BNCS User Guides

Atos IT Solutions and Services

BNCS Workstation Manager Documentation

From an original application by David Yates

BNCS

Contents

BNCS

1 Introduction

This document describes the start-up procedure for a BNCS Runtime machine (i.e. a machine that is either client or driver).

This procedure is based around the Workstation Manager but other external applications are called by it.

2 Workstation Manager

Usually this application would be run from a Start Menu->Startup shortcut.

Its purpose is to manage the machine – run and shutdown other applications, and update code and configuration on that machine.

Workstation Manager runs for the entire time that BNCSl applications are running.

It also has the opportunity to confirm correct installation such as environment variables etc. and do a daily deletion of old files.

From version 4.5.6.0 the status of the applications run by workstation manager will also be maintained and notified to a central workstation tracker application.

From version 4.5.6.0 Workstation Manager provides a number of "test program" options, allowing come modification of the normal start-up sequence.

2.1 Test Program Options

There are a number of "Test Programs" available, which allow changes to the following parts of the launch sequence.

- Stop all programs
- Clear the Launcher dialog
- Synchronize from the server (if one is configured)
- Load the Launcher
- Start any programs as configured

3 Workstation Manager In Use

Figure 1: Workstation Manager Running shows Workstation Manager in use.

The upper part of the display, within the Status frame, is referred to here as the Launcher.

Each line represents one application configured in launch.xml

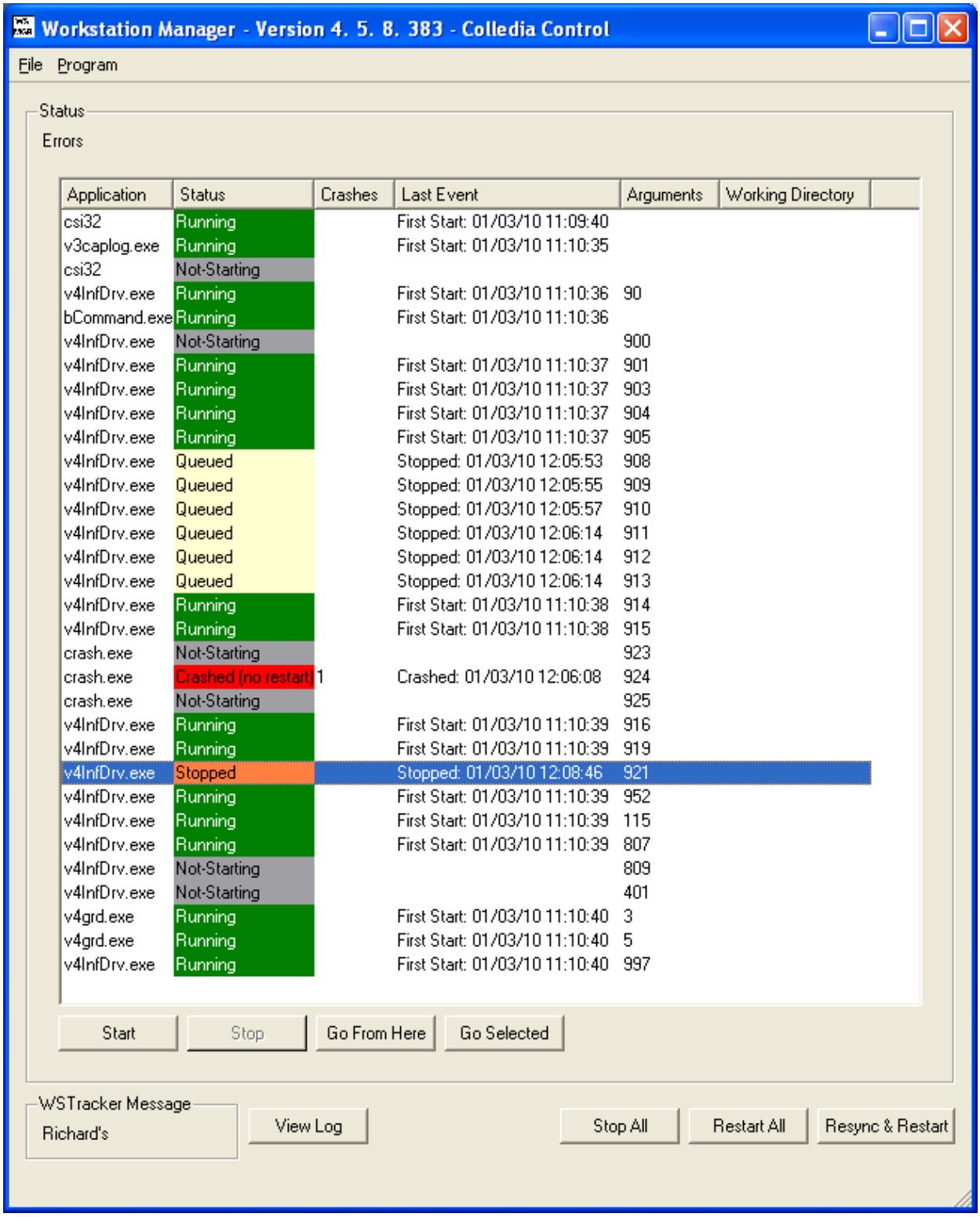

Figure 1: Workstation Manager Running

3.1 Status Column

The Status column indicates the current state of each application.

3.1.1 Running

The application has been started, either automatically or manually.

3.1.2 Not-Starting

The application has been configured not to auto-start, and has not been started manually.

3.1.3 Queued

The application has been configured to auto-start, but has not yet started. Or it has been stopped.

3.1.4 Stopped

The Stopped status appears briefly when an application is stopped manually. It is then replaced with either Queued or Not-Starting, according to its auto-start setting.

3.1.5 Crashed

The application has crashed. There are variations on this message according to whether or not the application is set to restart.

3.2 Last Event Column

The Last Event column shows a message indicating the most recent event.

3.3 Launcher Buttons

Applications may be selected using this mouse within the Launcher window.

Shift-click may also be used, to select a range of items.

Ctrl-click may also be used, to select multiple individual items.

3.3.1 Start

Start the currently selected applications.

This button will be disabled if none of the currently selected applications are running.

This action takes no notice of the auto-start settings for applications. It will attempt to start them even if auto-start is off.

3.3.2 Stop

Stop the currently selected applications.

This button will be disabled if all of the currently selected applications are already stopped.

3.3.3 Go From Here

Starts any application that is not currently running and has auto-start set, from the currently selected application to the end of the list.

This option is only available if only one application is selected, and there are applications later in the list that are not running but do have auto-start enabled.

3.3.4 Go Selected

Starts any selected application that is not currently running and has auto-start set.

This option is only available if there are applications selected that are not running but do have auto-start enabled.

3.4 Main Buttons

The buttons in the lower part of the display are the main overall controls.

3.4.1 Stop all

- Stops all applications
- Clears and deletes the Launcher

3.4.2 Restart All

- Stops all applications
- Clears and deletes the Launcher
- Reloads the Launcher
- Attempts to start all applications that are set to auto-start

3.4.3 Resync & Restart

- Stops all applications
- Clears and deletes the Launcher
- Preforms a file-sync if there is a server configured
- Reloads the Launcher
- Attempts to start all applications that are set to auto-start

3.5 Context Menu

Right-clicking on any application item will show the context menu.

Most of the entries are the same as the Launcher Buttons, with the following additions.

3.5.1 Set Auto-Start

Sets the auto-start property to true

This is only available if one or more of the selected applications has auto-start set to false.

3.5.2 Set No Auto-Start

Sets the‟auto-start property to false.

This is only available if one or more of the selected applications has auto-start set to true.

4 Difference between Start and Go?

The Start and Go actions have a couple of differences.

The most obvious is that Start will attempt to start on any application, but Go will only do so if the application is set to auto-start.

The less obvious difference is that Start will act on all the applications almost simultaneously, but Go will be much clearer about starting them in order. An application will only be started when the previous selected one has reported to Workstation Manager that it is running. This is the same logic that is used by the program start functionality triggered by the Restart All or Resync & Restart buttons, or what the program does on a normal start-up.

5 Startup Sequence

The start-up procedure of workstation manager is shown below.

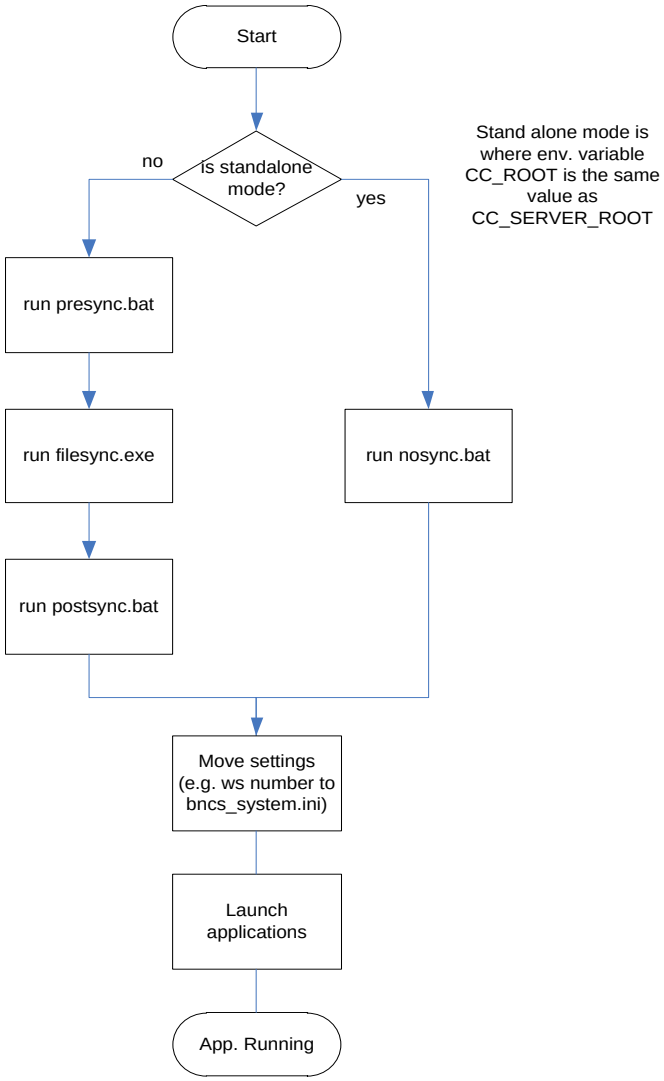

Figure 2: Workstation Startup Sequence

6 Batch files

None of the batch files noted need to exist in order to run a system – if they don't exist that step is simply skipped.

These batch files are located in %CC_ROOT%\%CC_SYSTEM%\windows\bin.

None of the batch files listed should contain any entries that might cause them to stop responding e.g. waiting for user input, or use of a "pause" command. If they do workstation manager will be seen to pause indefinitely waiting.

6.1 presync.bat

This batch file can be used for any purpose (not that I can think of a sensible one at the moment….).

Note: this file is copied off the server along with everything else but has already been run before any updated version is copied by the filesync. A workstation must be restarted twice for an updated presync.bat to be run.

6.2 postsync.bat

Following completion of the file sync process a post-sync step tidies up the copy, moving files to their correct locations (e.g. to the windows directory for those applications that need it). This may be used to move BNCS V2 files that must exist in the Windows directory.

A common entry for this batch file is to copy workstation specific settings from their server location to the workstation location. For example every workstation has a "bncs_system.ini" file – to be stored on the server this must be stored in a directory for each workstation (you can't have 999 copies of a file of the same filename otherwise). The postsync.bat file moves this file so that it appears where applications expect to find it.

The following example is a generic postsync.bat entry:

```
xcopy
"%CC_ROOT%\%CC_SYSTEM%\CONFIG\WS\%CC_WORKSTATION%\bncs_system.ini"
"%CC_ROOT%\%CC_SYSTEM%\CONFIG\system" /c/i/f/y
```

Note the use of not only the system root/path environment variables but also the CC_WORKSTATION environment variable in the path. So for a CC_ROOT of "c:\bncs", and a CC_SYSTEM of "v4.5" for workstation 100 the copy would be from:

```
c:\bncs\v4.5\config\ws\100\bncs_system.ini
```
to
```
c:\bncs\v4.5\config\system\bncs_system.ini
```

It may be that there is a workstation specific post-sync step that should be run and this can be achieved with a line like:

```
"%CC_ROOT%\%CC_SYSTEM%\CONFIG\WS\%CC_WORKSTATION%\wspostsync.bat"
```

for tidiness however this is usually preceded by a check to see that this file exists before running it so the full (if slightly messy entry) would be:

```
IF EXIST
"%CC_ROOT%\%CC_SYSTEM%\CONFIG\WS\%CC_WORKSTATION%\wspostsync.bat" CALL
"%CC_ROOT%\%CC_SYSTEM%\CONFIG\WS\%CC_WORKSTATION%\wspostsync.bat"
```

6.3 nosync.bat

If this machine is running in stand-alone mode then this generic batch file is run.

7 File Sync Manager (*bncs_filesync.exe*)

This attempts to synchronise the file structure on this machine with the file server. This is a proper **synchronise, deleting** files that do not exist on the server. It's important to

remember that this is a synchronise process not just an update. This is particularly important if you are a developer wanting to run workstation manager - see below.

There is a list of directories that are excluded from the sync process because they contain workstation specific, temporary or redundant data.

The excluded directories are:

- source
- temp
- docs
- data
- install
- backup
- logs

8 Move Settings

This is where workstation specific settings that may have been overwritten by files copied off the server are corrected.

Hardware (this PC) settings are stored in %CC_ROOT%\local_settings.ini which is outside of the directory structure that is synchronised with the server.

The settings currently moved are:

Setting	**From**	**To**
Adaptor Number	File: local_settings.ini Setting: [Network] DefaultAdaptor=x	**File:** csi.ini (in windows dir or config location specified in V3 "c:\bncs_config.ini") Setting: [Network] DefaultAdaptor=x **File:** BBC_NBIF.ini (in windows dir) Setting: [NetBIOS] Adaptor=x **File:** BBC2NBIF.ini (in windows dir) Setting: [NetBIOS] Adaptor=x **File:** bncs_system.ini (in current system config\system dir) Setting: [Network]

		DefaultAdaptor=x
Workstation Number	Environment Variable: CC_WORKSTATION	File: csi.ini (in Windows dir or config location specified in V3 "c:\bncs_config.ini") Setting: [Network] Workstation=99 File: bncs_system.ini (in current system config\system dir) Setting: [Network] Workstation=99

9 Running Workstation Manager as a Developer

Normally a workstation knows nothing about anything and has to get it's configuration off the system server. This is fine since the system server always has the current or latest system configuration.

As a developer however you might easily have a different configuration on your machine – with many later files. In this case workstation manager can be set to work in "Stand alone" mode by setting CC_ROOT and CC_SERVER_ROOT to the same value (this can be done with the bncs_inst_env utility)

10 Updating Workstation Manager and File Synchroniser

The problem with leaving bncs_ws_man in charge of launching everything is that the file synchroniser can't update the workstation manager or itself (or DLL dependencies).

Version 4.5.19 updates the mechanism used so that workstation manager/file sync/DLLs can be updated without a reboot. This requires the utility bncs_deferredCopy.exe – this logs it's activity to /logs/bncs_deferredCopy.

The process for bncs_deferredCopy is this:

- The file copying application will fail to copy in-use files and so makes a log of those it had a problem with.
- Workstation manager notes the error code from the file copying application and runs the deferred copy application – as it does so ws man exits so freeing up in-use files.
- The deferred copy application tries up to 20 times to copy the files listed in the log (over a period of about three minutes to allow for slow-to close applications). Usually the files copy in a very few seconds.
- The deferred copy application, on success or failure re-runs the bncs_ws_man application.

To save update loops if the deferred copy application is run 5 times consecutively *with errors* it won't run again – it does however fall back to the original update mechanism (described immediately below). Subsequent successful file copy operations resets this count.

Default step is included in bncs_ws_man which it checks the time/date of an installer program CC_WS_Manager_Update. If there is a new version detected of this application then it is run. CC_WS_Manager_Update is an Inno-Setup installer application that is configured to be entirely automatic and will reboot the machine so that the workstation manager and file synchroniser can be replaced.

11 Launching applications

When the file sync is complete, bncs_ws_man.exe then runs applications directly.

Here's a sample launch file:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE instances SYSTEM "launch.dtd">
<launch>
	<workstation id="1" csi="csi32" >
		<application name="v4infdrv" workingdir="" arguments="123" />
	</workstation>
	<workstation id="2" csi="csi32" rebootoncrash="true" >
		<application name="v4infdrv" workingdir="" arguments="123" />
	</workstation>
	<workstation id="99" csi="csi32" >
		<application name="v3infdrv" arguments="1" />
		<application name="v3infdrv" arguments="2" delay="2000"
waitForIdle="30000"/>
		<application name="myApp" workingdir="c:\" arguments="4" delay="100" />
		<application name="docrash" arguments="auto" restartoncrash="true" />
	</workstation>
</launch>
```

Every workstation on a cluster uses the same launch file and the workstation number is used to identify the section. There are three workstations listed in the above launch file, 1, 2 and 99. Workstation number is taken from the CC_WORKSTATION environment variable.

CSI is taken to be such a core component that this is run regardless. No separate entry is required in the following section – nor is it allowed. CSI must be allowed to fully run up before any other apps can start so there is special checking to ensure that this happens.

Each <application> line is then interpreted running each application in turn.

Let's pick apart a single entry:

```
<application name="app" workingdir="c:\" arguments="4" delay="100"
restartoncrash="true"/>
```

name: This runs an application called "app" this is assumed to be "in the path" and normally %CC_ROOT%\%CC_SYSTEM%\windows\bin. Explicit paths can be set but that's kinda missing the point of having all under one tree with relative paths to make the tree portable.

workingdir: (optional parameter) the working directory for each application can be independently set. From version 4.5.26 environment variables are expanded appropriately.

arguments: (optional parameter) the argument(s) is a space delimited set of parameters. Note that spaces are not permitted in individual arguments even if they're quoted (this is a bug).

From version 4.5.26 environment variables are expanded appropriately.

delay: (optional parameter) You can set a delay of how long to wait (in mS) before trying to launch the next application in the list. Think of this as a "settling delay" for this application.

waitForIdle: (optional parameter - from ws manager 4.5.17) this is a Windows function call that waits for an application to initialise before continuing. The value here is the maximum time in milliseconds to wait before carrying on regardless. Drivers will often start well before the timeout allowing workstation manager to continue - this is therefore preferable to using fixed delays. The waitForIdle delay can be safely set to a large value but (almost certainly) won't delay the normal launch of the applications.

You'd normally use this value or the delay value but if you want to use both that's ok too!

restartoncrash: if "true" then workstation manager will catch crashed programs and attempt to restart them up to 10 times. This is only suitable for especially adapted applications that are set to silently crash, logging errors to disk and not putting up dialogs. This option is not suitable at all for 16 bit applications.

12 Other Attributes, for Testing

There are other attributes that may be used in the workstation and application elements, generally for testing purposes.

12.1 Workstation Attributes

Name	Values permitted, or type, and default	Description
autostart_csi	"true" or "false" If absent, a value of true is assumed.	Controls whether csi will be started automatically on this workstation, using settings from the workstation element. Since nearly all BNCS applications need to connect to CSI to run it is assumed that if this attribute is set to true it is because CSI is being run manually, separately from the Workstation Manager.
autostart_ws	"true" or "false" If absent, a value of true is assumed.	Controls whether any of the applications will be automatically started. If set to false, the Launcher will be set-up based on the contents of launch.xml, but no applications will be started automatically. They may then be started manually using the context menu, or the Start button.

Application Attributes

Name	Values permitted, or type, and default	Description
autostart	"true" or "false" If absent, a value of true is assumed.	Controls whether the application will be started automatically on this workstation, using settings from the application element. Setting autostart to "false" will cause the application not to be started, but it will ramain present in the list. It may then be started manually using the context menu, or the Start button.
Csi	Csi program name, eg "csi32", "v4csi", "v2csi"	This should only be used here if it is required that csi is not started as a result of an attribute in the workstation element. In that case the workstation element's csi attribute should be set to "none", or set blank or deleted.

Since nearly all BNCS applications need to connect to CSI to run they will not start if CSI has not been started. Hence it is assumed that if the autostart_csi attribute is set to true, CSI is being run manually, separately from the Workstation Manager.

12.1.1 Starting CSI from an application element, not from workstation

In some circumstances it may be necessary that CSI is not the first application to be started; perhaps CapLog should be started first. To do this it is necessary to indicate that the workstation element does not imply that CSI should start. This is done by the following changes

1. Set the workstation's "csi" attribute to "none", or remove it. This will stop csi being started using information from the workstation element.
2. In the application element, set the "name" to "csi". This will cause the application handling to be set to be csi-specific and slightly different from other applciations.
3. In the application element, add a "csi" attribute with its value set to the required csi application name – "csi32", "v4csi" etc.

13 Configuration Editor

Note that these new features (ie Test Programs and new attributes) are not currently supported by the Configuration Editor plug-ins. These must be edited by hand. They are likely to be lost if launch.xml is edited using the Configuration Editor.

14 Rebooting the machine on CSI crash

From version 4.5.1.0

On CSI crashing workstation Manager can be configured to restart the workstation. This requires

rebootoncrash="true"

to be configured in launch.xml for the CSI entry. (Note this is **rebooton**crash not **restarton**crash)

e.g.

```
<workstation id="2" csi="v4csi" rebootoncrash="true" >
…
…
```

Setting CC_DEVELOPER environment variable to true will disable this restart feature and put up a dialog box instead.

The first reboot attempt is "clean" in that workstation manager will wait for applications to close before restarting.

Subsequent reboot attempts are "unclean" in that they forcibly try and restart this workstation. This is so unattended workstations will reboot even if left in an unsuitable state. These attempts occur every 20 seconds following the first attempt.

15 Command-Line Options

Workstation Manager has a couple of command-line options available to control how the program acts when started.

15.1 -test

The –test option causes the program not to do anything at startup. No configuration is loaded, nor programs run.

This leaves it in essentially the same place as at the end of a Stop All sequence.

15.2 -load

The –load option causes the program to resync and load the launch configuration but not to start any programs.

15.3 resync/restart

Running an instance of bncs_ws_man even if one is already running with this command line will cause it to be signalled to resync and restart

16 Test Programs

A number of Test Programs are available from the "Programs" menu. These allow changes to the following parts of the launch sequence.

- Stop all programs
- Clear the Launcher dialog
- Synchronize from the server (if one is configured)
- Load the Launcher
- Start any programs as configured

16.1 Stop All

Stop any currently running programs.

Clear the launcher.

(Finishes with all programs stopped)

16.2 Restart All

Stop any currently running programs.

Clear the launcher.

Reload the launcher from launch.xml.

Start any programs configured to auto-start.

(Finishes with all auto-starting programs started)

16.3 Resync & Restart

Stop any currently running programs.

Clear the launcher.

Resync from the server (if one is configured).

Reload the launcher from launch.xml.

Start any programs configured to auto-start.

(Finishes with all auto-starting programs started)

16.4 Stop without Clearing Launcher

Stop any currently running programs.

(Finishes with all programs stopped)

16.5 Stop (if running) and Reload Launcher

Stop any currently running programs.

Clear the launcher.

Reload the launcher from launch.xml.

(Finishes with all programs stopped)

16.6 Stop (if running), Resync and Reload Launcher

Stop any currently running programs.

Clear the launcher.

Resync from the server (if one is configured).

Reload the launcher from launch.xml.

(Finishes with all programs stopped)

16.7 Restart without Reloading Launcher

Stop any currently running programs.

(Don't consider synchronizing , clear or reload the launcher)

Start any programs configured to auto-start.

(Finishes with all auto-starting programs started)

17 Closing down

On closing down bncs_ws_man.exe it will attempt to close down all applications that it has launched.

18 Logging

This application logs all interesting activity in:
%CC_ROOT%\%CC_SYSTEM%\logs\Workstation Manager

19 Deleting Logs

This application can be configured to delete old files if required, the configuration for this is in logdelete.xml, an example is shown here.

```
<?xml version="1.0" ?>
<!DOCTYPE logdelete SYSTEM "logdelete.dtd">
<logdelete>
	<workstation id="global" enable="true" daysback="24">
		<dir name="c:\bncslogssssss"/>
	</workstation>
	<workstation id="10" enable="true" default="false" daysback="24">
		<dir name="c:\bncslogsssssss"/>
		<dir name="blah"/>
	</workstation>
</logdelete>
```

There is an entry for each workstation, if that entry does not exist for a workstation the global entry is used instead. It is expected that most installations will only require the global entry.

To enable log deletion the "enable" setting needs to be set to true. When enabled all files in the default and configured directories will be deleted if they have not been modified within the number of days back specified in "daysback".

The default directories are the current system "logs" directory and the directories used by caplog. The caplog directories are determined by looking at the caplog.ini file in the windows directory and the directory indicated in "c:\bncs_config.ini." If you want to ignore the default directories set the optional parameter"default=false".

Additional directories can be listed as required. If these directories are not absolute paths they are assumed to be in the current system and will have the values of CC_ROOT and CC_SYSTEM pre-pended to them.

20 Workstation Monitoring

From version 4.5.6.0 workstation monitoring has been introduced, this has the ability to report to other systems the status of the applications. Note that the status is running or not, it does not include loss of comms. As part of this the workstation can also be remotely stopped and restarted, the restart also includes doing an update.

Within the information sent by workstation manager is a text string, if this is set on the server to a known value once a workstation has successfully updated the string will be sent to the workstation tracker to confirm the update.

20.1 Configuration

If there is no configuration for this the system will work without the monitoring.

In instances.xml there needs to be an entry for "workstation_tracker", the device number set here is what defines which device ini file is used to configure the monitoring as well as the separate workstation tracker application.

In the device ini file there is a section as detailed bellow.

`[WSTracker]`	Section header
`port=65508`	Tx port used. Rx port is this plus 1.
`Updateperiod=15`	Time period between sending updates
`main_server=192.168.1.123`	IP address of the main workstation tracker. Note to use broadcast messages use the relevant broadcast IP, i.e. 192.168.1.255.
`reserve_server=192.168.1.124`	IP address of the reserve workstation tracker.
`WS_Man_Text=build123`	Text string that can be set to confirm update. This should not contain the '|' character.

20.2 Network messages

To allow other applications to interact with this the networking is described, it is however primarily designed to be used with the workstation tracker application.

From the workstation manager the messages are in the format:-

`[CSI_channel]|[Workstation]|[Status]|[StartTime]|[WS_Man_Text]|[User]|[OS]`

Item	Example	Description
CSI_Channel	0	CSI channel number for the current system
Workstation	123	Workstation number (environment variable)

Status	2	integer as per the infodriver presentation.
StartTime	123453	Workstation manager start time in seconds from 1/1/1970 (QT see QdateTime::setTime())
WS_Man_Text	CCM_tag4	Text set in an ini file to identify the build
User	cc_runtime	User logged on to the machine
OS	v5.1 SP2	Version of Windows reported by the operating system

This message is sent on the port defined in the device ini file.

To the workstation manager the messages are in the format:-

`WS[workstation]-[action]-[delay]`

Item	Example	Description
Workstation	123	Workstation number (As confirmation message sent to correct WS)
action	stop	Action to be done, options are:- stop start restart
delay	100	Delay before re-starting, for a single workstation restart this will be set to 0. For other commands this is not required.

This message is sent on the port plus 1 defined in the device ini file.

21 Future enhancements?

These features may expect to have found their way into ws_man at some point.

- bncs_ws_man.exe then keeps listening for changes to the file server and on detecting a change may do the following:
  - update the file, notify applications there has been a change (configuration files mainly)
  - close and application, update that application, restart the application (drivers, binary UI components)
  - notify the user that a restart or re-launch is required

22 Document Version Control

Version	Date	Author	Comments
0	2004	DGY	Original bncs_ws_man documentation from which this derives
0.1	09/05/2007	MB	First Draft
0.2	21/11/2007	MB	Added workstation monitoring.
0.3	22/11/2007	MB	Updated to software version 4.5.6.1
0.4	27/11/09		Removes duplicate of version information
0.5	7/1/10	RK	Added mention of autostart attribute.
0.6	1/3/10	RK	Added other new attributes. Added “Test Programs” and explanations.
0.7	13/10/15	DGY	Adds note on resync/restart command line option