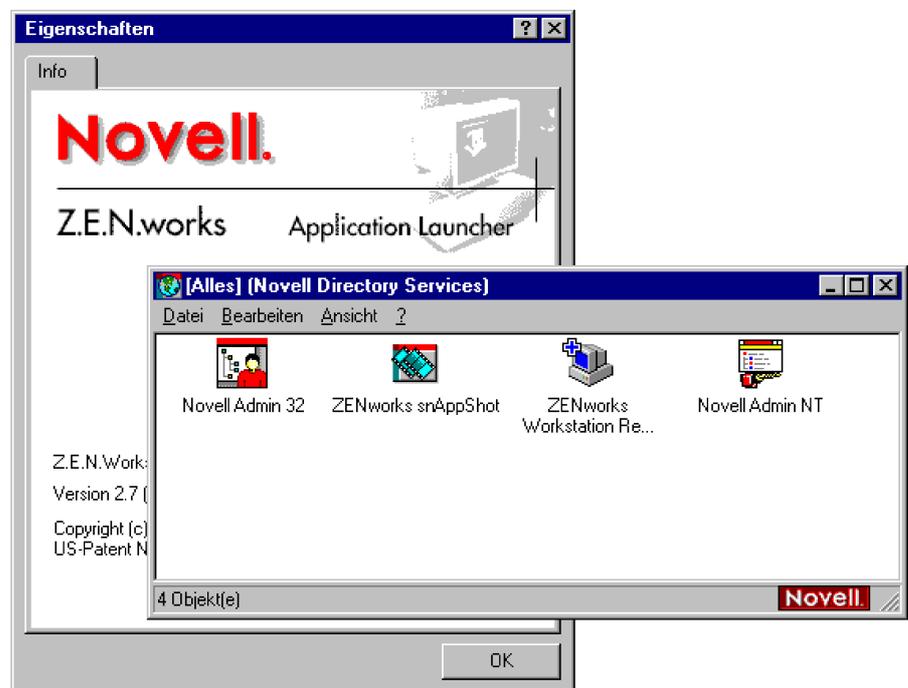


Novell Z.E.N.works

Version 1.1

Arbeitspapier



Qualifizierte Softwarekurse in der Beruflichen Weiterbildung
Die GKDVZ in Zusammenarbeit mit der VHS Aachen

Novell und NetWare sind eingetragene Warenzeichen von Novell, Inc.
MS-DOS, Windows und Word sind eingetragene Warenzeichen der Microsoft Corp., USA

Die Zusammenstellung dieses Materials wurde mit großer Sorgfalt vorgenommen. Trotzdem kann für fehlerhafte Angaben und daraus ggf. entstehende Folgen weder eine juristische Verantwortung noch irgendeine Haftung übernommen werden.

Autor dieses Scripts: Georg Helmes, Postfach 764, 52008 Aachen

Alle Rechte vorbehalten

Inhalt

Novell's Z.E.N.works	3
HINWEISE zur Installation	4
HINWEIS zur Migration	5
Die Konfiguration des Novell Client auf der Workstation	6
Einige Register aus der Client-Konfiguration	6
Neue NDS-Objekte	9
1. Das Objekt <i>Arbeitsstation (Workstation)</i>	9
2. Die <i>Arbeitsstationsgruppe (Workstation Group)</i>	10
3. <i>Richtlinienpakete (Policy Packages)</i>	10
Das <i>WINNT Arbeitsstationspaket</i>	11
Das <i>WINNT Benutzer Paket</i>	14
4. Die <i>Suchrichtlinie</i>	18
5. <i>Standort-Objekte</i>	18
6. <i>Anwendungsordner</i>	19
7. <i>License Container</i> (nur Vollprodukt)	19
Der Planer	19
Erstellen einer NDS-basierten Aktion	20
Benutzererstellte Aktionen	22
Die Anzeige der Login-Skripts	22
Der Application Launcher (NAL 2.7)	23
Einleitung	23
Installation	23
Neuigkeiten	25
Upgrade	25
Einige Hinweise zu Applikationsobjekten	25
Identifikation	25
Planen	27
Beendigung	27
Laufwerke / Anschlüsse	27
Dateirechte	28
Scripten	28
Zu den anderen Eigenschaften	28
Neue Eigenschaften im Objekt <i>Container</i>	29
Verknüpfte Richtlinienpakete	29
Arbeitsstations-Registrierung	29
Arbeitsstationsinfo	29
Arbeitsstationsfilter	29
Konfiguration des Startprogramms	29
Das neue Objekt <i>Anwendungsordner</i>	30
Das Programm <i>snAppShot</i>	31
Erstellung eines <i>Application Objects</i> mit Hilfe eines Templates	37
Eine öfters auftretende Fehlermeldung	47
... wie verschwindet das Application Launcher-Symbol vom Desktop?	47
Batchdateien in Skripten und bei Aktionen	47

Dateien löschen und wiederherstellen	48
Zusammenfassungen	49
1. Arbeitsschritte bei der Installation von Z.E.N. works	49
2. Vorgehensweise bei der Erzeugung eines Applikationsobjekts	49
Ablauf 1: Was geschieht beim Einloggen?	50
Ablauf 2: Was macht der NAL beim Starten einer Anwendung?	51
Ablauf 3: Was macht der SnAppShot?	52
Index	53
Anhänge	54
ANHANG 1: Getting Started	54
ANHANG 2: Essential Concepts	54
ANHANG 3: How to Register a Workstation	56
ANHANG 4: Upgrading from NAL 2.x to Z.E.N.works 1.0	62
ANHANG 5: Is it a Clean, Baseline Machine?	63
ANHANG 6: Update the Secure Portion of the Registry on Winwos NT	65
Mehr Infos	67

HINWEIS: Dieses Skript wurde unter Anwendung der Version Z.E.N.works 1.1 / Client 4.6 (Deutsch) erstellt. Fragen, Hinweise und Korrekturen bitte an Georg Helmes, Tel. 0172 9881533 oder an georg@stadtbaug.rwth-aachen.de. Alle Z.E.N.works-Bezeichnungen sind in *Kursivschrift* dargestellt, Zitate aus der Online-Hilfe oder HTML-Dokumentation in kleiner Schrift und letztere fallen weder unter das Copyright noch sind sie von mir geprüft.

Novell's Z.E.N.works

Z.E.N. steht für "Zero Effort Networking", also sozusagen "Netzwerk ohne Arbeit". Für die Annäherung an dieses hochgesteckte Ziel stellt das Z.E.N.-Starter-Pack, das hier beschrieben werden soll, einige neue Objekte und Werkzeuge zur Verfügung, die in ihren Leistungsmerkmalen weit über das hinausgehen, was schon mit dem Workstation Manager und dem Application Launcher 2.01 erreicht werden konnte. Dazu einige Zitate aus der (HTML-) Online-Information, die zusammen mit dem Starter-Pack geliefert wird:

Z.E.N.works* is an integrated set of technologies that lets you deliver and manage applications, configure Windows** desktops, and remotely repair workstation software problems using a single management utility--NetWare Administrator* 32.

The overall objective of the Z.E.N.works product is to reduce the cost of owning networked PCs. This is accomplished in two ways. First, Z.E.N.works lets users concentrate on their business productivity, without the need to become PC experts. Second, Z.E.N.works gives administrators tools that leverage NDS to greatly reduce the amount of necessary PC support.

Z.E.N.works is powered by NDS, which gives it a large advantage over the current marketplace offerings. Novell leverages the power of NDS in Z.E.N.works by expanding the NDS schema, which allows you to store information related to application access, desktop preferences, and printer access directly in an NDS tree. For example, using NetWare Administrator 32, you can set workstation configurations like desktop policies and client configurations and have these changes propagated (automatically by NDS) to the workstations or users you want to affect.

Z.E.N-Works steht in zwei Varianten zur Verfügung:

- dem **kostenlosen** Starter-Pack. Dieses Paket kann aus dem Internet herunter geladen werden. Im folgenden wird nur auf die Funktionen des Starter-Packs Bezug genommen.
- der (kostenpflichtigen) Vollversion

Zu den Features wieder ein Zitat:

Features Included in Z.E.N.works Starter Pack

In conjunction with NDS and NetWare Administrator, Z.E.N.works uses NDS objects (such as Application, Workstation, and Policy Package objects) to provide the following features that help you manage workstations and applications.

- Distribute and maintain applications on the workstation
- Launch applications for users automatically on the workstation
- Import Workstation object information into NDS
- Dynamically distribute and update print drivers
- Centralize the location and administration of user profiles
- View and update client configurations without visiting the workstation
- Update workstation software automatically from your office
- Create Scheduled Actions for users (Available if using Desktop Management features, this automatically runs programs you specify for Windows 95** and Windows NT** clients)

Additional Features Available in Full Z.E.N.works Product

With the full Z.E.N.works product (available from your reseller), you can:

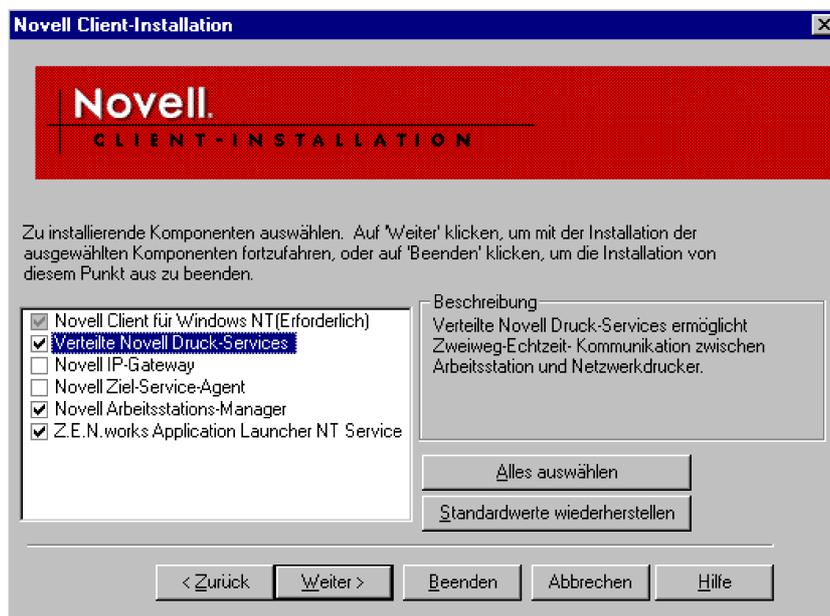
- Solve user software problems without visiting the workstation or NT server (using NDS-authenticated remote control).
- Provide end-users with a help request system. The Help Requester is an end user program that helps users provide pertinent information about themselves and their workstations to your Help Desk using e-mail or phone. This can mean faster problem resolution.
- Control the remote control feature and end-user help request system for workstations. You can set these once for a container or group or set them individually for users with special needs.
- Automatically place hardware inventory into NDS for tracking and troubleshooting.

Weitere Informationen im Anhang 2: "Essential Concepts"

HINWEISE zur Installation

Um Z.E.N.works erstmalig auf einem Server installieren zu können, muß man Admin-Rechte auf die Root des entsprechenden Baumes haben, weil die NDS-Struktur um mehrere neue Objekt-Typen erweitert wird. Außerdem werden eine Reihe von Dateien in das Verzeichnis SYS:PUBLIC kopiert. Zusätzlich wird ein Unterverzeichnis WIN32 für den neuen 32bit-Administrator angelegt, der die bisher getrennten Administratoren für Win95 und WINNT ersetzt. Installationen durch den Client 4.11a (im Verzeichnis SYS:PUBLIC\WINNT) bleiben unberührt.

Client- und Serverinstallation werden mit dem Programm WINSETUP ausgeführt. Man hat bei der Client-Installation die Auswahl zwischen *Benutzerdefiniert* und *Standard*. Wurde eine frühere Version installiert, kann es sein, dass bei einem Upgrade nur die Komponenten "upgedated" werden, die unter der früheren Version installiert wurden. Wenn man ganz sicher gehen möchte, nimmt man die benutzerdefinierte Variante:



- die erste Komponente ist obligatorisch
- die *Verteilten Novell Druck-Services* müssen installiert werden, wenn man mit NDPS-Druckern die bidirektionalen Druckertreiber nutzen möchte
- die *IP-Gateway* braucht man fast nie (dient zur Umsetzung von IPX nach IP)
- der *Ziel-Service-Agent* muß installiert werden, wenn man mit dem neuen SBACKUP lokalen Laufwerke eines Arbeitsplatzrechners auf ein Bandlaufwerk auf einem Novell-Server sichern können möchte
- *Z.E.N.works Application Launcher NT Service* muß installiert werden, wenn man diese Technik auf der entsprechenden Station nutzen möchte.

In der Regel wird man also mindestens die in der Abbildung markierten Optionen wählen.

HINWEIS: Das Installationsfenster von Z.E.N.works erscheint nicht in der Taskliste und hat kein Systemmenü. Man kann deshalb die einzelnen Fenster nur - falls nötig - mit <Alt> <Esc> wechseln.

HINWEIS zur Migration

Das Objekt *NT Konfiguration* wird von Z.E.N. nicht mehr unterstützt und kann von dem neuen 32bit-Administrator nicht mehr bearbeitet werden. Man muß hierzu weiterhin den "alten" Administrator - der ja nicht gelöscht wird - verwenden. Workstation Manager und Z.E.N.-Client können jedoch parallel benutzt werden. Am besten geht man bei der Migration wie folgt vor:

1. Installation von Z.E.N.works
2. Einrichten von *Richtlinienpaketen (Policy Packages)*, den "Nachfolgern" der *NT Konfigurationen*
3. Updaten der Workstations auf den Z.E.N.-Client
4. Integration der Workstations in die NDS
5. Wenn alle Workstations integriert sind: Löschen der alten *NT Konfigurationen*
6. Migrieren der alten Applikationsobjekte (siehe dazu den Anhang 4: „Upgrading from NAL 2.x to Z.E.N. works 1.0“).

Z.E.N.'s Homepage im Internet

Novell's Internet Seite mit den neusten Informationen, "Tips of the Week" und einer ständig aktualisierten FAQ sowie der Möglichkeit, selbst Fragen zu stellen heißt:

<http://www.novell.com/cool solutions/zenworks/index.html>

HINWEIS zur Benutzung der HTML Online-Dokumentation

Die Online-**Dokumentation im Internet** ist gepackt in die vier Archivdateien ZSPDOCE1 bis ZSPDOCE4. Diese Archive müssen möglichst nahe an der Root (Default-Vorschlag: SYS:NOVDOC) entpackt werden, da die Verzeichnisse sehr tief strukturiert sind. Die Startseite INDEX.HTML befindet sich danach im Verzeichnis ..\ENGLISH\CLIENT\DOCUI.. Die **Dokumentation auf der CD** kann von der CD aus gestartet oder auf den Server kopiert und dann von dort aus benutzt werden.

Bei der Installation auf einem Server wird auch die Registry der Station um Eintragungen für die Java-Suchmaschine erweitert - dies muß man danach auf allen Stationen nachholen, auf denen die Suchmaschine gebraucht wird (Applikationsobjekt bei der Installation erstellen oder Aufruf des Programms SETUP im Verzeichnis \NOVDOC\INSTALL\KSERVER\. Das Quellverzeichnis für die Dokumentation lautet \NOVDOC. Danach muß der Rechner neu gebootet werden, bevor man die Suchmaschine benutzen kann).

ACHTUNG: beim Start von der CD aus muß man warten, bis die Registry wieder von temporären Eintragungen bereinigt wird - deswegen verschwindet der Installationsbildschirm beim Schließen zunächst nicht ... Auch hier gilt, dass einige Bildschirme nur durch <Alt> <Esc> erreicht werden können ...

Die Konfiguration des Novell Client auf der Workstation

Nach der Installation des Z.E.N.client auf einer Workstation gibt es unter den **Netzwerkdiensten** (rechte Maustaste auf *Netzwerkumgebung*, dann *Eigenschaften* oder *Start > Einstellungen > Systemsteuerung > Netzwerk*) drei neue Eintragungen:

- Den *Novell Client für Windows NT* (näheres dazu im Anschluss)
- Den *Arbeitsstationsmanager* (hier muß man den Baum einstellen, in dem die Richtlinienpakete liegen)
- die *Verteilten Novell Druckservices* (näheres dazu im Skript zu Novell 5)

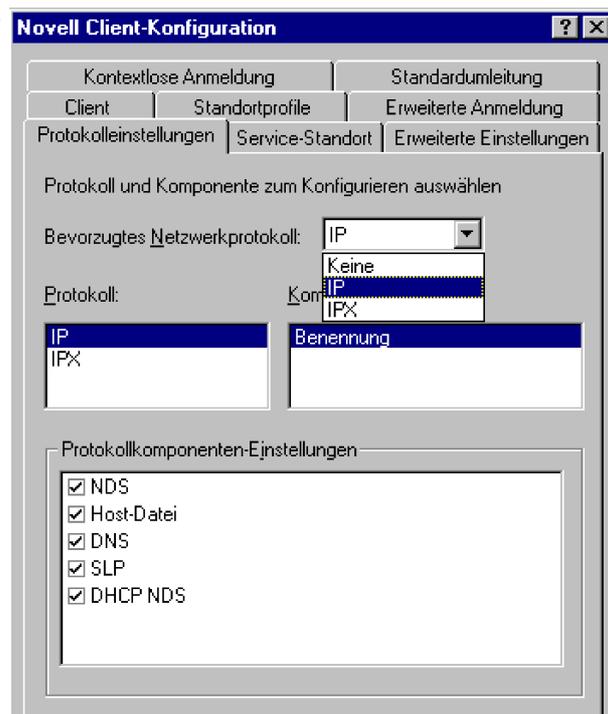
Die Konfiguration des *Novell Client* auf den Arbeitsstationen (das, was im Client 4.11 z.T. in der *NT Konfiguration* vorgenommen wurde) und die Einstellungen in den *Richtlinien (Policies)* kann **entweder lokal auf jedem Rechner** oder in einem **Arbeitsstationspaket (Workstation Policy Package)** vorgenommen werden. Diesem Paket werden danach Workstation(s) oder Zusammenfassung von Workstations (*Arbeitsstationsgruppen*) zugewiesen. Die Einstellmöglichkeiten sind in beiden Fällen völlig identisch. Näheres zu den Richtlinienpaketen ab Seite 10.



Einige Register aus der Client-Konfiguration

Protokolleinstellungen

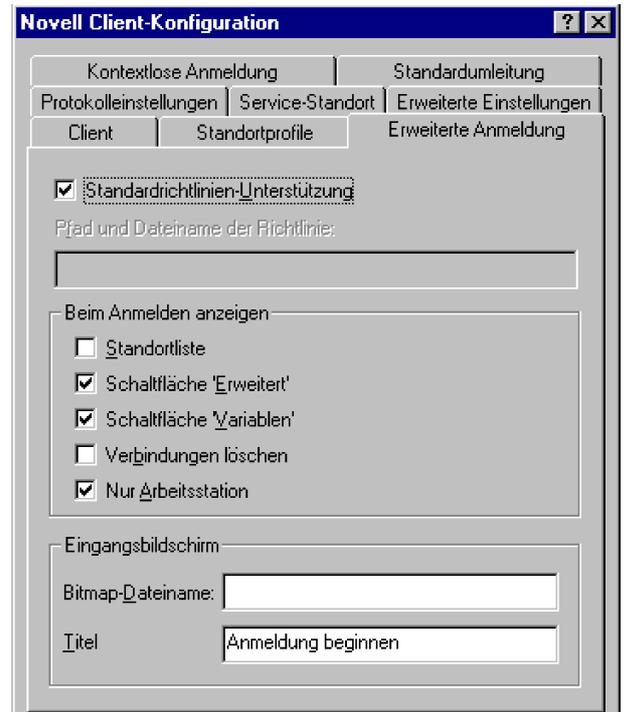
Hier wählt man das **bevorzugte Protokoll** (*IPX* oder *IP* oder *Keines* von beiden) und die **Dienste**, die das Protokoll nutzen soll. In einem gemischten Netzwerk mit 4.x und 5er-Servern stellt man am besten *IPX* (oder notfalls *Keines*) ein, weil sonst der Einlogg-Vorgang in einen 4er-Server sehr lange dauern kann (vorausgesetzt der 5er-Server nutzt auch das IPX-Protokoll). Benutzt man nur den 5er-Server, kann man natürlich *IP* wählen.



Erweiterte Anmeldung

Standardrichtlinien-Unterstützung ist per default aktiviert und sollte es bleiben - es sein denn man nutzt nicht die Workstation-Richtlinienpakete und möchte die Einstellung lokal für jeden Computer einzeln vornehmen.

- *Beim Anmelden anzeigen* erlaubt die Auswahl der Optionen im Anmeldefenster.
- Die *Schaltfläche Erweitert* ist nur notwendig, wenn sich Benutzer an einem Rechner in mehreren Bäumen und/oder Kontexten bewegen müssen - was in der Regel nicht vorkommt.
- *Schaltfläche Variablen* kann ausgeblendet bleiben, wenn man keine beim Anmelden eingeben muß. Dasselbe gilt für *Verbindung löschen* - es sei denn, Benutzer müssen sich in mehreren Bäumen **gleichzeitig** anmelden und daher die alte Verbindung beibehalten.
- *Nur Arbeitsstation* erlaubt einem Benutzer, sich nur lokal anzumelden und nicht im Novell-Netz. Wird man in der Regel ausblenden, weil z.B. bei einer nur lokalen Anmeldung keine Novell-Druckservices, Richtlinien und Applikationsobjekte zur Verfügung stehen. Steht das Netzwerk nicht zur Verfügung, wird dem Benutzer **in jedem Fall** angeboten, sich "behelfsweise" nur lokal anzumelden, auch wenn diese Option nicht aktiviert ist.



HINWEIS: Dies geht natürlich auch nur dann, wenn er nicht als temporärer Benutzer konfiguriert ist und ohne Netz gar nicht auf der Workstation existiert!

Die Kontextlose Anmeldung

... ist eine lange erwartete neue Eigenschaft, die nur zusammen mit einem LDAP-Katalog - einem neuen Objekt unter Novell 5 - funktioniert. Der Client sucht in verfügbar(en) Katalog(en) bis zum Erreichen des *Zeitüberschreitungsintervalls* nach einem Eintrag. Dabei kann sich der Benutzer sogar unter Angabe von *Platzhaltern* anmelden ("Hel*"). Anschließendes Drücken der Tabulatortaste zeigt alle passenden Benutzer in einem Fenster an.

Gibt es mehrere Kataloge in einem Baum oder liegt der Katalog auf einem anderen Server, trägt man zur Beschleunigung des Anmeldevorgangs im unteren Bereich den zu bevorzugenden Katalog ein. **ACHTUNG:** Der Katalog muss mit vollständigem Kontext angegeben werden (d.h. mit führendem Punkt).

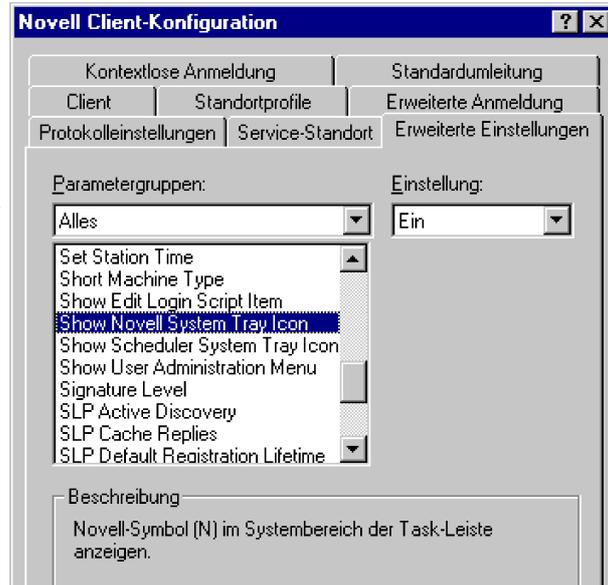


Erweiterte Einstellungen ...

Hier kann man u.a. einstellen, ob und wie die zwei neuen "Tasten" in der Taskleiste erscheinen sollen (Eigenschaften: *Show Novell System Tray Icon* - das **N** in der Abb. unten und *Show Scheduler System Tray Icon* - das Zeichen rechts daneben).

Die Eigenschaften *Show Edit Login Script Item* und *Show User Administration Menu* bestimmen, ob diese Menüpunkte in den Menüs Abb. 8 auftauchen (in der Abbildung sind sie aktiv).

Das **System Tray Icon** stellt die Funktionen zur Verfügung, die man nach Installation des "alten" *Workstation Managers* durch Klicken mit der rechten Maustaste auf Netzwerkumgebung vielleicht schon kennt und bietet zusätzlich einige neue Funktionen unter dem Menüpunkt *Benutzerverwaltung für <Baumname>* (in der Abbildung: *Benutzerverwaltung für BWB*).



Das **Scheduler System Tray Icon** dient zur Anzeige der Identifizierung der Workstation in der NDS (eine Workstation ist wie ein User in der NDS angemeldet) und zur Information und Definition der Aktionen des *Planers/Schedulers*, d.h. wann oder bei welchem Anlass bestimmte Aktionen/Programme gestartet werden. Soll der *Planer* benutzt werden, muß der *Workstation Manager* aktiviert sein (diese Aktivierung geschieht im *Arbeitsstations-Paket (Workstation Policy Package)* - siehe das Kapitel dazu.

Signature Level erlaubt, die Sicherheit der Kommunikation zwischen Server und Client einzustellen, indem man vorgibt, ob und wie Novell jedes Paket, das über das Netzwerk geht, signiert - näheres dazu in der Online-Dokumentation.

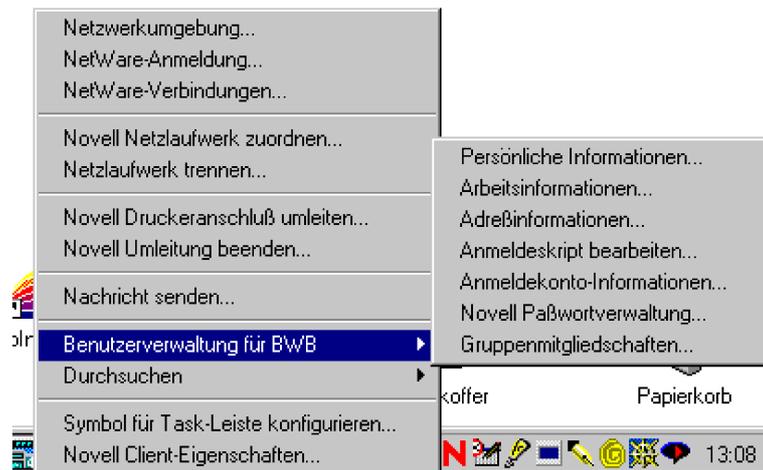


Abb. 8: die beiden neuen "Tasten" in der Taskleiste

Unter dem Register **Standardumstellungen**

... schließlich legt man die Novell-spezifischen Vorgaben für die Standard-Druckumleitung fest.

Neue NDS-Objekte

Wie am Anfang erwähnt stehen nach der Installation von Z.E.N.-Works neue Objekte zur Verfügung - **unabhängig davon, ob man Z.E.N.works auf einem 4.x- oder 5-er Server installiert:**

1. Das Objekt *Arbeitsstation (Workstation)*

Dieses Objekt kann nicht manuell erzeugt werden, sondern nur durch Z.E.N.-Hilfsprogramme. Der Server liest dabei (mit Hilfe des Z.E.N-Client) bestimmte Informationen aus der Workstation aus und speichert sie in der NDS. Der Umfang der ausgelesenen Daten unterscheidet sich deutlich bei beiden Versionen - dem Starter Pack und der Vollversion. Im *Workstation Object* werden (unter 32bit) auf jeden Fall der PC-Name (DN = *Distinguished Name*, dessen Zusammensetzung man vorher festlegen muß), der Operator des Objekts, eine Historie aller Benutzer und das Datum der letzten Registrierung festgehalten. Das Objekt wird bei jedem Lauf des Registrierungsprogramms mit den neuesten Angaben zu Registrierungszeit, Netzwerkadresse, Server und Benutzer ergänzt. Standardmäßig wird dieses Programm durch den *Planer (Scheduler - s.u.)* ausgeführt, bei dem man Zeitpunkt oder Anlass der Ausführung wählen kann - z.B. einmal täglich/wöchentlich oder z.B. bei jedem Login.

Erzeugung (siehe auch die Hinweise in Anhang 3):

1. Zunächst muß der Baum für die Aufnahme der *Workstation Objects* vorbereitet werden. Bei der Installation von Z.E.N. geschieht das automatisch mit allen vorhandenen Containern. Werden neue hinzugefügt, muß dieser Vorgang manuell ausgelöst werden. Vorgehensweise: im Administrator unter *<Werkzeuge> Arbeitsstations-Dienstprogramme > Arbeitsstations-Registrierung vorbereiten* un dort den Container auswählen, in dem die Objekte platziert werden sollen.
2. Anlegen einer *Arbeitsstationsimport-Richtlinie (Workstation Import Policy)* - sie ist Bestandteil des *Benutzer-Richtlinienpaketes (User Policy Packages)*. Danach müssen alle Benutzer, die PCs registrieren können sollen, diesem Benutzerpakete (dieser Policy) zugewiesen werden. In der Policy wird unter *Standort der Arbeitsstation (Workstation Location)* festgelegt, in welchem Container die Objekte für die Workstations landen und unter *Benennung der Arbeitsstation (Workstation Naming)*, aus welchen Bestandteilen (PC-Name, IPX/IP-Adresse, Benutzer, Container usw.) der Objektname zusammengesetzt wird. Zugewiesene Benutzer können die Workstation **nur registrieren, nicht importieren** - letzteres erfordert das Objektrecht *Erstellen* im entsprechenden Container!
3. Aufruf von WSREG32 (SYS:PUBLIC\): Die Workstation wird dadurch zum Import **registriert**. (Im Baum ist danach keine Veränderung festzustellen.)
4. Aufruf von WSIMPORT (SYS:PUBLIC\WIN32): Die Workstation wird in den Baum **importiert** (siehe Anmerkung unter 2). Der Import kann auch im Administrationsprogramm unter *Werkzeuge > Arbeitsstationen importieren* erfolgen. Nach dem Import gibt es ein *Workstation Objekt* im unter 2) eingestellten Container.
5. Erneuter Aufruf von WSREG32: Die Workstation gibt Informationen "über sich" an das ihr zugehörige Objekt in der NDS weiter - die NDS "kennt" jetzt die Workstation. Bei jedem weiteren Aufruf von WSREG32 wird das *Arbeitsstations-Objekt* in der NDS aktualisiert.

HINWEISE: Die Workstation unterhält von da ab nach jedem Booten eine eigene Verbindung (Connection) zur NDS, die allerdings **keine Lizenz „verbraucht“** - es sei denn, „ihr“ Scheduler kopiert gerade eine Datei vom Netz (deswegen am besten Nachtermine zum Scheduling benutzen). Durch diese Verbindung können z.B. (durch den *Planer (Scheduler)*) Installationen auf ihr vorgenommen werden, ohne dass jemand eingeloggt ist.

Die Punkte 3-5 müssen mit **jeder** Workstation durchgeführt werden. Danach erfolgt deren weitere Verwaltung zentral - Punkt 3 kann durch ein Applikationsobjekt erfolgen, Punkt 5 erfolgt automatisch durch den Planer.

2. Die Arbeitsstationsgruppe (Workstation Group)

Dieses Objekt erlaubt, mehrere Workstation-Objekte zusammenzufassen und dieser Gruppe ein *Arbeitsstationspaket (Policy Package)* zuzuweisen (ähnlich, wie man über Benutzergruppen einer Reihe von Benutzern die gleichen Eigenschaften zuweisen kann).

3. Richtlinienpakete (Policy Packages)

... sind - neben den *Application objects* - der zentrale Platz in der NDS, wo man sozusagen die "Arbeitsersparnis definiert". Einige Zitate aus der Online-Dokumentation:

Richtlinienpakete sind NDS-Objekte, die Sammlungen von Richtlinien darstellen, die nach dem Typ der Objekte gruppiert sind, mit denen die Richtlinien verknüpft werden können. Verwalten Sie Arbeitsstationsobjekte und andere mit Arbeitsstationen verknüpfte NDS*-Objekte mit Hilfe von Richtlinienpaketen und NetWare-Administrator.

Es gibt drei grundsätzliche Typen von Richtlinienpaketen: Container, Benutzer und Arbeitsstation. Da jedoch die Benutzer- und Arbeitsstationspakete plattformspezifisch sind, stehen eigentlich sieben Richtlinienpakete zur Auswahl.

Hinweis: Informationen zur Anzeige der tatsächlichen Richtlinienlisten für jedes Richtlinienpaket finden Sie unter "Die sieben Richtlinienpakete" in dieser Hilfe.

Die Richtlinienpakete sind auf die Richtlinien begrenzt, die mit jeder Version von Z.E.N.works* ausgeliefert werden. Weitere Informationen zu Richtlinien finden Sie unter Richtlinien.

Sie müssen Richtlinienpakete erstellen und verwenden, um die gesamte Funktionalität von Z.E.N.works ausnutzen zu können. Wenn Sie ein Richtlinienpaket erstellt und eine Richtlinie aktiviert haben, können Sie die anfänglichen Einstellungen für diese Richtlinie anzeigen lassen.

Als Netzwerkverwalter müssen Sie ermitteln, welche Richtlinienpakete erstellt und mit welchem Objekt sie verknüpft werden müssen. Weitere Anregungen für die Verwendung von Richtlinienpaketen finden Sie unter Richtlinienpaket-Szenarios, eine Liste der anfänglichen Einstellungen der entsprechenden Richtlinien unter Richtlinienbeispiele.

Wirksame (effektive) Richtlinien

Was sind wirksame Richtlinien?

Wenn Sie mit effektiven Rechten in NDS* vertraut sind, verstehen Sie bereits das Prinzip der wirksamen Richtlinien in Z.E.N.works. Unter den wirksamen Richtlinien versteht man die Summe aller aktivierten Richtlinien aller Richtlinienpakete, die direkt oder indirekt mit einem Objekt verknüpft sind. Genauso wie die effektiven Rechte in NDS im Baum nach unten fließen, fließen auch Verknüpfungen der Richtlinienpakete im Baum nach unten, außer wenn eine eindeutige Verknüpfung eines Objekts mit einem Richtlinienpaket besteht. Das System beginnt die Berechnung der wirksamen Richtlinien für ein Objekt bei allen Richtlinienpaketen, die mit diesem Objekt verknüpft sind, und durchsucht dann den Baum aufwärts (unter der Voraussetzung, dass die Suchfolge an den Blattobjekten beginnt und aufwärts zum Stamm des Baums geht) nach Verknüpfungen, die mit übergeordneten Containern hergestellt wurden. Die erste angetroffene aktivierte Richtlinie gilt, genauso wie das System den Baum aufwärts nach effektiven Rechten absucht.

Einzelheiten

Richtlinienpakete gibt es in 2 Varianten für 3 Plattformen und zusätzlich die Container Policy, die für alle Plattformen gilt, insgesamt also 7 Varianten:

- Container
- Windows 3.1 Benutzer und Windows 3.1 Arbeitsstation
- Windows 95 Benutzer und Windows 95 Arbeitsstation
- Windows NT Benutzer und Windows NT Arbeitsstation

Richtlinien werden zu bestimmten **Zeitpunkten** oder bei bestimmten **Ereignissen** von der NDS an die Workstation geschickt. Dazu gibt es bei jeder Richtlinie einen **Zeitplan** (im unteren Fensterbereich) mit einer Standardvorgabe, die sich aber besonderen Wünschen anpassen läßt. Näheres zu diesen Zeitplänen (zum "Schedulen") bei den Applikationsobjekten - in der Regel kann man in den Richtlinienpaketen die Standardeinstellungen übernehmen.

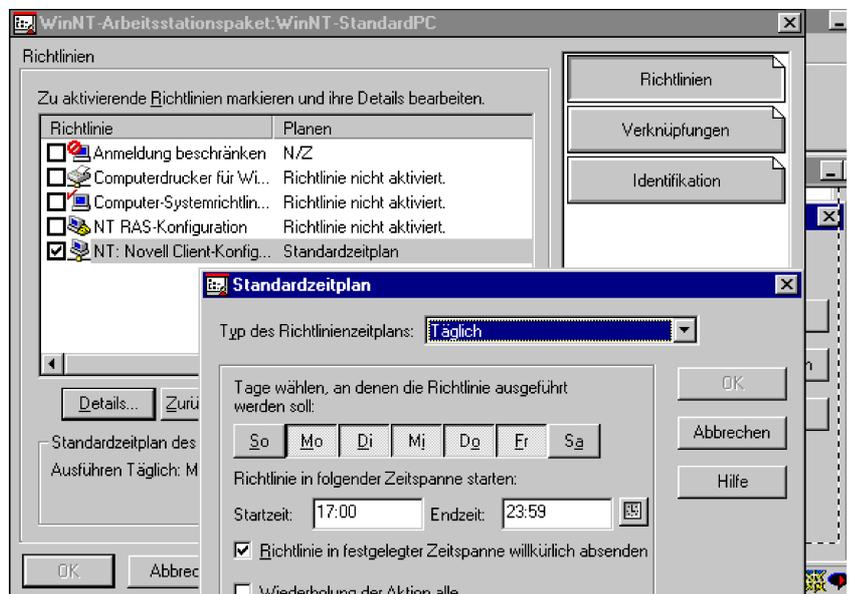
Nach dem Erstellen eines Paketes stehen je nach Typ des Betriebssystems unterschiedliche "Einzel"-Richtlinien zur Verfügung. Im folgenden gehe ich nur auf die Pakete für Windows NT ein. (Die Pakete für Windows 95 sind sehr ähnlich, bei Windows 3.1 gibt es natürlich betriebssystem-bedingt wesentlich weniger Einflussnahme). Die Pakete für Workstations- und Benutzer enthalten in einigen Punkten dieselben Richtlinien. So kann man z.B. Drucker personen- oder workstation-bezogen zuweisen, je nachdem, welche Vorgehensweise gewünscht ist. Arbeiten mehrere Personen an einem PC und alle sollen an diesem PC die gleichen Drucker zur Verfügung haben, weist man die Drucker PC-abhängig zu. Soll ein Benutzer überall im Netzwerk die gleichen Drucker zur Verfügung haben, weist man Sie personenbezogen (oder im *Login Skript*) zu.

Die *Richtlinien-Pakete* sind im mit Z.E.N. ausgelieferten Administrator nur als komplettes Paket sichtbar, in den anderen Administratoren sind alle Einzel-Richtlinien eines Paketes im Baum dargestellt - der Baum wird dadurch verwirrend unübersichtlich mit den alten Admin-Versionen.

Das WINNT Arbeitsstationspaket

Ein Arbeitsstationspaket besteht - wie der Name sagt - aus einer Ansammlung einzelner Richtlinien (Policies), die im Baum jedoch nur als ein Objekt auftauchen. Aktiviert wird eine Richtlinie durch Anklicken des Kästchens links neben dem Namen, konfiguriert durch Doppelklick auf den Namen.

HINWEIS: Durch Zuweisung (Taste *Verknüpfung*) eines *Richtlinienpaketes* zu einem *Arbeitsstations-Objekt* kann u.a. die komplette Konfiguration des Clients auf einer (oder einer Gruppe von) Station(en) „ferngesteuert“ werden.



Zu den Einzelrichtlinien:

- unter *Novell Client Configuration* wird der Client konfiguriert. Die **Einstellungen hier überschreiben die lokal vorgenommenen**. Sie sind nicht dynamisch sondern werden gemäß dem Zeitplan auf den jeweiligen PC (in die Registry) kopiert. Beim Klicken auf diese Taste erscheint zunächst ein Baum mit den Angaben *Client*, *Arbeitsstationsmanager* usw. Doppelklick auf *Client* öffnet das nebenstehende Fenster.

Die nebenstehenden Registerkarten sind im vorigen Kapitel beschrieben.



- unter *Computerdrucker für Windows NT* werden dem PC ein oder mehrere Drucker zugewiesen. Diese Drucker können entweder für Benutzer oder für Workstations definiert werden - je nachdem, ob alle Benutzer an einem PC immer dieselben Drucker oder ein Benutzer an allen PC's seine gewohnten Drucker vorfinden soll. **Diese Richtlinie ist nur für Warteschlangen-basierte Druckdienste interessant - die neuen NDPS-Drucker haben eigene Zuweisungsmechanismen.**

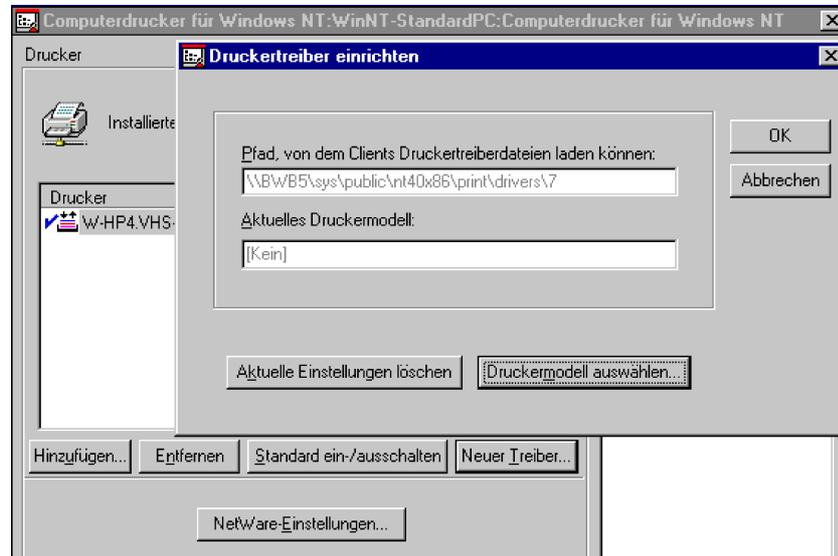


Abb. 11: Die Computerdrucker-Richtlinie

Diese Policy bringt eine wesentliche Vereinfachung bei der Druckerinstallation auf den einzelnen Workstations: wurde einer Novell *Druckwarteschlange* ein Druckertreiber zugewiesen (dies muß nur einmal geschehen und wird in der NDS festgehalten), so installiert der *Novell Client* diesen Drucker beim ersten Ausdruck in die entsprechende Warteschlange automatisch auf der Workstation! Sie greift dabei auf Druckertreiber zu, die im Netzwerk liegen (standardmäßig auf das Verzeichnis SYS:PUBLIC\NT40x86\PRINT\DRIVERS\<SPRACHE>, wobei Englisch den Code 7 hat).

Lokal installierte und nicht installierte Drucker tauchen **nach dem Einloggen** ganz "normal" unter *Start > Einstellungen > Drucker* auf - der Benutzer merkt also nicht, welche von ihnen tatsächlich lokal installiert sind. Lokale Drucker haben das normale Freigabe-Symbol (in der Abb. der Canon-Drucker), Netzwerkdrucker erscheinen an das Netzkabel angeschlossen (die beiden anderen definierten Drucker).



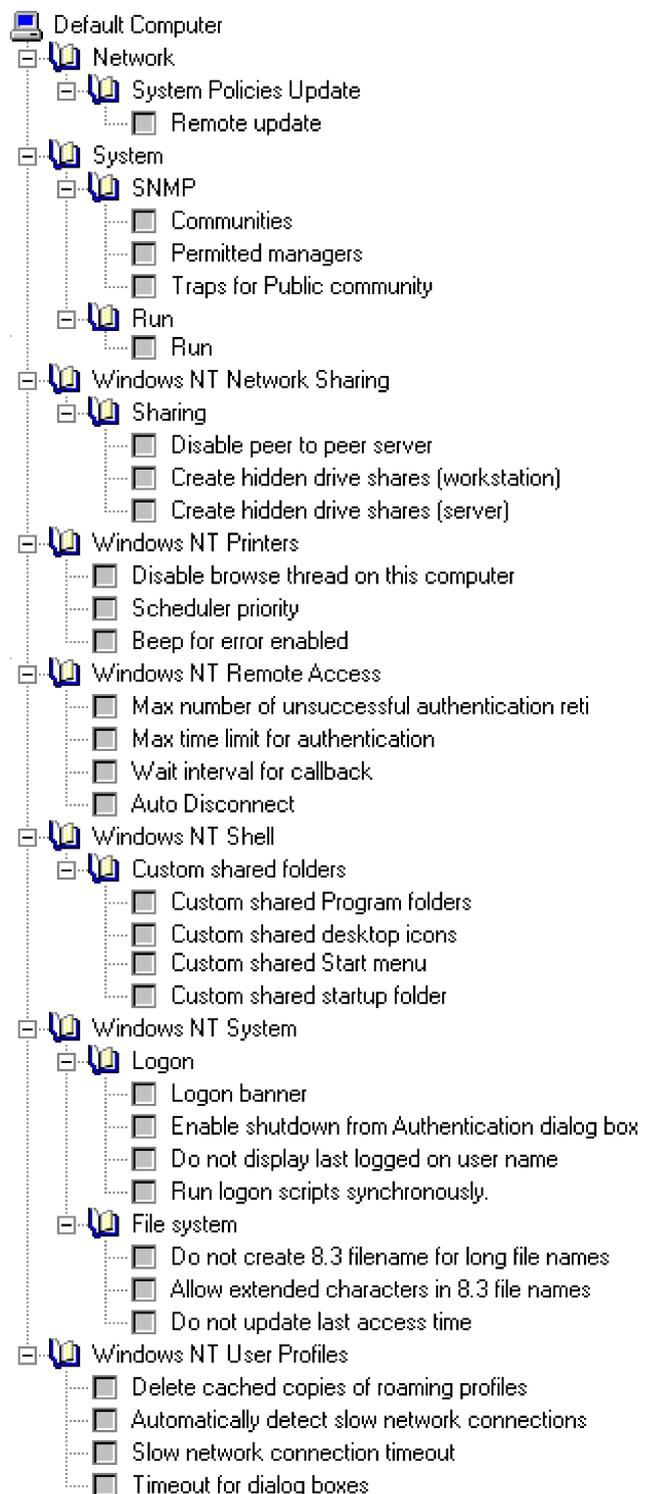
Voraussetzung für diese Technik ist das Kopieren der Druckertreiber ins Netzwerk!

siehe dazu Abb. 11. Vorgehensweise (die Windows NT-Installations CD muß hierzu evtl. zur Verfügung stehen):

- Definieren der *Druckwarteschlange* - wie gewohnt mit dem Administrator (falls nicht schon vorhanden - dies ist Voraussetzung für die folgenden Schritte)
- in der Workstation- (oder Benutzer-) *Computerdrucker* anklicken
 - Doppelklick auf die Warteschlange, der ein Treiber zugewiesen werden soll
 - Auswahl von *neuer Treiber*
 - Auswahl von *Druckermodell auswählen*
 - Auswahl eines Druckes aus der dann erscheinenden Liste und Kopieren des Treibers ins Netz.

Danach taucht der Treibername unter *Aktuelles Druckermodell* auf. Erst bei einem Update des Treibers muß dieser Vorgang wiederholt werden. Nach Anklicken von *NetWare Einstellungen* können NetWare-spezifische Einstellungen wie *Banner* und *Formularvorschub* u.ä. ein- bzw. ausgeschaltet werden.

- unter *Computer System Richtlinien für Windows NT* können die System-Policies für den entsprechenden Computer eingestellt werden. In der nebenstehenden Abbildung sind alle Eigenschaften zu sehen. (Sie werden inzwischen auch in deutsch angezeigt.)



- unter *NT RAS Konfiguration* der Fernzugriff auf die Station konfiguriert
- unter *Anmeldung beschränken* Einschränkungen festgelegt, falls sich nicht alle Personen auf einer Station einloggen können sollen.

Das WINNT Benutzer Paket

... ähnelt in einigen Unterpunkten (Richtlinie *Printer* und *System*) dem Workstation Paket, hat aber zusätzlich die Richtlinien zum *Dynamischen lokalen Benutzer* und zu den *Desktop-Standardereinstellungen*. Die Drucker-Richtlinien stimmen vollständig überein, bei den System-Richtlinien gibt es einige Unterschiede. Zu den einzelnen Komponenten des Pakets:

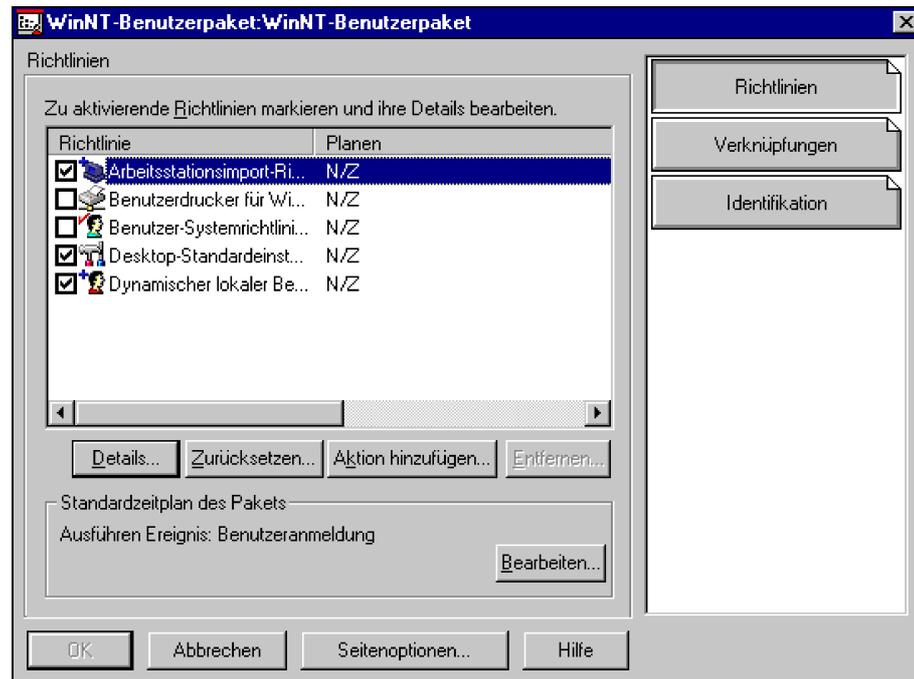


Abb. 14: Das WINNT-Benutzerpaket

- Unter *Dynamischer lokaler Benutzer* findet sich ein Menü, das aus dem *Arbeitsstations-Manager (Workstation manager)* bekannt sein dürfte - es fehlen jedoch (leider) die Einstellmöglichkeiten zum Login-Fenster, die jetzt nicht mehr benutzer-, sondern stationsbezogen vergeben werden (im Anmeldefenster selbst - siehe Seite 22). Um die folgenden Techniken nutzen zu können, muß *Dynamischen lokalen Benutzer aktivieren* aktiviert sein!

Erläuterung zum *Dynamischen lokalen Benutzer*

- *Bestehendes NT-Konto verwalten*: Bei dieser Einstellung wird der Benutzer nur lokal erzeugt, wenn er dort nicht schon existiert. Gibt es ihn lokal, werden die dortigen Einstellungen (z.B. seine Rechte auf der NT-Station) verwendet und er bleibt nach dem Abmelden lokal bestehen. Das ursprüngliche (lokale) Passwort wird bei der ersten Anmeldung (nach Vorankündigung) überschrieben.
- *Temporärer Benutzer*: Ist diese Eigenschaft aktiviert, müssen Benutzer, die eine Workstation benutzen wollen, nur mehr in der NDS angelegt werden. Loggt sich ein Benutzer ein, wird er zunächst unter Novell beglaubigt. Danach wird er auf der Workstation temporär (d.h. für die aktuelle Sitzung) als Benutzer erzeugt, sein Profil aus seinem Heimatverzeichnis auf die Station kopiert und er kann ohne lokal eingerichtet zu sein, „normal“ arbeiten. Beim Abmelden wird er lokal zusammen mit seinem Profil wieder gelöscht.

- *NetWare-Referenzen verwenden*: unter NT und Novell soll die gleiche User/Passwort-Kombination gelten. Falls das NT-Passwort beim ersten Einloggen noch anders lautet, kann es synchronisiert werden.
- *Mitglied von / nicht Mitglied von*: hier stellt man ein, welche Rechte der dynamische Benutzer lokal auf der NT-Station hat.

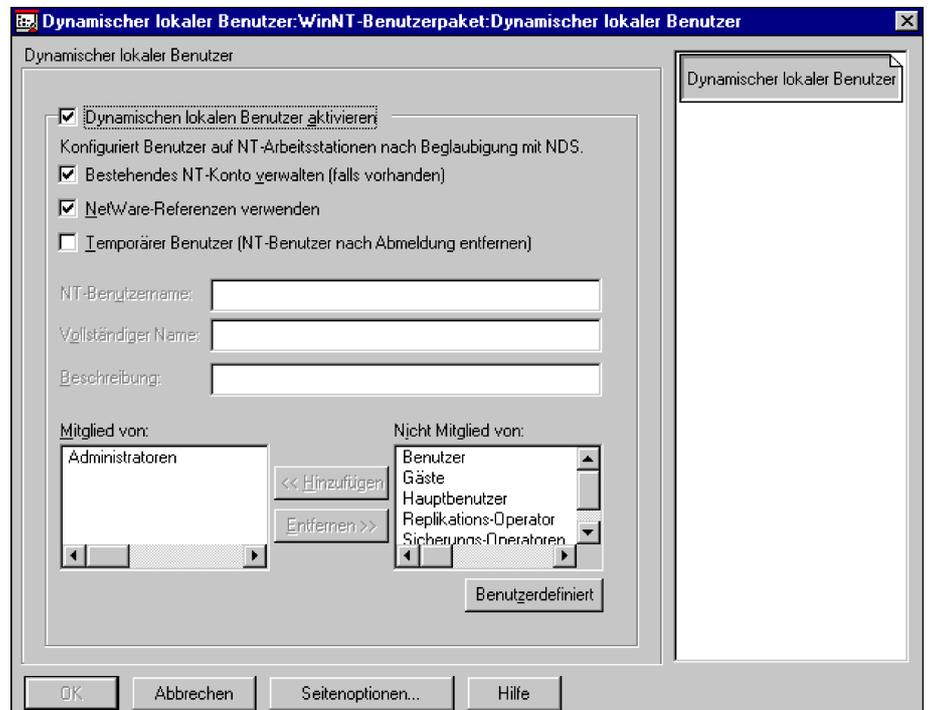


Abb. 15: Richtlinie *Dynamischer lokaler Benutzer*

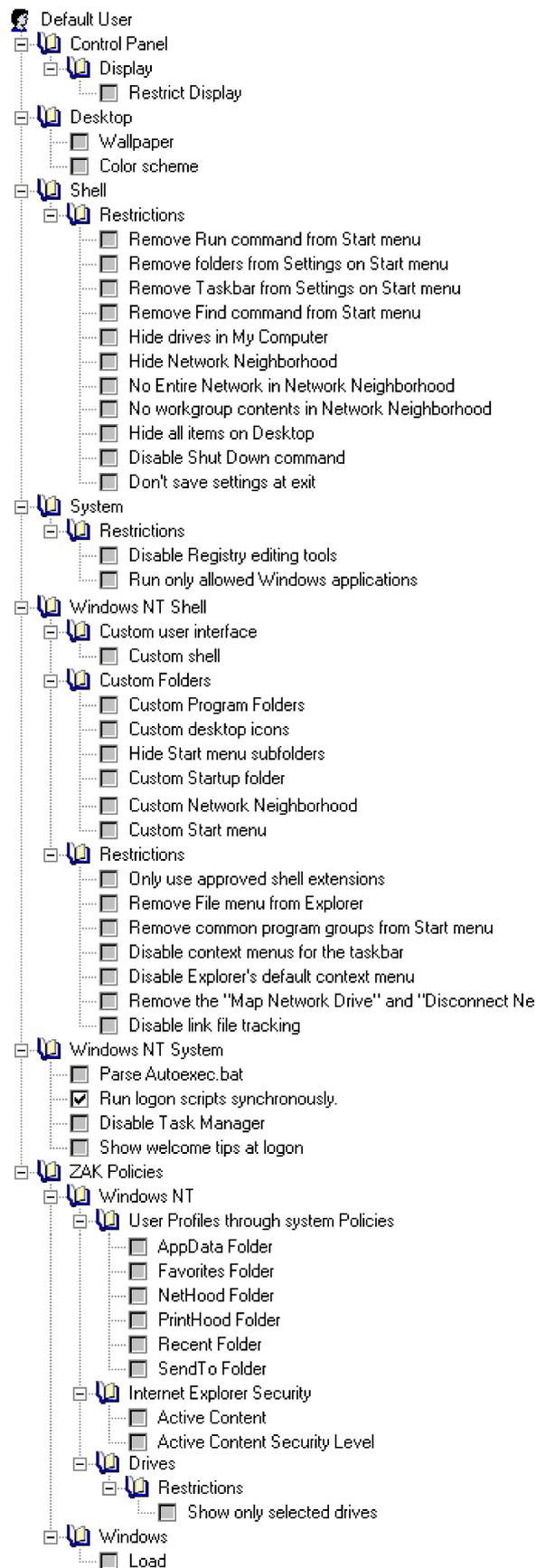
- *Desktop Standardeinstellungen* erlaubt Einstellungen zu den Punkten, die auch unter den Eigenschaften des Desktops zur Verfügung stehen: Maus, Bildschirmattribute usw. Außerdem werden hier die (früher im *NT Konfigurationen* platzierten) Einstellungen zum *Zentral gespeicherten Profil (Roaming Profiles)* und ihrer Ablage auf einem Server vorgenommen - schließlich steuern diese ja den Desktop! **HINWEIS**: Die Dateien für Mauszeiger, Hintergründe u.ä. müssen auf einem Server liegen - sonst findet sie der Benutzer evtl. nicht überall vor!

Erläuterung zu *Zentral gespeicherten Profilen*

Unter *Zentral gespeicherten Profilen (Roaming Profiles)* versteht man Profile, die dem Benutzer „hinterherlaufen“, egal von welcher Station aus er sich einloggt. Die Profile werden bei jedem Anmeldevorgang auf den lokalen NT-Rechner kopiert und beim Abmelden mit den während der Sitzung evtl. vorgenommenen Veränderungen wieder auf den Server zurückkopiert. Standardmäßig sollten sie im Heimatverzeichnis des Benutzers abgelegt werden. Das entsprechende Volume muß lange Dateinamen verwalten können (der *name space LONG* muß auf dem Server geladen und das Volume mit *ADD NAME SPACE LONG TO <Volume-Name>* mit dem Name Space versehen worden sein).

HINWEIS zur Performance: Je mehr Daten im Profil stehen, desto länger dauert der Vorgang des Ein- und Ausloggens. Verzeichnisse wie *Eigene Dateien* und *Vorlagen* sind Bereiche, in die NT-Programme gerne Benutzerdateien schreiben, die eigentlich in Daten- bzw. Heimatverzeichnisse gehören und nicht ständig beim Einloggen hin- und herkopiert werden sollten. Man muß hier bei der Programmkonfiguration und besonders bei der Zuweisung der Datenverzeichnisse meist die Programm-Voreinstellungen ändern!

- die Benutzer Systemrichtlinien erlauben, die System policies zu den Benutzern einzustellen, für die dieses Paket gilt. Die einzelnen Eigenschaften sind in der nebenstehenden Abbildung zu sehen (inzwischen in Deutsch).



- Benutzerdrucker für Windows NT ist identisch mit der entsprechenden Policy zum Workstation Package
- Arbeitsstationsimport-Richtlinie bietet die Einstellung, wer wohin welche Workstations in die NDS importieren darf und welche Namenszusammensetzung diese Objekte haben sollen.

ANMERKUNG

Die Entfernung der Profile und Benutzer von den Workstations funktioniert nicht immer einwandfrei. Profile zu einem Benutzer existieren dann mehrfach, sie bekommen die Endungen .001, 002 usw. Oft bringt die im folgenden beschriebene Maßnahme eine Abhilfe. Die dort beschriebene Änderung der Registrierung kann man entweder "lokal" an jedem PC mit dem Programm POLEDIT oder netzweit automatisch durch ein WINNT Arbeitsstationspaket vornehmen.

Zwei Einstellungen in der lokalen Registrierung

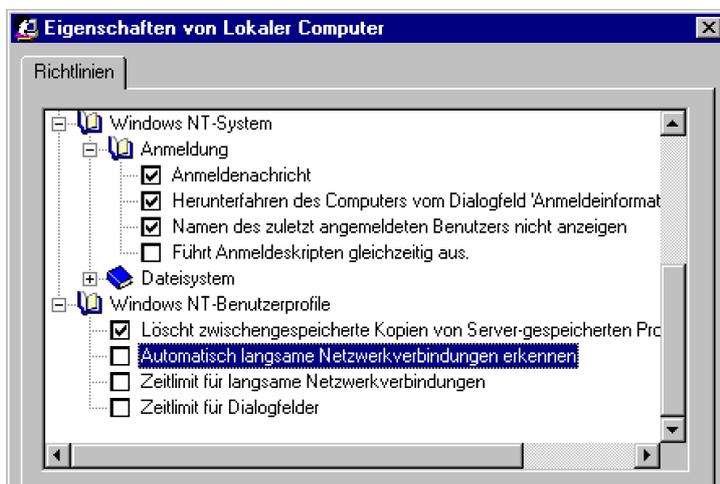
(Die Abbildung stammt aus dem Policy-Editor, in der Richtlinie findet man gleichlautende Schlüssel.)

Windows NT Benutzerprofile

- aktiviert man die erste Option, werden Serverbasierte Profile beim Abmelden lokal gelöscht. Der Benutzer hat dann bei einer **lokalen** Anmeldung kein Profil. Dafür geht die Anmeldung schneller. Gibt es die in der obigen Anmerkung beschriebenen Schwierigkeiten im Zusammenspiel zwischen lokalen Kopien und dem Server-Profil und arbeitet der Anwender an dieser Station immer im Netz, sollte diese Option aktiviert werden.
- Aktivieren von *Automatisch langsame Netzwerkverbindungen erkennen* führt dazu, dass bei "langsamem" Zugriff auf den Server (was langsam heißt, definiert man unter *Zeitlimit für langsame Netzwerkverbindungen*) nach Ablauf der Zeit auf jeden Fall - falls vorhanden - die lokal zwischengespeicherte Profilkopie genommen wird. Falls Benutzer häufig das Profil verändern (also z.B. auch Programm-Standardvorgaben) und die Station wechseln, ist dies natürlich nicht sinnvoll. Andererseits beschleunigt es den Anmeldevorgang.

Anmeldung

- Arbeiten mehrere Benutzer an einer Station, aktiviert man am besten die dritte Option. Bei einer Neuansmeldung steht der Cursor dann direkt im Namensfeld und man muß nicht erst den "alten" Namen löschen.



Auch danach sollte man regelmäßig kontrollieren, ob lokal nicht mehrere Profile zu einem Benutzer vorliegen (<Name>.001, <Name>.002 usw.). Diese Profile kann man unter *Start > Einstellungen > Systemsteuerung > System > Benutzerprofile* löschen. "Hängengebliebene" Benutzer löscht man im Benutzer-Manager (*Start > Programme > Verwaltung (Allgemein) > Benutzer-Manager*).

TIPP: Danach kontrollieren, ob evtl. noch weitere Exemplare im Explorer gelöscht werden müssen. **Vor allem bei Programminstallationen können merkwürdige "Erscheinungen" auftreten, wenn mehrere Profile existieren**, da im Profil die benutzerspezifischen Registrierungseinträge stehen (NTUSER.DAT).

4. Die Suchrichtlinie

... dient zur Festlegung, wie weit **Benutzer oder Workstations** nach Richtlinien, die für sie „zuständig sind“, im NDS Baum „nach oben“ suchen, **wenn ihnen keine direkt zugeordnet ist**. Man kann in der *Suchrichtlinie* einen Container angeben (*Ausgewählter Container*), die Anzahl der Stufen oder auch festlegen, dass nicht über den *Objektcontainer*, das ist der Container, in dem das Benutzer-/Workstation Objekt liegt, hinaus gesucht wird. Näheres in der Online-Hilfe. Man sollte darauf achten, dass nie nach Richtlinien im WAN gesucht wird, weil dies den Login-Vorgang stark verlangsamen kann. Das *Containerpaket* sollte im höchsten Level des Baumes platziert werden, in dem es gebraucht wird, also oberhalb aller Container, die assoziiert sind und unterhalb einem Level, der einen Container/eine Partition im WAN repräsentiert. Näheres im Artikel „NDS guidelines for Z.E.N.works and NDS for Windows NT“ unter <http://support.novell.com>.

WICHTIG: Wurde für einen Benutzer keine Richtlinie per Benutzer/Gruppe/Container zugewiesen, so gibt es zwei Szenarios:

1. Es **gibt** eine Suchrichtlinie **für den Container des Benutzers**. In diesem Fall sucht Novell bis zum in der Suchrichtlinie angegebenen Container nach einer Richtlinie. Gibt es eine passende (also eine WINNT-Benutzerrichtlinie), wird diese genommen.
2. Es **gibt keine** Suchrichtlinie für den Container des Benutzers: In diesem Fall gilt keine Richtlinie für alle Benutzer des Containers.

Man kann die geltenden Richtlinien für einen Benutzer ermitteln durch Klicken auf den Button *Wirksame Richtlinien* im Eigenschaftsfenster *Wirksame Richtlinien*.

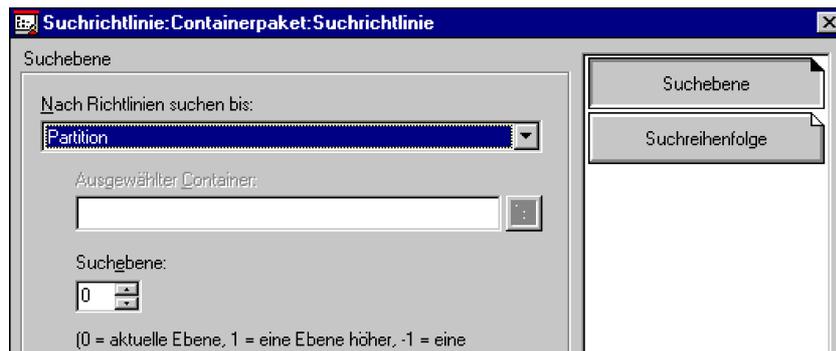
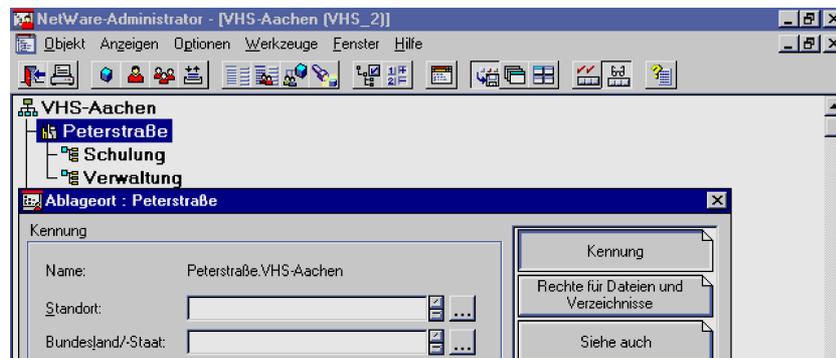


Abb. 18: Die Suchrichtlinie im Containerpaket

5. Standort-Objekte

... (oder auch *Locality-Objekte*) bieten eine neue (zusätzliche) Möglichkeit, den NDS-Baum zu strukturieren: Sie können unterhalb eines beliebigen Containers (*Land, Organisation, Organisatorische Einheit*) angeordnet werden und selber nur weitere Container und *Alias*-Objekte enthalten.



6. Anwendungsordner

... ermöglichen jetzt eine Strukturierung der Applikationsobjekte im Startmenü - siehe dazu die Ausführungen im Kapitel zum Application Launcher.

7. License Container (nur Vollprodukt)

... die Objekte, mit deren Hilfe die *Novell Licensing-Services*, d.h. die Lizenzüberwachung, gesteuert wird. Hier stehen ab Novell 5 auch die Server- und Benutzerlizenzen von Novell selbst.

Der Planer

... (oder *Scheduler*) bietet die Möglichkeit, beliebige Programme oder Module (*.EXE, *.COM, *.DLL und Java-Applikationen) zeit- oder ereignisgesteuert zu starten. Es gibt ihn in zwei Varianten: NDS-basiert und Workstation-basiert. Im ersten Fall wird er einem *Richtlinienpaket* zugeordnet und gilt dann für die Benutzer/Gruppen/Container, die diesem Paket assoziiert sind. Im zweiten Fall wird er einer Workstation zugeordnet und gilt nur für diese Workstation.

Der *Planer* wird durch Klicken mit der rechten Maustaste auf das zweite Icon in der Taskleiste aus Abb. 8 und anschließende Auswahl von *Planer anzeigen* aufgerufen. **Bitte den Hinweis auf Seite 7 beachten!**

Wenn Workstations in die NDS integriert werden, wird automatisch ein Ereignis in den Planer eingetragen:

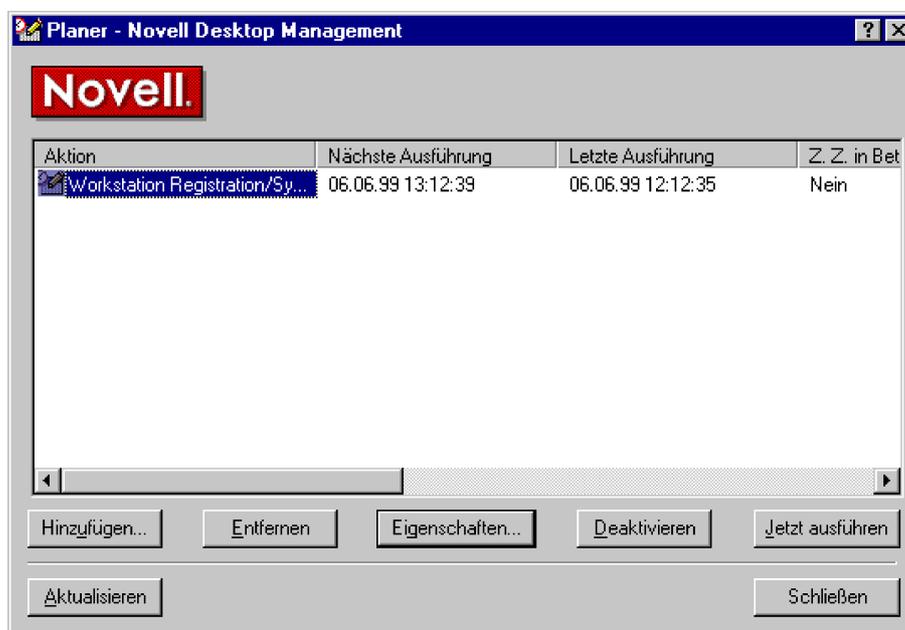


Abb. 20: Der „lokale“ Scheduler

HINWEIS zu Abb. 20: Der Eintrag aktualisiert das Workstation Object in der NDS. Alle Aktionen können nach Anklicken mit *Jetzt Ausführen* auch sofort gestartet werden.

Erstellen einer NDS-basierten Aktion

Dazu wählt man im Administrator das Richtlinienpaket, für deren „Mitglieder“ die Aktion gelten soll und dort *Aktion hinzufügen* und gibt der Aktion einen Namen.

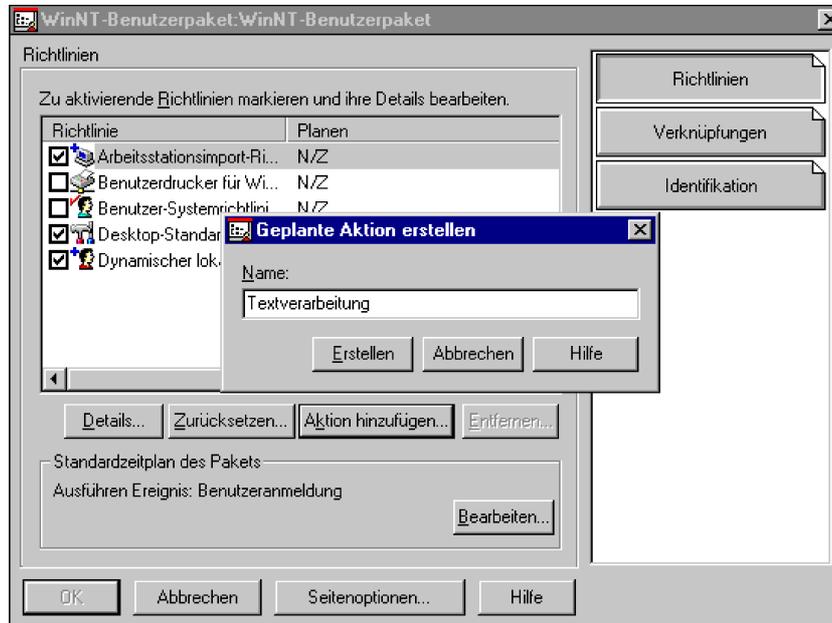


Abb. 21: Erzeugen einer neuen Aktion

Durch Klicken auf Details öffnet sich das Fenster in Abb. 22, in der man die Details zur Ausführung der Aktion eingeben kann:

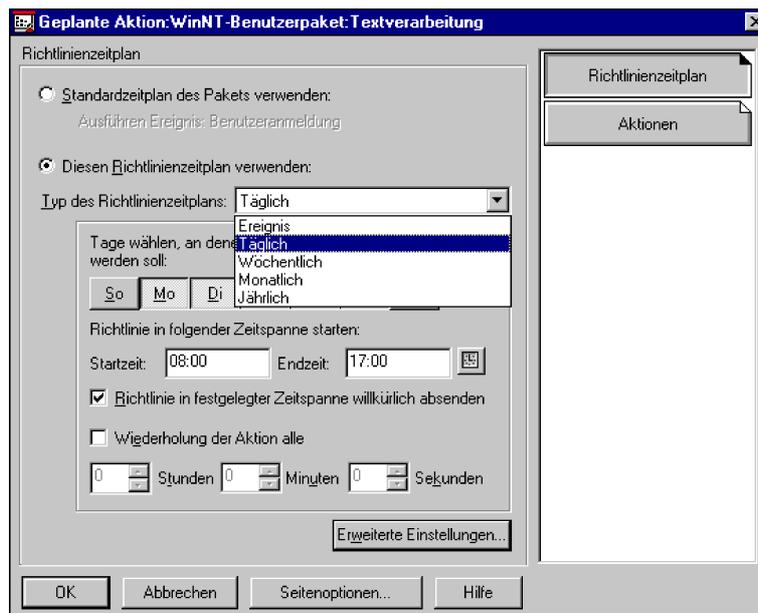
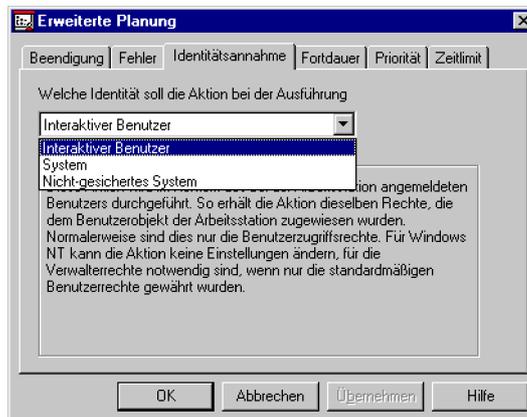


Abb. 22: Auslöser für den Start der Aktion

Unter *Erweiterte Einstellungen* wird definiert, wie der Start und die Beendigung der Aktion laufen soll (Informationen sind oft im unteren Fensterbereich zu sehen, nachdem man die entsprechende Auswahl getroffen hat).

Aktionen können auf verschiedene Weise, u.a. auch im Hintergrund durch das System, durchgeführt werden. Diese Auswahlmöglichkeiten befinden sich unter dem Register *Identitätsannahme*. Hier bedeuten im einzelnen (diese Angaben stehen je nach Auswahl auch im unteren Fensterbereich):



Identitätswechsel

Legt fest, wie die Aktion (der Service) auf der Arbeitsstation ausgeführt werden soll. Jede Option beeinflusst die Aktion und alle in ihr enthaltenen Aktionselemente.

Verfügbare Optionen:

Bei "**Interaktiver Benutzer**" wird die Aktion im Kontext des Benutzers ausgeführt, der bei dieser Arbeitsstation angemeldet ist. Alles, was dem Benutzerobjekt der Arbeitsstation zugewiesen wurde, wird an die Aktion übergeben. Normalerweise sind dies nur die Zugriffsrechte des Benutzers. Wurden nur die Standard-Benutzerrechte vergeben, kann die Aktion keine Einstellungen verändern, für die Verwaltungsrechte erforderlich sind. (Durch Anwahl von "Interaktive Benutzer" unter Windows 95 wird angegeben, dass für den Zugriff die Benutzerrechte verwendet werden.)

Bei "**System**" wird die Aktion als lokaler Service im Hintergrund ausgeführt. Wenn die Aktion abgesendet wird, wird sie im Kontext des lokalen Systems ausgeführt. Die Aktion verfügt über Verwaltungsrechte für die Arbeitsstation. Verwenden Sie diese Option nur dann, wenn die Aktionen keinen weiteren Eingriff durch einen Benutzer erfordern (es wird keine Benutzerschnittstelle angezeigt, bei der eine Eingabe vorzunehmen ist). (Mit der Wahl von "System" unter Windows 95 wird angegeben, dass für den Zugriff die Rechte der Arbeitsstation verwendet werden.)

Bei "**Nicht-gesichertes System**" (nur NT) können Aktionen im Hintergrund ausgeführt werden. Gleichzeitig besteht aber die Möglichkeit eines Dialogs mit dem Benutzer, falls erforderlich. Verwenden Sie diese Option bei Anwendungen, die einen Benutzerdialog erfordern. Diese Methode ist ungesichert, da das Betriebssystem Adreßzuordnungen zwischen Benutzer- und Systembereich zulässt (was normalerweise ausgeschlossen ist). Sie können diese Option verwenden, wenn in den geschützten Bereichen der Registrierung und des Dateisystems Änderungen vorgenommen werden müssen und für die Ausführung eine Benutzerschnittstelle erforderlich ist. Allerdings sollte diese Option nach Möglichkeit vermieden werden.

Klicken auf den Button *Aktionen* (Abb. 22) öffnet das Menü, in dem eingetragen wird, was die Aktion durchführen soll (z.B. Programme starten). Bitte zu Batchdateien den Hinweis auf Seite 47) beachten!



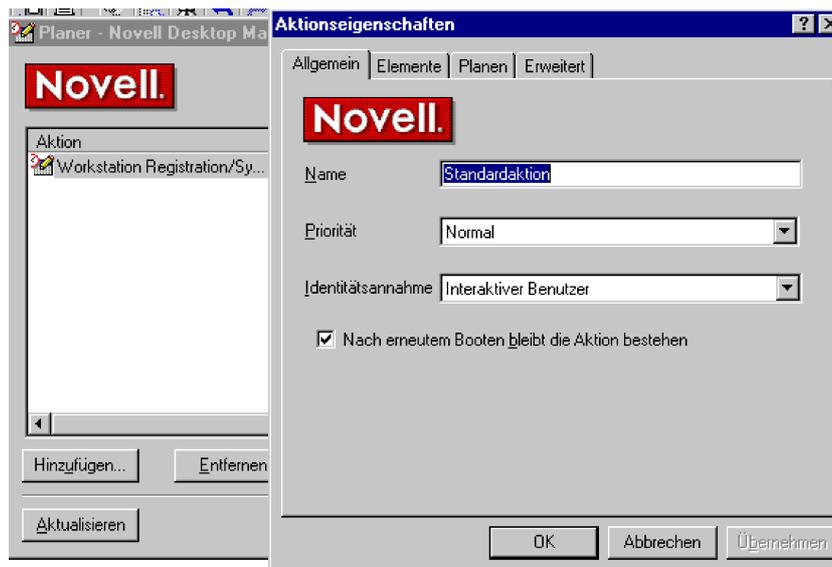
Eine auf diese Weise erzeugte Aktion wird beim nächsten Einloggen eines Benutzers, für den die Richtlinie gilt, auf die Workstation übertragen und erscheint danach im Planer. Loggt sich der User danach erneut ein, wird die Aktion erstmalig durchgeführt.

WICHTIG: NDS-basierte Aktionen können nicht durch den lokalen Planer, sondern nur über den Administrator (das Programm *Administrator!*) in der entsprechenden Richtlinie bearbeitet oder gelöscht werden.

Benutzererstellte Aktionen

Auf ähnliche Weise können sich auch Benutzer selbst Aktionen programmieren - sie machen dies jedoch nicht mit dem Administrator, sondern über den Planer selbst. Dabei stehen ihnen - außer den erweiterten Einstellungen - dieselben Eigenschaften zur Verfügung wie bei NDS-basierten Aktionen. Benutzerdefinierte lokale Aktionen können **ohne Netzwerkanbindung** durch den "lokalen" Teil des Planers erstellt und ausgeführt werden.

Der Benutzer öffnet den Planer und klickt dann auf *Hinzufügen*:



Die Anzeige der Login-Skripts ...

Und das Offenhalten oder Schließen des Ergebnisfensters kann jetzt (leider) nur noch im Anmeldefenster selbst (Taste *Erweitert*, dort Register *Skript* nach Drücken von <Stg><Alt><Entf>) und nicht mehr zentral über eine Richtlinie ein- oder ausgeschaltet werden.

Der *Application Launcher* (NAL 2.7)

Einleitung

Der Application Launcher ist eine - bisher - einzigartige Erfindung von Novell. Mit seiner Hilfe ist es möglich, Programme in der Form von *Anwendungsobjekten* (*Application Objects*) so in der NDS zu integrieren, dass Benutzer, wenn sie dieses Objekt im *Anwendungs-Explorer* anklicken,

- entweder das entsprechende Programm starten, falls es auf ihrem PC bereits installiert ist
- oder dieses Programm vollautomatisch und ohne ihr Zutun zunächst installieren und danach starten.

Dabei ist es gleichgültig, ob das Programm

- lokal (auf dem entsprechenden PC),
- auf einem Novell Server oder
- auf einem Server unter einer anderen Plattform (z.B. Windows NT) liegt.

Sämtliche Voreinstellungen, die für die Installation nötig sind, sind im *Anwendungsobjekt* definiert.

Somit müssen Programminstallationen nur einmal (von einem beliebigen Rechner aus) vorgenommen und für die entsprechenden Benutzer freigegeben werden. Alles andere erledigen die Benutzer (ohne dass sie es merken), selbst. Dabei können nicht nur Neuinstallationen, sondern auch Updates, Patches und Service Packs auf diese Weise konfiguriert werden, denn der Administrator kann auch festlegen, dass ein Programm nur einmal aufgerufen werden kann und danach nicht mehr angeboten wird. Außerdem kann er festlegen, ob lokal ein bestimmtes Betriebssystem, ein bestimmter Prozessor, eine minimale freie Festplattenkapazität o.ä. vorhanden sein muß, damit ein Programm überhaupt angeboten wird (weil es sonst z.B. nicht ordnungsgemäß laufen würde).

Installation

Der NAL Explorer wird durch den Aufruf des Programms NAEXPLD (auf SYS:PUBLIC) auf einem Rechner installiert. Er muß beim Einloggen jedesmal erneut gestartet werden - am besten über einen Eintrag im Container-Login Script (if OS="WINNT" then #nalexpld).

Danach gibt es (wenn diese Standardeinstellung nicht deaktiviert wurde) ein neues Icon auf dem Desktop. Doppelklick auf dieses Icon öffnet den NAL Explorer, ein Fenster, in dem alle Applikationen (=Programme), die für den entsprechenden Benutzer freigeschaltet sind, angeboten werden. Der NAL Explorer kann auch zur Standard-Oberfläche unter Windows gewählt werden, sodass die Benutzer sich gar nicht mehr unter Windows selbst „verlaufen“ können (siehe dazu die Hinweise in der Online-Hilfe. Unter Windows 3.1 wird dazu z.B. in der SYSTEM.INI im Abschnitt [boot] unter SHELL=PROGMAN.EXE eingetragen: SHELL=NAL.EXE).



Der NAL Explorer unter Windows NT ist in Abb. 27 zu sehen, unter Windows 3.1 und Windows95 gibt es kaum Unterschiede. Zusätzlich können in beiden 32bit-Systemen Programme auch in der Startmenü, auf dem Desktop und in der Taskleiste angeboten werden. **HINWEIS:** auch die Einträge im Startmenü tauchen nur auf, wenn der NAL Explorer nach dem Einloggen aufgerufen wurde! Siehe hierzu Abb. 28.

Zusätzlich zu diesen „Fähigkeiten“ bietet der NAL

- eine *Auslastungsverteilung*: ist eine Anwendung auf mehr als einem Server installiert, wird sie von dem Server gestartet, der gerade am „wenigsten zu tun“ hat.

- eine *Fehlertoleranz*: ist eine Anwendung auf mehr als einem Server installiert und der erste Server in der Liste gerade nicht am Netz, wird sie vom einem der anderen Server geladen.
- zeitliche Begrenzung (*Zeitplanung*) für die Benutzung einer Anwendung (wichtig z.B. bei Datenbanken, die zu bestimmten Zeiten gesichert oder auf andere Server überspielt werden sollen und zu diesen Zeiten nicht in Benutzung sein dürfen).
- eine Lizenzüberwachung für Programme, die in begrenzten Lizenzzahlen vorliegen. Diese Lizenzüberwachung greift nur, wenn die entsprechenden Programme über den NAL aufgerufen werden.
- *Verify*: hat ein Benutzer aus Versehen Bestandteile eines Programms gelöscht und kann diese Applikation nicht mehr starten, bietet der NAL automatisch an, das Programm zu "verifizieren": es wird neu installiert bzw. fehlende Komponenten ergänzt und dann erneut gestartet.
- *Aktualisieren* (Klick mit der rechten Maustaste auf das NAL Explorer-Icon auf dem Desktop oder auf der Startleiste): hat der Administrator neue Objekte eingerichtet, erscheinen diese nicht sofort im NAL-Explorer, sondern erst nach einer (definierbaren) Refresh-Periode (standardmäßig ausgeschaltet, damit nicht zu viel Netzwerkverkehr entsteht). Klicken auf *Aktualisieren* aktualisiert den NAL Explorer, den Desktop und die Taskleiste sofort (nicht jedoch Anwendungsordner - diese werden erst nach einem erneuten Einloggen aktualisiert!).
- Informationen zum entsprechenden Programm und zu dem dafür verantwortlichen Mitarbeiter (Eigenschaften).

HINWEIS: für *Auslastungsverteilung* und *Fehlertoleranz* müssen mindestens zwei *Applikationsobjekte* eingerichtet werden, die auf Programme auf verschiedenen Servern weisen. Alle Objekte können dabei im gleichen Container stehen, die zusammengehörenden Objekte müßten sich dann im Namen unterscheiden. Die Objekte der Zusatzinstallationen werden zwar für die User freigegeben, aber nicht in ihre *Startmenü/Desktop/Taskleiste* aufgenommen. Die **Benutzer** sehen so nur **ein Icon**, nämlich das des Hauptobjekts. Ist die Haupt-Applikation verfügbar, wird das zugehörige Programm ausgeführt, im anderen Fall das Programm, das zum ersten Zusatz-Objekt in der Liste gehört usw.

Der NAL Explorer unter Windows NT:

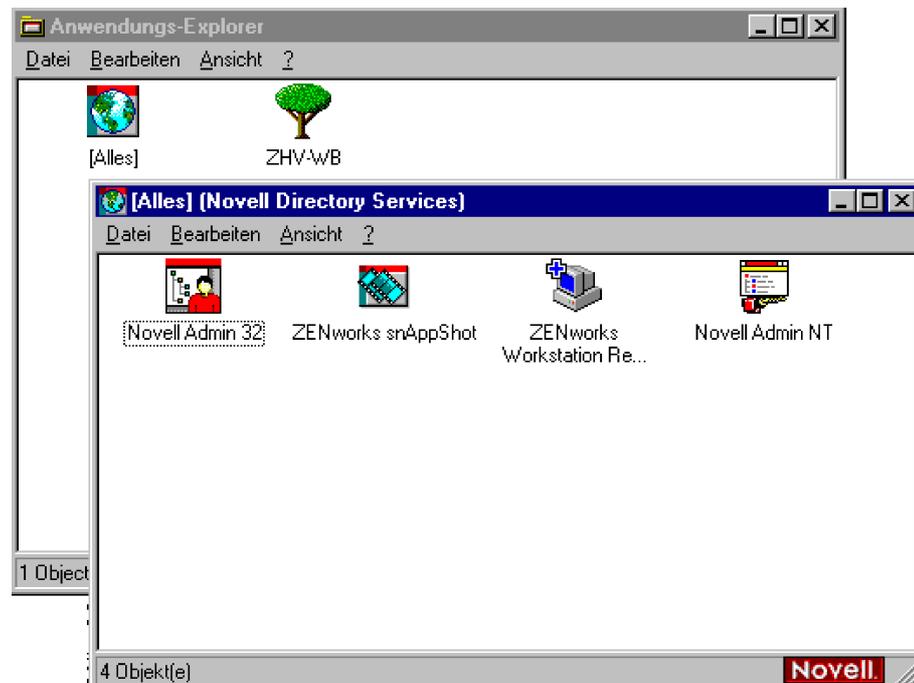


Abb. 27: Der NAL Explorer unter Windows NT

Der NAL Explorer im Startmenü. Es wurden zwei Application Folder definiert, einer mit dem Namen *Administration* und einer mit dem Namen *NAL Programme*. Näheres dazu im Kapitel über Folder.

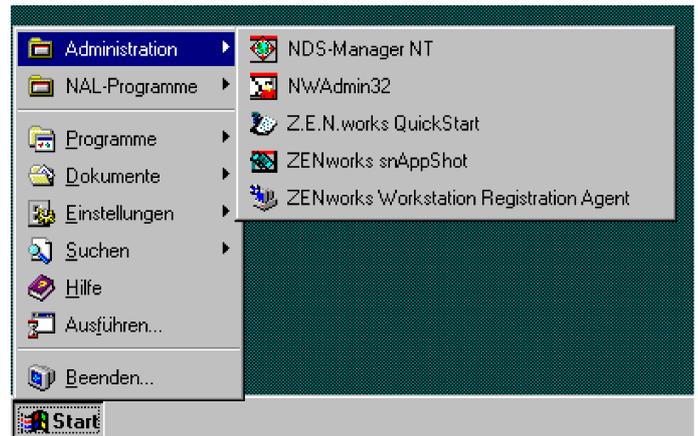


Abb. 28: Der NAL Explorer im Startmenü

Neuigkeiten

Der NAL 2.7 hat im Vergleich zur Vorgängerversion 2.01 einige neue Eigenschaften, vor allem aber die Möglichkeiten

- Dateien auch ohne die entsprechende Berechtigung in Systembereiche zu kopieren oder für den aktuellen Benutzer gesperrte Teile der Registry zu ändern (siehe dazu Anhang 6).
- Applikations-Installationen automatisch zeitgesteuert (z.B. mitternachts) durchzuführen, ohne dass ein Benutzer eingeloggt ist - die entsprechende Station muß lediglich eingeschaltet und der Novell-Client geladen sein.
- *Rechte an Dateien und Verzeichnissen im Applikationsobjekt selbst* zuzuweisen (bisher war dies ein zusätzlicher Schritt Administrator). Damit ist gewährleistet, dass alle, die die Applikation benutzen dürfen, für die **Dauer der Nutzung** die entsprechenden Rechte an Dateien und Verzeichnissen haben und nach Verlassen der Applikation evtl. sogar das entsprechende Verzeichnis nicht mehr "sehen" können, wenn sie die Rechte an dem Verzeichnis **nur durch das Applikationsobjekt** bekommen haben. Diese Methode bietet somit **Komfort** (Rechte zusammen mit dem Objekt an **einer Stelle** zuweisen) und **Sicherheit** zugleich! Applikationen können damit nur durch das Applikationsobjekt und nicht über den Explorer gestartet werden.

Upgrade

Der NAL 2.5 kann mit den alten *Application Object Templates* arbeiten und die alten *Application Objects* „starten“, nutzt aber nicht alle Möglichkeiten - siehe dazu den Anhang 2: „Upgrading from NAL 2.x to Z.E.N. works 1.0“).

Einige Hinweise zu Applikationsobjekten

Identifikation (siehe Abb. 22)

- *Anwendungssysboltitel* zeigt die Unterschrift, die das Icon, wo immer es auftaucht (Start-Menü, Desktop, System Tray usw.) haben wird.

- *Pfad zur ausführbaren Datei* muß einen Eintrag enthalten, wenn die Applikation gestartet werden soll. Ist die Option *Nur Installieren* aktiviert, wird die Applikation beim Klicken auf das Applikationsobjekt nur installiert.
- *Symbole anordnen* erlaubt, die für einen Benutzer freigegebenen Applikationen zu sortieren. Das ist vor allem wichtig, wenn sie beim Start des Application Launchers per *Ausführung erzwingen* automatisch gestartet werden sollen und dabei eine bestimmte Reihenfolge eingehalten werden muß.
- Ist *Einmal ausführen* aktiviert, kann die Applikation **pro Benutzer** nur einmal gestartet werden und das Applikationsobjekt wird danach nicht mehr angeboten (die Ausführung wird in der lokalen Registry eingetragen). Diese Eigenschaft ist z.B. geeignet für Installationen, die pro Benutzer nur einmal durchgeführt werden müssen.

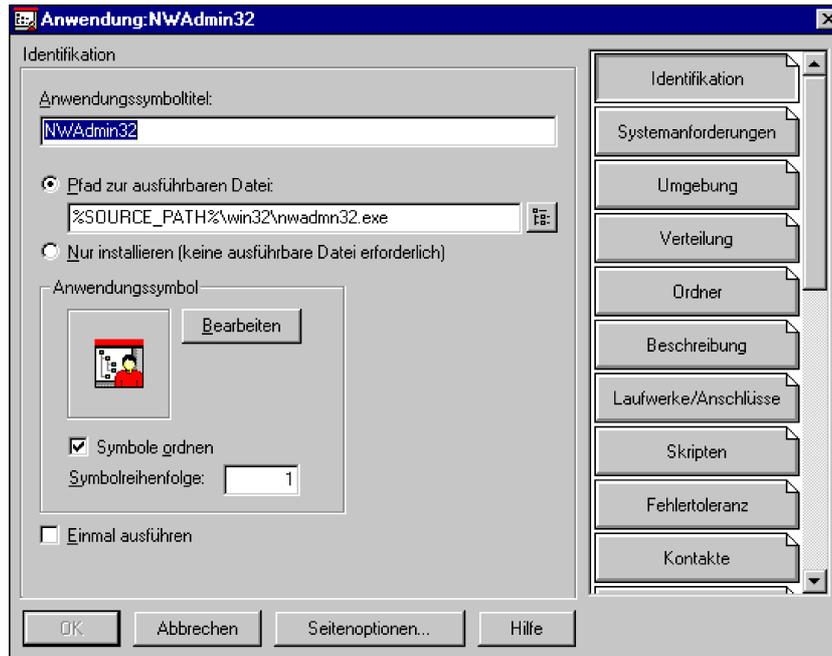


Abb. 29: Eigenschaften *Identifikation*

Hinweis aus der Online-Hilfe zum Punkt *Symbole ordnen*:

Anwendungssymbole ordnen und Ausführungsreihenfolge erzwingen

Die Option "Symbole ordnen" umfaßt zwei sehr nützliche Funktionen: erstens organisiert sie die Symbole in Application Launcher und Anwendungs-Explorer und zweitens legt sie die Reihenfolge fest, in der erzwungen auszuführende Anwendungsobjekte ausgeführt werden. Die Reihenfolge ist von links nach rechts oder von oben nach unten, entsprechend der Art und Weise, wie der Benutzer Symbole im Application Launcher anzeigen läßt. Zum Ordnen geben Sie in das Textfeld "Symbolreihenfolge" einen Zahlenwert ein. Alle Anwendungsobjekte, die geordnet werden sollen, müssen einen Zahlenwert aufweisen. Der Wert Null gibt dem Symbol die höchste Priorität und damit den höchsten Rang auf der Liste. Der Maximalwert ist 999. Falls Sie die Anwendungsobjekte nicht ordnen, werden sie standardmäßig alphabetisch geordnet.

Angenommen, Sie haben zehn Symbole (Anwendungen A, B, C, D, E, F, G, H, I und J), die Sie im Application Launcher anordnen möchten. Sie legen für Anwendung G die Ordnungsnummer 0 fest, für Anwendung F die 1, für Anwendung E die 2 und für Anwendung D die 3. Für die Anwendungen C, B, H und I legen Sie die Ordnungsnummer 4 fest. Die verbleibenden Anwendungen A und J ordnen Sie nicht.

Als Ergebnis werden die ersten vier Anwendungen geordnet, wobei G die erste in der Liste ist, gefolgt von F, E und D. Danach folgen die Anwendungen C, B, H und I in alphabetischer Reihenfolge. Die letzten beiden Anwendungen A und J stehen am Ende der Liste in alphabetischer Reihenfolge (der Standardreihenfolge).

Wenn (mit diesen Anwendungsobjekten verknüpfte) Benutzer den Application Launcher ausführen, sehen sie eine Liste der Symbole in dieser Reihenfolge. Falls alle diese Anwendungen so eingerichtet sind, dass sie erzwungen ausgeführt werden, werden sie in dieser Reihenfolge ausgeführt, unmittelbar nachdem Application Launcher selbst in den Speicher geladen wurde.

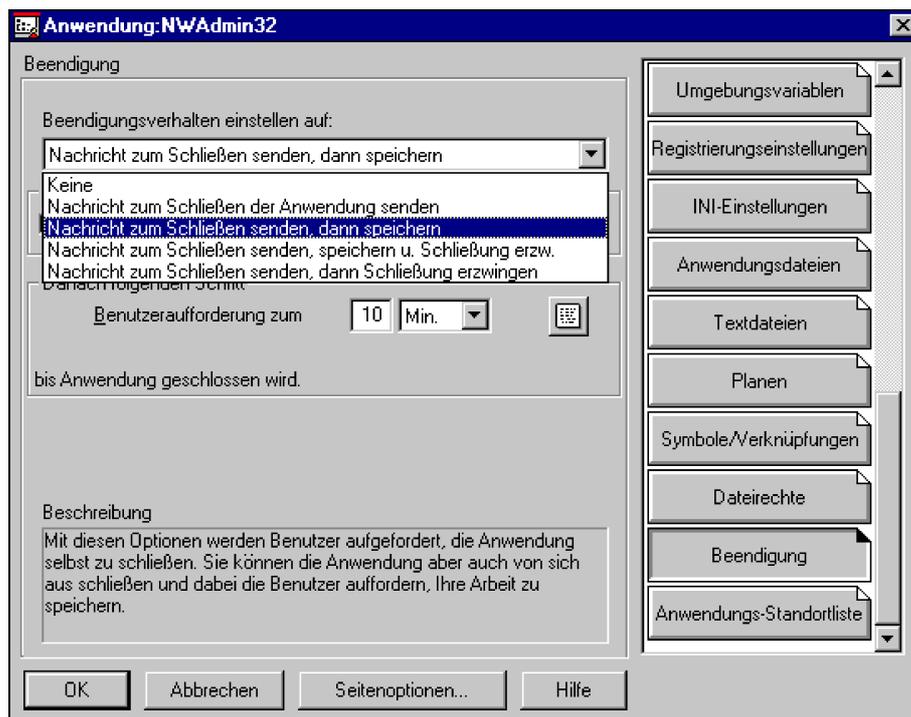
Hinweis: Geordnete und erzwungen auszuführende Anwendungen werden in sequentieller Reihenfolge ausgeführt, ohne dass gewartet wird, bis die vorherige erzwungene Anwendung beendet wurde.

Planen

... erlaubt, Applikationen nur für bestimmte Zeiträume freizugeben. Soll z.B. eine Datenbank gesichert werden, so kann man hier festlegen, dass das zugehörige Programm in dieser Zeit nicht gestartet oder wenn es läuft, beendet wird - siehe dazu der folgende Punkt:

Beendigung

Die Beendigung eines Programms (z.B. weil im Applikationsobjekt Benutzungszeiten festgelegt sind) ist jetzt steuerbar unter der Eigenschaft *Beendigung*:



Laufwerke / Anschlüsse

In der neuen Version können bis zu 16 Laufwerkszuordnungen und 9 Drucker-Ports zugewiesen werden. Die Laufwerkszuordnungen können als *Drive map*, *Root map* und *Search map* definiert werden. Außerdem lassen sich die Berechtigungen für die entsprechenden Festplattenbereiche jetzt im *Applikationsobjekt* selbst festlegen (siehe der nächste Punkt *Dateirechte*). Hinweise aus der Online-Dokumentation:

Laufwerke und Anschlüsse einrichten

Da die Anwendungen, die Benutzer im Application Launcher oder Anwendungs-Explorer sehen, direkt mit Anwendungsobjekten verknüpft sind, müssen Laufwerkszuordnungen nicht über die Anmeldeskripten der Benutzer eingerichtet zu werden.

Jedes Anwendungsobjekt kann jedoch zusätzliche Laufwerke zuordnen und Drucker nach Bedarf umleiten. Mit den Einstellungen auf der Eigenschaftsseite "Laufwerke/Anschlüsse" werden Laufwerkszuordnungen und Druckeranschlußumleitungen für eine Anwendung eingerichtet. Solche Laufwerkszuordnungen und Anschlußumleitungen werden eingerichtet, bevor die Application Launcher- oder Anwendungs-Explorer-Software die Anwendung ausführt.

Falls auf der Eigenschaftsseite "Umgebung" die Option "Netzwerkressourcen ordnen" ausgewählt ist, werden die Laufwerkszuordnungen und Anschlußumleitungen, die der Application Launcher oder Anwendungs-Explorer erstellt, freigegeben, wenn der Benutzer alle Anwendungen verläßt, die diese Ressourcen verwenden.

Angenommen zum Beispiel, Sie konfigurieren eine Textverarbeitungsanwendung, die ein bestimmtes zugeordnetes Laufwerk benötigt. Diese Anwendung wird vom selben Laufwerksbuchstaben aus installiert und ausgeführt. Sie möchten Laufwerk W: für diese Anwendung zuordnen und legen auf der Eigenschaftsseite "Identifikation" im Textfeld "Pfad zur ausführbaren Datei:" W:\APPS\WORD1\WORD1.EXE fest. Auf der Eigenschaftsseite "Laufwerke/Anschlüsse" ordnen Sie anschließend Laufwerk W: dem Server zu, auf dem die Anwendung gespeichert ist. Wenn der Application Launcher oder Anwendungs-Explorer die Anwendung ausführt, sucht er nach einer Zuordnung für Laufwerk W:.. Falls w: dem richtigen Volume und Server zugeordnet ist, wird das Laufwerk verwendet. Falls w: einem anderen Server oder Volume zugeordnet ist, wird es nicht verwendet und die Anwendung wird nicht ausgeführt. Falls w: überhaupt nicht zugeordnet ist, ordnet es der Application Launcher oder Anwendungs-Explorer zu. Nur wenn der Application Launcher oder Anwendungs-Explorer das Laufwerk tatsächlich zuordnet, bereinigt er die Zuordnung nach Beendigung der Anwendung. Falls mehrere Anwendungen dieselbe Laufwerkzuordnung verwenden, werden die Laufwerkzuordnungen erst bereinigt, wenn die letzte Anwendung beendet ist.

Hinweis: Der Pfad zu ausführbaren Dateien, die Laufwerkzuordnungen und Anschlußumleitungen werden als Zeichenketten gespeichert, nicht als NDS*-Objektnamen. Falls ein Teilbaum verschoben wird, verweisen daher einige Anwendungsobjekte unter Umständen noch auf die vorherigen, jedoch möglicherweise ungültigen Objekte.

Dateirechte

Erhält ein Benutzer das Recht, eine Applikation auszuführen, so wurde im NAL 2.01 das Recht, die entsprechenden Verzeichnisse benutzen zu dürfen, nicht automatisch zugeteilt. Dies ist jetzt geändert worden: unter *Dateirechte* trägt man die Rechte an Dateien und Verzeichnissen ein, die für den Start der Applikation notwendig sind. Dazu ein Hinweis aus der Online-Dokumentation:

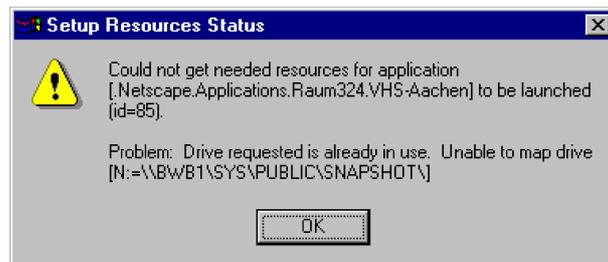
Dateirechte beim zugeordneten Anwendungsobjekt gewähren

Mit der Eigenschaftsseite "Dateirechte" werden Rechte auf Dateien, Verzeichnisse und Volumes gewährt, wenn dieses Anwendungsobjekt mit einem Benutzerobjekt, mit einem Gruppen-, Organisations- oder Länderobjekt bzw. einem organisatorischen Einheitsobjekt verknüpft ist, mit dem der Benutzer bereits verknüpft ist. Derartige Rechte werden gelöscht, sobald die Verknüpfung des Anwendungsobjekts aufgehoben wird.

Falls zwei verschiedene Anwendungsobjekte Rechte auf dieselbe Datei, dasselbe Verzeichnis oder Volume gewähren und nur die Verknüpfung eines der beiden mit dem Benutzer getrennt wird, verliert der Benutzer trotzdem alle Rechte, obwohl er mit dem anderen Anwendungsobjekt weiterhin verknüpft ist.

Hinweis: Dateirechte sind in dieser Version nicht dynamisch. So hängen die Dateirechte, die Sie einrichten, zum Beispiel nicht davon ab, ob der Benutzer auch tatsächlich mit der Anwendung arbeitet.

WICHTIG: hier verwendete Laufwerksbuchstaben dürfen nicht vorher (z.B. im *Login Skript*) zugewiesen worden sein, sonst erscheint die nebenstehende Fehlermeldung und die Applikation wird nicht gestartet. Auf der anderen Seite muß man sich nur wenige Laufwerksbuchstaben "freihalten", da diese ja für jede Applikation, die nicht parallel mit anderen laufen soll, "wiederverwendet" werden kann.



Scripten

Hierzu nur ein Hinweis: Eintragungen in den Skripten müssen den Login Script-Konventionen entsprechen, d.h. externe Programme müssen mit einem führenden # versehen sein (z.B. #doskey) und Befehle wie *copy* oder Batchdateien müssen durch vorangestelltes *#command /c* (Win3.1/95) oder *cmd /c* (WinNT) eingeleitet werden.

Zu den anderen Eigenschaften ... bitte die Online-Hilfe und die Doku-CD befragen.

Neue Eigenschaften im Objekt *Container*

Nach der Installation von Z.E.N. tauchen im Container neue Eigenschaftsgruppen auf. Fünf davon beziehen sich auf die Druckdienste, wobei **NDPS** für **Novell Distributed Print Services** steht (näheres dazu im Skript zu Novell 5). Unter den Eigenschaften, die mit *Nicht NDPS* anfangen findet man die gewohnten alten (Warteschlangenbasierten) Druckdienst-Konfigurationsmöglichkeiten.

Verknüpfte Richtlinienpakete

zeigt die dem Container zugeordneten Pakete. Doppelklick auf den entsprechenden Eintrag öffnet das Menü zu dessen Bearbeitung.

Arbeitsstations-Registrierung

... zeigt die Workstations, die zwar schon registriert, aber noch nicht importiert sind. Hier stehen also in der Regel keine Einträge.

Arbeitsstationsinfo

... zeigt Bewegungen von Workstations im NDS-Baum

Arbeitsstationsfilter

... erlaubt eine Filterung, welche Workstations in einen Container importiert werden dürfen.

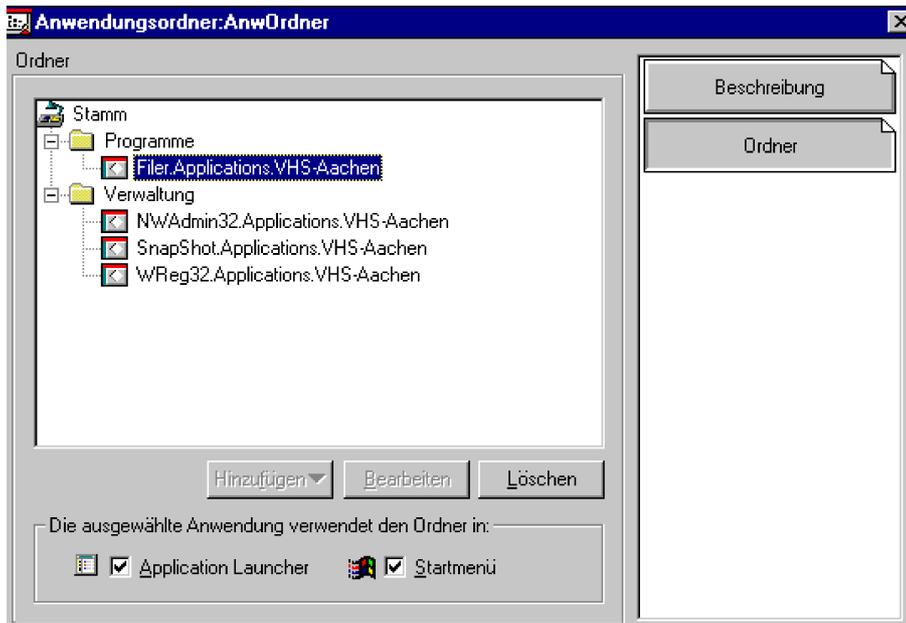
Konfiguration des Startprogramms

... dient zur Festlegung, wie sich der Launcher verhält und was der Benutzer damit machen darf. Diese Eigenschaft gibt es für Benutzer, Gruppen und Container:



Das neue Objekt *Anwendungsordner*

Applikationsobjekte können jetzt in Anwendungsordnern organisiert werden. Sobald ein Ordner angelegt wurde, verschwindet die bis dahin gültige Struktur (*Container-* und *User-*bezogene Applikationen). Natürlich tauchen die Objekte in den Ordnern nur dann beim aktuellen User auf, wenn er dem entsprechenden Objekt assoziiert wurde und nur dort, wo es unter den Eigenschaften des Containers, der Gruppe oder des Users selbst festgelegt wurde. (Applikationsobjekte, die **keinem Ordner** zugewiesen wurden, tauchen immer noch unter dem Organisations-/Gruppen-/Containernamen auf, dem sie zugeordnet sind.)

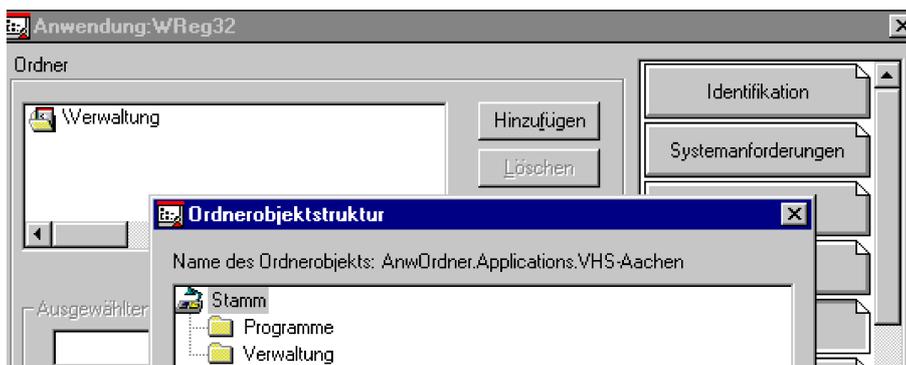


Anders ausgedrückt: wurde im **Container** definiert, dass eine Application **nicht im Startmenü** auftaucht und aktiviert man jetzt im **Ordner**, dass Sie **im Startmenü auftauchen soll**, so tut Sie es DENNOCH NICHT aufgrund der gegenteiligen Einstellung im Container.

HINWEISE

Alle Applikationen direkt unterhalb von *[Root]* tauchen oberhalb des Startmenüs auf (wenn Startmenü beim entsprechenden Eintrag aktiviert ist), alle unterhalb der Ordernamen nur im Folder selbst.

Möchte man einem Folder eine Application zufügen, macht man das am besten nicht innerhalb des Folder-, sondern innerhalb des Applikationsobjektes selbst. Zunächst wählt man *Hinzufügen*, dort *Verbundener Ordner*, dann im folgenden Fenster den entsprechenden Ordner und am Schluß die Position im Ordner selbst:



Das Programm *snAppShot*

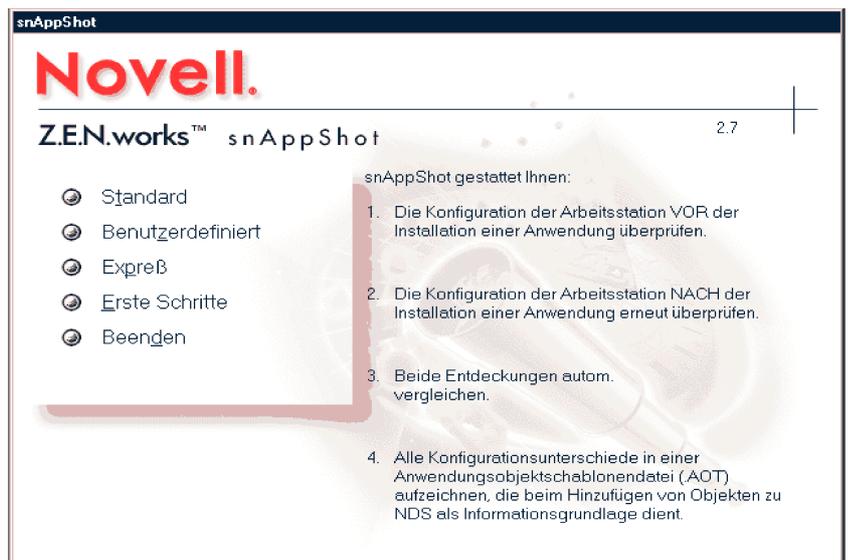
Das Programm *snAppShot* gehört zum *Application Launcher* und dient dazu, automatisch Vorlagen (Templates) für *Applikationen* zu erstellen. Im folgenden werden alle Menüs dieses Programms besprochen. Es gibt den *snAppShot* für alle Plattformen (16bit und 32bit) und er führt die folgenden Arbeitsschritte durch:

1. Erster „Schnappschuss“ (*snAppShot*) des Computers, auf dem es gestartet wird. Soll ein Applikationsobjekt für einen Windows 3.1-Rechner erstellt werden, muß *snAppShot* auf einem Windows 3.1-Rechner gestartet und das Programm dort installiert werden. Festhalten aller Programme und Module, der *.INI's und Programmgruppen (16bit), der Registry-Einträge und Verknüpfungen in Desktop und Startmenü (32bit).
2. Installation des gewünschten Programms, entweder lokal oder im Netzwerk
3. Zweiter „Schnappschuss“
4. Erstellung einer AOT-Datei (eines *Application Object Templates*, der oben angesprochenen Vorlage) in der die Unterschiede (Dateien/INI's/Registry usw.), die durch die Installation (d.h. zwischen erstem und zweitem Schnappschuss) aufgetreten sind, festgehalten werden.

Danach kann man mit dem Administrator unter Verwendung der AOT-Datei das *Applikationsobjekt* ohne weitere „manuelle“ Eingriffe erstellen und für Benutzer freigeben. Meistens wird man jedoch noch weitere Eigenschaften festlegen wollen, u.a., für welche Plattform das Programm ist, Laufwerkszuordnungen, Druckerumleitungen u.ä.

Nach dem Aufruf erscheint zunächst eine Aufforderung auszuwählen, ob man

- die *Standard-* (basierend auf von Novell vordefinierten oder vom Benutzer bei einer früheren Verwendung erstellte Vorgaben)
- *Benutzerdefinierte (Custom* - alle Angaben werden abgefragt) oder die
- *Express-Methode* bei der Erstellung anwenden möchte.
- Außerdem kann man sich Informationen zum Programm anzeigen lassen (*Getting Startet*)



Die erste Methode greift auf eine Datei STANDARD.INI (standardmäßig im Verzeichnis SYS:PUBLIC\NALLIB zurück, in der man die Einstellungen für die meisten Vorgaben in den nun folgenden Fenstern ablegen kann. Kurz vor Verlassen des Programms wird man gefragt, ob die bis dahin gemachten Angaben als „Preferences“, d.h. bevorzugte Einstellungen, gespeichert werden sollen. Man kann hier beliebig viele dieser INI-Datei anlegen - sie enthalten dann jeweils die kompletten Einstellungen zum letzten *snAppShot*.

Ich beschreibe hier die Erstellung *Benutzerdefiniert*.

Auf dem zweiten Bildschirm werden zwei Angaben abgefragt:

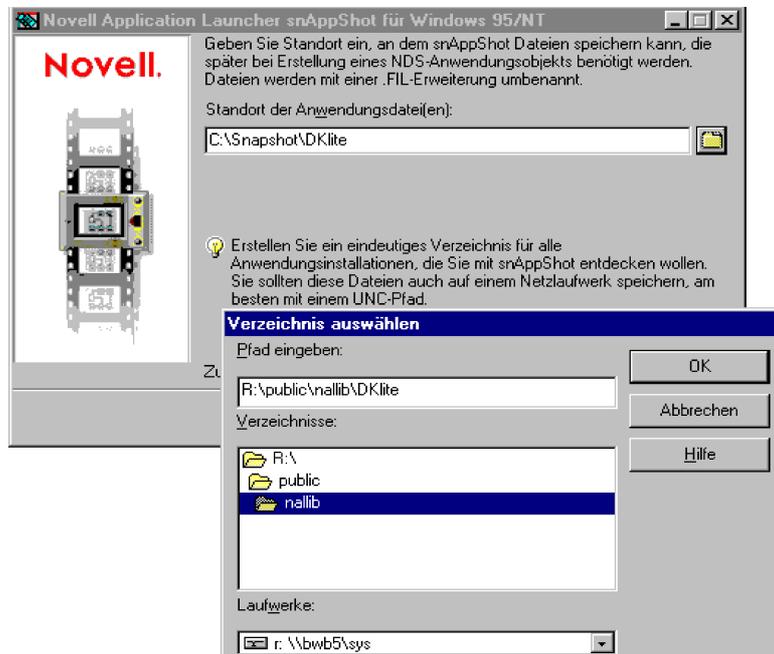
1. *NDS Application Object Name*: Wie das **Applikationsobjekt** heißen soll. Es geht hierbei um den Namen, der (unter dem Administrator sichtbar) im **NDS-Baum** auftaucht
2. *Application Icon Title*: Wie die **Unterschrift unter dem Icon** bzw. der Titel der Applikation im Startmenü lauten soll. Hierbei geht es um den Namen, den der **Anwender sieht** - entweder im NAL-Explorer, auf dem Desktop, in der Taskleiste oder im Startmenü.



Auf dem dritten Bildschirm gibt man an, **wo die Dateien landen sollen, die von einem Benutzer bei der Installation benötigt werden.**

Wenn ein Benutzer diese Applikation aufruft und das Programm noch nie bei ihm auf dem Rechner gelaufen ist, muß er sich in diesem Moment die Dateien und Einstellungen zu der entsprechenden Anwendung in den INI's bzw. in der Registry auf seinen Rechner kopieren können. Der hier angegebene Pfad weist auf das entsprechende Verzeichnis - standardmäßig SYS:PUBLIC\NALLIB\<>Unterverzeichnis mit dem Programmnamen>, hier z.B. DKLite).

snAppShot kopiert alles, was an Unterschied zwischen den beiden Schnappschüssen festgestellt wird, in dieses Verzeichnis. Es sollte ein eigenes (Netzwerk-) Verzeichnis für jede Programminstallation angelegt werden.

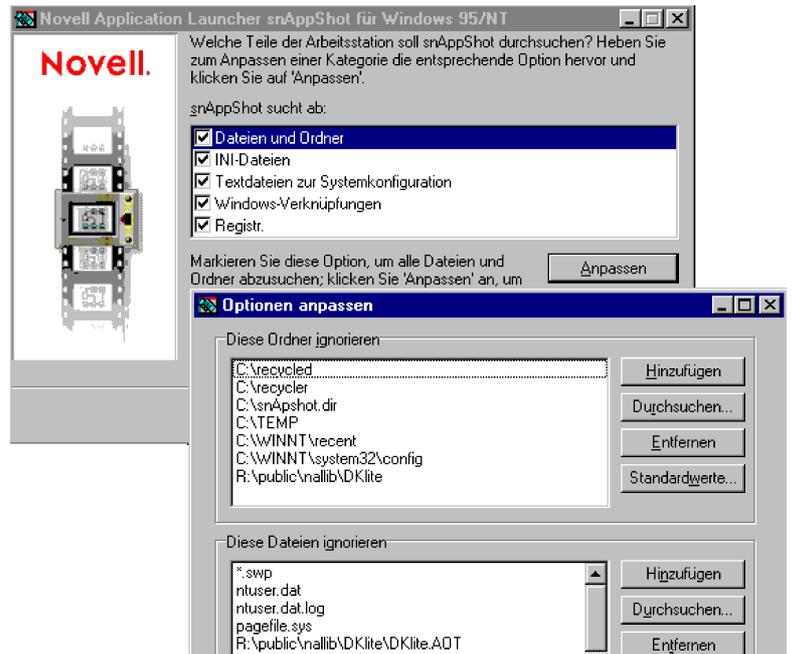


Im nächsten Fenster legt man **Namen und Ort für das Template selbst** fest. snAppShot schlägt die im vorigen Fenster vorgenommene Auswahl vor. Ich würde dies übernehmen. Es ist am sinnvollsten, wenn das Verzeichnis und das Template den Namen des Programms haben - damit weiß man immer, was zusammengehört.



Jetzt kann man vorgeben, **wo snAppShot Veränderungen verfolgen soll**. Nur, wenn man ganz sicher ist, dass ein Programm irgendwo nicht eingreift oder ungewöhnlicherweise doch eingreift, kann man die Vorgaben anpassen. **Veränderungen der Vorgaben können aber immer dazu führen, dass Veränderungen nicht entdeckt werden und damit spätere automatische Installationen auf anderen Rechnern nicht erfolgreich sind!**

Daher der TIPP: Einstellungen übernehmen.



Hier kann man vorgeben, **wo nach Datei-Unterschieden gesucht werden soll**, d.h., wo snAppShot untersuchen soll, ob während der Installation Dateien dorthin kopiert werden. Dies ist standardmäßig mindestens das Laufwerk, auf dem sich die Verzeichnisse WINNT und SYSTEM32 befinden, weil viele Programme auch bei Netzwerkinstallationen dorthin DLL's und andere Bestandteile kopieren - entweder, weil es das Betriebssystem verlangt oder weil zu häufige Netzwerkzugriffe vermieden werden sollen.



Im nächsten Fenster kann man sich die **bisher gemachten Einstellungen noch einmal ansehen** und evtl. für die Auswahl *Standard* im ersten Fenster (Seite 31) **als Preferences speichern** (siehe dazu die Anmerkungen zum ersten Fenster).



Jetzt beginnt der **erste Schnappschuss**. Interessant an diesem Fenster ist u.U. der Überblick über die Anzahl der Einträge in den verschiedenen Bereichen, der hier präsentiert wird.



Das nächste Fenster enthält sehr klar formuliert alles, was zu tun ist:

1. soll man sich genau **das Verzeichnis** merken, das man bei der nun folgenden **Programminstallation als Ziel** angibt.
2. soll man das Programm, bevor man mit dem snAppShot fortfährt, starten und **Standardvorgaben dort vornehmen** (z.B. Menüleisten, Verzeichnisse u.ä. einstellen). Diese Vorgaben landen mit im Template und werden bei späteren automatischen Installationen mit übernommen. Man spart sich so manuelle Anpassungen des Programms auf jedem Rechner und muß nicht doch noch an die einzelnen Arbeitsplätze.



Danach wird man aufgefordert, das Installationsprogramm zu starten. **Handelt es sich um ein Programm, das von einem Server aus gestartet werden soll, muß hier die Workstation-Installation aufgerufen werden!** Die zugehörige **Serverinstallation muß vorher** - ohne snAppShot - vorgenommen werden, da die Benutzer nicht jedesmal die Installation auf dem Server vornehmen sollen.

Das Programm wird jetzt beispielhaft **auf dem Rechner, auf dem snAppShot läuft**, installiert und kann dort sofort danach gestartet werden.

Der nebenstehende Bildschirm erscheint, wenn das Installationsprogramm verlassen wurde.

ACHTUNG: Jetzt unbedingt das Anwendungsprogramm bzw. alle Komponenten eines Officepaketes starten und konfigurieren (Datei-ablage, Temp-Pfad, Symbolleiste, Position der Verknüpfungen (Einträge auf dem Desktop löschen und im Startmenü von AllUsers in den "privaten" Userbereich verschieben. Erst danach den zweiten Schnappschuss beginnen.



Hier kann man festlegen, wie der **Application Launcher** vorgehen soll, wenn er das Programm auf einem Rechner automatisch installiert.

Wählt man bei *Ordner- und Dateieinträge* die Einstellung aus der Abb., werden Dateien grundsätzlich auf den Rechner kopiert und die Wahrscheinlichkeit ist groß, das snAppShot alle Änderungen mitbekommt.

Dennoch kann es vorkommen, dass das Kopieren einer Datei auf den lokalen Rechner nicht entdeckt wird, da die neue Datei identisch mit einer bereits vorhandenen ist. Automatische Installationen schlagen dann zunächst fehl, weil diese Datei nicht auf einen Rechner kopiert wird, der diese Datei noch nicht enthält. Damit ist das Programm auf diesem Rechner nicht lauffähig und der Eintrag muß (mit dem Administrator) manuell im Application object nachgetragen werden. **Mehr dazu in Anhang 5!**

Bei *INI-Einträge* würde ich die in der Abb. gezeigte Einstellung wählen. Bestehende INI's werden so ergänzt und nicht überschrieben. Bei den *Registrierungseinträgen* würde ich auf jeden Fall die Einstellung *Immer erstellen* übernehmen.



Im nächsten Bildschirm trägt man das Verzeichnis ein, das **im Installationsprogramm für das Programm gewählt wurde** (Abbildung in der Mitte Seite 34).



Der nächste Bildschirm ist etwas für Fortgeschrittene. *Makros* sind **Variablen**, die man später bei der Konfiguration des Application Objects im Administrator benutzen kann. Diese Variablen lassen sich hier definieren. Man kann an dieser Stelle vor allem Variablen für Laufwerke/Verzeichnisse und Benutzer anlegen.

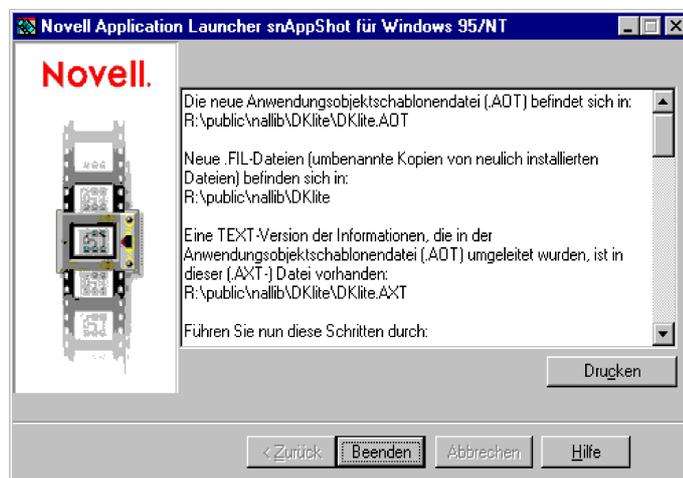
Näheres zu Makros ab Seite 36.



Jetzt folgt der **zweite Schnappschuss**. Hier und auf den folgenden Bildschirmen kann man verfolgen, wo wie viele Änderungen entdeckt werden. Außerdem wird man darüber informiert, welche Eintragungen im *Application Object Template* erfolgen.



Das **Template ist nun fertig** und kann in diesem Fenster vollständig eingesehen werden. Man kann es von hier aus auch ausdrucken.



Im folgenden die Angaben zum Template für die Anwendung DKLite:

Die neue Anwendungsobjektschablonendatei (.AOT) befindet sich in:
R:\public\nalib\DKlite\DKlite.AOT

Neue .FIL-Dateien (umbenannte Kopien von neulich installierten Dateien) befinden sich in:
R:\public\nalib\DKlite

Eine TEXT-Version der Informationen, die in der Anwendungsobjektschablonendatei (.AOT) umgeleitet wurden, ist in dieser (.AXT-) Datei vorhanden:
R:\public\nalib\DKlite\DKlite.AXT

Führen Sie nun diese Schritte durch:

1. Starten Sie NWAdmin und erstellen Sie ein neues Anwendungsobjekt.
2. Bei der Verwendung des Anwendungsobjekt-Erstellungsassistenten suchen Sie die Anwendungsobjektschablonendatei (.AOT), die Sie soeben mit snAppShot erstellt haben.
3. Beim Suchen in den Eigenschaftenseiten des Anwendungsobjekts können Sie feststellen, dass das Anwendungsobjekt mit den Konfigurationsänderungen 'bevölkert' wurde, die snAppShot entdeckt hat.
4. Öffnen Sie die Makros-Eigenschaftenseite im Anwendungsobjekt. Das standardmäßig erstellte Makro SOURCE_PATH verweist auf den Standort der 'installierten' Anwendungsdateien (diejenigen, die mit .FIL-Erweiterungen umbenannt wurden).

5. Beachten Sie, dass TARGET_PATH auf den Standort auf der Arbeitsstation verweist, wo Anwendungsdateien während der Verteilung kopiert werden.
6. Beim Anpassen der Einstellungen oder Makros auf dem Anwendungsobjekt können Sie die Anwendung auf beliebige Weise verteilen. Weitere Informationen finden Sie in der Online-Hilfe des Anwendungsobjekts.
7. Für weitere Informationen über snAppShot und NAL gehen Sie zu folgender Website: <http://www.novell.com/cool solutions/zenworks>.

snAppShot-Gesamtzahlen seit der zweiten Aufnahme
Anzahl der abgesuchten Ordner und Dateien: 10940
Anzahl der abgesuchten Verknüpfungsdateien (.LNK): 579
Anzahl der abgesuchten .INI-Dateien: 112
Anzahl der abgesuchten Textdateien: 2
Anzahl der abgesuchten Registereinträge: 33547

Der Schablonendatei (.AOT) hinzugefügte Einträge
Anzahl der Ordner- und Dateieinträge, die der (.AOT) Datei hinzugefügt wurden: 19
Anzahl der Verknüpfungsdateieinträge (.LNK), die der (.AOT) Datei hinzugefügt wurden: 2
Anzahl der .INI-Dateieinträge, die der (.AOT) Datei hinzugefügt wurden: 0
Anzahl der Textdateieinträge, die der (.AOT) Datei hinzugefügt wurden: 0
Anzahl der Registereinträge, die der (.AOT) Datei hinzugefügt wurden: 25

Erstellung eines *Application Objects* mit Hilfe eines Templates

Wie in dem obigen Text erwähnt startet man nun den Administrator, um aus dem soeben erstellten Template ein *Application Object* zu erstellen. Das erste Fenster im Administrator nach der Auswahl *Erstellen > Application Object* bietet die folgende Auswahl:



Einfaches Anwendungsobjekt erstellen: diese erste Option erlaubt die Erzeugung eines *Application Objects* **ohne Vorlage** (Template). Dies ist nur dann sinnvoll, wenn man sicher ist, dass weder Dateien auf den lokalen Rechner kopiert noch Veränderungen in Registry, Icons und INI's bei einer Installation vorgenommen werden.

Anwendungsobjekt mit einer .aot/.axt-Datei: Erzeugung eines Objektes mit einer Vorlage

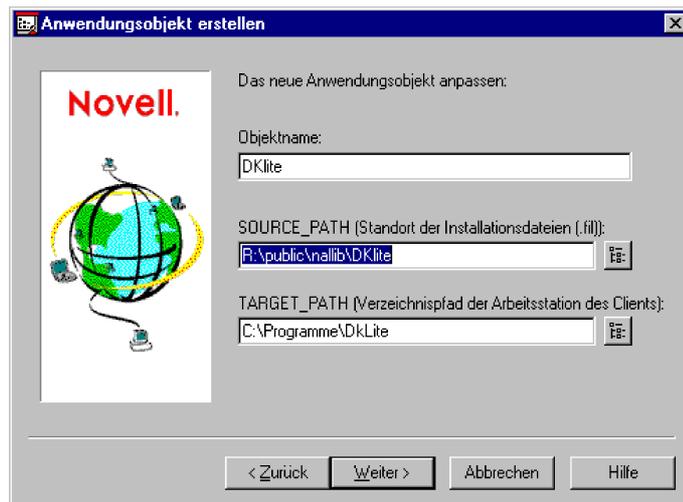
Ein bereits existierendes Anwendungsobjekt duplizieren: falls ein ähnliches oder identisches *Applikationsobjekt* in einem anderen Container erzeugt werden soll, wählt man diese Option. Dies ist z.B. sinnvoll, wenn Benutzer dieselbe Applikation von einem anderen (näher liegenden) Server starten sollen und deswegen alle Angaben bis auf die, wo die Dateien, die beim Installieren benötigt werden und die Applikation selbst liegen. Man kann so alle Angaben übernehmen (z.B. zu Registry, Icons, INI's usw.) und nur diese beiden Verzeichnisse manuell abändern.

Außerdem ist diese Option nützlich bei der Erstellung von doppelten Objekten für die Option *Auslastungsverteilung* und *Fehlertoleranz*.

Im nächsten Fenster gibt man **Ort und Namen des Application Object Templates**, also der Vorlage, an.



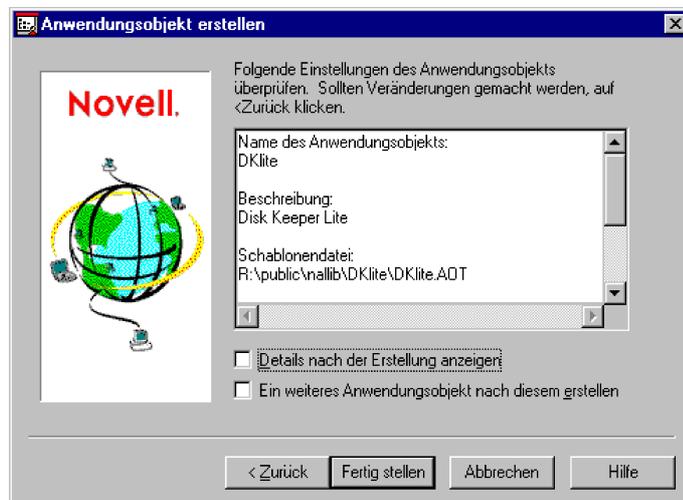
An dieser Stelle kann man noch einmal das Verzeichnis, in das das Programm installiert werden soll, abändern. Der Vorschlag hier stammt aus den Angaben, die man beim Lauf von snAppShot vorgegeben hat.



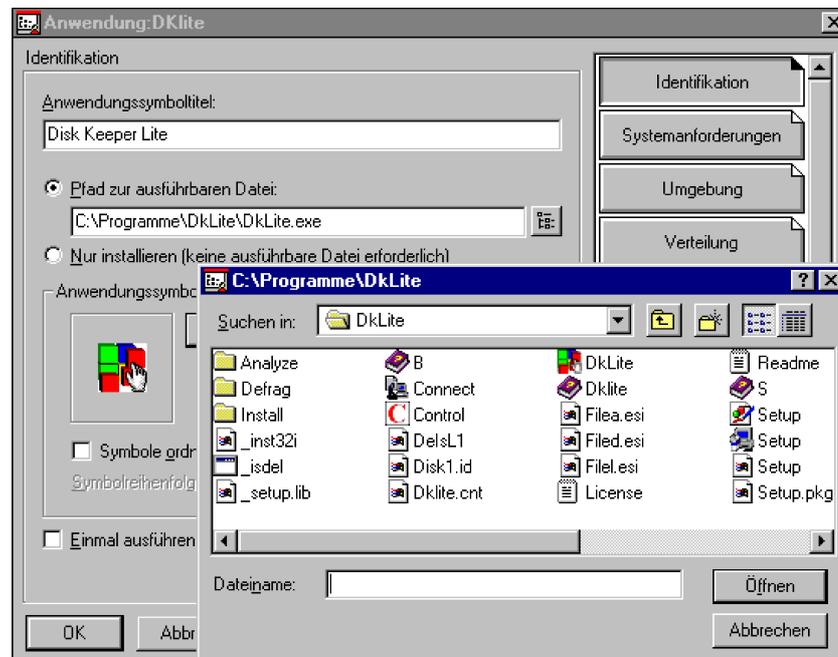
Jetzt lassen sich alle Einstellungen nochmals einsehen.

Wenn man *Details nach der Erstellung anzeigen* aktiviert, lassen sich die Details, die nicht im Template stehen, für eine Funktion des Objektes aber nötig sind, sofort anschließend vornehmen.

Im folgenden Fenster zu den Unterpunkten, die man auf jeden Fall ergänzen muß.



Pfad zur ausführbaren Datei: hier muß der Name der Applikation, d.h. der *.EXE-Datei eingetragen werden, mit der die Anwendung gestartet wird. Wird hier nichts angegeben, wird die Anwendung durch das Applikationsobjekt nur installiert und nicht gestartet. Dies ist aber nur selten sinnvoll, z.B. bei der einmaligen Installation von Dateien für einen Benutzer. Dann sollte aber gleichzeitig die Option *Einmal ausführen* aktiviert werden. Damit verschwindet das Icon für den Benutzer, sobald es einmal aufgerufen wurde (und damit die Installation erfolgte) - der Application Launcher macht einen Vermerk in der lokalen Registry.



Jetzt müssen noch folgen:

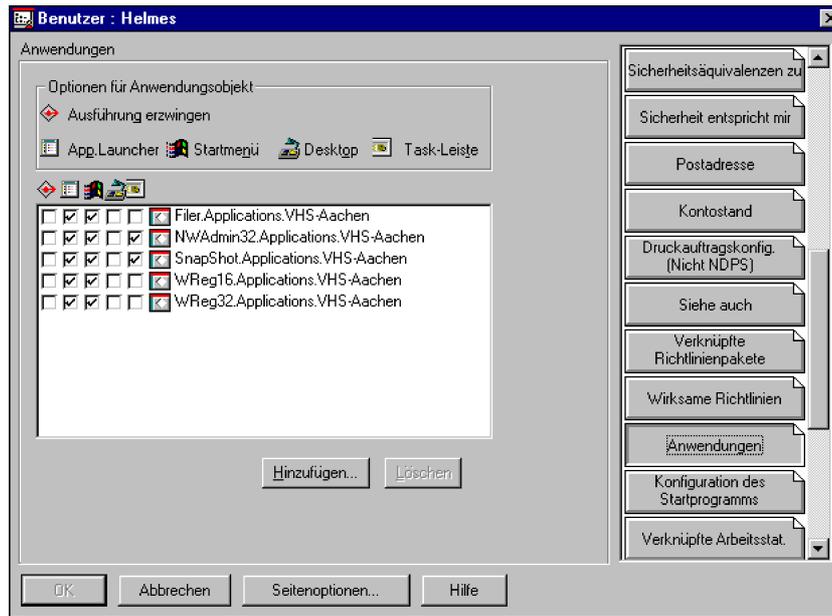
- Angaben dazu, **wer die Anwendung benutzen darf** (im Application Launcher "sehen" soll) (*Verknüpfungen*). Es stehen die Objektarten *Benutzer*, *Gruppe* und *Container* zur Auswahl.
- Angaben zu den Systemanforderungen. Standardmäßig ist nichts aktiviert, so dass die Applikation nicht angeboten wird, **auch wenn sie zugeordnet wurde!**

Jetzt können bei Bedarf noch folgen:

- Angaben zum Arbeitsverzeichnis (*Umgebung*, dort *Arbeitsverzeichnis*), **notwendig, wenn die Applikation DLL's nicht finden kann**. Man trägt dann das Programmverzeichnis ein.
- zu Laufwerken und Druckern (*Laufwerke/Anschlüsse*)
- zu Zeitbeschränkungen (*Planen*) und Programmbeendigung bei Überschreitung dieser Beschränkung (*Beendigung*)
- zu Rechten an Dateien und Verzeichnissen (*Dateirechte*) - siehe den Hinweis auf Seite 25.

Danach ist das Applikationsobjekt fertig. Bei den Benutzern/Gruppen/Containern, für die es freigegeben wurde, muß man am Schluß noch eintragen, wo das Objekt auftaucht. Dies geschieht bei Benutzer/Gruppe/Container unter der Eigenschaft *Anwendungen* durch Anklicken der entsprechenden Option(en):

In der Auflistung stehen alle für den Benutzer freigegebenen Applikationen. In den Kästchen links daneben kann man folgende Einstellungen vornehmen (von links nach rechts):



- *Ausführung erzwingen*: die Applikation wird beim Laden des Launchers automatisch gestartet (siehe dazu den Hinweis auf Seite 26).
- *App. Launcher*: die Anwendung erscheint im Launcher, d.h. dem Fenster, das sich nach Doppelklick auf das nebenstehende Icon öffnet. 
- *Startmenü*: die Applikation erscheint im Startmenü bzw. einem Folder im Startmenü (siehe dazu das Kapitel auf Seite 30).
- *Desktop*: das Icon zu dieser Applikation erscheint auf dem Desktop
- *Taskleiste*: das Icon erscheint in der Taskleiste.



Sämtliche Einstellungen zu Registry, Dateikopien und Icons lassen sich nachträglich korrigieren und anpassen. So kann man z.B. durch Löschen aller Einträge unter *Symbole/Verknüpfungen* verhindern, dass bei der Programminstallation durch den NAL Icons zu diesem Programm auf dem Desktop oder im Startmenü erzeugt werden oder es lässt sich ein anderer Ort für die *Symbole* einstellen. So kann man erreichen, dass Verknüpfungen nicht unter *All Users* (unter *Verknüpfungsstandort* in Abb. 57 wird hier das Macro* *COMMONPROGRAMS* benutzt), sondern z.B. beim aktuellen Benutzer gespeichert werden (Makro hierfür: *PROGRAMS*. Eine Verknüpfung für *All Users* ist meistens unsinnig, da Anwendungen nicht immer für alle Benutzer freigegeben sind, diesen dann aber trotzdem im Startmenü angeboten würden!

* Näheres zu Makros auf der folgenden Seite

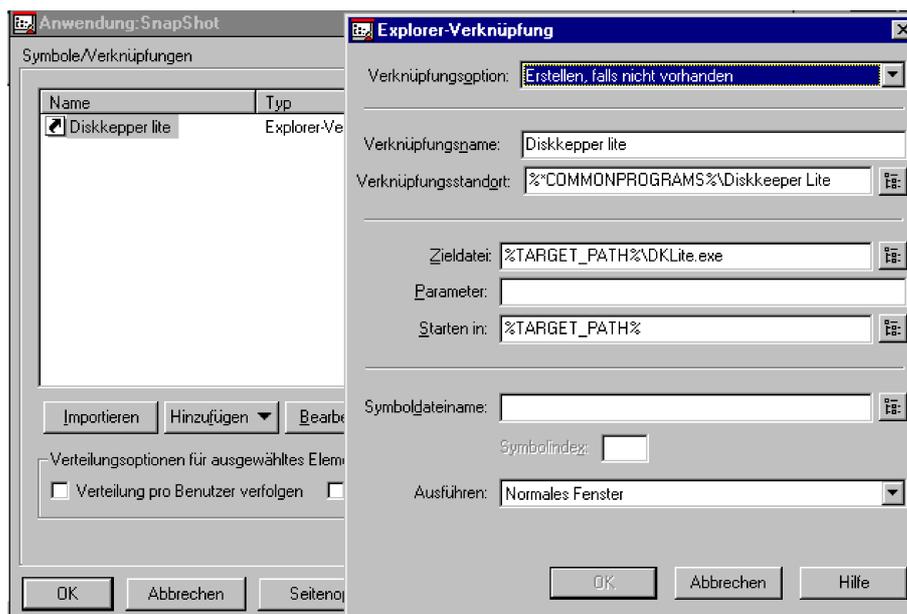


Abb. 57: Icons/Shortcuts

Ich will diese Möglichkeiten hier nicht näher vertiefen. Sämtliche Variablen kann man sich unter der Online-Hilfe zu *Makros* bei Applikationsobjekten im Administrator zeigen lassen. Im folgenden eine Zusammenfassung:

Beschreibung

Mit der Eigenschaftsseite "Makros" lassen sich die Anwendungsobjektmakros verwalten, die ausschließlich für dieses Anwendungsobjekt erstellt wurden und die auf anderen Eigenschaftsseiten des Anwendungsobjekts verwendet werden. Alle Makrotypen (einschließlich der Anwendungsobjektmakros) können an den folgenden Anwendungsobjektorten verwendet werden:

- Pfad zur ausführbaren Datei (Eigenschaftsseite "Identifikation")
- Befehlszeile (Eigenschaftsseite "Umgebung")
- Arbeitsverzeichnis (Eigenschaftsseite "Umgebung")
- Zuordnungspfad (Eigenschaftsseite "Laufwerke/Anschlüsse")
- Anschlußpfad umleiten (Eigenschaftsseite "Laufwerke/Anschlüsse")
- Eigenschaftsseite "Registrierungseinstellungen": "Schlüssel", "Name", "Wert" (nur "Zeichenkette")
- Eigenschaftsseite "INI-Einstellungen": "Gruppe", "Name", "Wert"
- Eigenschaftsseite "Anwendungsdateien": "Ursprung/Ziel", "Verzeichnis"
- Eigenschaftsseite "Textdateien": "Suchen" und "Zeichenkette hinzufügen"
- Eigenschaftsseite "Symbole/Verknüpfungen": überall

Makrotypen und Rangfolge

Ein Makro ist ein Variablenname. Der Wert einer Variablen wird an Stelle des Variablennamens eingesetzt. Wird die Variable nicht gefunden, so wird er nicht durch einen Wert ersetzt. Ein Makrowert ist in der Regel ein Zeichenkettenwert. Im folgenden werden die fünf verschiedenen Makrotypen oder Variablen aufgelistet, die mit dem Application Launcher verwendet werden können. Die Reihenfolge, in der die folgenden Makros erscheinen, entspricht ihrer Rangfolge (Anwendungsobjektmakros haben beispielsweise Vorrang vor Anmeldeskriptmakros).

Anwendungsobjektmakros

Eine Anwendungsobjekt-Makrovariable wird ausschließlich für die Eigenschaftsseite "Makros" eines bestimmten Anwendungsobjekts definiert. Der Makroname muß in %-Zeichen eingeschlossen werden. Zum Beispiel:

- %SOURCE_PATH% Standort der Ursprungsdateien
- %TARGET_PATH% Standort, an den Dateien kopiert werden sollen

Hinweis: Damit Sie die besten Ergebnisse zu erzielen, empfehlen wir, statt eines zugeordneten Laufwerks für den Ursprungspfad einen UNC-Pfadnamen zu verwenden. Wird eine Laufwerkzuordnung als Ursprungslaufwerk verwendet, werden möglicherweise einige Dateien nicht ordnungsgemäß kopiert.

Beispiel: Angenommen, Sie haben ein komplexes Anwendungsobjekt, in dem das Makro %SOURCE_PATH% auf mehreren Eigenschaftsseiten verwendet wird. Der Pfad, auf den %SOURCE_PATH% verweist, soll geändert werden. Anstatt alle Vorkommen einzeln anzupassen, ändern Sie den Wert von %SOURCE_PATH% einmal auf der Eigenschaftsseite "Makros"; die Änderung wird auf alle Eigenschaftsseiten übertragen, auf denen das Makro verwendet wird.

Aber: ich habe auch schon erlebt, dass Applikationen UNC-Pfade nicht verwenden können: so findet der Netscape Navigator 4.5 seine Java-Bibliotheken nicht mehr, wenn man für sein Programmverzeichnis einen UNC-Pfad angibt!!

Spezielle Windows-Makros

Ein spezielles Windows-Makro definiert Windows** 3.x-, 95- und NT-Verzeichnisse. Die unten aufgeführten typischen Pfade beruhen auf Standardinstallationen und entsprechen möglicherweise nicht Ihrem besonderen Setup. Auf Windows 95*-Arbeitsstationen verhalten sich Makros anders, wenn Benutzerprofile aktiviert sind.

Beispiel:

Angenommen, Sie haben Windows auf Laufwerk D: installiert (zum Beispiel D:WINDOWS). Eine Anwendung setzt jedoch voraus, dass Windows auf Laufwerk C: abgelegt ist (zum Beispiel C:WINDOWS). Durch Verwenden des speziellen Makros %*WinDisk% können Sie Laufwerk C: durch das Makro für die Dateien ersetzen, die es erforderlich machen. Siehe Eigenschaftsseite "Anwendungsdateien".

Spezielle Windows-Makros

Die folgenden Makros sind sehr nützlich für das Umleiten von Anwendungsdateien, die davon ausgehen, dass sich die Windows-Verzeichnisse an einem bestimmten Ort befinden.

Wichtig: Das Sternchen (*) ist für die Syntax dieser Makros erforderlich. Verwechseln Sie diese Sternchen nicht mit dem Sternchen in Novell-Marken.

%*WinDir%	Windows-Verzeichnis, in der Regel c:\windows oder c:\winnt
%*WinSysDir%	Windows-Systemverzeichnis, in der Regel c:\windows\system oder c:\winnt\system32
%*WinDisk%	Laufwerkbuchstabe (plus Doppelpunkt) für das Windows-Verzeichnis, in der Regel C:
%*WinSysDisk%	Laufwerkbuchstabe (plus Doppelpunkt) für das Windows-Systemverzeichnis C:
%*WinSys16Dir%	16-Bit-Verzeichnis von Windows NT** (c:\winnt\system)
%*TempDir%	temporäres Windows-Verzeichnis (c:\windows\temp)

Spezielle Windows 95- und NT-Makros

Die folgenden Variablen sind nur in Windows 95 und NT verwendbar. Diese Makros stammen aus dem Registrierungsschlüssel HKEY_CURRENT_USER\software\microsoft\windows\currentversion\explorer\shell folders.

Wichtig: Das Sternchen (*) ist für die Syntax dieser Makros erforderlich (wo es angezeigt wird). Verwechseln Sie diese Sternchen nicht mit dem Sternchen in Novell-Marken.

%*AppData%	Dateisystemverzeichnis als gemeinsame Ablage anwendungsspezifischer Daten. In der Regel c:\winnt\profiles\administrator\application data. Wird nur von Windows NT Version 4 verwendet.
%*WinDesktop%	Windows-Desktop-Verzeichnis (c:\windows\desktop oder c:\winnt\profiles\ <username>\desktop für Windows NT). Hinweis: Dies ist ein Application Launcher 2.0-Makro, das Abwärtskompatibilität berücksichtigt.</username>
%*Desktop%	Dateisystemverzeichnis zum physikalischen Speichern von Dateiobjekten auf dem Desktop (nicht zu verwechseln mit dem eigentlichen Desktop-Ordner). In der Regel c:\windows\desktop oder c:\winnt\profiles\administrator\desktop.
%*CommonWinDesktop%	Gemeinsames Desktop-Verzeichnis für Windows NT (c:\winnt\profiles\all users\desktop). Hinweis: Dies ist ein Application Launcher 2.0-Makro, das Abwärtskompatibilität berücksichtigt.
%*CommonDesktop%	Dateisystemverzeichnis mit Dateien und Ordnern, die auf den Desktops aller Benutzer angezeigt werden. In der Regel c:\winnt\profiles\all users\desktop. Falls nicht verfügbar, wird der Wert *DESKTOP verwendet.
%*Favorites%	Dateisystemverzeichnis als gemeinsame Ablage für die favorisierten Elemente des Benutzers. In der Regel c:\windows\favorites oder c:\winnt\profiles\administrator\favorites.
%*Fonts%	Virtueller Ordner mit Schriften. In der Regel c:\windows\fonts oder c:\winnt\fonts.
%*NetHood%	Dateisystemverzeichnis mit Objekten, die in der Netzwerkumgebung angezeigt werden. In der Regel c:\windows\nethood oder c:\winnt\profiles\administrator\nethood.
%*PrintHood%	Dateisystemverzeichnis als gemeinsame Ablage für Druckerverknüpfungen. In der Regel c:\winnt\profiles\administrator\printhood. Wird nur von Windows NT 4 verwendet.
%*Personal%	Dateisystemverzeichnis als gemeinsame Ablage für Dokumente. In der Regel c:\myfiles oder c:\winnt\profiles\administrator\personal.
%*Programs%	Dateisystemverzeichnis mit den Programmgruppen des Benutzers (die ebenfalls Dateisystemverzeichnisse sind). In der Regel c:\windows\start menu\programs oder c:\winnt\profiles\administrator\start menu\programs.
%*CommonPrograms%	Dateisystemverzeichnis mit den Verzeichnissen für die allgemeinen Programmgruppen, die im Startmenü aller Benutzer angezeigt werden. In der Regel c:\winnt\profiles\all users\start menu\programs. Falls nicht verfügbar, wird der Wert *PROGRAMS verwendet.

%*Recent%	Dateisystemverzeichnis mit den Dokumenten, die der Benutzer zuletzt verwendet hat. In der Regel c:\windows\recent oder c:\winnt\profiles\administrator\recent.
%*SendTo%	Dateisystemverzeichnis mit Menüelementen "Senden an". In der Regel c:\windows\sendto oder c:\winnt\profiles\administrator\sendto.
%*StartMenu%	Dateisystemverzeichnis mit Startmenü-Elementen. In der Regel c:\windows\start menu oder c:\winnt\profiles\administrator\start menu.
%*CommonStartMenu%	Dateisystemverzeichnis mit den Programmen und Ordnern, die im Startmenü aller Benutzer angezeigt werden. In der Regel c:\winnt\profiles\all users\start menu. Falls nicht verfügbar, wird der Wert *STARTMENU verwendet.
%*Startup%	Dateisystemverzeichnis, das der Programmgruppe "Start" des Benutzers entspricht. In der Regel c:\windows\start menu\programs\startup oder c:\winnt\profiles\administrator\start menu\programs\startup.
%*CommonStartup%	Dateisystemverzeichnis mit den Programmen, die im Ordner "Start" aller Benutzer angezeigt werden. Das System startet diese Programme immer, wenn sich ein Benutzer in Windows NT anmeldet oder Windows 95 startet. In der Regel ist dieses Verzeichnis c:\winnt\profiles\all users\startmenu\programs\startup. Falls nicht verfügbar, wird der Wert *STARTUP verwendet.
%*Templates%	Dateisystemverzeichnis als gemeinsame Ablage für Dokumentschablonen. In der Regel c:\windows\shellnew oder c:\winnt\shellnew.
%*TempDir%	Temporäres Windows-Verzeichnis (c:\windowst

Anmeldeskriptvariablen

Application Launcher unterstützt die bekannten Anmeldeskriptvariablen. Es werden jedoch nicht alle Anmeldeskriptvariablen unterstützt.

Unterstützte Anmeldeskriptmakros

Im folgenden werden die unterstützten Anmeldeskriptmakros und ihre Bedeutung aufgelistet. Alternative Makronamen werden in Klammern angegeben.

DAY	Numerischer Tag des Monats. Zum Beispiel: 01, 10, 15 etc.
FILESERVER (FILE_SERVER)	Name des NetWare-Dateiservers der von NDS* überwachten Verbindung. Zum Beispiel: APPS_PROD.
FULL_NAME	Vollständiges Namensattribut des Benutzerobjekts. Zum Beispiel: Gisela Müller. Hinweis: Dies wird unterstützt durch die Unterstützung von Application Launcher für NDS-Attribute aus dem Benutzerobjekt.
HOURL24 (24HOUR)	Uhrzeit im 24-Stunden-System. Zum Beispiel: 02, 05, 14, 22 etc.
HOUR (HOURS)	Uhrzeit. Zum Beispiel: 0 = 12, 13 = 1 etc.
LAST_NAME	Nachname des aktuellen Benutzers (auch als NDS-Nachnamensattribut des Benutzers bekannt). Zum Beispiel: Meier.
LOGIN_NAME	Die ersten acht Byte des NDS-Objektnamens des Benutzers. Zum Beispiel: fhuber.
MINUTE (MINUTES)	Aktuelle Minute. Zum Beispiel: 02, 59 etc.
MONTH	Aktuelle Monatszahl. Zum Beispiel: 01 für Januar etc.
NDAY_OF_WEEK	Numerischer Tag der Woche. Zum Beispiel: 1 für Sonntag etc.
NETWORK (NETWORK_ADDRESS)	Netzwerkadresse der Arbeitsstation. Zum Beispiel: 01010120
OS_VERSION	Version des Betriebssystems (OS). Zum Beispiel: v5.00 (Win3 zeigt DOS-Version, Win 95 und NT zeigen Windows-Version)
Betriebssystem	Betriebssystemtyp. Zum Beispiel: MSDOS, WIN95, WINNT etc. (Win3 zeigt MSDOS.)
PLATFORM	Ausgeführte Plattform. Zum Beispiel: WIN, W95, WNT etc.
PHYSICAL_STATION (P_STATION)	MAC-Adresse. Zum Beispiel: 0000C04FD92ECA.
REQUESTER_CONTEXT	Kontext des Requester (für den ausgewählten Baum).
SECOND (SECONDS)	Anzahl der Sekunden. Zum Beispiel: 03, 54 etc.
SHORT_YEAR	Kurze Jahreszahl (zweistellig). Zum Beispiel: 97, 00 etc.
WINVER	Windows-Version. Zum Beispiel: v3.11, v4.00 etc.
YEAR	Vollständige Jahreszahl (vierstellig). Zum Beispiel: 1997 etc.

Nicht unterstützte Anmeldeskriptmakros

Die im folgenden aufgelisteten Anmeldeskriptmakros werden von Application Launcher NICHT unterstützt:

ACCESS_SERVER
AM_PM
DAY_OF_WEEK
DIALUP
ERROR_LEVEL
GREETING_TIME
LOCATION
LOGIN_ALIAS_CONTEXT
LOGIN_CONTEXT
MACHINE
MONTH_NAME
NEW_MAIL

OFFLINE
 PASSWORD_EXPIRES
 REQUESTER_VERSION
 SHELL_VERSION
 SMACHINE
 STATION
 USERID

NDS-Attribut-Makros

Beschreibung

Application Launcher unterstützt Makros, die Informationen aus den Attributen des aktuell angemeldeten Benutzers, des aktuellen Anwendungsobjekts oder aus den Attributen anderer NDS*-Objekte beziehen.

Beispiel:

Ein GroupWise*-Anwendungsobjekt führt OFWIN.EXE mit einem Befehlszeilenparameter aus:

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

USERNAME kann durch ein Makro ersetzt werden, das den NDS*-Eigennamen (Common Name, CN) des Benutzers verwendet:

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

Falls der NDS-Objektname gleich der Email-Anmeldung für GroupWise ist, wird für jeden Benutzer, der die Anwendung ausführt, der richtige Benutzername an GroupWise weitergegeben.

NDS-Attributvariablen für aktuellen Benutzer

Variablen, die durch ein Attribut in einem NDS-Objekt definiert sind, können hinzugefügt werden. Zum Beispiel:

%CN%	Eigennamen (Objektname oder Anmeldename des Benutzers)
%DN%	vollständiger eindeutiger Name des Benutzers (wird nur im Application Launcher verwendet)
%Given Name%	Vorname
%Surname%	Nachname
%Full Name%	Vollständiger Name
%Telephone Number%	Telefon
%Home Directory%	Stammverzeichnis
%Email Address%	Email-Adresse
%Mailbox ID%	Mailbox-ID

Hinweis: Als Standard ist der aktuell angemeldete Benutzer das Ursprungs-NDS-Objekt für diese Attribute.

NDS-Attribute aus anderen Objekten

Verwendung: %.OBJNAME.CONTAINER;NDSATTRIB%, als Beispiel: %.UserName.Novell;Description% wobei das Beschreibungsattribut des Objekts username.novell gezogen wird.

NDS-Attribute aus "Aktuelle Anwendungsobjekte:"

Verwendung: %*;NDSATTRIB%, Beispiele umfassen

%*;DN%	DN des Anwendungsobjekts (wird nur mit Application Launcher verwendet)
%*;App:Caption%	Anwendungssymboltitel
%*;App:Path%	Pfad zu ausführbarer Datei
%*;App:GUID%	GUID
%*;App:Version String%	Versionsstempel

Umgebungsvariablen

Application Launcher unterstützt alle Umgebungsvariablen. Einige Beispiele:

```
%NWLLANGUAGE%  

%TEMP%  

%PATH%
```

Hinweis: Der Wert der Variablen darf die Länge des Anwendungsobjektnamens nicht überschreiten, andernfalls schlägt die Variable fehl.

Im Fenster *Anwendungsdateien* lässt sich einsehen, welche Dateien bei einer Installation wohin kopiert werden. Hier lassen sich einzelne Dateien löschen - sie werden dann nicht kopiert - und zusätzliche Dateien einfügen - falls ein snAppShot nicht alle Komponenten „entdeckt“ hat, die benötigt werden. Die Dateien im Installationsverzeichnis im Netzwerk (dem Verzeichnis, aus dem der NAL die Dateien kopiert) sind von 1.FIL bis n.FIL durchnummeriert. Ihre tatsächlichen Namen stehen in einer ebenfalls dort befindlichen Datei FILEDEF.TXT

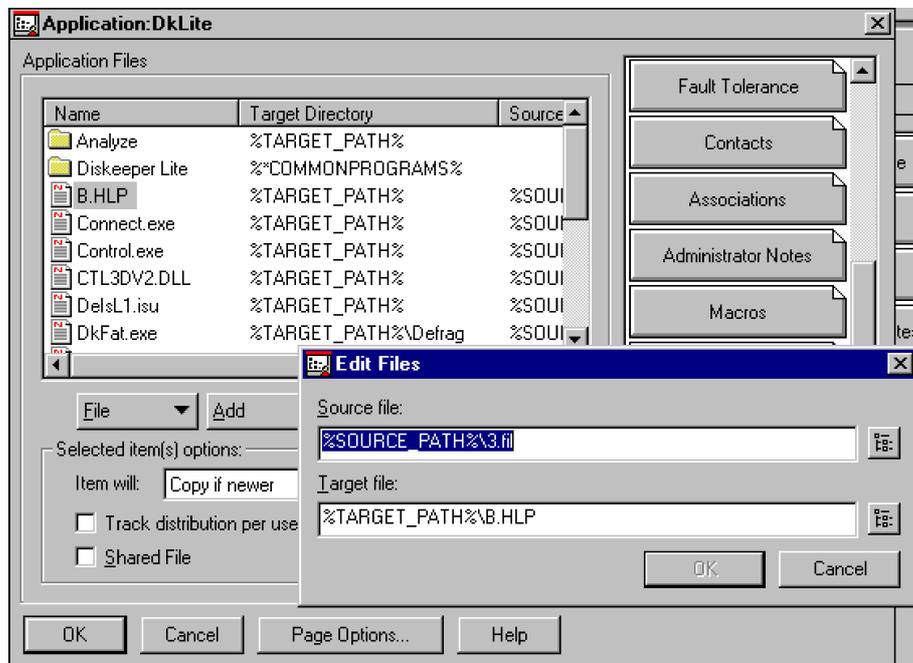


Abb. 58: Auflistung der installierten Dateien

Der NAL 2.7 bietet noch eine weitere Möglichkeit, die automatische Installation nach dem Schnappschuss zu editieren. Zusätzlich zur der .AOT (Template-) Datei, die Binärformat hat, wird beim snAppShot eine .AXT- (Text-) Datei erzeugt, die sich manuell ergänzen oder bearbeiten lässt. Sie wird anschließend (im Administrator unter *<Werkzeuge> Application Launcher Werkzeuge > AOT/AXT-Datei-Werkzeuge > AXT -> AOT*) als .AOT-Datei importiert und kann dann als Basis für ein *Applikationsobjekt* dienen. Eine solche .AXT-Datei für den hier besprochenen snAppShot sieht wie folgt aus (Auszüge):

AXT_FILE 2.5

[Application Date]
Value=36052

[Application Time]
Value=1

[Application Name]
Value=DkLite

[Application Caption]
Value=Disk Keeper Lite

[Application Flags]
Flag=Install Only

[Macro]
Name=SOURCE_PATH
Value=X:\NALLIB\DkLite

```
[Macro]
Name=TARGET_PATH
Value=C:\ExecSoft\DkLite

[Registry Key Create]
Flag=Write Always
Key=HKEY_LOCAL_MACHINE\SOFTWARE\Executive Software\Diskeeper Lite\1.1

[Registry Value Create]
Type=String

Flag=Write Always
Key=HKEY_LOCAL_MACHINE\SOFTWARE\Executive Software\Diskeeper Lite\1.1

Name=BuildNumber
Value=0.127

.....

[Directory Create]
Flag=Write Always
Directory=%*COMMONPROGRAMS%\Diskeeper Lite

[File Copy]
Flag=Update Create
Source=%SOURCE_PATH%\1.fil
Target=%TARGET_PATH%\Analyze\ScanFat.exe

[File Copy]
Flag=Update Create
Source=%SOURCE_PATH%\2.fil
Target=%TARGET_PATH%\Analyze\ScanNtfs.exe

[Shortcut Add]
Link File Path=%*COMMONPROGRAMS%\Diskeeper Lite\Diskeeper Lite Help.Ink
EXE Path=%TARGET_PATH%\DKLITE.HLP
Command-line=
Icon Path=
Icon Index=0
Working Directory=%TARGET_PATH%
Show Command=Normal
Separate Memory Space=No

[Shortcut Add]
Link File Path=%*COMMONPROGRAMS%\Diskeeper Lite\Diskeeper Lite.Ink
EXE Path=%TARGET_PATH%\DkLite.exe
Command-line=
Icon Path=
Icon Index=0
Working Directory=%TARGET_PATH%
Show Command=Normal
Separate Memory Space=No

.....
```

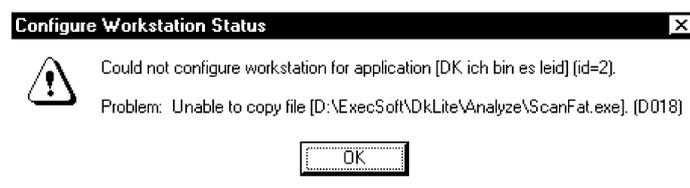
Abschließende Hinweise und Tips

Zwei Dinge sind noch wichtig:

1. Wenn man während der Installationen Löschungen vornimmt, werden diese Papierkorbveränderungen mit in das Applikationsobjekt übernommen - es ist ja eine Veränderung auf dem Rechner! Man kann diese "Müll-" Objekte im Eigenschaftsfenster Application Files (Abb. 58) an *RECYCLER* ... erkennen und löschen. Besser noch man nimmt gar keine Löschungen vor dem zweiten Schnappschuss vor.
2. Manche Applikationen kommen mit der parallelen Verwendung von Laufwerksbuchstaben und UNC-Pfaden nicht klar (z.B. Netscape), weil sie intern Laufwerksbuchstaben in UNC-Pfade umwandeln, ohne dass man es merkt! Im Zweifelsfall nur UNC-Pfade verwenden.

Eine öfters auftretende Fehlermeldung ...

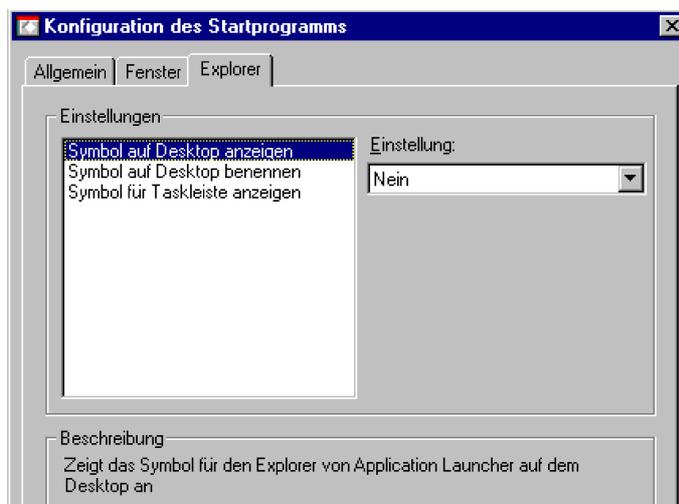
Die untenstehende Fehlermeldung ist etwas missverständlich. Sie bedeutet nämlich (meistens) nicht, dass die Datei nicht auf ein lokales Laufwerk kopiert werden kann, sondern dass das Makro SOURCE_PATH falsch ist. Meist verweist es auf ein Netzwerk-Laufwerk, auf das der Administrator beim Arbeiten mit SnAppShot zwar ein Mapping mit Laufwerksbuchstaben hatte, nicht aber der Anwender, der das Application Object aufruft.



LÖSUNG: SOURCE_PATH im Administrator für das Objekt (unter Makros) mit **UNC-Pfad** angeben oder einen Laufwerksbuchstaben für die Root der Festplatte vergeben (und den nie mehr ändern oder entfernen).

... wie verschwindet das Application Launcher-Symbol vom Desktop?

Ganz einfach: beim Benutzer unter *Konfiguration des Startprogramms* auf *Bearbeiten* klicken und folgendes Setting vornehmen:

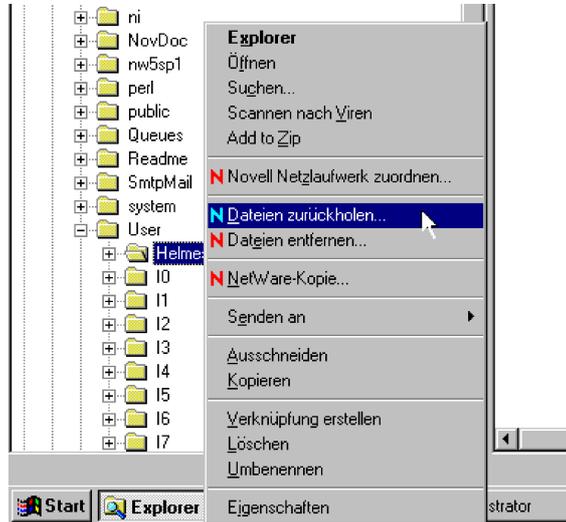


Batchdateien in Skripten und bei Aktionen

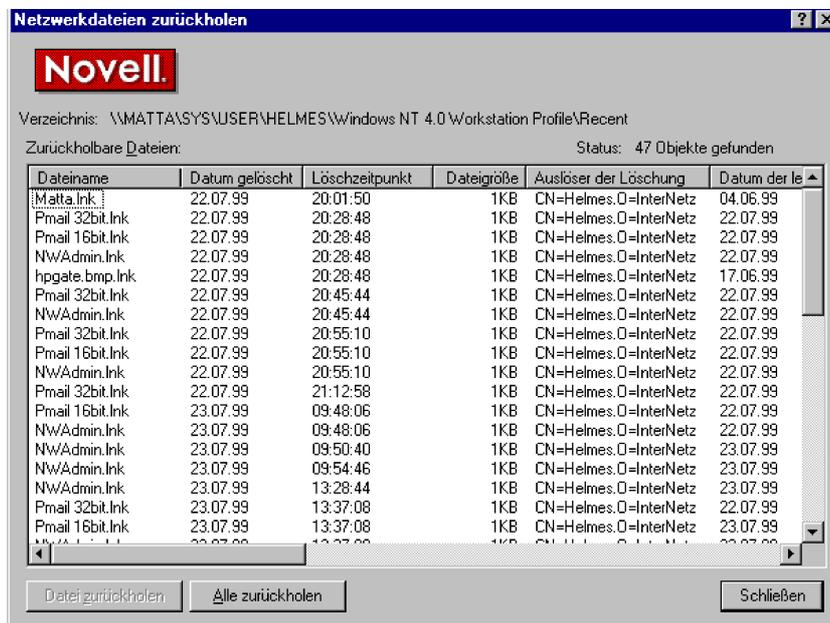
Ganz wichtig: Batchdateien sind ASCII-Texte und daher nur bei "Anwesenheit" eines Command-Prozessors ausführbar. (Login-) Skripten (auch in Applikationsobjekten) und Aktionen (im Planer) verlangen bei Programmaufrufen ausführbare Dateien - deshalb muß in beiden Fällen zusammen mit der Batchdatei eine Instanz des Command-Prozessors geladen werden. Dies geschieht **unter NT** mit **cmd /c <Batchdatei>** und **unter Win3.1/95/98** mit **command /c <Batchdatei>**. In Skripten muß dem ganzen das Doppelkreuz vorangestellt sein (also **#cmd /c <Batchdatei>**). Bei mehreren Plattformen im Netz muß man evtl. mit Hilfe der Novell-Variablen OS abfragen, welche Plattform lokal vorliegt (**if OS="WINNT" then ... else ... end**).

Dateien löschen und wiederherstellen

... ist unter ZENworks (endlich) so organisiert, dass auch ein "normaler" Benutzer diese Tätigkeit durchführen kann. Man benötigt dazu nicht mehr den Administrator, sondern klickt einfach im Explorer auf dem entsprechenden Verzeichniseintrag die rechte Maustaste und wählt dort Dateien zurückholen bzw. Dateien entfernen:



Danach erscheint ein aus dem Administrator bekanntes Menü, das die Auswahl von Dateien und deren Wiederherstellung anbietet:



Mit dieser Methode können jedoch nach wie vor nur Dateien aus dem aktuellen Verzeichnis zurückgeholt werden. Wurden komplette Verzeichnisstrukturen gelöscht, muss der Administrator oder ein Vertreter eingreifen, der Zugang zu dem Verzeichnis <VolumeName>\DELETED.SAV hat. Dort „landen“ nach wie vor alle Dateien, deren Ursprungsverzeichnis nicht mehr existiert.

Zusammenfassungen

1. Arbeitsschritte bei der Installation von Z.E.N. works

Auf dem Server

- Installation von Z.E.N. works (ZWNT46DE). Das Archiv muß zunächst „entpackt“ werden - es ist selbst-extrahierend. HINWEIS: das Verzeichnis, in das entpackt werden soll, muß nahe an der Root liegen (z.B. SYS:ZENWORKS), da z.T. tiefe Verzeichnisstrukturen erzeugt werden.
- Aufruf des Installationsprogramms WINSETUP im Verzeichnis, in das entpackt wurde. Dabei wird die NDS im gewählten Baum im Schema erweitert und der neue 32bit-Administrator installiert.
- Einrichten von *Richtlinienpakete* für Benutzer und Workstations. Man muß mindestens ein *WINNT Benutzerpaket* einrichten, damit die Benutzerprofile zentral gespeichert (*Zentral gespeichertes Profil / Roaming Profiles*) und die Benutzer nicht lokal eingerichtet werden müssen (*Dynamischer lokaler Benutzer*) sowie ein *WINNT Arbeitsstationspaket*, damit die Konfiguration des Client für Windows NT zentral gesteuert werden kann (*Client Configuration*) - sonst muß sie auf allen Workstations lokal vorgenommen werden.

Auf den Workstations

- Installation des Z.E.N. works-Client für Windows NT auf den Workstations. Sie befinden sich ebenfalls im oben genannten Paket.
- Aufruf von WINSETUP - wie oben - oder SETUPNW im Verzeichnis ..\PRODUCTS\WINNT\I386.
- Registrierung, Import und Re-Registrierung aller Workstations in die NDS

2. Vorgehensweise bei der Erzeugung eines Applikationsobjekts

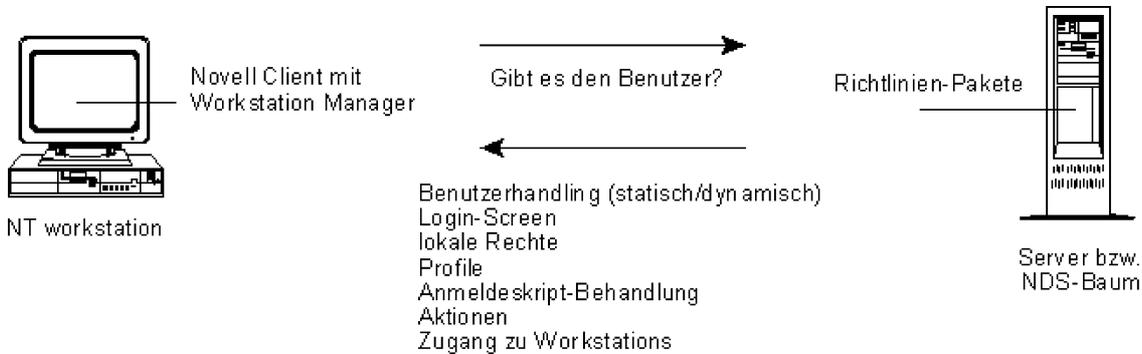
A) Vorbereitung

- bei **Netzwerk-basierten Programmen**: (Server-) Installation des Programms von einer beliebigen Station aus auf einem Server im Netzwerk
- **sonst**: Kopieren der Programmkomponenten und des zugehörigen Setup-Programms in ein Netzwerkverzeichnis

B) Erzeugen des Objekts

- bei **einfachen** Programmen (siehe dazu die Anmerkung auf Seite 37)
 - Aufruf des Administrators, Erzeugung eines *Simple Application Objects*
- **sonst**:
 - Aufruf des snAppShot-Programms unter dem Betriebssystem, unter dem das Programm später aufgerufen werden soll, Erzeugung eines Application Object Templates (bei Netzwerkprogrammen mit dem **Workstation-Setup**, bei lokalen Programmen mit dem „normalen“ Setup). **HINWEIS**: Soll das Programm unter 16- und 32bit laufen, muß je ein Template für jede Plattform erzeugt und die Applikationen unter *Systemanforderungen* entsprechend freigeschaltet werden.
 - evtl. Start und Konfiguration des soeben installierten Programms
 - danach: Wechsel in ein 32bit-System, Aufruf des Administrators. Dort: Erzeugung des Objekts, Freigabe für die Benutzer, Ergänzung notwendiger zusätzlicher Eigenschaften (u.a. *Dateirechte* - Rechte an Dateien und Verzeichnissen)

Ablauf 1: Was geschieht beim Einloggen?



Anmeldung

- Kontrolle, ob es den Benutzer gibt
- Kontrolle, ob er ein Passwort hat und sich anmelden darf (Zeit / Knoten usw.)
- falls es eine Arbeitsstationsrichtlinie gibt: Kontrolle, ob sich der Benutzer auf der entsprechenden Station einloggen darf; evtl. Start von der Station zugeordneten Aktionen
- Kontrolle, ob der Workstation Manager aktiv ist und ihm ein Baum zugeordnet wurde - sonst gesonderte lokale Anmeldung nötig
- falls aktiv und zugeordnet: Kontrolle, ob ihm Benutzer-Richtlinien zugeordnet sind oder eine Suchrichtlinie für ihn gilt
 - falls ja
 - falls dynamischer user
 - falls temporärer Benutzer: Workstation Manager richtet den Benutzer nach Maßgabe des Benutzer-Richtlinienpaketes auf der Workstation ein
 - sonst: Workstation Manager kontrolliert, ob der Benutzer "lokal" existiert
 - falls ja: er übernimmt die lokalen Einstellungen
 - falls nein: der Benutzer wird nach den Maßgaben der Richtlinie eingerichtet
 - falls nicht dynamischer user
 - falls der Benutzer lokal existiert, wird er lokal und im Baum gleichzeitig angemeldet
 - falls der Benutzer nicht existiert, kann er sich evtl. lokal unter einem anderen Namen anmelden. Kennt er lokal kein Konto, kann er sich nicht einloggen.
 - evtl. Start von dem Benutzer zugeordneten Aktionen

Profile

- falls Roaming Profile: das Profil wird vom Heimatverzeichnis auf die Workstation überspielt
- sonst: das lokal vorhandene Profil wird benutzt oder ein neues Profil auf der Basis des "Default User" auf der entsprechenden Workstation erstellt.

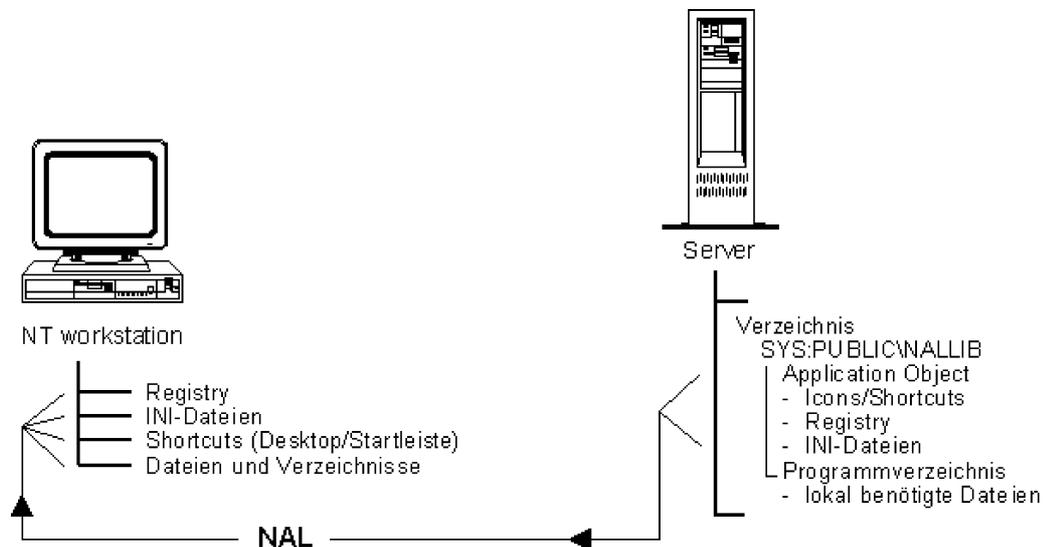
Abmeldung

- falls dynamischer user
 - falls temporärer Benutzer: der Benutzer wird auf der Workstation gelöscht.

Profile

- falls Roaming Profile: das lokale Profil wird auf das Heimatverzeichnis kopiert.
- falls temporärer Benutzer: das Profil wird lokal gelöscht

Ablauf 2: Was macht der NAL beim Starten einer Anwendung?



Zunächst

- Kontrolle, ob lokal alle Bedingungen für den Start des Programms gegeben sind
- Benutzerberechtigungen
- Systemvoraussetzungen / Zeitbeschränkungen
- falls ja: das Programm wird angeboten
- falls nein: das Programm wird gar nicht angeboten

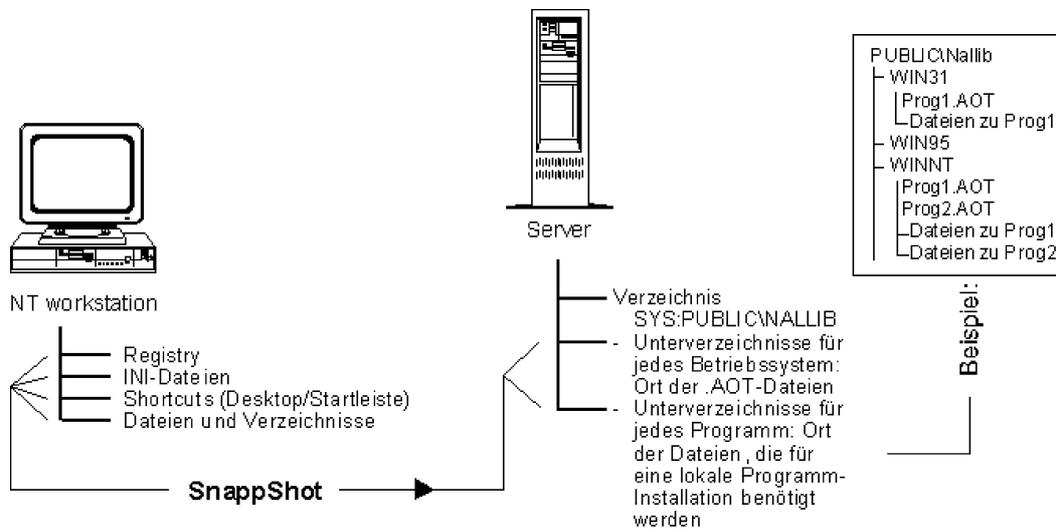
Beim Anklicken eines Eintrags in der Startleiste/dem Desktop/dem System-Tray

- Kontrolle (in der Registry des Benutzers), ob das Programm bereits auf dem Rechner per NAL installiert wurde
 - falls ja:
 - falls Skripts vorhanden: Abarbeitung des Startskripts
 - Gewährung der Verzeichnis- und Dateirechte
 - Mappings von Laufwerken, Druckerumleitungen, Verbindung zu den entsprechenden Servern aufnehmen
 - Start des Programms
 - falls nein: Installation des Programms
 - Kopieren der lokal benötigten Programmteile vom Server auf die Workstation
 - Modifikation der Registry
 - Erzeugung / Erweiterung der INI-Dateien (bei 16bit-Programmen)
 - Erstellung der Shortcuts (Startmenü/Desktop/System-Tray)
- falls entsprechend konfiguriert: Start des Programms
- **falls Probleme beim Start auftreten:** der Programmstart wird abgebrochen
 - dem Benutzer wird angeboten, ein "Verify", d.h. eine Neu- oder Nach-Installation des Programms zu veranlassen.

Beim Beenden des Programms

- falls Skripts vorhanden: Abarbeitung des Skriptes für das Programmende
- evtl. Sonderverbindung zu Servern und Druckern beenden
- evtl. Berechtigungen zu Dateien und Verzeichnissen aufheben

Ablauf 3: Was macht der SnAppShot?



Vorarbeiten

- falls netzwerkbasierendes Programm
- Netzwerk-Installation des Programms

Beginn

- 1 Start des Programms SnAppShot
 - a Festlegung des Application Objekt-Namens und der Unterschrift des Icons bzw. der Beschriftung im Startmenü
 - b Festlegung, wie das Application Object Template heißen und wo es abgelegt werden soll
 - c Festlegung, wo die Dateien, die für eine Installation über den NAL notwendig sind, abgelegt werden sollen
 - d Festlegung, welche Dateien und Verzeichnisse beim folgenden Installationsvorgang "beobachtet" werden sollen
- 2 **1. Schnappschuss** von Dateien, Registry, INI-Dateien und Shortcuts
- 3 **Installation des Programms** - rein lokal oder als Arbeitsplatzversion
 - a evtl. Aufruf und Konfiguration des Programms (Desktop / Dateiablage usw.)
 - b evtl. Löschen von Shortcuts im Profilbereich "All Users" und Kopie in den Bereich des aktuellen Users
 - c Beenden des Programms, Rückkehr zum SnAppShot
- 4 **2. Schnappschuss**
 - dabei: Kontrolle
 - a welche Dateien in dem Verzeichnis, wo das Programm eben installiert wurde, "gelandet" sind
 - b welche Veränderungen in den zu beobachtenden Dateien vorgenommen wurden
 - c welche Eintragungen in die Registry vorgenommen wurden
 - d welche INI-Dateien neu angelegt oder verändert wurden
 - e welche Icons und Eintragungen in die Startleiste vorgenommen wurden
 - Kopie aller "neuen" oder veränderten Dateien in das Verzeichnis, das unter 1c angegeben wurde
 - Kopie aller anderen Änderungen in das Application Object Template

Danach kann das Application Object mit dem Administrator erstellt und angepasst werden.

Index

Administrator 3-5, 9-12, 20, 22-25, 31, 32, 35, 37, 41-43, 45, 47-49, 52, 54-60, 62
Application 3, 4, 10, 19, 23, 25-28, 30-32, 35-39, 41-47, 49, 52, 54, 57, 59-61, 63-66
Application Launcher 3, 4, 19, 23, 26-28, 31, 35, 39, 41-45, 47, 57, 59-61, 63, 65, 66
Applikation 24-26, 28, 32, 37, 39, 40
Arbeitsstationspaket 6, 11, 17, 49
Benutzer 7, 9-12, 14-19, 21-29, 31, 32, 34, 35, 37, 39, 40, 42-44, 47-51
Benutzerpaket 14, 49
Container 3, 9, 10, 18, 19, 23, 24, 29, 30, 37, 39, 44, 54-60, 62, 63
Drive 27, 57, 64, 65
Folder 25, 30, 40, 63
Group 3, 10, 54, 55, 59, 62
Gruppe 10, 11, 18, 30, 39, 41
Icon 8, 19, 23-25, 32, 39, 40, 46, 63
Import 3, 9, 49, 54-56, 58-60
INI 23, 31, 35, 37, 41, 51, 52, 63, 64
Laufwerk 28, 33, 42, 47
License 19
Lizenz 9
Macros 62
Makros 35-37, 40-42, 44, 47
NAL Explorer 23-25
NDS 3-5, 8-10, 12, 14, 16, 18-20, 22, 23, 28, 29, 32, 43, 44, 49, 54-57, 59, 60, 62, 63, 67
NT Konfiguration 5, 6
Ordner 30, 35, 37, 42, 43
Planer 8, 9, 19, 22, 47
Policies 3, 6, 13, 16, 54-56, 58, 67
Policy 3, 5, 6, 8-10, 12, 16, 17, 54-56, 58, 59
Policy Package 3, 6, 8, 10, 54-56, 58, 59
Port 59, 60
Profile 15, 17, 50
Profiles 3, 15, 42, 43, 49
Registrierung 9, 17, 21, 29, 49
Registry 5, 11, 25, 26, 31, 32, 37, 39, 40, 51, 52, 58, 60-65
Richtlinie 9-12, 14-18, 22, 50
Richtlinienpaket 10, 19, 20
Scheduler 8, 9, 19, 57, 59-61
Taskleiste 8, 19, 23, 24, 32, 40
User 3, 8, 9, 15, 22, 24, 30, 42, 50, 54-59, 61, 63, 65, 66
Verknüpfung 10, 11, 28
Verzeichnis 4, 5, 12, 25, 28, 31, 32, 34, 35, 38, 41-43, 45, 48, 49, 51, 52, 67
Workstation 3, 5-12, 14, 16, 18, 19, 22, 34, 49-51, 54-67
Workstation Group 10, 54, 55
Workstation Object 3, 9, 19, 54, 55, 57, 58, 60-62
Workstation-Installation 34

Anhänge

ANHANG 1: Getting Started

Requirements

NetWare 5 contains the Z.E.N.works Starter Pack. Before you attempt to install and use the Z.E.N.works Starter Pack, ensure that you have:

- NetWare 4.11 or higher running on the server 70 MB of available memory and 205 MB available disk space on the NetWare server
- The Supervisor right to the NetWare server where you install Z.E.N.works
- The Supervisor right to the NDS container where you install Z.E.N.works
- Rights to modify the schema of the NDS tree you are installing Z.E.N.works in
- Z.E.N.works client software on each workstation participating in Z.E.N.works

ANHANG 2: Essential Concepts

This section provides you with a foundation in Z.E.N.works, and explains the following concepts.

NDS Objects Unique to Z.E.N.works
How Workstation Objects are Created
Using Workstation Group Objects
What Is the Workstation Registration Process?
Policies and Policy Packages
Effective Policies

NDS Objects Unique to Z.E.N.works

The power of Z.E.N.works lies in its use of new NDS objects, (Workstation, Workstation Group, and Policy Package objects), and its use of existing NDS objects (User, Group, Organizations, Organizational Units, and Application objects).

Workstation objects are created when you register and import workstations in your NDS tree. They represent physical workstations and serve many functions, such as letting you collect workstation inventory and remote control (full product only).

You can use NetWare Administrator 32 to create Policy Packages and associate them at the User, Group, Workstation, Workstation Group, or container level. Within Policy Packages are Policies that you can enable and apply to one or more objects.

For example, a Windows 95 User Policy Package contains Policies that allow you to configure the desktop and set up login restrictions. Once you enable one or more Policies in a User Policy Package, all Users--and Users within Groups or containers--associated with the Policy Package will be affected by the enabled Policies in the package.

How Workstation Objects are Created

Unlike most NDS objects, Workstation objects are created automatically after all of the following occur:

1. Users must have the Write right to the WM:Registered Workstation attribute on the parent container (the container where the Workstation object will be created).

The wsrights.exe accomplishes this automatically for all existing containers during the Z.E.N.works installation. You can set these rights manually using the Prepare Workstation Registration utility in the Tools menu, or by running wsrights.exe explicitly from sys:\public.

NOTE: If you add a container after installing Z.E.N.works, you need to use one of the methods described here to set up rights to the WM:Registered Workstation attribute in the container manually.

2. A workstation registers with NDS.
The registration program creates a Workstation entry in the Workstation Registration page of the container where the logged in user's User object resides
3. The registered workstation is imported into a container object.

The import process adds a Distinguished Name (DN) to the Workstation entry, and then adds a Workstation object to the tree, automatically naming it according to the name format specified in the Workstation Import Policy in the User Policy Package associated with the user logging in.

4. Once a Workstation entry is imported, the Workstation Registration program must run again to find the Workstation object's DN, store the DN locally on the workstation, and then update the Workstation object.

Now the workstation is synchronized with its NDS object, and each time the Workstation Registration program runs on this workstation, it will update the Workstation object with the following information: registration time, network address, last server, and last user

Using Workstation Group Objects

A Workstation Group object lets you manage or maintain a group of workstations. By creating Workstation Group objects according to various group needs, you reduce the time it takes to maintain software versions and workstation configurations.

For instance, if a group of Windows NT workstations contain the same word processing software, they might all need to be upgraded at the same time to the newest version of the software.

You can create a Workstation Group object called Windows NT and add the workstations to the group. Then, when you need to make a change, apply the change to the Workstation Group object instead of the individual Workstation objects.

To associate a Workstation with a Workstation Group, either add a Workstation to the Members page of the Workstation Group object, or add a Workstation Group to the Workstation Groups page of the Workstation object.

What Is the Workstation Registration Process?

Workstations must register with NDS before they can be imported into the tree and managed as NDS objects. When the Workstation Registration program is run, the workstation sends its registration time, network address, last server, and last user information to NDS, and places a Workstation entry in the Workstation Registration page of the container.

The registered workstation must be imported before an object will be created in the tree. Once the workstation is imported, the Workstation Registration program must run again to complete the following.

- Finds the Workstation object's Distinguished Name (DN).
- Stores the DN locally on the workstation.
- Updates the Workstation object and synchronizes the workstation with NDS.

Once the workstation is synchronized with NDS, each time the Workstation Registration program runs on this workstation, it will update the Workstation object.

Policies and Policy Packages

Z.E.N.works Policies are NDS objects that are created automatically when you choose to create a Policy Package object using NetWare Administrator.

Each Policy contains one or more pages where you can specify settings or configurations related to User, Workstation, Group, or container objects, depending on the type of Policy.

A Policy Package contains Policies which have been grouped, for ease of administration, according to the types of objects the Policies can be associated with. You can use Policy Packages and NetWare Administrator to maintain Workstation objects and other NDS objects associated with workstations.

There are three basic types of Policy Packages: Container, User, and Workstation. However, since the User and Workstation Packages are platform-specific (Windows 3.1, Windows 95, or Windows NT), there are actually seven Policy Packages to choose from. Policies must be enabled from within a Policy Package before they become effective. The checked boxes on the left at the Policies screen indicate, which Policies have been enabled.

NOTE: Policies are added to a Policy Package when their .dlls are added to the Z.E.N.works product. Policy Packages vary by platform and by Z.E.N.works product. For example, the sample Policies page in the previous figure is from a Windows 95 User Package in the full Z.E.N.works 1.0 product. Since the Remote Control Policy .dll does not ship in the Z.E.N.works 1.0 Starter Pack, the Policies page in the Z.E.N.works 1.0 Starter Pack won't list the Remote Control Policy.

When you want the settings in a Policy Package to apply to an object, you must associate the Policy Package with the object. One way to do this is to choose the Policy Package's Associations page and add the object to the Associations list.

Effective Policies

Much like Effective Rights in NDS, Z.E.N.works has Effective Policies. The effective Policies for any object is the sum of all of the enabled Policies in all Policy Packages associated directly or indirectly with that object.

The system looks up the tree for effective Policies (assuming you use the default search order, which starts at the leaf objects and goes up towards the root of the tree), just as the system looks up the tree for effective rights. The first enabled Policy it finds sets the rule.

For example, if the Desktop Preferences Policy is not enabled in a User Policy Package associated with the User object, but the Desktop Preferences Policy is enabled in the User Policy Package associated with the container where the User objects reside, the enabled Desktop Preferences Policy is the effective Policy for the user.

ANHANG 3: How to Register a Workstation

by Brian Vaughn and Jayne Mooney

One of the truly cool values of Z.E.N.works is the ability to represent all those boxes on your network as objects in NDS. Stop and think about what that means for a minute: You won't have to embark on a pilgrimage to the nether regions of your company to get information about some troublesome PC. Now, with Z.E.N.works Workstation objects, you can, from the comfort of your own office, look up a workstation's vital statistics with a few clicks in your NDS tree. Is the Directory awesome, or what?

Before You Begin

We highly recommend you at least read (and complete, if possible) the steps required for Z.E.N.works setup. To view or print these setup instructions:

1. Install Z.E.N.works.
2. Launch NetWare Administrator (nwadmn32.exe).
3. Choose Help > Help Topics > Z.E.N.works > Setting Up Z.E.N.works.
4. View or print these topics:

Set Up Z.E.N.works

Set Up Workstations for Z.E.N.works

Topic Overview

Before you can realize the benefits of Z.E.N.works, of course, you need to first register those workstations in the container where the User object resides. Registering a workstation involves the following tasks:

Granting User Rights to Register Workstations
Registering Workstations for Importation
Creating and Associating a Workstation Import Policy to a User
Importing the Registered Workstations
Re-registering Workstations

These two sections explain some management tasks pertaining to registered workstations:

Un-registering Workstations or Starting Over
Moving or Renaming Workstation Objects in the Tree

Granting User Rights to Register Workstations

Before the workstation registers to the user's container, you need to prepare the NDS container. To prepare a container for registration, do one of two things:

1. Run WSRights.exe during install.
2. Using NetWare Administrator (NWAdmn32), highlight a container and then choose Tools > Workstation Utilities > Prepare Workstation Registration. Once you have selected this option, the container is ready for workstations to begin registering.

Registering Workstations for Importation

You can register a workstation for importation in one of three ways:

1. Install the Z.E.N.works client. This is the automated way of registering a workstation.
2. If you don't have the Z.E.N.works client, then you can run WSREG.EXE from a login script.
3. Run the WSREG Application object using the Application Launcher (NAL).

What is the Workstation Agent?

The workstation registration agent comes in two forms: WSREG.DLL and WSREG16.EXE.

WSREG.DLL is the workstation registration agent for WIN32 platforms.

WSREG.DLL can run in the Workstation Manager scheduler, or it can be run with WSREG32.EXE in a login script or through NAL. It has a resource DLL, named WSREG32R.DLL, that it uses to get internationalized resource strings. If WSREG.DLL cannot locate the resource DLL, the WSREG32.LOG file will be blank.

WSREG.DLL looks for the WSREG32R.DLL in the same directory from which the workstation registration agent is run. If WSREG32R.DLL is not there, WSREG.DLL looks for it in the \NLS\ directory. Finally, if it can't find it in the \NLS\, it will search the Environment Path settings.

After the workstation registration agent runs, at the root of the workstation's first local drive (most cases C:), a WSREG32.LOG file will be created. This file shows what the workstation registration agent did. If errors or problems occur, look at the WSREG32.LOG file to determine why a workstation didn't register or to determine why certain Workstation object attributes are not being updated correctly.

WSREG16.EXE is the second form of the workstation registration agent.

WSREG16.EXE can be executed on a DOS or Windows** 3.1 workstation. Like WSREG.DLL, the WSREG16.EXE will write out a log file named WSREG16.LOG to the first local drive found. It will have all the same information that is found in the WSREG32.LOG.

Under DOS, WSREG16.EXE uses the resource file WSREG16R.MSG. This file, like WSREG32R.DLL, has internationalized strings and is needed for WSREG16.LOG to be able to display messages. WSREG16.EXE uses the same search mechanism for finding the WSREG16R.MSG file as WSREG.DLL uses to find WSREG32R.DLL.

Under Windows 3.1, WSREG16.EXE creates the WSREG16.LOG file (just as it does when it is run under DOS), but the resource file is named WSREG16R.DLL. WSREG16R.DLL works the same as the other two resource files previously explained (WSREG32R.DLL and WSREG16R.MSG).

What Does the Workstation Registration Agent Do?

The workstation registration agent's priority is to get the workstation registered for importation (creating a Workstation object in an NDS tree) and then associate the physical workstation with the Workstation object in the tree after the workstation has been imported. It then will update the following Workstation object information: Network Addresses, User History, Last User, Last Server, and Last Registered Time. If the workstation registration agent can't find the workstation in the tree, it determines whether the workstation has been MOVED or RENAMED. We will discuss moving and renaming later in this document.

How Does the Workstation Registration Agent Know Which Workstation Object Belongs to the Workstation on Which It Is Running?

Under Windows NT and Windows 95 and 98, the Distinguished Name of the Workstation object is written to the Registry, where you can find the registration cookie also. This area is located at:

```
HKEY_LOCAL_MACHINE
\Software
\Novell
\Workstation Manager
\Identification
```

In this area you will find five values that the workstation registration agent uses:

1. The first value is the EXE Path. This will be used in future releases.
2. The second value is Registered In, which is used as a reference point for the workstation registration agent.
3. The third value is the Registration Object, which is the workstation registration cookie that is written to the container's Register Workstation attribute. The registration cookie looks something like this:
1:BrianD,01010480:00c04fd8eee2,137.65.61.85,BrianD.NS.Novell.com,BriansWorkstation,PRVFLY,WINNT,
PENTIUM

The first field is the name of the user who is logged in and is registering. The next field is the IPX address, followed by the IP address. The fourth field is the DNS Name, followed by the computer name, primary server, os type, and finally the type of CPU the workstation has. If the workstation registration agent cannot find a piece of information for one of these fields, the field is left blank. The information in the registration cookie is then used as NAMING VALUES for the actual importation and naming of the workstation.

4. The fourth value is the Tree. Under NT, the Trusted Tree can be set while installing the Z.E.N.works client, or you can right-click the Network Neighborhood and select Properties > Novell Workstation Manager > Properties.

You will see a place to enter a Trusted Tree. Under Windows 95, 98, and 3.1 platforms, the Tree value is set when the workstation registration agent runs for the first time. The workstation registration agent will use the primary tree as the Tree value. This value then becomes the Trusted Tree for the workstation to use.

5. The fifth and final value is Workstation Object. This value is the fully Distinguished Name of the Workstation object. When a Workstation object can be located by the Distinguished Name and the Tree name in the workstation Registry, the workstation is then associated. Under DOS and Windows 3.1, a hidden, system file called WORKSTAT.ID is created. This file maintains all the same information that the workstation registration agent writes to the Registry.

Creating and Associating a Workstation Import Policy to a User

Before you can import a workstation into the tree, you first need to create and associate a Workstation Import Policy to the user. To do this using Z.E.N.works 1.0 (Full Product or Starter Pack Version), follow these steps:

1. Launch NetWare Administrator (nwadmn32.exe) and select the Organization or Organizational Unit where you want to store Z.E.N.works Policy Packages.
2. Right-click the Organization or Organizational Unit, and then choose Create > Policy Packages > OK.
3. Choose to create the User Policy Package that corresponds to your users' operating systems (Windows 3.1, Windows 95, or Windows NT). For example, if you need to support one or more users who are using Windows 95 workstations, choose 95 User Package. If you support multiple platforms, create a User Policy Package for each platform.
4. Give the User Policy Package a name that is meaningful to you.
5. Choose the location where you want the policy package.
6. From the Policies list, check the Workstation Import Policy and choose Details.
7. Choose Workstation Location page, and then choose the container where you want the Workstation object to be created. If you want Workstation objects in the same location as the User objects, then accept the default User Container. If you want to change the location, choose Selected Container and browse for a location.
8. Choose Workstation Name page, choose how you want the workstations named, and then choose OK. The Workstations will be named according to the items in the list. If you accept the default, then your Workstations will be named as follows:

*Computer + Network Address for Windows 95 or NT workstations

*User + Network Address for Windows 3.1 workstations

9. Associate the User Policy Package you created in the previous steps with a User, Group, or container whose workstation you want to manage with this Policy Package:
 - a. In the User Policy Package, choose the Associations page.
 - b. Choose Add, and then browse for the object you want to associate with.
 - c. Choose OK twice to save changes and return to the tree view.

Importing the Registered Workstations

You can schedule imports to occur automatically using the Application Launcher or the Scheduler, or you can import workstations manually from the Tools menu or the command line. You can also import Workstations on a container-by-container basis (with or without sub-containers), or you can choose to import selected Workstations.

Note: After completing any of these options, you must run the Workstation Registration program again following the import to complete the registration process and synchronize the workstation with NDS.

Importing Selected Workstations Manually

You can selectively import workstations by selecting workstation entries on the Workstation Registration page of a container.

1. Ensure you have completed the steps outlined in the "Set Up Z.E.N.works" help topic.
2. Launch NetWare Administrator (nwadmn32.exe).
3. Highlight the container where the User objects reside whose workstations you want to import. The container must be associated (either directly or indirectly) with the User Policy Package (Windows 3.1, Windows 95-98, or Windows NT) you created previously.
4. Choose Details > Workstation Registration.
5. Select the registered workstations you want to import.
6. Choose Import Workstations. The import is complete at this point, and the Workstation objects are displayed in the tree (after you refresh the view of the container).
7. Choose Cancel to close the container's Details.
8. View the new Workstation objects in the tree.
9. Collapse the container, and then open it again to refresh the list.
10. View the workstations you imported in step 5.

Note: The Workstation objects are created in the context specified in the Workstation Import Policy found in the associated User Policy Package.

Scheduling Automatic Imports Using the Scheduler

If you use Desktop Management features, you can schedule the Import process by creating a scheduled action for wsimport.exe.

The scheduled action should include the .exe name, the context of the container you want to import to, and any command line parameters you might want to use.

For example, if your context were USERS.ORG and you didn't want to include subcontainers in the import session you were creating the scheduled action for, you might enter the following as your scheduled action target: Z:\public\wsimport.exe "users.org" /s-

Note: Enter wsimport.exe /? at the command line to view the available parameters.

Importing Groups of Workstations Manually Using Tools Menu

1. Ensure you have completed the steps outlined in the Set Up Z.E.N.works help topic.
2. Launch NetWare Administrator (nwadmn32.exe).
3. Highlight the container where the imported Workstations will reside (the container the registered workstation is associated with).
4. Choose Tools > Import Workstations.
5. Choose [...] to browse and choose the container from which you want to import registered workstations.
6. Choose OK to complete the Import process.

Note: The Workstation Registration program must run again following the import to complete the registration process and synchronize the workstation with NDS.

7. Close and reopen the browser to view the latest imported workstations in the applicable list (such as Valid Network Addresses).

Importing Groups of Workstations Manually Using Command Line

1. Ensure you have completed the steps outlined in the Set Up Z.E.N.works help topic.
2. Launch NetWare Administrator (nwadmn32.exe).
3. On the command line, enter the .exe, the context to the container you want to import to, and any command line parameters you want to use. For example, you might enter the following on the command line: `Z:\public\wsimport.exe "users.org" /s-`
if your context is USERS.ORG and you don't want to include subcontainers in the import session you are creating the scheduled action for.

Note: Enter `wsimport.exe /?` at the command line to view the available parameters.

Re-Registering Workstations

This last task serves two purposes: getting the physical workstation associated to the newly created Workstation object in the tree, and keeping the Workstation object attributes updated with some changes to the workstation.

1. Remember, for a workstation to be associated to a Workstation object in the tree, the distinguished name needs to be written to the Registry (WIN32 client) or the `workstat.id` (WIN16/DOS client) file in the Workstation Object value field. That fully distinguished name found in the Workstation Object field is what Z.E.N.works programs use to identify which Workstation object in the tree to update or modify.
2. To complete the last step and associate the physical workstation with a Workstation object in the tree, the `WSREG.DLL` (Win32) or `WSREG16.EXE` (Win3.1/DOS) must run a second time after an importation has been completed. How you register determines how you will finish this last step.

For example, . . .if you have installed the new Z.E.N.works client and use the Workstation Manager's scheduler to run the workstation registration agent (`WSREG.DLL`) under Windows NT, the Workstation Manager's scheduler will run `WSREG.DLL` every five minutes until the workstation registration agent can find the Workstation Object or fully distinguished name of the newly created Workstation object. Under Windows 95 and 98, the `WSREG.DLL` is run at the beginning of every cold login. (Cold login: logging in after starting up the workstation. Warm login: already logged in, and using the NetWare login utilities to log in again.) A warm login won't trigger the workstation registration agent to run in the scheduler. A cold login triggers the workstation registration agent to run. Under Windows 3.1, you must reboot and log in since the workstation registration agent is run in the login script.

Or, if you are not using the New Z.E.N.works client and are running the workstation registration agent in the login script, you will need to reboot and log in again. This applies to all platforms.

If, on the other hand, you are not using the new Z.E.N.works client and are using the Application Launcher, you will need to schedule a time for the workstation registration agent to run again.

3. The workstation registration agent, as mentioned above, actually serves two purposes: It gets the workstation created in the tree and associated, and it also updates certain Workstation object attributes in order for other applications to work.

For example, the physical workstation can have the IPX address changed by replacing the old network card with a new network card or even have the IP address change periodically through DHCP. Keeping the workstation's network addresses up to date is vital for some applications, such as Remote Control. Because of this, you should run the workstation registration agent periodically, at least at every login, with one exception:

the Windows NT workstation scheduler. In this case, after the workstation is associated to the Workstation object in the tree, the scheduler will not run the workstation registration agent again, and the scheduler will make all workstation updates that are needed, including the updating of the network addresses for the workstation.

The only time the Windows NT workstation scheduler will run the workstation registration agent again is if you move or rename a Workstation object or remove a Workstation object from the tree. In all other cases, the workstation registration agent is run at login or scheduled in the Application Launcher.

Other attributes that are updated by the workstation registration agent are as follows: User History, Last User, and Registered Time. The User History is a list of users who have modified the Workstation object by running the workstation registration agent. For example, the Last User is the user who modified the Workstation object last by running the workstation registration agent. The Last Registered Time is set every time the workstation registration agent runs and shows the last time the workstation registration agent ran on the workstation.

Un-Registering Workstations or Starting Over

You can un-register and start over in these three ways:

1. Run unreg32 (95, 98, or NT) or unreg16 (3.1) at the workstation.
2. Delete the Workstation object from the tree.
3. Remove the registration cookie from the workstation registration page.

Each method is described in more detail here:

1. If the workstation you are going to un-register is Windows 95, 98, or NT, run unreg32.exe at the workstation. If it is a Windows 3.1 workstation, then run unreg16.exe. When unreg32.exe is run on an NT workstation, the wsreg32.log is deleted and the Registry settings found at:

```
HKEY_LOCAL_MACHINE
\Software
\Novell
\Workstation Manager
\Identification
```

... are deleted, except for the Tree, which is the trusted tree that is set in workstation manager. If you want to change the tree name, set it through workstation manager by going into the Properties page for the Network Neighborhood. Select Novell Workstation Manager and click Properties. You can change it there.

Under Windows 95 and 98, unreg32.exe will delete the wsreg32.log and the workstat.id file along with all the settings, including the Tree value, in the Registry located at:

```
HKEY_LOCAL_MACHINE
\Software
\Novell
\Workstation Manager
\Identification
```

The workstat.id file under Windows 95 is a special case and is only used for Remote Control purposes. It is basically a copy of the Registry at:

```
HKEY_LOCAL_MACHINE
\Software
\Novell
\Workstation Manager
\Identification
```

... and is treated as such under 95. Under Windows 3.1, the wsreg16.log is removed along with the workstat.id file.

After the workstation has been un-registered, the workstation is ready to be registered and imported again.

2. You can also un-register a workstation by removing the Workstation object from the tree. If the Workstation object is removed from the tree, the workstation registration agent will create a new registration cookie and will re-register for importation. It is important to note here that the workstation registration agent will only re-register for importation if the workstation has a DS Authenticated connection to the Trusted Tree or the tree where the workstation should be. If the workstation ever logs in to a different tree and does not have a DS Authenticated connection to that Trusted Tree, the workstation registration agent will not un-register but will maintain its identity.
3. The third way to un-register a workstation is to remove the registration cookie from the workstation registration list found in the workstation registration page on a container. By removing the registration cookie, the workstation registration agent will create a new registration cookie and will register it to the user's location in the tree.

Note: If you are part of an IS group that works on the workstation boxes in your own area and then sends them back to the users, you must prepare your container for importation. Otherwise, the registration will never happen, and users will get workstations that have not been registered.

Moving and Renaming Workstation Objects in the Tree

As long as a workstation is moved or renamed with nwadmin32 that has the Z.E.N.works snap-ins, the snap-ins will capture the rename and move commands for a Workstation object and write the changes to the container's workstation tracking attribute. You can view this list by going to the container details and selecting the workstation tracking page.

Basically, as a workstation is moved and renamed, a trail is left behind to show where the workstation resides. This trail is found in the container's workstation tracking attribute. After the Workstation object has been renamed and moved, the next time the workstation registration agent runs on the workstation, the workstation registration agent will first determine if the workstation is still in the tree. If the Workstation object is not found in the tree, it will read the workstation tracking attribute and follow the trail to the new workstation name or place. As it follows the trail, the workstation registration agent also removes the trail from the attribute, thereby cleaning up the workstation tracking.

Once it finds the new name or place, the workstation registration agent will write the new distinguished name down to the workstation Registry, and the workstation is then associated with the Workstation object in the tree. If the workstation registration agent doesn't find the workstation, it will re-register for importation.

ANHANG 4: Upgrading from NAL 2.x to Z.E.N.works 1.0

by matt brooks

If you've been administering a network with the NAL 2.0 or NAL 2.01 snap-in, and are planning to upgrade to the new NAL 2.5 included in Z.E.N.works, one of your concerns might be "What about all those snAppShots...do I have to do them again?" or "Are new 2.5 snAppShots backwards compatible?"

NAL 2.5 introduces several new macros and new .AOT entry types. If you create an .AOT with NAL 2.5 and use that .AOT in NAL 2.01, the older version will not recognize the new macros and, therefore, they won't do what macros are supposed to do: substitute data. The same thing happens to newer .AOT entries: They are ignored because 2.01 does not know what to do with them. In short, NAL 2.5 is backward compatible with 2.01 data; however, NAL 2.01 is NOT forward compatible with 2.5 data.

NOTE: Before you install Z.E.N.works, be sure to read the NDS Design document we have. It has some good pointers that could save you a lot of headaches.

Now that we've cleared the air, let's get that system of yours upgraded! To upgrade from NAL 2.0 to Z.E.N.works 1.0, follow these steps:

1. Because the Z.E.N.works installation process overwrites NAL 2.01 snap-ins with NAL 2.5 snap-ins, make a backup copy of the NetWare Administrator (NWAdmin) software you are currently using.

For example, if you use NWAdmin95, copy the SYS:Public\Win95 directory to a safe place other than SYS:Public. If you use NWAdminNT, copy the SYS:Public\WinNT directory to a safe place other than SYS:Public.

2. Install Z.E.N.works to SYS:Public, which includes NWAdmin32 as one of the components in the installation.

NOTE: If you intend to use NAL 2.01 with the new Windows NT Client that comes with Z.E.N.works, disable the installation of the NAL NT Service, which might cause conflicts when NAL 2.01 runs. By default, the new Windows NT Client installs the NAL NT Service, which consists of three files in the System32 directory: NALNTSRV.EXE, NALNTRES.DLL and NWAPP32.DLL. You can disable the NT Service installation while installing the new Windows NT Client.

3. Use the old NWAdmin you saved in Step 1 to manage your network as you transition from NAL 2.01 to NAL 2.5.

NOTE: The "transition" includes the period of time it takes you to update NAL.EXE and NALEXPLD.EXE, the executables that enable users to access Application Launcher and Application Explorer at their workstations. It's possible that some users will be running the new NAL and others will be running the old NAL. Hence, it's best to administer the network with the older NWAdmin because NAL 2.01 does not recognize the new features in NAL 2.5.

4. When you are sure all users are running NAL 2.5 software at their workstations, delete the copy of NWAdmin you created in Step 1.
5. Start the new NWAdmin32 (found in SYS:Public\Win32).
6. OPTIONAL: Open the Launcher Configuration property page of each User, Organization, and Organizational Unit object where you previously set launcher configurations and uncheck the Backwards Compatible option so that NAL 2.01 data is not saved anymore.
7. Open the Launcher Configuration property page on the parent container that you've previously set as the "top" of the configuration inheritance tree. On that page, check the Use As Top of Configuration Tree option.

NOTE: When Application Launcher searches the NDS tree for configuration settings, it starts at the lowest possible leaf object (a User or container object) and works up the inheritance tree. It continues walking up this tree searching for custom settings until it reaches a container object that has been designated as the "top" of the inheritance tree. If it finds custom configurations in any of the objects along the way, those are the configurations that are applied. If it doesn't find any custom configurations, then the configuration is considered "unset" and the default configuration is applied. This method of inheritance lets you control when and where custom configurations are applied instead of the defaults.

8. In addition to the changes to the Launcher Configuration property page, notice the new or changed property pages in the Application object.

For example, check out the new Order Icon feature on the Identification property page, which lets you order the application icons on the desktop as well as order how applications are forced to run.

The Registry Settings, Application Files, Text Files, Icons/Shortcuts, and INI Settings property pages have some great new stuff, including a Find option and a bevy of distribution options that let you control how specific settings are distributed.

Speaking of distribution, the global distribution options (those that affect the entire Application object) have been moved to a new property page called "Distribution." Look at it and you'll see some familiar faces and some new bells and whistles.

Lest we forget, you can also create Application folder objects to help you set up multi-level foldering at the workstation, and also an Application Launcher Tools menu (off of the main Tools menu). Of course, to really get to know all of these new features, the best thing to do is click that Help button on the dialog box. We'll also be posting Feature Articles and Tips in the future that illuminate even more.

ANHANG 5: Is it a Clean, Baseline Machine?

Ed S., an IS Manager working for the Salvation Army, asks:

Your Intro to NAL says that I should start with a baseline workstation, snAppShot it, modify it as needed, and take another snAppShot. Well, looks good to me, but I have 70 IBM PCs, all running Windows 95, and 14 machine types using different drivers and the like. Does this mean that I will be stuck creating 14 different sets of snAppShots?

It depends on the application, Ed. If you're installing a run-of-the-mill application that doesn't depend on any particular hardware configuration, you can likely create a "one size fits all" Application object. If, however, you're installing an application that makes decisions depending on the hardware configuration, you might need to create a unique snAppShot for each configuration.

It became clear to us after reading Ed's question (and others like it) that we need to clarify some terms, introduce some new features, and clear up a couple misunderstandings about what snAppShot can and can't do. This week's Feature Article is intended to make your experience with snAppShot hopefully a little better.

For best results, we recommend that you run snAppShot on a machine that is both clean and baseline. Here's the difference:

What is a clean and baseline machine?

- A baseline machine is a "rank and file" machine...one that represents all of the machines on the network to which you intend to distribute applications.

For example, a Pentium 333 with 128 RAM and 3 Gigs of hard disk space is hardly representative of a network that has over three thousand 486 and P-150s with 16 RAM and 300 Megs of hard disk space per copy.

Furthermore, if a particular application installs a particular piece of software because it detects a particular hardware device, then that machine ceases to be a baseline system. It is now considered atypical. The resultant .AOT file will represent the peculiarities of that machine, which might or might not transfer to other machines to which you distribute.

- A clean machine is one that has very little or no extra software loaded. If a machine is dirty, that is, if it has applications already loaded, there is the possibility that snAppShot's discovery of the workstation will wind up inaccurate. When you're dealing with Application objects, you want the "snapshot" to be crystal clear for best results.

For example, suppose a workstation has had NETSCAPE.EXE already installed, and you install Netscape Navigator using snAppShot. Because the file already exists, snAppShot does not detect the change and the resultant .AOT file (which you use to build the Application object) does not truly represent a Navigator installation.

Our recommendation is that before you use snAppShot, ensure that the machine has nothing but an operating system and a NetWare Client loaded. Look for the great tip at the end of this article explaining how to achieve a clean machine with a small investment in imaging software. Before we get to that, let's clear up the mother of all misunderstandings: what is snAppShot really doing?

The snAppShot Space: Nuthin' But Apps

Several readers have asked us if snAppShot can capture the logic of an installation involving choices based on existing hardware, software, or other settings. For example, if the application's Setup program installs a particular video driver or modem setting file to a workstation, will these settings be valid when transferred to another workstation?

The answer is "probably not." The reason is because snAppShot confines itself to working only in the very top application software layer of a machine. It ignores the layers underneath such as the Windows disk driver, device drivers, and the physical hard disk drive (note the hardware-specific Registry settings that it ignores). snAppShot simply captures the application software changes as a result of installation. We said it before and we'll say it again: If a particular piece of application software is installed because the installation detects a particular hardware device, snAppShot captures the "delta" (or difference). However, that .AOT might not working properly on a machine that does not have that particular hardware device.

Fortunately, most applications don't care too much what they run on, as long as there's plenty of hard disk space and RAM.

What Does snAppShot Scan?

In keeping with the "application software only" philosophy, snAppShot scans only the application-related files that you tell it to scan. For example, only the following application-related hardware-independent files are scanned:

- 1) File system files (such as C: or D: drives)
- 2) .INI files)

- 3) Text files (such as CONFIG.SYS and AUTOEXEC.BAT)
- 4) Registry

What Does snAppShot Ignore?

snAppShot ignores any drive, directory, or file you tell it to ignore. In addition, snAppShot version 2.5 ignores the following Windows 32-bit areas:

- 1) All Registry keys except for the HKEY_LOCAL_MACHINE and HKEY_CURRENT_USER hives (both selected by default). The other registry hives are either redundant backup hives or are hardware-specific.
- 2) The following HKEY_LOCAL_MACHINE\SYSTEM keys (most of which are Windows NT), are either redundant or hardware-specific (and dynamically generated at startup):
 - 001, 002, 003, 004,...
 - Clone
 - CurrentControlSet\Enum
 - Select
 - Startup
 - CurrentControlSet\Services\\Enum

Using snAppShot with Imaging Software

Our snAppShot engineer is a quality assurance nut. He's snapped and debugged about every application under the sun a hundred times (including some gargantuan and complex suite applications). Quite simply, he's pushed snAppShot to the limits and stands by his product.

So, when it came time to write this article, this guy was the perfect candidate to give us a few suggestions about how best to use snAppShot. What follows is his (free) advice:

1. Pick a baseline workstation on your network that, in your best judgement, represents the other machines on your network. You're looking for an average set of hardware.
2. Create a DOS boot disk complete with an operating system and a NetWare client.
3. Use software such as Drive Image by Powerquest or Ghost by Ghost Software to save an "image" of the machine. Store the image on the network.
4. Reformat the machine and install only the operating system and NetWare client (using the boot disk you created in Step 2 is a good method).
5. Customize the machine exactly the way you want it while keeping things clean.
6. Image the machine again and save the "clean machine" image to the network.
7. Install an application using snAppShot and create the .AOT file.
Note: Now the machine is considered "dirty" because you have installed an application to it.
8. Restore the "clean image" of the machine you saved in Step 6.
9. Repeat Steps 7 and 8 as many times as you want to create .AOT files with snAppShot.
10. Restore the image you saved in Step 3.
11. Test the Application object against the original baseline configuration.
12. Test the Application object against the "clean image" created in Step 6.
13. Leave the office early for a quick nine holes, confident that you will be installing multiple applications to hundreds of machines without having to personally visit them.
14. Repeat Step 13 tomorrow.

ANHANG 6: Update the Secure Portion of the Registry on Winwos NT

... our team of engineers has added a service to NAL 2.5 (that ships with Z.E.N.works 1.0) that lets you register Application Launcher with the Windows NT Service Control Manager. This lets Application Launcher make the necessary changes to a secure Windows NT Server or Workstation. For security reasons, the service cannot be used by any application other than the Application Launcher.

Note: You need to install the Windows NT Service Pack 3 before installing the NT service.

Mehr Infos ...

Weitere Infos unter den Novell Internetadressen (auch <http://support.novell.com> und <http://www.novell.com/cool-solutions/zenworks>), von mir auf Diskette oder unter <ftp://gast@ftp.architektur.rwth-aachen.de>. Unter NOVELL/ZENWORKS/DOCS finden sich u.a. die folgenden Dateien:

Cleanmach: is it a clean baseline machine? (cool solutions)
Ndps: novell deliverd print services white paper
Policies: specifying global settings (zen-dokumentation)
Zen-Conc: essential concepts (aus der zen-dokumentation)
ZenEfPol: why i love effective policies (cool solutions)
WstMg-WP: White Paper für den Workstation Manager 1.0 (NT Client 4.11)
Zen-Inst: what happens during installation (zen-dokumentation)
ZenNds1: nds design for zen works
ZenNetsc: Distributing Netscape Navigator using Novell's ZENworks
Zen-Off: using zen works to distribute microsofts office 97 (cool solutions)
Zen-Remc: secure remote control (cool solutions)
Zen-Strt: getting startet (aus der zen-dokumentation)
ZenRgWrk: how to register a workstation (cool solutions)
Zen-Strt: How to get startet
Zen-Usg: zen userguide (zen-dokumentation)
Zen-Wp: zenworks white paper

Unter NOVELL/ZENWORKS befinden sich

ZENworks.pdf dieser Skript
ZENdesgn.pdf Anmerkungen zum Design des NDS-Baumes
ZENguide.pdf eine ausführliche Referenz zu ZENworks 1.1

Andere Infos bitte der Datei INFO in diesem bzw. FILEINFO im Verzeichnis DOCS entnehmen.

Unter WINNT / WIN95 /WIN98 jeweils im Unterverzeichnis NWCLIENT befinden sich in der Regel die neuesten Clients, Downloads und Patches.

Weitere interessante Adressen im Internet:

Eudora-Spiegel (für [ftp.novell.com](ftp://ftp.novell.com)- patches und updates): HOST=[ftp.qualcomm.com](ftp://ftp.qualcomm.com), DIR=quest/windows/eudora
Die *Novell Minimum Patchlist* findet sich auf der Seite <http://support.novell.com/misc/patlst.htm>