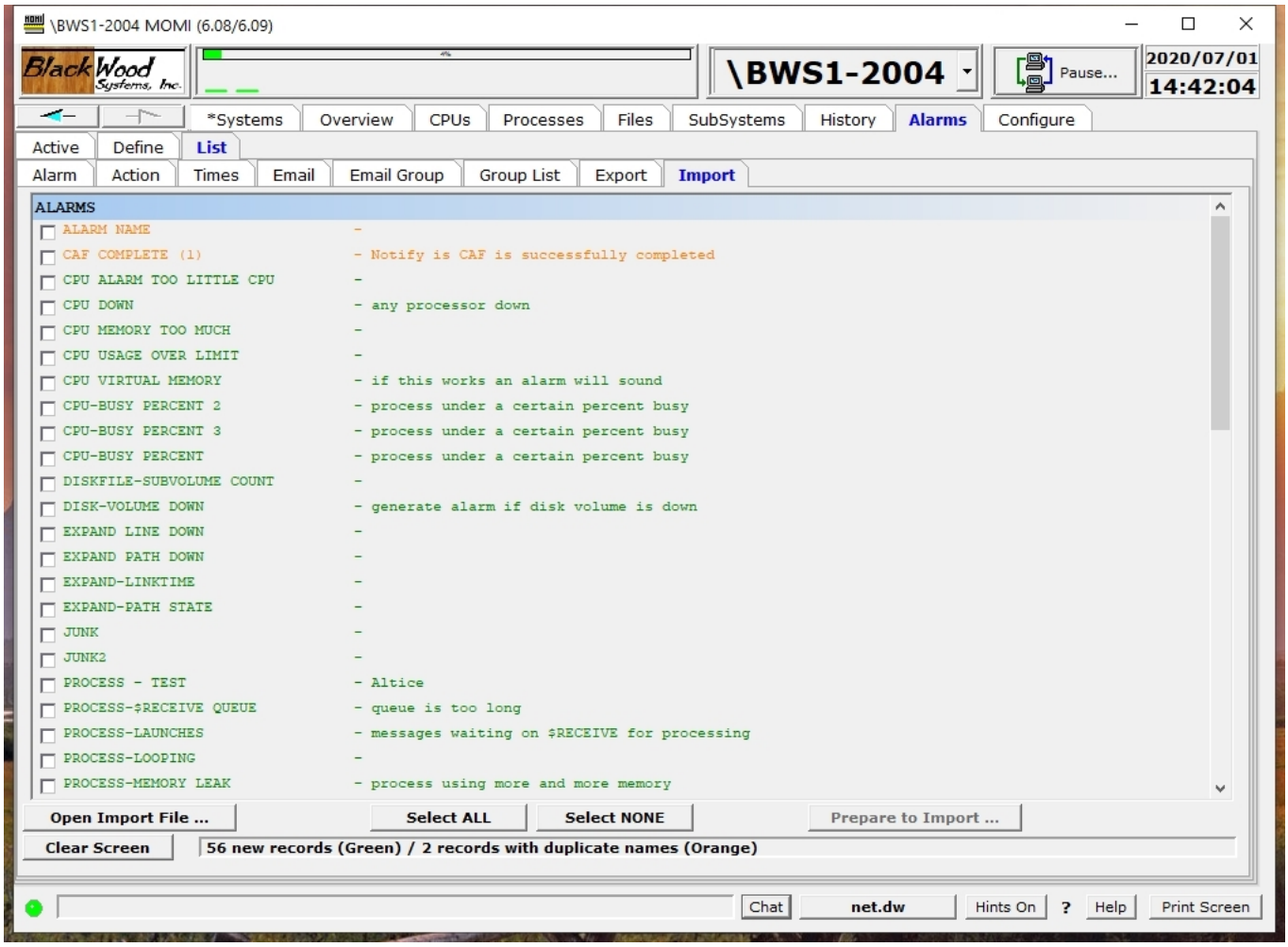
The List / Export screen provides a means to transfer to a text file Alarm and supporting definitions. The screen Import is used to bring the information back into the system, usually to a different MOMI environment.

Check the desired items to export and press Export Selected button. Items selected for Export will automatically contain any dependencies.

A pop-up window displays the result of the export operation.


\BWS1-2004 MOMI (6.08/6.09)
Alarms
\BWS1-2004
Pause...
2020/07/01
14:28:25
*Systems
Overview
CPUs
Processes
Files
SubSystems
History
Alarms
Configure
Active
Define
List
Alarm
Action
Times
Email
Email Group
Group List
Export
Import
ALARMS
ALARM NAME
CAF COMPLETE (1)
CPU ALARM TOO LITTLE CPU
CPU DOWN
CPU MEMORY TOO MUCH
CPU USAGE OVER LIMIT
CPU VIRTUAL MEMORY
CPU-BUSY PERCENT
CPU-BUSY PERCENT 2
CPU-BUSY PERCENT 3
DISKFILE-SUBVOLUME COUNT
DISK-VOLUME DOWN
EXPAND LINE DOWN
EXPAND PATH DOWN
EXPAND-LINKTIME
EXPAND-PATH STATE
JUNK
JUNK2
PROCESS - TEST
PROCESS-$RECEIVE QUEUE
PROCESS-LAUNCHES
PROCESS-LOOPING
PROCESS-MEMORY LEAK
Export Items Status
ALARM "TMF TEST" Exported OK
ALARM "TMF-TEST 2" Exported OK
ALARM "VOLUME-AST" Exported OK
ALARM "VOLUME-DISK RATE" Exported OK
ALARM "VOLUME-DST" Exported OK
ACTION "<TEST>" Exported OK
ACTION "CLASS ACTION" Exported OK
ACTION "EMAIL" Exported OK
ACTION "EMAIL ONLY" Exported OK
ACTION "JUNK" Exported OK
ACTION "MOMI LOG" Exported OK
ACTION "SCREEEN - LOG - EMS" Exported OK
ACTION "SCREEN" Exported OK
ACTION "SCREEN AND EMS" Exported OK
EMAIL "DALE" Exported OK
EMAIL "DALE'S CELL PHONE" Exported OK
EMAIL "JUNK" Exported OK
EMAIL "STEVE BLACK" Exported OK
EMAIL "JUNK" Exported OK
EMAIL "DALE'S CELL PHONE" Exported OK
EMAIL "DALE" Exported OK
EMAILGROUP "*GRP-TEST" Exported OK
EMAIL "JUNK" Exported OK
EMAIL "DALE" Exported OK
EMAILGROUP "*GRP-TEST 2" Exported OK
TIMES "BUSINESS HOURS" Exported OK
TIMES "THIS IS OUR TIME" Exported OK
***** Export Finished *****
Export Count
63
Alarm Report
Select ALL
Select NONE
Export Selected
Clear Screen
Exported total of 63 records
Chat
net.dw
Hints On
?
Help
Print Screen

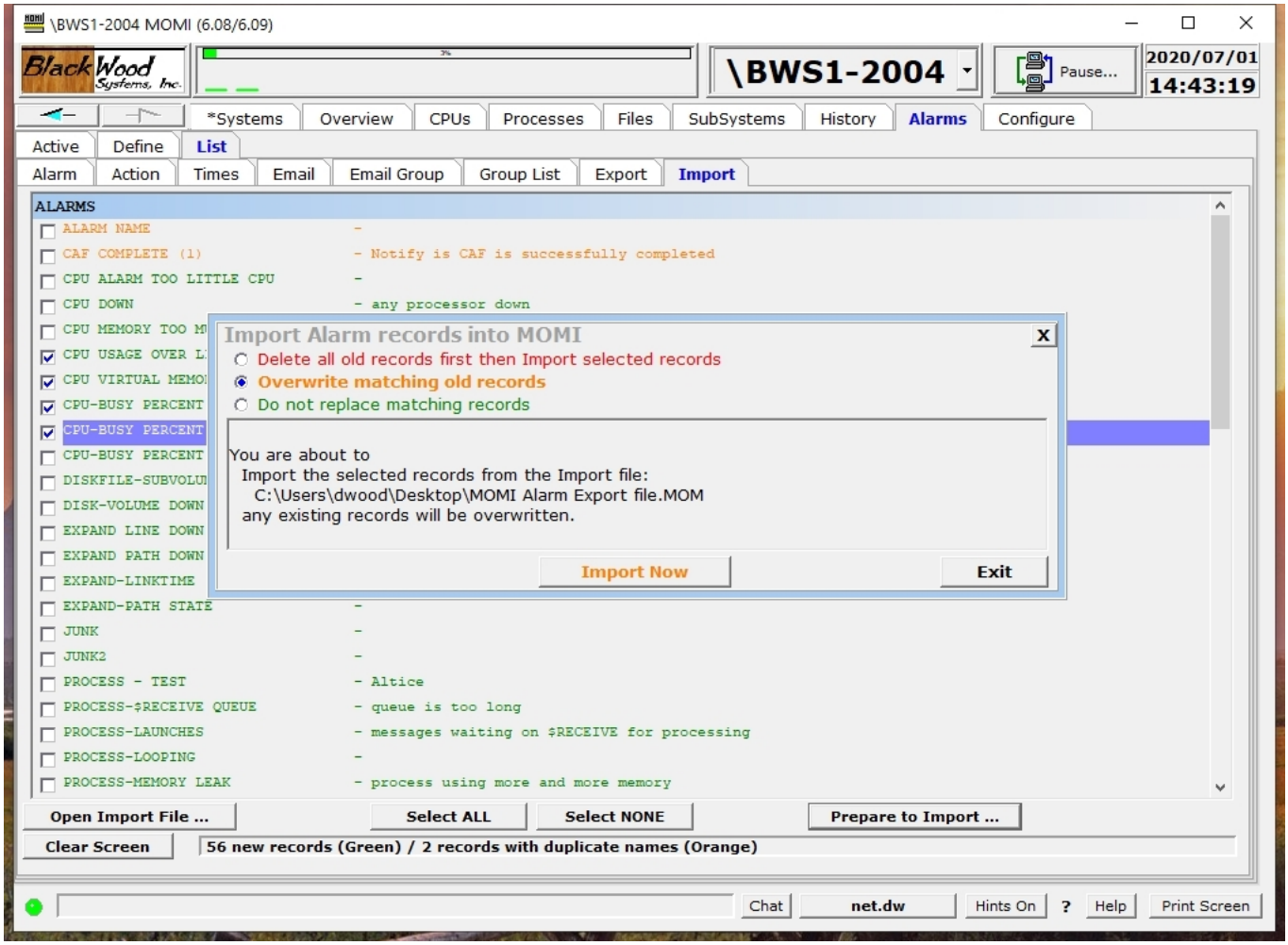
List / Import



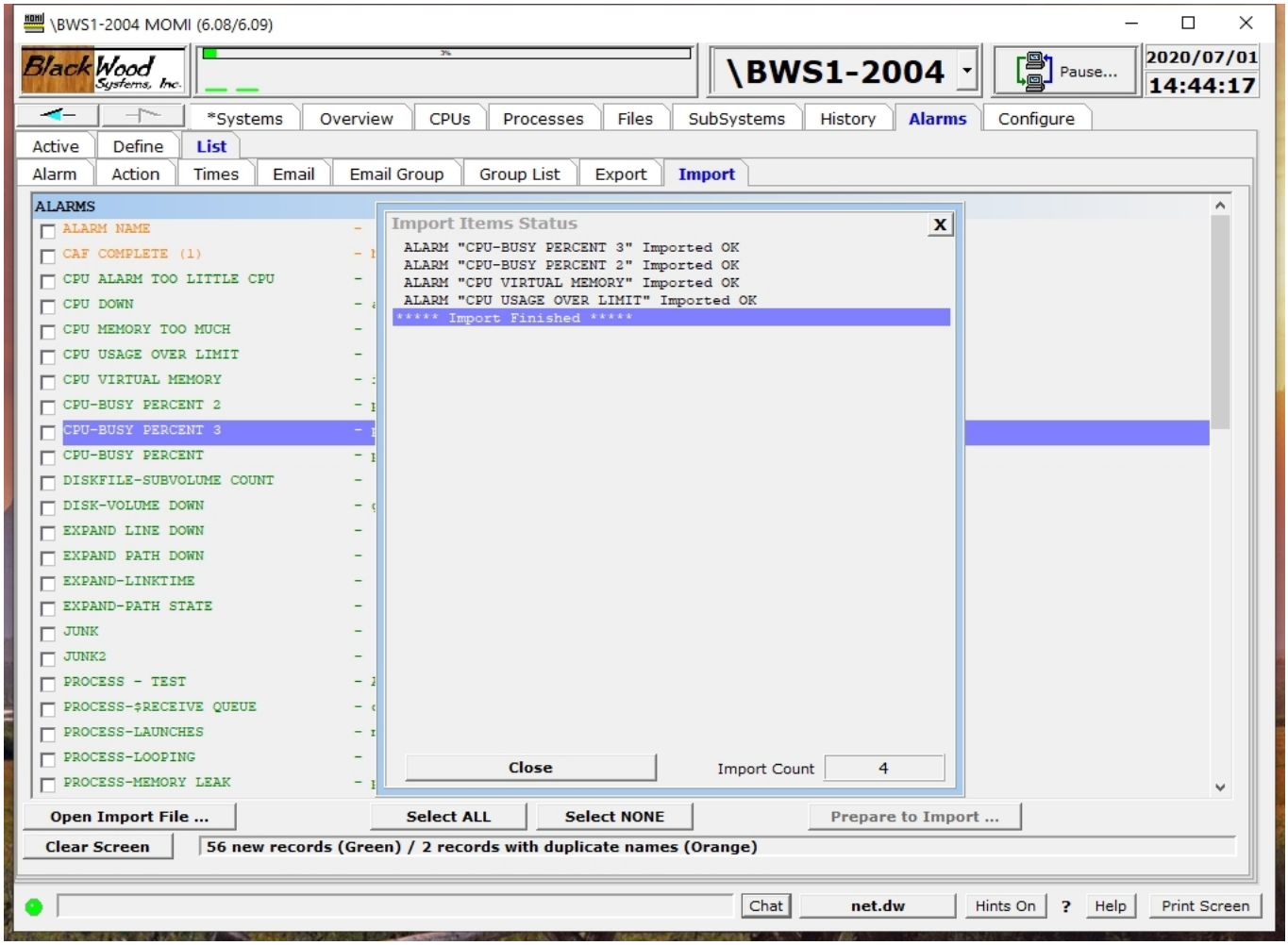
The List / Import screen provides a means to transfer from a text file Alarm and supporting definitions. This file is created on the screen Alarms / Export.

The import operation involves two phases. The first phase is to read the import file and second to select items to import operation.

Press the button **Open Import File...** and selected the desired text file. The entries are read and displayed. Check the desired items to Import and press **Prepare to Import...**. Options are available to delete existing records prior to import, overwrite any matching records or do not replace matching records. Select option and press **Import Now**.

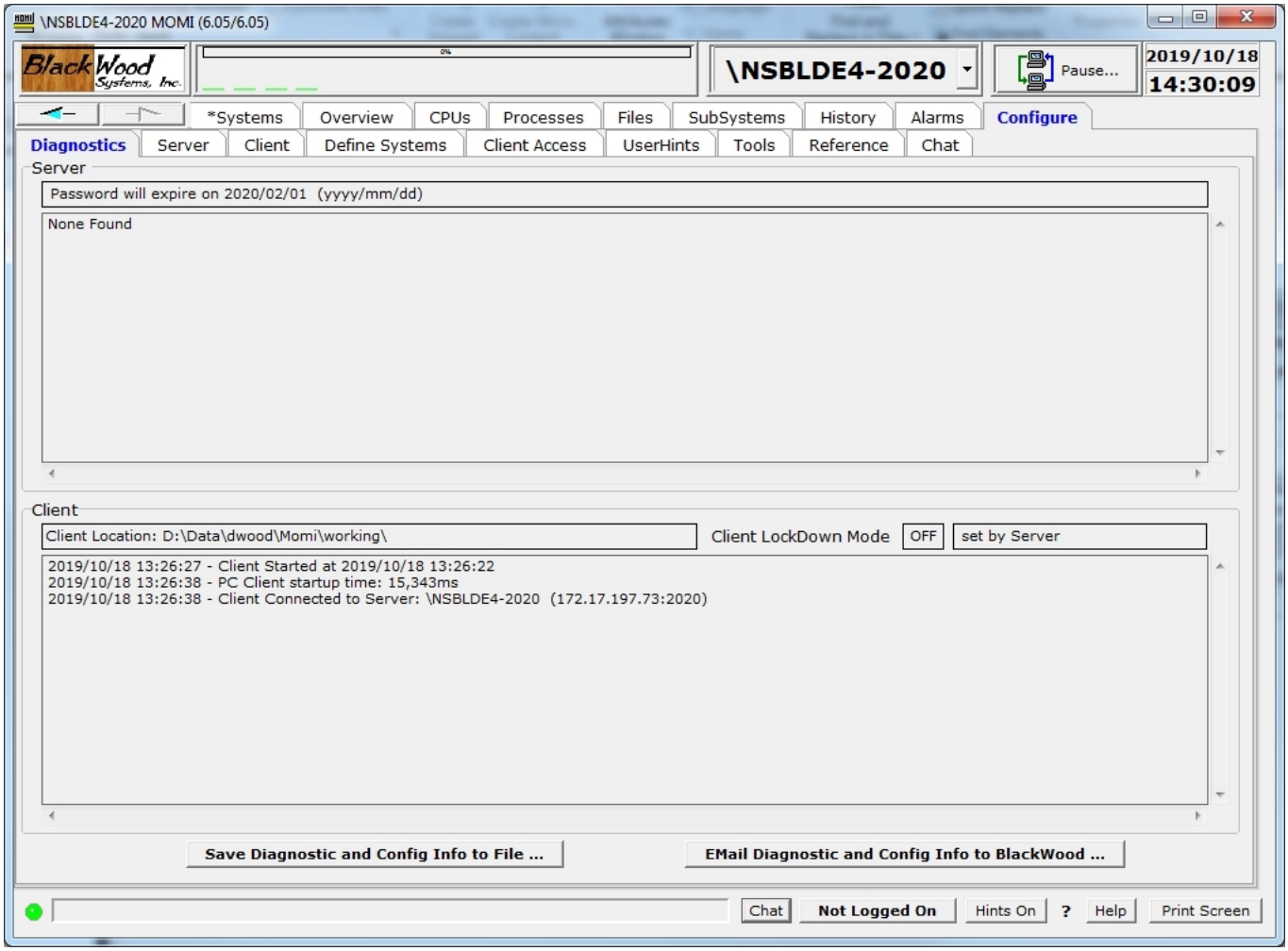


A pop-up window displays the result of the import operation.


\BWS1-2004 MOMI (6.08/6.09)
\BWS1-2004
Pause...
2020/07/01
14:44:17
*Systems
Overview
CPUs
Processes
Files
SubSystems
History
Alarms
Configure
Active
Define
List
Alarm
Action
Times
Email
Email Group
Group List
Export
Import
ALARMS
ALARM NAME
CAF COMPLETE (1)
CPU ALARM TOO LITTLE CPU
CPU DOWN
CPU MEMORY TOO MUCH
CPU USAGE OVER LIMIT
CPU VIRTUAL MEMORY
CPU-BUSY PERCENT 2
CPU-BUSY PERCENT 3
CPU-BUSY PERCENT
DISKFILE-SUBVOLUME COUNT
DISK-VOLUME DOWN
EXPAND LINE DOWN
EXPAND PATH DOWN
EXPAND-LINKTIME
EXPAND-PATH STATE
JUNK
JUNK2
PROCESS - TEST
PROCESS-$RECEIVE QUEUE
PROCESS-LAUNCHES
PROCESS-LOOPING
PROCESS-MEMORY LEAK
Import Items Status
ALARM "CPU-BUSY PERCENT 3" Imported OK
ALARM "CPU-BUSY PERCENT 2" Imported OK
ALARM "CPU VIRTUAL MEMORY" Imported OK
ALARM "CPU USAGE OVER LIMIT" Imported OK
***** Import Finished *****
Close
Import Count
4
Open Import File ...
Select ALL
Select NONE
Prepare to Import ...
Clear Screen
56 new records (Green) / 2 records with duplicate names (Orange)
Chat
net.dw
Hints On
?
Help
Print Screen

Configure

Configure / Diagnostics



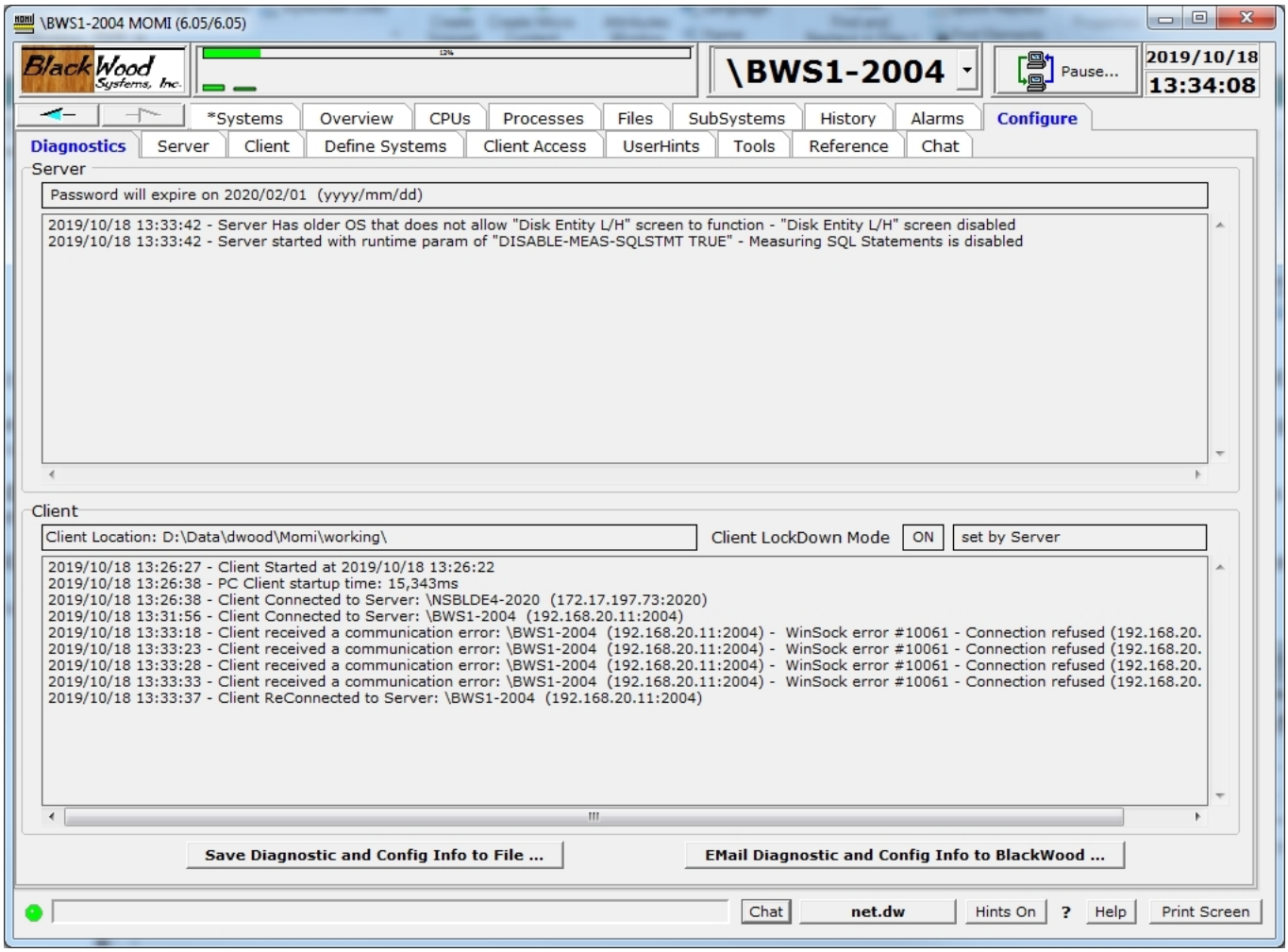
The Configure / Diagnostics screen presents the status of the client and server from a diagnostic point of view to help determine the cause of operational issues with the MOMI PC Client and MOMI server.

The top portion of the screen displays the Server status and the lower portion displays the client status. Server status includes the state of the MOMI Server password and general high level server information. Client status includes items such as Speech status or emulator not available.

The button labeled **Save Diagnostic and Config Info to File** allows the creation of an ASCII text file that contains Server, Client and Client PC configuration and status information helpful to BlackWood Systems in problem resolution and configuration assistance.

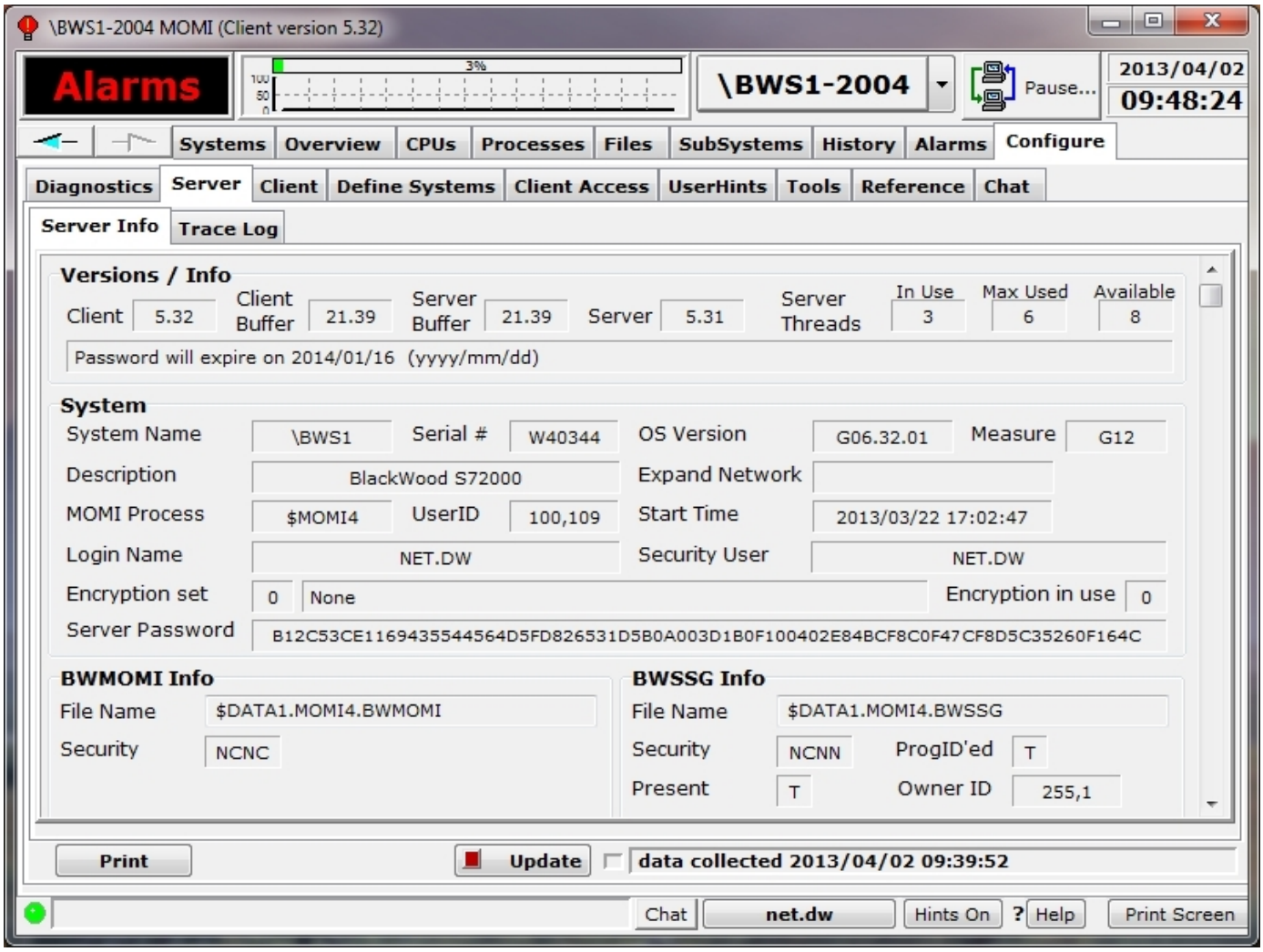
The button labeled **Email Diagnostic and Config Info to BlackWood** launches a utility to directly transmit a message with configuration information to BlackWood Systems.

The image below show an example of where the BWMOMI server was started with DISABLE-MEAS-SQLSTMT set to true.



Server

Server / Server Info



The Server / Server Info screen presents a variety of version, password status and configuration information from the server perspective. The window has a scrollable area to view all of the information (see below a sample image)

Place the mouse over the fields to obtain additional information. If the value is configurable, the tool-tip displays the CONFMOMI keyword.

Versions / Info displays various version information about the client, server and the message structures. The Server Threads fields shows how many TCP/IP connections (i.e. threads) are current, high-water and available between clients and the server. One client generally requires two threads.

System displays general information about the NonStop System environment and the MOMI password. See the Security sections General considerations and Client Access for additional information on MOMI security.

Priorities displays the configured for the main MOMI process and the various servers used by MOMI. See Process Priority considerations for additional information on how MOMI uses priorities to manage the workload.

Values Set displays the Interval in which MOMI gathers basic System information, various timeout parameters and limits for CPU consumption.

Values Returned displays internal errors and values reported by MOMI and/or MOMI servers.

Flags displays settings of flags used to alter the operation of MOMI. Some flags are for internal development and diagnostic purposes.

Expand displays timer values associated with MOMI collecting Expand information.

Measure displays timer values associated with the collection of MEASURE information for screens such as Files / Disk Entity.

Alarm displays information associated with resources available an configuration associate with Alarms for MOMI. See Alarm Overview for additional information.

SMTP TCP/IP Name is the process used by MOMI for conversing with the SMTP server. This TCP/IP process definition is separate from the normal TCP/IP processes used by the MOMI server for listening for inbound connections from the MOMI PC Client. Defines the TCP/IP process name that the MOMI uses to access the SMTP server which MOMI uses to send EMAIL. SMTP is outbound email.

SMTP Server Address defines the name or IP address of the SMTP server which MOMI uses to send EMAIL. This is the server MOMI uses to send outbound email. Note that the NonStop System must have the resolver active if a DNS-Name is used.

Domain Name defines the users domain name which MOMI uses in alarm email. Specifying this keyword activates in the SMTP (i.e. outbound email) protocol the HELO protocol. Some outbound email servers may require the usage of this parameter in conjunction with SPAM detection.

EMail Address From defines the return email address which MOMI used in alarm email. This email address should be valid monitored address so that problems in sending

email or replies from users receiving email are seen.

Email Subject has three possibilities:

1. If the CONFMOMI keyword is not present, the subject reads: MOMI Alarm
2. Use the CONFMOMI keyword and specify the desired subject line.
3. Use the CONFMOMI keyword but and specify an empty double quote (i.e. ""). This causes the first line of the alarm message to appear as the subject (in other words the "fixed subject" is eliminated).

ALARMS-MAXIMUM-PER-DEFINITION defines the maximum number of alarms that may be generated from a single alarm definition.

ALARM-EMS-EVENT-NBR defines the message number used for the EMS events generated for alarm purposes.

ALARM-SMTP-SERVER-TIMEOUT determines the maximum amount of time to wait for I/O operations to the external SMTP server.

ALARM-BREAKPOINTS-MAXIMUM determines how many CPU breakpoints, a resource for debugging / Inspect purposes, are available with the NonStop processors. Older systems had a fixed limit. This value is used in conjunction with the MOMI alarm CPU / Breakpoints.

TCP/IP shows the TCP/IP process name, port and IP address that the MOMI server is 'listening' on. See the CONFMOMI keyword TCPIP-LISTEN for additional information.

Spooler / SpoolerPlus shows the object file names which are considered Spooler supervisors. Spooler subsystems cannot pro-grammatically identify themselves. MOMI uses these entries to identify supervisors by object file name. See the CONFMOMI keywords FNAME-SPOOLER-n and FNAME-SPOOLER-PLUS-n for additional information.

I/O Stuck Checking shows the object file names which are excluded from the I/O stuck logic. See the CONFMOMI keyword IO-STUCK-EXCLUDE-n for additional information.

SNTP displays configuration for Simple Network Time Protocol or the ability for MOMI to maintain the time on the System. See Adjust System time via SNTP for additional information.

The **MOMI Default Work Location** section shows the $vol.subvol where MOMI work files are created. See the CONFMOMI keyword DEFAULT-WORK-LOCATION for additional information.

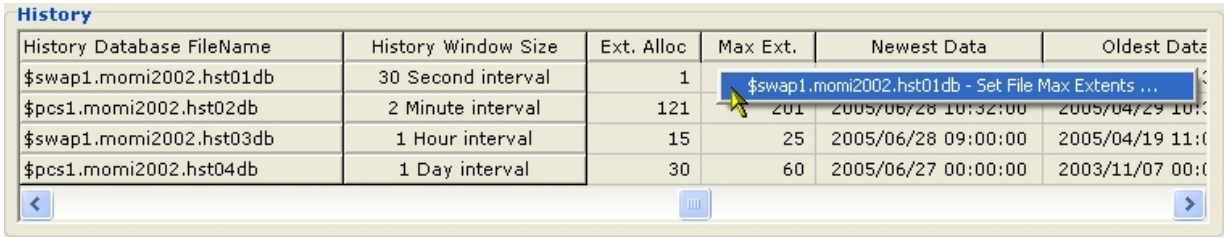
VHS Default Location shows the $vol.subvol where MOMI will initially takes a user when selecting a location for displaying VHS Log files. See the CONFMOMI keyword SELECTION-DEFAULT-VHS for additional information.

Alternate PATHTCP2 Location shows the file name used as an additional object for identifying a Pathway TCP process. See the CONFMOMI keyword PATHWAY-TCP for additional information.

BWSSG Info display the File Name and various attributes for the Super Group helper. This file is a copy of the BWMOMI program created by the administrator of the MOMI software during installation. This object is used unless MOMI is started under the Super.Super User ID. See the General considerations of Security for additional information.

History shows history files defined and their percentage full. MOMI automatically deletes the oldest data from the file to allow inserting new history data. Running a history file at 100% full does not cause errors. See Enabling History for additional information.

The extents for a history file may be altered by right-clicking on a value in the Max Ext. (Maximum Extents) column.



Versions / Info

Client	Client Buffer	Server Buffer	Server	Server Threads	In Use	Max Used	Available
5.32	21.39	21.39	5.31		2	6	8

Password will expire on 2014/01/16 (yyyy/mm/dd)

System

System Name	\BWS1	Serial #	W40344	OS Version	G06.32.01	Measure	G12
Description	BlackWood S72000			Expand Network	
MOMI Process	$MOMI4	UserID	100,109	Start Time	2013/03/22 17:02:47
Login Name	NET.DW			Security User	NET.DW
Encryption set	0 None			Encryption in use	0
Server Password	B12C53CE1169435544564D5FD826531D5B0A003D1B0F100402E84BCF8C0F47CF8D5C35260F164C				

BWMOMI Info

File Name	$DATA1.MOMI4.BWMOMI
Security	NCNC

BWSSG Info

File Name	$DATA1.MOMI4.BWSSG		
Security	NCNN	ProgID'ed	T
Present	T	Owner ID	255,1

CNF01DB Location

File Name	$DATA1.MOMI4.CNF01DB

LOG01DB Location

File Name	$DATA1.MOMI4.LOG01DB

MOMI Work Location

Vol.SubVol	$DATA1.MOMI4

VHS Default Location

Vol.SubVol	$DATA1.VHSLOG

Alternate PATHTCP2 Location

File Name	$DATA1.DALE.PATHTCP2

Priorities

MOMI Start	160	MOMI Current	160	Collector Master	159	Expand	157
		Process Action	158	EMS	91	Batch	90
Alarm EMS	92	Alarm File	93	Alarm Database	156	Logging	94
HST01DB	95	HSTxxDB read	85	HSTxxDB delete	80	HSTxxDB consolidate	80

Values Set

Interval	5.00	CPU Limit	none	SPI I/O Timeout	5.00
TCP/IP Limit	none	CPU Limit History Consolidation	4	CPU Collector NoReport Timeout	15.00
TCP/IP Timeout	60.00	CPU Limit Alarm Detection	5	SQLSTMT wildcard	space
OSS API version	2	CPU Limit EMS	none	Max Count Context Allowed	30

Values Returned

			XMEM Pool Sizes		
Last SubType 30 Error	0	Error Count	0	18,565,184 In Use	50,000,072 Maximum	3,323 Blocks

Flags

History Active	T	Disable Logons	F	Client Accel	T	G06.26 or later	T
Alarms Active	T	Disable SQL Statement	F	Collect All Process	F	H06.04 or later	F
		Disable SQL Process	F	Disable TMF Report	F	H06.09 or later	F
		Disable OSS CPU	F	Disable ListLocks	F	H06.14 or later	F
Super.Super	F	Disable OSS NameServer	F	Client Initial Msg	F	------- Debug1/5 -------	
Super.Group	F	Disable MeasCounter Wrt	F	Logon View Job	T	F F T F F	

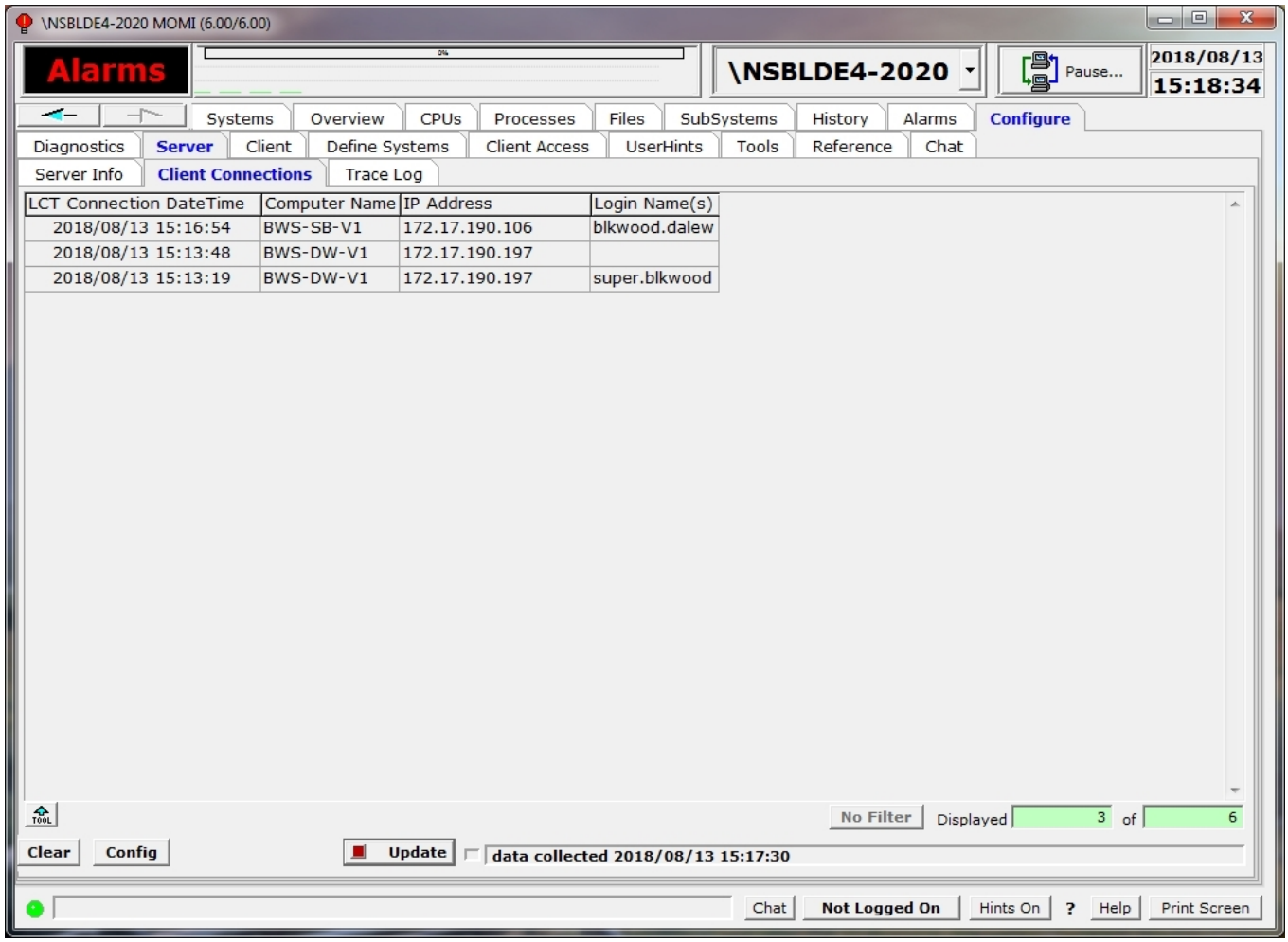
Expand

Expand I/O Delay	0.50

Measure

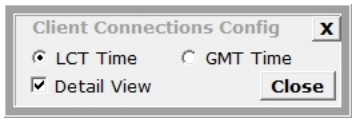
Starting Interval	10

Server / Client Connections

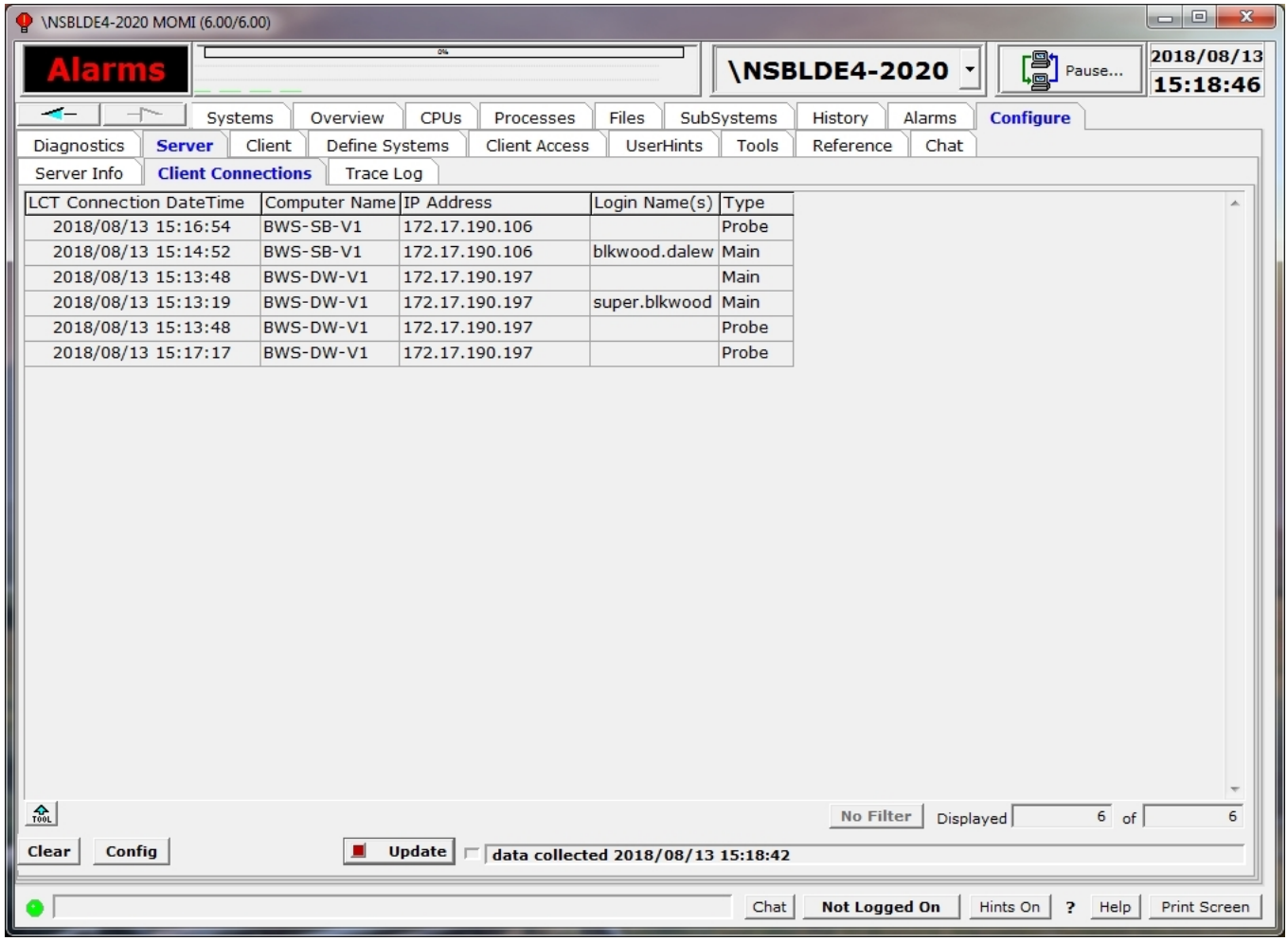


The Server / Client Connections screen display the current connections to the MOMI server.

The information displayed shows the time the client connected, the Windows computer name, IP address of the client and Login Names. The default view shows one line for each client currently connected.

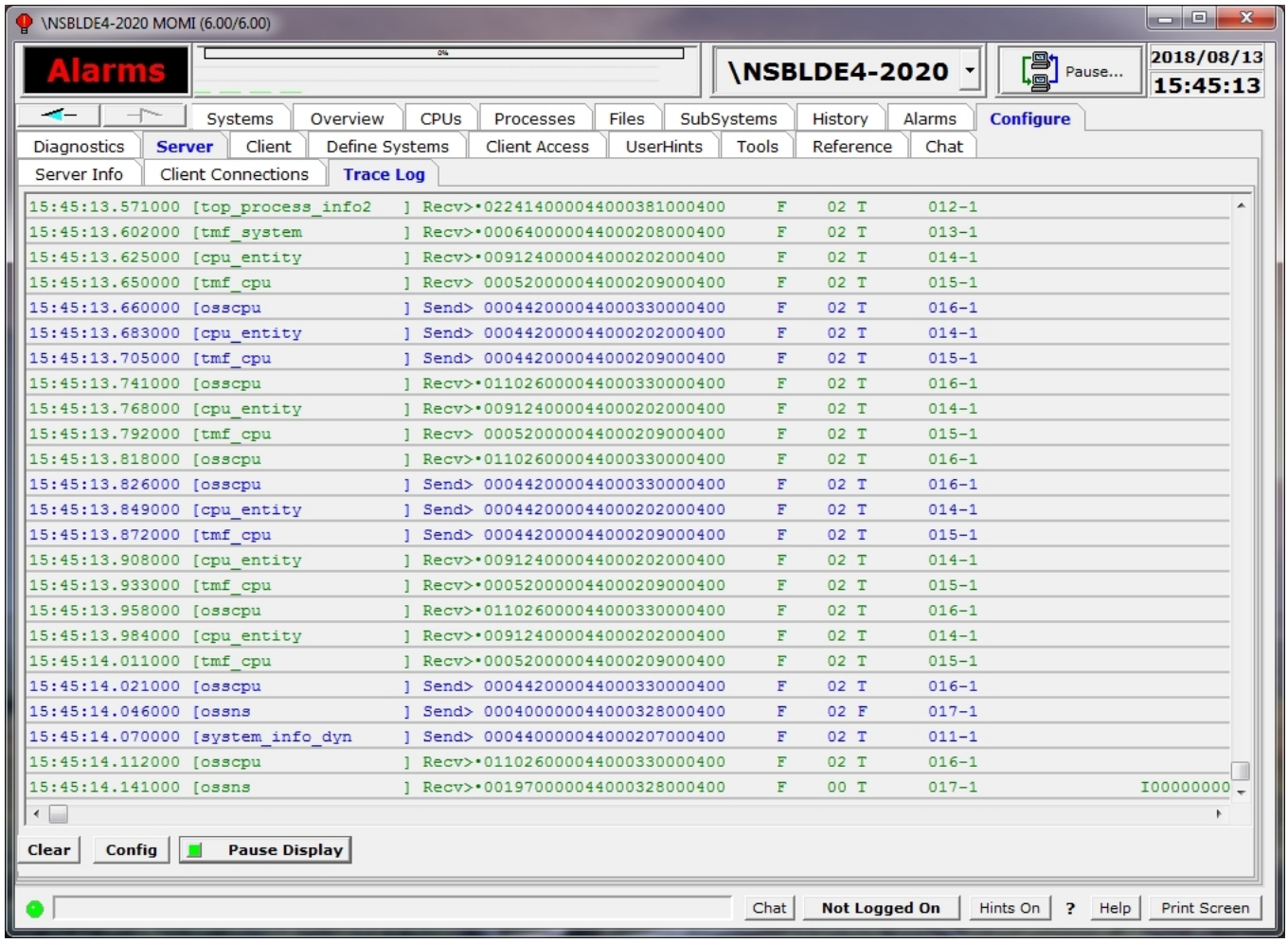


Pushing the **Config** button displays a pop-up to allow the selection of client connection time in either **LCT** or **GMT** and if the **Detail View** is displayed.



The MOMI PC Client established multiple socket connections to the MOMI server, depending on the screens or functions enabled. The detailed view displays all of these connections and their type.

Server / Trace Log

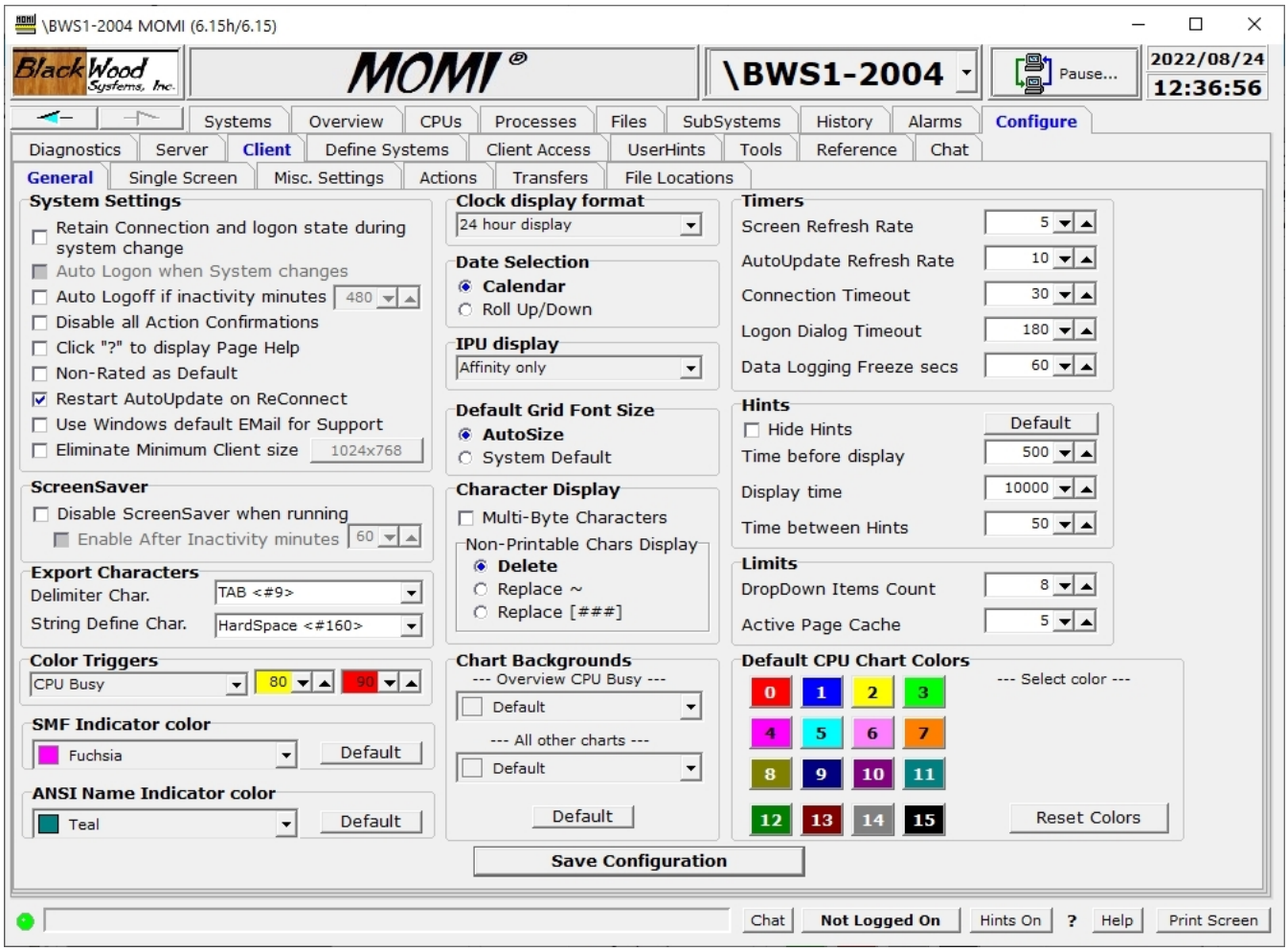


The Trace Log screen shows the I/O activity between the Client and server(s).

The screen is provided primarily for internal development purposes, but may be referenced during problem resolution. User ID and password information is separately and always encrypted and not viewable.

Client

Client / General



The Client / General screen shows the current state of attributes throughout the entire Client.

Click the check box or make the selection to alter the current state of an item. Place the mouse over an item to see a tool-tip for additional information about that setting.

After making any change, press **Save Configuration** to cause the settings to be stored in the MOMI.INI file.

System Settings alters global aspects of the program such as if the Windows screen saver should be deactivated while MOMI is running, disabling of various action confirmation messages or if AutoUpdate found on various screens should be re-enabled if the client connection to the server is severed and reconnected.

Settings notes

Retain Connection and logon state during system change
-the underlying connection to the MOMI server is maintained after changing to another system and any logon is retained.

Auto Logoff if inactivity minutes
- grayed out (unavailable) if over-ridden by Client Access
- does not function if an Internal emulator is active

ScreenSaver determines if Windows screen saver is disabled when the MOMI Client is running.

Export Characters determines the character used to delimit fields when MOMI data is exported. The defaults of TAB and HardSpace work with Excel and Access.

Color Triggers determines Red and Yellow thresholds for graphs such as CPU busy, CPU Queue and File Full. These triggers just affect graphs at the client level that specifically display this particular information.

Date Selection determines how dates are entered within Windows. The default of Calendar provides a pop-up selector calendar by month to directly click on a date within a month. The alternate selector of Roll up/down provides up/down arrows to increment or decrement the currently selected Year/month/day.

Character Display determines how non-printable characters are handled on various screens that display data from the NonStop System. By default, unprintable ASCII characters are deleted. Different options include replacing the character with a tilde (~) or the numeric value enclosed with brackets (for example [23]). Selecting the option Multi-Byte Characters turns off all character filtering which allows Windows to display multi-byte character sets such as Taiwan. Examples of screens that reference this setting are SubSystems / EMS Msgs and Enscribe / View File.

Chart Background determines the color of the furthest away (i.e. the area behind everything else) for all charts except Overview.

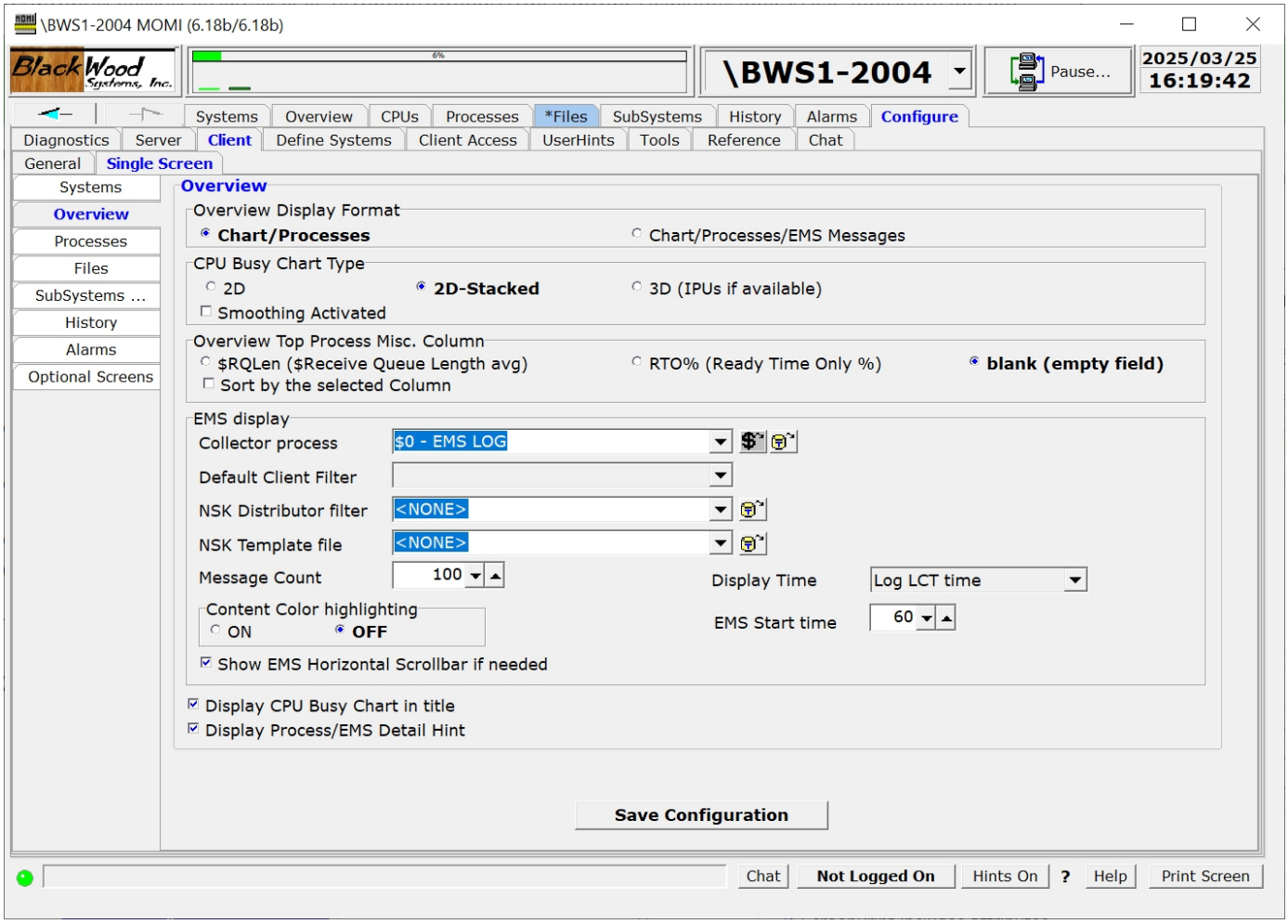
Timers determines general update and timeout rates. Screen Refresh Rate determines how frequently screens request data where they are automatically loaded with data. Note that some screens and completely manual in operation and others display data as made available from the MOMI server. AutoUpdate Refresh rate determines the frequency when an AutoUpdate option is present on the screen. Connection Timeout determines the maximum amount of time the client waits for data from the server. If this amount is exceeded the connection is dropped and reestablished.

Limits determines the number of previous entries to display when a Windows drop-down box is available.

Hints determines how often and for how long pop-up tool tips are displayed when the mouse hovers over a field that provides additional information.

Default CPU Chart Colors determines the selection of colors for charts referencing CPUs. Click on the CPU number to bring up a drop-down box of predefined color selections. Select Custom to display a window that allows the creation of a color.

Client / Single Screen

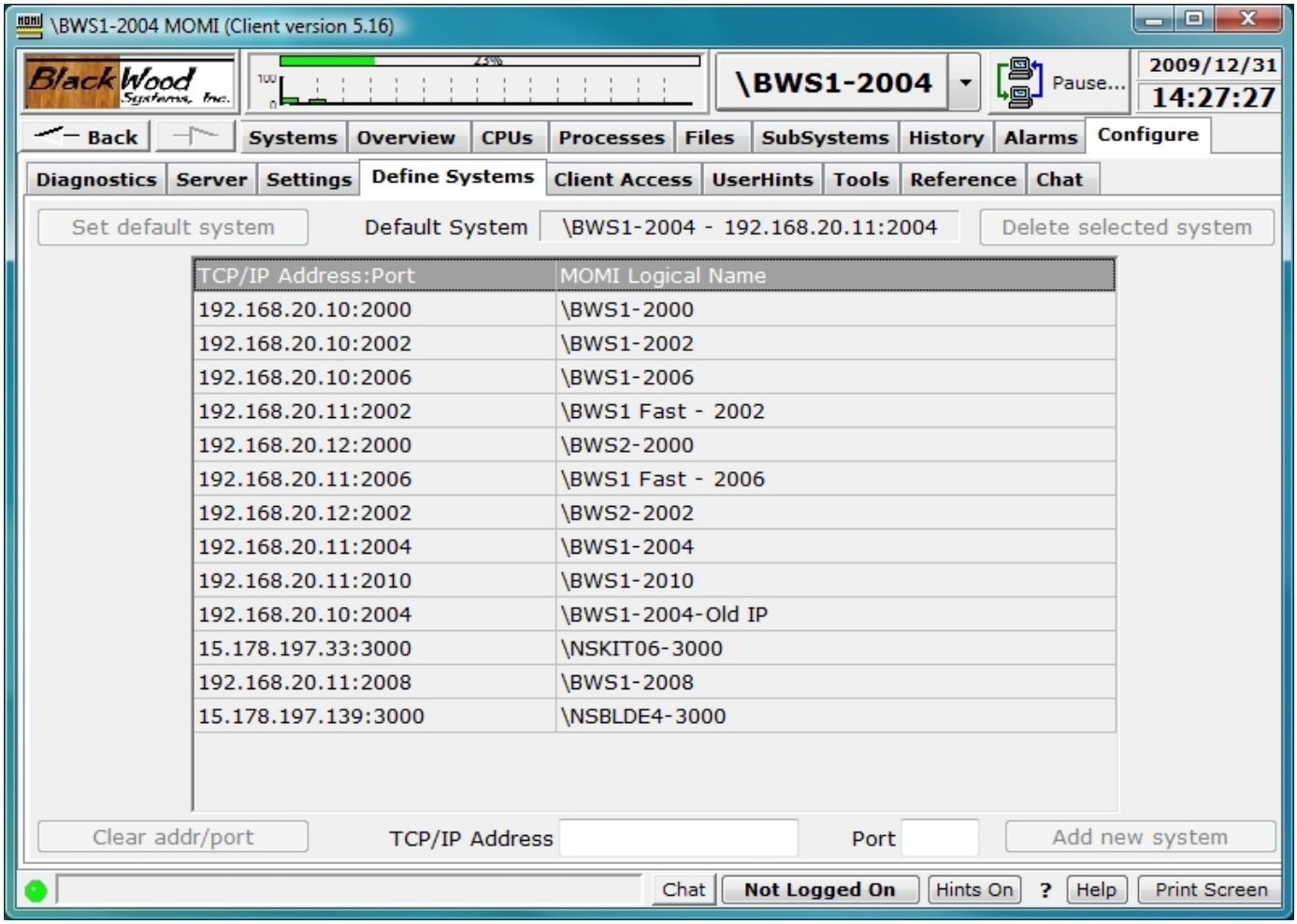


The Client / Single Screen shows the current state of attributes that are generally specific to a particular screen or group of screens within MOMI.

Click the sub-categories on the left hand side of the window to display available options. Place the mouse over an item to see a tool-tip for additional information about a particular setting.

After making any change(s), press **Save Configuration** to cause the settings to be stored in the MOMI.INI file.

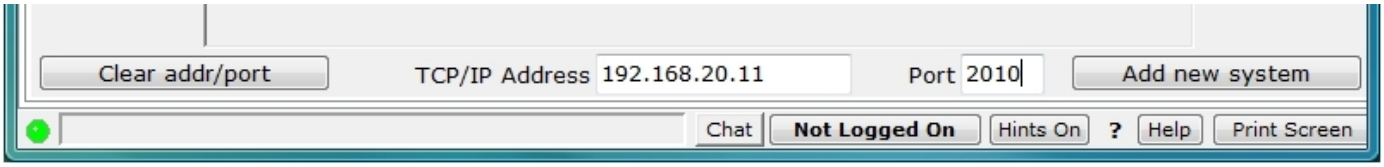
Define Systems



The Define Systems screen provides for the configuration and display of the MOMI servers available to the client. An entry, or connection, is composed of the TCP/IP address (or DNS name) and port of a 'listening' MOMI server. The MOMI server is configure to 'listen' for client connections using the CONFMOMI keyword TCPIP-LISTEN.

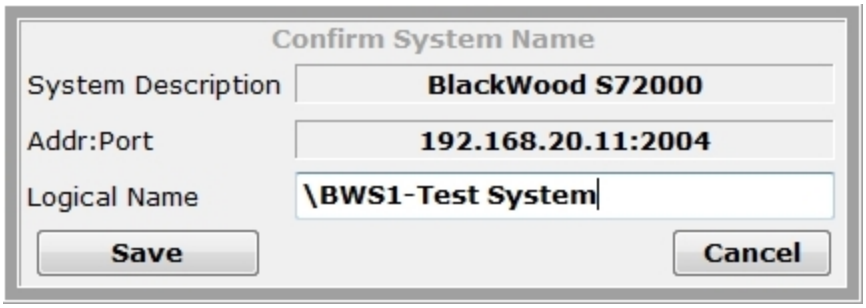
Existing entries may be deleted by clicking on it (highlights the line) and pressing **Delete selected system**. The System currently in use by the client may not be deleted. The default entry, or entry automatically connected at client start up, is configured by clicking on an entry and pressing **Set default system**.

The first time the client is started this screen is automatically displayed. Enter the TCP/IP address (or DNS name) and port then press '**Add new system...**' . The Client attempts communication with that MOMI server process.

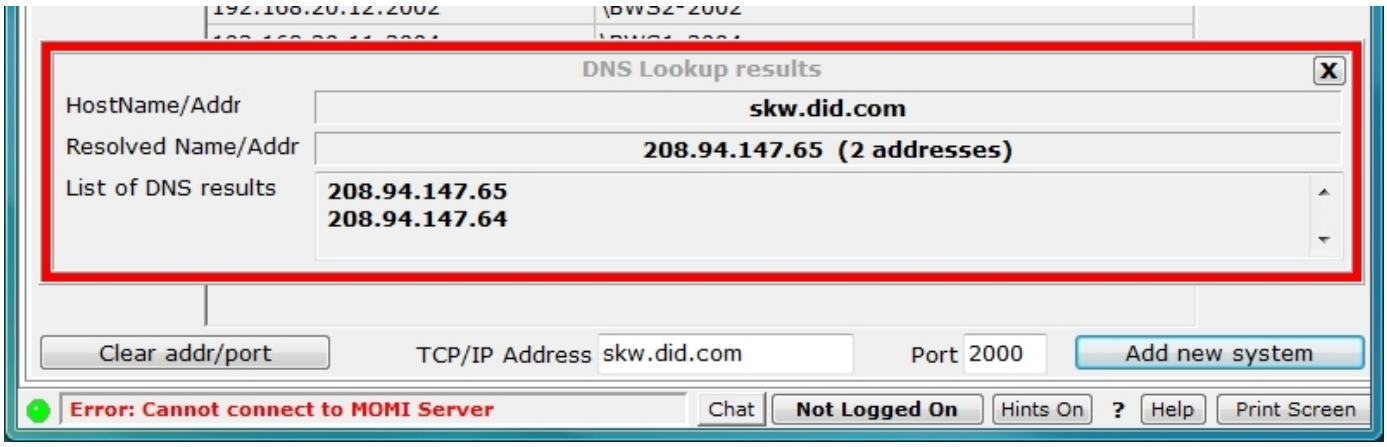


Connecting is displayed during the communications attempt.

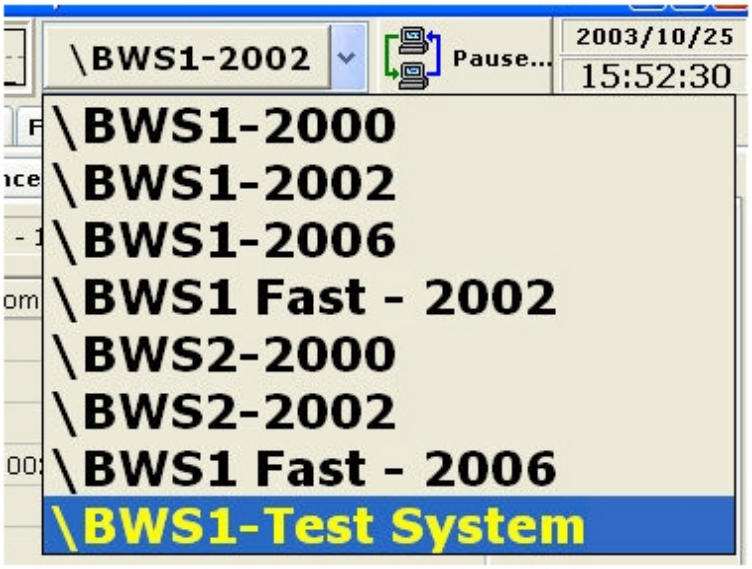
If communication is successful, a pop-up window displays to allow entry of descriptive text for this connection (below, the text *Test System* was added to the end of the NonStop System name that was automatically provided). Press **Save** to confirm and keep the entry.



If communications was unsuccessful, an error is displayed a pop-up window displays what address was used (if a DNS name was specified). See Network Information / Troubleshooting for additional information.



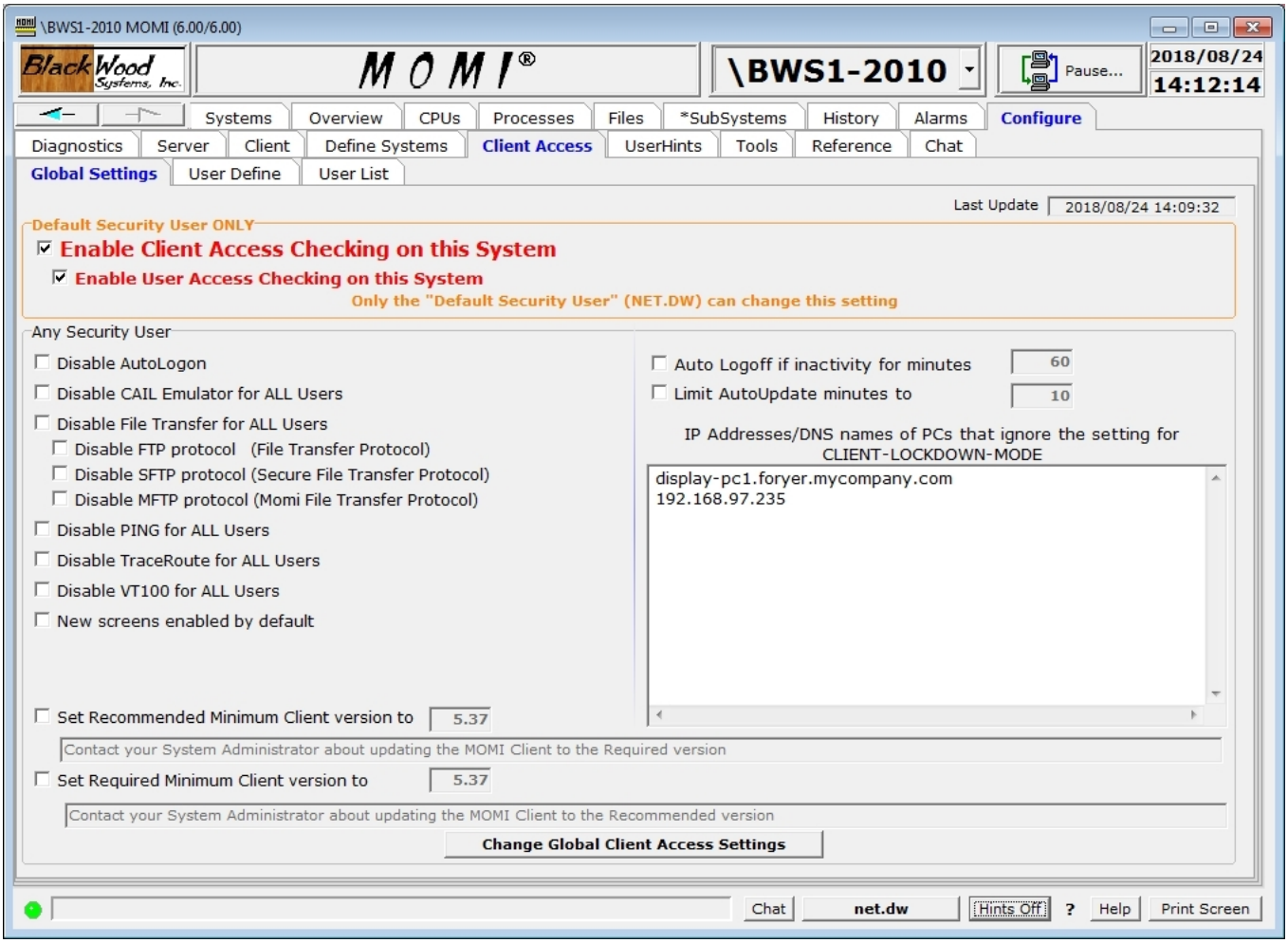
The first system defined causes the Client to connect to that entry and display data. Subsequent entries are made available on the drop down list at the top right of the screen.



Client Access

Client Access / Global Settings

see Client Access



The Client Access / Global Settings screen is used to manage Client Access and global limits. After making any change, press the **Change global Client Access Settings** button to save and activate any changes.

The Default Security User must be logged on to enable or disable Client Access. Any Security User may change the remaining settings.

Enable Client Access Checking on this System is the overall enable or disable of Client Access. Only the Default Security User may change this setting.

Enable User Access Checking on this System controls checking of specific users. If this item is enabled, all features of Client Access are in effect. If this item is not enabled, only the settings on this screen are effective.

Disable AutoLogon provides the ability to control if the Client issues a logon after changing systems if the user had already logged on in that session of MOMI. In shops where passwords may only be used one time, Disable AutoLogon should be checked.

Disable CAIL Emulator for ALL Users, **Disable FTP | SFTP | MFTP for ALL Users**, **Disable PING for ALL Users, Disable TraceRoute for ALL Users** and **Disable VT100 for ALL Users** provides the ability to control access to these tools.

New screens enabled by default, when checked, allows users to access MOMI PC Client screens added to a new software version without explicate authorization. If not checked, new MOMI screens must be authorized under the User Define tab.

Auto Logoff if inactivity for minutes enables the capability to cause the MOMI PC Client to automatically logoff from the NonStop System after the specified number of minutes occur and no activity occurs. Activity is defined as keys pressed in the client.

Limit AutoUpdate minutes to limits the amount of time screens directed by the user continue to update. Many MOMI screens provide data gathered for the benefit of all users and to perform other internal activity. However, screens that are specifically activated or gathering data as a result of user selection, such as the Process Detail, are limited by this setting. Screens that have the AutoUpdate option are also affected by this setting.

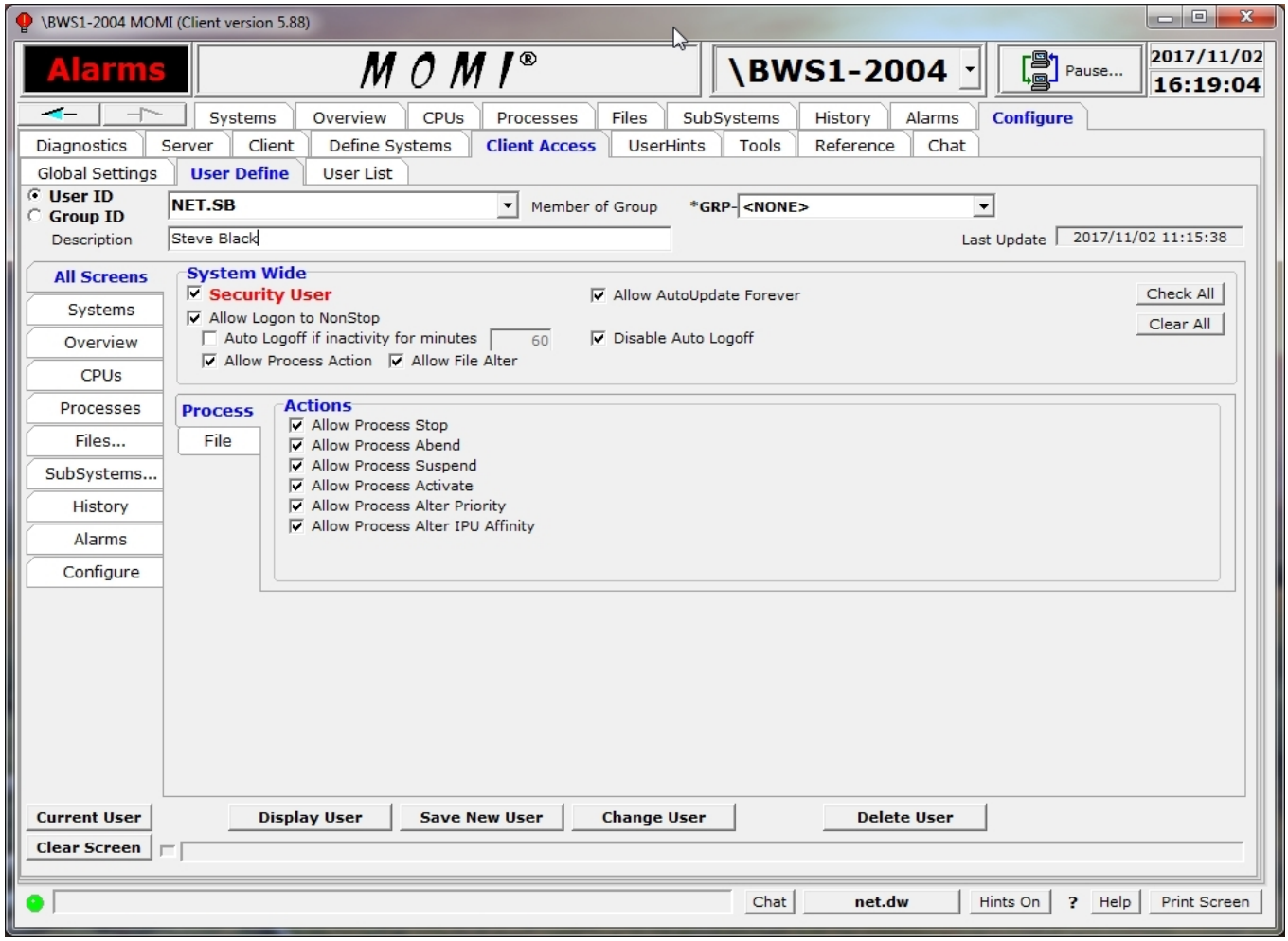
Set Recommended Minimum version to causes the display of a "by-passable" warning when the MOMI PC Client version is below the specified value. A value higher than the version of the client attempting to make a change is automatically reduced.

Set Required Minimum Client version to causes the display of an error messages and prevents further access if the MOMI PC Client version is below the specified value. A value higher than the version of the client attempting to make a change is automatically reduced (to prevent a lockout of all clients).

IP Addresses/DNS names of PCs that ignore the setting for CLIENT-LOCKDOWN-MODE
This area is used to define a list of IP Addresses and/or DNS names of PCs allowed to display data without logon. Typical usage would be for a PC used to display MOMI screens, for example Overview, in a lobby or other public display area where logon is not needed nor desired. Note that logon is still required to execute sensitive commands (such as stopping a process or deleting a file).

Client Access / User Define

see Client Access



The Client Access / Define screen is used to determine which screens and features within MOMI are available to individual users or groups of users.

In the lower left portion of the screen, click thought the sub-categories on the left hand side of the screen to see what options are available for the currently selected User ID or Group ID Change as needed. Place the mouse over an item to see a tool-tip for additional information about a particular setting.

Two types of ID's are used; a User ID and a Group ID. A User ID corresponds to a Guardian User or Safeguard Alias defined on the NonStop System. A Group ID is a MOMI specific entity used to assign specific settings to one or more User ID's.

No passwords are defined. All security validation is performed by the Nonstop Operating System.

Two special User ID's are provided (and automatically created) by MOMI :

- *NOT LOGGED ON* represents a MOMI PC Client that has not logged on. This defines the capabilities of the client when it initially connects to the server and until (or if) the user logs on.

- *USER NOT DEFINED* represents a logon to MOMI where the User ID has not been defined to Client Access. This "catch all" provides for a default level of configuration where the specific user ID is not known. This feature may be disabled by not checking **Allow Logon to Nonstop**.

To change an existing entry, first select if a User ID or Group ID is to be displayed in the upper left hand corner then select the existing entity in the drop-down box.

To add a new **User ID**, select **User ID**, enter the User ID or SafeGuard Alias, description and configure the various options.

To add a new **User ID** into a group, select **User ID**, enter the User ID or SafeGuard Alias, description and select the desired group in the drop-down box.

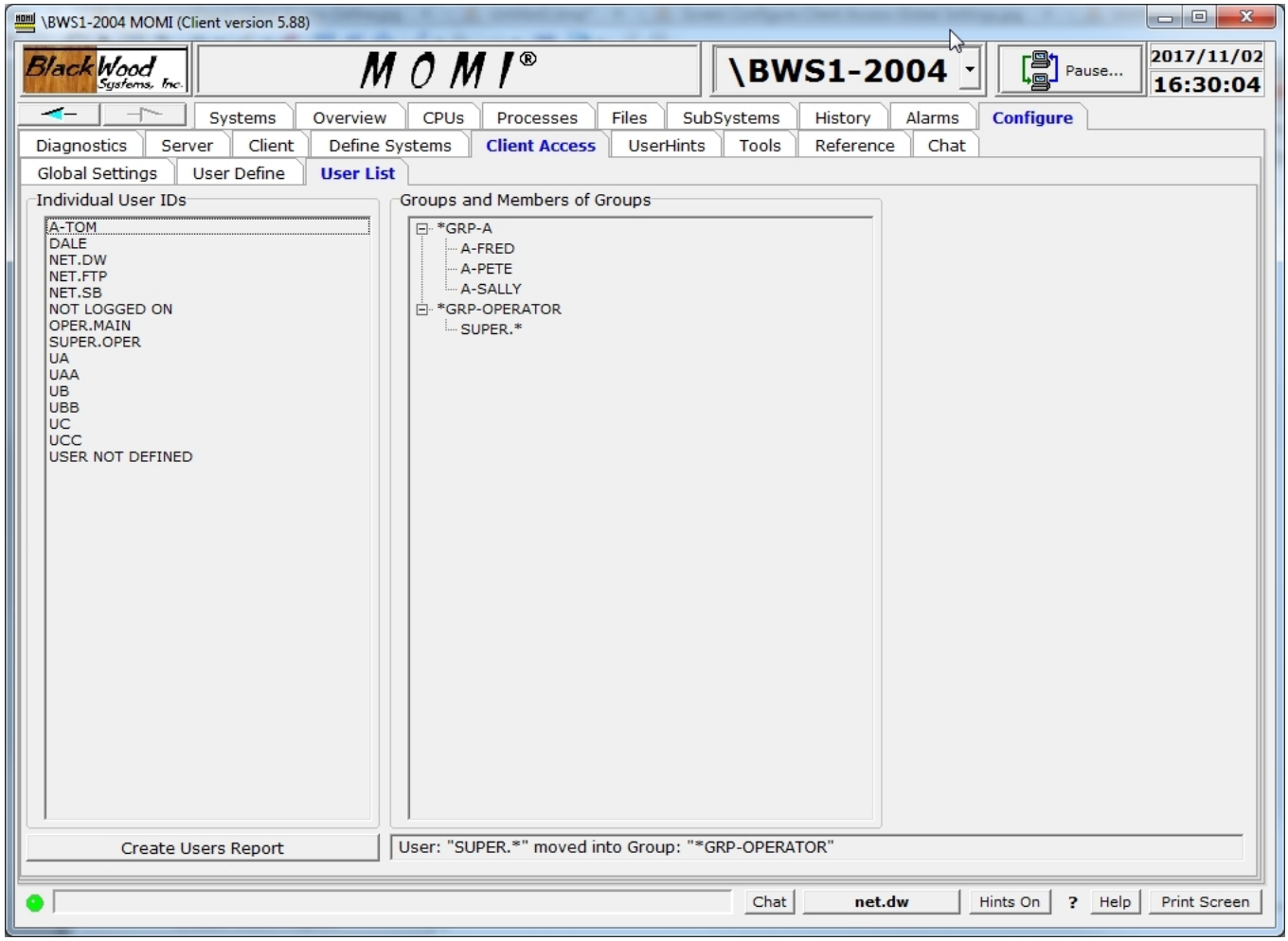
To add a new **Group ID**, enter the Group ID (an arbitrary name), description and configure the various options applicable to that group.

Individual users or Groups are given various permissions. The section System Wide area provides for general abilities such as allowing process actions or automatic Logoff. The tabbed area below selects what screens are available and other screen features.

A User ID or Group ID may be assigned as a Security User (in RED above). A User ID with the Security User attribute, have the same authority as the Default Security User except they may NOT change certain settings on the Global Settings screen such as disable of Client Access checking.

Client Access / User List

see Client Access



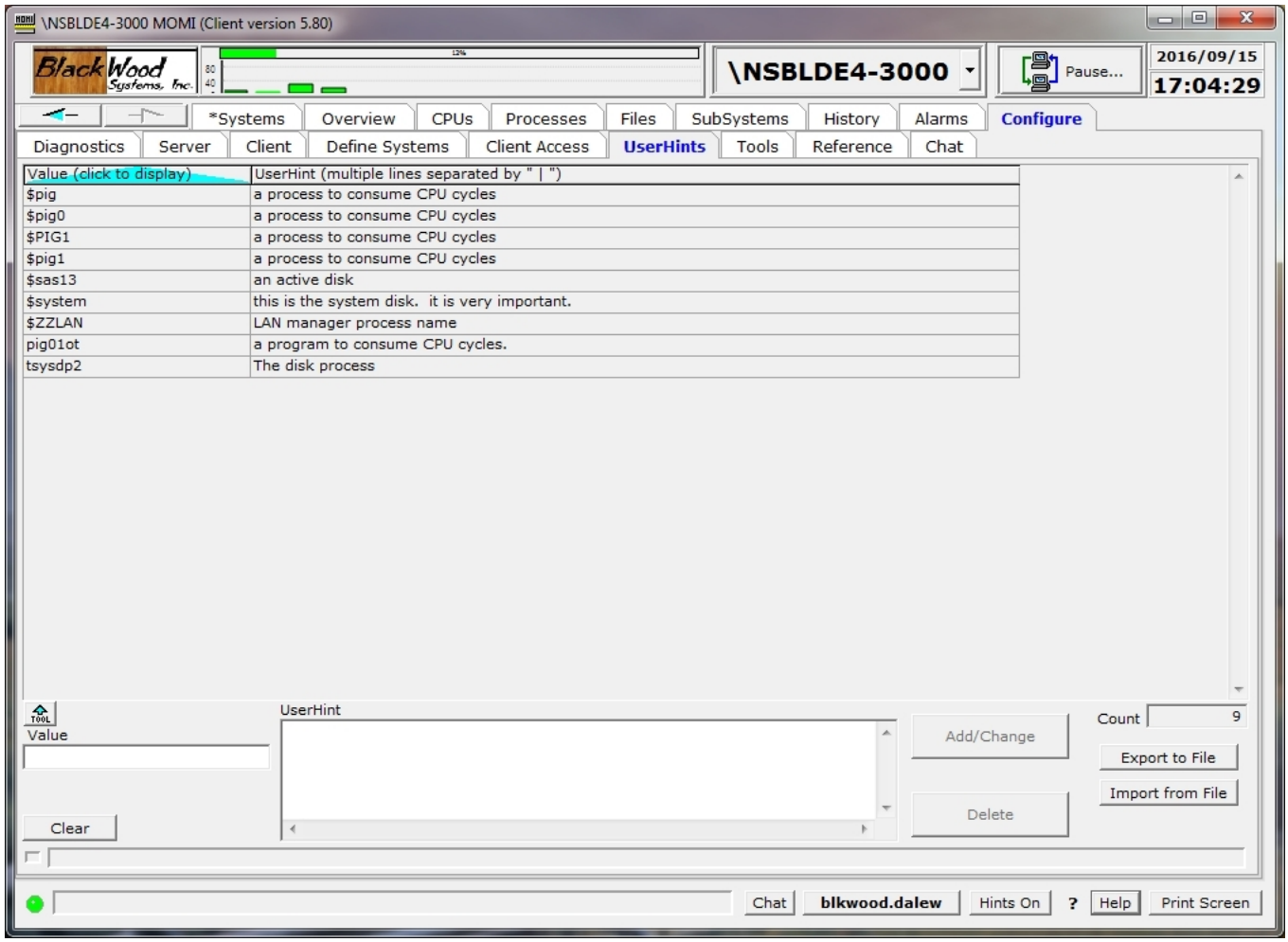
The Client Access / List screen displays the User and Group ID's established with MOMI. A report can be generated with the Create Users Report button.

Left-click on a User or Group ID to jump to Define screen. Click on the [+] to expand members of a group.

The screen also supports drag-and-drop to add or remove users within a Group. To add a user to a group, click and drag a User ID and drag it to (i.e. right over) a Group name. To remove a user from a group, click and drag a User ID under a Group and release it in anywhere in the Individual User IDs column.

Configure / UserHints

(updated client version 5.16 or later)



The UserHints screen presents hints defined by the customer and provides for maintenance activities (add/change/delete).

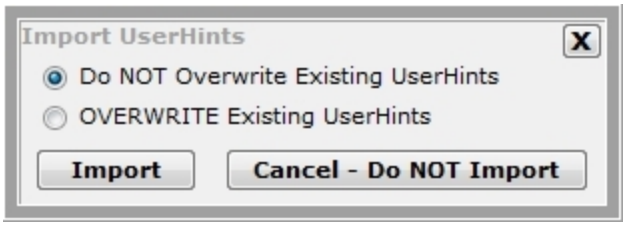
Throughout the MOMI PC Client Tool tips, or Hints, are provided 'all over the place'. This information is usually static based on that particular field. This screen allows the user to define a Value (i.e. text) that if it EXACTLY matches the current value displayed on the screen (in that particular cell or location) the supplied UserHint is added to the existing MOMI Tool Tip.

UserHints are displayed with a key value matches text within a field displayed on a MOMI screen. For example, the process name $ZPNS is an OSS Name Server. Enter $ZPNS in the Value field, the text "OSS Name Server" in the UserHint area then press **Add/Change**. When a field with a MOMI hint is displayed and matches $ZPNS, the displayed hint will add "OSS Name Server" to the displayed hint.

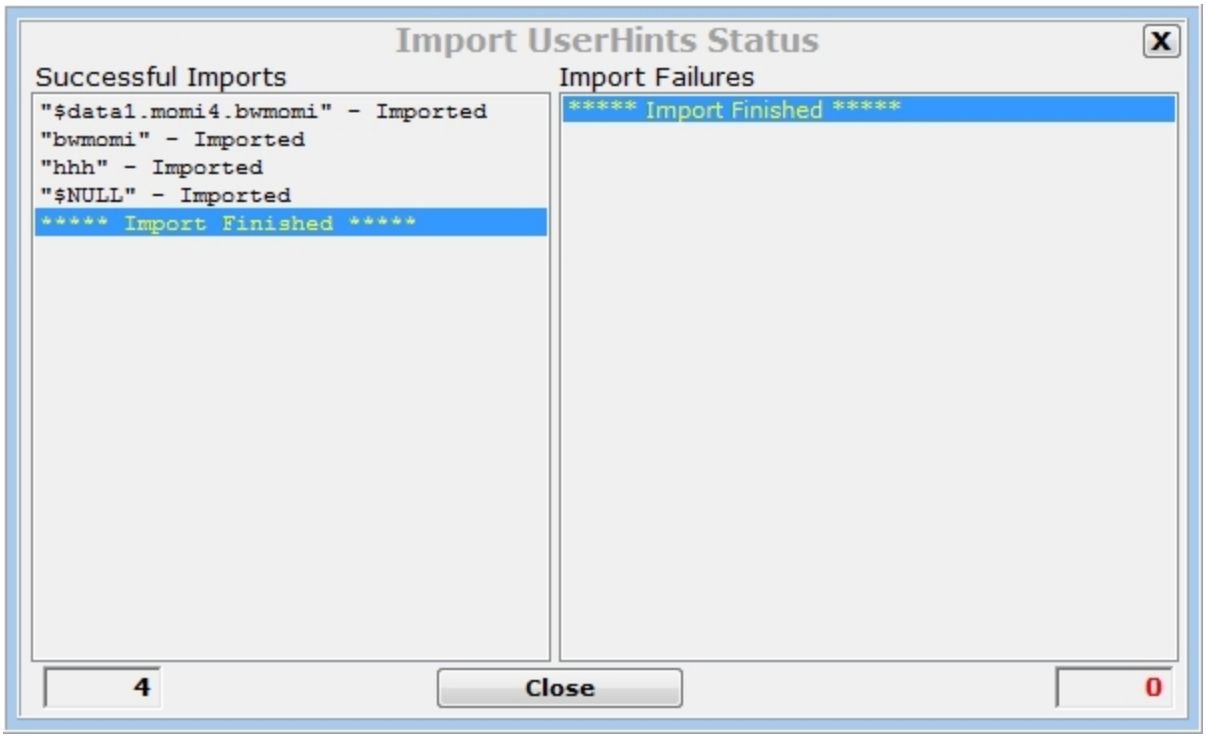
Existing hints may be removed by clicking on the hint and then pressing **Delete**.

UserHints may be exported from one system and imported into another. Press **Export to File** to create a UserHints file. Connect the MOMI PC Client to the desired system and then press **Import from File** to load the saved hints.

When Import is selected, the following window displays to determine if the imported hints replace existing. By default, existing UserHints are not replaced (i.e. not overwritten). Select OVERWRITE to cause existing hints to be replaced with a matching one in the Import file.



During the import process, the following resulting status screen is displayed:



The user many also create their own ASCII text file of Hints to import into MOMI. The file has the following format:

== comment line - ignored and not processed

<key>=<hint>

Where:

== - comment line, not processed
<key> - unique text to match, but not case sensitive
= - required separator
<hint> - text to display, optionally using <br> to create a multi-line displayed tool tip

Examples:

```
== file of user hints - created by Pete Williams
== updated by John Doe
==
10,33=Audit.Mike
$FRED=Fred Smith (IT - Software Developer)
$SALLY=Sally Field <br> QA Group <br> Building 2 <br> Fourth floor - Cube 1234
$NCP=Network Control Process
```

Special functions

<key>=

delete key

DELETE-ALL-RECORDS

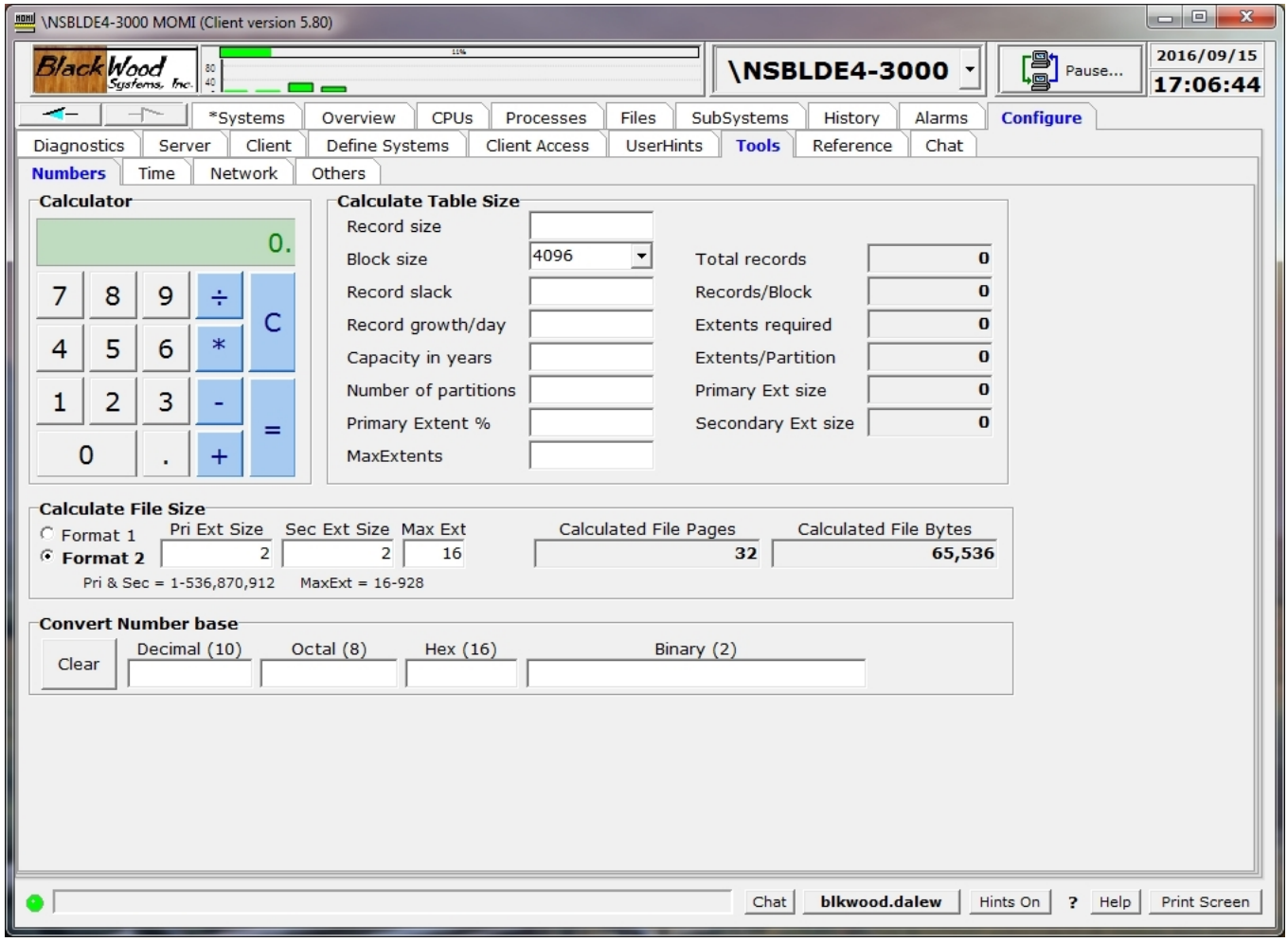
deletes all UserHints

The import file is processed sequentially.

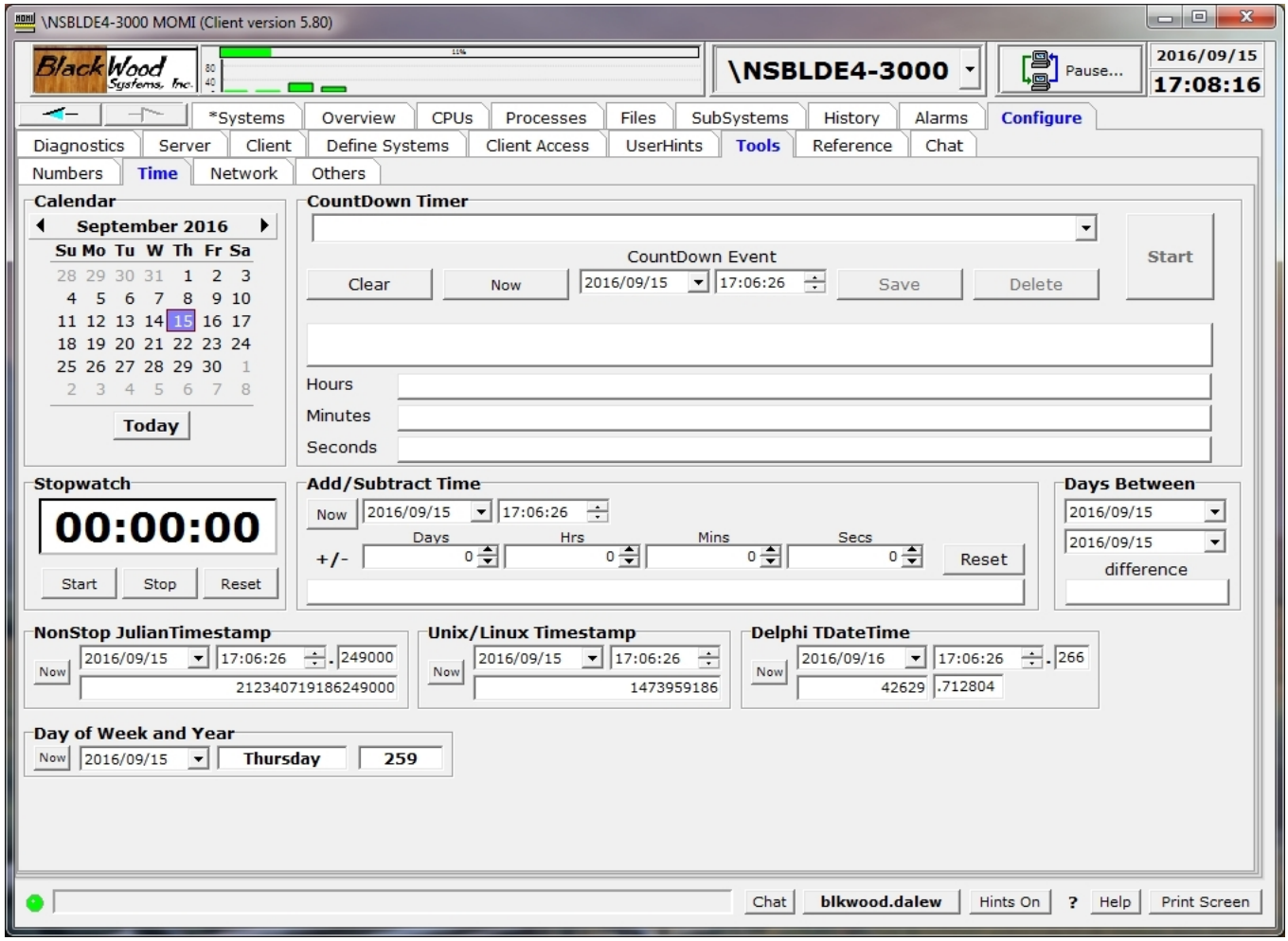
The Special functions to delete a key and DELETE-ALL-RECORDS are provided where automated maintenance activities may required removal of a few existing user hints or the complete replacement of the entire UserHints database.

The MOMI Export to File function uses a multi-line separator that appears as the | character. This is internally encoded as a <#160><|><#160> where #160 is the ASCII value of a hard-space character.

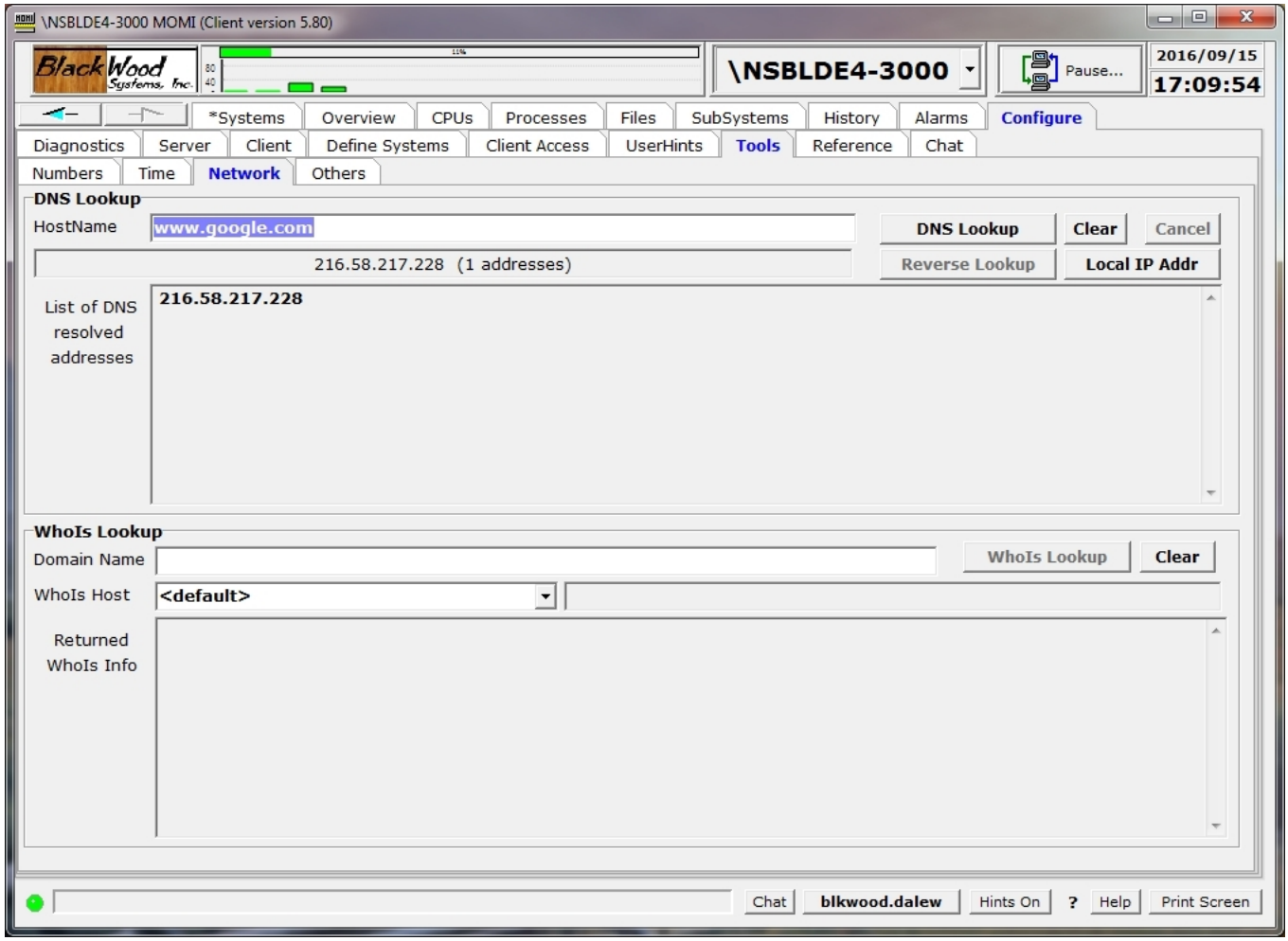
Tools



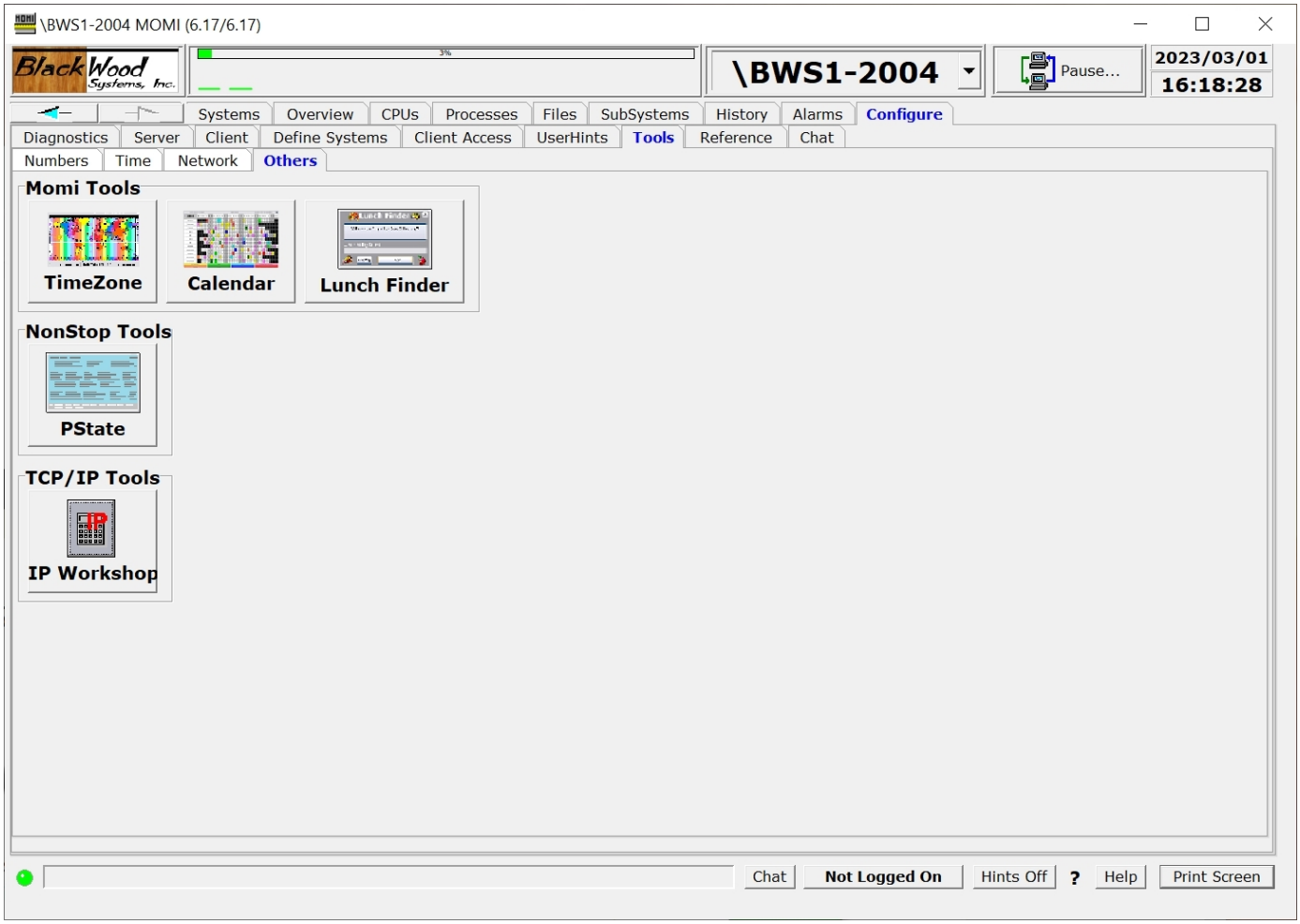
The Tools screens provides a variety of helpful utility functions such as a Calculator, Stopwatch, Count down timer, numeric base converter, DNS and WhoIs look-up.



The **CountDown Timer** drop-down box allows storing of multiple events for later reference



The **DNS Lookup** function provides DNS name to IP address(es) and an IP address to name capabilities. The **WhoIs Lookup**



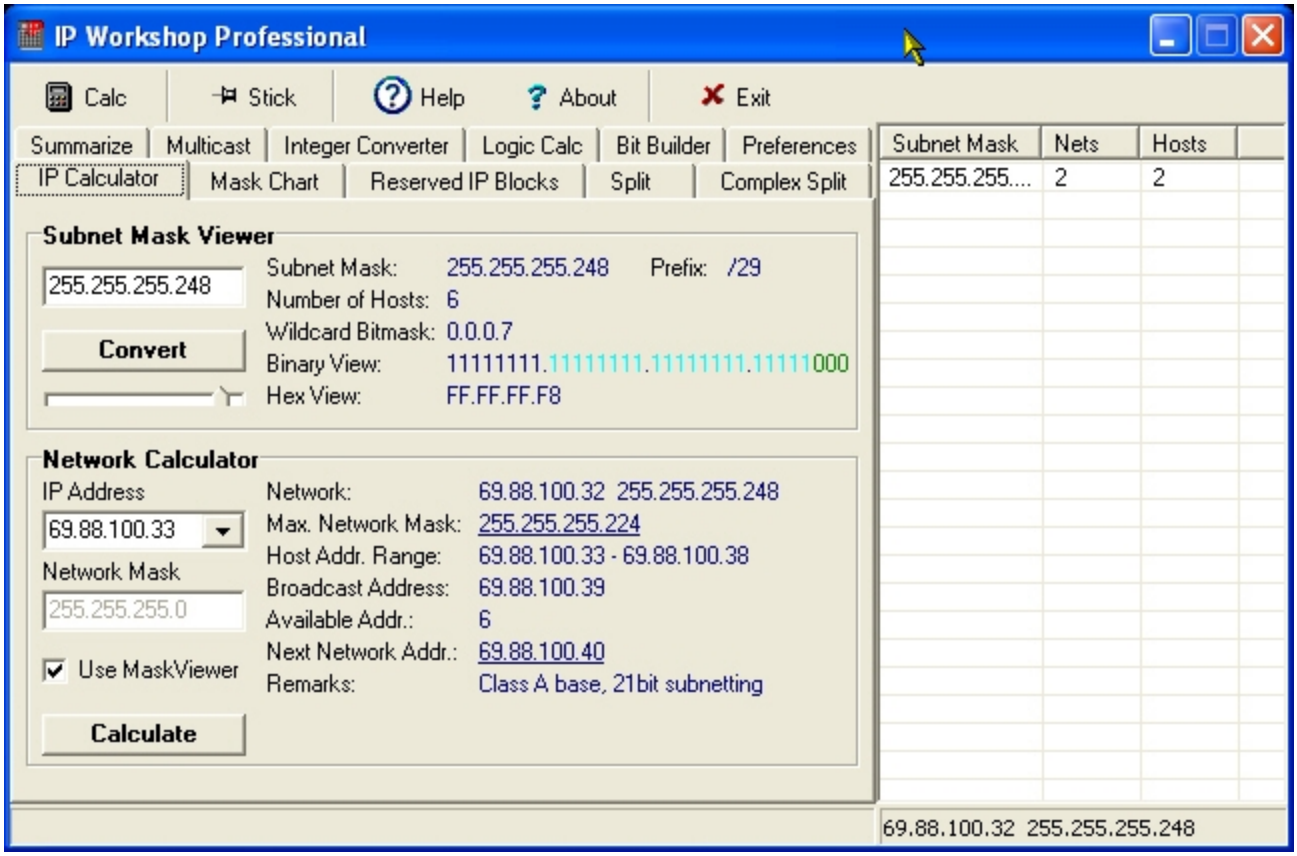
The **Others** screen provides access to various tools.

TimeZone displays a world map of timezones and displays the current location of sun at Noon.

Calendar displays a calendar of the current month.

Lunch Finder displays a help tool to determine where to goto lunch.

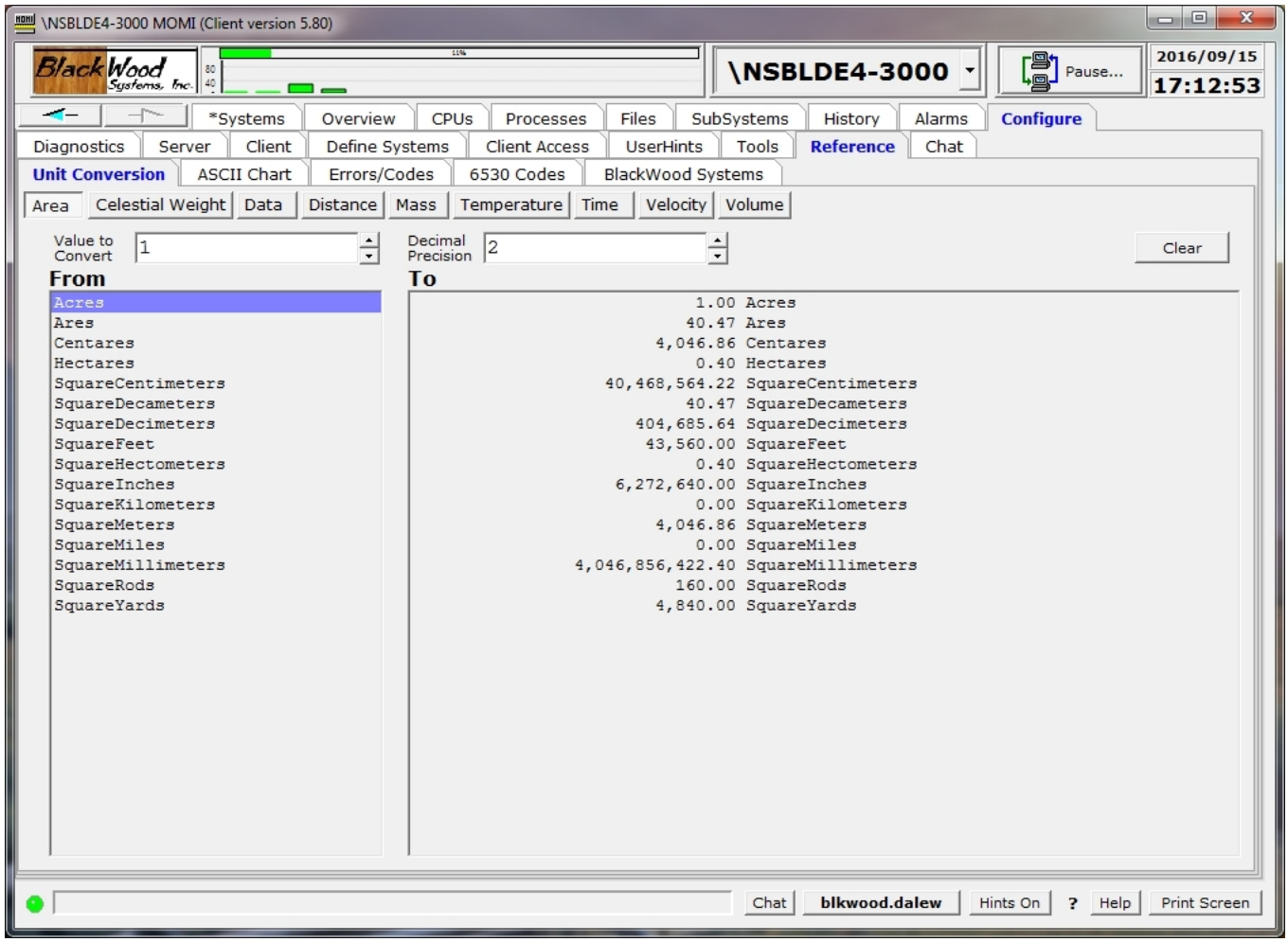
PState launches the PSTATE / PSTATE2 utility to display internal information about a process.



The **IP Workshop Professional** TCP/IP Subnet calculator is provided courtesy of Peter Kostov and Peko Software. This tool supports a wide range of IPv4 address operations such as converting the classical subnet format into a CIDR, split up a Class B network into different sized subnets and obtain a quick summery of subnets.

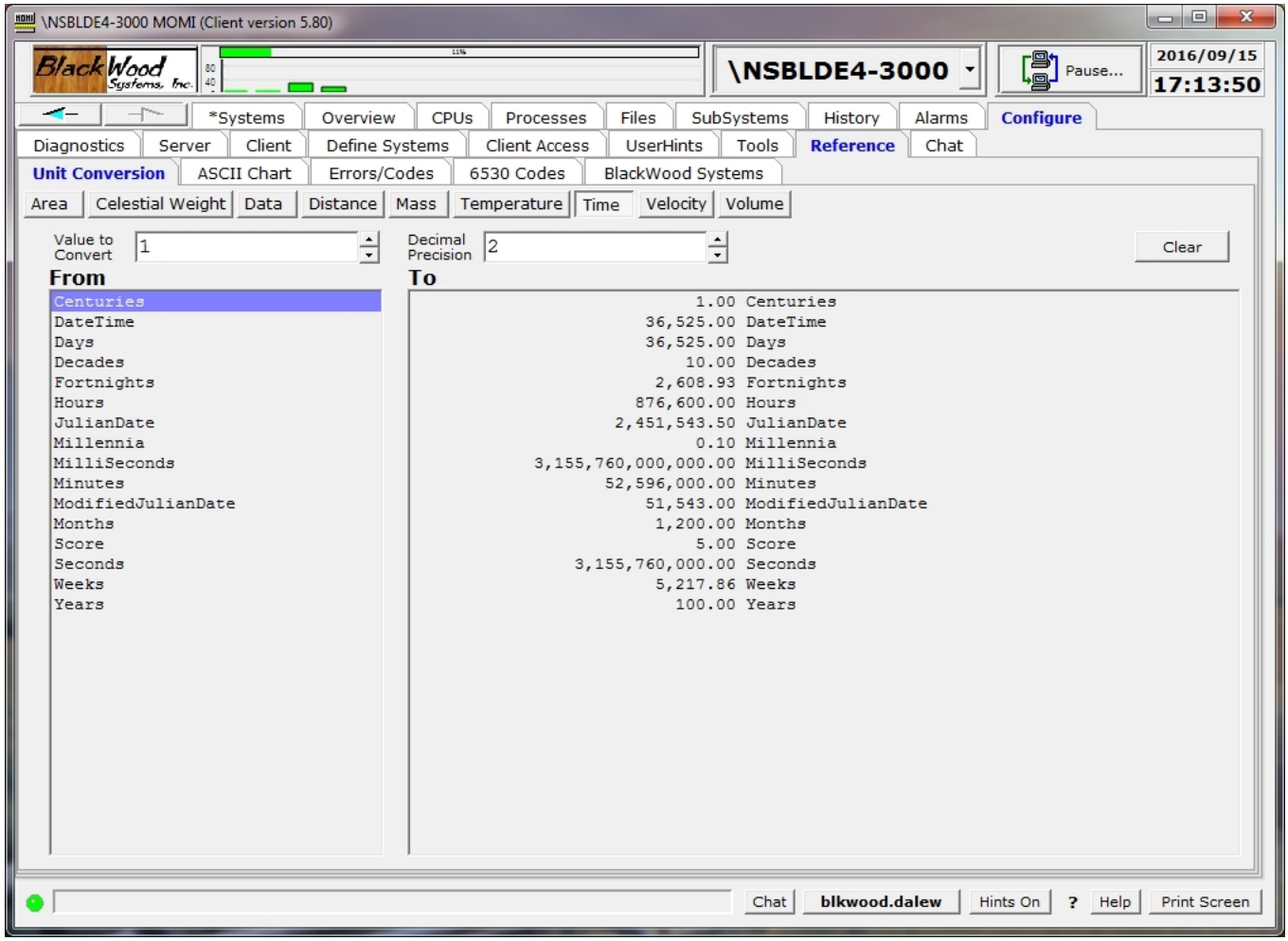
Reference

Reference

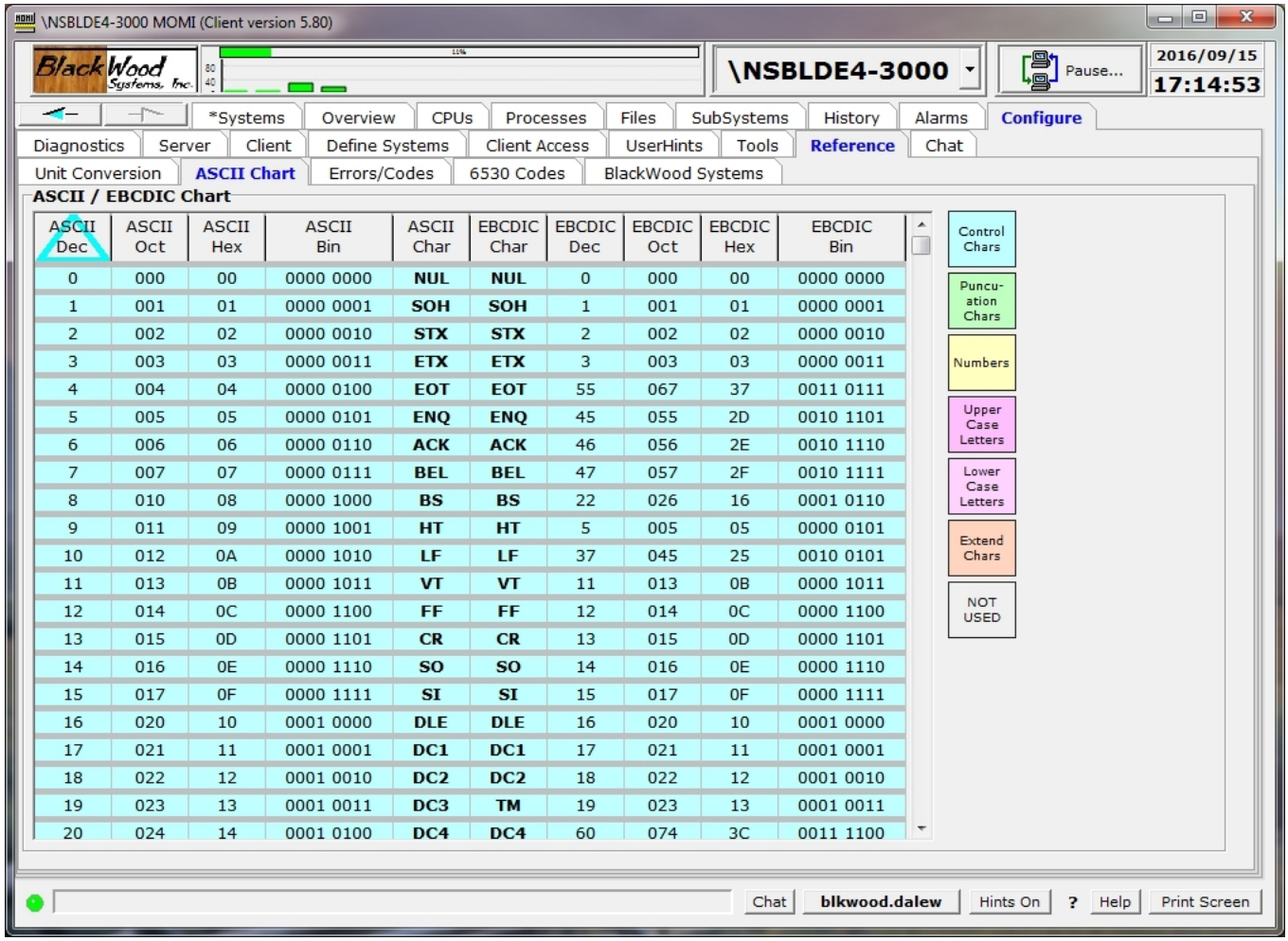


Remember the last time you needed to know how many Hectares there were in an Acre (0.40) ? The Reference section provides the answer to this question and many others. Unit Conversion, File System Error codes and the long lost 6530 Control Codes and Escape Sequence references are provided.

Unit Conversion works by selecting the types of units to be converted, entering the value and clicking on the 'From' units. In the example below under Time, 20 Fortnights is equivalent to 280 Days.

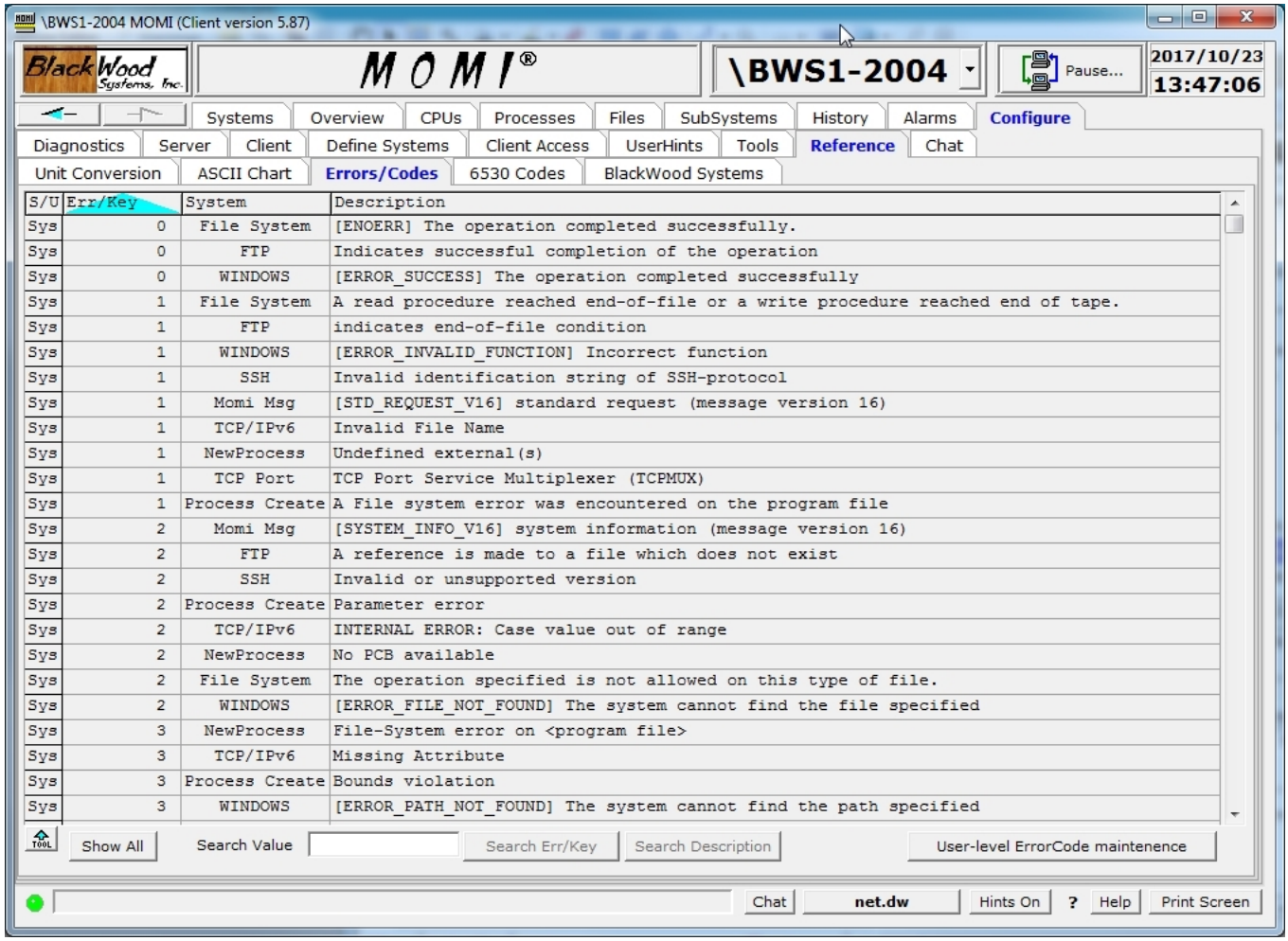


The ASCII to EBCDIC chart may be sorted in different ways by placing the mouse over the column heading and right-clicking. Place the mouse over the center two columns, ASCII Char and EBCDIC Char, for a tool-tip description.

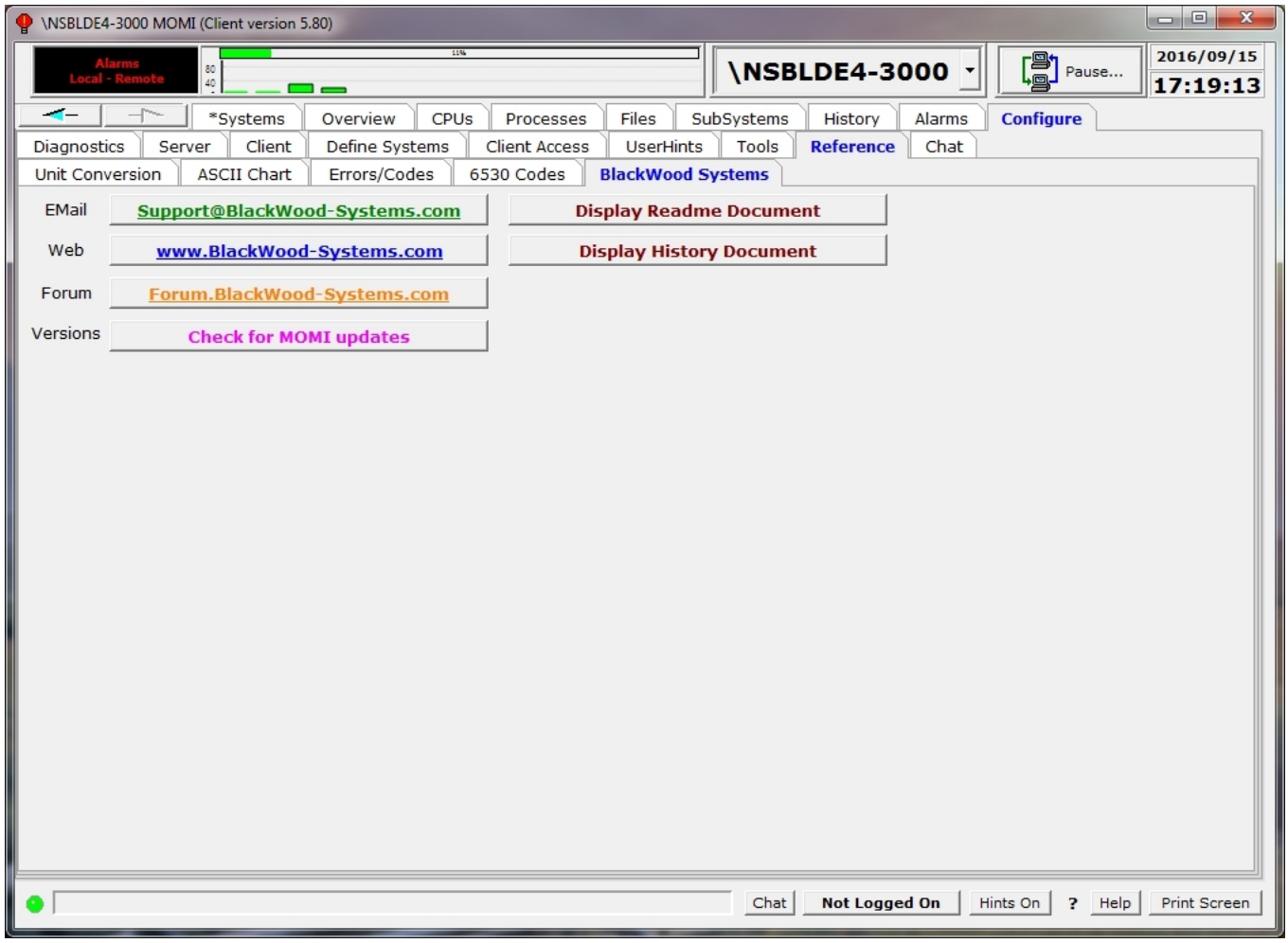


ASCII Dec	ASCII Oct	ASCII Hex	ASCII Bin	ASCII Char	EBCDIC Char	EBCDIC Dec	EBCDIC Oct	EBCDIC Hex	EBCDIC Bin
0	000	00	0000 0000	NUL	NUL	0	000	00	0000 0000
1	001	01	0000 0001	SOH	SOH	1	001	01	0000 0001
2	002	02	0000 0010	STX	STX	2	002	02	0000 0010
3	003	03	0000 0011	ETX	ETX	3	003	03	0000 0011
4	004	04	0000 0100	EOT	EOT	55	067	37	0011 0111
5	005	05	0000 0101	ENQ	ENQ	45	055	2D	0010 1101
6	006	06	0000 0110	ACK	ACK	46	056	2E	0010 1110
7	007	07	0000 0111	BEL	BEL	47	057	2F	0010 1111
8	010	08	0000 1000	BS	BS	22	026	16	0001 0110
9	011	09	0000 1001	HT	HT	5	005	05	0000 0101
10	012	0A	0000 1010	LF	LF	37	045	25	0010 0101
11	013	0B	0000 1011	VT	VT	11	013	0B	0000 1011
12	014	0C	0000 1100	FF	FF	12	014	0C	0000 1100
13	015	0D	0000 1101	CR	CR	13	015	0D	0000 1101
14	016	0E	0000 1110	SO	SO	14	016	0E	0000 1110
15	017	0F	0000 1111	SI	SI	15	017	0F	0000 1111
16	020	10	0001 0000	DLE	DLE	16	020	10	0001 0000
17	021	11	0001 0001	DC1	DC1	17	021	11	0001 0001
18	022	12	0001 0010	DC2	DC2	18	022	12	0001 0010
19	023	13	0001 0011	DC3	TM	19	023	13	0001 0011
20	024	14	0001 0100	DC4	DC4	60	074	3C	0011 1100

The Errors / Codes screen provides brief description / translation of commonly used numbers including Defines, file system error codes, MEASURE, process creation, Pathcom, Pathmon, Pathsend, SPI and TCP/IP are available. User defined error codes are added by typing in the fields Error / Keyword, System and Error Text and then pressing Update / Add.


\BWS1-2004 MOMI (Client version 5.87)
BlackWood Systems, Inc.
M O M I®
\BWS1-2004
Pause...
2017/10/23
13:47:06
Systems
Overview
CPUs
Processes
Files
SubSystems
History
Alarms
Configure
Diagnostics
Server
Client
Define Systems
Client Access
UserHints
Tools
Reference
Chat
Unit Conversion
ASCII Chart
Errors/Codes
6530 Codes
BlackWood Systems
S/U Err/Key System Description
Sys 0 File System [ENOERR] The operation completed successfully.
Sys 0 FTP Indicates successful completion of the operation
Sys 0 WINDOWS [ERROR_SUCCESS] The operation completed successfully
Sys 1 File System A read procedure reached end-of-file or a write procedure reached end of tape.
Sys 1 FTP indicates end-of-file condition
Sys 1 WINDOWS [ERROR_INVALID_FUNCTION] Incorrect function
Sys 1 SSH Invalid identification string of SSH-protocol
Sys 1 Momi Msg [STD_REQUEST_V16] standard request (message version 16)
Sys 1 TCP/IPv6 Invalid File Name
Sys 1 NewProcess Undefined external(s)
Sys 1 TCP Port TCP Port Service Multiplexer (TCPMUX)
Sys 1 Process Create A File system error was encountered on the program file
Sys 2 Momi Msg [SYSTEM_INFO_V16] system information (message version 16)
Sys 2 FTP A reference is made to a file which does not exist
Sys 2 SSH Invalid or unsupported version
Sys 2 Process Create Parameter error
Sys 2 TCP/IPv6 INTERNAL ERROR: Case value out of range
Sys 2 NewProcess No PCB available
Sys 2 File System The operation specified is not allowed on this type of file.
Sys 2 WINDOWS [ERROR_FILE_NOT_FOUND] The system cannot find the file specified
Sys 3 NewProcess File-System error on <program file>
Sys 3 TCP/IPv6 Missing Attribute
Sys 3 Process Create Bounds violation
Sys 3 WINDOWS [ERROR_PATH_NOT_FOUND] The system cannot find the path specified
Show All
Search Value
Search Err/Key
Search Description
User-level ErrorCode maintenence
Chat
net.dw
Hints On
?
Help
Print Screen

BlackWood Systems

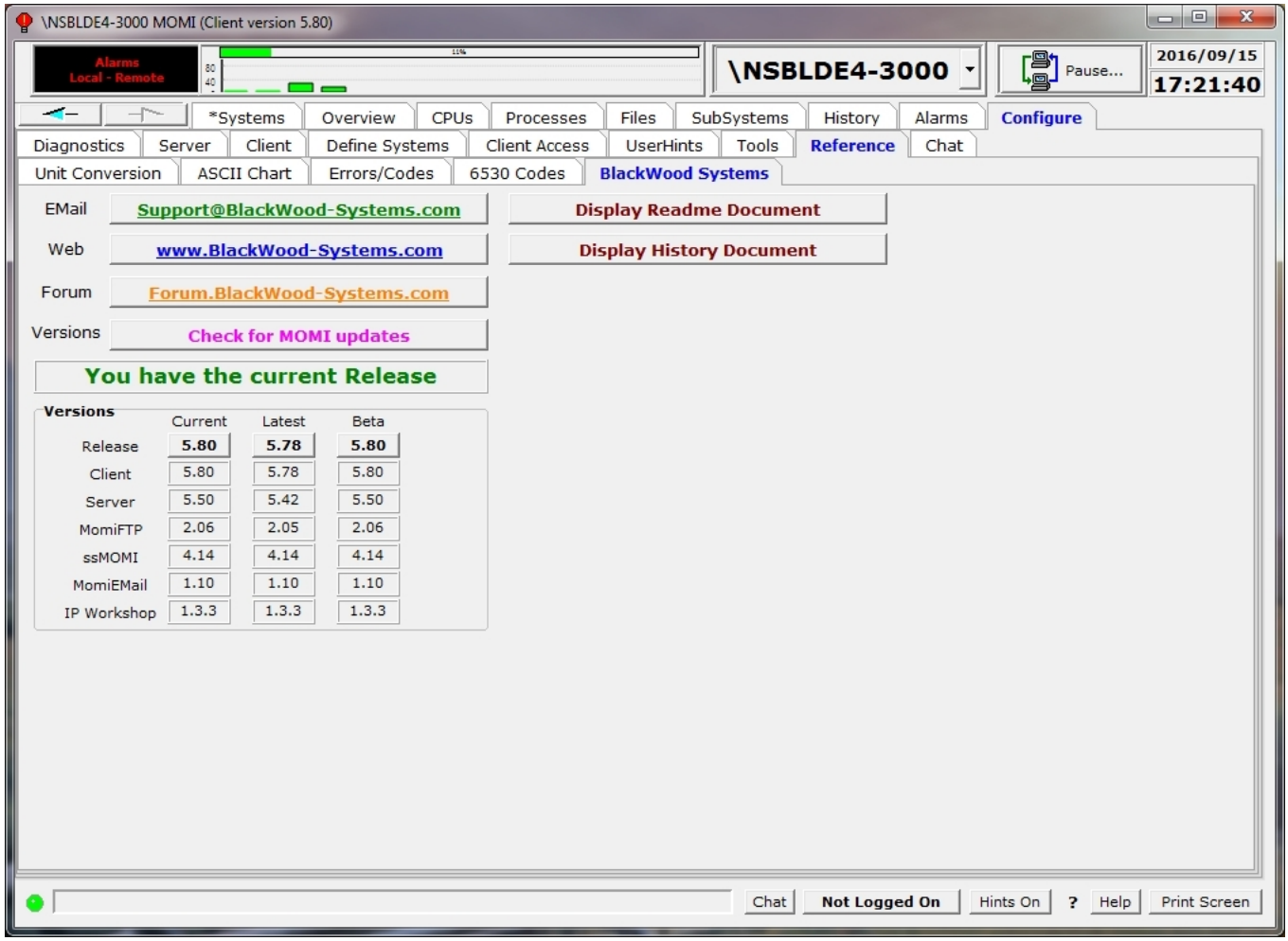


The Reference / BlackWood Systems screen provides short cuts to bring up the BlackWood System web site or Forum, generate an email to MOMI support, and check for MOMI software updates.

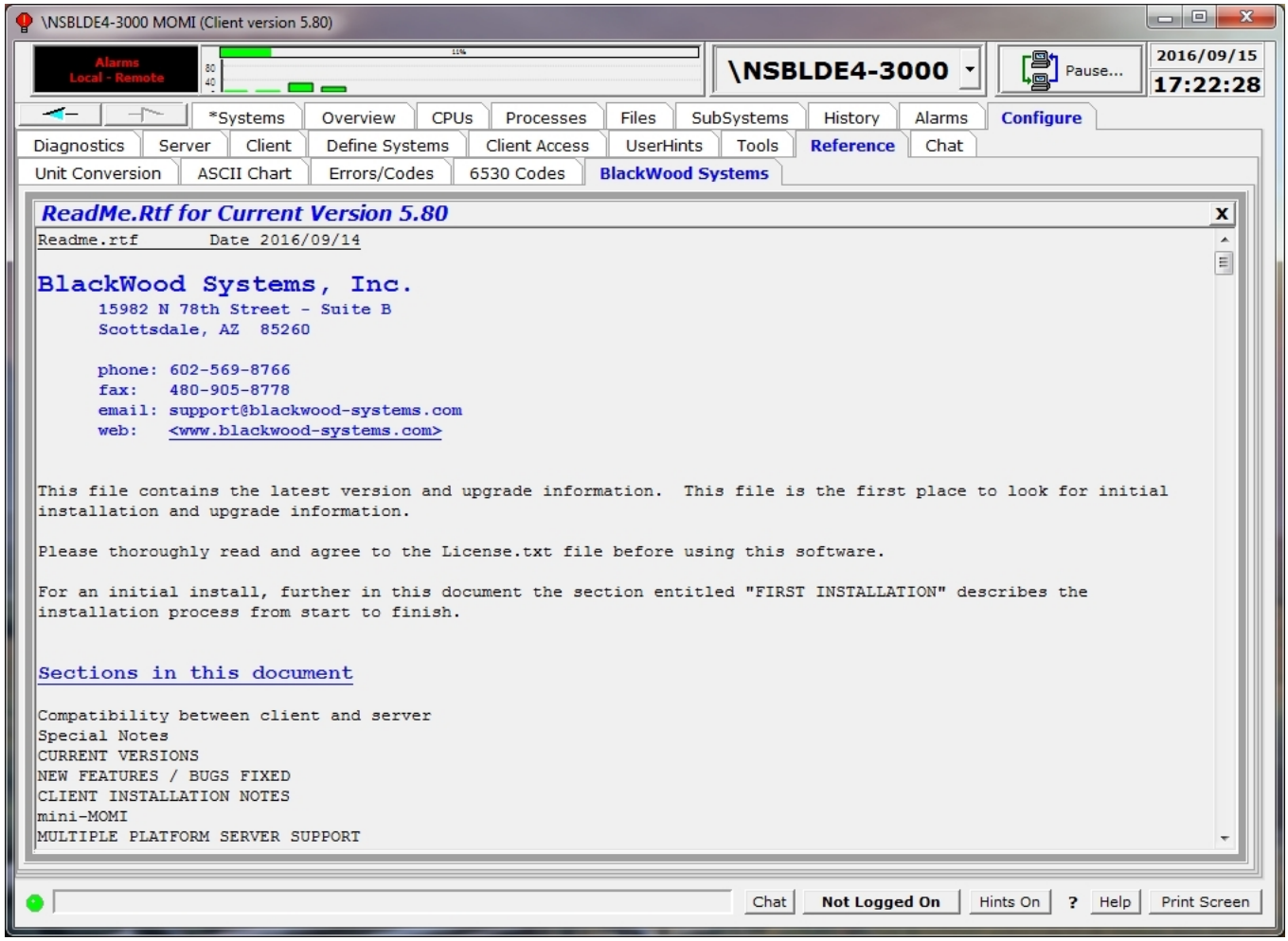
The Web and Forum functions will launch the Window PC browser to these web sites.

The Email function will cause the email client configured on the PC to launch a 'send email' window. To, Subject, System Serial Number and Client / Server version information are automatically loaded into the email.

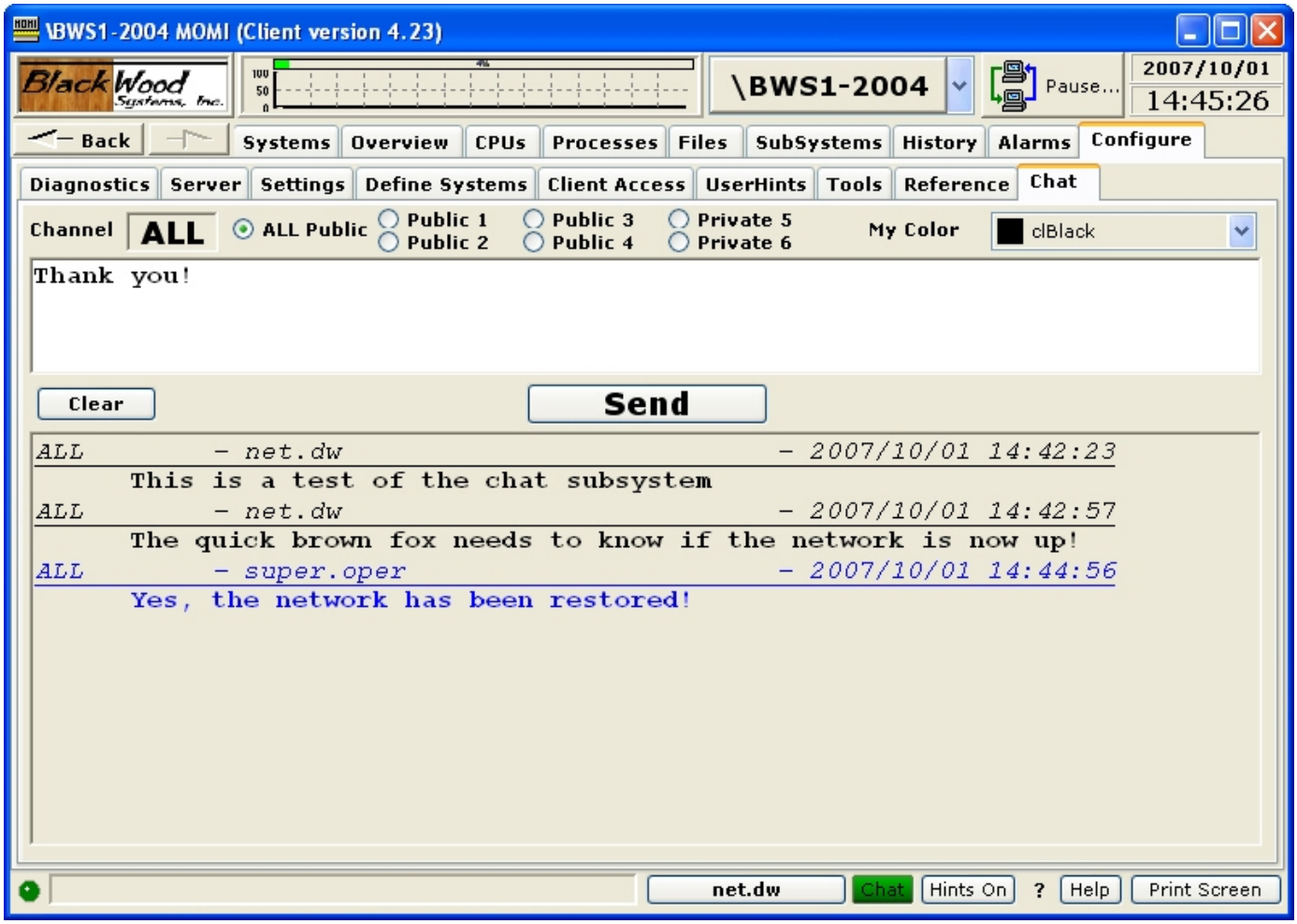
The Versions function accesses the BlackWood Systems web site to obtain version information. An area on the screen is populated with the currently installed version information, latest version available and Beta version (if available).



Pressing the buttons under **Current**, **Latest** and **Beta** accesses the BlackWood Systems web site and retrieves the corresponding ReadMe.txt file (right click to access the history file).


\NSBLDE4-3000 MOMI (Client version 5.80)
Alarms
Local - Remote
\NSBLDE4-3000
Pause...
2016/09/15
17:22:28
*Systems
Overview
CPUs
Processes
Files
SubSystems
History
Alarms
Configure
Diagnostics
Server
Client
Define Systems
Client Access
UserHints
Tools
Reference
Chat
Unit Conversion
ASCII Chart
Errors/Codes
6530 Codes
BlackWood Systems
ReadMe.Rtf for Current Version 5.80
Readme.rtf Date 2016/09/14
BlackWood Systems, Inc.
15982 N 78th Street - Suite B
Scottsdale, AZ 85260
phone: 602-569-8766
fax: 480-905-8778
email: support@blackwood-systems.com
web: <www.blackwood-systems.com>
This file contains the latest version and upgrade information. This file is the first place to look for initial installation and upgrade information.
Please thoroughly read and agree to the License.txt file before using this software.
For an initial install, further in this document the section entitled "FIRST INSTALLATION" describes the installation process from start to finish.
Sections in this document
Compatibility between client and server
Special Notes
CURRENT VERSIONS
NEW FEATURES / BUGS FIXED
CLIENT INSTALLATION NOTES
mini-MOMI
MULTIPLE PLATFORM SERVER SUPPORT
Chat
Not Logged On
Hints On
?
Help
Print Screen

Chat



The Chat screen provides a simple means to communicate with other MOMI users.

Anyone may view Chat, unless restricted by Client Access, but you must be logged on in order to participate. Chat is NOT a secured means of communication.

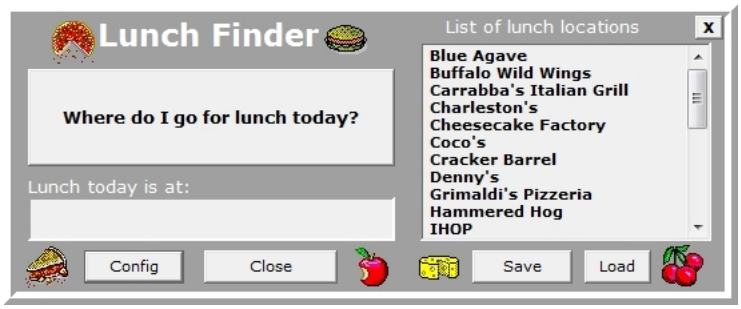
To use chat, logon, select a Channel, type in text and press **Send**.

Chat is grouped into channels. The default selection ALL, displays all chat over the public channels. The selection of any of the other channels, Public 1 - 4 or Private 5 & 6 displays only its traffic. The Chat selection at the bottom on the MOMI PC Client window displays green if any Chat has been entered in the last 5 minutes.

Additional Screens

Lunch Finder

This little gem is located by right-clicking the BlackWood Systems Emblem. It solves a major problem we have discovered for Tech Support people, "Where to have lunch today?" It's amazing how people who are able to make major decisions involving the 'care-and-feeding' of a multimillion dollar computer system are unable to decide where to eat! This was the major reason for the development of MOMI but it is hidden behind all the Tech Support stuff to make management happy.



To use the lunch finder:

1. Press the **Config** button and enter available lunch locations (press **Load** for a preconfigured list)
2. Press **Save** to store the list (step 1 & 2 are only required for initial, new or updated locations)
3. Press **Where do I go for lunch today?**
4. Consider the location displayed - if unsure of the location, repeat step 3
5. Press **Close** and secure your workstation
6. Goto and eat lunch at indicated location
7. Return back to work

MOMI's lunch finder uses the latest in Human Biometrics to determine the optimum lunch for the User for that day. Some of the many factors it takes into account are:

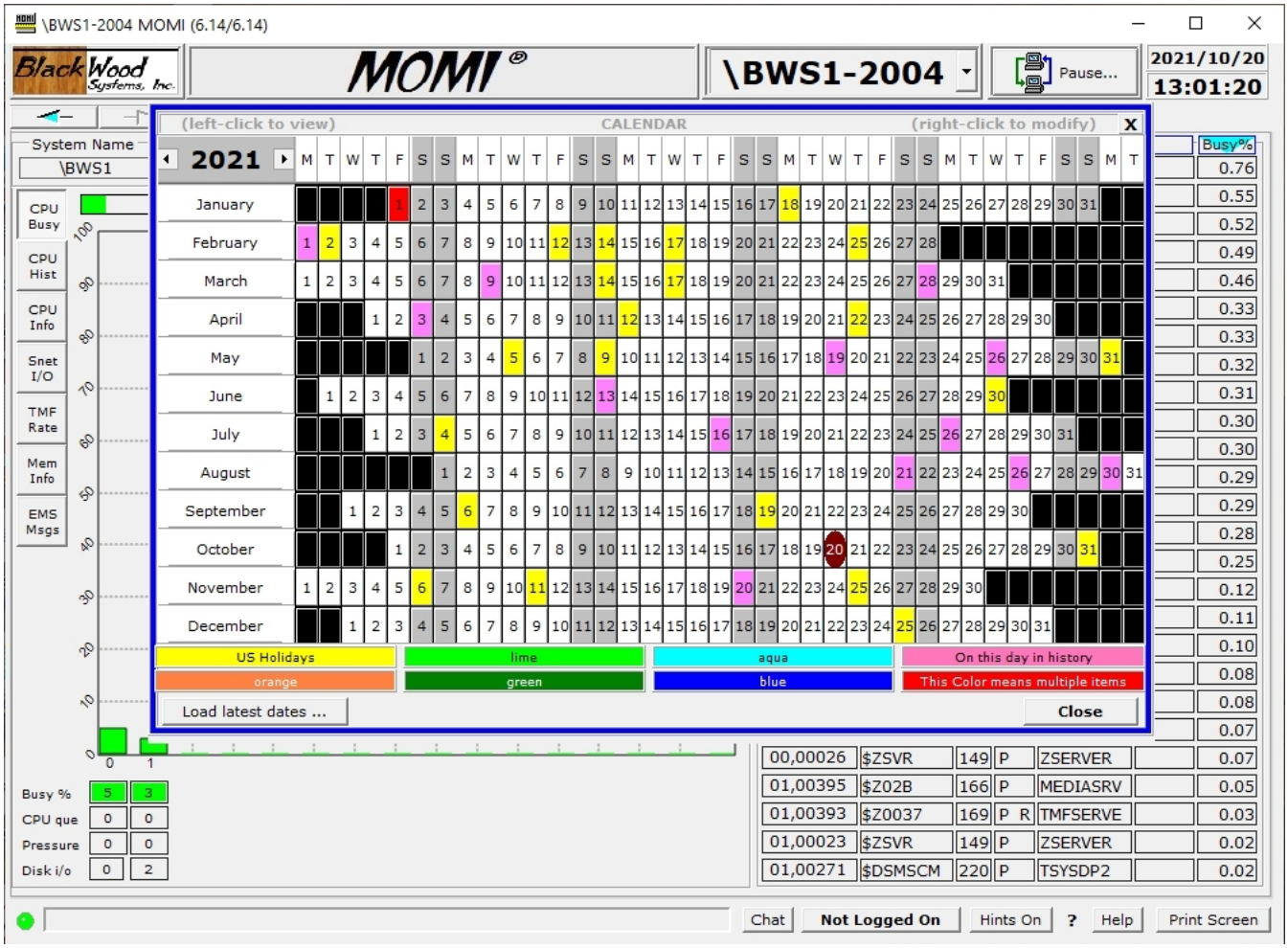
a) Key press impact strength (your need for nourishment)
b) User Biorhythm cycle (your Emotional, Intellectual, and Physical needs)

c) What lunch you've had the last week (reduce duplication and increase oral stimulation)
d) Recent computer use (if you've been making a lot of mistakes lately indicating a lack of certain minerals)

After MOMI has taken all these factors into account it uses an advanced *Needs Requirement* formula and determines the correct lunch destination by picking one from your list at random.

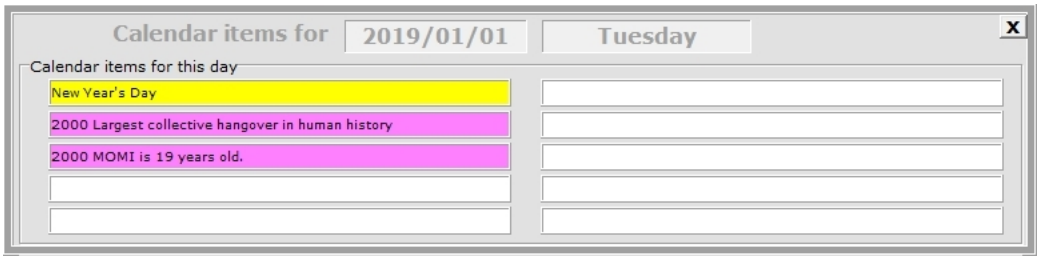
The ROI (Return on Investment) of this tool alone will pay for MOMI within a few months when you figure Systems Management people can easily spend a half hour trying to decide where to eat. The resulting increase in productivity could be phenomenal !!!

Calendar

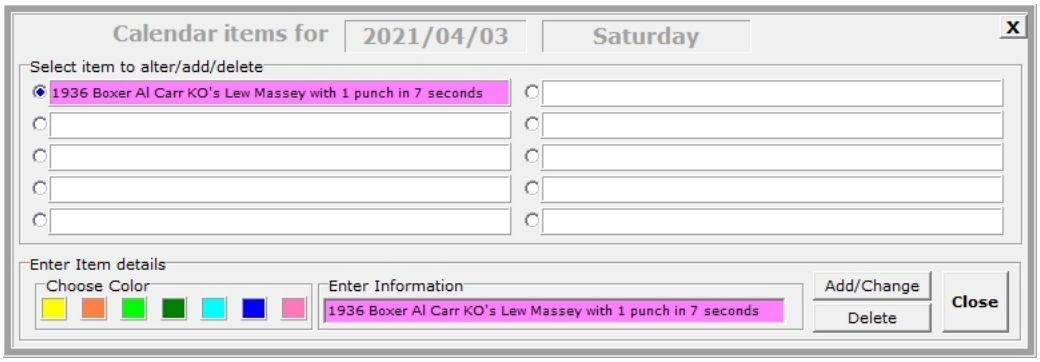


The Calendar is located by clicking the current date panel at the top of the screen and allows users to set up dates on a calendar with a color coded display to indicate some information. For example: a user might define the color ORANGE as an indicator for a scheduled Maintenance window, or BLUE and AQUA as vacation days for the Tech Support personnel. Seven colors (YELLOW, ORANGE, LIME, GREEN, AQUA, BLUE and PINK) are available for general use. The RED color is reserved for any date with more than 1 color assigned to it.

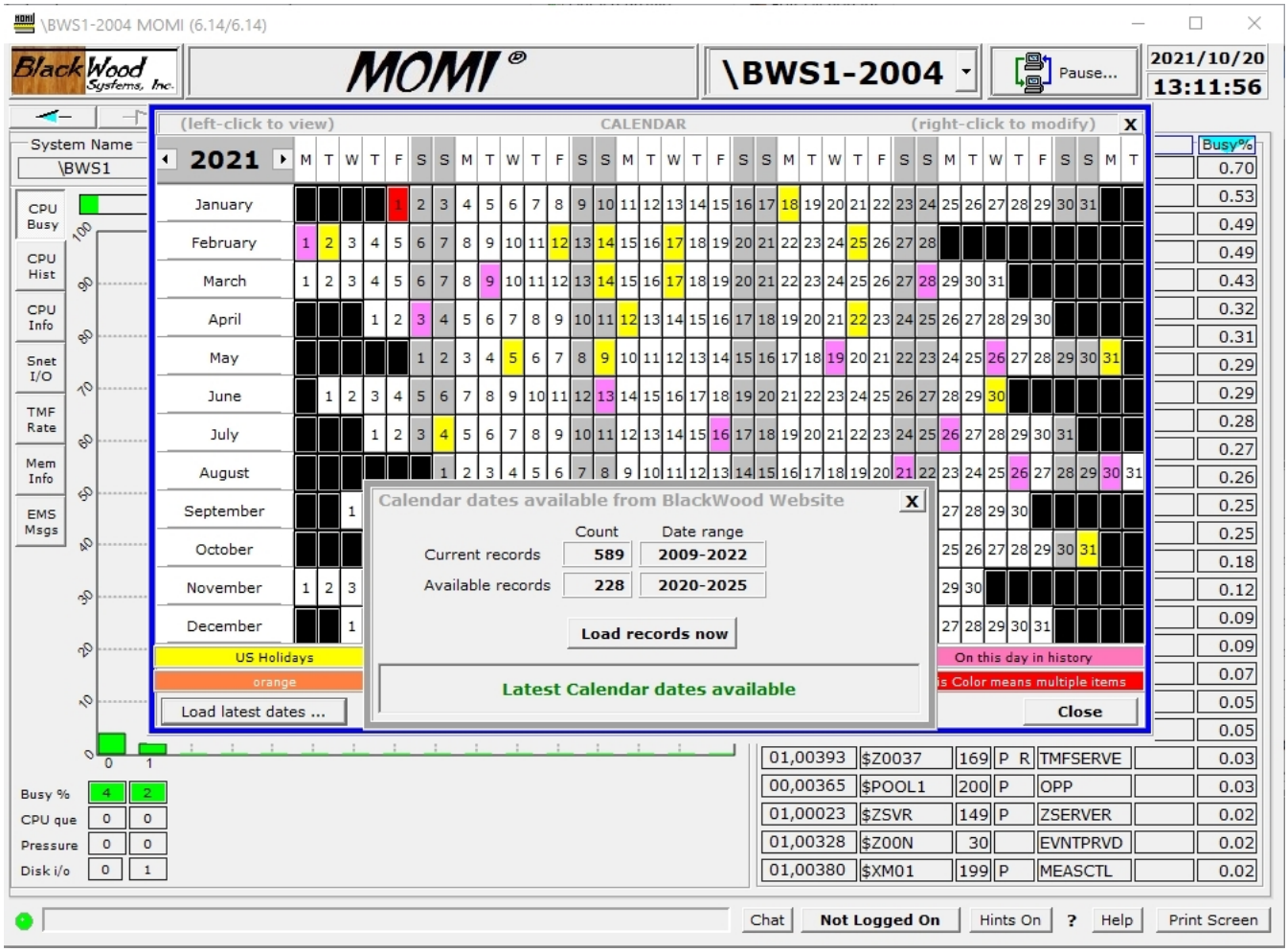
Calendar color labels are saved in the local MOMI.INI file and Calendar date items are saved in the CALENDAR.MOM database. MOMI is shipped with a basic database containing a couple years of US Holidays and some interesting *On this day in history* facts.



Left-click on a day will display a pop-up with information saved about that day in history.

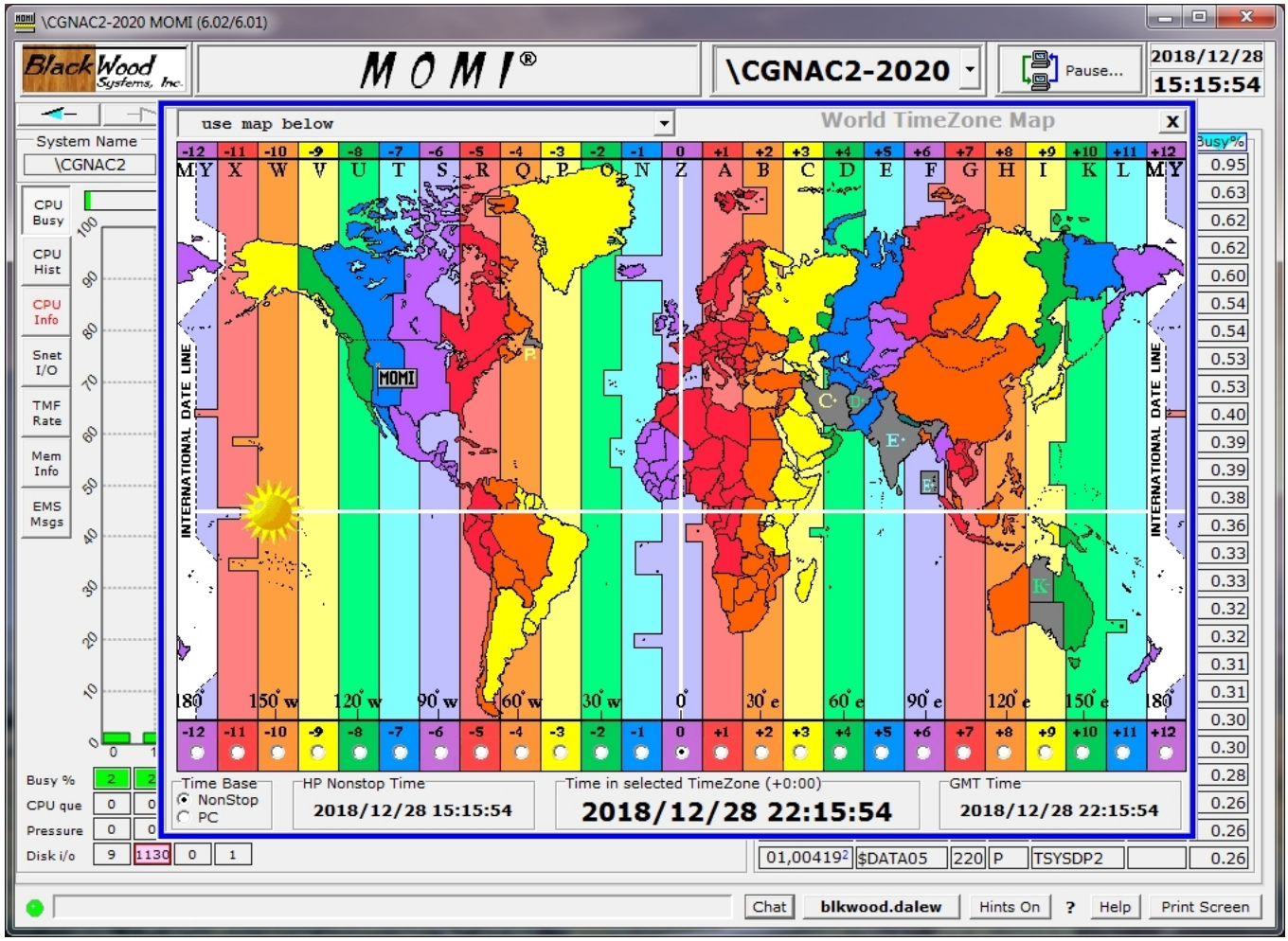


Right-click on a day to alter existing or add new items.



A new software release automatically updates Calendar information. Alternatively, pressing the **Load latest dates...** obtains the latest updates from our website. Press **Load records now** to obtain the update.

World TimeZone Map



The TimeZone Map is located by clicking the time at the top right of the screen and allows Users to get basic time zone info around the world. It does NOT account for daylight savings time and is just a guide. Please don't consider it accurate, just fun! The file that loads the supplied default TimeZone list is TIMEZONE.MOM . You may make changes and add new time zones by editing this file using Notepad. Instructions on the format are found within in the file.

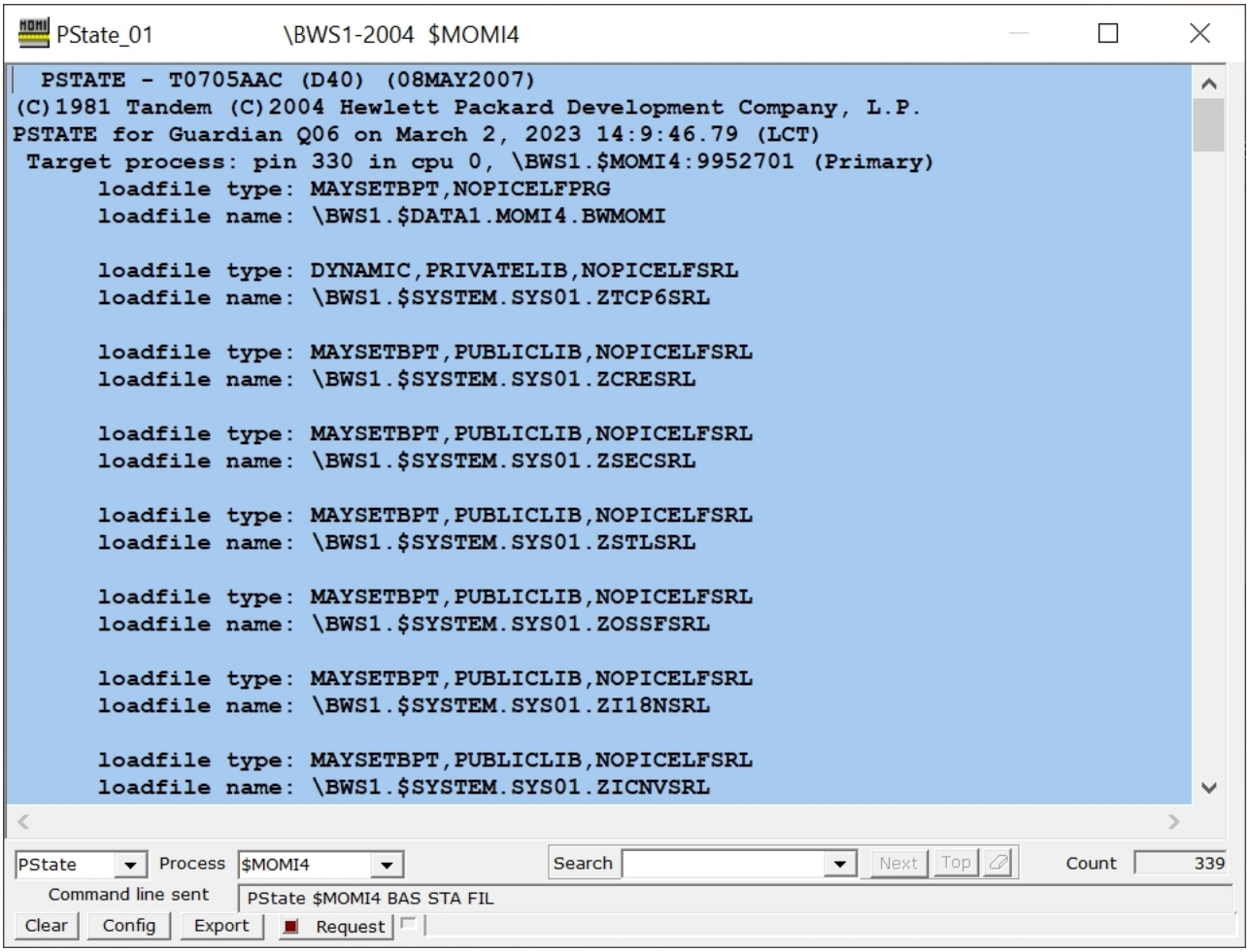
MOMI is shipped with a basic TIMEZONE.MOM database containing several hundred entries.

A User may select a time zone either by selecting an entry from the drop-down list or by clicking an offset from the GMT value near the bottom of the window.

The **Time in selected TimeZone** and **GMT Time** fields originates from either the NonStop currently selected, which is the time displayed in the upper right corner of the screen, or time from the PC. This selection is made in the **Time Base** field.

To exit this screen, click on the X below the date/time.

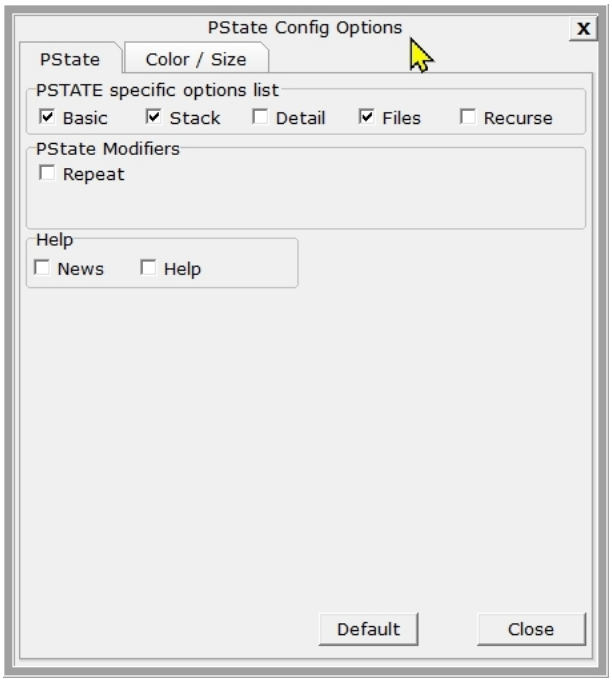
PSTATE

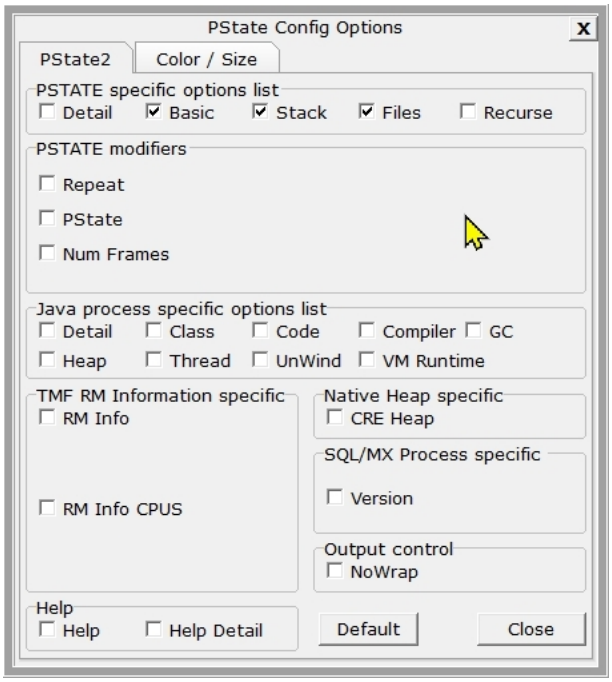


PSTATE/PSTATE2 are system utilities to display detailed internal information about a process. When invoked MOMI the options selected in Config are passed to the program and the pop-up windows displays the output. Multiple windows can be used when option to Repeat is enabled. The windows are modal or displayed outside the MOMI PC Client (any open windows close when the client stops).

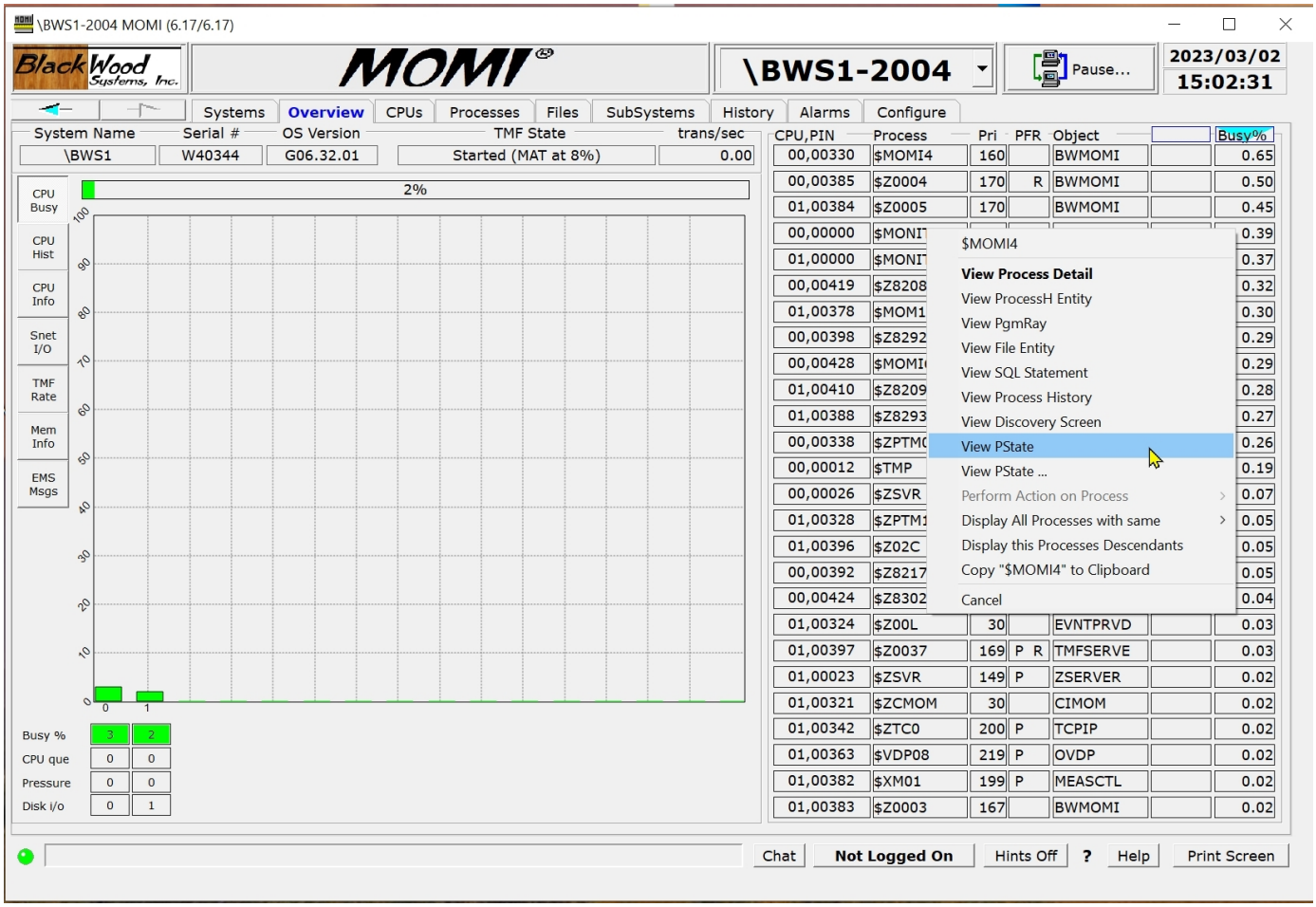
PSTATE2 is an enhanced version of PSTATE. A drop-down box determines which program is launched.

Newer versions PSTATE/PSTATE2 impose run-time security restrictions. See the CONFMOMI keyword LOGON-PSTATE for additional information.





Pressing the **Config** button displays options available when **Request** is pushed.



MOMI provides two means to launch the utility. Right-clicks are available on many screens, such Overview,when the mouse is over a process name. The program can be directory launched on the Tools screen.

PC Quirks

This section describes various situations or "quirks" present on the PC:

Client window re-size

The MOMI PC Client adjusts the size of most screen elements when its window size is changed to allow most aspects to remain in proportion.

This entire adjustment process may take, depending on how many screens are active and the computer's performance, 15-30 seconds.

System resources - "Out of Memory"

If a PC encounters an error indicating, to the effect, that insufficient memory is available, there are three general possibilities:

1. The Windows paging file, also referred to as virtual memory, is nearing capacity.
2. Physical memory is at capacity.
3. Windows internal control blocks are at capacity.

The Windows paging file is physical disk space used for swapping to and from main memory or the computers RAM. This disk space provides an extension of the physical memory, unfortunately at a much slower rate of access. Certain operations within MOMI, such as viewing a Spooler job or a disk file may require a great deal of virtual memory.

To view the paging file usage:

- Start MOMI and any other applications normally used.
- Under the Windows start menu, click on Windows Administrative Tools / Performance Monitor.
- Right-click on the graph and select Add Counters... from the context menu.
- Under Available counters, select Paging File.
- Click on the down-arrow icon to the right of Paging File.
- Select % Usage and then click the Add >> button.
- Click OK to close the Add Counters dialog.

By default, Windows automatically manages the Paging file. However, if the usage is nearing 100% consider increasing the size.

Physical memory usage can be examined by the Windows Task Manager. Under the Performance tab click on Memory. If usage is above 80% consider adding memory.

If the first two items are not the issue, another possibility (item 3) is that Windows has exhausted internal control blocks. See Configuration recommendations for additional information.

Print dialog box

The printer dialog box, that displays when **Print Screen** is pressed, may not appear, but is really located behind the MOMI window.

Pressing ALT + TAB and select the dialog box will cause it to return to the foreground.

Information

Acknowledgments

CAIL emulator

The CAIL Emulator was provided courtesy of CAIL (www.cail.com) and is not a component of any purchased MOMI product. The Emulator is no longer provided and was removed from the MOMI installation package.

IP Workshop Professional

IP Workshop is provided courtesy of Peter Kostov (www.pkostov.com).

Export Information

MOMI is assigned the following classification for export purposes:

ECCN: 5D992

MOMI does not contain an open encryption capability. The embedded encryption capability is for the protection of passwords and internal data flow and considered Ancillary Cryptography.

General Limitations

Non-privileged operation

MOMI operates in a non-privileged mode and generally relies on information available via documented interfaces. This restriction means that certain information is not available. For example, viewing the screen Processes / Process Detail cannot show the last error that occurred on a file. The last error is not available via either a documented Guardian procedure call nor via MEASURE.

Processes in the future

In its default mode of operation, MOMI does not generally display processes that are created after the current system time. This unusual situation could arise if either 1) the system is being used to experiment with changes in the system time, or 2) by accident the system time was set ahead, some processes were started, and the system time was returned to normal.

A normal change in the system time, such as where the system time is gradually adjusting, is not what this issue is describing. If the system operator was to force the system time backward say a couple of hours, MOMI would not generally pick up processes created within that couple of hour period of time. Once the couple of hours has passed, then MOMI would begin reporting on the missing processes.

The CONFMOMI keyword COLLECT-ALL-PROCESSES is provided to change the default mode of operation.

Snapshot operation

CPU and Process overview is generally displayed as the difference between two points in time. A way to think about this is that MOMI is constantly taking a limited picture of the system and compares the difference between the previous picture and the current picture.

For example, viewing the screen Main Overview the top processes on the system are the result of taking a snapshot of the system and comparing the differences from another snapshot 10 seconds later (by default). If a process was running during the entire window of time, it will be picked up by the snapshot. If the process was not running the entire window of time, the process itself should not be captured but the CPU busy (or some other field of the CPU such as Overhead) would be captured. This does not mean it will be displayed, unless it was busy enough to make the Top Processes on the system, but the snapshot captures the process.

Some screens, such as the screen Processes / Process Entity, that directly invoke MEASURE and read its result do capture all process activity.

The intention of these snapshots is not to try to capture every single process activity on the system, but capture process activity that occurs long enough that its activity could be noticed or acted on.

MEASURE can capture every process start up and shutdown. MOMI snapshots cannot. MOMI can display short duration processes on the screen Processes / Process Entity by starting this screen with a process name of $* and by sorting ascending on the Measure Window Size (last column scrolling the screen all the way to the right).

SubSystems / TCP/IP

MOMI is unable to retrieve detailed information on the screens in the **SubSystems / TCP/IP** when an IP CLIM is used. A documented interface is not available.

Optional Features

Certain functions within MOMI are optional and require authorization in the MOMI password in order to function. This document does not generally differentiate optional features.

If you have any questions about your level of optional features, please contact us.

Where MOMI stores data

Generally speaking, MOMI data concerning the overall subsystem is stored on the NonStop System and user specific settings on the local PC.

MOMI Server

The MOMI server stores configuration data in the CNF01DB file and contains the following:

Alarm definitions (including actions, times, email and email group)
Client Access (all settings and users)
EventBX
User Hints

The file can be moved from one NonStop System to another. Stop $MOMI prior to making a copy or moving in a new file.

The MOMI server stores history data in the HSTnnDB file(s) and generally contain the following entities:

CPU
DISC
MSYSTEM
OSS CPU
OSS NS
Process
System

These files can be moved from one NonStop System to another. Stop $MOMI prior to making a copy or moving in a new file.

Temporary MEASURE data files are located by default in the same subvolume as the BWMOMI executable. They may be directed to another location with the CONFMOMI keyword DEFAULT-WORK-LOCATION.

MOMI PC Client

When a Client is started for the first time, it creates the MOMI.INI file with a default location:

Windows 10 and later -

C:\Users\<User ID>\AppData\Roaming\BlackWood Systems\MOMI\

Where <User ID> is the name used at logon to Windows. You must enable *Show hidden files, folders and drives* under Windows Folder Options to enable their display.

The MOMI.INI file contains all client specific settings for the Windows logged user such as: last screen size and position, color trigger levels, timers, fonts and other screen defaults. Generally, if the setting is located under the screens **Configure / Client** & **Define Systems** it is stored within the MOMI.INI file.

The file can be moved to another PC to transfer settings. Export and Import abilities are found on the screen Configure / Client / Actions .

Systems known to the MOMI PC Client on the screen Configure / Define Systems can be preloaded for new MOMI Client users via this procedure.

SNMP Trap Recommendation

MOMI does not have the ability to directly generate SNMP traps.

However, the NonStop System has a 'built-in' capability to convert EMS messages into SNMP traps. This is described in NonStop System Manual ***SNMP Configuration and Management Manual***. In particular, you might want to start with section 9. The manual is a little old, but all S-Series and later NonStop Systems automatically have this software installed.

The concept is a MOMI alarm generates an EMS message, the EMS message is read by an EMS Distributor, the EMS Distributor sends to an SNMP Agent, and the SNMP Agent generates the SNMP trap.

The manual for the NonStop SNMP is somewhat confusing, but here is some 'quick start' information to get you in the right direction (this is just a sample) -

a) On the NonStop System start an SNMP agent to send traps to a destination (an IP address). The TACL obey file below starts an agent which is then configured to send traps to a PC with an IP 192.168.20.21:

```
                                                    == start snmp agent
          SNMPAGT / NAME $trap ,CPU 1,PRI 170, TERM $ZHOME,OUT
$ZHOME,NOWAIT / &
                    0, TCPIP^PROCESS^NAME $ztc0

                                                    == clear existing
destinations
                                                    == and add new (the
IP address
                                                    == is where traps are
sent)
          scf delete trapdest $trap.#*

          scf add trapdest $trap.#mypc, hostaddr "192.168.20.21"

          scf start trapdest $trap.#*
```

b) Create an EMS filter (copy text below to an edit file named FILTER) to control what events are passed. In this example, all events pass:

```
              filter MYFILTER;

                  begin
```

```
destination
rid 1,
type process,
name =_snmp_agent,
format off;

[#def ssid^structure struct

    begin

        char z^owner
    (0:7);
        int z^number;
        int z^version;

    end;]

pass;

end;
```

c) Compile the filter. This is a one-time action unless the filter is changed:

```
#PUSH DUMMY
#LOAD /KEEP 1, LOADED DUMMY/ $SYSTEM.ZSPIDEF.ZSPITACL
#POP DUMMY
EMF /IN filter/ filter
```

d) enter the define =_snmp_agent (which is referenced by the filter) and point to $trap started earlier then start the EMS distributor:

```
                    == add output destination (note
.#TRAP is required)
delete define **
add define =_snmp_agent,file $trap.#TRAP

                    == start snmp agent
emsdist / NAME $etrap ,CPU 0,PRI 170,TERM $ZHOME,OUT
$ZHOME,NOWAIT / &
          type printing, collector $0, filter filter
```

e) generate some EMS messages to $0 and view the traps at the destination specified above:

A program to display traps on your PC was found at (no affiliation with us - no guarantees - just the result of a Google search):

http://www.bttsoftware.co.uk/snmptrap.html

(software is very simple - did not require an install - download, unzip and run)

Support

Reporting a problem

To report a problem, ask a question, or provide other feed-back send an email to:

support@blackwood-systems.com

Please consider the following in your email:

- attach screen shot(s) so we can see where a problem occurred or see where a question arose.
- always capture the entire MOMI PC Client window with any screen shot.
- with the client connected to the NonStop System in question, attach the ASCII text file created on the screen **Configure / Diagnostic**s by pressing the button **Save Diagnostic and Config Info to File**. this file provides us with version, configuration and environment information.
- If a MOMI process is looping or consuming an excessive amount of CPU - ABEND the process instead of issuing a STOP so that a SAVEABEND (i.e. ZZSAnnnn) file is created. A SAVEABEND file may be forced by issuing from a TACL prompt the command **#ABEND $name** where $name is the process consuming the excessive amount of CPU. After issuing the #ABEND command located the ZZSAnnnn file(s) created by looking in the same subvolume as the $MOMI object file. PAK and send us the compressed SAVEABEND file(s).

FTP site

Our FTP or SFTP site is accessed with the following information:

```
address     : ftp.blackwood-systems.com
user name   : momisave
password    : momisave          (both are the same and lower case)
```

The logon will take you to a special area on our FTP site where this User ID has access to place (i.e. write) files. Please send us an email if you have any trouble accessing the FTP site.

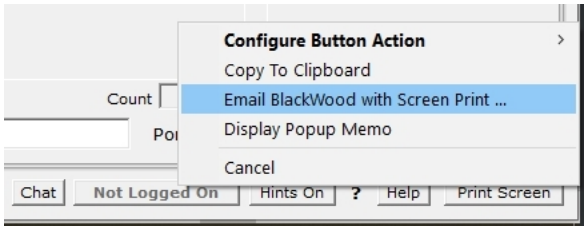
If sending a SAVEABEND via email, the file **must** be compressed (either using PAK on the Nonstop Server or ZIP on the PC) then attach the resulting file. When compressing a file, please use a file extension to denote the type of compression (i.e. .zip for ZIP and .1729 for PAK).

- **A SAVEABEND file must be PAK'ed or ZIP'ed then attached to an EMAIL. A direct attachment is not usable.**

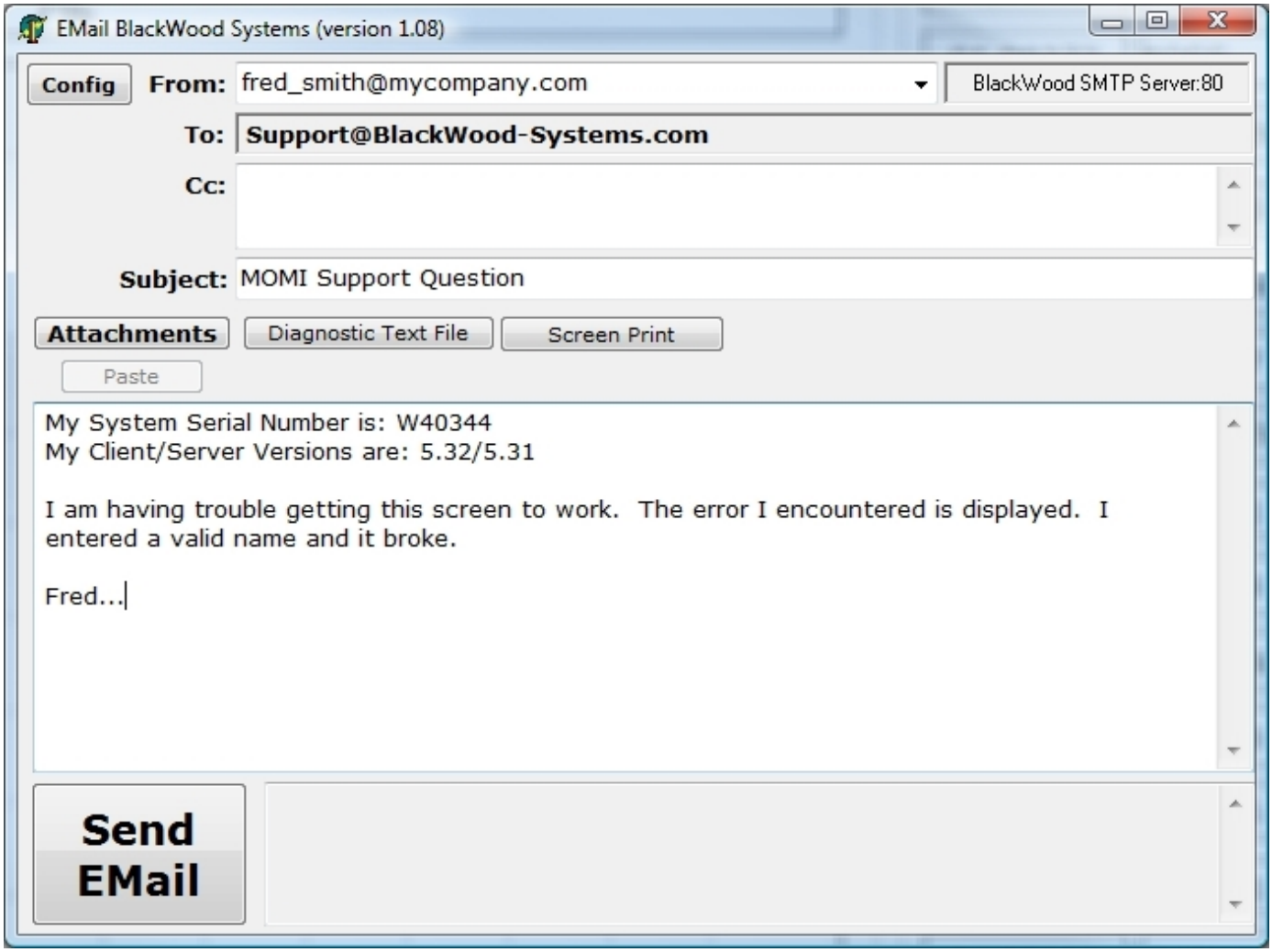
MOMI Problem Reporting Tool

MOMI has a built-in problem reporting capability. When launched, and image of the current PC MOMI Client window (not the entire screen) and the Diagnostic Text File are obtained and attached This file is in plain ASCII text (you can examine it) helps us understand your environment and resolve issues (also saves us from asking lots of basic questions such as what software releases are being used, how many processors, what type of PC O/S, etc). Screen captures are the JPG format. All of the gathered information can be reviewed prior to sending.

To use MOMI's built-in problem reporting software, with the screen in question displayed (if applicable), right-click on the **Print Screen** (or **Clipboard**) button located in the lower right-hand corner of the MOMI window and select **Email BlackWood with Screen Print...**.



This selection displays the window below and captures the current PC MOMI Client window. Additional screens may also be attached to the same email by selecting another screen within the MOMI PC Client again right-click on the **Print Screen** button and select **Email BlackWood with Screen Print...** (up to 30 screens may be attached in one message). Once all desired screens are captured, fill in the **From** with your email address and optionally add **CC** email addresses. In the body of the message type in any descriptive information concerning the problem or question (please try to give us enough detail so we may reproduce the problem or understand the question within the context of your environment). Push **Send Email** to transmit the data.



MOMI's problem reporting software uses the DNS addresses of momi-mail.blackwood-systems.com (backup is momi-mail-b.blackwood-systems.com) over port 80. If access via port 80 is blocked in your local network, pressing **Config** displays a window with the following selections:

1) **Use the BlackWood Systems Server**

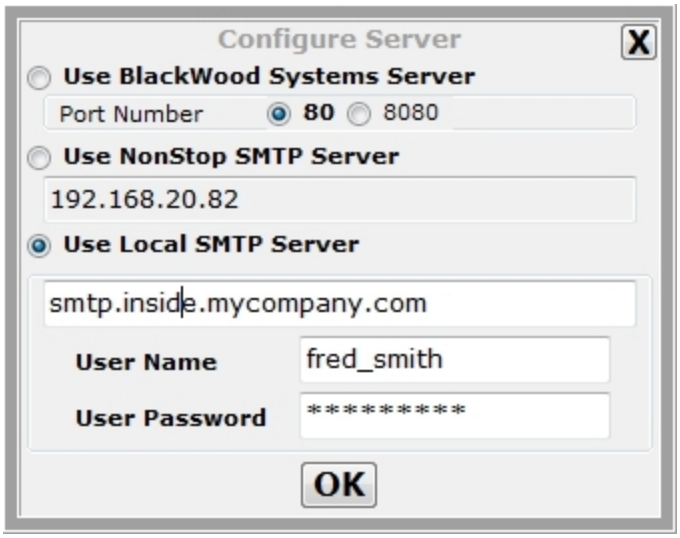
   This selection is the default and sends directly to BlackWood. Port 80 is used by default and optionally port 8080. If one does not work try the other.

2) **Use NonStop SMTP Server**

   An entry will be present is an SMTP server is define for MOMI on the NonStop (ALARM-SMTP-SERVER-ADDR).

3) **Use Local SMTP Server**

Manually enter an SMTP server DNS or IP-Address and if needed specify a User Name and Password to access to the local SMTP server.



Any selection other than **Use BlackWood Systems Server** will communicate using standard SMTP protocol over port 25.

Comments and suggestions

We welcome your comments and suggestions!

Use either MOMI's built-in problem reporting software or send an email to:

support@blackwood-systems.com

Please note that while suggestions are considered, they may or may not be included in a future release.

Network / Troubleshooting

Overview

Today's networks are increasingly complex and installing a new application may require interaction from several operational groups in order to insure success. This section describes how MOMI communicates between the PC Client and NonStop based MOMI server.

The MOMI PC Client uses native TCP/IP (i.e. no RSC or anything like that is used) and opens a TCP socket to the MOMI Server on the NonStop platform.

Common problem scenarios are described and discussed.

Basic Network Information

MOMI software communicates from the Windows based MOMI PC client to the NonStop System based MOMI server over TCP level sockets. The MOMI server listens on an assigned TCP/IP stack, IP address and port for incoming connections from the MOMI PC client. Communication over an established socket is two-way or bi-directional. The MOMI PC Client may establish one or more socket connections to MOMI server or to other MOMI servers on other NonStop Systems. MOMI servers do not establish sockets to other MOMI servers. Generally, once a socket is established it remains connected.

The MOMI PC Client directly uses Windows native TCP/IP to communicate to the NonStop System. No other comm related software is required.

If a network firewall, protecting the NonStop server, is located in-between the PC Client and NonStop server it must be configured to allow an incoming connection to the TCP/IP address and port as specified for the listening MOMI server.

The TCP/IP stack, IP address and port used by the MOMI server is defined on the NonStop System in the edit format CONFMOMI file (located by default where MOMI is installed on the NonStop System) with the keyword TCPIP-LISTEN.

Data encryption

MOMI network traffic always encrypts passwords. By default data is encrypted, except message headers. The level of data encryption may be altered with the CONFMOMI keyword ENCRYPTION-ALGORITHM.

If everything goes right

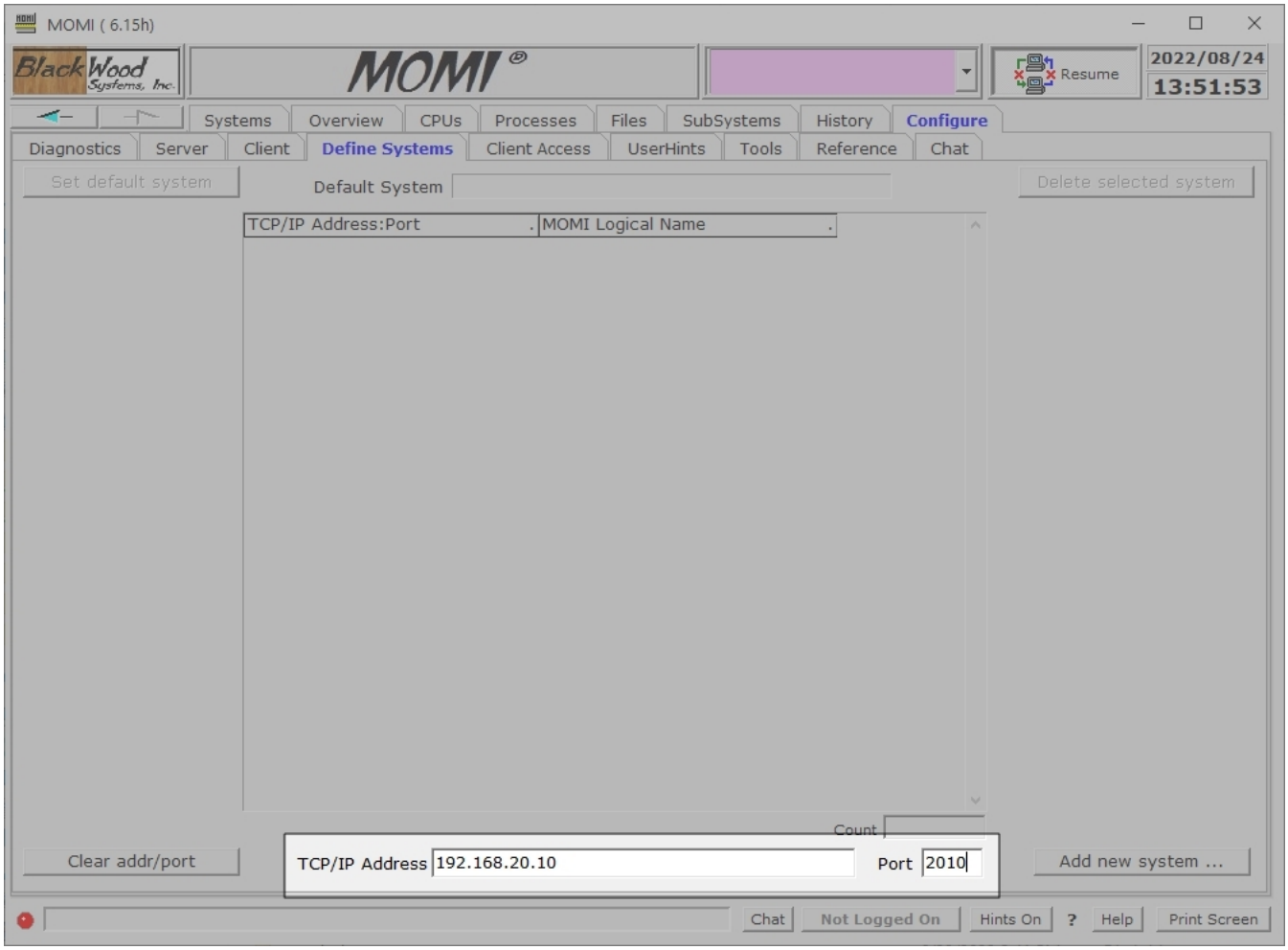
The MOMI server on the NonStop System is started first. This allows the server to initialize and prepare to receive incoming connections.

On the NonStop server, in the subvolume where MOMI is installed the edit format file CONFMOMI contains at least one of the following lines:

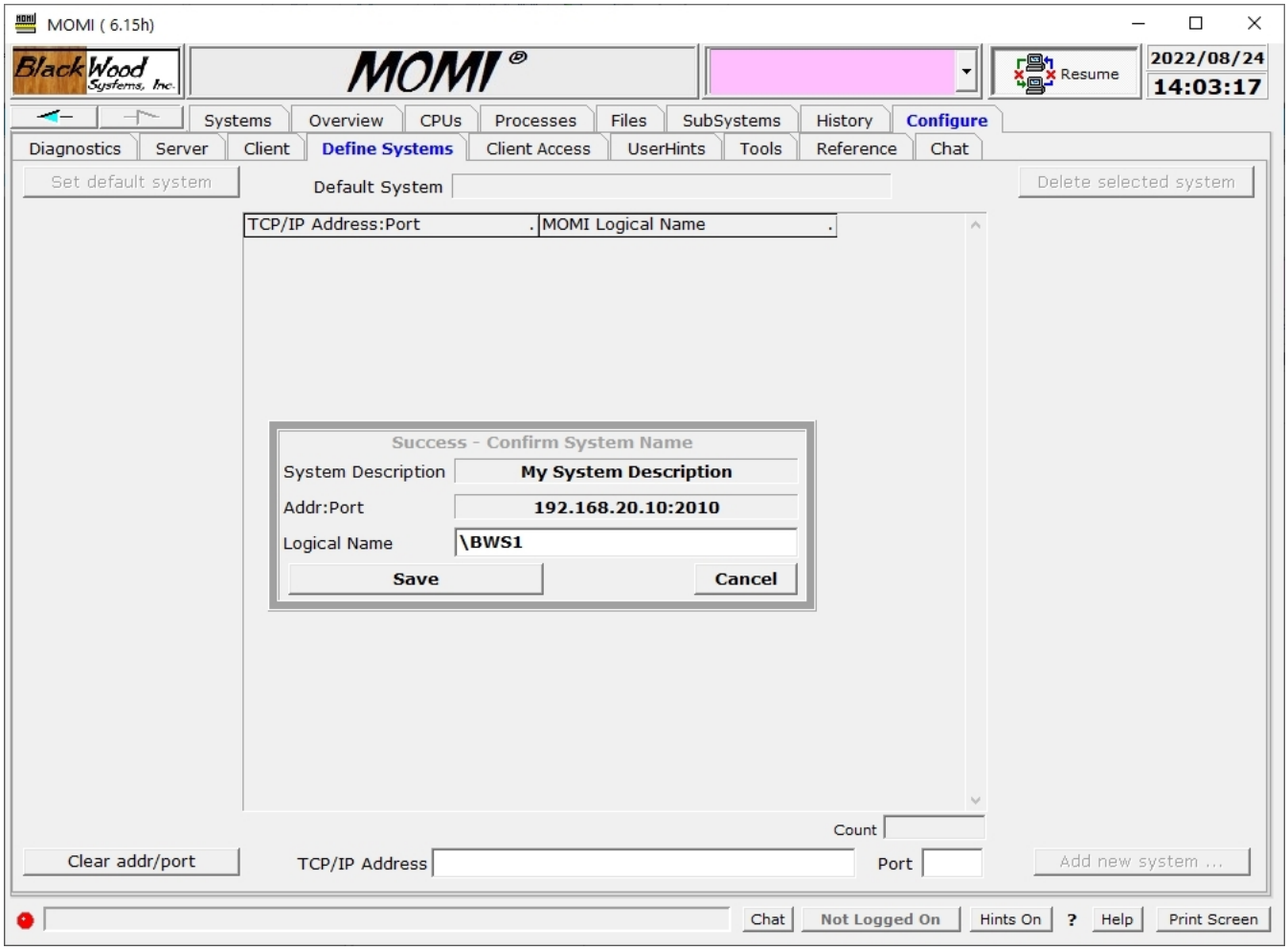
```
                          == TCP/IP process and port
                          == the MOMI server listens
                          == $stack IP-addr port
TCPIP-LISTEN $ZTC0  192.168.20.10  2010
```

In this example, the line specifies that the MOMI server is listening on the TCP/IP process $ZTC0, all IP addresses on port 2010.

Once the MOMI Server is running, give it a couple of minutes after starting, in the MOMI PC Client, assuming that $ZTC0 has a TCP/IP address of 192.168.20.10, you would enter the following information (color highlighted) on the screen Configure / Define Systems:



If communication is successfully established, the following displays:

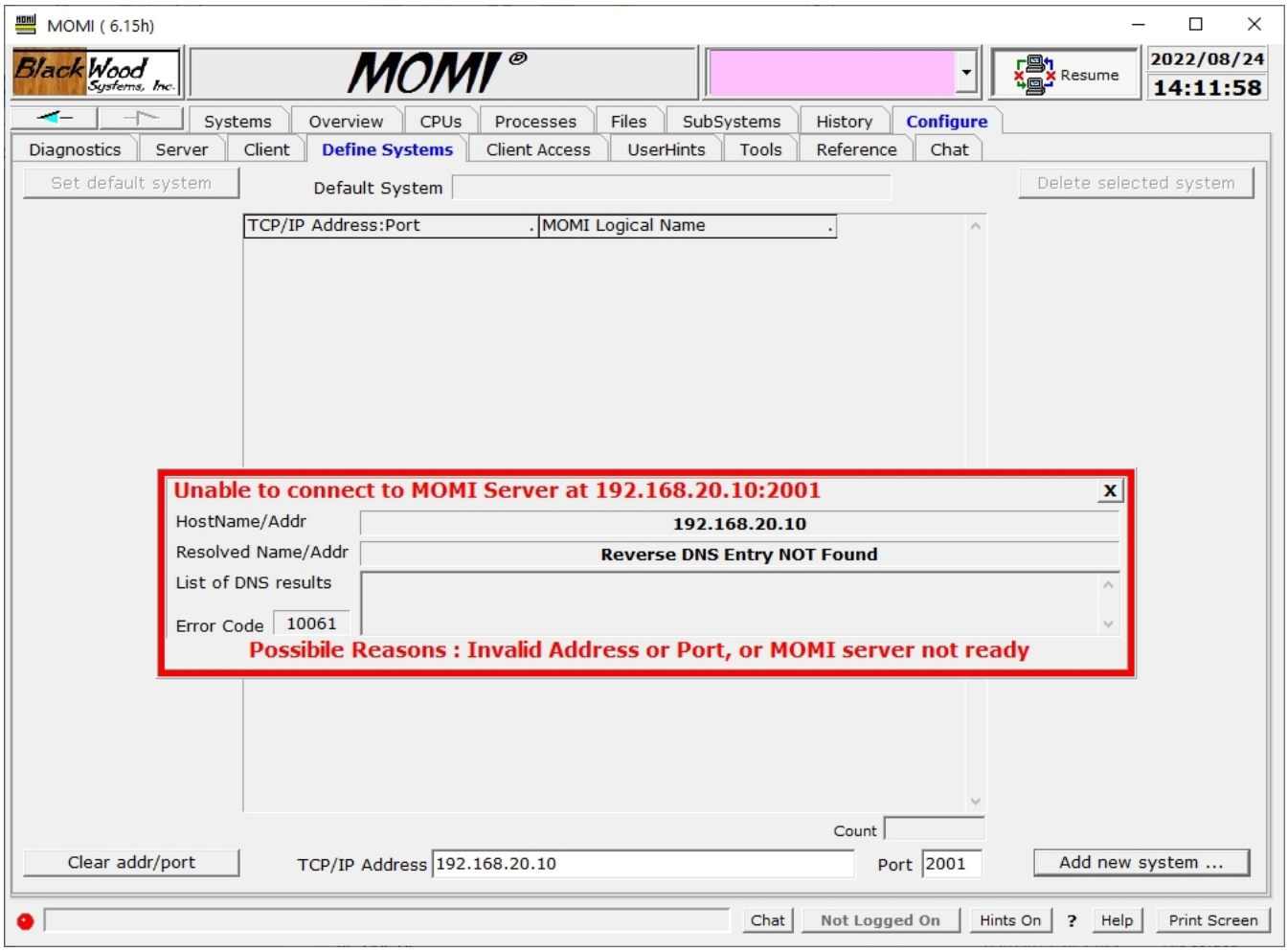


The pop-up window entitled "Confirm System Name" indicates communication was successfully established. If desired update the text field, showing the system name \BWS1 above, and then push Save to complete the operation of defining a system to MOMI.

The first system successfully defined becomes the default system displayed on subsequent startup of the MOMI PC Client. The default system displayed may be changed on this screen.

Now, what can go wrong (i.e. troubleshooting)

If communication was not successful, the following displays:



*I pressed **Add new system...** and nothing happens*

> It can take up to 30 seconds for a response if a timeout or retries occur. Normally, after pressing **Add new system...**, the response from the MOMI server is immediate.

*I pressed **Add new System...** and the MOMI PC Client locks up or terminates*

> This indicates that AntiVirus and/or firewall software is blocking the TCP socket connect from the client to the server.

Check with your PC and/or network support groups and see what authorization is required to enable a new PC application. MOMI uses a TCP level socket over the port number you specify. You may need 1) the MOMI PC Client itself authorized to operate, and 2) the TCP level socket and port authorized.

The TCP/IP process name is not valid

Shortly after starting the MOMI server, check the EMS log on the NonStop System. If MOMI encounters any error accessing the TCP/IP process, messages are logged to EMS.

To determine if the TCP/IP process named entered in the CONFMOMI file ($ZTC0 used in our example) is correct, from a TACL prompt enter the following command:

```
STATUS $ZTC0
```

If the process is found, several lines of data appear listing either TCPIP or TCP6SAM as the object file name (usually located in $SYSTEM.SYSnn). If the process name is not found, TACL returns "Process does not exist". Check with your NonStop System administrator to determine the correct TCP/IP stack process name. Note that multiple stacks may exist on the system (i.e. you need the correct one and not any one).

The TCP/IP port is already in use

Shortly after starting the MOMI server, check the EMS log on the NonStop System. If MOMI encounters any errors accessing the TCP/IP socket (i.e. port), messages are logged to EMS.

To determine if a TCP port is free, start a SCF from a TACL prompt and enter the following command:

```
SCF status process $ZTC0
```

Examine the output and find a port in the LPort column not used (greater than 1024 and less than 32767)

The TCP/IP address entered in the client is not correct

Check with the system and/or network administrator to confirm the address(es) for the TCP/IP stack.

It is possible that your network re-maps addresses. Literally, the NonStop System address may not be directly accessible, but an alternate mapped address is referenced.

The TCP/IP address and port are both correct and not already in use but I still can't connect

This may indicate a firewall and/or firewall software is in-between your MOMI PC Client and the NonStop Server. The firewall may be running on your PC and/or is a physical box on the network in-between your PC and the NonStop Server. You can try to PING from your PC to the NonStop Server by opening up a command prompt window and entering the following (using the example IP address above):

PING 192.168.20.10

Unfortunately, it is possible that the PING works but a MOMI connection fails as a PING is a different type of data (ICMP data) and the network firewall allows but MOMI data flow is not (TCP with the specified port).

The best bet, assuming the required TCP/IP address and port information is valid and the MOMI server is started and successfully running on the NonStop System, is to speak with your network and/or PC administrators and determine what procedures are required to enable a new TCP/IP network application within your environment. It is very possible that both local PC and network authorization is required.

My client successfully connects but data does not flow

In some environments, the old MOMI default port **2000** is used internally used by network itself.

In the CONFMOMI file alter the port to another value (select a free port on the NonStop side), restart the MOMI server (wait a bit) then try to reconnect in the client using the IP and new port. **Note that network firewall changes may also be required when the port is changed.**.

Information provided by a customer: Basically TCP port 2000 is a default port for Skinny Client Call protocol (SCCP) traffic. SCCP is a Cisco proprietary protocol for VoIP. It looks like any current FW or NGFW is going to inspect traffic on this port and treat it as VoIP traffic.

What things may stop MOMI data flow?

Generally speaking, firewall and Anti-Virus port scanning software.

Network firewalls in-between corporate PC's and NonStop Systems are common place. This level of security means that any new application must be enabled. Literally, firewalls prevent communication until authorized.

Anti-Virus port scanning software can be configured to examine all TCP/IP port traffic. The examination itself may interfere with MOMI's communication protocol by erring on the side of caution.

In both cases, environments running protections of this type would have a defined procedure available (i.e. forms to fill out) to establish a new software product. Since Introducing new software applications such as MOMI is probably not an everyday occurrence, don’t be surprised if the required underlying tasks need to be dug out from a drawer and 'dusted off' :)

Configuration recommendations

Overview

Several configuration settings should be reviewed in your Windows environment to insure that MOMI and other programs can operate to the fullest.

24 bit color or better

The PC must be configured with at least 24 bit color for the client to operate fully. A warning message is displayed at client start up if a lower color depth is detected. While the client will generally function correctly, certain features are not available such as the a pop up window when the mouse is placed over the connector lines between systems on the SubSystems / Expand / Diagram page.

GDI & USER Handles

MOMI makes extensive use of the Windows resource of GDI and USER handles. These values each have a default limit of 10,000.

Increasing these limits allows a Windows program to have more simultaneous graphically intensive screens. This limit may also need to be increased if accessing the Windows desktop remotely via software such as Citrix.

Important: The information below requires modification of the Windows registry. Serious problems might occur if you modify the registry incorrectly. Make sure that you take care and change only the fields recommended. For added protection, back up the registry and/or key being modified before you implement this change.

1. Run Registry Editor (REGEDT32.EXE).

2. From the HKEY_LOCAL_MACHINE subtree, go to the following folder:

   \SOFTWARE\Microsoft\Windows NT\CurrentVersion\Windows

3. On the right hand side of the screen double-click on the key:

   GDIProcessHandleQuota

4. On the pop-up window

   a) select a Base of Decimal
   b) in the "Value Data:" box enter 15000 (or larger).
   c) press OK

5. On the right hand side of the screen double-click on the key:

   USERProcessHandleQuota

6. On the pop-up window

   a) select a Base of Decimal
   b) in the "Value Data:" box enter 15000 (or larger).
   c) press OK

7. Exit the Registry Editor and reboot the PC for the changes to take effect.

Index

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

Z