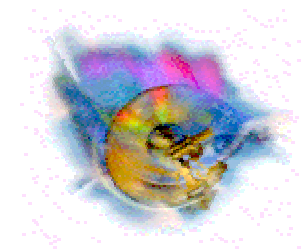


An Administrators guide for using Novell Application Launcher (NAL) 2.01



Prepared By: Kevin Prior
Novell Consulting

Table of Contents

Table of Contents	i
Copyright © 1997 by Novell, Inc.	iii
Introduction	Page 1 of 67
Installing NAL 2.0	Page 1 of 67
What Version do I have?	Page 1 of 67
What is required?	Page 2 of 67
Network Requirements	Page 2 of 67
Client Requirements	Page 2 of 67
snAppShot Workstation	Page 2 of 67
Installing snAppShot	Page 3 of 67
Using snAppShot to create a template	Page 5 of 67
Step 1	Page 4 of 67
Step 2	Page 5 of 67
Step 3	Page 6 of 67
Step 4	Page 7 of 67
Step 5	Page 8 of 67
Step 6	Page 9 of 67
Step 7	Page 10 of 67
Step 8	Page 11 of 67
Step 9	Page 12 of 67
Step 10	Page 13 of 67
Creating an Application from a Template (*.AOT)	Page 14 of 67
Application Object Properties	Page 21 of 67
Identification	Page 22 of 67
System Requirements	Page 26 of 67
Environment	Page 28 of 67
Description	Page 30 of 67
Drives/Ports	Page 32 of 67
Scripts	Page 34 of 67
Fault Tolerance	Page 36 of 67
Contacts	Page 38 of 67
Associations	Page 39 of 67
Administrator Notes	Page 40 of 67
Macros	Page 41 of 67
Registry Settings	Page 45 of 67
INI Files	Page 46 of 67
Application Files	Page 47 of 67
Text Files	Page 48 of 67
Schedule	Page 49 of 67
Icons/Shortcuts	Page 52 of 67
NAL License Metering	Page 54 of 67
NAL Helpful TidBits	Page 55 of 67
Container Login Script for NAL	Page 57 of 67
Application placement for NAL Explorer	Page 58 of 67
Inheritance of an Application	Page 58 of 67



Error Codes	Page 61 of 67
0xD000	Page 61 of 67
0xD001	Page 61 of 67
0xD002	Page 61 of 67
0xD003	Page 61 of 67
0xD004	Page 61 of 67
0xD005	Page 62 of 67
0xD006	Page 62 of 67
0xD007	Page 62 of 67
0xD008	Page 63 of 67
0xD009	Page 63 of 67
0xD00A	Page 63 of 67
0xD010	Page 64 of 67
0xD011	Page 64 of 67
0xD012	Page 64 of 67
0xD013	Page 65 of 67
0xD014	Page 65 of 67
0xD015	Page 65 of 67
0xD016	Page 65 of 67
0xD017	Page 66 of 67
0xD018	Page 66 of 67
0xD019	Page 66 of 67
0xD01A	Page 66 of 67
0xD01C	Page 67 of 67



Copyright © 1997 by Novell, Inc.

All rights reserved. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose without the express written permission of Novell

All product names mentioned are trademarks of their respective companies or distributors.

NOVELL CONSULTING



Revision 1.2

©1998 Novell, Inc.

Novell.

Edited: February 15, 1998

v



Introduction

This is a document on the Novell Application Launcher that covers the creation of the template and creating an Application Object based on the template to use with NAL. The components are snAppShot and the Application Object properties that are in NDS for an Application Object. Since the release of NAL 2.0 the files have been publically available at no charge to our customers. The files are available from Novell in a few fashions. The easiest and most accurate means is via the web. When NAL is on a CD and the NAL files have been updated the CD is not up to date. The URL for this is

<http://www.novell.com/products/managewise/product/nal.html>

You might also visit the support web site to see if there may be any patches as well That URL is <http://support.novell.com/>

You may also refer to the online help as well as the January 98 AppNote on “A Practical Guide to Using Novell Application Launcher”. The AppNotes are available online at <http://developer.novell.com/research/appnotes/> There is also a **GREAT** web site that is devoted to nothing but NAL! Take a look at <http://nal.netpub.com/>. There is an abundance of information here on NAL today and what is in the future. If you have a question be sure to look at the “Frequently asked questions” and if the answer is not there be sure to ask them via E-mail and look for your answer posted there soon.

Installing NAL 2.0

Installing NAL requires Admin rights to the [Root] of the tree as the schema is updated when installing NAL. The install is as easy as doble clicking the SETUPNAL.EXE (or running from Run in the Start Menu). Again full file rights are required to update the files in the SYS:PUBLIC directory and sub directories below. Be sure that this is installed to each server as there are files that are updated in the SYS:PUBLIC directory and subdirectories. If you are using NWADMIN on a server that has not been updated the objects will not appear properly. Be sure to verify that all servers are upgraded.

What Version do I have?

Since the release of NAL 2.0 there has been an update to NAL 2.01. The way to know your installing or have installed version 2.01 is the file size.

Version 2.01 File Sizes:

[setupnal.exe 5,690,954 Bytes] 5,558 KB

[setupsnp.exe 1,562,365 Bytes] 1,526 KB

If the files you have do not match these file sizes you want to obtain them from the Novell web pages as mentioned earlier in this document.

What is required?

Network Requirements

To use NAL you must have the following:

- A Novell NetWare 4.10 or later server that has NDS installed
 - Application server(s), this can be a NetWare or NT server. Users must have access to the server(s) as this is where the files may be stored
- To use applications on an NT Server the client must be running Client 32 and Microsoft Networks Client. At this time applications stored on 3.X servers are not supported from NAL 2.0.*

Client Requirements

To use NAL at the client you must have the following:

- Workstations that have Windows 3.X, Windows 95, or Windows NT
 - The latest Client 32 software for the desktop OS or the latest VLM client software version 1.2.1
 - A Directory Services connection to the network.
- For Windows 95 Clients the Service Pack 1 is required if NAL Explorer will be used.*

snAppShot Workstation

This is the machine in which applications will be installed to create AOT files. This should have only the following pieces:

- Windows 95 or Windows NT
- Latest version of Client 32
- snAppShot utility installed locally
- Latest Service Pack for desktop OS (*you may want to create an Application Object to deploy the Service Pack after you become more familiar with the snAppShot utility*)

Novell Consulting recommends that once this clean machine has these pieces installed that it be imaged so you can quickly revert back to a clean machine to test the deployment of the application or to install a new application. There is a utility called "GHOST" from GhostSoft (<http://www.ghostsoft.com/>) that will provide this functionality.

You may also wish to create a new user in NDS called "InstallApp" or something to distinguish the snAppShot user. This user will need all rights (except supervisor) to the directory that the Application files will be stored. For example if all application files will be stored under FS1\SYS:NALAPPS then assign RWCMF to the



NALAPPS directory.

If NT Workstation is being used for the snAppShot Workstation you may want to create an NT Configuration Object and assign users to the Administrators group on the NT Workstation. Then make this user a member of that object so they will have administrator rights on the NT Workstation a all times. For more information on this refer to the AppNote I wrote in June 1997 titled "Implementing Novell's NT Workstation Manager". The URL is <http://developer.novell.com/research/appnotes/>

Installing snAppShot

Installing snAppShot is very easy. The install is as easy as double clicking the SETUPSNP.EXE (or running from Run in the Start Menu). This install will default to install to C:\snAppShot on your local drive. You may want to install on a clean machine that has only the following pieces installed:

1. Desktop OS, either Win95 (with Service Pack 1) or Windows NT (latest Service Pack recommended).
2. Client32 already installed and configured for network access (Windows NT will need to have Administrator access on that local workstation)

What the process is for snAppShot

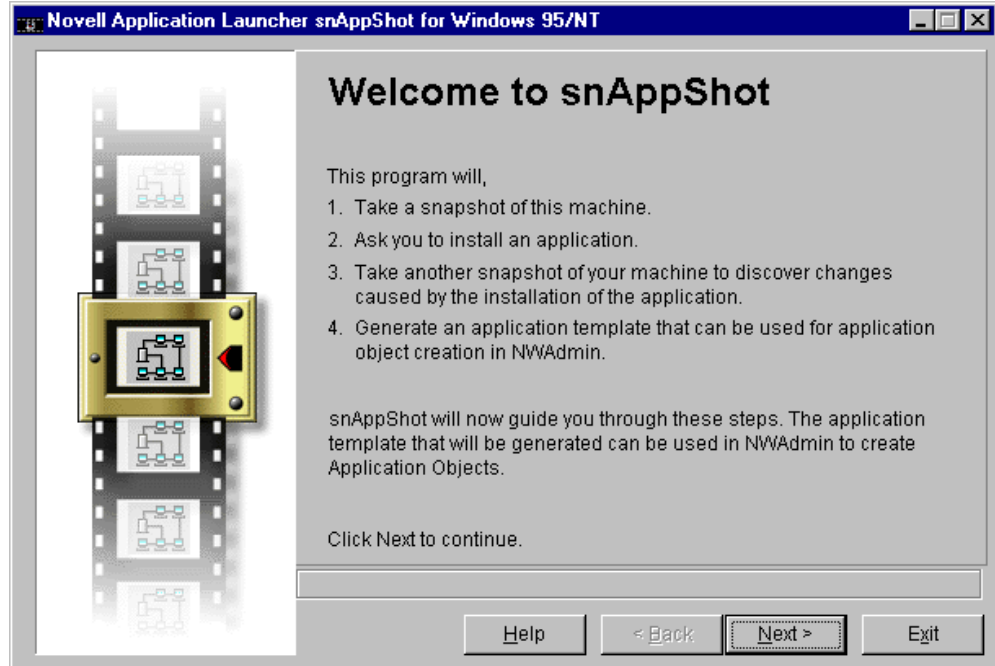
Novell Application Launcher can be used to distribute software to the client desktop. To gather this information Novell uses the "snAppShot" utility to gather what was installed to the "snAppShot Workstation". From a high level we create a clean vanilla desktop with the desktop OS, Service Packs, Client 32 and snAppShot utility. Next we run the snAppShot utility insert some information on the application we are going to install. Then we take a snAppShot before any software is installed. Next, install an application of your choice or make some desktop changes you will want to distribute. After the application is installed we take a second snAppShot and compare the changes. From the changes we will create an AOT file in which you can create an Application Object to distribute to 1 or all of your users. Once we have the template we can create multiple applications from this.

Using snAppShot to create a template

These next few pages will walk you through using snAppShot to create an AOT (Application Object Template) file. From this AOT file you will quickly and easily create application objects. For this paper I am using the shareware product WinZip 6.3 as the application I will distribute. I do have a licenses version for for demo purpose I am using the shareware version. The WinZip application can be obtained at <http://www.winzip.com/> Since WinZip is a shareware product you are required to follow the license agreement that accompanies the WinZip application. With this in mind lets get started with creating the template.



Step 1



Welcome to snAppShot

To start the snAppShot utility which in my case is installed to my local desktop. This is located under the Start Menu | Programs | NAL snAppShot. (C:\snapshot\snapshot.exe) When this is run we are presented with the above screen. To continue click on “Next”

This describes what the snAppShot application is about to do. This was already discussed and can be read in the screen shot above.



Step 2

Novell Application Launcher snAppShot for Windows 95/NT

Application Information

Specify the following information for the application you will install.

Enter a name for the application you will install:
WinZip 6-3 Install

Enter a short description for the application (this will be used as a caption for the icon):
Installation of WinZip 6-3 to local Windows Drive

Help < Back Next > Exit

Application Information

For the Application Information we must enter some descriptive information.

Enter a name for the application you are about to install:

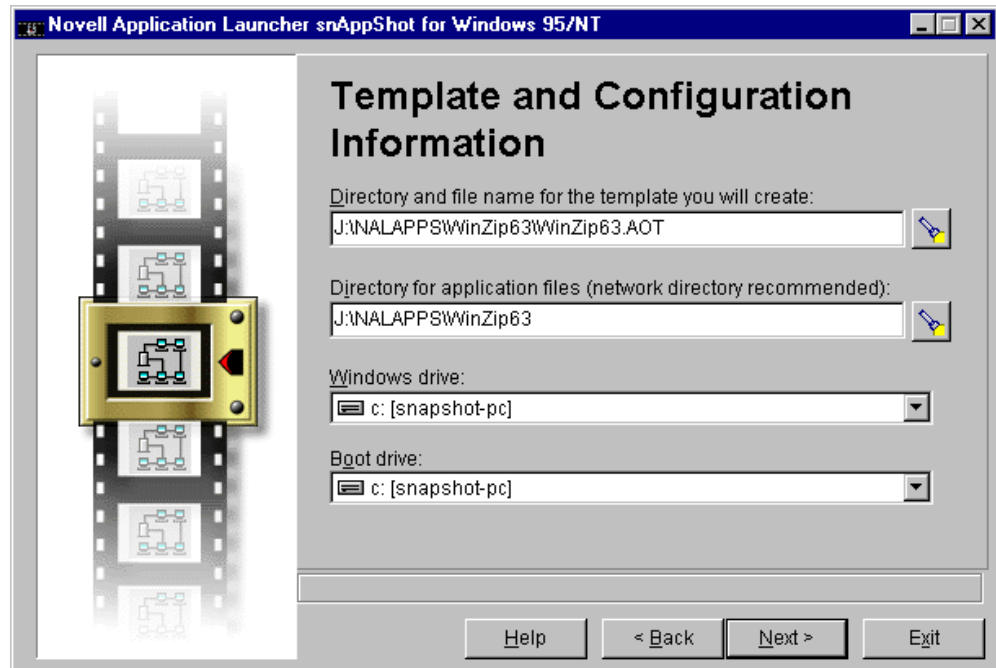
This is the name of the application that will be seen in NWADMIN for the application object when it is created.

Enter a short description for the application (This will be used as a caption for the icon):

The information input here is what the user will see from the NAL.EXE Window or from the NAL Explorer (NALEXPLD.EXE). The Application name and description may be different and will be stored in the AOT file. Each time an Application Object is created from the AOT file this information will be populated in the Application Object.



Step 3



Template and Configuration Information

This is very nice to have all Application Objects populated with this information upon creation.

Novell recommends that both the AOT file and application files (*.fil) be stored on the network. You may find it easier at times to store these files locally (C:\snapshot\appname). If this is the case be sure to copy files to network after snAppShot is complete. You will also need to update the macros of the SOURCE_PATH.

Directory and file name for the template you will create:

This will be the path that the AOT file will be placed. You may use the browse button to select the placement of this file.

Directory for the application files (network directory recommended)

This will be the path that the application files will be placed. You may use the browse button to select the placement of this file. Novell Consulting recommends that these files and the AOT file be stored in the same location. This will make it easier to locate and manage the files for each AOT file as well as if you wish to use Novell Replication Services to replicate the application files to other servers. File rights will also be easier to assign because users will need Read and File Scan to this directory to have NAL deploy the application files to the client workstation.



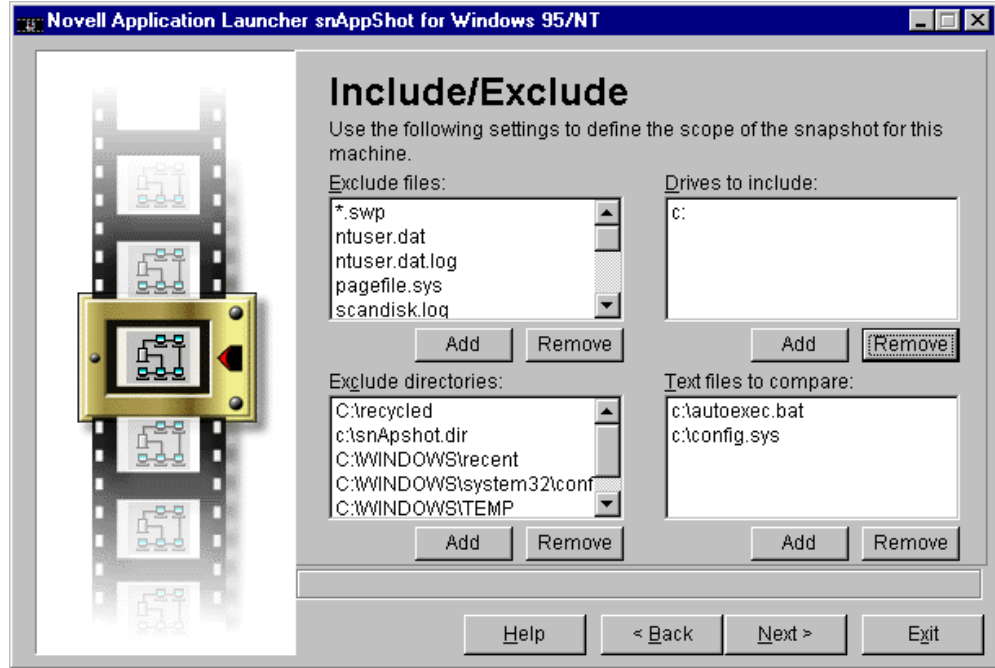
Windows drive:

This is the drive that your snAppShot workstation has Windows installed.

Boot drive:

This is the drive that your snAppShot workstation booted from.

Step 4



NOVELL CONSULTING

Include / Exclude

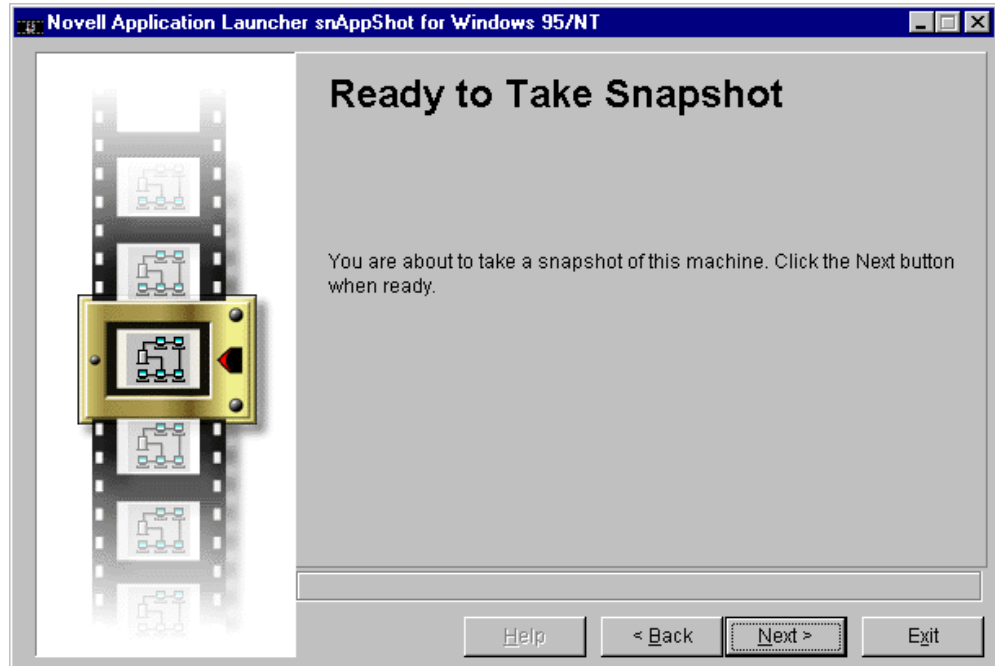
This window will allow us to configure what to include and exclude in the scan of snAppShot. Sometimes you may wish to limit the directories scanned to speed up the process. For example I know files will not be installed to C:\KPRIOR which contains about 200 Meg of files. I can select "Exclude Directories" C:\KPRIOR so this directory will not be scanned for any changes. This also means if any files do get installed to this directory they will not be part of the template.

If files will be installed to the network like the users home directory which is W: I can include this drive also to be scanned. Later in this document I will discuss using NDS variables where %HOME DIRECTORY% could be used.

Again we do recommend using a clean machine for using snAppShot which in this format the scans should be fairly quick.



Step 5



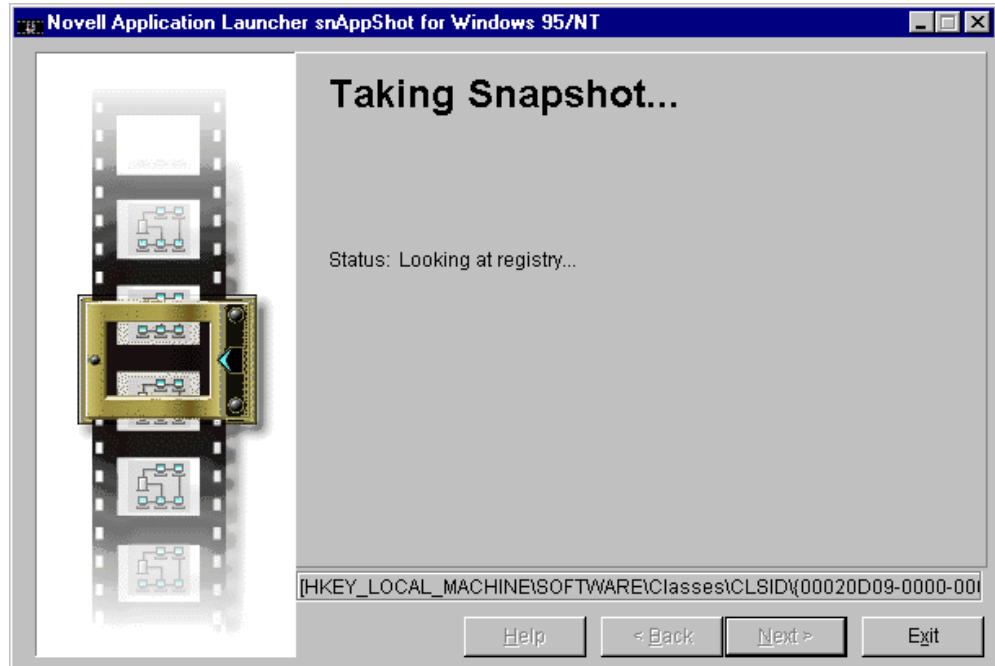
Ready to Take snAppShot

As you see we are ready to take the first snAppShot. By clicking the “Next” button we will start looking at the local machine. This includes what we selected in Step 4 so if a network drive mapping was included snAppShot will look at that drives as well (this is the directory that the drive is mapped and the user has rights to, it will not scan the entire volume if the user does not have rights).

In Step 4 we did our Include / Exclude of drives and files. A reason for having a clean machine to install an application to is to shorten the time of this process. The more files and registry entries we must scan the longer this will take as well as the final snAppShot.



Step 6

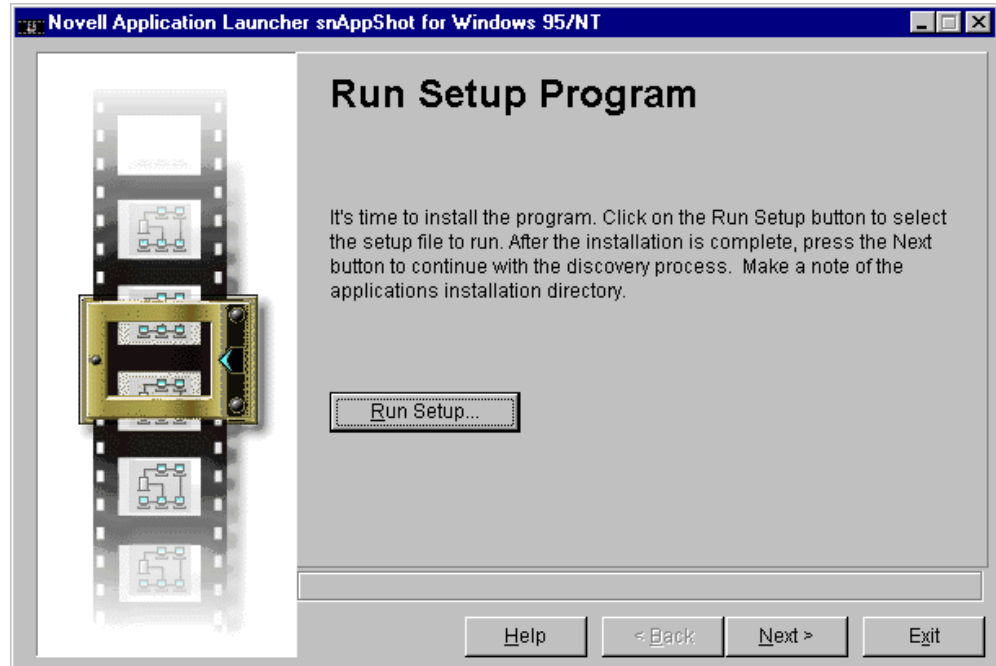


Taking snAppShot...

At this point here the first snAppShot process has started. The smaller the include drives and directories are the faster this step can be. Also be aware that 486/33 will be slower than Pentium 90. If you have faster hardware to use this would also be beneficial while creating the template.



Step 7



Run Setup Program

At this stage we are ready to install our application or make some changes to the snAppShot workstation. You may either select the “Run Setup” dialog box, use the Start | Run, or use Windows Explorer. This is your personal preference, any changes made from this point forward will be captured and made as part of the AOT file we are creating.

After the application has been completely installed you may be prompted to reboot for changes to take effect do you want to restart your computer? You may say “Yes” to restart your computer with no fear of losing this information. After the computer has restarted and you log into NDS again snAppShot will restart. You are prompted that a template is already been started do you want to continue with this or create a new template? Select the existing template and you will come back to this screen.

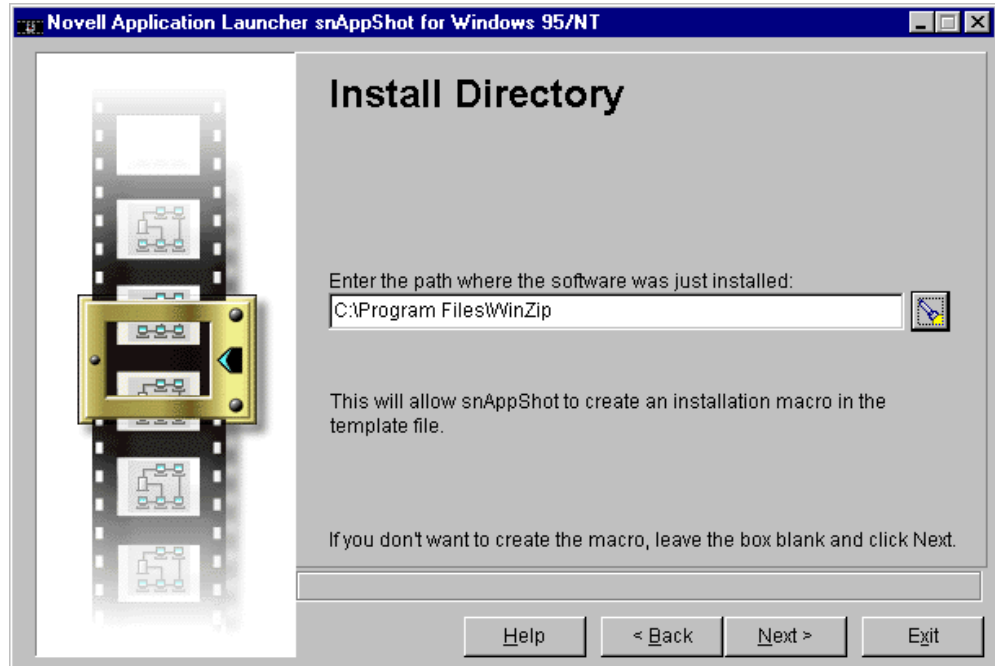
Make any changes necessary for example there was a shortcut placed on the desktop that you do not want. Simply delete the icon from the desktop and it will NOT be placed on other users desktop with NAL.

You may wish to leave these and delete from the Application Object Shortcuts and Icons property using NWADMIN. You will also get a good idea on the Macro syntax that is used in which you may decide to use manually at a later time.

When all changes have been made for this template click “Next”.



Step 8

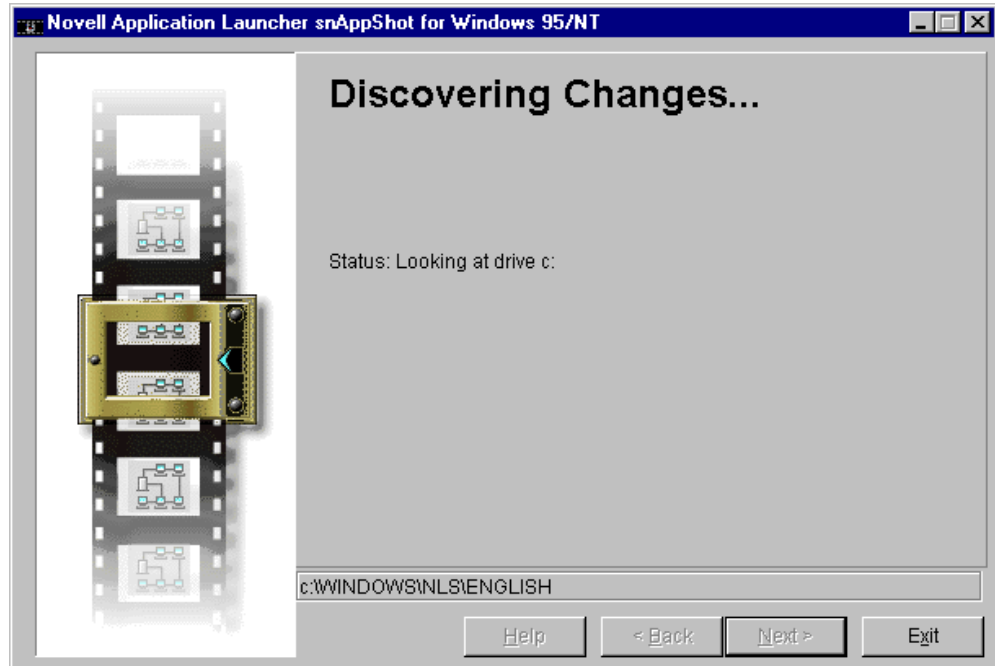


Install Directory

This is the directory in which the application was just installed to in step 7. This is normally where the setup.exe will default to installing an application to. For example my application was installed to C:\Program Files\WINZIP so this is what we put for the install directory. If we only typed in C:\ the “Target_Path” Macro would be C:\ as well. This is discussed more later in the properties page for “Macros”.



Step 9

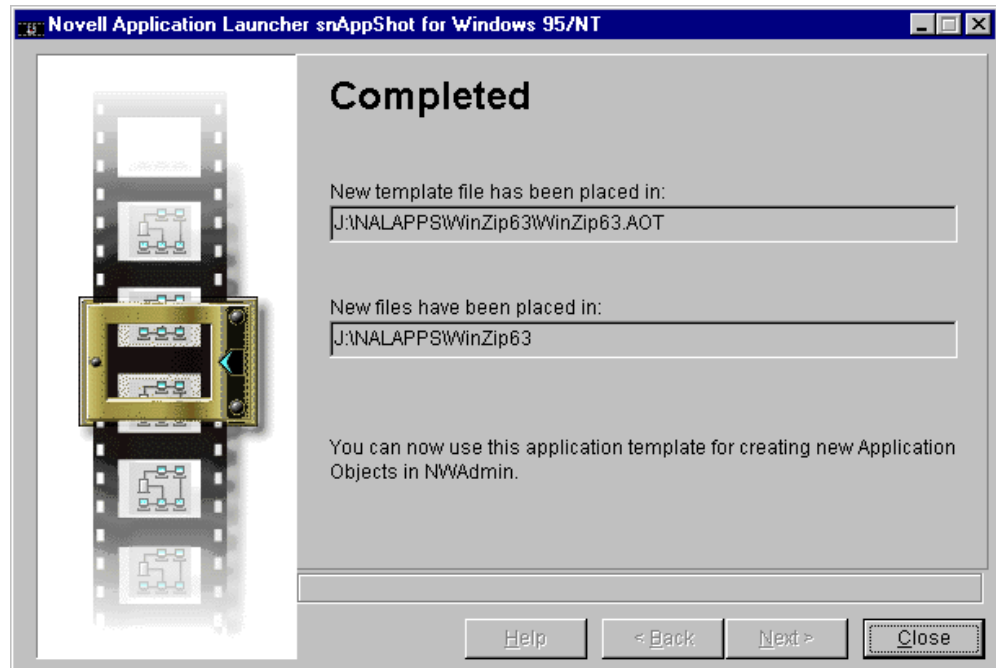


Discovering Changes...

At this point we are doing our second snAppShot. We will then compare these results to the first scan we ran. This will look at all drives we included as well as the registry on this machine. For large applications that many changes take place this may take quit a few minutes to complete. If you are able use a fast machine Pentium 233 with as much memory as possible. This step will take longer if slow hardware with minimal RAM is being used.



Step 10



Completed

The AOT file has been created and is information where the information is located. Also all files that were written to disk that were in Step 3 as part of the Include Drives: and where they were stored.

New template file has been placed in:

This is the location of where the AOT file has been placed. This should be the same as what you entered in Step 3 on page 6.

J:\NALAPPS\WinZip63\WinZip63.AOT

New files have been placed in:

This is the location of where the AOT file has been placed. This should be the same as what you entered in Step 3 on page 6.

J:\NALAPPS\WinZip63



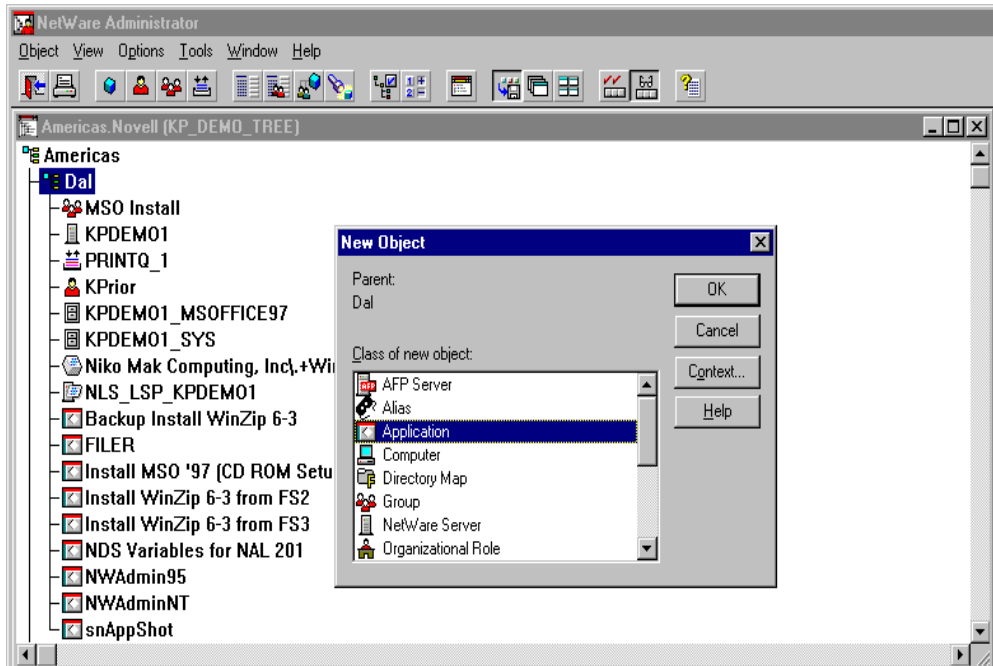
Creating an Application from a Template (*.AOT)

This will walk you through creating the Application Object from an AOT file. This example will use WinZip 6.3 that was created using SnAppShot. These steps are basically the same steps with every Application Object created.

Create the Application

From with NWADMIN we will go to the container where you wish to create the Application. From there right click the container and select create. Select the Application Object as shown here in Figure 12.

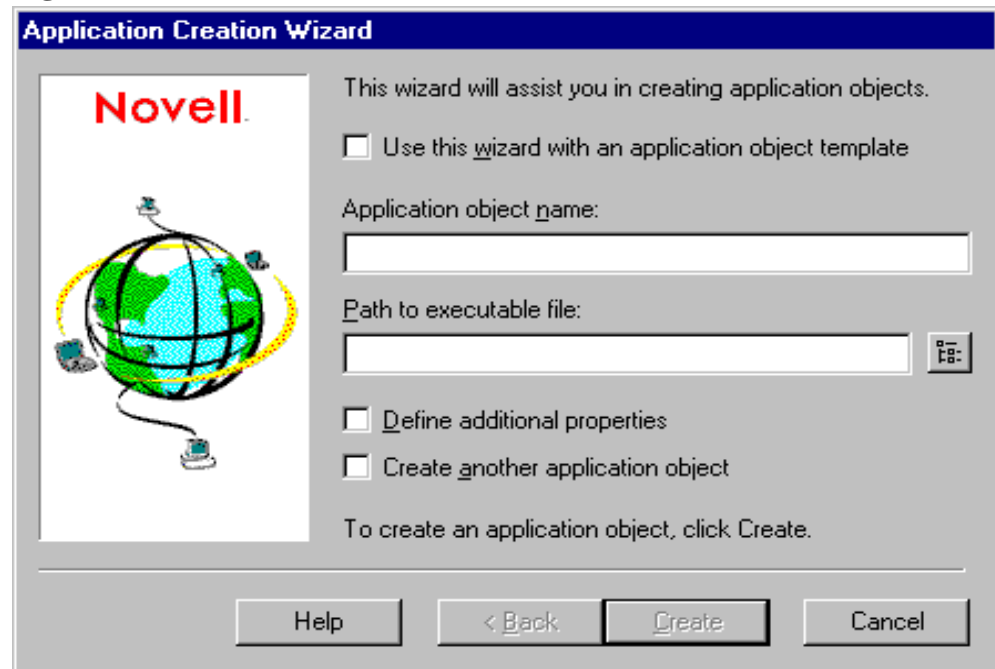
Figure 12



Next in Figure 13, we are presented with the Application Object creation Window. If we are just creating an Application Object and **NOT** using a template you would type in an “Application object name” as well as the “Path to the executable file”. A simple example would be NWADMIN.

If we typed in NWADMIN for WIN95 for the “Application object name” and either browsed for the location or type in the path to the executable. We can use UNC so for this example I would use “\\KPDEMO1\SYS\PUBLIC\WIN95\NWADMN95.EXE” for the “Path to the executable file”

Figure 13

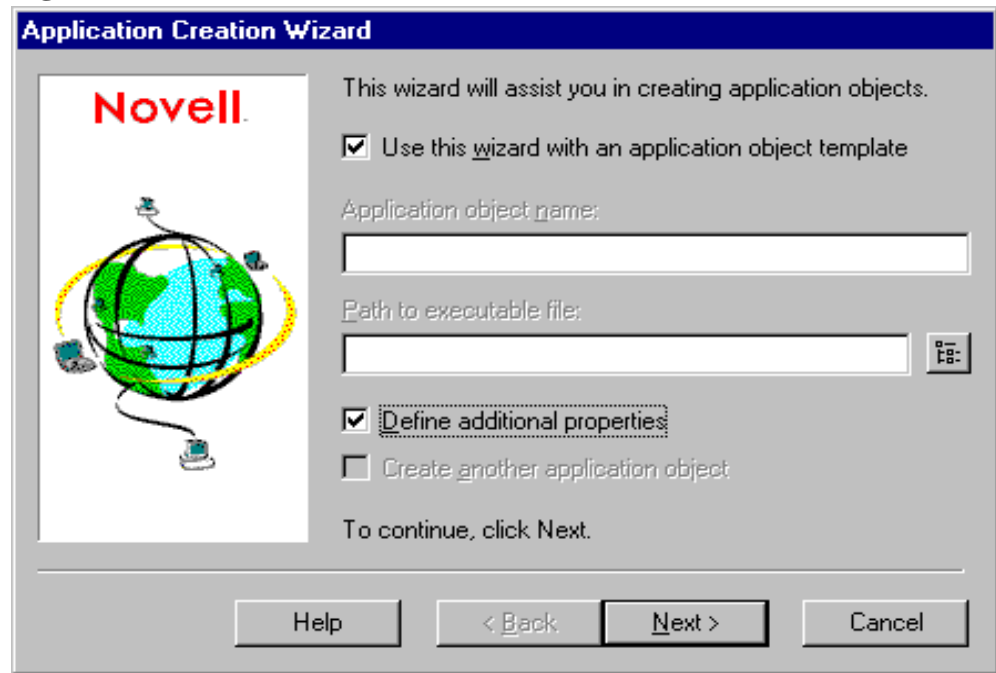


If we select the “Define additional properties” after the Application Object is created the properties for this application will be opened.



Since we are using the AOT file we will select “Use the wizard with an Application Object Template”. We will also select the “Define additional properties” as shown in figure 14, this will open the properties of the Application after we create the object in NDS. If this option is not selected after the application is created we will need to manually open the Application Object to edit the properties of the Application. To be sure users will be able to see the Application Icon they must be “Associated” to the Application Object. I normally always select this option to be sure I associate the Application Object to a User, Group or Container.

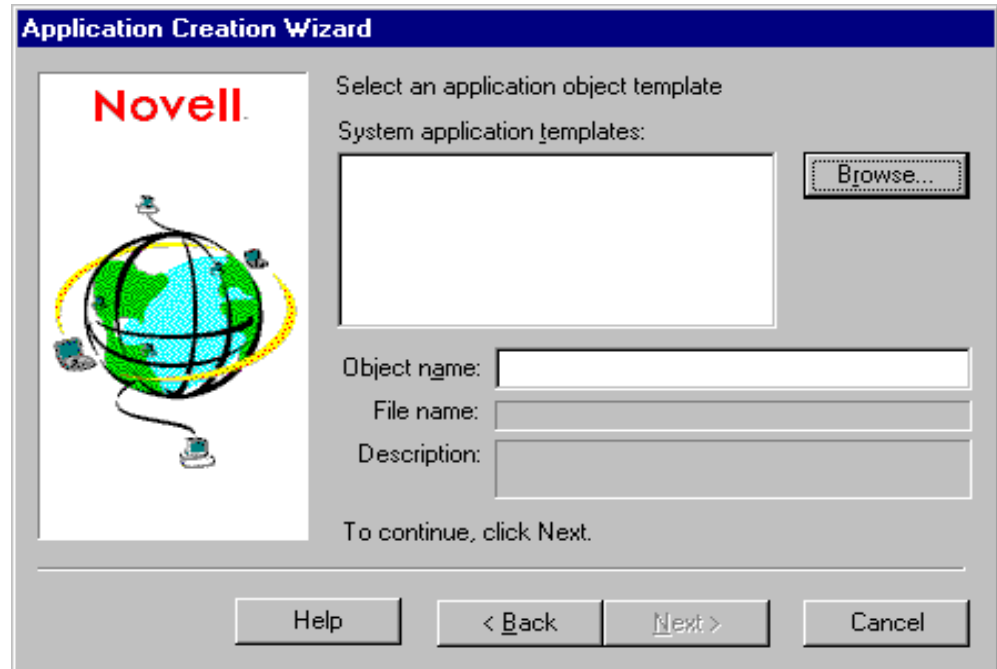
Figure 14



We will click “Next” and need to insert the path for the AOT file we wish to use.

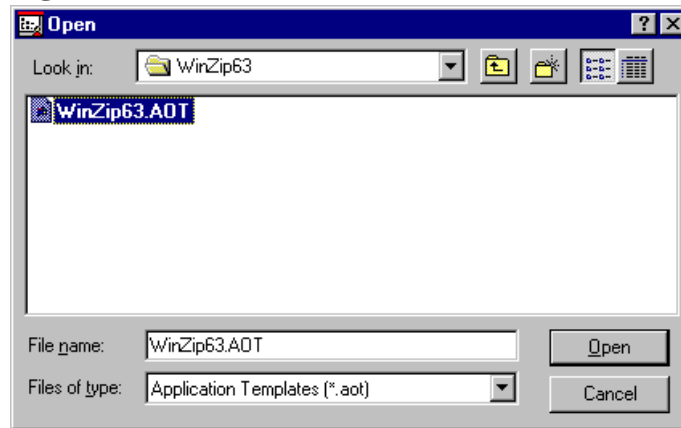


By selecting “Browse” as figure 15 displays, we can locate the template file.
Figure 15



The “Browse” Window in Figure 16 presents us a window that is asking for a path to where the AOT file is located that we want to use. We simply will browse and locate the AOT file and select this file.

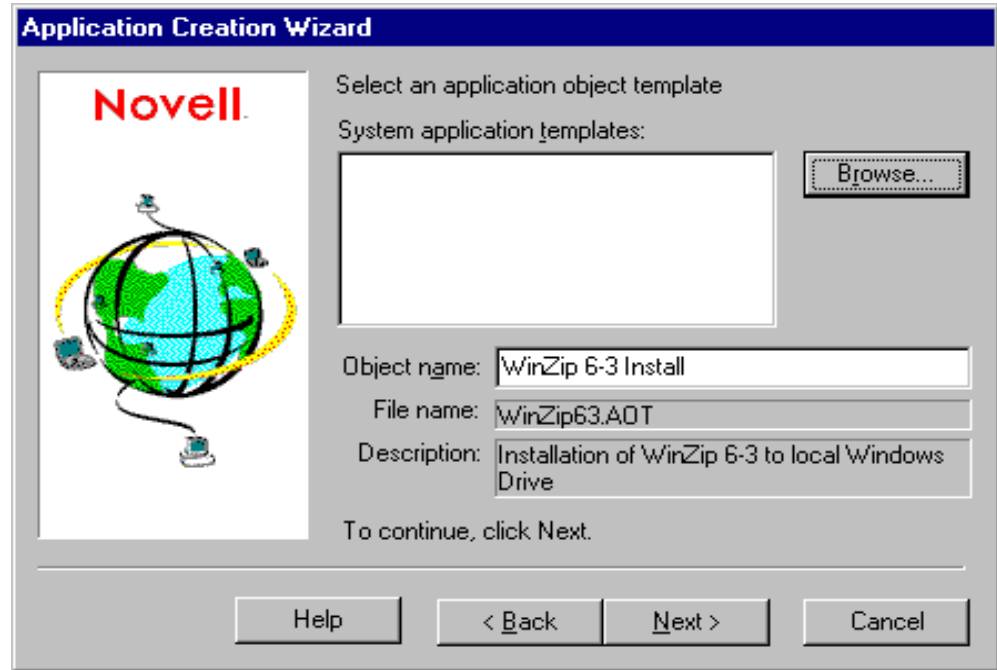
Figure 16



Once we select the AOT file we want and select “Open” the Wizard will display the information that was entered from snAppShot when the template was created that is shown in figure 17.



Figure 17

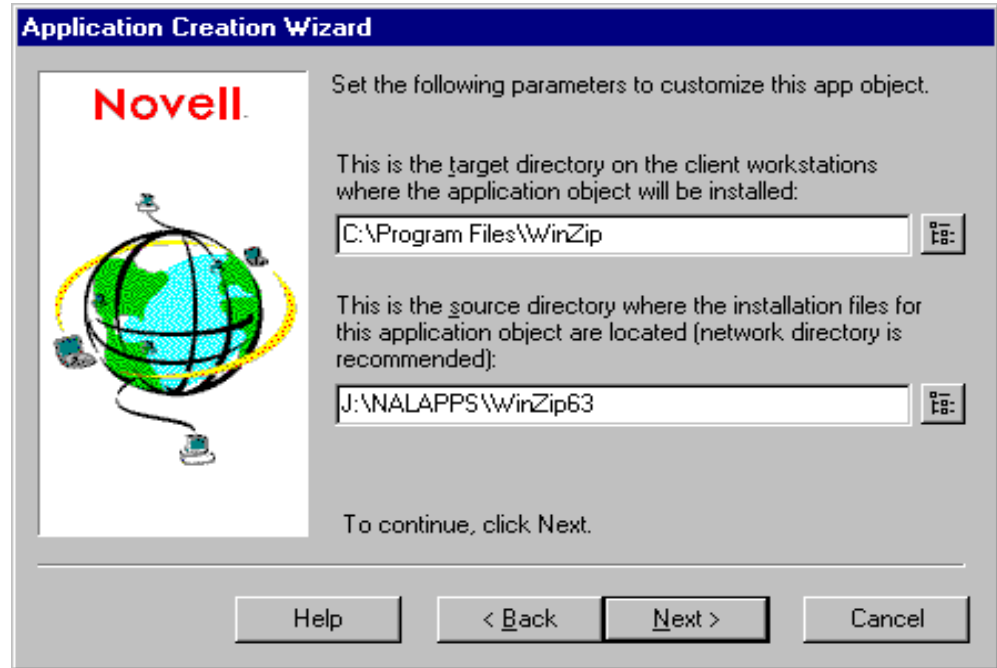


We now see that the information from the AOT file has populated the information needed for this window. The “Object name” is what is displayed in NWADMIN to name this Application Object. This can be referenced in Step 2 of the snAppShot utility on page 5. If we wish to change this Application Objects name we must do so here. Remember that we can use an AOT file as many times as needed so you can create many Application Objects from one template in the same or different containers provided each has a unique Object name. To continue we select next.

With this in mind using Novell Replication Services you can create “Master” packages and have them replicated to other servers in your NDS Tree. This will save the problems of multiple installs of the same application at other sites in different ways. The help desk can reliably have them verify the install and be sure that the application will work properly every time they run this application.



Figure 18



This information again was gathered from the AOT file that can be referenced in Step 3 on page 6 and also Step 8 on page 11. Here is the Target Directory where the application will be installed as well as the Source Directory where the Application Object files are physically located. Either of the can be changed by using the browse button but may be left unchanged. The information here is what creates the “Source_Path” and “Target_Path” Macros when the Application Object is created. We can change this information later but realize this information was generated from the AOT file we created with snAppShot.

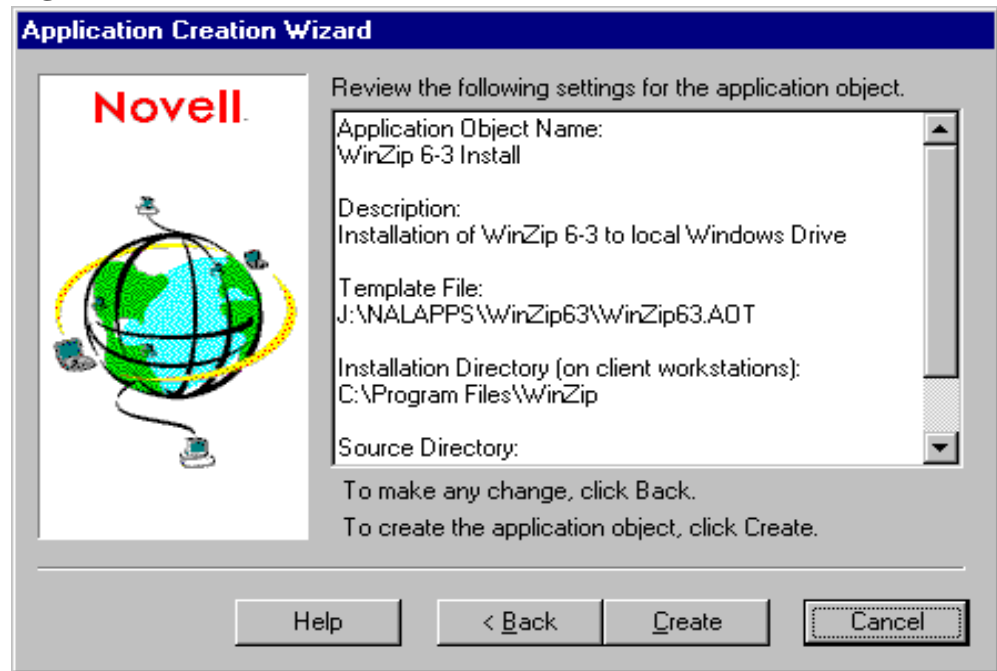
As an informational note you could browse for the location and the UNC path would be placed in the Source Directory. You cannot browse for the client directory for applications being installed to local drives at the workstation. This will have to be manually typed in.

To continue again we must click “Next”.

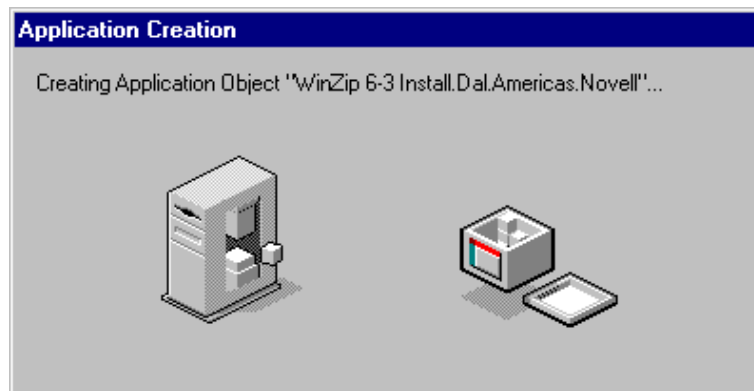
Now we are presented the next window in figure 19 to review the information we have input for the creation of this Application Object. This screen is simply to view and verify if changes need to be made. You may either go “Back” and edit the information or edit this after the Application Object is created. I personally feel that editing after the Application is created is easier.



Figure 19



If we now click “Create” the application will be created.



We can now go on to view and configure the Properties of this Application Object.



Application Object Properties

As with all NDS objects that an administrator can view they each have properties (also referred to as attributes) in which information can be input. When an application object is created it also has properties that an administrator may input information about the object for example the path of where the executable will be run from. The following pages detail each of these properties and the information associated to each property. There are some examples and possible ways of using each property page effectively. You can also refer to the online HELP that installed with NAL 2.0, some information in this document was obtained from the online help.

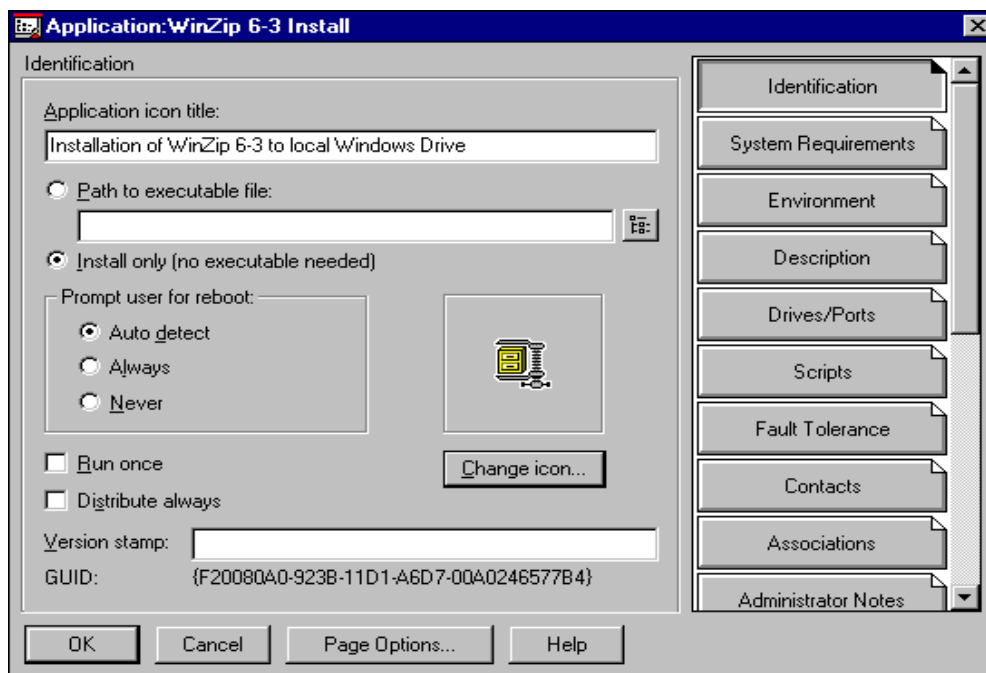
Often in this document the term NAL Window will be used. This can be either from the NAL.EXE or from NALEXPLD.EXE and this term is used interactively between the two. Unless it is directly mentioned the term NAL Window will reflect either of these 2 executables.

We will also later in this document discuss the differences between NAL.EXE and NALEXPLD.EXE and how to add this to your login scripts so when users login they always will launch NAL.

The following pages will go through the WinZip Application Object that I created. This is for demo purposes only and I do own a personal licensed copy of this application. All applications that you create will follow the same steps that I am documenting in this paper. Some items will need to be edited to work in your environment. Novell does recommend that you always test these in a test environment before rolling out to your production environment.



Identification



Application icon title

Text appears as the caption beneath the Application Icon in the NAL or NAL Explorer Windows. The default name is applied which is the same name from when you created the Application object Template (*.AOT File).

The application icon title is mandatory although this can be different than the Application object name that is displayed in NWADMIN that uniquely identifies the application object and might contain periods and other special characters. If a longer name must be used the Description property page for longer descriptions can be used to describe the application further which is discussed here shortly.

Path to executable file

This will point to the path where an application is installed (and will be launched) on a NetWare 4 server (or local drive). When a user double-clicks the Icon this is where the application will be launched. This can be edited if you later move the application to another directory. You may wish to point to a Directory Map Object so this will not need to be changed after updating the Directory Map Object.

You can type in the path as a UNC path or a file directory path. Use the Browse button to browse the file directory structure to find the executable you want. If the file resides on the local drive this must be available on all machines that user(s) will use or an error will occur and the application will not launch.

The following paths are example of how you might type the information in:



SERVER\VOLUME:PATH

KPDEMO1\SYS:NALAPPS\WINZIP63\WINZIP32.EXE

\\SERVER\VOLUME\PATH

\\KPDEMO1\SYS\NALAPPS\WINZIP63\WINZIP32.EXE

VOLUME_OBJECT_NAME:PATH

KPDEMO_SYS:NALAPPS\WINZIP63\WINZIP32.EXE

DIRECTORY_MAP_OBJECT_NAME:PATH

NALAPPS:WINZIP63\WINZIP32.EXE

DRIVELETTER:\PATH

R:WINZIP63\WINZIP32.EXE

Doing a “Push” of an application

Install only (no executable needed)

To “Push” or install files to a NAL user you do not want to select “Path to executable file” but “Install only (no executable needed). At this point any properties selected will be installed to the appropriate NAL users. This is very nice for the purpose of just updating some files or the registry on the workstation. Take for example that you have NetScape Navigator deployed and the proxy address will be changing. If you either use snAppShot to make changes on one machine and create an AOT file then create the Application Object “Update Proxy for NetScape” you can then deploy the application (or proxy changes) to all users without having to physically visit the workstation. If you use the Install Only option you do not specify a path in this section.

Prompt user for reboot

This option will allow you to choose whether a machine should reboot without user intervention after the application has installed to that machine. This is very nice to notify a user to reboot before the user uses the application so the required changes are put into effect to use the application. If the application does not require a reboot after the installation the “Never” or “Auto detect” are also available options. This option can best be implemented when using snAppShot. If the Application asked you to reboot when the install was completed you will want to select “Always” to reboot the client after the packaged application has been installed.

If you select Auto Detect (the default setting), and if the NAL Window or NAL Explorer needs to make changes that cannot occur while Windows is running (such as replacing open DLLs), it prompts for a reboot.

The Always option prompts the user to reboot every time a distribution takes place.

The Never option does not prompt the user to reboot. In this case, the changes take effect the next time the workstation is rebooted.



**Run once**

This option will allow the application icon to be seen and launched by the user in the NAL window one time. If the application is an install or "Push" having many icons in the NAL Window can be confusing. Use the Run Once option if you want the application to run just once (mainly to install an application or files) and then have its icon disappear so it is not visible in the NAL window. In this fashion after the application is installed there is one less icon in the NAL Window. This can also be very useful to avoid having 1 user installing an application on other workstations in that department when only 1 license was purchased.

There are a variety of ways to use Run Once in combination with other object properties. For example, if the application is specified as a Force Run on the Applications property page (meaning that the user is not involved, the application runs when Windows starts), the application is forced to run one time.

If you selected Run Once and also specified a Version Stamp for this application, the application runs once until the next time you change the Version Stamp, whereupon the application runs once one more time. This method is useful for upgrading applications.

Distribute always

When checked will "always" distribute the application package to the workstation. This may be good for small applications or registry changes but not a good idea for an application suite that may be 150 MB. This option will ensure that certain Registry settings are present, or that settings are updated every time a user runs an application.

This is an idea for NWADMIN where for each OS you add the Snap In information to the Application Object so the registry will always be updated with the appropriate Snap Ins.

Version Stamp

This allows you to setup a version for the application. This will Place a "stamp" on the workstation with a string. This string has nothing to do with the version of the product (although you could type this in here) but a version of the distribution package. This information gets stamped in the Registry (HKEY_CURRENT_USER\SOFTWARE\NETWARE\NAL)

The Version Stamp option is simply a text string representing the version of the application. The Version Stamp might or might not have anything to do with the actual version of the software; it is simply a tool to help you upgrade applications.

For example, suppose you created an application and gave it a version stamp of "1.0." You distribute this application and your users run it. Then you realize that you need to change several settings to the application, either as a result of importing an Application Object Template (.AOT) file or manually changing settings on the Registry, .INI Files, Macro, Text Files or Icons/Shortcut property pages. By

changing the Version Stamp text to something such as "1.1," the changes go into effect the next time the application is run.

If the Run Once option is checked and you change the Version Stamp, the Run Once option causes the application to run again once. This is useful when upgrading application software to a new or different version. For example, suppose you purchased new application software and want to update an Application object. By changing the Version Stamp number and selecting the Run Once option, the application runs once after installation even though a previous version might have already run once.

GUID

The Globally Unique Identification is how the Application Launcher tracks application objects on a workstation without the name being unique. For example if NAL did not do this you could install WordPerfect, remove WordPerfect and again install WordPerfect and it would not work. To view the GUID you can do a registry search for the GUID number, an example will look like:

```
[HKEY_CURRENT_USER\Software\NetWare\NAL\1.0\Distribute\KP_DEMO_TREE\E17A6AA0-8DDF-11D1-9184-00A0246577B4]
"LastRun"=dword:34c1117d
```

Change icon - You may change the Icon of an application by pointing to the icon file of your choice. This icon is what the user will see in the NAL Window, NAL Explorer, Start Menu, System Tray and the Desktop. You can also use the Description property page for longer descriptions of the application. To change the icon click the "Change icon" and select a file that contains the icon of your choice.



System Requirements

Application: WinZip 6-3 Install

System Requirements

Operating System

Windows 3.x Version: Major Minor

Windows 95 Version: Major Minor

Windows NT Version: Major Minor

Display applications on machines that have at least:

MB of RAM: 16 (95 and NT) Processor: 486

Free Disk Space

5 MB on Windows Directory Drive

MB on

MB on

OK Cancel Page Options... Help

Identification

System Requirements

Environment

Description

Drives/Ports

Scripts

Fault Tolerance

Contacts

Associations

Administrator Notes

The System Requirements Page acts as a “filter” to control what applications appear in the NAL Window based upon workstation-specific criteria that you setup here. You can filter based on “Operating System”, and Hardware requirements. If no Operating System is selected nobody will be able to see the application even though they are associated to the Application. In the example above this application requires Windows 95 or NT with a 486 Processor, 16 MB of RAM and 5 MB minimum of free disk space on the Windows Directory Drive. If all options are met they will be able to see this icon from the NAL Window. When all options are not met the user will not have this icon displayed in the NAL Window.

As you see we can get specific on which desktop OS can see this application. You can also take advantage of Major and Minor versions of the Operating System that is stored in the registry. To see your versions use regedit and search for “VersionNumber” (Win95)

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion]
"Version"="Windows 95"
"VersionNumber"="4.00.950"
"SubVersionNumber"="a"
```

and “Major Version” (WINNT).

This is very easy to test and try by associating an application to you and set the requirements higher than what you currently have. Another example is have an application for only Windows NT and use a Windows 95 machine. If you do



associate an application to a user and they cannot see the application in NAL, verify that you did select an operating system or it will not be available to anybody. Next verify the hardware settings meet the requirements that were setup.

Operating System

Select the Operating System this application will be available for. You may have an application that is Operating System specific (For Windows 95 only and not Windows 3.X or NT). Be sure an Operating System is selected or nobody will see this Application in the NAL Window.

Display application on machines that have at least:

When this is checked the options you have selected for hardware specific requirements are checked. If the workstation does not meet these minimum requirements the user will not see the Application Icon.

MB of RAM (95 and NT)

This will set a minimum requirement for RAM that is in the workstation that the user is logged in with.

Processor

For the processor you can choose from 386, 486, or Pentium. For example, we have selected 486 and the user has a 386 workstation, the Application object is not displayed on the workstation in the NAL Window

NOTE: The Windows 3.x Application Programming Interface does not return values higher than 486. Therefore, if NAL 2.0 software queries the processor type of a Windows 3.x workstation using a Pentium processor, Windows 3.x returns "486." In other words, even if you select the Pentium processor type for the Processor option, the application is displayed on 486 workstations running Windows 3.x.

Free Disk Space

You can check three different drives for available disk space. You can be specific for Windows System Directory Drive, Windows Directory Drive and Temp Directory Drive. The available drive letters are A - Z. For example, you can check the workstation for 2 MB free on the Windows Directory drive, 10 MB free on the TEMP drive, and 80 MB free on the D: drive. As with the other settings, if you leave these text boxes blank, no checking is done for available disk space.

NOTE: If the drives selected for any of the three fields happen to be the same drive, NAL adds those minimum requirements together to determine available space. For example, if you specify **20 MB** free on the Windows Directory drive (which happens to be the C: drive), **10 MB** free on the TEMP drive (which is also the C: drive), and 50 MB free on the D: drive, NAL only shows the Application object as if there were **30 MB** available on the C: drive, and **50 MB** free on the D: drive.

Environment

Revision 1.2

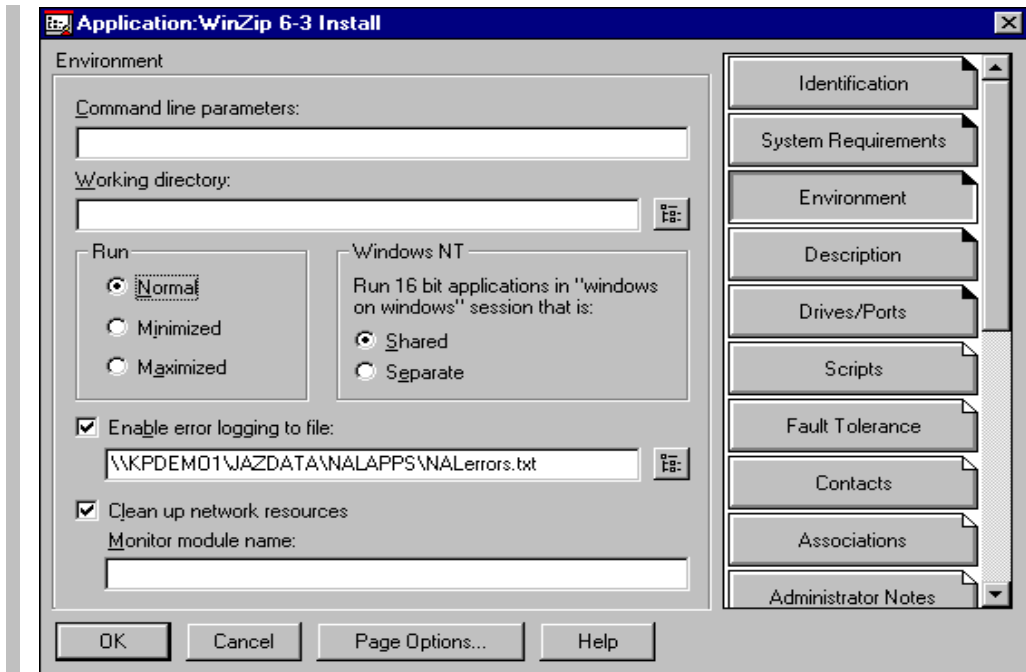
©1998 Novell, Inc.

Edited: February 15, 1998

Novell.

Page 27 of 67





Command line parameters

This may be used for an application that requires a special start-up switches for the application's executable. If NetScape were the application you might place the IP Address of your Intranet Department server. Although the "Default Home Page" is setup to Novell.Com when this application is started from NAL, NetScape is passed this information and will go to the Intranet web server.

Working directory

This may be used to provide the path to the application's working directory. This might be the default directory when users do a File, Save. Consult the application documentation for more information on the working directory needs. Not all applications will require this information and some applications will use the information from the registry instead of the information placed here.

Run

This will allow you to select how the initial Window size. By default Normal is always selected. You might want to select Maximize so when the application is started it is a maximized window to start.

Windows NT

This option will allow you to setup a 16-bit application to run on Windows NT. You may specify Shared or Separate Windows on Windows (WOW) sessions.

To run on Windows NT (a 32-bit operating system), 16-bit applications must run in a Virtual DOS Machine (VDM) that emulates Windows 3.1 functionality. The 16-bit Windows emulator called WOW (for Win16-on-Win32) isolates 16-bit application



errors from the rest of the 32-bit operating system.

16-bit Windows applications can run in their own separate WOW sessions or they can share a WOW session with other 16-bit applications. Sharing a WOW session saves memory and allows applications to use DDE or OLE to communicate with other applications in the same WOW session. Running an application in its own separate WOW session allows for more robust operation because the application is isolated from other 16-bit applications. Because some applications do not run properly in a separate WOW session, the default option is to run the 16-bit Windows application in a shared WOW session.

Enable error logging to file

Enter the path to a file where any errors will be logged if the application fails to install or launch. No other status is tracked here except for errors (this will be enhanced in the future). Users will need write file system rights to write to this file in order for it to work. This is easily setup from a Group object so all members will obtain the correct rights to the Error Log file as well as all application files. An example of the error log output is below.

```
-----  
01/21/98 07:31:50 pm  
KP_DEMO_TREE  
.USER3.Dal.Americas.Novell  
0101D9CE:00A0246577B4  
.Install WinZip 6-3.Dal.Americas.Novell  
>>Could not configure workstation for application [Install WinZip 6-3] (id=5).
```

```
Problem: Unable to process registry setting  
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows  
NT\CurrentVersion\File Manager\AddOns :: WinZip]. (D016)<<  
-----
```

Clean Up Network Resources

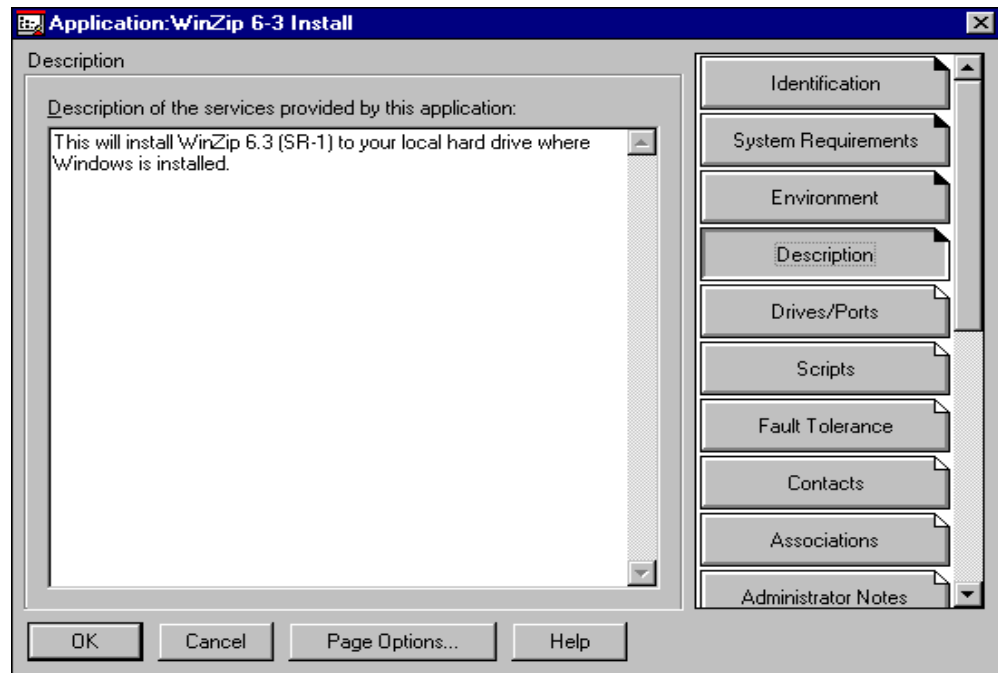
Upon exiting the application NAL will clean up resources allocated, such as drives mapped, ports captured, and licensed connections made to file systems for accessing the application. This is very useful from the point that a users licensed connection is removed from a server so unnecessary licenses are not being used when the user is not accessing the application. If a connection is already in place when an application is launched NAL will use this connection and not clean it up.

The Clean Up Network Resources option is enhanced in NAL 2.0 with a new “**Monitor module name**” option. Some applications, like WordPerfect, are actually wrappers. WPWIN.EXE runs WPWIN7.EXE and then terminates, while WPWIN7.EXE continues to execute. The problem is that NAL monitors WPWIN.EXE and quits before the application is really done. Monitor Module Name allows the administrator to type in the name of the real application that is going to run (WPWIN7.EXE in this case) and NAL can then monitor it for termination. The usual error message you see, indicating you need to use this feature, is “Application



Not Releasing Resources” when you exit the application.

Description



Description Page

This can give your users more detailed information on the application above what they see as the caption in the NAL Window. Users would right-click on the object in the NAL Window and select Properties to view this description.

Users can right-click an Application object in the NAL Window to see details containing both the descriptive name of the application and the more lengthy description that you provide using the Description property page.

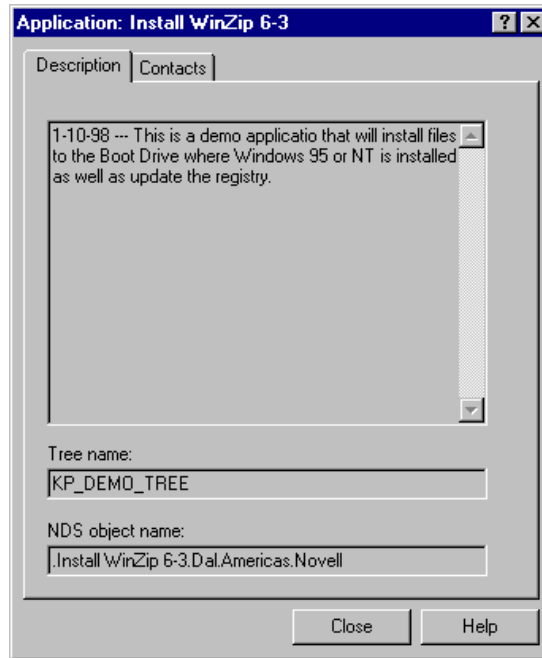
For this example, this application will install WinZip 6.3 to the C: drive and is a demo application for a proof of concept pilot. The description that is typed, which may longer than the icon's title can contain, might include a description and any special instructions you'd like your users to see.

To view this information the users will do the following:

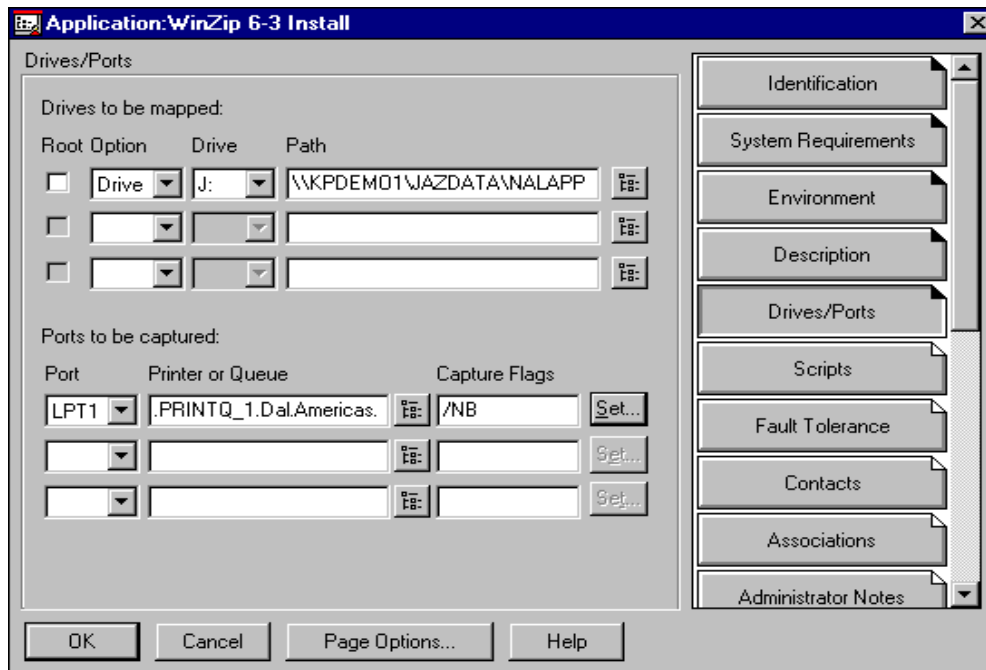


1. From the NAL Window, Right-click the Application object and select Properties
2. Read the Description .
3. When completed, click close.

An example is displayed here.



Drives/Ports



Drives/Ports

This option will allow you to map a network drive thus eliminating some of the long container scripts that you might have for drive mappings. This will eliminate unneeded drive mappings except for when they are required for an application. In this example we are also mapping a drive to the NALAPPS directory on drive J:. When NAL runs the application, it checks for a mapping of drive J:. If J: is mapped to the correct volume and server as in the application object, the drive is used. If J: is mapped to some other server or volume, it is not used, and the application does not run. If J: is not mapped at all, NAL maps it. Only in the case where NAL actually maps the drive does it clean up the mapping after the application terminates. If several applications use the same drive mapping, then drive mappings aren't cleaned up until the last application terminates.

Root

When this is enabled or checked, NAL will map root the drive. For example, if KPDEMO1\JAZDATA:NALAPPS\WINZIP63 is the path, the user or application cannot change to a directory that is higher than WINZIP63.

Option

Here you can select Drive, S1 or S16. This option is to select the type of drive to map. For example, select S1 to map the path as the first search drive. If the search drive does not already exist, it is inserted. Select S16 to map the path as the last search drive. Select Drive to map the drive you select using the Drive option.



Drive

Use this option to assign a regular NetWare drive. You may select next, and letters A: through Z: are available.

Path

You will insert the full path to map. You can also click the Browse button, and then search the NDS tree and select the directory you want to map. You can map to a server volume, volume object, directory map object and an existing mapped drive. Some example of the syntax for mapping a drive

SERVER\VOLUME:PATH

KPDEMO1\SYS:NALAPPS\WINZIP63\WINZIP32.EXE

\\SERVER\VOLUME\PATH

\\KPDEMO1\SYS\NALAPPS\WINZIP63\WINZIP32.EXE

VOLUME_OBJECT_NAME:PATH

KPDEMO_SYS:NALAPPS\WINZIP63\WINZIP32.EXE

DIRECTORY_MAP_OBJECT_NAME:PATH

NALAPPS:WINZIP63\WINZIP32.EXE

DRIVELETTER:\PATH

R:WINZIP63\WINZIP32.EXE

Port

Use this option to assign the workstations printer port to the Application object. Select from nine different ports. Your workstation operating system might need to be configured to recognize the port number if it is larger than 3.

Printer or Queue

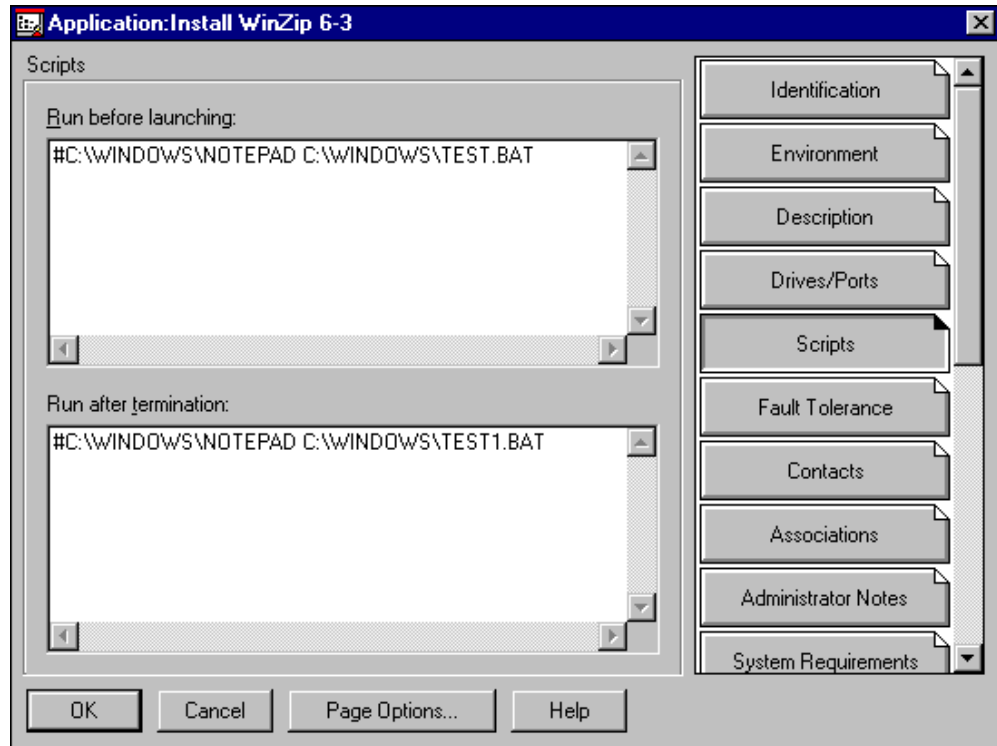
Click the Browse button and then select the NetWare print queue or printer that you want to print to from the application. Application objects can access any printers that are available and created in the NetWare environment provided the user has rights to that print queue.

Capture Flags

You can set the capture options to use when the application is launched by using the Set button. You have the option to select Notify, Banner, or Form Feed. These options will override the existing capture settings at the workstation.



Scripts



Both Pre-launch application and Post-launch application scripts are available. The format used here is the same as with IntranetWare login scripts. Any drive mappings or ports captured here are *not* cleaned up, even if Clean Up Network Resources is turned on. Also, drives mapped or ports captured here can overwrite existing ones vs. if they're set from the Drives/Ports page (discussed earlier).

In this example we are using the prescript to call notepad to display a message. Notice that we are calling the NOTEPAD.EXE from C:\WINDOWS. This must be available on the local drive in this exact path or an error will occur. Also the BAT file must be located in the location that is input. This may be helpful if an application is being updated on the server. You can create a text file stating the application is not available and will be available by 2:00 PM. The post script is in this example only for display purposes. If a drive was mapped in the pre script we would need to delete the drive in the post script.

Examples of what you can do with application scripts

Provide extra mappings beyond those defined on the Drive/Ports property page.

Provide a mapping to override another mapping.

Run other applications.

Log in to other servers or NDS trees.

Terminate applications under certain circumstances.



Example of two ways to run an application in a script

#calc.exe Run the Calculator application, pausing the script processing until Calculator returns control.

@calc.exe Run the Calculator application concurrently with the remainder of the script processing.

NAL scripting does not do the following:

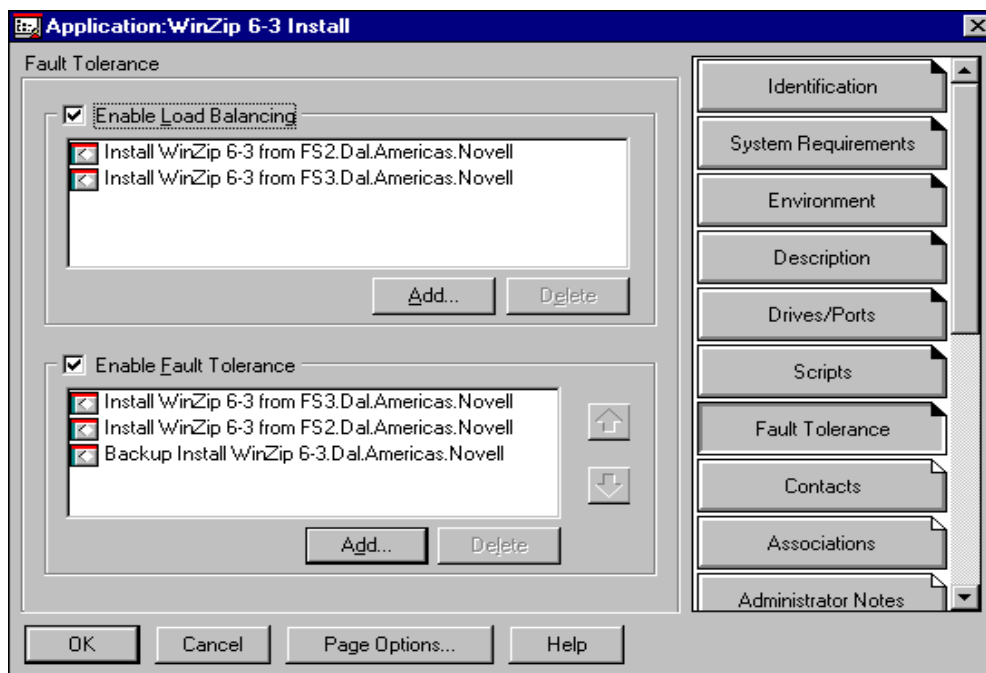
Output anything to the screen
Display errors
Pause

Scripting Commands That Do Not Work with NAL:

CLS
DISPLAY
FDISPLAY
INCLUDE
LASTLOGINTIME
MACHINE=
PAUSE
WRITE
EXIT
NO_DEFAULT
NOSWAP
PCCOMPATIBLE
SCRIPT_SERVER
SET_TIME
SWAP



Fault Tolerance



Enable Load Balancing

Load Balancing lets the NAL Window software to spread the user load across multiple Application objects. Ordinarily, you use load balancing when all the servers the Application objects use are located in a single site on a local area network. If a particular server goes down for example FS1, rendering that Application object inoperable, the Application object is removed from the list of available applications, the list is re-randomized and a server is re-selected.

You could enable this feature for a site which has several applications servers. NAL uses a random number to distribute the load (very simple and basic). NAL does not distribute the load based upon CPU utilization at this time.

Enable Fault Tolerance

Application objects are set up to be accessed from a specific server. If that server is inaccessible, you can set up alternative servers from which the application can be accessed. By doing this, you keep the applications running with no disruptions in availability to your users. You can use two methods for setting up alternative servers.

There are design considerations here. Your client cannot have its primary DS connection server be the Fault Tolerant server that "crashes". (Right-click on Network Neighborhood and select NetWare connections; there will be an "*" next to your primary DS connection server.) If this is the case, you will get error messages when you try to re-launch the application off another server in the Fault Tolerant list. So, when designing this feature, your applications servers in the Fault Tolerant list



need to be different from the server where you have your primary DS connection **if** you want fault tolerance to protect you from a server “crash”. It will still work in this scenario if something happens to the application but the server still remains up. This is actually a Client issue and not a NAL issue. Also, if you have more than two servers in the Fault Tolerant list, NAL selects the next server by going right down the list.

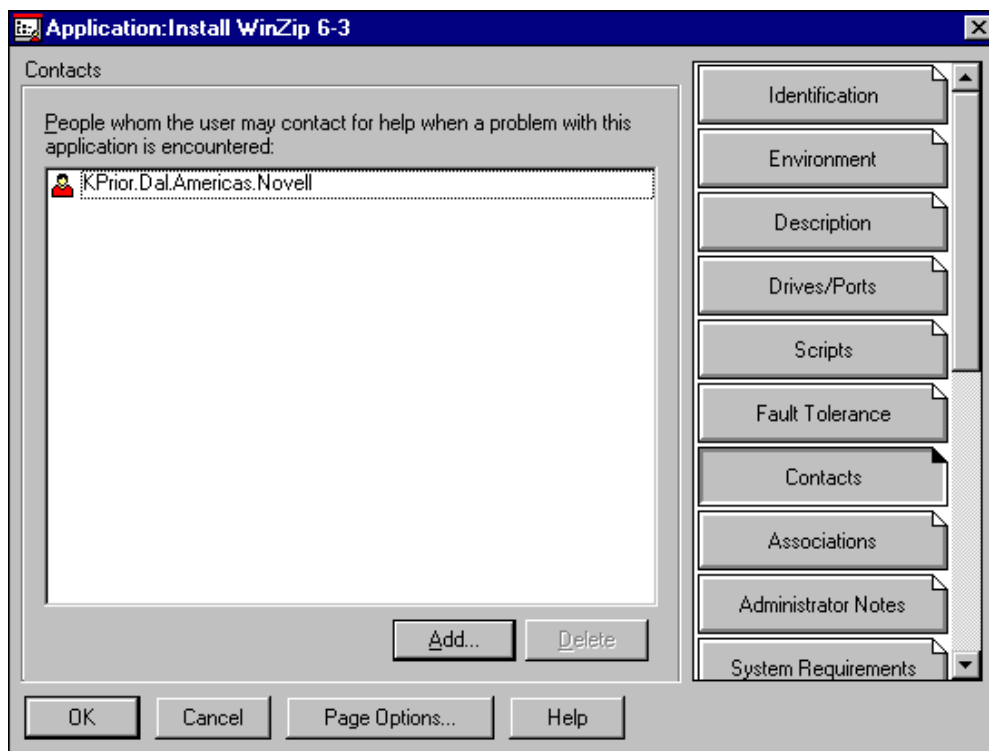
In this example the Fault Tolerance is enabled so that NAL will use the BACKUP Application on FS2. If FS1 were not available it would then go to the Application Object listed in the Fault Tolerance section.

A simple way to demo this is to create 2 Application Objects, we will use NOTEPAD1 and NOTEPAD2. Copy the NOTEPAD.EXE to the server, we will create a directory called NALDEMO. Below this create directories APP1 and APP2. Copy the NOTEPAD.EXE to the APP1 and APP2 directory so that the executable is located in each directory. Now the PATH to the NOTEPAD1 Application Object will point to the APP1 directory and NOTEPAD2 Application point to the APP2 directory.

Setup the NOTEPAD1 Application to use Fault Tolerance and insert NOTEPAD2 into this field. Associate the NOTEPAD1 to a User, Group or Container and run NAL. Launch the NOTEPAD1 Application and the Application opens. Now rename the NOTEPAD.EXE in the APP1 directory. This will simulate the Application no longer being available. NAL will then go to the NOTEPAD2 Application that is in the Fault Tolerance list.



Contacts



In the Contacts Page you would add the NDS name of the user(s) or group responsible to support this application. This information could provide the support group or Help Desk information. You may decide to create a HelpDesk user and provide the phone number to the help desk. Users could access this to see whom to contact if they are having a problem with their application. Again, users would right-click on the object in the NAL Window and select Properties to view the Contacts and their Phone Number/Email Address (assuming these properties are set on the Contact user's NDS object).



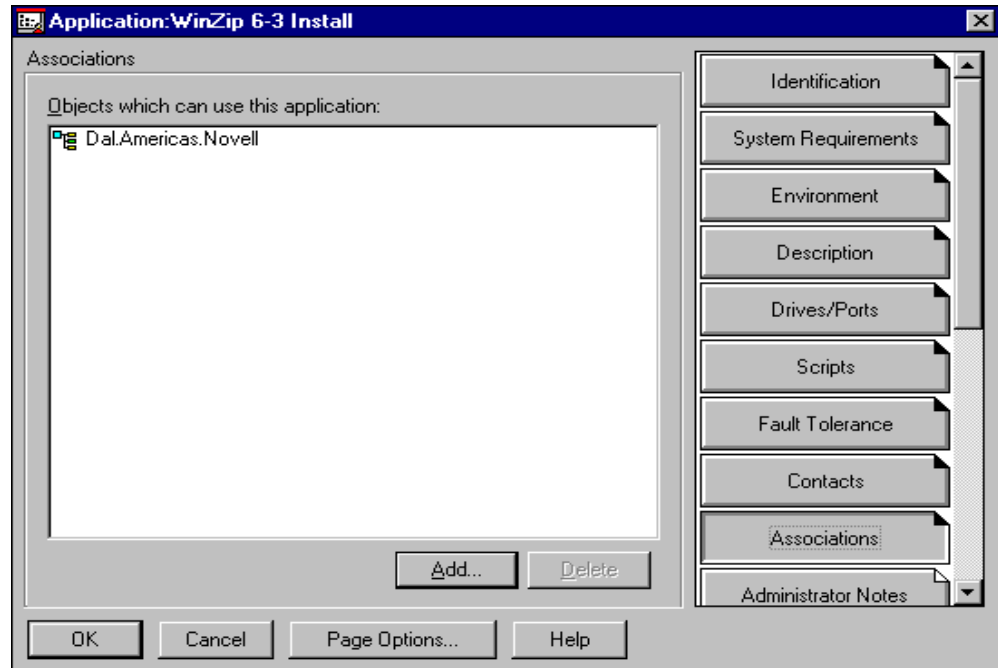
To view this information the users will do the following:

1. From the NAL Window, Right-click the Application object and select Properties
2. Select the Contacts tab.
3. When completed, click close.

Note: This information must be defined in NDS before NAL may obtain this information.



Associations



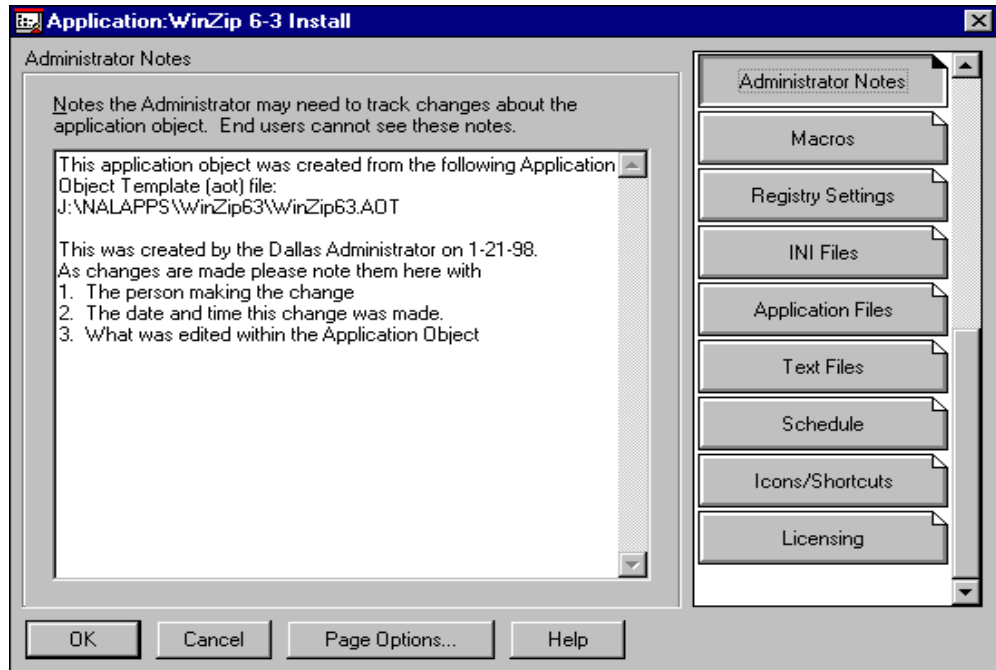
In the Associations Page we associate which Containers, Groups, and/or Users can run the application. If there are no associations then nobody will see this application, this is a Mandatory property if a user, group or container is to see the Application Icon.

Only direct associations appear. For example, if a user is a member of a Group and the Group is associated with an application, viewing the User object does not show the application association and viewing the application does not show the User association. However, the Group object shows both the User and the application associations. If you create several Application objects, you can use the Applications property page for a given User object to assign those applications to the User. Furthermore, you can specify how applications are displayed on the workstation (that is, through the NAL Window, NAL Explorer, Start menu, Desktop, and System Tray or even a Force Run).

NOTE: All applications are displayed in the NAL Window by default. By updating the Applications property page, on the User, Group or Container which the Association is assigned. There you can expand a user's access to include the NAL Explorer, Start menu, Desktop, and System Tray and Force Run (if your users are running the NAL Explorer software for Windows 95 or Windows NT).



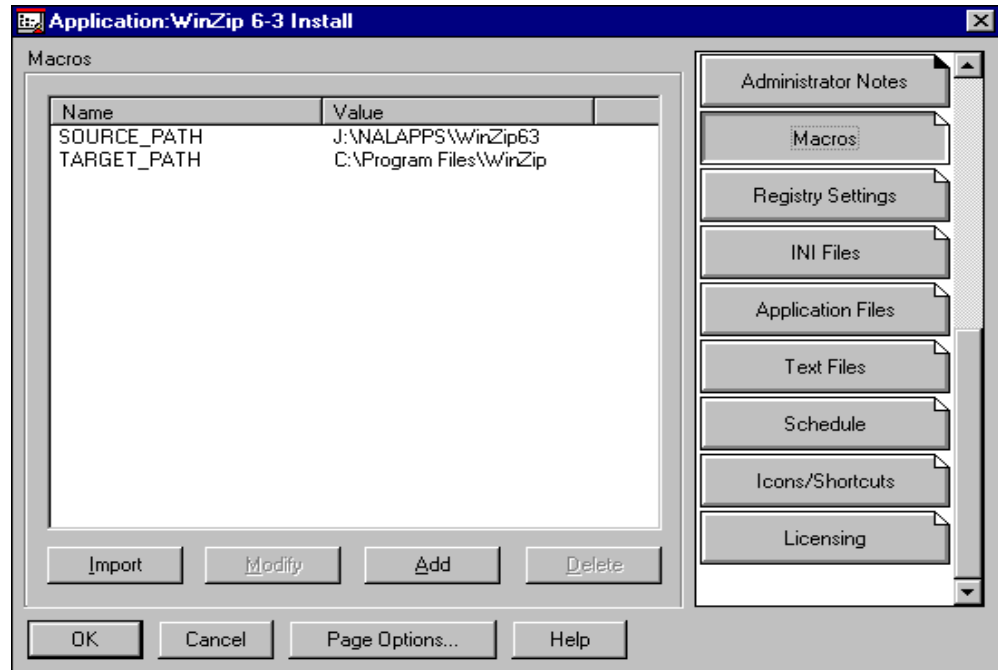
Administrator Notes



In the Administrator Notes Property there is a text field (not viewable by users) which may be used to track changes made to the application object or any other special requirements. This information in the first 3 lines actually came from the AOT file when the Application Object was created. As changes are made the Administrator may want to insert that information and date of change in this section. This can become very valuable for the Administrator to note changes or special information pertaining to the application object.



Macros



A macro is a variable name. The value of a variable is substituted in place of the variable name. If the variable is not found, it is not replaced with a value. A macro value is generally a string value. The order in which the following variables are listed below is also their order of precedence. In the Macros Page as with the Registry Settings Page and INI Files Page. Edits can be manually entered or imported from an AOT file that was created by the snAppShot utility.

Use the Macros property page to manage the macros that are used on other property pages of the Application object. You can use macros in the following Application object locations:

- Full Path
- Command Line
- Working Directory
- Mapping Path
- Capture Port Path
- Registry Property Page: Key, Name, Value (String only)
- .INI Files Property Page: Group, Name, Value
- Application Files Property Page: Source/Target, Directory

Text Files Property Page: Find and Add String

Icons/Shortcuts Property Page: All locations

For example, suppose that you have a complex Application object that uses the



%SOURCE_PATH% macro throughout numerous property pages. What if the path that %SOURCE_PATH% points to changes? Rather than change each case, you can change the value of %SOURCE_PATH% one time on the Macros property page, and the change is reflected on all other property pages where the macro is used.

For an example of macros and how they might be used:

A GroupWise Application object runs OFWIN.EXE with a command-line parameter:

```
/@U@USERNAME@
```

The USERNAME can be replaced with a macro that uses a user's NDS Object name (CN):

```
/@U@%CN%@
```

If the NDS Object name is the same as the e-mail login for GroupWise, every user that runs the application has the correct USERNAME passed in to GroupWise.

Special Variables

A special variable is one that defines Windows directories. Typical values are shown in parentheses.

%*WinDir%	Windows Directory (c:\windows)
%*WinSysDir%	Windows System Directory (c:\windows\system or c:\winnt\system32 for Windows NT)
%*WinDisk%	Drive Letter (Plus Colon) for Windows Directory (c:)
%*WinSysDisk%	Drive Letter (Plus Colon) for Windows System Directory (c:)
%*WinDesktop%	Windows Desktop Directory (c:\windows\desktop or c:\winnt\profiles\ <username>\desktop for Windows NT)</username>
%*StartMenu%	Windows Start Menu Directory (c:\windows\start menu or c:\winnt\profiles\ <username>\start menu for Windows NT)</username>
%*WinSys16Dir%	Windows NT 16-bit System Directory (c:\winnt\system)
%*CommonWinDesktop%	Windows NT Common Desktop Directory (c:\winnt\profiles\all users\desktop)
%*CommonStartMenu%	Windows NT Common Start Menu Directory (c:\winnt\profiles\all users\start menu)
%*TempDir%	Windows Temporary Directory (c:\windows\temp)

Application Object Macro Variables



An Application object macro variable is one that is defined only for the Macros property page for a particular Application object. For example:

%SOURCE_PATH% Location of source files
%TARGET_PATH% Location to copy files

NDS Attribute Variables for Current User

You can add variables that are defined by an attribute in an NDS Object. Usage:
%NDSATTRIB%

Examples

%CN%	Context Name
%Given Name%	Given Name
%Surname%	Last Name
%Full Name%	Full Name
%Telephone Number%	Telephone
%Home Directory%	Home Directory
%Email Address%	E-mail Address
%Mailbox ID%	Mailbox ID

NOTE: By default, the currently logged-in user is the source NDS object for these attributes.

NDS Attributes From Other Objects

Using the following usage, %.OBJNAME.CONTAINER;NDSATTRIB%, an example might be as follows:

%.UserName.Novell;Description%
Will provide Description from the UserName.Novell Object

NDS Attributes from Current Application Object

Using the following usage, %*;NDSATTRIB%, examples include

%*;APP:Caption%	Application Icon Title
%*;APP:Path%	Path to Executable
%*;APP:GUID%	GUID
%*;APP:Version String%	Version Stamp



Environment Variables

The following are defined by an environment variable:

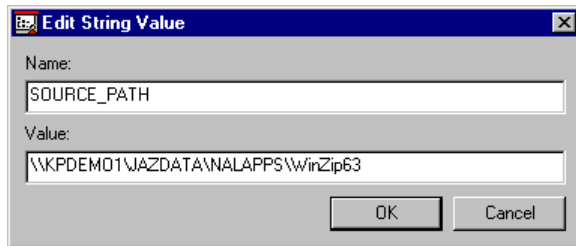
%NWLLANGUAGE%

%TEMP%

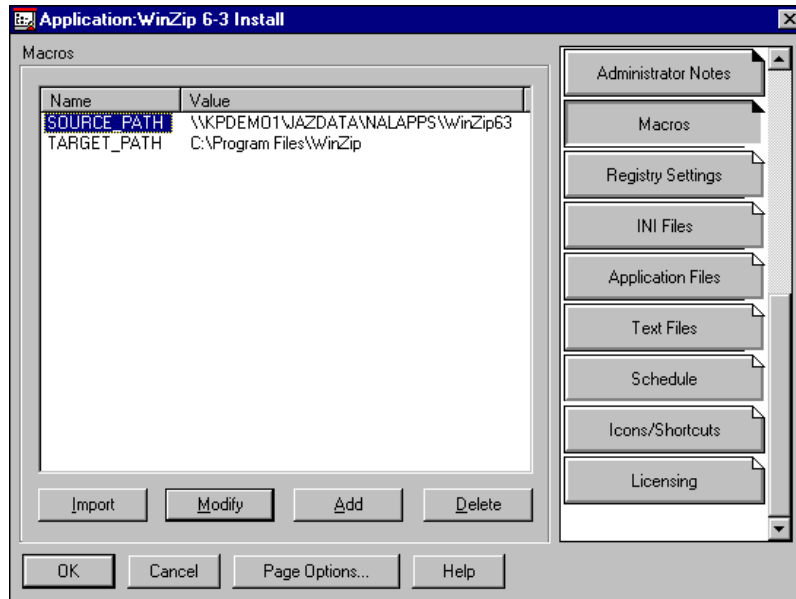
%PATH%

NOTE: The value of the variable must not exceed the length of the Application object name; otherwise, the variable fails.

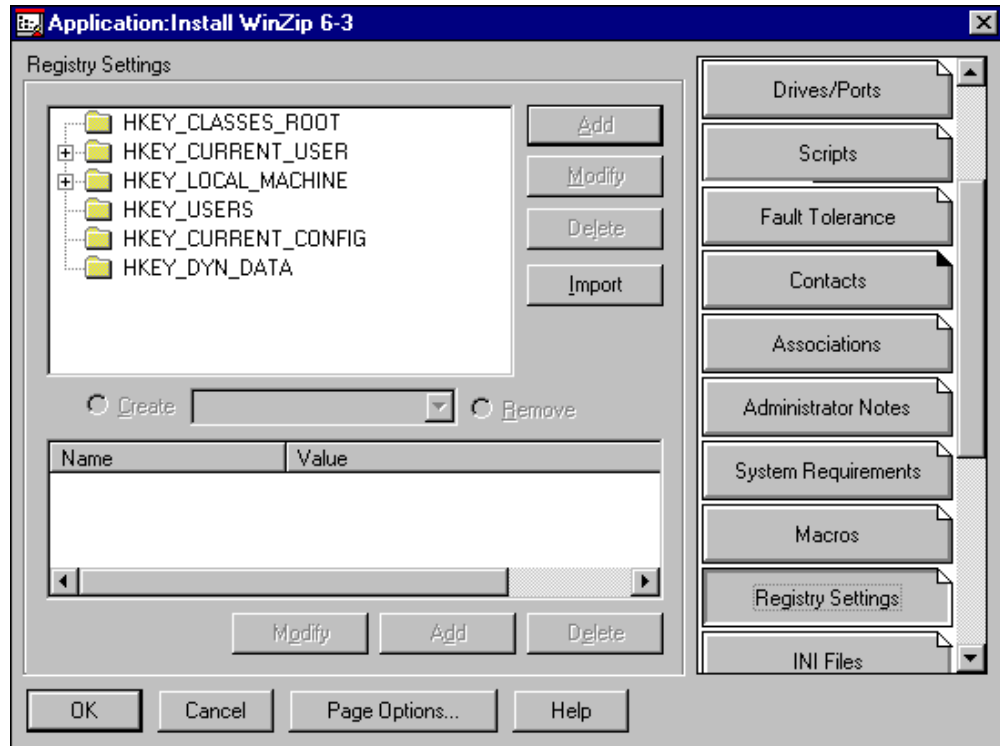
To edit this information either select the Macro and edit or double-click the macro and the following window will appear. Make the required changes and click OK.



The changes will then be reflected as seen here.



Registry Settings



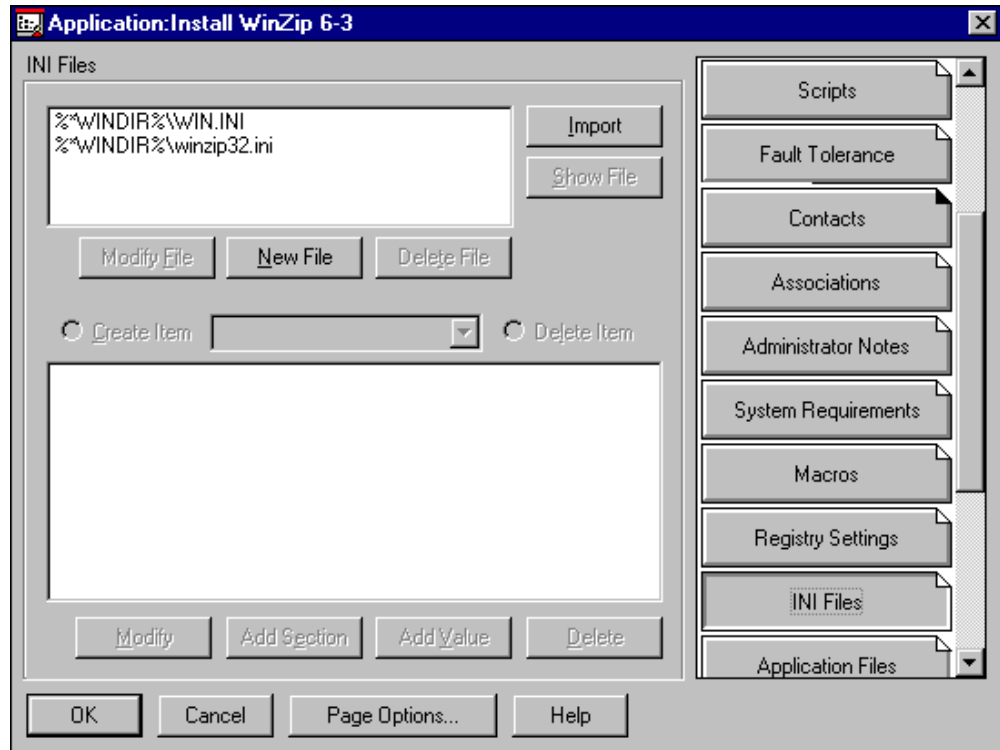
From the Registry Settings Page you can see what registry changes are made from this application. If there are any registry changes to be made they are indicated with a “+” next to the key. We mentioned earlier how snAppShot automatically made changes to this property. It is very similar to using REGEDIT if you decide to make manual changes.

This can be very useful for administrators to always push the Registry settings for all of the Snap Ins you might have installed. This will eliminate the “?” next to objects in NWADMIN if the Snap Ins are registered before launching NWADMIN.

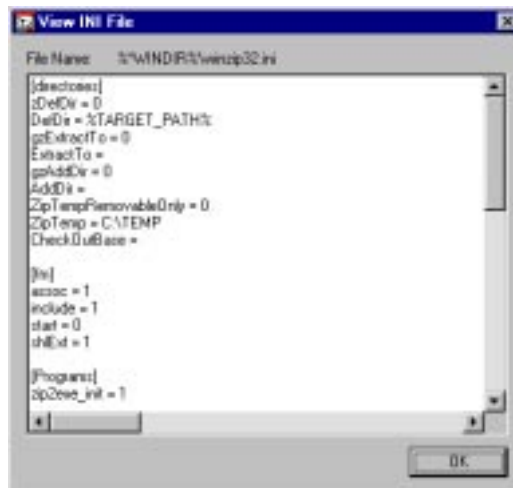
Also if you must make a simple registry change for example proxy configuration for NetScape Navigator. This is held in the registry and instead of touching every workstation update them when they launch NetScape so they will be directed to the new proxy and be productive immediately. As long as the administrator understands what is required to be changed in the registry this is extremely useful and administrators and help desk personnel can use their time more efficiently on other productive tasks they are responsible for.



INI Files



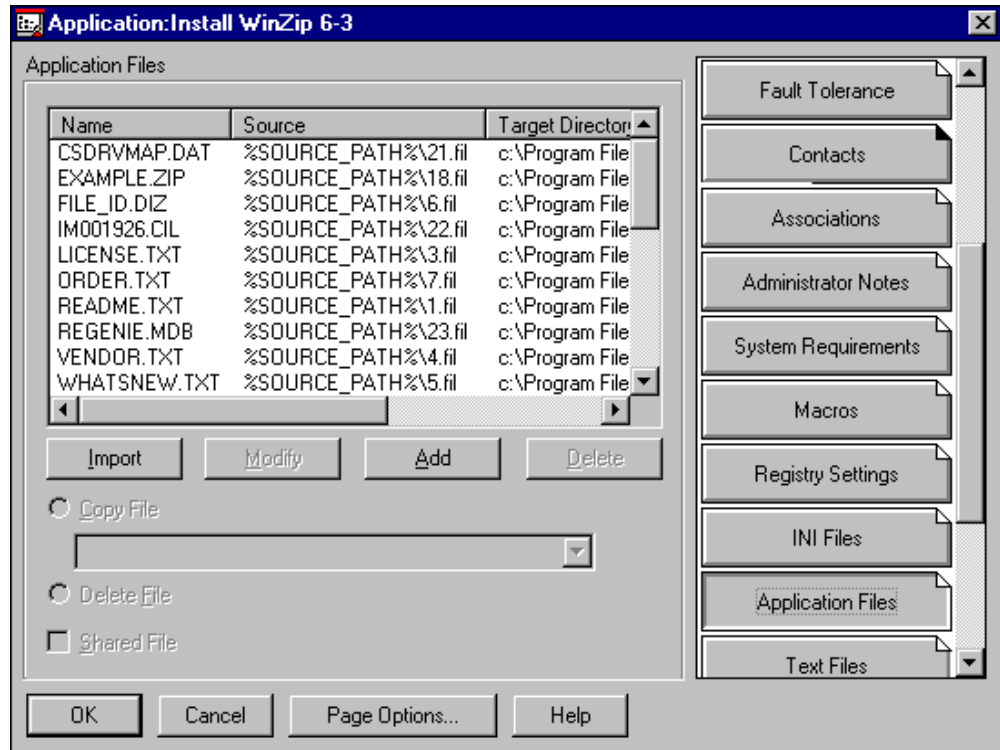
At the INI Files Page as with the Registry Settings Page, you can see what INI files are updated. Edits can be either manually entered or imported from an AOT file from the snAppShot utility. Use the INI Files property page to create or change the INI files on the target workstation when the application runs.



To edit or view these settings select the INI file you wish to view and click “Modify” or double-click the INI file. You can also Add a section, Add a value or Delete the INI file. An example of what can be viewed is in the window here to the left.



Application Files



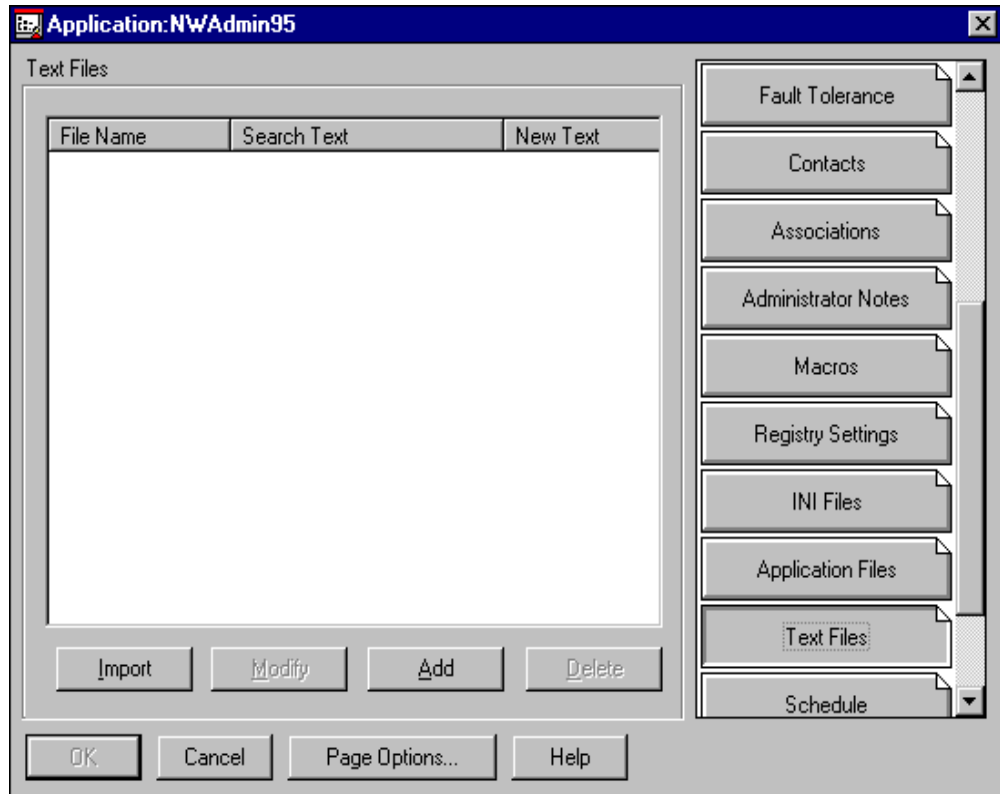
In the Application Files Page we will see a list of files installed to run the applications. The snAppShot utility generates these application files automatically and they are displayed in this window if any. This information is also logged in the directory you installed the application files to named FILEDEF.TXT. All of the install files will have a .fil extension. The FILEDEF.TXT file is an index of what files are installed and where they are placed. You will notice that the information in this window is a GUI view of where the files will be installed with the location being based on the Macros.

NOTE: Select the Shared File option to mark a file as a shared file (that is, one that is going to be used by more than one application). Files that are usually flagged as shared are Windows .DLL files (such as MFC40.DLL and MFC42.DLL). The NAL snAppShot component detects shared files when it discovers application installation changes on a workstation.

If you need to update only 1 file you want to find the file name in the FILEDEF.TXT file. This file will be renamed to a number.FIL for example we want to update WINZIP32.EXE. We can copy the new WINZIP32.EXE to this directory where the *.FIL files are located. Rename the 24.FIL to 24OLD.FIL and rename WINZIP32.EXE to 24.FIL. This will now use the newer WINZIP32.EXE we need to use the next time the application is run from NAL.



Text Files



In the Text Files Page you might use this to change workstation files such as CONFIG.SYS and AUTOEXEC.BAT. You might also like to use a text file for version tracking of sorts.

For example, suppose that users are experiencing problems due to an incorrect text string found in their workstation's CONFIG.SYS file. Rather than visit and change each workstation or run the risk of users incorrectly and inconsistently implementing a change, you can set up a Text File template that finds, deletes, modifies, or adds text strings to the text file of your choice. The Text File template implements the changes the next time the application runs.

NOTE: It is possible that the text files that you are changing might be in use. Click Changing Workstation Files In Use for information about how NAL changes such files.



Schedule

Application: WinZip 6-3 Install

Schedule

Specify the schedule when this application object is to be active:

Date:

By range: Start Date: 01/21/98 End Date: 01/30/98

Select the days when you want the application object active:

Sun Mon Tue Wed Thu Fri Sat

By selection:

Time for selected week day:

Start Time: 08:00:00 am End Time: 05:00:00 pm

Spread from start time: 5 Minutes

Use this schedule in GMT for all clients

Administrator Notes
Macros
Registry Settings
INI Files
Application Files
Text Files
Schedule
Icons/Shortcuts
Licensing

From the Schedule Page we can set an application to run only during certain times. If you schedule something to be run at 8 am and a user logs in at 9 am, the application will run provided the “End Time” is past 9 am..

For example, suppose you want users to have access to a particular application for only two weeks. After that, you want to remove the application from their view. Using the settings on the Schedule property page, you can automatically remove the icon from users' workstations.

By range

After clicking the Modify button, use this option to select a range of time in which to make the application available. For example, if you select a start date of 1/21/98 and an end date of 1/30/98 with a start time of 8:00 a.m. and an end time of 5:00 p.m., the Application object is available from 1/21/98 at 8:00 a.m. until 1/30/98 at 5:00 p.m.

You can also use the By Range option to filter out certain days of the week. For example, you could select a range that excludes Saturdays and Sundays. This effectively makes the application available on weekdays only from the hours of 8:00 am - 5:00 pm.



By selection

Use this option to select specific dates during which you want the application to be available. For example, if you select the dates 2/2/98, 3/2/98, 4/2/98, and so on, and start and stop times of 8:00 a.m. and 5:00 p.m., this makes the application available from 8:00 a.m. to 5:00 p.m. on each of the days selected. The application is not available on any other days or at any other time. This could be useful in making applications available only on particular days. You can select no more than 350 specific dates for this option.

NOTE: When scheduling applications, you can also force them to run at the scheduled time (in addition to merely displaying them). Because the Force Run option is available on a per-association basis, you must select it from the Applications property page located on a User, Group, Organization or Organizational Unit object. If the association is not set up for a Force Run, the application icon is displayed according to the location specified by the association.

Spread from Start Time

If an application were to become available at 8:00 a.m., you wouldn't want all users to run the application at the same time for fear of bringing down the network because of the load and traffic. The spread option literally "spreads out" user access times over the number of minutes specified so they don't all run the application at once.

For example, if the application is scheduled to be available at 8:00 a.m., and the Spread from Start Time is set to 10 minutes, the application becomes available, on a random basis, between the hours of 8:00 a.m. and 5:00 pm. Thus, the demand for the application is spread out over a longer period of time and network traffic is minimized. If users access applications after the spread time is expired but before the end time of the Application object, they access the application at that time and the spread variable has no effect.

GMT (Greenwich Mean Time)

All application scheduling that you do with the Application property page is based on the workstation's time zone. In other words, if your network spans different time zones and you schedule an application to run at 1:00 p.m., it runs at 1:00 p.m. in each time zone. However, if you select the GMT check box, workstations run applications according to GMT (or Greenwich Mean Time).

For example, suppose you schedule an application to run at 1:00 p.m. and you are in Utah (Mountain Standard Time). By selecting the GMT option you, in effect, have really scheduled this application to run at 8:00 p.m. GMT time. Timezones are ignored and the entire network around the world runs this application at the same moment.

NOTE: GMT time is not available if you are filtering out days of the week when in the By Range mode.

Setting Up Windows 3.x Clients for Timezones



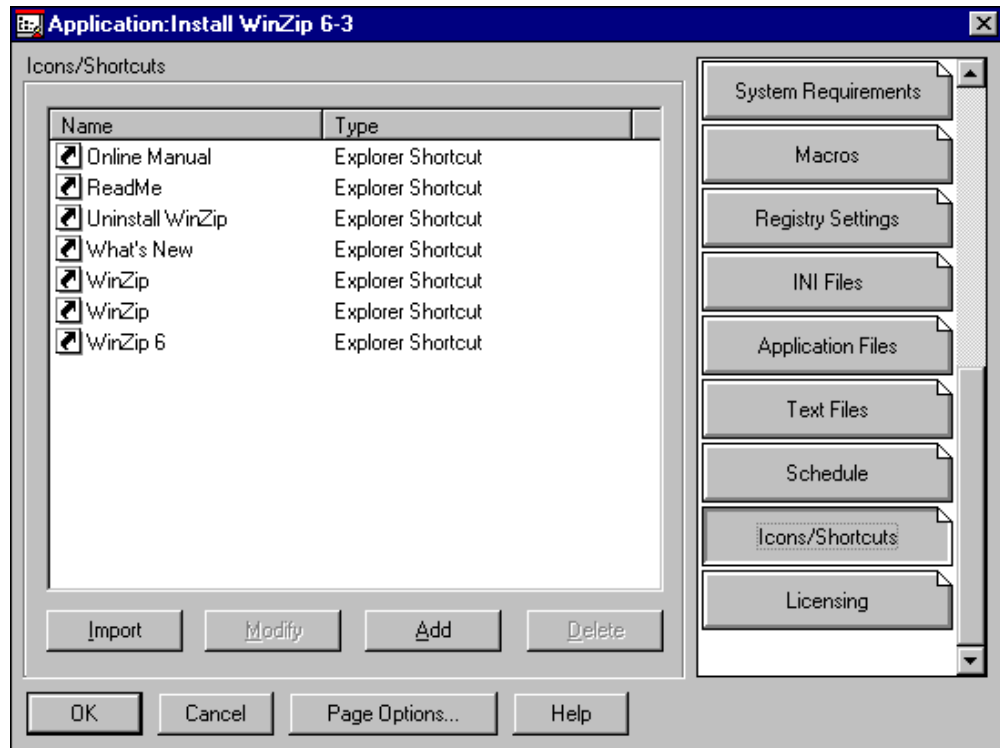
Windows 3.x clients do not use a timezone concept. The only way Windows 3.x clients can use a timezone is if it is set in their environment. You can do this by putting the command

```
SET TZ=MST7DST
```

in the AUTOEXEC.BAT or login script. This example sets up the time zone to Mountain Standard Time with the option of setting the Daylight Savings Time when Daylight Savings Time is in effect. NAL also looks for the NALTZ environment variable. Some applications do not run if you set the TZ environment variable. If no environment variable is set, the default time zone is Eastern Standard Time.



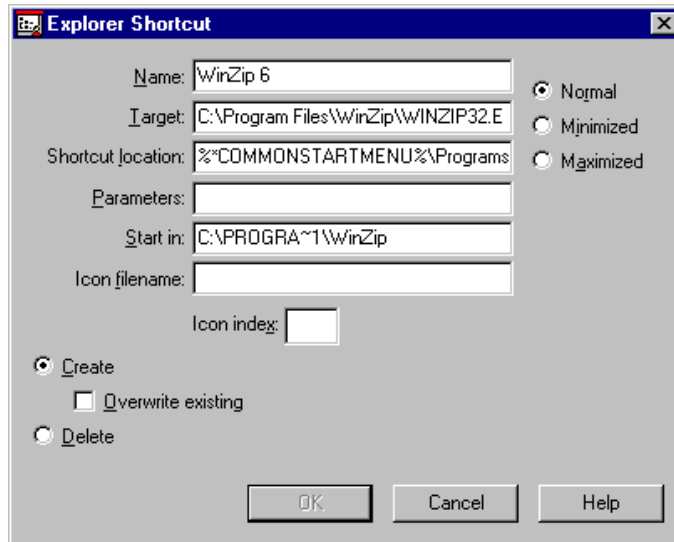
Icons/Shortcuts



From the Icons/Shortcuts Page you can control Icons that appear in the Windows 3.1x Program Manager and Shortcuts in Windows 95 and Windows NT Explorer. The items that appear in the Icons/Shortcuts list control the following: Program Group, Program Group Item, and Shortcut.

For example, suppose that your company uses special icons for the applications that users run on their Windows 95, Windows NT and Windows 3.x workstations. Using the Icons/Shortcuts property page for this Application object, you can change the icons for the other applications the next time this application runs.





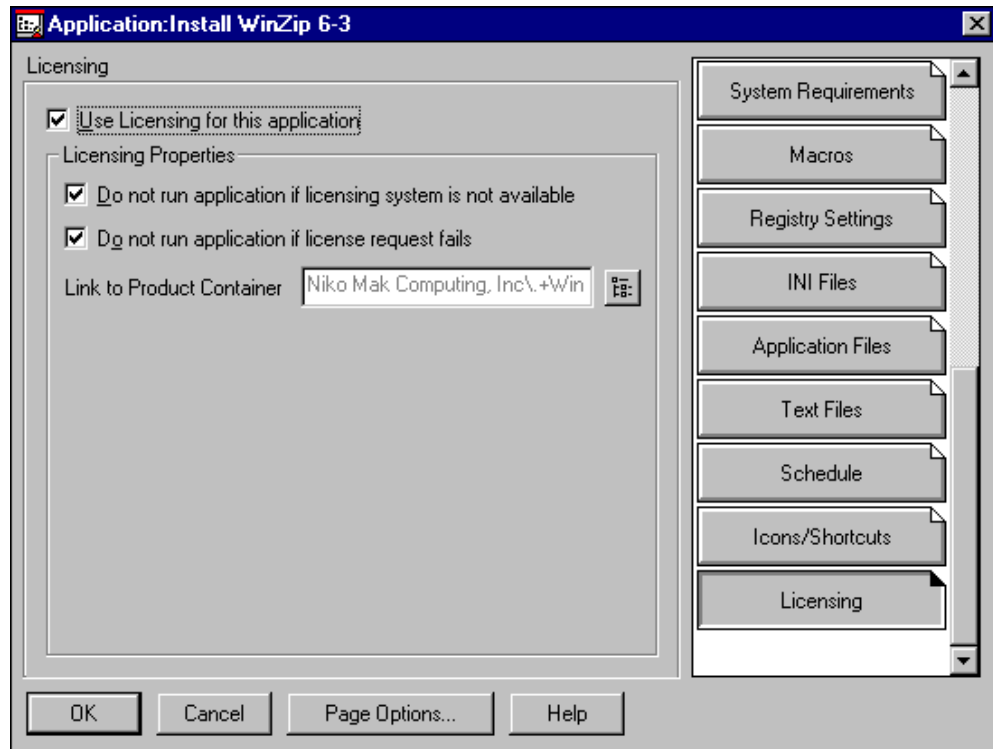
This window will allow you to change the information for this shortcut. As with this shortcut, this is set to run in a Normal window but you want this to be Maximized when the user launches this application. You might even want to change the location of the shortcut.



NAL License Metering

This is an additional piece to NAL 2.0 that Novell Consulting has written. Many customers want to meter applications to be legal. With this additional service this can easily be done by Administrators. This will be turned over to the NAL Product Team soon but still an additional fee will be charged for the NAL Metering service. Contact your local Novell Representative for pricing and availability information on this service.

Licensing

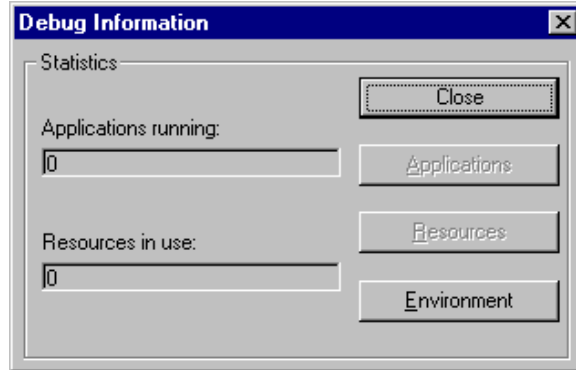


This property is new and provided currently only from Novell Consulting. This will soon be given to the NAL Product Team to implement in NAL but will be an additional charge. With the NAL Metering you can create licensing information so that when users launch an application from NAL this information is logged and maintained.

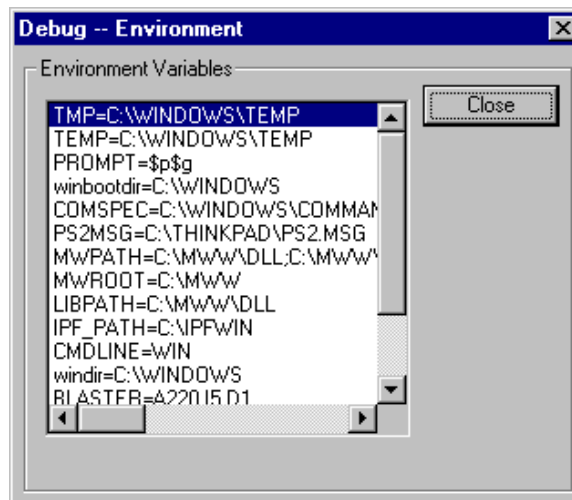


NAL Helpful TidBits

- From the NAL Window (NAL.EXE) Holding down <SHIFT> while selecting “Help, About NAL” gives debug information. This very useful to debug or verify environment information is correct.



If we select Environment we see the following information. This can be helpful to look at environment variables that have been setup and may be used by NAL.



- From Windows Explorer or Double Click “Network” hold down <CTRL>-<ALT> and double-clicking on the Tree Name while in “About” also gives you additional information.
- NRD = NetWare Registry Database. Writes and reads registry settings off the User object vs. the workstation’s Registry.



To store personal folder information in a User object

1. Make sure that your search path contains the location where NetWare Administrator files are stored.
2. Grant rights in the NetWare Registration Database to all users who are storing personal folder information in their User object. You need to grant rights to both NRD: Registry Data and NRD: Registry Index.

NOTE: Usually, Users' personal folders and contents are saved when the User exits the NAL Window. However, if you save folder information in the Novell* Network Registry* database and Users change their Novell Directory Services* (NDS*) tree connections while running the NAL Window, the personal folders and contents might be lost. To avoid this, your Users can save the NAL Window settings by holding down the Shift key while choosing File > Exit.

- Store your NWADMIN settings in your User Object
Create an Application Object for NWADMIN with the following syntax in the Path to the executable

NWADMNxx [/N | /Ffile]

For Windows 95 use the following, NT substitute WINNT for WIN95,
\\SERVERNAME\SYS\PUBLIC\WIN95\NWADMN95.EXE /N

Options

xx is "NT", "95" or "3X", depending on your Windows version.

/N causes your preferences to be registered in your User object in the

Directory.

/F causes your preferences to be registered in the specified file. If you don't include a path, the local Windows directory is assumed.

By not using these parameters causes your preferences to be registered locally (in the Windows registry, or in the WINDOWS\NWADMN3X.INI file if you are running Windows 3.1).



Container Login Script for NAL

Administrators may decide to have all or select users automatically start NAL from the Login Script. After you have used NAL in your lab and the pilot has been successful you may want to deploy to users on a group or department basis. With the variables that Novell provides from login you may setup the script as follows:

```
;---This is an example container login script to automatically start NALEXPLD for--
;---Windows 95 and Windows NT users that belong to the NALGROUP -----
Write "This is the beginning of the login script process"
Write "Good %GREETING_TIME, %LOGIN_NAME"
MAP F:=%FILE_SERVER\SYS:
  MAP ROOT INS S1:=%FILE_SERVER\SYS:PUBLIC
  MAP ROOT W:=%HOME_DIRECTORY
If "%OS" = "WIN95" Then begin
  GOTO WIN95
END
If "%OS" = "WINNT" Then
  GOTO WINNT
END
ELSE GOTO UPDATE
END

WIN95:
  Write "Your current desktop OS is WIN95"
  IF MEMBER OF "NALGROUP" THEN BEGIN
    GOTO NALGROUP
  END
ELSE GOTO EXIT
END

WINNT:
  Write "Your current desktop OS is WINNT"
  IF MEMBER OF "NALGROUP" THEN BEGIN
    GOTO NALGROUP
  END
ELSE GOTO EXIT
END

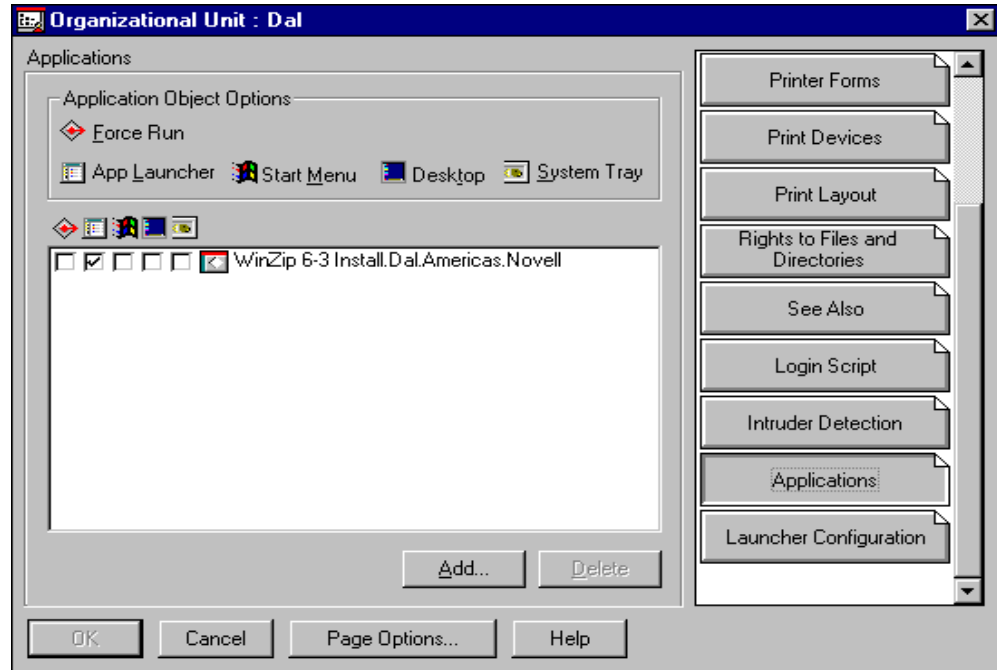
NALGROUP:
  @\%FILE_SERVER\SYS\PUBLIC\NALEXPLD.EXE /S
EXIT:
EXIT
```



With the line @\\%FILE_SERVER\SYS\PUBLIC\NALEXPLOD.EXE /S this will allow any “Force run” applications to be installed at the time the user logs into the network. To setup where the Applications are viewed in the NAL Explorer you will need to go to the User, Group or Container where the Application is Associated.

Application placement for NAL Explorer

From within NWADMIN select the Container where you have Associated the Application. From the example we will look at the Dal Container.

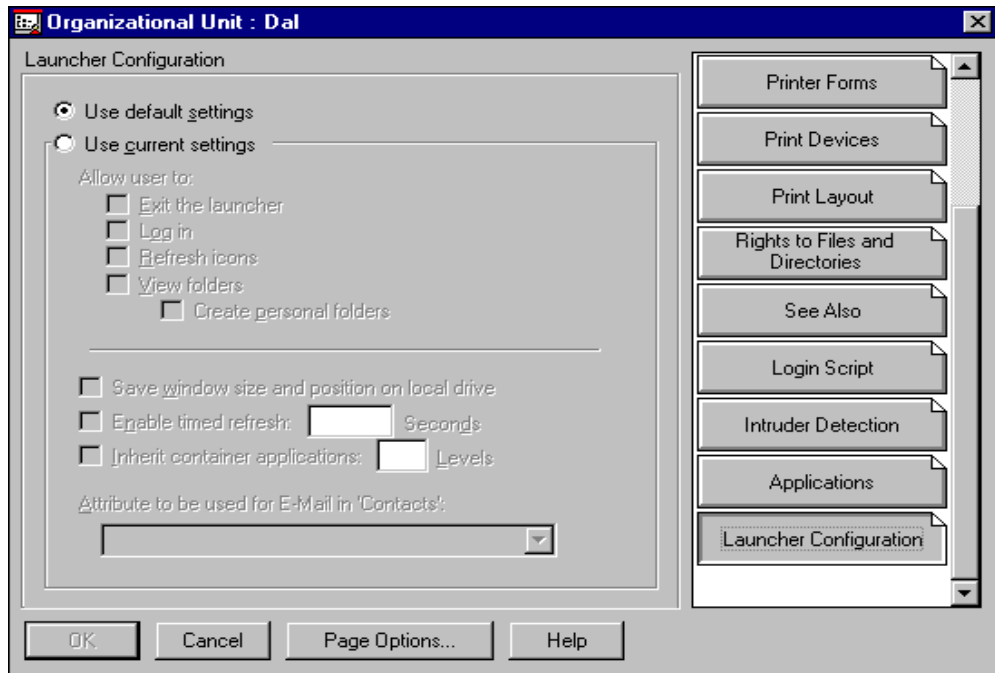


From this Property Page we can select the WinZip 6-3 Install.Dal.Americas.Novell Application to be a Force Run, displayed in the App Launcher (default), from the Start Menu | NAL Explorer, an Icon on the Desktop, or even placed in the System Tray. As you see by using the NAL Explorer you have more options and control of what the users sees and where the Application appears.

Inheritance of an Application

As you are already familiar with you can inherit rights or applications in this case from NetWare and NDS. With NAL extending the schema upon installation the User, Group and Container objects have another Property called Launcher Configuration. By default this property is set to Use default settings seen in the next figure.



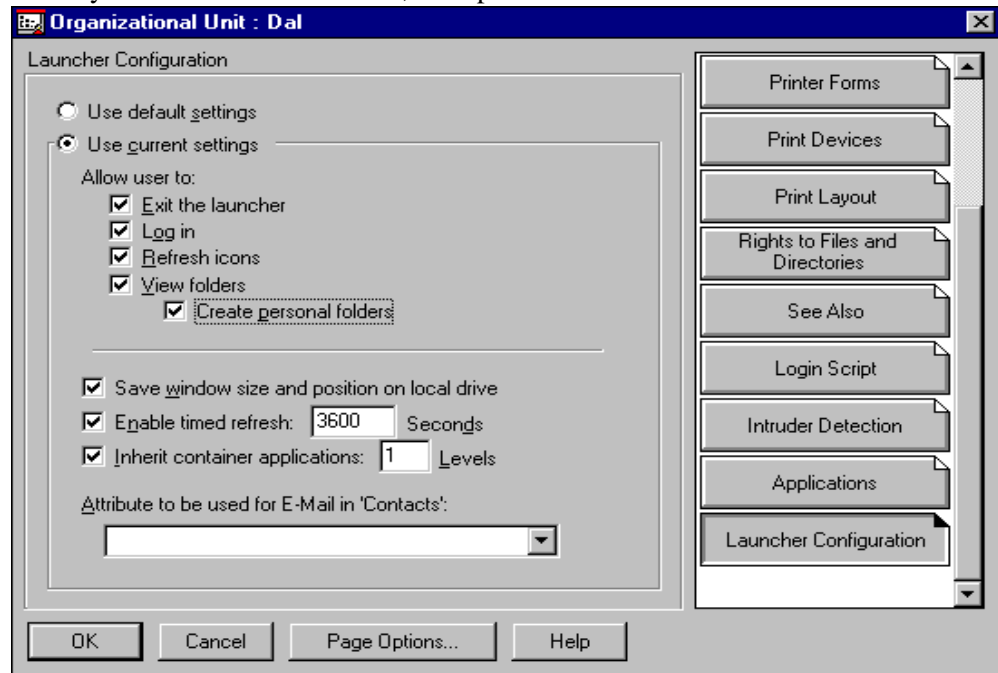


The container defaults are as follows:

Setting	State
Exit the launcher	On
Log in	On
Refresh Icons	On
View Folders	On
Save window size and position on local drive	On
Enable timed refresh: XX Seconds	Off (<i>if this is enabled 3600 seconds, or 1 Hour is the default</i>)
Inherit container applications: XX Levels	1 Level



We may customize this for a User, Group or Container as seen here.



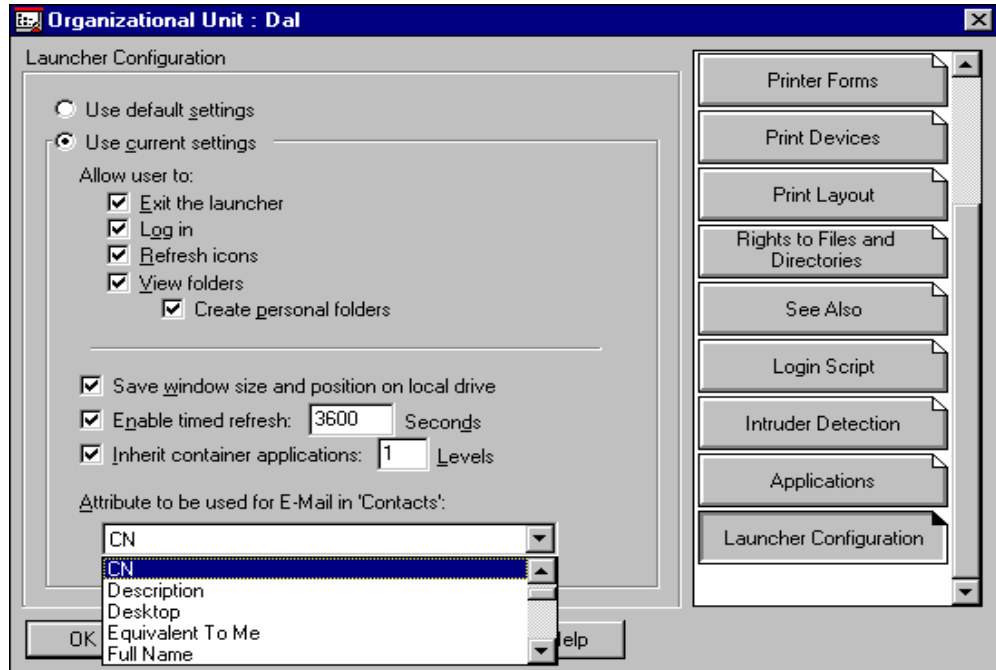
To enable each of the above attributes we must select the option box so the box has a check in the box as displayed above.

If I wanted a user to inherit an application that were at a level up in the tree I would need to change the “Inherit container application: 2 Levels”. This would allow a user to inherit an application from container Americas.Novell as well as the current container of Dal.Americas.Novell. If you are planning to place Applications that are high in the tree (at the top layer or the Organization level) so all users “Inherit” them you may set this property to *-1* which will browse to the [Root] of NDS checking for changes for NAL. This is not recommended as now you may be adding additional over head if that server does not have replicas of the parent partitions. Also if an application is an install of 50 MB that would be slow and could cause performance issues on the WAN link depending on the WAN infrastructure. Before creating an Application at that level of the tree review your NDS Design and WAN links. Remember to keep resources local to the users that will be using them.

The “Enable timed refresh: *XXXX* Seconds” is how often the user will poll NDS if there are any changes to my applications. If an Application were added to my container or user directly it will take up to 1 hour for NAL to obtain this change. The “Refresh icons” if enabled will allow a user to press “F5” to check NDS at that time if there are any changes. This would work the same whether the user is gaining or losing access to an application.



We can also setup an E-mail address to be viewed when the Properties are viewed by the user you can have the GroupWise Mail Address displayed by selecting the



The Attribute to use for E-Mail contact can be configure to use the NDS information in the user objects property that is selected. In this example we will use the Common Name “%CN” for the user that is placed in the “Contacts” property of the application (refer to page 38). If GroupWise is your E-mail package you can send the “Contact an E-Mail to obtain other information on the application.



Error Codes

The following is from the TID 2927376 at <http://support.novell.com/>
I have also added some additional information based on errors I have seen and what the error may be related to.

NAL 2.0 Error Codes (Last modified: 13JAN1998)

Issue

Novell Application Launcher (NAL) Error Codes - NAL 2.0, 2.01:

0xD000

Unable to get attributes for application object [object Name] (id=number)

Possible Causes:

Object does not exist in DS

User does not have rights to read object attributes

0xD001

Could not get needed resources for application [application name] to be launched (id=number).

Problem: Invalid drive path [specified drive path]

Possible Causes:

Specified path is incorrect format or not found

0xD002

Could not get needed resources for application [application name] to be launched (id=number).

Problem: Invalid queue path [specified queue path]

Possible Causes:

Queue does not exist in specified location

Specified Queue name is incorrect

0xD003

Could not get needed resources for application [application name] to be launched (id=number).

Problem: Unable to connect to server [server name]

Possible Causes:

Server is not available

Incorrect server name

0xD004

Could not get needed resources for application [application name] to be launched





(id=number).

Problem: Drive requested is already in use. Unable to map drive [drive letter]

Possible Causes:

Drive is already mapped to a location different from the requested location

0xD004

Could not get needed resources for application [application name] to be launched (id=number).

Problem: Unable to map drive [drive letter]

Possible Causes:

Target server is not available

Inadequate rights to target server

Directory does not exist

Volume does not exist

0xD005*

Could not get needed resources for application [application name] to be launched (id=number).

Problem: Port requested is already in use.

Possible Causes:

Port is already in use to a different Queue than the one requested

0xD005*

Could not get needed resources for application [application name] to be launched (id=number).

Problem: Unable to capture port [whole capture string]

Possible Causes:

No rights to Queue

Queue does not exist

0xD006

Could not get needed resources for application [application name] to be launched (id=number).

Problem: Invalid capture parameter [whole capture string]

Possible Causes:

Unknown capture parameter

0xD007

Could not get needed resources for application [application name] to be launched (id=number).



Problem: Setting capture settings [whole capture string]]

Possible Causes:

Unable to set desired capture flags

0xD008

Could not get needed resources for application [application name] to be launched (id=number).

Problem: Script execution failed (error #)

Possible Causes:

General error when trying to run script

0xD009*

Could not launch [application name](using specified path) (id=number)

Problem: File was not found or DLLs required to run are missing.

Possible Causes:

.EXE file not found

DLLs required for the .EXE not found

0xD009*

Could not launch [application name] (using specified path) (id=number)

Problem: Path specified is invalid.

Possible Causes:

Path specified to .EXE or working directory not found

0xD009*

Could not launch [application name] (using specified path) (id=number)

Problem: Not enough memory to run application.

Possible Causes:

Low on memory

0xD009*

Could not launch [application name] (using specified path) (id=number)

Problem: One of the library files needed to run this application is damaged.

Possible Causes:

DLL required by the application is corrupt

0xD009*

Could not launch [application name](using specified path) (id=number)

Problem: The specified program requires Microsoft Windows 32-bit extensions.

Possible Causes:

Trying to run a Win32 application on Windows 3.1

0xD00A

Could not get needed resources for application [application name] to be launched (id=number).

Problem: Script processing library (library name) could not be found.

Possible Causes:

Could not find the following libraries on :

Windows 3.1x - LOGIN31.DLL, LGNW3116.DLL

Windows 95 - LOGINW95.DLL, LGNW9532.DLL

Windows NT - LOGINWNT.DLL, LGNWNT32.DLL

0xD010

Application was not launched due to a terminate command in the pre-launch script.

Possible Causes:

Term A#@ where # was a non-zero number that was executed in the pre-launch script

0xD011

Could not get needed resources for application [application name] to be launched (id=number).

Problem: Search drives are not available for mapping [specified drive]

Possible Causes:

On Windows 3.1 must be running in enhanced mode and have NetWare.driv v3.0 or greater

0xD012

Could not get needed resources for application [application name] to be launched (id=number).

Problem: Scripts cannot be run without a newer [file name] file.

Contact your system administrator for newer script processing files.

Possible Causes:

Could not find the following libraries on :

Windows 3.1x - LOGIN31.DLL, LGNW3116.DLL

Windows 3.1x LOGIN31.DLL must be v.14 or greater

Windows 95 - LOGINW95.DLL, LGNW9532.DLL

Windows NT - LOGINWNT.DLL, LGNWNT32.DLL



0xD013

Could not configure workstation for application [application name] (id=number).
Problem: Invalid registry settings. (D013)

Possible Causes:

Registry settings stored in application object is corrupt

0xD014

Could not configure workstation for application [application name] (id=number).
Problem: Invalid INI settings. (D014)

Possible Causes:

.INI settings stored in application object is corrupt

0xD015

Could not configure workstation for application [application name] (id=number).
Problem: Unable to create rollback files. (D015)

Possible Causes:

Not enough disk space on local drive where TEMP directory is located

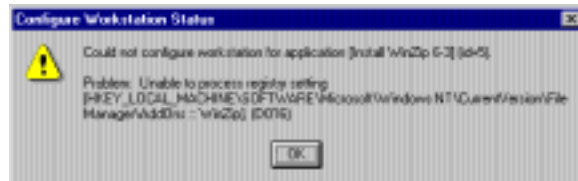
0xD016

Could not configure workstation for application [application name] (id=number).
Problem: Unable to process registry setting [specified setting]. (D016)

Possible Causes:

Invalid Key name

Unable to Create/Modify/Delete registry setting



I have seen this on Windows NT when an application is being installed and the user is not a member of the Administrators on the NT Workstation. The way around this at this time is use Novell Workstation Manager to create an install ID to install the application. Once the App is installed users should have access to the application to run normally.



0xD017

Could not configure workstation for application [application name] (id=number).
Problem: Unable to process INI setting [specified setting]. (D017)

Possible Causes:

Unable to Create/Modify/Delete INI setting

0xD018

Could not configure workstation for application [application name] (id=number).
Problem: Unable to copy file [file name]. (D018)

Possible Causes:

Source file does not exist.

No rights to copy file

No rights to source file



This error has been seen when the “Source_Path” Macro is mapped to a drive the user does not have mapped. For example if the “Source_Path”=J:\NALAPPS and the user does not have a drive mapped to J: where NALAPPS resides this error can be observed. To resolve this you have 2 options from with the NAP Application Object.

1. Use the Drives/Ports and map a drive to the source location.
2. Change the “Source_Path” to the UNC name thus avoiding a drive being mapped.

0xD019

Could not configure workstation for application [application name] (id=number).
Problem: Unable to delete file [file name]. (D019)

Possible Causes:

No rights to delete file

0xD01A

Could not configure workstation for application [application name] (id=number).
Problem: Unable to process text file setting [specified setting]. (D01A)

Possible Causes:

Unable to create file



0xD01C

Could not configure workstation for application [application name] (id=number).
Problem: Unable to process Group/Shortcut setting [specified setting].(D01C)

Possible Causes:

Unable to read group/link settings from application object

Unable to create groups - invalid characters in name

Invalid name or path for Icon or shortcut

