Network Station Manager Version 2

Configuration Files Overview

Network Station Education
IBM Network Computer Division
June 1999

Objectives/Contents

- V1R3 vs. V2R1 Configuration Files
- Terminology
- V2R1 Profiles and location
- Profiles Hierarchy
- Command Line Interface

Notes

The topic of this presentation is the Network Station Manager Configuration Profiles.

The objective of this topic is to provide a brief overview of the new V2R1 configuration profiles that have replaced the V1R3 configuration files.

These configuration profiles are managed directly by the Network Station Manager applications and there are even less reasons today to work directly with these files.

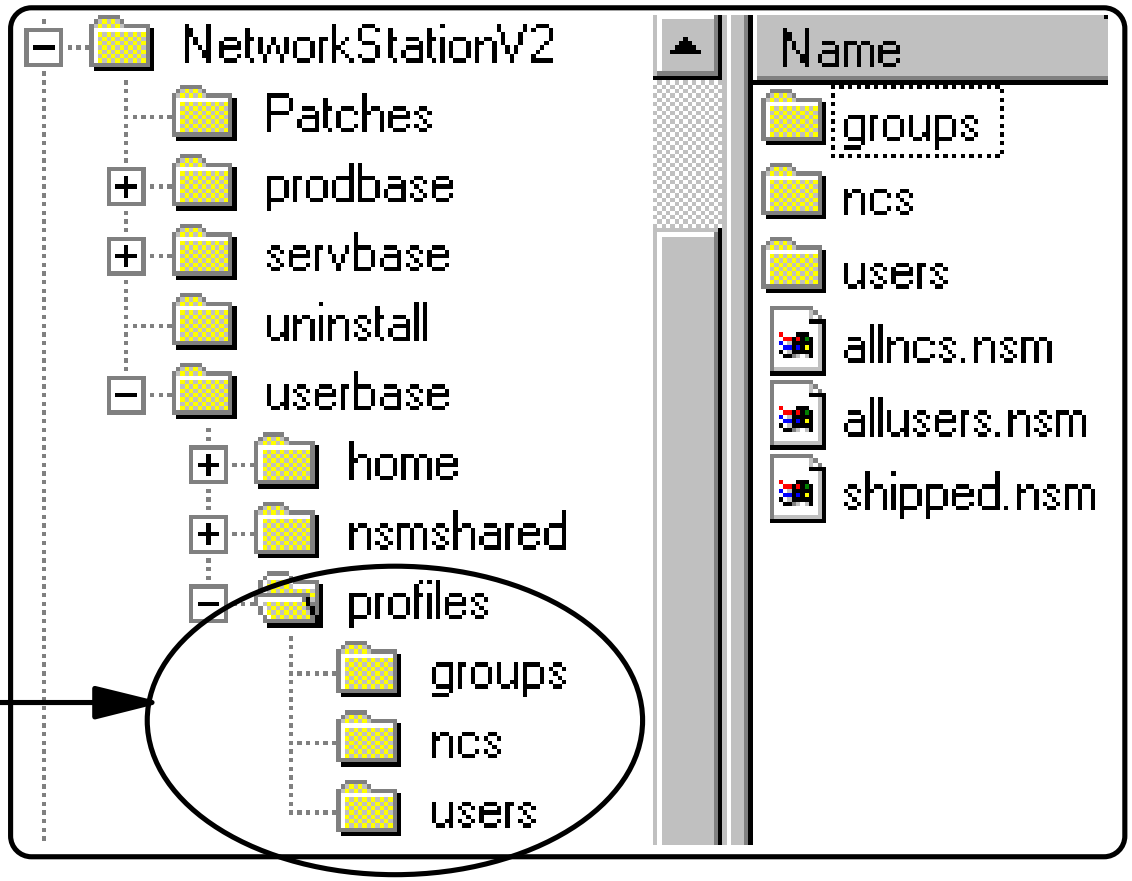
However, it is essential that an administrator understand what these files are and how they work for cases where there is a need to do some customization that is otherwise not available through other tools.

We take a look at the difference with V1R3, some terminology, the different profiles and their hierarchy as well as the new command line interface to make batch or interactive changes to the configuration profiles.

NSM V1R3 and NSM V2R1

- **In NSM V1R3**
  - Upwards of 45 configuration files
  - Many different formats
  - Located in some 15 or more subdirectories

- **In NSM V2R1**
  - Only 10 files (profiles)
  - Only one format (XML)
  - One directory and 3 subdirectories



Notes

The configuration files in V2R1 has been significantly simplified by reducing the number of files to just a few and by using a single common format.

These files are now called configuration profiles and they all reside in a single directory under ...userbase/profiles.

This chart shows that particular directory on a Windows NT system, along with its three subdirectories:

- The profiles that are applicable to all users and all stations, as well as the shipped defaults reside in the main directory /userbase/profiles
- Then there is one directory each for the group profiles, the individual user profiles and the individual unit (Network Station) profiles.

Terminology

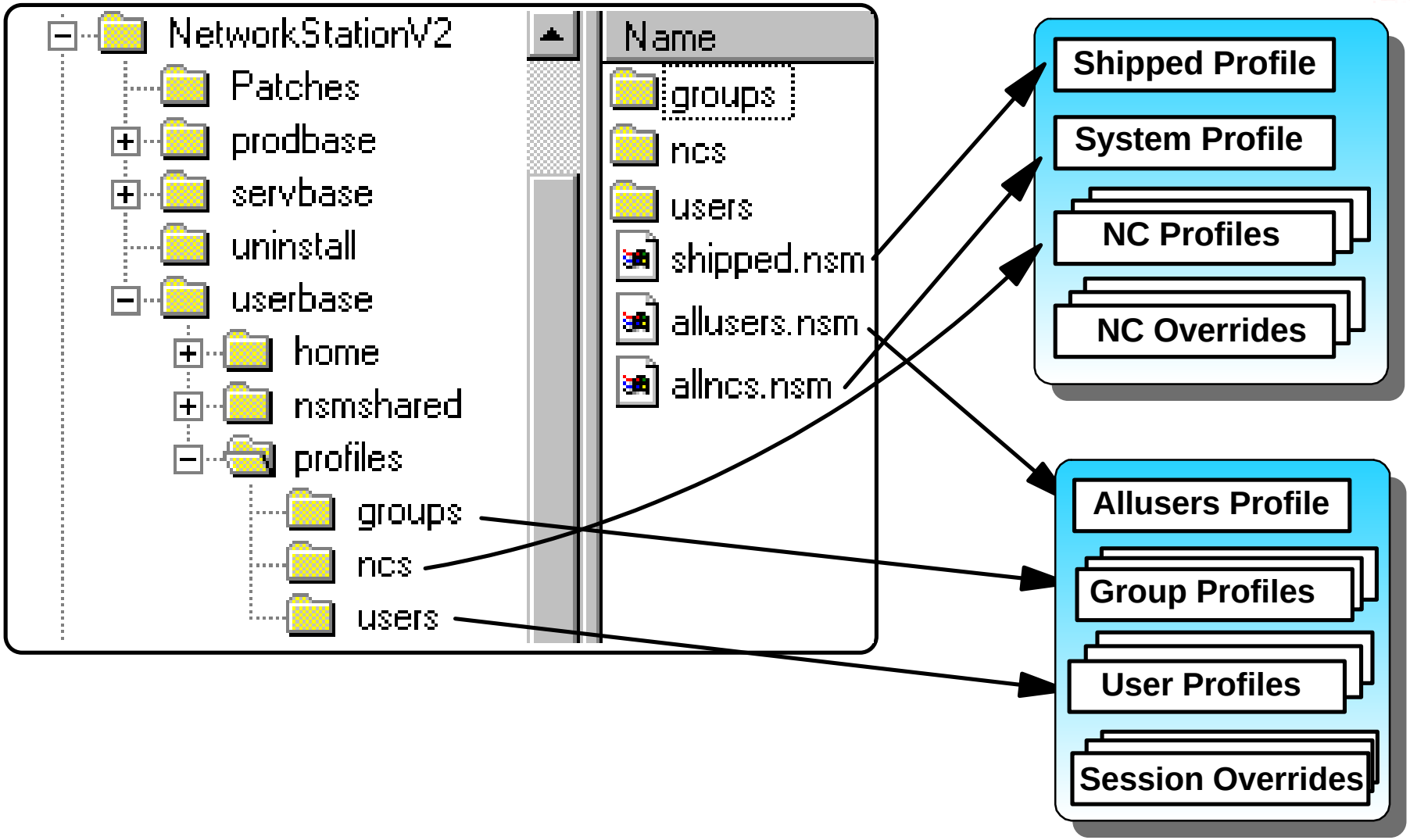
- **Download Profile**
  - One of several configuration files that replace the NSM V1R3 configuration files.
- **XML**
  - Extended Markup Language. An industry standard tag language used to define custom layouts that can be parsed by standard utilities. Download profiles are XML files.
- **Override File**
  - Similar to the V1R3 backdoor file. Provides a mechanism for setting configuration preferences outside of using NSM. Architected in V2R1 but disabled in the initial release.
- **Network Station Registry**
  - Repository on the client (Network Station) where the configuration parameters obtained from the download profiles are stored for access by local services and applications.

Notes

There are some new terms to be learned when talking about these new profiles, so here is a short terminology on the important ones.

- Download Profile is the new term for a configuration file
- Extended Markup Language (XML) is an industry standard tag language used to define custom layouts that can be parsed by standard utilities. Download profiles are all XML files.
- An Override File is similar to the V1R3 backdoor file and it provides a mechanism for setting configuration preferences outside of NSM. These however should seldom be needed, especially since there is another tool called NSMCL that can be used to work with these files.
- The Network Station Registry is a repository on the Network Station where the configuration parameters obtained from the download profiles are stored for access by local services and applications.

NSM V2R1 Profiles



Notes

There are different ways to look at these profiles or to categorize these profiles dependent on the particular aspect that we are looking at.

One way, when discussing downloading of the profiles, is to group them in two groups:

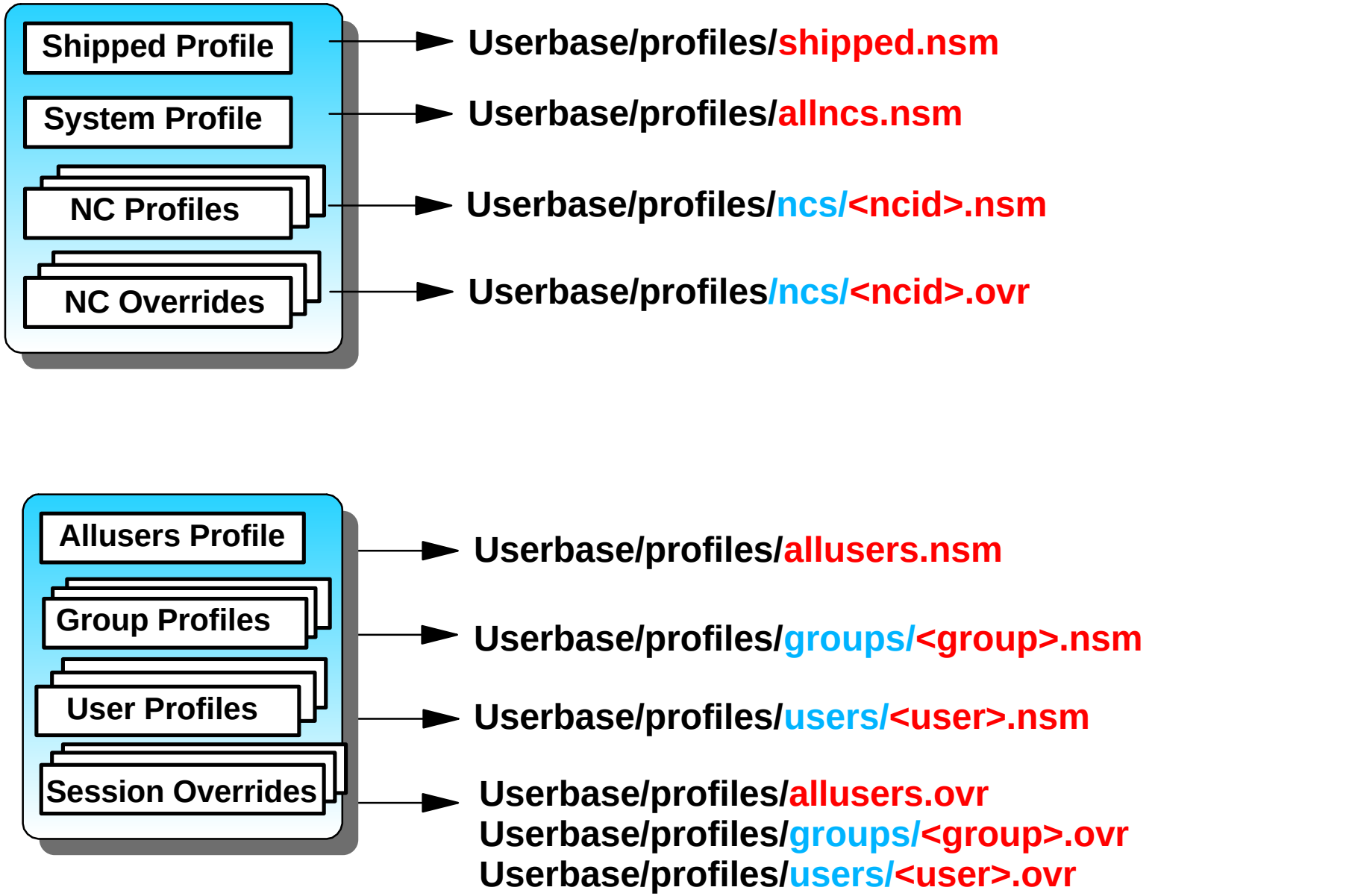
- Those profiles that are download before the user logs in
- Those that are downloaded after the user logs in

which is what we have represented here.

In other words, the shipped settings, the system settings that apply to all stations and the individual unit settings are profiles downloaded before the user login panel is presented.

After the login, since the system now is aware of who the user is, the all users, group and individual user profiles are then downloaded.

Download Profile Names and Path



Notes

This chart looks at the same groups of profiles and identifies the actual location where these profiles are stored.

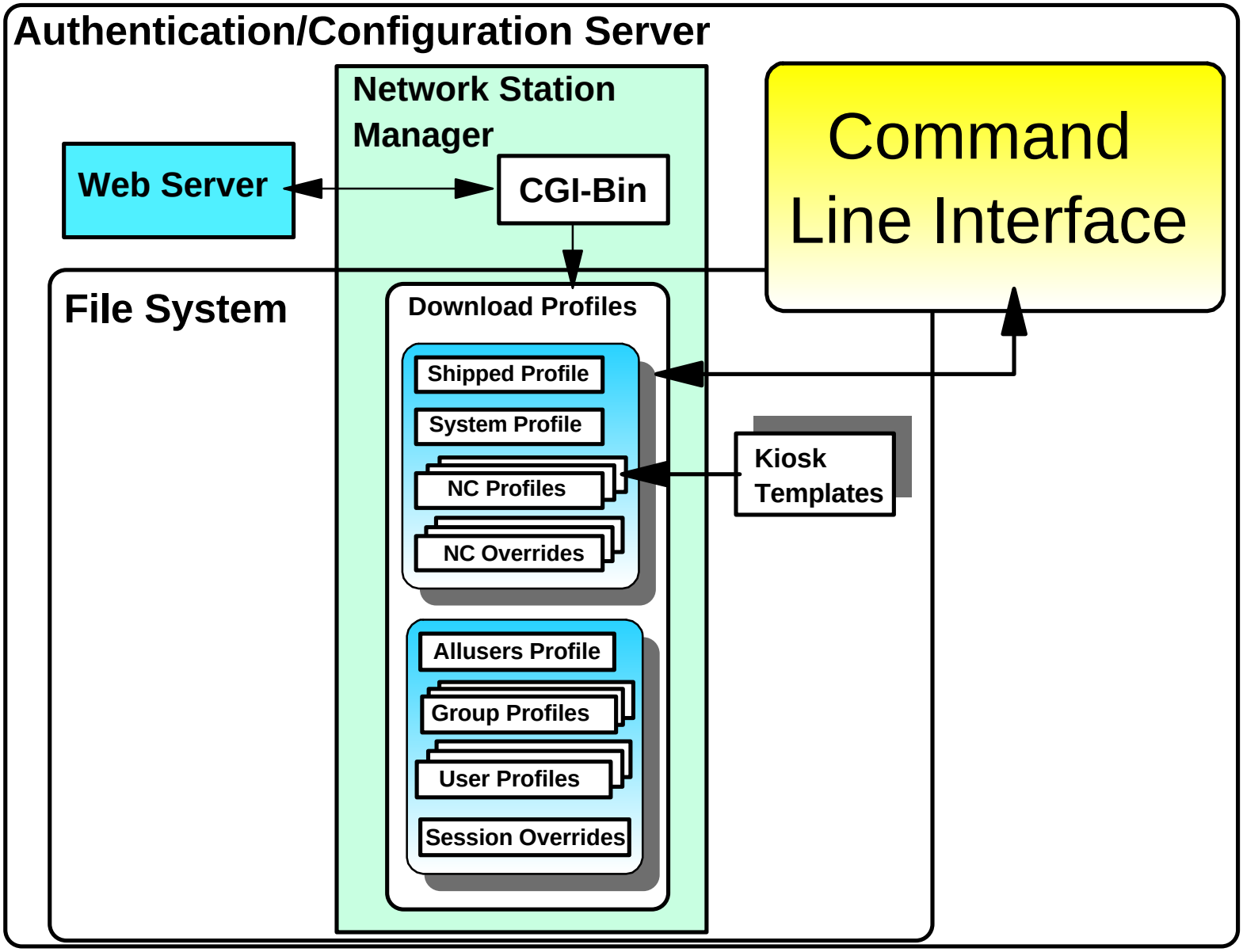
Notice that there are also override profiles listed, which we have not discussed yet.

These override profiles are similar to the backdoor files in V1R3 and are to be used for settings parameters that cannot be set through the NSM GUI interface.

However, since there is now a command line interface available to set or make changes to any setting, there should really be no need to use the override files except in very special cases.

In fact, these override files are disabled by default and have been architected into the product mainly for backwards compatibility reasons.

Authentication/Configuration Server



Notes

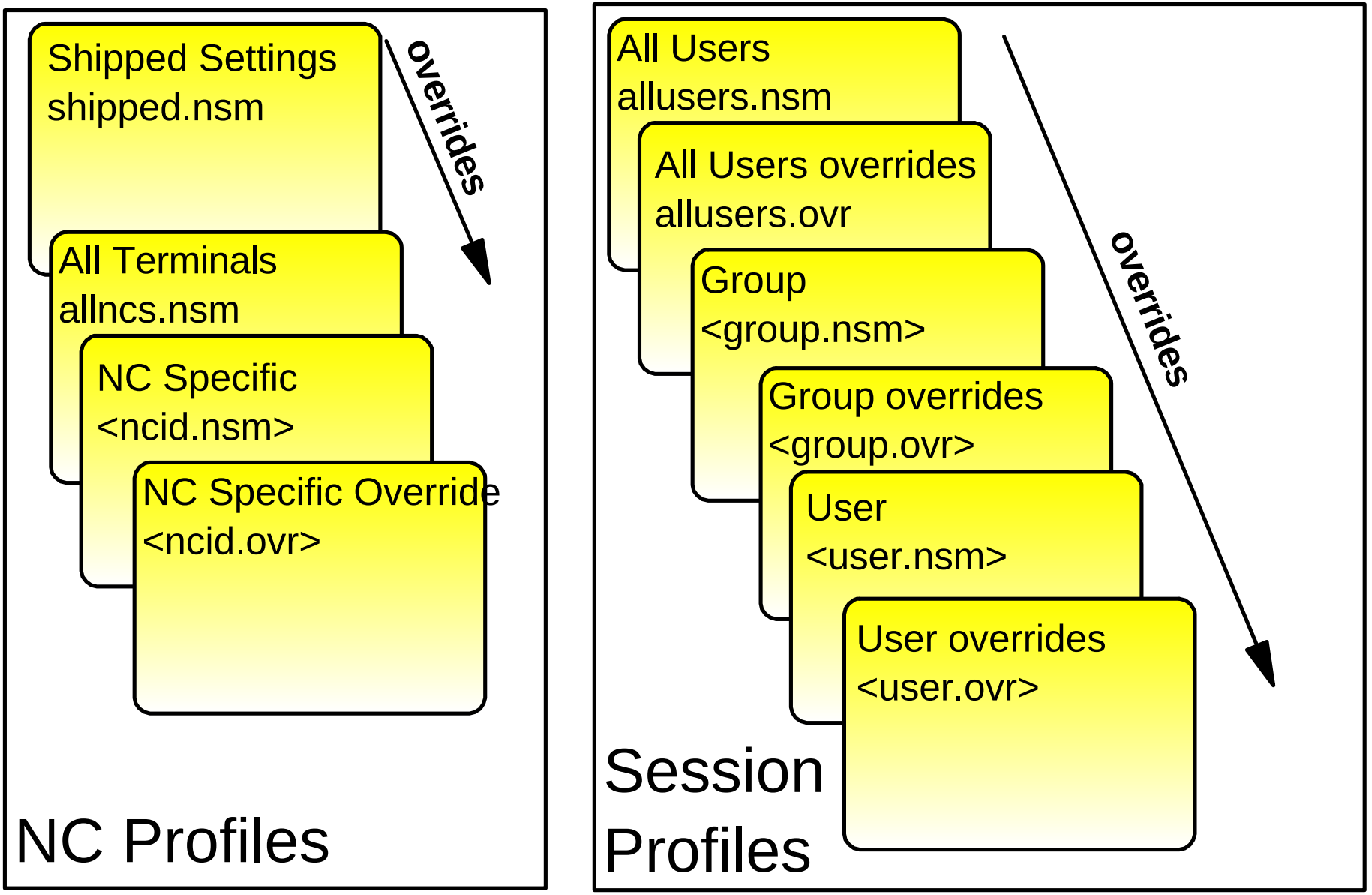
The download profiles are managed and manipulated by two entities:

- Either the administrator uses the Network Station Manager's Graphical Interface, which causes a set of CGI-bin programs to make the required additions and changes to the profiles
- Or the administrator uses the command line interface tool to make similar changes.

These profiles should never be altered manually. If a special situation requires manual changes to a profile, then the manual change should be made to an override profile.

This chart also shows an element called Kiosk templates. These are special profiles to operate the station in kiosk mode and we provide additional information in the kiosk presentation.

Profiles Hierarchy



Notes

As was the case in the previous release, the settings in the configuration profiles are processed in a specific order.

Some settings are additive, meaning that the total number of settings that the user finally gets is the sum of all the settings. For example, if each of three profiles each add an icon to the launchbar, the result will be that all three icons will appear on the launchbar.

Other settings are mutually exclusive, meaning that whatever setting is processed last is the setting that will be in effect when the desktop appears.

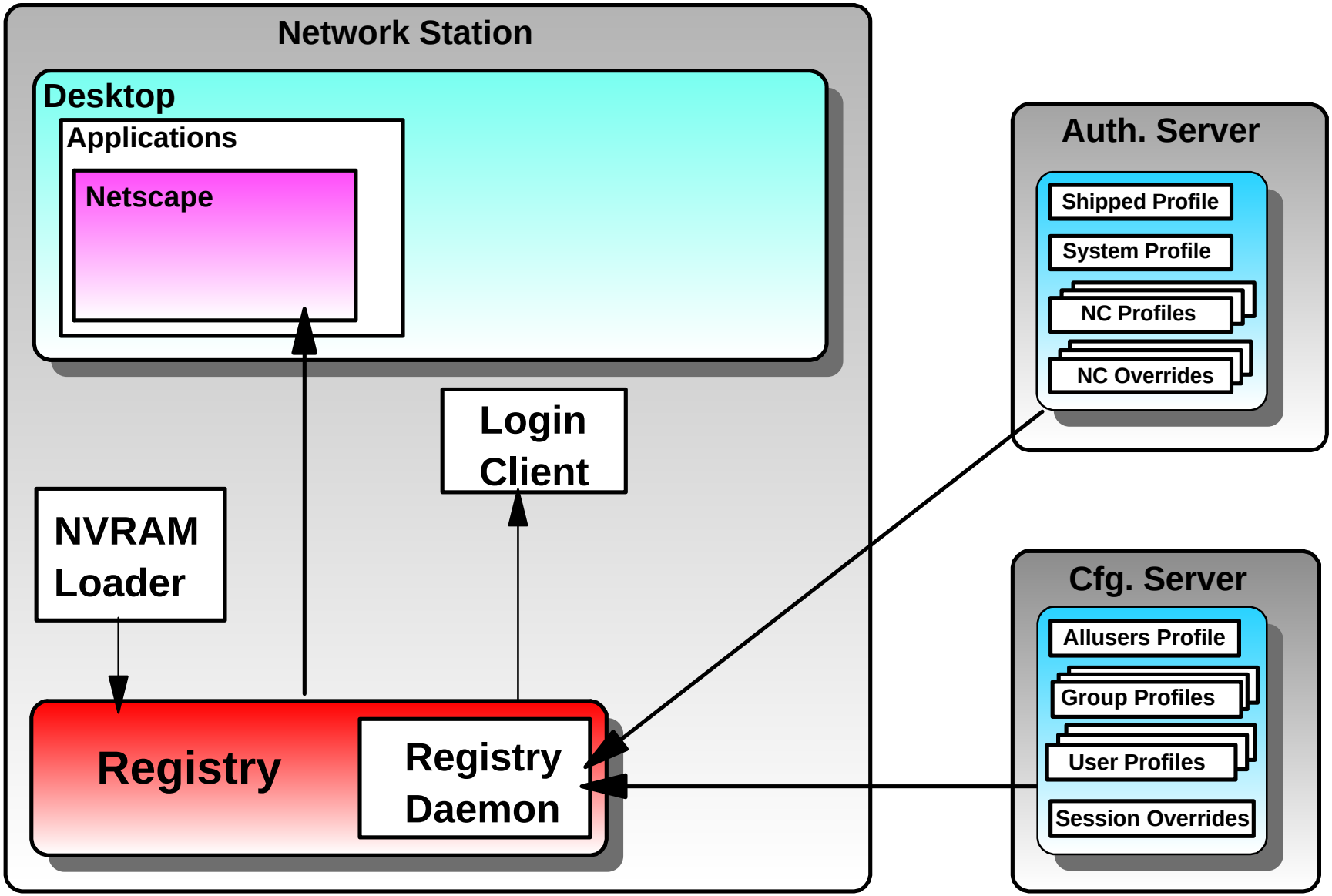
This chart separates the profiles into two groups:

- The NC profiles, which contain settings related to the unit
- The session profiles, which contain settings related to the user or group.

The shipped setting are somewhat separate but can probably be categorized with the NC profiles.

This chart shows the order of precedence for the mutually exclusive settings. If a setting is specified at the user level, it take precedence over all previous values specifies for that particular setting.

Downloading Profiles



Notes

In V2R1, the Registry is the one central place on the station where configuration parameters are stored and all components and applications access the registry to get configuration parameters.

The Registry Daemon is therefore the only component responsible for downloading configuration profiles.

This chart shows that the Registry daemon downloads the configuration profiles from the configuration server during the initial boot sequence, and after the user logs in, it downloads the session configuration profiles from the authentication server.

Applications then get their configuration parameters from the Registry as required.

Sample Download Profile

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE NCREGISTRY SYSTEM "registry.dtd" >
<NCREGISTRY VERSION="1.0">
<OBJECT NAME="/config">
<CATEGORY NAME="WORKSTATION">
<PROPERTY NAME="xserver-keyboard-type">5</PROPERTY>
<PROPERTY NAME="pref-screensaver-enable">true</PROPERTY>
<PROPERTY NAME="pref-screensaver-time">60</PROPERTY>
</CATEGORY>
</OBJECT>
</NCREGISTRY>
```

Notes

Here is an example of a download profile.

As shown, this is very similar to HTML format, only with different tags.

One can easily recognize that the parameter xserver-keyboard-type is set to the value 5 and that the pref-screesaver-enable is set to true, etc.

These configuration settings (actually property names) are part of the workstation category which is part of the config object.

Special Configuration Variables

Variable	Description	Values	Default Setting
NSM_ALLOW_OVERRIDES	Indicates if override profiles are allowed	ENABLE DISABLE	DISABLE
NSM_NC_NAME_TYPE	Indicates the type of file name used for a specific Network Station configuration	IP Address MAC Address IP Host Name ANY	ANY
NSM_ACCESS_NC_CONFIG	Indicates if terminal level configuration is used	ENABLE DISABLE	ENABLE
NSM_ACCESS_GROUP_CONFIG	Indicates if group level configuration is used	ENABLE DISABLE	ENABLE
NSM_ACCESS_USER_CONFIG	Indicates if user level configuration is used	ENABLE DISABLE	ENABLE

- These variables are initially set in the shipped profile
- They can only be changed using the Command Line Interface

Notes

This table identifies some special configuration variables that are initially set in the shipped.nsm file and that can be changed, if need be, using the command line interface.

The first one is the variable that controls whether override profiles are allowed or not. By default, this is set to disable.

The second one allows to specify the type of identifier that is to be used for the unit's filename. If any is used, all three possibilities will be tried in sequence.

Finally, the last three allow the disabling of a particular level of settings. For example, if group settings are never used, this variable is used to indicate that the system should not be looking for group related profiles, thereby increasing the efficiency of the download profiles processing.

NSM Command Line Interface

- **Utility (written in Java) to make batch changes directly to the XML-based download profiles (NSMCL)**
- **User interface; choice of:**
  - Interactive GUI interface
  - Run from the command line
- **Configuration change commands; choice of:**
  - Run individually
  - Run script files containing many commands
- **Scripting language is SGML (Standard General Configuration Language)**

Notes

The Network Station Manager Command Line interface (abbreviated NSMCL) is a utility written in Java to make batch changes directly to the XML-based download profiles.

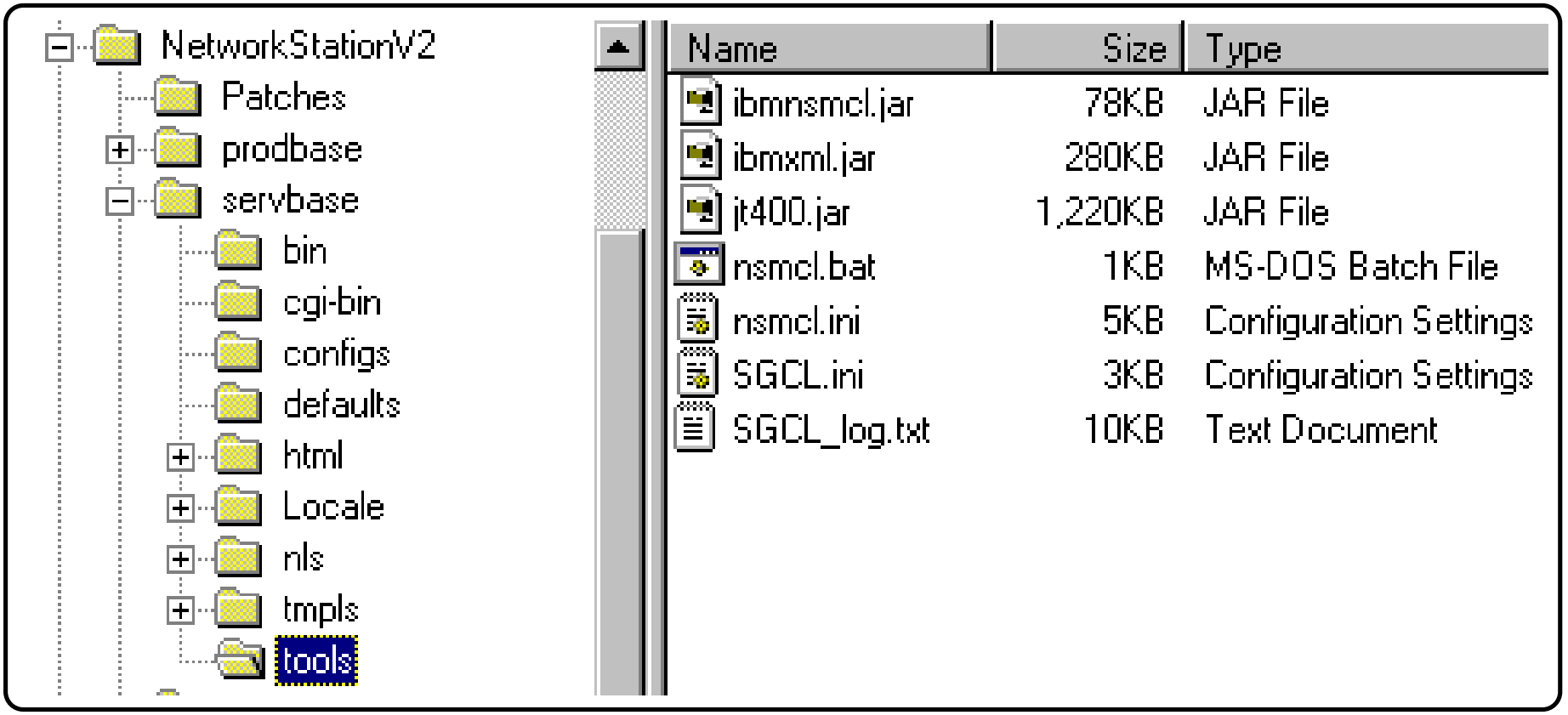
The user actually has the choice of two interfaces;

- Configuration commands can be run directly from the command line
- Or the user can use a graphical interface to enter the configuration commands

In addition, configuration change commands can be issued individually or a script file containing many commands can be specified as the input file.

The scripting language used is SGML (Standard General Configuration Language) Text.

The ../servbase/tools Directory



Notes

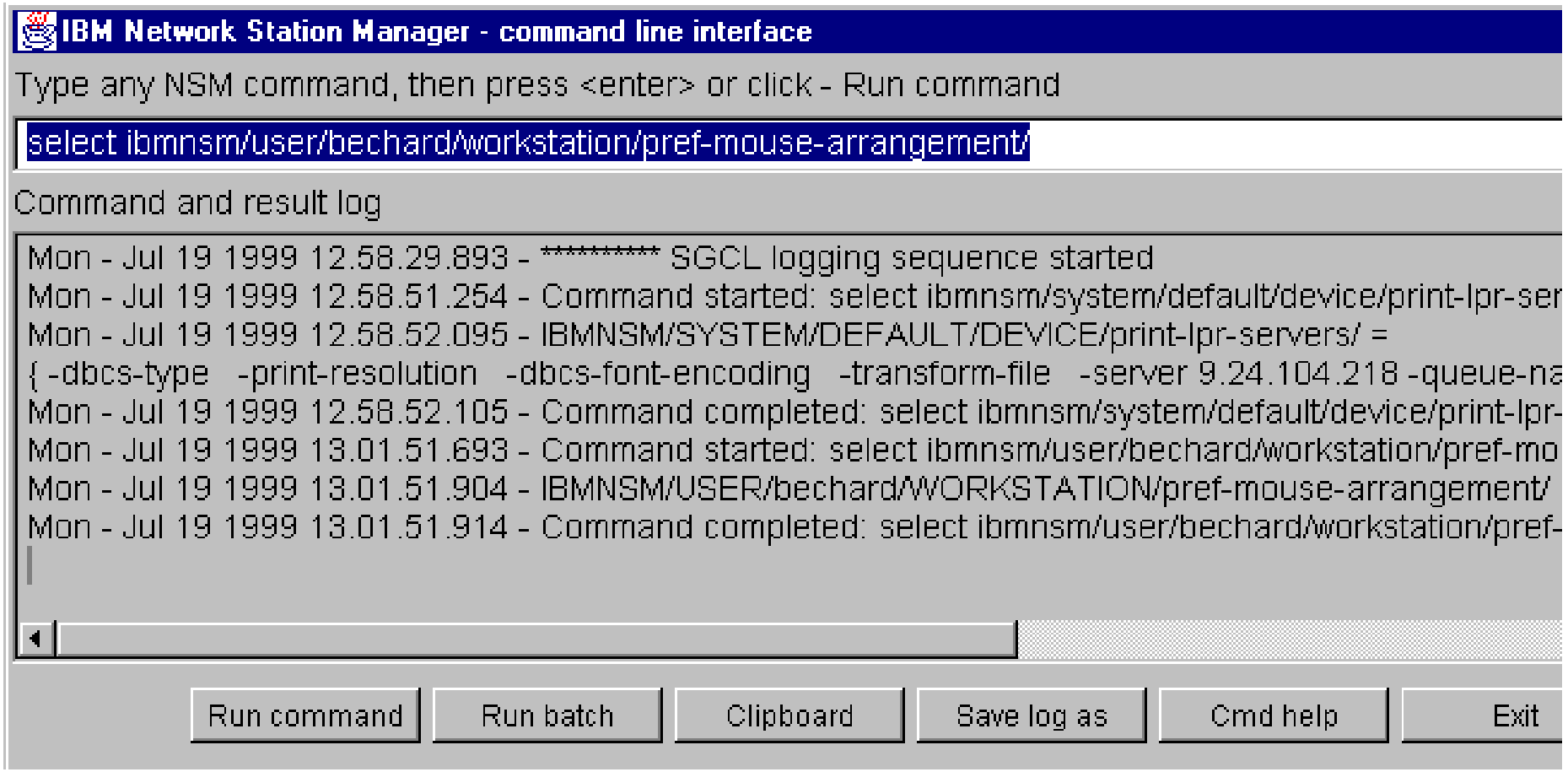
The NSMCL components reside in the servbase/tools directory, as illustrated on this chart.

The jar files contain the Java classes required by the NSMCL tool.

The SGCL.ini file contains the configuration settings required by the NSMCL tool.

The SGCL_log.txt file is the operational log resulting from running SGCL commands.

NSMCL GUI Interface Panel



Notes

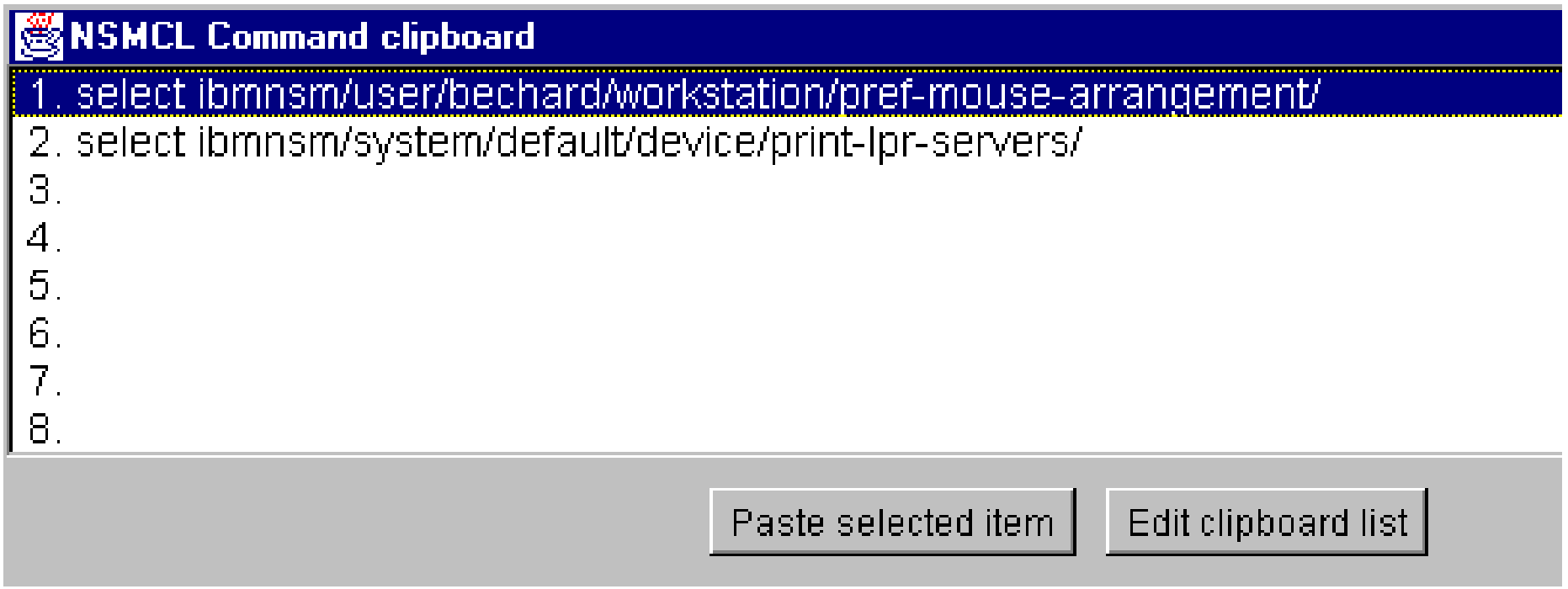
This chart shows the NSMCL graphical interface that allows individual commands to be entered and executed.

Commands are entered on the top line and the results of executing the commands are displayed in the lower window. This is a great advantage when testing because the results can be monitored immediately instead of having to display the log file.

Commands are saved to the clipboard and can therefore be easily recalled for execution or editing.

Notice the HELP button as well that can be used to look up the syntax of the available commands.

NSMCL Clipboard



Notes

This chart illustrates the clipboard where previous NSMCL commands are stored.

Commands can be selected from the clipboard and pasted back into the NSMCL command line.

NSMCL Command Syntax

command ibmnsm/level/name/category/configname/ configvalue

system
workstation
usergroup
user

Update	Change existing settings
Insert	Creates new or change settings
Delete	Deletes existing settings
Select	Returns existing settings
Copy	Copies settings
Call	Runs script files
Commit	Writes all pending (since last commit) changes to disk
Rollback	Discards all pending (since last commit) changes
Set	Temporarily sets any value in the SGCL.INI file
Exec	Runs a native operating system command in the operating system

Notes

This chart summarizes the syntax of the most commands (but not all) for NSMCL.

The available commands are listed here, along with their description.

The first parameter after the command is always IBMSNM.
Then the level is one of four possible values, as shown here.

Then the name parameter can be a group name, or a user name or it can be default when it concerns a system value applicable to all.

Then the category is one of many, that are too numerous to list here but that can be found in the Advanced User Information publication on the web.
This is true as well of all the configname which is the list of all possible parameters.

The last parameter is the value of the setting.

NSMCL Usage Examples

```
INSERT IBMNSM/WORKSTATION/9.24.104.192/EXTERNAL/
ip-address-at-next-boot/ 9.24.105.6

COMMIT
```

```
INSERT IBMNSM/SYSTEM/DEFAULT/USERGROUP/claude/ Sales_Local

COMMIT
```

Notes

Here are a couple of examples just to demonstrate how this works.

The first example sets the parameter ip-address-at-next-boot to the value 9.24.105.6 for the specific workstation whose IP address is 9.24.104.192.

So in this case, the level that we are working at is the workstation level, the name of the workstation is 9.24.104.192, the category to which this particular parameter belongs to is called EXTERNAL and the parameter is called ip-address-at-next-boot.

When this command (or more than one command) is executed, the results are not actually written permanently into the configuration profile until the COMMIT command is issued. And before a commit is issued, it is still possible to backout of the previously issued commands.

The second example makes the user called claude part of the Sales_local group. In this case, notice that this particular command is actually done at the system level even though this is setting a group name for a user.

Instead of being issued individually, these commands can also be entered into a script file and the script file executed. For example, if there were 300 users to be made part of the sales_local group, you would simply create a file with 300 statements similar to this last example, one for each user, and run the script once, followed by a commit.

Where to Go for More Information

- **Main Web Site**
  - **www.ibm.com/nc**
- **Current Network Station Redbook**
  - **SG24-5844 Network Station Manager V2R1 Guide**
- **Previous Network Station Redbooks**
  - **SG24-5187 AS/400 - Techniques for Deployment in a WAN**
  - **SG24-5221 Windows NT - NSM Release 3**
  - **SG24-5212 Printing**
  - **SG24-2127 Windows NT/WinCenter**
  - **SG24-4954 S/390, SG24-2016 RS/6000, SG24-2153 AS/400**
- **Product Publications**
  - **SC41-0684 Installing NSM for AS/400**
  - **SC41-0685 Installing NSM for RS/6000**
  - **SC41-0688 Installing NSM for Windows NT**
  - **SC41-0690 Using NSM**
  - **IBM Network Station Advanced Information (On the Web Site)**

Notes

More detailed information can be found at these sites and publications, in particular in the SG24-5844 redbook and in the Advanced User Information document available on the Web.