

1. MSI für Einsteiger

Autor: Thomas Geiger

Stand: 28.10.2008

Inhaltsverzeichnis

| | |
|--|---|
| 1.MSI für Einsteiger..... | 1 |
| 1.1.Einleitung..... | 1 |
| 1.1.1. Voraussetzungen und Hinweise..... | 2 |
| 1.1.2. Literaturempfehlungen..... | 2 |
| 1.2.Was ist ein msi-Paket? | 2 |
| 1.3.Woher bekommt man ein msi-Paket?..... | 3 |
| 1.4.Steuerung von msi-Paketen per mst-Datei..... | 3 |
| 1.5.Testen von fertigen msi-Paketen (evtl. mit mst-Datei)..... | 4 |
| 1.6.Bereitstellung im NAL (allgemeine Beschreibung)..... | 5 |
| 1.7.Erstellung eigener msi-Pakete (AdminStudio)..... | 5 |
| 2.Übungen zu msi-Paketen..... | 6 |
| 2.1.Ein fertiges Paket „von Hand“ installieren..... | 6 |
| 2.2.OpenOffice als msi-Paket verteilen..... | 6 |
| 2.3.Acrobat Reader 8.1 tunen und verteilen..... | 9 |
| 2.4.Den USB-Drive-Letter-Manager verteilen..... | 9 |

1.1. Einleitung

Die Bereitstellung von Software im Schulnetz gehört zu den zeitintensivsten Beschäftigungen des Netzwerkberaters. Dies liegt einerseits an der Vielzahl von Anwendungen, mit denen an einer Schule gearbeitet wird, andererseits an oft „netzunfreundlich“ programmierter Software von Seiten der Programmierer. Sehr informative Ausführungen zum Thema „Schulnetzgeeignete Software“ finden Sie in: http://support-netz.de/uploads/tx_dcfiles/empfehlungen-zur-software-in-schulischen-netzen.pdf !

Die bisherigen Methoden der Paketerstellung per Snapshot in der Novell-Musterlösung sind in vielen Dokumenten ausführlich dokumentiert und in den Arbeitskreisen sicher schon oft behandelt worden. Etwas stiefmütterlich wurde bisher aber auf die in der Windows-Welt weit verbreitete Verteilung per msi-Paket eingegangen. Da in zukünftigen ZenWorks-Versionen das Snapshot-Programm nicht mehr gepflegt werden wird, sollten bzw. müssen auch wir uns mit der Verteilung von msi-Paketen beschäftigen. Dieses „Einstiegs-Dokument“ beginnt „ganz vorne“ und soll folgende Fragen klären:

- Was ist ein msi-Paket?
- Woher bekommt man die Pakete?
- Wie kann man fertige Pakete beeinflussen bzw. steuern?
- Wie stellt man die Pakete per NAL im Netz zur Verfügung?
- Wie kann mal selbst msi-Pakete erstellen, wenn der Hersteller keine zur Verfügung stellt?

1.1.1. Voraussetzungen und Hinweise

Damit auf einem Windows-PC ein msi-Paket verteilt werden kann, muss der *Windows-Installer* installiert sein. Diese Voraussetzung ist bei aktuellen Windows-Versionen in der Regel erfüllt. Außerdem muss der angemeldete Benutzer dazu berechtigt sein, auf der WXP-Station Software zu installieren. Im Gegensatz zur ML2, wo alle Benutzer lokale Administratorrechte haben, ist dies bei der paedML Novell 3.0 nicht mehr der Fall! Außer Lehrern und Verwalter sind alle anderen Benutzer nur noch „normale Benutzer“, die in der Regel keine Software installieren dürfen. Damit die personenbezogene Verteilung per Zenworks dennoch funktioniert, ist der *Windows-Installer* per Gruppenrichtlinie so konfiguriert, dass eine Installation unabhängig vom angemeldeten Benutzer mit erhöhten Rechten durchgeführt wird. Außerdem führt ZenWorks eine Installation immer mit Systemrechten durch. Dennoch kann es manchmal vorkommen, dass sich ein Standard-msi-Paket nicht verteilen lässt und auf Administratorrechten „besteht“. Dann kommt man an der Nachbearbeitung des Paketes mit dem *AdminStudio* oder einem anderen Programm nicht vorbei.

Dieses Dokument ist für die paedML Novell 3.0 geschrieben, eine für die ML2.x geeignete Version findet man unter <http://www.lehrerfortbildung-bw.de> .

1.1.2. Literaturempfehlungen

Einsteigerliteratur:

1. Das Kapitel 7 „Programm-Installation“ im Basiskurs 3.0:
http://lehrerfortbildung-bw.de/netz/muster/novell/material/basis/07_programm_installation.pdf
2. Das Kapitel 5 „Anwendungen“ im Basiskurs 2.0
<http://lehrerfortbildung-bw.de/netz/muster/novell/material/basis/basism12/index.html>
3. Die Präsentationen: ML2_ZENApp.ppt (insbesondere die Folien 32- 46)

Für Fortgeschrittene, insbesondere zur Erstellung eigener Pakete:

1. <http://support-netz.de/nml-msi-adminstudio.html>

1.2. Was ist ein msi-Paket?

Quelle: o.g. Link „Für Fortgeschrittene“ (Autor: Stefan Falk)

Managed Software Installation, kurz MSI, steht für eine Installationsmethode, die von Microsoft (MS) mit der Veröffentlichung von Office 2000 eingeführt wurde. Es handelt sich dabei um eine relationale Datenbank, in der sämtliche Anweisungen zur Installation eines Programms auf einem Windows Rechner stehen. Microsoft hat damit einen Standard für Installationsroutinen beschrieben, an den sich Softwarehersteller halten müssen.

Dieses Verfahren hat sich zu einem Weltstandard etabliert und wird von Microsoft kontinuierlich weiter entwickelt. Jedes Windows Betriebssystem ab Windows 2000 bringt dazu den Installer-Dienst mit, der bei einer Installation gestartet wird. Die ausführbare Datei dieses Dienstes ist *msiexec.exe*.

Die Mehrzahl der Softwarehersteller setzt inzwischen auf dieses Verfahren. Es bietet gegenüber der bisherigen Setup Installation eine gewichtige Reihe von Vorteilen. So wird der gesamte Installationsprozess mitprotokolliert und der Endzustand gespeichert. Installationsfehler werden in das Ereignisprotokoll geschrieben. Bei einer misslungenen Installation wird dadurch automatisch ein Rollback ausgeführt (der blaue Balken läuft rückwärts). Dadurch bleiben keine "Dateileichen" und Registry Einträge auf der Arbeitsstation zurück. Auf dem Client wird eine konsistente Installation überwacht, die sogenannte Selbstheilung von Installationen. Fehlt eine Programmdatei, so wird sie, ohne Eingreifen des Anwenders, automatisch von der Installationsquelle nachinstalliert. Die De-Installation von Programmen wird vollständig ausgeführt. Weiterhin werden zum Beispiel ältere .dll Dateien durch neuere ersetzt aber niemals umgekehrt, wie dies bei bisherigen Setups häufig der Fall war.

Administratoren sprechen dann von der sogenannten "dll-Hölle".

Da jede MSI-Datenbank vom Programmierer bestimmte Vorgaben enthält und man die Datenbanken mit anderen Werten füllen kann, ist es möglich, Installationen zu beeinflussen. So können Sie beispielsweise die Sichtbarkeit der Symbole auf dem Desktop oder im Startmenü nach Ihren Bedürfnissen anpassen oder verhindern, dass nicht benötigte Teile der Programm-Komponenten installiert werden. Diese Anpassungen in der MSI-Datenbank (Transformation) werden dann in einer sogenannten mst-Datei (Managed Software Transform) gespeichert. Sie beinhaltet die Differenz zwischen dem MSI-Standard und dem gewünschten Ergebnis.

1.3. Woher bekommt man ein msi-Paket?

Hat man ein fertiges (nach Möglichkeit schon für die Musterlösung angepasstes) msi-Paket, so besteht der einzige Aufwand für den Netzwerkberater, dieses Paket per NAL im Netz zur Verfügung zu stellen. Dazu wird in der ConsoleOne ein „Anwendungspaket basierend auf einer msi-Datei“ erstellt und mit wenigen Klicks ist die Arbeit erledigt!

Doch woher kommen die msi-Pakete?

1. Man erhält ein msi-Paket von einer vertrauenswürdigen Quelle; dies könnten sein:
 - das Software-Portal des Support-Netzes (das hoffentlich bald online gehen wird)
 - ein Bekannter, der ein fertiges Paket zur Verfügung stellt
 - ein Website, auf der solche Pakete „angeboten“ werden (z.B. die „Tipps und Tricks“ der Windows-Musterlösung www.ml-tipps.de)
2. Man extrahiert das msi-Paket aus einer „Setup“-Datei
Beispiele: OpenOffice, Acrobat Reader
3. Man erstellt selbst mit geeigneter Software (z.B. AdminStudio) ein Paket aus der Installationsroutine einer Programminstallation (Diese Aktion nennt man „Repaketierung“).

Bitte beachten Sie in allen Fällen unbedingt die Lizenzbestimmungen des jeweiligen Herstellers!

1.4. Steuerung von msi-Paketen per mst-Datei

Bei der Verteilung eines msi-Paketes wird das Programm in der Regel mit vorgegebenen Standardeinstellungen installiert. Eine automatisierte Einflussnahme auf die Einstellungen ist normalerweise nicht vorgesehen. Will man jedoch z.B. in ein anderes Verzeichnis installieren, oder Einträge im Startmenü beeinflussen, so kann man dies mit einer selbst erzeugten Steuerdatei (auch Transformdatei genannt) erledigen. Zur Erstellung dieser mst-Datei ist ein Programm nötig, das die msi-Datei öffnet, die Einstellungen auslesen und die Veränderungen in der mst-Datei speichern kann.

Beispiele:

- Der aus der ML2.x bekannte *InstallTailor* (steht in der ML3 nicht mehr zur Verfügung) (Der *InstallTailor* bittet allerdings nur relativ wenig Konfigurationsmöglichkeiten und wurde vermutlich nicht allzuoft verwendet?)
- Vom Programmhersteller gelieferte Hilfsprogramme (z.B. der *InstallShield Tuner*, auch *Customization Wizard* genannt, für die verschiedenen Versionen des *Acrobat Readers*). Mit solchen Hilfsprogrammen sind oft viele erweiterte Einstellmöglichkeiten gegeben, die man sonst im Programm selbst nach der Installation über z.B. *Extras / Optionen* vornimmt. Damit kann das Paket so beeinflusst werden, dass es direkt nach der Verteilung schon mit den gewünschten Voreinstellungen auf der Maschine lauffähig ist. Nachteil dieser herstellereigenen Tools ist, dass sie nur für Produkte des jeweiligen Herstellers verwendbar sind. Pakete von anderen Herstellern können natürlich nicht konfiguriert werden.

- Das z.B. in der Windows-Musterlösung verwendete *WinInstall*; hier liegen allerdings keine Erfahrungswerte vor.
- Die vorgenannten Nachteile sind beim *Tuner* des *AdminStudios* nicht vorhanden. Mit ihm können alle msi-Pakete bearbeitet und passende mst-Dateien erstellt werden.

1.5. Testen von fertigen msi-Paketen (evtl. mit mst-Datei)

Bevor man von einem msi-Paket mit der *ConsoleOne* ein Anwendungsobjekt erstellt und dieses im NAL zur Verfügung stellt, sollte man das Paket ausführlich testen. Dabei kann man beispielsweise feststellen, ob sich ein Standardpaket problemlos unter den verschiedenen Benutzerkennungen verteilen lässt. Außerdem sieht man, ob es nach der Verteilung so installiert ist, wie man es gerne hätte (Einträge im Startmenü, Installationsverzeichnisse usw...). Gehen Sie dabei wie folgt vor:

- Erstellen Sie einen Snapshot von ML3-PC01! („Original“)
- Melden Sie sich als *PgmAdmin-LFB* an!
- Kopieren Sie das Verzeichnis (z.B. *programmxy_msi*), das die msi-Datei enthält auf die lokale Festplatte *C:* !
- Starten Sie die Installation über *Start / Ausführen* durch Eingabe von:
msiexec /i "C:\programmxy_msi\programmxy.msi" /qb
Durch diesen Befehl wird das Programm ohne Benutzereingaben mit Fortschrittsanzeige installiert.
- Testen Sie danach das Programm!
Wenn alles funktioniert, ist sichergestellt, dass das Paket in Ordnung ist und mit lokalen Administratorrechten installiert werden kann.
- Fahren Sie den PC herunter und setzen ihn auf „Original“ zurück!
- Melden Sie sich dann als Schüler an und testen Sie nochmals die Installation!
Falls die Installation durchläuft, kann das Programm auch mit eingeschränkten Rechten auf der WXP-Station installiert werden; Sie können dann mit dem nächsten Kapitel weitermachen!
- Bricht die Installation dagegen mit einer msi-Fehlermeldung ab, so haben Sie ein Problem: Das Paket lässt sich ohne vorherige Bearbeitung mit eingeschränkten Rechten offensichtlich nicht installieren (Im Test trat dies bei OpenOffice 2.4 auf). Jetzt bleibt nur noch die Möglichkeit, eine mst-Datei zu erzeugen, in der (hoffentlich) eingestellt werden kann, dass das Programm auch als Standardbenutzer verteilt werden kann. Die mst-Datei muss mit einem der o.g. Programme erstellt werden. Hier sind fortgeschrittene Kenntnisse zu msi-Dateien notwendig, auf die hier (noch) nicht eingegangen werden kann. Näheres dazu in dem Dokument „MSI für Fortgeschrittene“, Erscheinungsdatum z. Zt. noch nicht bekannt.
- Wenn eine passende mst-Datei (*programmxy.mst*) zur Verfügung steht, kann die Verteilung (als Lehrer / Schüler) wie folgt getestet werden:
Start / Ausführen und Eingabe von:
msiexec /i "C:\programmxy_msi\programmxy.msi" Transforms="C:\programmxy_msi\programmxy.mst" /qb
Für die Novell-Edition von OpenOffice 2.4 (die mit einer fertigen mst-Datei ausgeliefert wird, würde der Befehl wie folgt lauten:
msiexec /i "C:\OpenOffice24_NE\openofficeorg24.msi" Transforms="C:\OpenOffice24_NE\pack\openofficeorg24.mst" /qb

Falls diese Tests erfolgreich absolviert sind, kann mit der *ConsoleOne* das Anwendungsobjekt erstellt werden!

1.6. Bereitstellung im NAL (allgemeine Beschreibung)

Wenn wir von dem Szenario ausgehen, dass ein fertiges msi-Paket und eine dazugehörige mst-Datei vorhanden und getestet sind, ist die Bereitstellung der Anwendung im Netz schnell erledigt. Der Umgang mit der *ConsoleOne* und die Erstellung einer Anwendung werden hier als bekannt vorausgesetzt, deshalb werden lediglich die notwendigen Schritte ohne Screenshots erläutert. (Ausführlich wird das Vorgehen im oben erwähnten Kapitel des Basiskurses 3.0 beschrieben.)

Ausgangspunkt:

Es existiert irgendwo ein Verzeichnis mit msi-, mst- und evtl. weiteren Dateien, das Verzeichnis heißt beispielsweise `programmxy_msi`. Gehen Sie wie folgt vor:

1. Erstellen Sie auf `K:\` ein Verzeichnis `programmxy` und darunter das Verzeichnis `_pack`!
2. Kopieren Sie die Dateien aus `programmxy_msi` nach `K:\programmxy_pack`.
3. Starten Sie als `PgmAdmin-LFB` die *ConsoleOne*, navigieren Sie zu *LFB | Anwendungen* und erstellen Sie in der gewünschten OU eine neue Anwendung *programmxy*, basierend auf einer msi-Datei. Wenn noch keine OU zur Aufnahme der neuen Anwendung besteht, sollten Sie diese zunächst anlegen (z.B. *Office, Grafik, Technik, Multimedia* o.ä.)
4. Navigieren Sie zur msi-Datei in `K:\programmxy_pack`
5. Beim Betriebssystem wählen Sie *NT/2000/XP*, Version ≥ 5.0
6. Unter Verknüpfungen tragen Sie die gewünschte OU ein (z.B. alle *Benutzer* oder aus der OU *Benutzer* nur die *Lehrer* und *Schueler*).
7. Aktivieren Sie *Details anzeigen* und klicken Sie auf *Fertig stellen!*
8. Unter *Identifikation | Ordner* wählen Sie *Hinzufügen | Verbundener Ordner*, klicken dann eine Stufe hoch zu *Startmenue* und sortieren das Programm an der gewünschten Stelle ein.
9. Unter *Allgemein | Dateirechte* geben Sie dem Objekt die Rechte R und F (Lesen und Dateiabfrage) im Verzeichnis `K:\programmxy`! Die Rechte werden immer für das gesamte Programmverzeichnis vergeben, da bei einer serverbasierten Installation die Dateien direkt in `K:\programmxy` abgelegt werden.
10. Unter *MSI | Eigenschaften* wählen Sie *Hinzufügen* und tragen *Allusers=2* ein.
11. Damit ist das Objekt vorläufig fertiggestellt!
Lassen Sie die *ConsoleOne* noch offen, um evtl. noch Korrekturen vornehmen zu können!
12. Aktualisieren Sie die Anzeige im NAL; das neue Programm sollte im richtigen Ordner erscheinen. (Momentan noch mit dem NAL-Standard-Symbol)
13. Sie können das Programm dann installieren und im Windows-Startmenü starten.
(Der Start über das Windows-Startmenü ist nicht die bevorzugte Variante in der paedML Novell; hier empfiehlt es sich, ein weiteres Objekt zum Starten des Programms zu erstellen. Noch komfortabler wird es, wenn über abhängige Anwendungen Installation und Start miteinander verknüpft werden; s. dazu die nachfolgenden Übungen!)
14. Damit im NAL auch noch ein programmspezifisches Symbol erscheint, kopieren Sie die zugehörige exe-Datei nach `K:\programmxy_pack`! In der *ConsoleOne* wählen Sie *Identifikation | Symbol* und navigieren zu der exe-Datei. Mit der Zuweisung des Symbols sind Sie fertig!

1.7. Erstellung eigener msi-Pakete (AdminStudio)

Mit dem *AdminStudio* können aus Installationsdateien oder Installations-CD's msi-Dateien erstellt werden. Außerdem ist es möglich, mit Hilfe des *Tuners* zu einer vorhandenen msi-Datei eine angepasste mst-Datei zu erstellen, um Einfluss auf die Installation zu nehmen. Eine ausführliche Anleitung finden

Sie in den zu Beginn genannten Links. Da das *Adminstudio* momentan an den Standorten noch nicht zur Verfügung steht, muss auf die nächsten Termine verwiesen werden.

2. Übungen zu msi-Paketen

Vorbemerkungen:

- Die in den Beispielen genannten Dateien und Ordner erhalten Sie vom Referenten!
- Vor jeder neuen Übung sollten Sie die jeweilige VM in den Originalzustand zurücksetzen!
- Dazu erstellen Sie VOR dem allerersten Start der VM einen Snapshot „Original“!

2.1. Ein fertiges Paket „von Hand“ installieren

Um einen ersten Einblick zu erhalten, installieren wir ein Paket von Hand. Dazu benötigen wir ein Verzeichnis, in dem die msi-Datei zu finden ist. Mit einem Rechtsklick auf die msi-Datei sehen wir im Kontextmenü des *Windows-Explorers* den Punkt *Installieren*, mit dem die Installationsroutine gestartet wird.

Ein zweite Variante einer schnellen Test-Installation besteht im direkten Aufruf des *Windows-Installers* über den Befehl *msiexec /i* mit dem Pfad zur msi-Datei (s.o.)

Aufgabe: PaintshopPro lokal installieren (nur zum Kennenlernen)

- Erstellen Sie einen Snapshot von ML3-PC01! („Original“)
- Starten Sie den PC und melden Sie sich lokal als *Adam* an!
Bei dieser Übung benötigen Sie auf der WXP-Station Administratorrechte; als eingeschränkter Benutzer ist eine Installation nicht möglich! Dies könnten Sie testen, indem Sie auf der Station mit der Windows-Benutzerverwaltung einen eingeschränkten Benutzer einrichten (z.B. Benutzername *user* mit dem Passwort *lokal*).
- Kopieren Sie den Ordner *PaintshopPro_msi* vom Host auf die Platte *C:* der VM!
Sie sehen dann folgende Struktur: *C:\PaintshopPro_msi* !
- Installieren Sie das Programm *PaintshopPro* mit einem Rechtsklick auf die msi-Datei!
- Starten Sie das Programm und testen Sie die Funktion!
- Setzen Sie die VM wieder zurück auf „Original“!
(Bzw. de-installieren Sie das Programm per Rechtsklick auf die msi-Datei mit: *Deinstallieren*)
- Installieren Sie das Programm jetzt über *Start / Ausführen* und Eingabe folgender Befehlszeile:
msiexec /i "C:\PaintshopPro_msi\PaintshopPro.msi" /qb !
- Testen Sie wiederum die Funktion und setzen Sie die VM danach auf „Original“ zurück!

2.2. OpenOffice als msi-Paket verteilen

Eine Anleitung zur (serverbasierten) Installation einer älteren Version finden Sie unter:

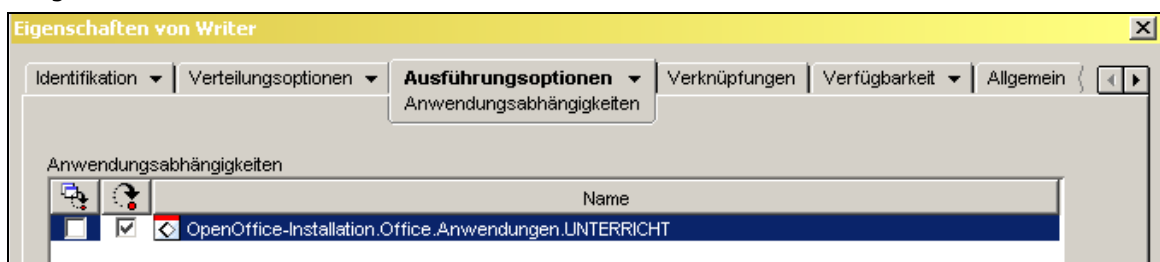
<http://support-netz.de/son-openoffice-2-01.html> Sie können diese Übung mit der Standard-Version von OpenOffice oder auch mit der Novell-Edition durchführen. Die Novell-Edition hat den Vorteil, dass beim Download eine fertige mst-Datei dabei ist. Außerdem kann diese Version mit den Dateiformaten von MS Office 2007 (docx, xlsx, pptx) umgehen, die Standard-Version dagegen nicht!

Die aktuelle Version von *OpenOffice* soll im Netz wie folgt zur Verfügung gestellt werden:

- Im NAL soll unter *Anwendungen / Office* ein Installationspaket erscheinen, das *OpenOffice* lokal auf der Maschine installiert (die lokale Installation wird hier aus Performance-Gründen bevorzugt)
- Es sollen (drei) Objekte zum Start von *Writer*, *Calc* und *Impress* erstellt werden! (Falls die weiteren OpenOffice-Programme benötigt werden, können dafür natürlich ebenfalls Verknüpfungen angelegt werden.)
- VOR dem Start der jeweiligen Anwendung (*Writer*, *Calc* ...) soll noch eine modifizierte Datei *bootstrap.ini* in das Programmverzeichnis von OpenOffice auf C: kopiert werden. In dieser Datei wird festgelegt, dass die benutzerbezogenen Einstellungen im Homeverzeichnis des jeweiligen Benutzers abgelegt werden (z.B. H:\Profil, (dieses Verzeichnis sollte in H:\ existieren), s. dazu die o.g. Installationsanleitung, in der die Syntax der *bootstrap.ini* beschrieben wird!)
- Über sog. *abhängige Anwendungen* soll letztendlich beim Klick auf z.B. *Writer* alles automatisch durchgeführt werden, d.h. installieren (falls noch nicht geschehen), *bootstrap.ini* kopieren, und Programm starten!

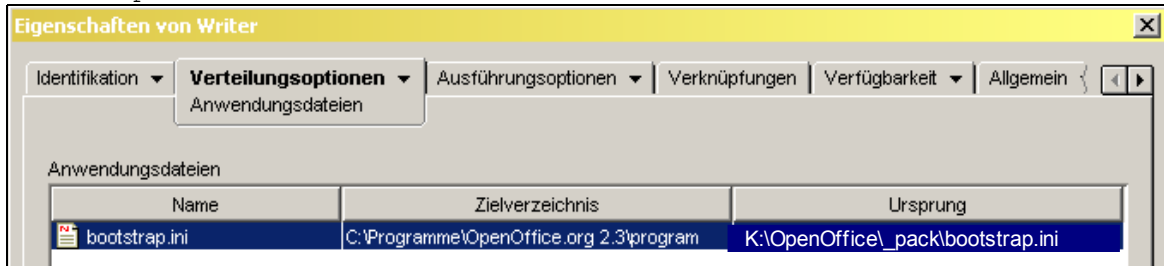
Aufgabe: OpenOffice komfortabel im Netz zur Verfügung stellen

- Erstellen Sie einen Snapshot von ML3-PC01! („Original“), bzw. gehen Sie zurück zu diesem Snapshot!
- Starten Sie die Maschine und melden Sie sich als *PgmAdmin-LFB* an!
- Legen Sie einen Ordner *K:\OpenOffice* und darunter den Ordner *_pack* an!
- Kopieren Sie die Dateien aus *OpenOffice_msi* nach *K:\OpenOffice_pack* !
- Starten Sie die *ConsoleOne* und erstellen Sie nach obiger Anleitung ein Anwendungsobjekt *OpenOffice-Installation*, basierend auf einer msi-Datei!
- Testen Sie als Lehrer / Schüler am zweiten Windows-PC ML3-PC02, ob die Verteilung funktioniert!
- Starten Sie z.B. *Writer* über das Windows-Startmenü! Es müsste der Registrierungsdialog erscheinen, den Sie mit minimalem Aufwand durchgehen können. (Die Eingaben sind nicht dauerhaft!)
- Im nächsten Schritt erstellen Sie drei (oder mehr) Anwendungsobjekte zum Start von *Writer*, *Calc* und *Impress* über den NAL! („Einfache Anwendungen“, keine msi-Anwendungen!)
- Testen Sie die Funktion der erstellten Objekte an ML3-PC02!
- Nun verknüpfen Sie noch die „Start-Objekte“ mit dem „Installationsobjekt“ als abhängige Anwendung.

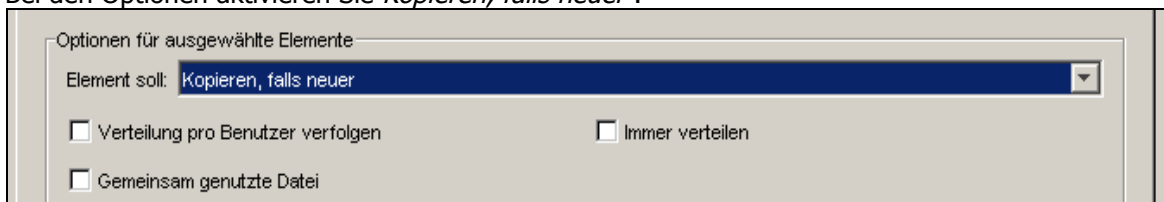


- Testen Sie auf einer neuen Maschine (ML3-PC02 mit dem Snapshot-Manager nach „Original“ zurücksetzen!), ob sich *OpenOffice* beim Klick auf *Writer* automatisch installiert und startet! Evtl. müssen Sie zur Installation im NAL auch auf *Überprüfen* klicken (Rechtsklick auf das *Writer*-Objekt. (Sie müssen jetzt wieder den Registrierungsdialog durchgehen!)
- Damit Benutzereinstellungen im Homeverzeichnis gespeichert werden, und der Registrierungsdialog nur einmalig auftaucht, wird jetzt noch die modifizierte *bootstrap.ini* automatisch in das entsprechende Verzeichnis kopiert!

- Kopieren Sie die modifizierte `bootstrap.ini` nach `K:\OpenOffice_pack!`
- Öffnen Sie in der *ConsoleOne* das „Start-Objekt“ *Writer!*
- Gehen Sie auf die Registerkarte *Verteilungsoptionen / Anwendungsdateien* und fügen Sie die `bootstrap.ini` hinzu!



- Bei den Optionen aktivieren Sie *Kopieren, falls neuer* :



Hinweis:

Die Verteilung dieser Datei sollten Sie bei allen Startobjekten hinzufügen (*Calc, Impress* usw...)

- Testen Sie als `SpechtB-LFB` an `ML3-PC02` im `NAL` mit *Überprüfen*, ob die neue `bootstrap.ini` wirklich kopiert wird!
(Datei in `C:` editieren und prüfen, ob der Eintrag „`UserInstallation=file:///H:/Profil`“ vorhanden ist!)
- Beim Start von *Writer* erscheint jetzt letztmalig (!) der Registrierungsdialog; Sie können jetzt endlich sinnvolle Daten (Name, Vorname) eingeben, sie bleiben erhalten!
- Überprüfen Sie, ob im Homeverzeichnis Dateien in `H:\Profil` geschrieben worden sind!
- Melden Sie sich als `GrossA-LFB` an und starten Sie erneut den *Writer!* Normalerweise sollte das Paket jetzt nicht mehr verteilt werden, der Registrierungsdialog müsste jedoch erscheinen. Ebenso sollten jetzt wieder Dateien im Homeverzeichnis erstellt werden.

Wichtiger Hinweis:

An dieser Stelle werden Sie feststellen, dass sich das Paket als Schüler wg. fehlender Administratorrechte auf der *WXP-Station* nicht installieren, bzw. der *Writer* nicht starten lässt, es erscheint ein `msi-Fehler` wg. fehlender Rechte! Die Rechte sollten eigentlich über *ZenWorks* bzw. die Gruppenrichtlinien bereitgestellt werden! Eine genauere Analyse des `msi-Paketes` hat ergeben, dass im Paket selbst ein Eintrag vorhanden ist, der bewirkt, dass eine Installation nur mit Administratorrechten möglich ist. Hier muss man eine `mst-Datei` (`openofficeorg24.mst`) verwenden, die diesen Eintrag unwirksam macht! Diese Datei wurde mit dem *AdminStudio* erzeugt.

- Wenn alles funktioniert, können Sie (nur) das Installationsobjekt im `NAL` auf unsichtbar schalten. (Bei den Verknüpfungen das Häkchen für die Anzeige im `NAL` entfernen)

Abschließender Hinweis zu OpenOffice:

Wenn man eine neue Version von *OpenOffice* bereitstellen will, muss der Registrierungsdialog bei jedem Update erneut durchgegangen werden. Dies ist relativ lästig, lässt sich aber mit einem undokumentierten Startparameter leicht unterdrücken. Dazu muss beim Programmaufruf der Parameter `-nofirststartwizard` angegeben werden. Für eine Standardinstallation sieht dies dann wie folgt aus:
`"C:\Programme\OpenOffice.org 2.x\program\swriter.exe" -nofirststartwizard`

2.3. Acrobat Reader 8.1 tunen und verteilen

Aufgabe: Ein angepasstes Installationspaket für den Reader 8.1 erstellen

- Setzen Sie die VM ML3-PC01 auf „Original“ zurück!
- Melden Sie sich als `PgmAdmin-LFB` an und erstellen Sie die benötigten Ordner in `K:\`
- Kopieren Sie die Dateien aus dem Ordner `acrobat8_msi` nach `K:\Acrobat8_pack!`
- Installieren Sie lokal den *Tuner* für den *Reader 8* (Standardeinstellungen verwenden)
- Starten Sie den *Tuner*, bearbeiten Sie die msi-Datei und nehmen Sie im ersten Versuch einige wenige sinnvolle Einstellungen vor!
Vorschläge:
 - festlegen, ob der Reader lokal oder serverbasiert installiert wird
 - automatisches Update deaktivieren
 - usw...
- Speichern Sie die erzeugte mst-Datei unter `AcroRead.mst` im gleichen Ordner wie die msi-Datei!
- Schließen Sie den *Tuner*, lassen Sie die veränderte msi-Datei aber NICHT speichern! (Veränderungen sollten nur in der mst-Datei stehen!)
- Erstellen Sie ein Anwendungsobjekt zur Verteilung des Readers!
Gehen Sie wie oben beschrieben vor; bei *MSI / Transformation* müssen Sie noch die mst-Datei eintragen.
- Verteilen UND starten können Sie den Reader, wenn Sie bei den *Ausführungsoptionen* direkt den Pfad zur Datei angeben! Ein eigenes Start-Objekt müssen Sie hier nicht anlegen, weil es sich nur um **eine** Anwendung handelt.
- Testen Sie wieder auf unterschiedlichen Maschinen mit verschiedenen Kennungen!
Im Gegensatz zu OpenOffice lässt sich der Reader auch als Schüler installieren, hier werden die benötigten Rechte von ZenWorks geliefert.

2.4. Den USB-Drive-Letter-Manager verteilen

Der USB-Drive-Letter-Manager ist ein kleines, aber feines Tool, mit dem Sie jeden Card-Reader in den Griff bekommen. Das Problem, dass ein Card-Reader sich über wichtige Netzlaufwerke legt, und somit die Mappings nicht mehr stimmen, gehört damit der Vergangenheit an! Das Tool sollte eigentlich auf jedem Netz-PC installiert sein.

Aufgabe: USBDLM.msi auf allen PC's im Netz automatisch verteilen

- Setzen Sie die ML3-PC01 auf „Original“ zurück!
- Melden Sie sich als `PgmAdmin-LFB` an und erstellen Sie die benötigten Ordner in `K:\`
- Kopieren Sie die Dateien aus dem Ordner `usbdlm_msi` nach `K:\usbdlm_pack!`
- Erstellen Sie ein neues Anwendungsobjekt und verknüpfen Sie es mit allen **Arbeitsstationen** per *Ausführung erzwingen!*
- Testen Sie die Verteilung! (Verzeichnis `usbdlm` auf `C:\` suchen!)
- Prüfen Sie, ob das Tool automatisch gestartet wurde!
(Im *Windows-Task-Manager* unter *Prozesse* nach `usbdlm.exe` suchen!)
- Falls JA: Fertig!
- Falls NEIN: Starten Sie das Tool von Hand im Installationsordner!
Erzeugen Sie ein neues Objekt zum automatischen Start nach der Verteilung!