



WIHU

– WINDOWS INSTALLATION HELPER UTILITY –

offizielle

**deutsche
Bedienungsanleitung
zur Version 2.1.2.0**

Programm entwickelt von:
Benjamin Kalytta

Anleitung geschrieben von:
Andreas H.

Augsburg, im August 2004
(Anleitung v1.0827.0)

Inhaltsverzeichnis

Inhaltsverzeichnis	2
Abbildungsverzeichnis.....	6
Tabellenverzeichnis	6
Vorwort & Danksagung – einleitende Worte.....	7
1 Programmbeschreibung.....	8
2 Systemanforderungen & Installation.....	9
3 Programmablauf.....	10
3.1 primäres Benutzerkonto erstellen	10
3.2 Benutzerkonten erstellen und bearbeiten.....	12
3.3 Software	16
3.4 Installationsstatus.....	17
4 Konfiguration	19
4.1 Befehlszeilenparameter.....	20
4.1.1 Allgemeines zum Programmaufruf.....	20
4.1.2 mögliche Fehlerquellen beim Aufruf von WIHU.....	21
4.1.3 Pfadangaben & (WIHU-)Variablen an der Befehlszeile	22
4.1.4 Übersicht der Befehlszeilenparameter	24
4.1.4.1 /? oder /help	24
4.1.4.2 /AutoExit=<Sekunden>	24
4.1.4.3 /AutoInstall=<Sekunden>.....	25
4.1.4.4 /beep=<Sekunden>	26
4.1.4.5 /RestartWait=<Sekunden>.....	27
4.1.4.6 /NoRestartChange	28
4.1.4.7 /NoCancel	28

4.1.4.8	/SkipRestart.....	28
4.1.4.9	/SkipSettings	29
4.1.4.10	/SkipSoftware.....	29
4.1.4.11	/Admin=<Name>	30
4.1.4.12	/AdminPwd=<Paßwort>	30
4.1.4.13	/User=<Name>	31
4.1.4.14	/UserPwd=<Paßwort>.....	32
4.1.4.15	/AutoLogon	33
4.1.4.16	/UseCurrent	33
4.1.4.17	/Computer=<Name>	34
4.1.4.18	/Workgroup=<Name>.....	34
4.1.4.19	/Owner=<Name>	35
4.1.4.20	/Org=<Name>	35
4.1.4.21	/INI=<Datei>	36
4.1.4.22	/Users=<Datei>.....	37
4.1.4.23	/EncryptINI	38
4.1.4.24	/DecryptINI.....	39
4.1.4.25	/Log=<Datei>	40
4.1.4.26	/verbose=[0 1 2 3].....	40
4.2	Konfigurationsdatei	42
4.2.1	Allgemeines zur dateibasierten Konfiguration	42
4.2.2	Numerierungssystem von Anweisungen	43
4.2.3	Abschnitt [<beliebige Bezeichnung>]	44
4.2.3.1	command.x.y.z=<Befehl>	45
4.2.3.2	workdir.x.y.z=<Pfadangabe>.....	46
4.2.3.3	description.x.y.z=<Beschreibungstext>.....	46
4.2.3.4	selected.x.y.z=[0 1]	47
4.2.3.5	hidden.x.y.z=[0 1]	48
4.2.3.6	collapsed.x.y.z=[0 1]	48
4.2.3.7	file.x.n=<Datei>?[= < >]<Versionsangabe>.....	49
4.2.3.8	file.comparison.x.y.z=[AND OR].....	51

4.2.3.9	key.x.n=[HKLM HKCU]:<Reg.schlüssel>?[=! <>]<Zeichenkette>	52
4.2.3.10	key.comparison.x.y.z=[AND OR]	53
4.2.4	Abschnitt [Environment]	54
4.2.4.1	Hilfsumgebungsvariablen für den Installationsprozeß definieren	54
4.2.4.2	Dauerhafte Umgebungsvariablen pro Benutzer definieren	56
4.2.4.3	Standardvorgaben benutzerkontospezifischer Ordner manipulieren	57
4.2.5	Abschnitt [Users]	61
4.2.5.1	user.x=<Name>	62
4.2.5.2	group.x=[0 1 2].....	62
4.2.5.3	status.x=<Kennzahl>	64
4.2.5.4	profile.x=<Pfadangabe>	65
4.2.5.5	RoamingProfile.x =<Pfadangabe>	65
4.2.5.6	comment.x=<Beschreibung>.....	65
4.2.5.7	script.x=<Datei>	66
4.2.5.8	domain.x=<Name>	66
4.2.5.9	password.x=<Paßwort>	66
4.2.5.10	crypted.x=<Wert>.....	67
4.2.5.11	<reservierter Variablenname>.x=<Pfadangabe>.....	67
4.2.5.12	default=<Indexnummer>	68
4.2.6	Abschnitt [Users.Operation]	69
4.2.6.1	rename.n=<altes Konto>{ @Domäne }, <neues Konto>.....	70
4.2.6.2	enable.n=<Benutzerkonto>{ @Domäne }.....	70
4.2.6.3	disable.n=<Benutzerkonto>{ @Domäne }.....	71
4.2.6.4	delete.n=<Benutzerkonto>{ @Domäne }	71
4.2.7	Abschnitt [Settings]	72
4.2.7.1	PageHeaderTitle.x=<Überschrift>.....	72
4.2.7.2	PageHeaderSubTitle.x=<Kurzbeschreibungstext>.....	72
4.2.7.3	dimension.width=<Pixelwert>.....	73
4.2.7.4	dimension.height=<Pixelwert>.....	73
4.2.7.5	dimension.update=[0 1]	73
4.2.7.6	OsdText=<Text>.....	74

4.2.7.7	AutoExit=<Sekunden>	74
4.2.7.8	AutoInstall=<Sekunden>	74
4.2.7.9	beep=<Sekunden>	74
4.2.7.10	RestartWait=<Sekunden>.....	75
4.2.7.11	NoRestartChange=1	75
4.2.7.12	NoCancel=1	75
4.2.7.13	SkipRestart=1.....	75
4.2.7.14	SkipSettings=1	75
4.2.7.15	SkipSoftware=1	75
4.2.7.16	Admin=<Name>	75
4.2.7.17	AdminPwd=<Paßwort>	75
4.2.7.18	User=<Name>.....	75
4.2.7.19	UserPwd=<Paßwort>.....	75
4.2.7.20	AutoLogon=1	75
4.2.7.21	Computer=<Name>	76
4.2.7.22	Workgroup=<Name>.....	76
4.2.7.23	Owner=<Name>	76
4.2.7.24	Org=<Name>	76
4.2.7.25	Log=<Datei>.....	76
4.2.7.26	verbose=[0 1 2 3].....	76
5	Unterstützung des Programmentwicklers	77
6	Verweise und Quellen.....	78
7	Rechtliches	79
7.1	zum Computerprogramm	79
7.2	zur Anleitung	79

Abbildungsverzeichnis

Abbildung 1: Dialog Nr. 0 – primäres Benutzerkonto erstellen.....	10
Abbildung 2: Dialog Nr. 1 – Benutzerkonten erstellen und bearbeiten.....	12
Abbildung 3: Dialog 1a – Reiter Benutzerkonto	13
Abbildung 4: Dialog 1b – Reiter Systemordner	14
Abbildung 5: Dialog 1c - Reiter Umgebungsvariablen	15
Abbildung 6: Dialog Nr. 2 – Softwarekomponenten.....	16
Abbildung 7: Dialog Nr. 3 - Installationsstatus	18
Abbildung 8: Befehlszeilenparameterkurzübersicht nach Aufruf von WIHU /?	23
Abbildung 9: Meldungsbox beim Herunterfahren des Systems	27
Abbildung 10: Windowsdialog zu dauerhaften Benutzervariablen.....	57
Abbildung 11: Anpassung der Kurztexte in den WIHU-Fenstern.....	72
Abbildung 12: Bildschirmanzeige der verbleibenden Countdowndauer	74

Tabellenverzeichnis

Tabelle 1: Kennzeichnung des Benutzerkontenstatus	12
Tabelle 2: WIHU-spezifische Umgebungsvariablen	22
Tabelle 3: Auswahl der im Protokollfenster anzuzeigenden Meldungen.....	40
Tabelle 4: Vergleichsoperatoren.....	49
Tabelle 5: Fallunterscheidung bei fehlender Dateiversionsinformation.....	49
Tabelle 6: reservierte Variablennamen für benutzerspezifische Ordner	59
Tabelle 7: Sondervariablen für die Konfigurationsdatei.....	59
Tabelle 8: Festlegung der Gruppenzugehörigkeit eines Benutzerkontos	62
Tabelle 9: Überblick über die Statuskennzahlen bei Benutzerkonten	64

Vorwort & Danksagung – einleitende Worte

Nicht vom Umfang dieser Anleitung schockieren lassen! Ja, ich weiß, das Programm, um welches es sich hier dreht, ist nicht mal ein halbes Megabyte groß und doch umfaßt dieses Dokument hier satte 79 Seiten, so um die 12457 Worte und gesalzene 87504 Zeichen, Deutsch eben eine Sprache der Komposita. Und man möchte schnell annehmen, daß WIHU wohl komplizierter sei als es auf den ersten Blick erscheint. Diese Befürchtung kann ich aber entkräften. Diese Anleitung versucht nur, mehr oder minder ausführlich auf alle möglichen Fragen rund um WIHU einzugehen, um möglichen Problemen und Unklarheiten gleich von Anfang an aus dem Weg zu gehen. Dabei verliert man dann natürlich schon mal ein paar Worte mehr. Außerdem soll diese Anleitung auch Nicht-Gurus weiterhelfen. Zudem war mein Ansporn beseelt durch all die mit Verlaub gesagt meiner Meinung nach höchst mangelhaften Hilfestellungen bei namhaften Programmen, Namen seien hier mal wieder außen vor; ich hoffe, ich hab's zumindest im vorliegenden Fall ein wenig besser gemacht. Daher gibt's neben anderen Aspekten für schnelleres Finden auch ein Inhaltsverzeichnis sowie prinzipiell eine logische Struktur, ich hoffe jedenfalls, daß sich diese nicht nur mir erschließt. Für konstruktive Verbesserungsvorschläge habe ich aber natürlich nach wie vor ein offenes Ohr.

„Geschrieben hab ich's ganz allein“, strenggenommen träfe das zu, wenn man nur auf die Tastatur geachtet hätte, egal, ob sich das Zitat nun auf die Programmentwicklung oder die Entstehung dieser Anleitung bezieht. Doch wäre es schlicht vermessen, all die Leute auszublenden, die hier kräftig mitgeholfen haben. Und so mögen sich alle Deutschlehrer meinetwegen im Grabe umdrehen, wenn ich die Danksagung an den Anfang des Dokumentes stelle, ich laß es mir aber trotzdem nicht nehmen, genau hier zu aller erst natürlich Benjamin Kalytta für die Geduld mit meinen vielen Anfragen bezüglich seines Programms zu danken, darüber hinaus aber natürlich auch all den Leuten im MSFN Forum, die mit Ihren Anfragen und Vorschlägen das Programm erst zu dem gemacht haben, was es ist (und so auch diese Anleitung indizierten), na, und zuletzt auch meinen Eltern und den hiesigen Stadtwerken für das Bezahlen bzw. Bereitstellen des Stroms, damit mein PC auch schön warmlief. Und all die, die ich vergessen hab ... nun, lassen wir das.

Zu guter Letzt wünsche ich dieser Anleitung, daß sie viel Nutzen stiftet und deren Benutzern Spaß damit, sofern man das überhaupt sagen kann. Ach ja: Die alte Rechtschreibung ist pure Absicht. Und nun endlich in medias res ...

1 Programmbeschreibung

KURZ: automatisierte und unbeaufsichtigte Installation von Software unter Windows sowie Anlegen von Benutzerkonten

Die Abkürzung WIHU steht für „Windows Installation Helper Utility“. Dieses kleine aber feine Hilfsprogramm erlaubt es, nach individuellen Vorgaben vollautomatisiert zum einen beliebige Software zu installieren und zum anderen auch Benutzerkonten unter Windows anzulegen. Haupteinsatzgebiet des Programms liegt damit in unbeaufsichtigten Installationen (engl. „unattended installations“), wo ohne oder ggf. nur mit minimalen Eingaben seitens des PC-Benutzers zahlreiche Softwarepakete aller Art installiert werden sollen. Dies schließt alles ein, was unter Windows regulär als Programm ausgeführt werden kann, wie zum Beispiel:

- Windowskomponenten wie DirectX oder das .NET Framework
- Hotfixes und Patches z.B. zum Schließen von Sicherheitslücken
- reguläre Anwendungssoftware wie bspw. das Office-Paket
- Registrierungseinstellungen („registry tweaks“)
- Treiber, sofern diese über eine reguläre Installationsroutine verfügen

Wer sich die Funktionsweise besser mit Bildschirmfotos vorstellen kann, sei auf das Kapitel 3 „Programmablauf“ verwiesen.

2 Systemanforderungen & Installation

KURZ: Windows XP oder Windows 2000; Programm entpacken und loslegen

Wie der Programmname schon verrät, handelt es sich dabei um ein **Windows-Programm**. Getestet wurde es ausschließlich unter Windows XP und Windows 2000. Damit sollten die heute gebräuchlichsten Windowssysteme, bei denen unbeaufsichtigte Installationen stattfinden, abgedeckt sein. WIHU arbeitet übrigens unabhängig vom Vorhandensein oder der Version eines Service Packs, das spielt keine Rolle.

Eine Portierung auf andere Systeme wie bspw. Linux oder Macintosh erfolgt naturgemäß nicht, da dort Programminstallation anders ablaufen.

Von seiten der **hardware-technischen** Ausstattung wie Arbeitsspeichergröße oder Festplattenplatz gibt es keine Einschränkungen; es gilt einzig und allein: Falls Windows läuft, dann klappt's auch mit WIHU! ☺

Software-technisch wird ein simpler Texteditor benötigt, um WIHU für den unbeaufsichtigten Modus konfigurieren zu können. Ein solcher Editor ist im Windows-Lieferumfang bereits standardmäßig enthalten („Notepad“).

Zur **Installation** das Programm in Form der ZIP-Datei herunterladen, in ein beliebiges Verzeichnis entpacken und dort die Datei WIHU.EXE aufrufen ... voilà! ☺

3 Programmablauf

Bei Aufruf WIHUs ohne Befehlszeilenparameter sehen Sie die in diesem Kapitel beschriebenen Dialogfenster. Betrachten Sie diese einmal näher, um sich einen ersten Eindruck des prinzipiellen Programmablaufs zu machen. Wie ersichtlich bietet WIHU Ihnen umfangreiche Einstellungsmöglichkeiten rund um dessen zwei Haupteinsatzgebiete Benutzerkonten und Softwareinstallation. Diese Anleitung beschreibt, wie Sie diese Dialoge soweit automatisieren können, so daß nach dem Programmaufruf keine Benutzereingaben mehr von Nöten sind, um eine unbeaufsichtigte Installation durchzuführen.

3.1 primäres Benutzerkonto erstellen




Abbildung 1: Dialog Nr. 0 – primäres Benutzerkonto erstellen¹

¹ Die Dialognummern entsprechen den benötigten Indizes für Kapitel 4.2.7.1 und 4.2.7.2

Hiermit richten Sie das sog. „primäre Benutzerkonto“ ein. Dies soll vor allem zwei Zwecken dienen: Zum einen sollen damit im Rahmen von WIHU die eigentlichen Softwareinstallationen durchgeführt werden und zum anderen soll es später beim alltäglichen Gebrauch des Rechners als reguläres „Arbeitskonto“ dienen, unter dem Sie die täglich anfallenden Routineaufgaben standardmäßig erledigen. Im Dialog können Sie für dieses Konto zunächst die wichtigsten Daten eingeben, was den Benutzernamen und das Paßwort umfaßt. Im nächsten Dialogfenster haben Sie anschließend die Möglichkeit, das Konto detailgenauer zu konfigurieren. Als Benutzername wird von vielen oft einfach praktischerweise der eigene Vorname gewählt. Das Paßwort müssen Sie aus Sicherheitsgründen zweimal eingeben, um Tippfehler auszuschließen; solange beide Eingaben nicht übereinstimmen, bleibt die Schaltfläche „Weiter“ inaktiv. Zudem müssen Sie einen Computernamen als auch eine Arbeitsgruppe eingeben.

Generell wird empfohlen, vom primären Benutzerkonto Gebrauch zu machen! WIHU verknüpft mit der Verwendung dieses Kontos nämlich eine besondere Funktionalität: Für den Zeitraum der Softwareinstallation sorgt WIHU im Rahmen dieses Kontos automatisch für ausreichende Zugriffsrechte, unabhängig von der Gruppenzugehörigkeit des Kontos (Benutzer- vs. Administratorengruppe). Nachdem alle Installationsvorgänge abgeschlossen sind, setzt es die Rechte mit der ersten Benutzeranmeldung des Kontos auf den normalen Benutzerrechtstatus zurück. Dieses prinzipielle Verhalten wird ausdrücklich angeraten, da zum einen für Installationen in annähernd allen Fällen volle Zugriffsrechte benötigt werden, die weitere Verwendung dieser Rechte für die tägliche Arbeit am PC aber zu stark erhöhten Sicherheitsrisiken (Anfälligkeit für Viren & Trojaner, Unversehrtheit Ihrer persönlichen Daten, etc.) führt und zudem nicht notwendig ist.

Natürlich können Sie dieses Verhalten aber auch außer Kraft setzen, indem Sie das Kästchen „Angemeldeten Benutzer verwenden“ ankreuzen; dann werden die Installationen unter den Rechten des Kontos ausgeführt, mit dem Sie gerade angemeldet sind. WIHU zeigt Ihnen den Namen dieses Kontos oben im Dialogfenster an. Achten Sie aber darauf, Installationen immer mit Administratorrechten durchzuführen, da nur dann die nötigen Schreibzugriffe auf alle Systembereiche zugelassen werden.



Mehr Informationen zur Arbeit unter Windows mit eingeschränkten Rechten können Sie dem beigefügten c't-Artikel entnehmen. Zur Anzeige benötigen Sie mind. den Adobe Acrobat Reader v6.

Die weiteren Dialogeingabefelder betreffen keine Benutzerkonten, sondern sind Daten bezüglich des Rechners. Mehr Informationen hierzu im späteren Verlauf der Anleitung.

3.2 Benutzerkonten erstellen und bearbeiten



Abbildung 2: Dialog Nr. 1 – Benutzerkonten erstellen und bearbeiten

Dieser Dialog zeigt Ihnen alle vorhandenen Benutzerkonten in einer Übersicht an. Dies umfaßt das im vorhergehenden Dialog erstellte primäre Benutzerkonto sowie alle in der Konfigurationsdatei von WIHU definierten Konten. Mit Hilfe dieses Dialoges können Sie entweder neue Konten hinzufügen, vorhandene löschen oder genauer bearbeiten. Zur verwendeten Symbolik:




Symbol	Bedeutung
	neues, im Rahmen von WIHU erstelltes Benutzerkonto
	ein schon vor Aufruf WIHUs existierendes Benutzerkonto, welches nun in WIHU geändert werden soll und dazu im Abschnitt [Users] aufgeführt wurde
	siehe oben, Konto ist jedoch deaktiviert

Tabelle 1: Kennzeichnung des Benutzerkontenstatus

Durch einen Klick auf die Schaltfläche „Laden“ können Sie über die graphische Benutzeroberfläche von WIHU alternative Konfigurationsdateien einlesen, die Sie ggf. für unterschiedliche Zwecke vorbereitet haben. Im Falle von Abbildung 2 wird aus der angegebenen Datei nur der Abschnitt mit den benutzerkontospezifischen Daten ausgelesen, alles andere wird ignoriert. Es kann zudem pro Dialog immer nur eine derartige Datei ausgewählt und von WIHU abgearbeitet werden, d.h. beim erneuten Laden einer anderen Datei verfallen die Einstellungen aus der vorherigen Datei. Zu diesem Zeitpunkt in WIHU werden Ihre Einstellungen auch noch nicht ins System übernommen. Dies erfolgt erst, nachdem WIHU mit dem eigentlichen Installationsprozeß begonnen hat.

Sobald Sie die Eigenschaften eines Kontos über die graphische Benutzeroberfläche anpassen bzw. ein neues Konto anlegen möchten, erscheint ein weiteres Fenster. Je nach Benutzerstatus (siehe Tabelle 1) werden Ihnen hier unterschiedliche Konfigurationsmöglichkeiten angeboten. Detailgenauere Erklärungen der Möglichkeiten im Verlauf der Anleitung.

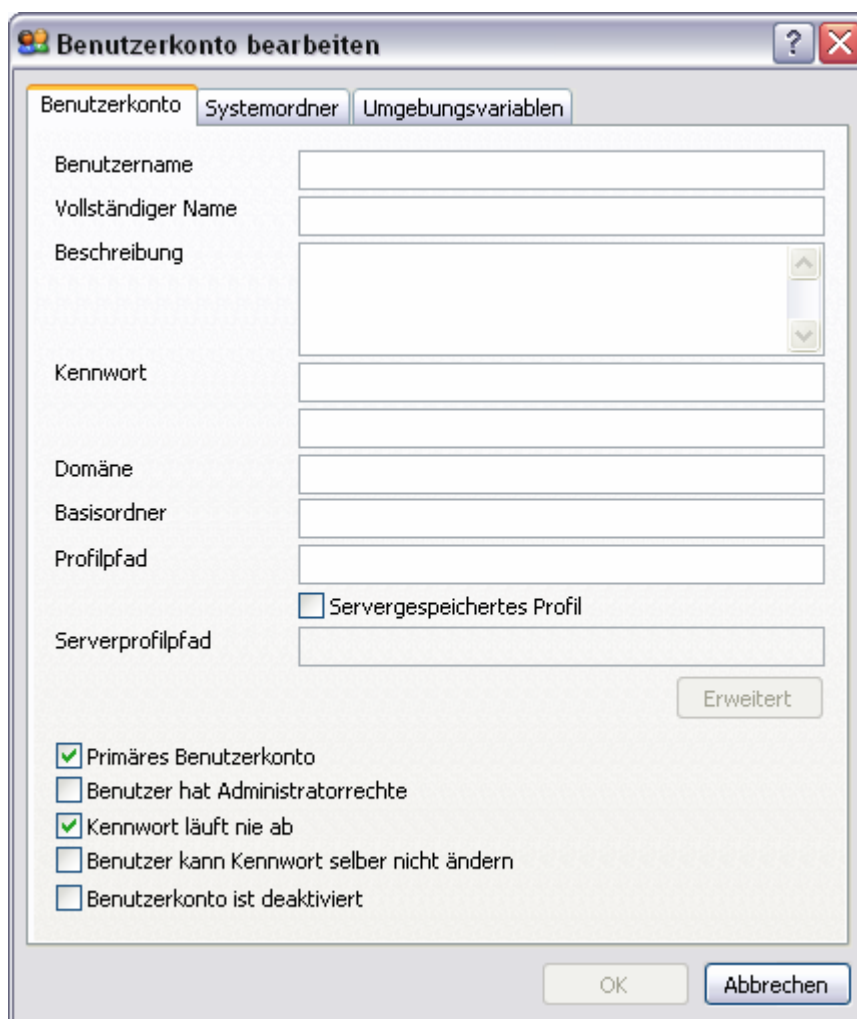


Abbildung 3: Dialog 1a – Reiter Benutzerkonto

Eigenschaft / Feld	Kurzanmerkung
Benutzername	entspricht der Sondervariablen %ThisUser%, s. Tabelle 7; bei bereits existenten Konten nicht abänderbar
Basisordner	entspricht der Sondervariablen %ThisHome%, s. Tabelle 7
Profilpfad	entspricht der Sondervariablen %ThisProfile%, s. Tabelle 7
prim. Benutz.konto	siehe hierzu u.a. auch Kapitel 4.1.1.
Administratorrechte	empf. Einstellung für primäres Benutzerkonto: <i>nicht</i> angekreuzt

Im Folgenden sehen Sie den zweiten der drei möglichen Reiter:

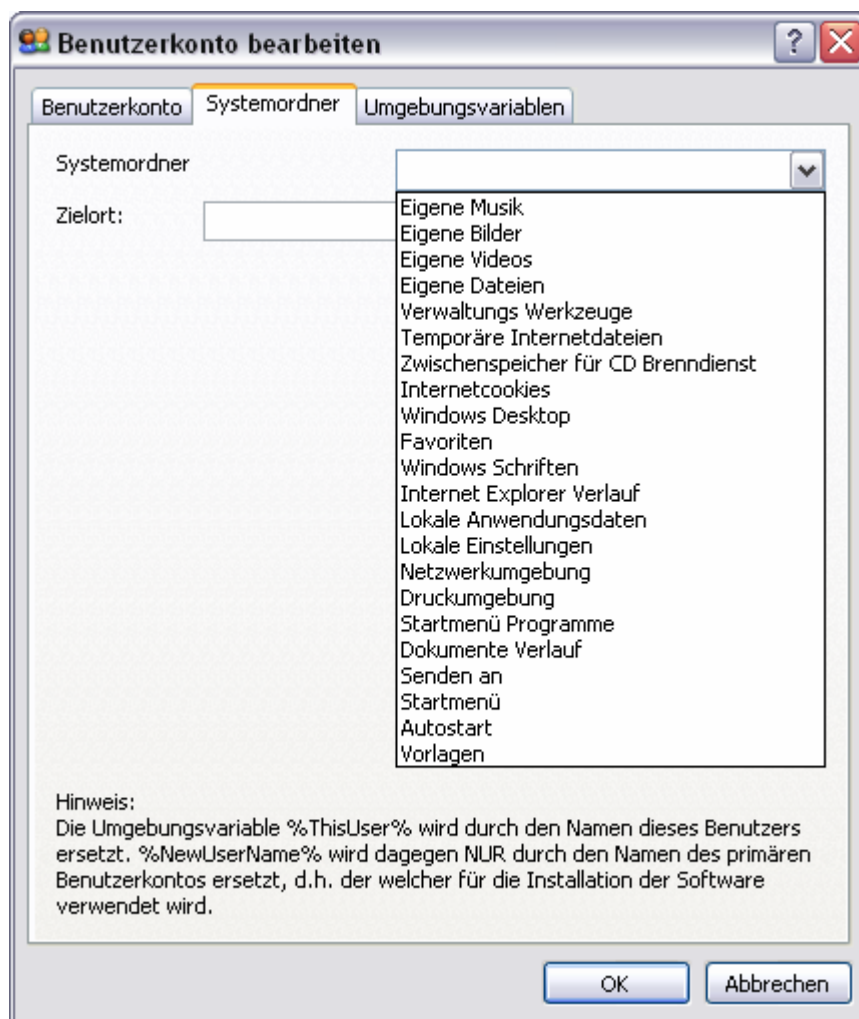


Abbildung 4: Dialog 1b – Reiter Systemordner

Hier können Sie für den ausgewählten Benutzer die Standardvorgaben von Windows für die dargestellten Systemordner modifizieren. Ausführlichere Informationen hierzu finden Sie in Kapitel 4.2.4.3.

Beachten Sie bitte, daß WIHU selbst im Kontext eines Kontos mit Administratorrechten ausgeführt werden muß, um an den Benutzerkonten des Betriebssystems Änderungen egal welcher Art durchführen zu können. Die Erstellung und Verwendung des primären Benutzerkontos hat keinen Einfluß hierauf, ist davon unabhängig.

Es folgt der letzte Reiter, der Ihnen benutzerkontospezifische Einstellungen erlaubt:

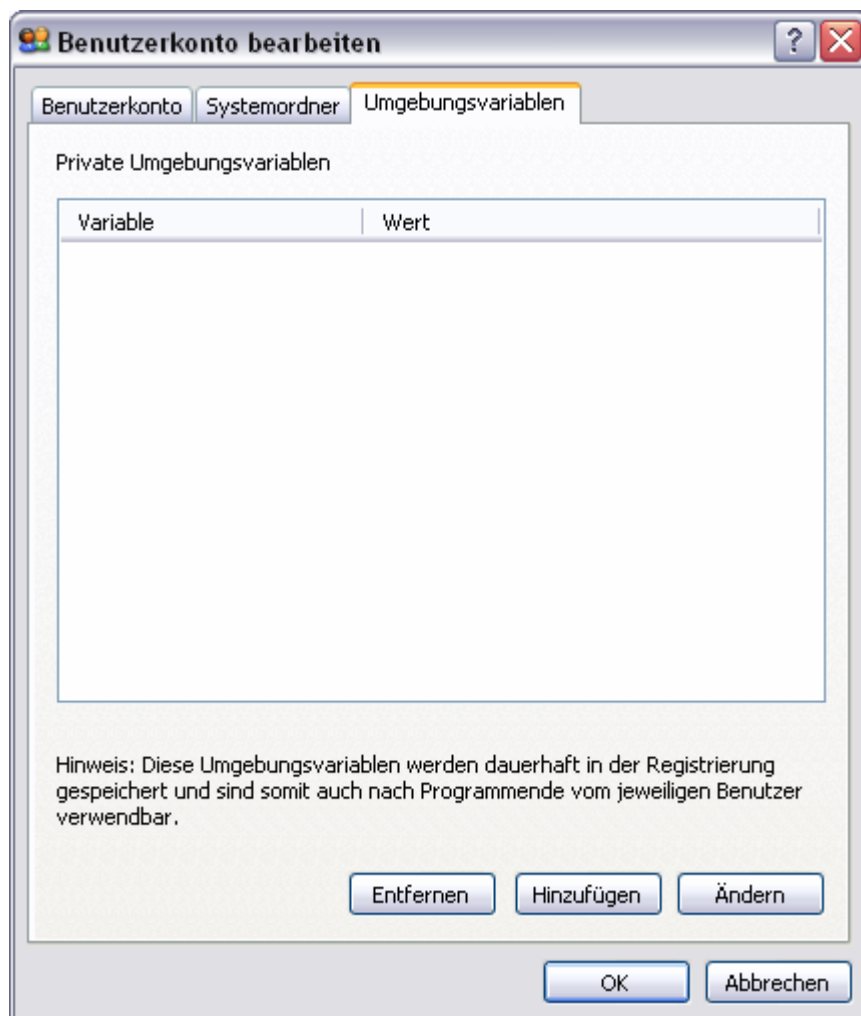


Abbildung 5: Dialog 1c - Reiter Umgebungsvariablen

Hier können Sie für nur für den jeweiligen Benutzer geltende Umgebungsvariablen definieren, die in Windows zukünftig immer unter dem entsprechenden Benutzerprofil verfügbar sein werden. Ausführlichere Informationen dazu finden Sie in Kapitel 4.2.4.2.

3.3 Software

Als letzten Schritt vor Beginn der Programminstallationen sehen Sie den Dialog aus Abbildung 6, mit dessen Hilfe Sie festlegen, welche Programme im nächsten Schritt von WIHU automatisch installiert werden sollen. WIHU greift hierfür ausschließlich auf eine Konfigurationsdatei zurück, welche all diese Informationen (Programmnamen, Installationsparameter, Pfade, etc.) beinhaltet. In dem Dialog der graphischen Benutzeroberfläche können Sie nur eine Aus- oder Abwahl bereits vorhandener Punkte vornehmen, jedoch keine neuen anlegen, vorhandene löschen oder editieren. Auf welche Weise die benötigte Konfigurationsdatei erstellt wird, erfahren Sie im Kapitel 3.4 dieser Anleitung. Wie ersichtlich erfolgt die Softwareauswahl außerdem mit der unter Windows gewohnten kaskadierenden Baumstruktur.

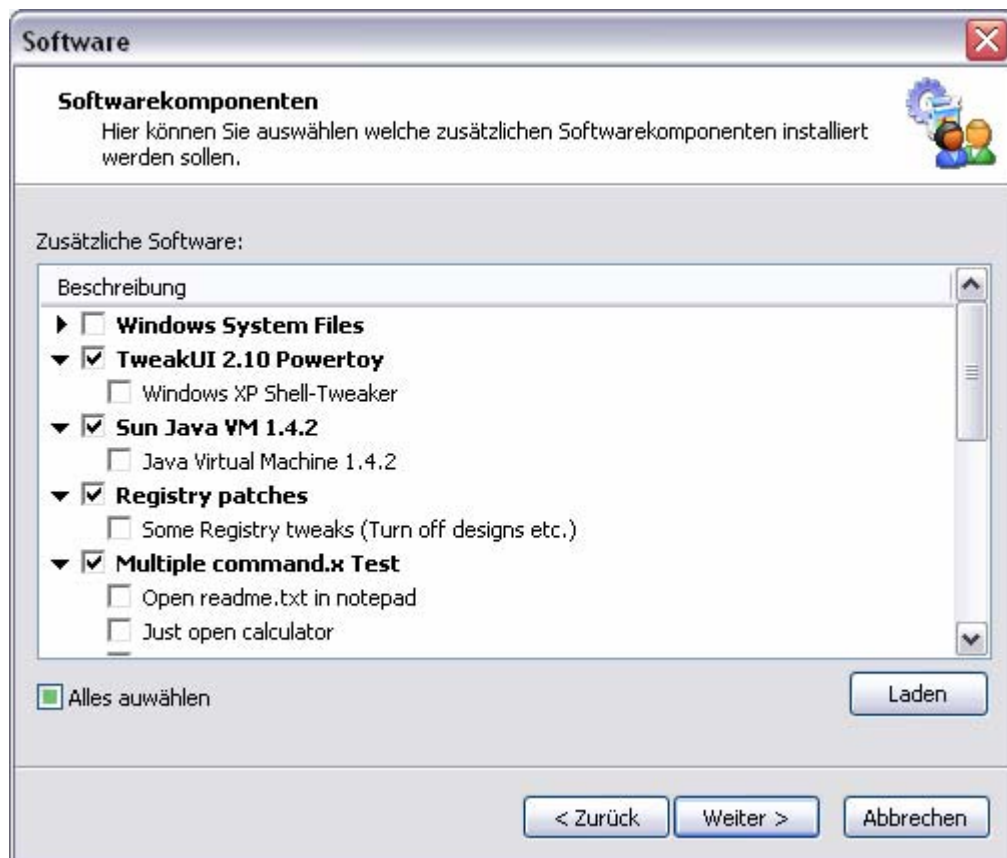


Abbildung 6: Dialog Nr. 2 – Softwarekomponenten

Durch einen Klick auf die Schaltfläche „Laden“ können Sie über die graphische Benutzeroberfläche von WIHU alternative Konfigurationsdateien einlesen, die Sie ggf. für unterschiedliche Zwecke vorbereitet haben. Im Falle von Abbildung 6 werden aus der angegebenen Datei nur die installationsrelevanten Anweisungen ausgelesen, alles andere wird ignoriert. Es kann zudem pro Dialog immer nur eine derartige Datei ausgewählt und von WIHU abgearbeitet werden, d.h. beim erneuten Laden einer anderen Datei verfallen die Einstellungen aus der vorherigen Datei. Zu diesem Zeitpunkt in WIHU werden Ihre Einstellungen auch noch nicht ins System übernommen. Dies erfolgt erst, nachdem WIHU mit dem eigentlichen Installations(-arbeits-)prozeß begonnen hat.

Durch einen Klick auf „Weiter“ beginnt WIHU mit der eigentlichen Installation der ausgewählten Komponenten und übernimmt nun zudem die gewünschten Einstellungen an den Benutzerkonten, sog. Arbeitsprozeß.

WICHTIG:

Installationen unter Windows benötigen in Gro der Fälle Administratorrechte, da viele Installationsroutinen auf Systembereiche zugreifen, wo eingeschränkte Benutzerrechte keinen Schreibzugriff zuließen! Um daher Probleme von vornherein zu vermeiden, ist es anzuraten, WIHU mit Administratorprivilegien auszuführen bzw. das [primäre Benutzerkonto](#) von WIHU zu verwenden, da es ansonsten die geplanten Aufgaben u.U. nicht fehlerfrei ausführen kann.

3.4 Installationsstatus

Während und nach der Installation sehen Sie das folgende Fenster. Es signalisiert den eigentlichen Arbeitsprozeß, in dem alle vorhergehend festgelegten Änderungen schließlich ins System übernommen werden. Solange der Arbeitsprozeß am Laufen ist, zeigt Ihnen WIHU einen Fortschrittsbalken an und quittiert die Ausführung der Befehle mit entsprechenden Hinweisen im Abschnitt „Installationsprotokoll“. Die weitere Ausführung des Prozesses können Sie mit „Abbrechen“ beenden. WIHU fragt hier aber zur Sicherheit nochmals extra nach. Nach der Installation wird das Programm mit einem Klick auf „Fertigstellen“ beendet.

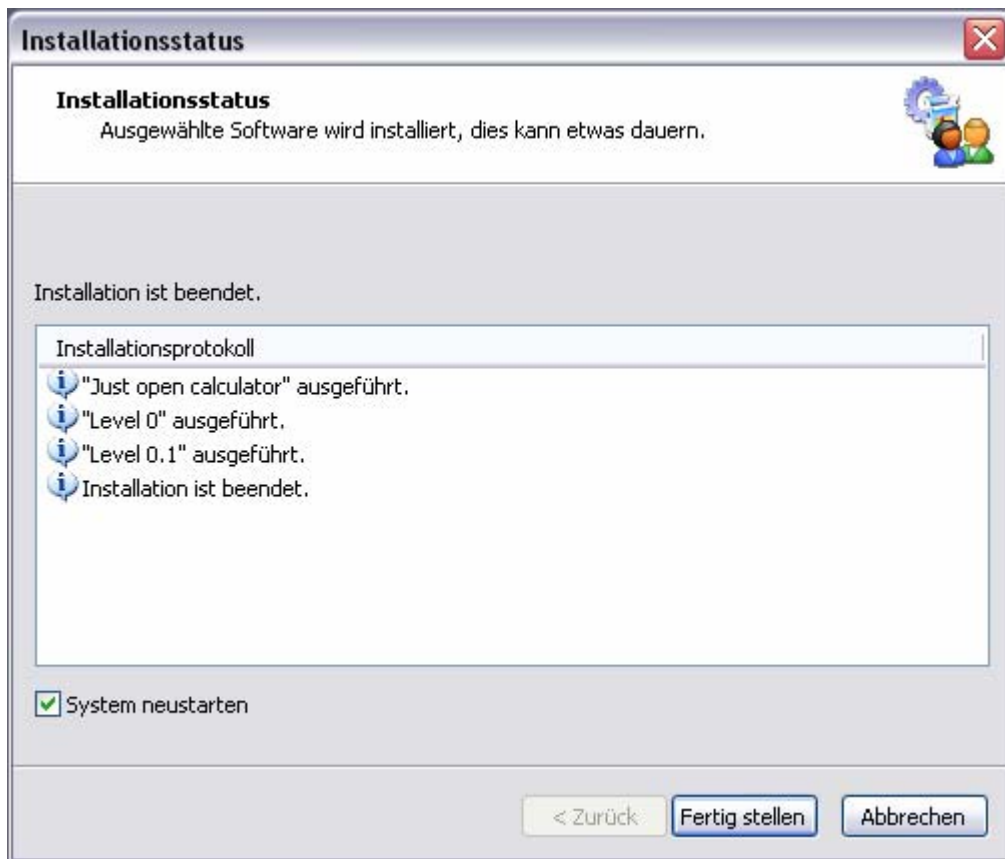


Abbildung 7: Dialog Nr. 3 - Installationsstatus

4 Konfiguration

Dies ist das Hauptkapitel dieser Anleitung. Es beinhaltet die Beschreibung, auf welche Weise WIHU unabhängig von der graphischen Benutzeroberfläche bzw. Benutzereingaben konfiguriert werden kann, um die gewünschte Verhaltensweise zu erzielen. Grundsätzlich bestehen hierzu zwei Möglichkeiten:

- 1.) per Befehlszeilenparameter
- 2.) per Konfigurationsdatei

Beide Möglichkeiten werden parallel zueinander eingesetzt, da sie sich wie bereits weiter oben erwähnt gegenseitig ergänzen. Der Großteil der Einstellungen erfolgt übrigens über eine Konfigurationsdatei, da die Befehlszeile auf Grund der potentiellen Menge der Einstellungen überfordert wäre. In den folgenden Unterkapiteln werden nun die Einstellungsoptionen beider genannten Konfigurationsmöglichkeiten näher beschrieben.

Im Regelfall gilt folgende Hierarchie:

graphische Benutzeroberfläche VOR Befehlszeile VOR Konfigurationsdatei

Merken Sie sich diese Reihenfolge bitte, sie wird nicht mehr explizit erwähnt.

4.1 Befehlszeilenparameter

4.1.1 Allgemeines zum Programmaufruf

Der Aufruf des Programms erfolgt prinzipiell in der Form:

```
WIHU.EXE /Option1 /Option2 /Option3{"<ARGUMENT>"} ...
```

Die Befehlszeilenparameter werden der Reihenfolge nach abgearbeitet; bei widersprüchlichen Angaben – zum Beispiel bei mehrmaliger Angabe ein- und derselben Option mit unterschiedlichen Argumenten – dominiert folglich die letzte über die vorangegangenen. Des Weiteren wird nicht zwischen Klein- und Großschreibung unterschieden. Kommt im Argument eines Parameters mind. ein Leerzeichen vor, muß das Argument in Anführungszeichen eingeschlossen werden, ansonsten ist dies freiwillig.

Beispiel:

```
WIHU /AutoExit=50 /auTOexIT=40 → /auTOexIT=40 dominiert!
```

WIHU kann von unterschiedlichen Orten aus aufgerufen werden, bei unbeaufsichtigten Installationen umfaßt dies insbesondere: Stapelverarbeitungsdatei (.BAT und .CMD), CMDLINES.TXT, SVCPACK.INF, GuiRunOnce, RunOnceEx als auch das CDROM-Laufwerk. Mehr Informationen zu einigen der Startorte finden sich auf den im Kapitel „Verweise und Quellen“ angegebenen Internetseiten. Beachten Sie dabei, daß je nach Aufrufsort noch nicht alle Betriebssystemfunktionen zur Verfügung stehen – z.B. gibt es bei einem Start aus der Datei CMDLINES.TXT heraus Probleme mit der Benutzerkontenkonfiguration.² Falls Sie WIHU dennoch zu so einem frühen Zeitpunkt benutzen möchten, können diese Schwierigkeiten umgangen werden, indem Sie es zeitversetzt zweimal starten und zunächst nur die Programminstallation ausführen und erst zu einem späteren, geeigneteren Zeitpunkt dann die benutzerkontospezifischen Einstellungen vornehmen. Auch die Verwendung des sog. [primären Benutzerkontos](#) zu so einem frühen Installationszeitpunkt ist von dieser Einschränkung betroffen.³ WIHU beherrscht die angeratene zeitliche Trennung dessen beider Haupteinsatzgebiete ohne weiteres.

Eine kurze Übersicht der Befehlszeilenparameter und WIHU-spezifische Umgebungsvariablen erhalten Sie mit dem Befehl (siehe auch Abbildung 8):

```
WIHU /?
```

² siehe auch <http://greenmachine.msfnhosting.com/READING/cmdlines.htm>

³ Der dafür benötigte Dienst „Sekundäre Anmeldung“ ist hier von Windows noch nicht initialisiert.

4.1.2 mögliche Fehlerquellen beim Aufruf von WIHU

- Achten Sie darauf, bei einer unbeaufsichtigten Installation aus einer *Batchdatei* heraus nicht das „start /wait“ vor dem Programmnamen zu vergessen! Ansonsten wird WIHU gestartet, der Computer wartet aber nicht, bis dessen Arbeit, sprich die Programminstallationen, beendet ist. Dies würde aller Wahrscheinlichkeit nach zu Problemen führen, da die eigentlich erst nach Ende von WIHU auszuführenden Prozesse nun parallel hierzu ablaufen und es dadurch eventuell zu Zugriffsverletzungen auf bereits geöffnete Dateien käme.
- Jede einzelne Option beginnt mit einem Leerzeichen und direkt darauffolgenden normalen Schrägstrich „/“. Leerzeichen innerhalb eines Arguments sind nur innerhalb von Anführungszeichen erlaubt. Wertzuweisungen immer mit „=“. Beispiel: `/User="Otto Normal"`
- Falls WIHU gar nicht aufgerufen zu werden scheint, überprüfen Sie die Pfadangabe zum Programm. Gerade bei unbeaufsichtigten Installationen sind falsch gesetzte Pfangaben eine häufige Fehlerursache.
- Falls die letzten Kommandozeilenparameter nicht erkannt zu werden scheinen, haben Sie womöglich die maximal zulässige Länge der DOS-Befehlszeile überschritten. Benutzen Sie dann weniger Optionen an der Befehlszeile oder aber kürzen Sie die Angaben (insb. Pfade). Beachten Sie, daß Variablen an der Kommandozeile gefüllt werden, also aus dem kurzen „%WIHU%“ sehr schnell der längere Ausdruck „C:\Windows\Temp“ werden kann.
- Überprüfen Sie die korrekte Syntax/Rechtschreibung aller von Ihnen verwendeten Parameter. Groß- und Kleinschreibung spielt dabei keine Rolle. Achten Sie auf korrekte Variablenschreibweise wie beispielsweise bei „%USERNAME%“. Sind zudem auch alle Pfade und Dateien, auf die Sie in den Parametern verweisen, an dem erwarteten Ort vorhanden? Die entsprechenden Angaben müssen hier immer relativ zum Speicherort von WIHU erfolgen.

4.1.3 Pfadangaben & (WIHU-)Variablen an der Befehlszeile

Pfadangaben innerhalb von Befehlszeilenargumenten interpretiert WIHU im Standardfall relativ zu dessen Programmdatei WIHU.EXE; bei Dateinamen ohne Pfadangabe in Argumenten müssen die entsprechenden Dateien also im selben Verzeichnis wie WIHU aufzufinden sein. Dieses Verhalten können Sie aber per Konfigurationsdatei ändern.

Beispiel (für den Standardfall):

Angenommen WIHU liegt im Ordner „%systemdrive%\Install\Anwendungen\WIHU“ und Sie rufen es dort folgendermaßen auf:

```
WIHU /INI="..\..\config/wihu_install.ini"
```

In diesem Fall wird im Verzeichnis „%systemdrive%\Install\config“ nach der Datei wihu_install.ini gesucht.

Selbstverständlich können Sie auch die Systemvariablen wie %SystemRoot% oder %UserProfile% ganz normal an der Befehlszeile benutzen, diese werden regulär aufgelöst. Für eine Übersicht geben Sie an der DOS-Eingabeaufforderung den Befehl SET ein. Daneben existiert seitens WIHU noch eine Auswahl spezieller Variablen, welche nur im Rahmen von WIHU Gültigkeit besitzen:

Beispiel: Programmaufruf mit: `c\:>cmd.exe /c C:\Inst\WIHU\wihu.exe`

%curdir%	Verzeichnis, von dem aus WIHU gestartet worden ist im Beispiel: c:\
%wihu%	Verzeichnis, in dem WIHU.EXE liegt im Beispiel: c:\Inst\WIHU\
%srcdrive%	Laufwerk mit den Windowsinstallationsarchiven (im Regelfall das CDRROM-Laufwerk)
%srcpath%	Verzeichnis mit den Windowsinstallationsarchiven im Regelfall: %srcdrive%\I386
%inidir%	Verzeichnis, in dem die Konfigurationsdatei (für Softw.inst.) liegt

Tabelle 2: WIHU-spezifische Umgebungsvariablen

Andere Variablen wie bspw. aus Tabelle 6 sind an der *Befehlszeile* generell nicht gültig!

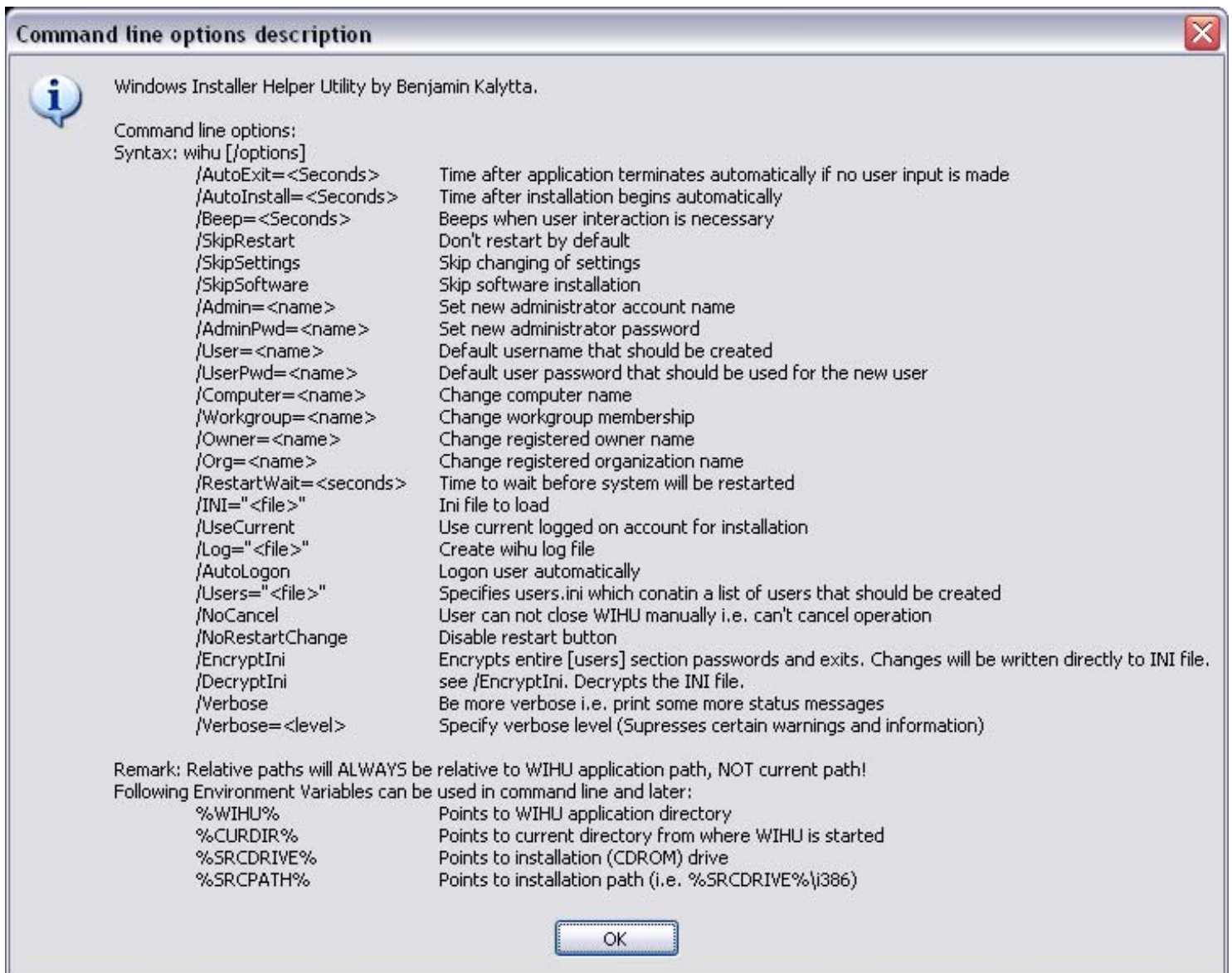


Abbildung 8: Befehlszeilenparameterkurzübersicht nach Aufruf von WIHU /?

4.1.4 Übersicht der Befehlszeilenparameter

4.1.4.1 /? oder /help

Zweck:

Gibt ein kleines Informationsfenster mit den verfügbaren Befehlszeilenparameter am Bildschirm aus (siehe Abbildung 8) und beendet WIHU danach.

4.1.4.2 /AutoExit=<Sekunden>

Zweck:

Zeitspanne in Sekunden, nach der WIHU beendet wird, sofern während dieser Zeit keine Benutzereingaben gemacht werden.

Dies soll verhindern, daß WIHU unter Umständen ohne Erfolg auf Eingaben eines Benutzers wartet, der gar nicht am System sitzt, und dadurch der Gesamtinstallationsprozeß, in den WIHU eingebunden ist, zeitlich bis zu nächsten Benutzereingabe gleichsam einfriert.

Beispiel:

`/AutoExit=900` → wartet 15 Minuten auf Benutzereingaben und beendet WIHU danach, sofern keine gemacht wurden

Anmerkungen & Hinweise:

- a) Wird während der angegebenen Zeit dennoch eine Tastatur- oder Mauseingabe (Mausbewegung zählt nicht) vorgenommen, wird der Countdown abgebrochen und die Funktion für den weiteren Programmverlauf außer Kraft gesetzt, d.h. der Countdown wird nicht neu gestartet. Der Benutzer muß dann entweder in WIHU den Installationsprozeß selbst durch Benutzereingabe starten oder aber das Programm selber abbrechen.
- b) Bei gleichzeitiger Angabe dieser Option in Verbindung mit `/AutoInstall=<Sekunden>` zählt allein die Angabe mit dem niedrigeren Zeitwert; diejenige mit dem höheren Zeitwert wird ignoriert.
- c) Es gibt kein Standardtimeout, d.h. ohne Angabe dieser Option oder von `/AutoInstall=<Sekunden>` wartet WIHU bis zu einer Benutzereingabe.
- d) siehe auch `OsdText=<Text>`

4.1.4.3 /AutoInstall=<Sekunden>

Zweck:

Zeitspanne in Sekunden, nach deren Ablauf WIHU selbständig mit der Abarbeitung der Konfigurationsdatei beginnt.

Diese Funktion erlaubt es einem evtl. am System sitzenden Benutzer, in den Dialogboxen von WIHU die vorgegebenen Einstellungen noch zu ändern bzw. den eigenen Wünschen anzupassen. Sofern keine Eingaben während der angegebenen Zeitspanne stattfinden, beginnt WIHU nach Ablauf der Karenzzeit automatisch mit der Installation gemäß den in der Konfigurationsdatei vorgegebenen Standardwerten.

Beispiel:

`/AutoInstall=600` → wartet 10 Minuten auf Benutzereingaben und beginnt danach, sofern keine vorgenommen wurden, selbständig mit der Abarbeitung der Konfigurationsdatei

Anmerkungen & Hinweise:

- a) Wird während der angegebenen Zeit dennoch eine Tastatur- oder Mauseingabe (Mausbewegung zählt nicht) vorgenommen, wird der Countdown abgebrochen und die Funktion für den weiteren Programmverlauf außer Kraft gesetzt, d.h. der Countdown wird nicht neu gestartet, der Benutzer kann alle Einstellungen in WIHU ganz regulär ändern und muß im Anschluß die eigentliche Installation selbst durch einen Klick einleiten.
- b) Die Option bewirkt in jedem Fall, daß sich WIHU nach Abschluß der Arbeit selbständig beendet, was ohne Angabe dieses Parameters erst durch einen bestätigenden Klick des Benutzers auf die Schaltfläche „Fertigstellen“ (siehe Abbildung 7) der Fall wäre.
- c) Die Angabe `/AutoInstall=0` bewirkt, daß WIHU sofort nach dessen Aufruf mit der Arbeit gemäß den Einstellungen in der Konfigurationsdatei beginnt (Countdown der Länge Null). Dies bezieht sowohl den benutzer- also auch softwarespezifischen Teil mit ein. Dem Benutzer am System wird somit keine Möglichkeit eingeräumt, die vorgegebenen Werte zu ändern, es läuft alles vollautomatisch ab. Ein Abbruch von WIHU ist aber weiterhin möglich; gegen letzteres hilft die Option `/NoRestartChange` und `/SkipRestart`

- d) Bei gleichzeitiger Angabe der Option in Verbindung mit `/AutoExit=<Sekunden>` zählt allein die Angabe mit dem niedrigeren Zeitwert; diejenige mit dem höheren Zeitwert wird ignoriert.
- e) Es gibt kein Standardtimeout, d.h. ohne Angabe dieser Option oder von `/AutoExit=<Sekunden>` wartet WIHU bis zu einer Benutzereingabe.
- f) siehe auch `OsdText=<Text>`

4.1.4.4 `/beep=<Sekunden>`

Zweck:

Gibt ein akustisches Signal im angegebenen Sekundenintervall über die Systemlautsprecher aus, solange WIHU bereit für Benutzereingaben ist.

So muß ein im Raum anwesender Benutzer während einer unbeaufsichtigten Installation nicht notwendigerweise andauernd auf den Bildschirm achten, um nicht den richtigen Moment für die Benutzereingaben zu verpassen, sondern kann sich von WIHU hierüber akustisch in Kenntnis setzen lassen.

Beispiel:

```
/beep=60
```

Anmerkungen & Hinweise:

- a) Das Signal beginnt, sobald WIHU gestartet worden ist und hört mit Beginn seines Arbeitsprozesses (siehe auch Abbildung 7) automatisch wieder auf.
- b) Sobald der Benutzer während der Wartezeit auf Eingaben Änderungen in WIHU vornimmt (Mausbewegungen allein zählen nicht hierzu), verstummt das Signal.
- c) Sinnvoll kombiniert werden kann diese Option mit `/AutoExit=<Sekunden>` oder `/AutoInstall=<Sekunden>`.
- d) Zur optischen Anzeige der verbleibenden Countdownzeit siehe `OsdText=<Text>`

4.1.4.5 /RestartWait=<Sekunden>

Zweck:

Nachdem WIHU dessen Arbeit abgeschlossen hat und beendet wird (egal ob durch Benutzereingabe oder automatisch), wird der Rechner durch Angabe dieser Option neu gestartet. Vor dem Neustart erfolgt ein Countdown, welcher mittels einer Meldung angezeigt wird:

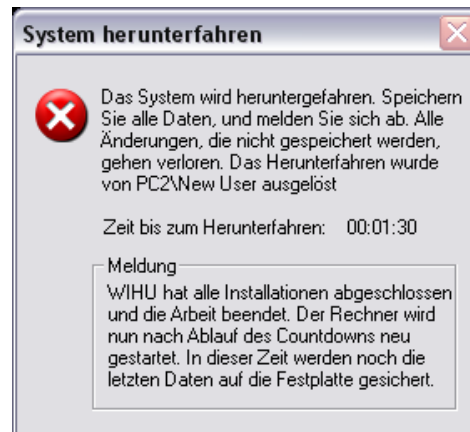


Abbildung 9: Meldungsbox beim Herunterfahren des Systems

Der Countdown soll dazu dienen, dem System ausreichend Zeit zu geben, die letzten Daten nach den vorangegangenen Installationen vor dem Neustart noch in Ruhe auf die Festplatte schreiben zu können.

Beispiel:

```
/RestartWait=90
```

Anmerkungen & Hinweise:

- Der Countdown beginnt erst nach Beendigung von WIHU. Insofern haben Optionen wie `/AutoInstall=<Sekunden>` keinen Einfluß auf den Countdown.
- Die Option `/SkipRestart` aber setzt den Countdown außer Kraft, der Neustart wird nicht durchgeführt.

4.1.4.6 /NoRestartChange

Zweck:

Verhindert die Änderung der Voreinstellung für den Systemneustart durch den Benutzer

So kann ein Neustart nach der Installation entweder erzwungen oder verboten werden.

Anmerkungen & Hinweise:

a) Diese Option inaktiviert das Auswahlkästchen aus Abbildung 7.

b) siehe auch /RestartWait=<Sekunden>

4.1.4.7 /NoCancel

Zweck:

WIHU kann nicht mehr durch einen Klick auf das Schließen-Kreuz oder auf die Schaltfläche „Abbrechen“ beendet werden. Der Benutzer muß WIHU bis zum regulären Ende ausführen, um es ordnungsgemäß zu beenden.

4.1.4.8 /SkipRestart

Zweck:

Übergeht den Neustart des Rechners nach Beendigung von WIHU.

Anmerkungen & Hinweise:

a) Bei Angabe dieser Option wird /RestartWait=<Sekunden> ignoriert.

b) siehe auch Auswahlkästchen in Abbildung 7: Dialog Nr. 3 - Installationsstatus

c) Die Angabe dieser Option beeinflusst nur die Vorauswahl, letztlich verbieten können Sie einen Neustart in Verbindung mit der Option /NoRestartChange.

4.1.4.9 /SkipSettings

Zweck:

Überspringt die benutzerspezifischen Einstellungen (siehe Abbildung 2).

Folglich wird nur der Teil mit den Programminstallationen ausgeführt (siehe Abbildung 6), Benutzerkonten werden weder erstellt, verändert noch gelöscht.

Auf diese Weise kann die Arbeit von WIHU aufgeteilt werden: Bei einem Aufruf konfigurieren Sie beispielsweise allein alle benutzerrelevanten Einstellungen nach Ihren Wünschen und zu einem davon unabhängigen anderen Zeitpunkt führen Sie ausschließlich die Programminstallationen durch.

Anmerkungen & Hinweise:

- a) Als Basis für die Softwareinstallationen dienen die aktuell gültigen Benutzereinstellungen, z.B. aktuell angemeldetes Konto mit dessen spezifischen Einstellungen wie Systemverzeichnissen oder Rechten.
- b) Der gegenteilige Schalter lautet `/SkipSoftware`
- c) nach wie vor geladen und verarbeitet wird der Abschnitt [Environment]
- d) Diese Option ist gerade in Verbindung mit den in Kapitel 4.1.1 dargestellten Problemen hilfreich.

4.1.4.10 /SkipSoftware

Überspringt die Softwareinstallation (siehe Abbildung 6). Folglich können nur die Benutzerkonten samt den damit zusammenhängenden Einstellungen (Variablen, Systemordner, etc.) konfiguriert werden.

Auf diese Weise kann die Arbeit von WIHU aufgeteilt werden: Bei einem Aufruf konfigurieren Sie beispielsweise allein alle benutzerrelevanten Einstellungen nach Ihren Wünschen und zu einem davon unabhängigen anderen Zeitpunkt führen Sie ausschließlich die Programminstallationen durch.

Anmerkungen & Hinweise:

- a) Der gegenteilige Schalter lautet `/SkipSettings`
- b) Diese Option ist gerade in Verbindung mit den in Kapitel 4.1.1 dargestellten Problemen hilfreich.

4.1.4.11 /Admin=<Name>

Zweck:

Ändert den Namen des nativen Administratorkontos.

Beispiel:

```
/Admin="PC Guru"
```

Anmerkungen & Hinweise:

- a) Ändert ausschließlich den Namen des nativen Administratorkontos (im deutschen Windows XP lautet der Name „Administrator“) und nicht die systeminterne SID, d.h. reine Umbenennung, keine Löschung oder Erstellung eines Kontos! Alle anderen evtl. schon vorhandenen Administratorkonten bleiben zudem unberührt.
- b) Der Name wird 1:1 übernommen, sprich Groß- und Kleinschreibung spielt hier keine Rolle. Falls in dem Namen Leerzeichen vorkommen, muß er in Anführungsstriche gesetzt werden (siehe Beispiel).

Unzulässige Zeichen sind: / , \ [] : | < > + = ; ? * und alles kleiner ASCII-Code 32

- c) Die von WIHU maximal akzeptierte Namenslänge beträgt 100 Zeichen.

4.1.4.12 /AdminPwd=<Paßwort>

Zweck:

Weist dem nativen, d.h. systemeigenen und standardmäßigen, Administratorkonto das gewählte Paßwort zu.

Beispiel:

```
/AdminPwd="9äQ4k ehÜj"
```

Anmerkungen & Hinweise:

- a) Wird die Option nicht angegeben, wird das aktuell gültige Paßwort nicht geändert.
- b) Mit /AdminPwd="" kann ein evtl. gesetztes Paßwort gelöscht werden.
- c) Diese Option kann auch ohne /Admin=<Name> verwendet werden. Für diesen Fall wird einfach das Paßwort des aktuell vorhandenen Standardadministratorkontos entsprechend abgeändert.

- d) Falls durch die Option `/Admin=<Name>` das Standardadministratorkonto geändert wurde, bezieht sich das Paßwort auf dieses neue Administratorkonto.
- e) In den Windows-Gruppenrichtlinien (`gpedit.msc`) kann festgelegt worden sein, daß alle neuen Benutzerkontenpaßwörter bestimmte Mindestsicherheitsanforderungen (alphanumerisch, Mindestlänge, etc.) erfüllen müssen. WIHU muß sich ebenfalls an diese Einschränkungen halten, übernimmt aber keine dementsprechende Prüfung des Kennwortes. Achten Sie daher auf die Übereinstimmung des von Ihnen gewählten Paßwortes mit den diesbezüglichen Sicherheitsanforderungen des Systems.
- f) Das Paßwort wird 1:1 übernommen, sprich Groß- und Kleinschreibung spielt hier eine Rolle!

Unzulässige Zeichen sind: `/ , \ [] : | < > + = ; ? *` und alles kleiner ASCII-Code 32

- g) maximal zulässige Länge beträgt 100 Zeichen.

4.1.4.13 `/User=<Name>`

Zweck:

Legt ein neues Benutzerkonto unter dem angegebenen Namen an.

Dieses dient WIHU als sog. „[primäres Benutzerkonto](#)“.

Beispiel:

```
/User="Otto Normal"
```

Anmerkungen & Hinweise:

- a) Das Konto mit dem entsprechenden Namen sollte noch nicht existieren!
- b) Es kann nur ein primäres Benutzerkonto geben, insofern kann die Option auch nur einmal an der Befehlszeile angegeben werden.
- c) maximal zulässige Länge beträgt 20 Zeichen
- d) dominiert `default=<Indexnummer>`
- e) entspricht folgendem Eintrag im Abschnitt `[Users]` der Konfigurationsdatei, hat davor aber Vorrang:

```
[users]
default=0
user.0=Otto Normal
```

- f) Verwechseln Sie diese Option bitte nicht mit der ähnlich klingenden Option
/Users=<Datei>

4.1.4.14 /UserPw=<Paßwort>

Zweck:

Weist dem per /User=<Name> angelegten Benutzerkonto das gewählte Paßwort zu.

Beispiel:

Anmerkungen & Hinweise:

- a) In den Windows-Gruppenrichtlinien (gpedit.msc) kann festgelegt worden sein, daß alle neuen Benutzerkontenpaßwörter bestimmte Mindestsicherheitsanforderungen (alphanumerisch, Mindestlänge, etc.) erfüllen müssen. WIHU muß sich ebenfalls an diese Einschränkungen halten, übernimmt aber keine dementsprechende Prüfung des Kennwortes. Achten Sie daher auf die Übereinstimmung des von Ihnen gewählten Paßwortes mit den diesbezüglichen Sicherheitsanforderungen des Systems.
- b) Das Paßwort wird 1:1 übernommen, sprich Groß- und Kleinschreibung spielt hier eine Rolle! Unzulässige Zeichen sind: /, \ [] : | < > + = ; ? * und alles kleiner ASCII-Code 32
- c) entspricht folgendem Eintrag im Abschnitt [Users] der Konfigurationsdatei, hat davor aber Vorrang:

```
password.x=<Paßwort>
; wobei sich .x hier auf den per default=x festgelegten Wert bezieht
```

- d) maximal zulässige Länge beträgt 100 Zeichen

4.1.4.15 /AutoLogon

Zweck:

Meldet das [primäre Benutzerkonto](#) beim Systemstart automatisch an, ohne daß Sie den Anmeldebildschirm von Windows sehen oder ein Paßwort eingeben müßten.

Anmerkungen & Hinweise:

- a) Diese Option betrifft nur das [primäre Benutzerkonto](#).
- b) Das automatische Anmelden erfolgt solange, bis sie die Funktion in Windows deaktivieren.

4.1.4.16 /UseCurrent

Zweck:

Übergeht die Erstellung des [primären Benutzerkontos](#) und veranlaßt WIHU somit, alle Änderungen im Kontext des aktuell angemeldeten Benutzerkontos durchzuführen.

Anmerkungen & Hinweise:

- a) entspricht Auswahlkästchen „Angemeldeten Benutzer verwenden“ in Abbildung 1
- b) Achten Sie darauf, daß das aktuell gültige Benutzerkonto zur Ausführungszeit von WIHU prinzipiell mit Administratorrechten ausgestattet sein sollte, da ansonsten im Regelfall sämtliche der vorzunehmenden Änderungen fehlschlagen werden.
- c) Es wird allgemein empfohlen, von der Funktion des primären Benutzerkontos zur Softwareinstallation Gebrauch zu machen bzw. die obige Option nicht zu verwenden. Eine Ausnahme besteht aber darin, wenn Sie WIHU noch zu einem sehr frühen Zeitpunkt der Windowsinstallation aufrufen, z.B. aus der Datei CMDLINES.TXT heraus. Siehe hierzu auch Kapitel „Allgemeines zum Programmaufruf“.

4.1.4.17 /Computer=<Name>

Zweck:

Legt den Computernamen fest.

Anmerkungen & Hinweise:

- a) Anzeigen lassen können Sie die Informationen, wenn Sie unter Windows die Tastenkombination [WIN]+[PAUSE] drücken und dort den Reiter „Computername“ auswählen
- b) Zulässige Zeichen sind (Groß- und Kleinschreibung natürlich auch erlaubt):
abcdefghijklmnopqrstuvwxyz1234567890!@#\$%^&') (. - _ { } ~
- c) maximal zulässige Länge beträgt 15 Zeichen
- d) siehe auch Konfigurationsdateianweisung Computer=<Name>
- e) siehe auch Feld „Computername“ in Abbildung 1

4.1.4.18 /Workgroup=<Name>

Zweck:

Legt den Namen der Arbeitsgruppe fest, zu der der Computer gehört.

Anmerkungen & Hinweise:

- a) Anzeigen lassen können Sie die Informationen, wenn Sie unter Windows die Tastenkombination [WIN]+[PAUSE] drücken und dort den Reiter „Computername“ auswählen
- b) Zulässige Zeichen sind (Groß- und Kleinschreibung natürlich auch erlaubt):
abcdefghijklmnopqrstuvwxyz1234567890!@#\$%^&') (. - _ { } ~
- c) maximal zulässige Länge beträgt 15 Zeichen
- d) siehe auch Konfigurationsdateianweisung Workgroup=<Name>
- e) siehe auch Feld „Arbeitsgruppe“ in Abbildung 1

4.1.4.19 /Owner=<Name>

Zweck:

Legt den Benutzernamen fest, unter dem Windows registriert ist.

Beispiel:

```
/Owner="Prof. Dr. Otto Normal"
```

Anmerkungen & Hinweise:

- a) Diese Angabe ist quasi rein kosmetisch und hat nichts mit Benutzerkonten, Systemrechten oder Softwareinstallationen zu tun.
- b) Anzeigen lassen können Sie die Informationen, wenn Sie unter Windows die Tastenkombination [WIN] + [PAUSE] drücken
- c) siehe auch Konfigurationsdateianweisung Owner=<Name>
- d) siehe auch Feld „Eigentüername“ in Abbildung 1

4.1.4.20 /Org=<Name>

Zweck:

Legt das Unternehmen/die Organisation fest, für die das Windowssystem lizenziert ist.

Beispiel:

```
/Org="Universität Lechstadt"
```

Anmerkungen & Hinweise:

- a) Diese Angabe ist quasi rein kosmetisch und hat nichts mit Benutzerkonten, Systemrechten oder Softwareinstallationen zu tun.
- b) Anzeigen lassen können Sie die Informationen, wenn Sie unter Windows die Tastenkombination [WIN] + [PAUSE] drücken
- c) siehe auch Konfigurationsdateianweisung Org=<Name>
- d) siehe auch Feld „Firmenname“ in Abbildung 1

4.1.4.21 /INI=<Datei>

Zweck:

WIHU lädt all seine Einstellungen aus der angegebenen (Konfigurations-)Datei.

Auf diese Weise können Sie beispielsweise mehrere unterschiedliche Konfigurationen für verschiedene Systeme, auf denen WIHU eingesetzt werden soll, vorhalten und sich schnell und gemütlich jeweils die passende einfach durch Angabe der zu ladenden Datei heraussuchen.

Beispiel:

```
/INI="%systemroot%\Konfig\Meine WIHU-Konfiguration #7.ini"
```

Anmerkungen & Hinweise:

- a) entspricht der „Laden“-Schaltfläche der graphischen Benutzeroberfläche (siehe Abbildung 6).
- b) Ohne Angabe der Option sucht WIHU standardmäßig nach einer Datei namens `INSTALL.INI` im selben Verzeichnis, in dem sich `WIHU.EXE` befindet.
- c) Findet WIHU dort keine `INSTALL.INI` oder handelt es sich bei der zu ladenden Datei um eine ungültige Konfigurationsdatei, wird diese einfach ignoriert bzw. die darin enthaltenen Einstellungen eingelesen, soweit diese erkennbar sind. Es erscheint auch keine Fehlermeldung, um die unbeaufsichtigte Installation durch das dann nötige Bestätigen der Fehlermeldung mit „OK“ nicht zwangspausieren zu lassen.
- d) Umgebungsvariablen an der Befehlszeile werden regulär aufgelöst. Lediglich die [in der WIHU-Konfigurationsdatei definierten Variablen](#) stehen zu diesem Zeitpunkt noch nicht zur Verfügung.
- e) Die Pfadangabe zur Konfigurationsdatei wird in der [Variablen %INIDIR%](#) abgelegt.
- f) Die Dateinamenserweiterung spielt keine Rolle, „INI“ wird aber des Standards wegen empfohlen.
- g) Mehr zum Aufbau und zur Syntax der Konfigurationsdatei im Kapitel 4.2
- h) siehe auch `/Users=<Datei>`

4.1.4.22 /Users=<Datei>

Zweck:

WIHU lädt seine benutzerkontospezifischen Einstellungen ausschließlich aus der angegebenen (Konfigurations-)Datei.

Auf diese Weise können Sie beispielsweise mehrere unterschiedliche Benutzerkontenkonfigurationen für verschiedene Systeme, auf denen WIHU eingesetzt werden soll, vorhalten und sich schnell und gemütlich jeweils die passende einfach durch Angabe der zu ladenden Datei heraussuchen.

Beispiele:

```
/Users="Meine WIHU-Benutzerkontenliste #2.ini"
```

```
/Users="%systemroot%\Konfig\wihu_users.ini"
```

Anmerkungen & Hinweise:

- a) entspricht der „Laden“-Schaltfläche der graphischen Benutzeroberfläche (siehe Abbildung 2).
- b) Ohne Angabe der Option sucht WIHU standardmäßig nach einer Datei namens `INSTALL.INI` im selben Verzeichnis, in dem sich `WIHU.EXE` befindet. Innerhalb der Datei wird nach einem Abschnitt mit der Bezeichnung `[users]` gesucht, welcher die benutzerkontospezifischen Angaben enthalten soll.
- c) Mit Angabe dieser Option steht Ihnen die Möglichkeit offen, die Konfiguration von WIHU auf zwei Dateien aufzuteilen bzw. bei Bedarf den benutzerkontospezifischen Teil aus der eigentlichen Hauptkonfigurationsdatei auszulagern.
- d) Handelt es sich bei der zu ladenden Datei um eine ungültige Konfigurationsdatei, wird diese einfach ignoriert bzw. die Einstellungen soweit wie erkennbar ausgewertet. Es erscheint auch keine Fehlermeldung, um die unbeaufsichtigte Installation durch das dann nötige Bestätigen der Fehlermeldung mit „OK“ nicht zwangspausieren zu lassen.
- e) Umgebungsvariablen an der Befehlszeile werden regulär aufgelöst. Lediglich die [in der WIHU-Konfigurationsdatei definierten Variablen](#) stehen zu diesem Zeitpunkt noch nicht zur Verfügung.
- f) Die Dateinamenserweiterung spielt keine Rolle, „INI“ wird aber des Standards wegen empfohlen.

- g) Mehr zum Aufbau und zur Syntax der Konfigurationsdatei im Kapitel 4.2
- h) siehe auch `/INI=<Datei>`
- i) Bei gleichzeitiger Angabe der Option und Verwendung des [users]-Abschnitts in der allgemeinen Konfigurationsdatei werden beide Möglichkeiten berücksichtigt; als erstes wird die jedoch Befehlszeile abgearbeitet.
- j) Verwechseln Sie diese Option bitte nicht mit der ähnlich klingenden Option `/User=<Name>`

4.1.4.23 /EncryptINI

Zweck:

Verschlüsselt alle Paßwörter, welche im Abschnitt [Users] der Konfigurationsdatei vorkommen und schreibt die Änderungen direkt in die Datei.

So kann verhindert werden, daß die Klartextkennwörter einfach abzulesen sind.

Beispiel:

vor Anwendung des Befehls steht in der der Konfigurationsdatei:

```
password.0=Dies ist ein Testpaßwort
```

nach Anwendung des Befehls steht anstatt dessen in der Konfigurationsdatei:

```
password.0=3F44562D3348403B601B09727A01...  
crypted.0=50
```

Anmerkungen & Hinweise:

- a) Ohne Angabe von `/INI=<Datei>` oder `/Users=<Datei>` sucht WIHU standardmäßig nach einer Datei namens `INSTALL.INI` im selben Verzeichnis, in dem sich `WIHU.EXE` befindet. Ansonsten wird nach der Datei gesucht, die Sie per Befehlszeilenparameter angegeben haben, wobei die Option `/Users=<Datei>` Vorrang hat.
- b) Die Änderungen werden direkt in die Konfigurationsdatei geschrieben, achten Sie daher darauf, daß WIHU die nötigen Schreibrechte besitzt (CDROM, etc.).
- c) Nach der Ausführung des Befehls wird WIHU beendet.

- d) Der Befehl schreibt für jedes verschlüsselte Paßwort eine neue Zeile `rypted.x=<Wert>` in die Konfigurationsdatei. Bitte ändern oder löschen Sie diese Zeile nicht! Ohne den richtigen Wert schlägt die korrekte Übernahme des Kennworts fehl und sie haben dann aller Regel nach auf Grund von Paßwortproblemen keinen Zugriff mehr auf das jeweilige Konto.
- e) Beachten Sie, daß es sich um keine starke kryptographische Verschlüsselung handelt! Es soll lediglich das Auslesen der Paßwörter in Klartextform verhindert werden.

4.1.4.24 /DecryptINI

Zweck:

Verwandelt die mittels `/EncryptINI` verschlüsselten Paßwörter wieder in Klartextform und schreibt die Änderungen direkt in die Konfigurationsdatei.

Beispiel:

vor Anwendung des Befehls steht in der der Konfigurationsdatei:

```
password.0=3F44562D3348403B601B09727A01...
rypted.0=50
```

nach Anwendung des Befehls steht anstatt dessen in der Konfigurationsdatei:

```
password.0=Dies ist ein Testpaßwort
```

Anmerkungen & Hinweise:

- a) Ohne Angabe von `/INI=<Datei>` oder `/Users=<Datei>` sucht WIHU standardmäßig nach einer Datei namens `INSTALL.INI` im selben Verzeichnis, in dem sich `WIHU.EXE` befindet. Ansonsten wird nach der Datei gesucht, die Sie per Befehlszeilenparameter angegeben haben, wobei die Option `/Users=<Datei>` Vorrang hat.
- b) Die Änderungen werden direkt in die Konfigurationsdatei geschrieben, achten Sie daher darauf, daß WIHU die nötigen Schreibrechte besitzt.
- c) Nach der Ausführung des Befehls wird WIHU beendet.
- d) Der Befehl entfernt zudem für jedes verschlüsselte Paßwort die dazugehörige `rypted.x=<Wert>`-Zeile in der Konfigurationsdatei. Diese Anweisung wird bei Klartextkennwörtern nicht mehr benötigt.

4.1.4.25 /Log=<Datei>

Zweck:

Schreibt die Hinweise, die im [Protokollfenster](#) von WIHU während des Arbeitsprozesses ausgegeben werden, auch in eine Datei.

Beispiele:

```
/Log=D:\Logs\WIHU.log
```

```
/Log=%systemdrive%\WIHU-Protokoll.txt
```

Anmerkungen & Hinweise:

- a) Das Dateiformat entspricht dem einer simplen Textdatei, kann also mit jedem einfachen Editor problemlos geöffnet werden.
- b) Umgebungsvariablen an der Befehlszeile werden regulär aufgelöst. Lediglich die [in der WIHU-Konfigurationsdatei definierten Variablen](#) stehen zu diesem Zeitpunkt noch nicht zur Verfügung.

4.1.4.26 /verbose=[0|1|2|3]

Zweck:

Gibt im [Protokollfenster](#) WIHUs detailliertere Meldungen aus.

Anwendung:

Wert	Bedeutung
0	Meldungen aus Modus 1 sowie Modus 2 nicht ausgeben
1	bestimmte Statusmeldungen ausgeben
2	Text von <code>description.x.y.z=<Beschreibungstext></code> ausgeben
3	Alle Statusmeldungen ausgeben (Modus 1+2)

Tabelle 3: Auswahl der im Protokollfenster anzuzeigenden Meldungen

Anmerkungen & Hinweise:

- a) Gedacht insbesondere für die Fehlersuche und für die Meldung von Fehlern an den Programmentwickler
- b) Die Meldungen können natürlich auch per /Log=<Datei> gespeichert werden
- c) Ohne Wertangabe wird standardmäßig Kennziffer 3 verwendet

4.2 Konfigurationsdatei

4.2.1 Allgemeines zur dateibasierten Konfiguration

Diese Anleitung beschreibt im Folgenden den Aufbau einer WIHU-Konfigurationsdatei. Hierbei handelt es sich wie meist üblich um eine ganz normale Textdatei gefüllt mit Anweisungen. Zur Bearbeitung bedarf es daher nur eines simplen Texteditors; in Form des Programms „Notepad“ ist ein solcher schon auf jedem Windowssystem standardmäßig vorhanden.

Die prinzipielle Syntax der Konfigurationsdatei entspricht denen von INI-Dateien; dadurch wird im Gegensatz zu XML oder anderen Notationen eine möglichst breite Verständlichkeit durch relativ geringe Codekomplexität gewährleistet.

INI-Dateien sind generell in logische Abschnitte (manchmal auch als „Sektionen“ bezeichnet) unterteilt, innerhalb derer die jeweils passenden Anweisungen aufgeführt sind.

Beispiel für eine allgemeine INI-Datei:

```
[Abschnitt1]
```

```
Anweisung1
```

```
Anweisung2
```

```
...
```

```
[Abschnitt2]
```

```
; Kommentar 1
```

```
Anweisung3
```

```
...
```

Groß- und Kleinschreibung der Anweisungen ist für WIHU nicht von Bedeutung. Kommentare können immer am Beginn einer Zeile mit einem Semikolon (;) eingeleitet werden.

TIP:

Im Installationspaket von WIHU befindet sich eine Beispielkonfigurationsdatei namens „INSTALL.INI“. Öffnen Sie diese parallel zum Lesen dieser Anleitung und verdeutlichen Sie sich in Verbindung mit den Erklärungen von hier die Funktionsweise.

4.2.2 Numerierungssystem von Anweisungen

Wie aus Abbildung 6 ersichtlich, erlaubt WIHU dem Benutzer, eine Baumstruktur vorzugeben und damit die Abarbeitungsreihenfolge innerhalb eines Abschnitts vorzugeben. Die Realisation ist denkbar einfach und erfolgt schlicht mit Hilfe eines numerischen Indexes in der Form `.x.y.z`, direkt angehängt an die entsprechende Anweisung.

Beispiel:

Folgende Einträge in der Konfigurationsdatei ...

```
Anweisung.0="Hauptebene, Punkt 1/2"
```

```
Anweisung.1="Hauptebene, Punkt 2/2"
```

```
Anweisung.0.0.0="2. Unterebene von Anweisung.0, Punkt 1/2"
```

```
Anweisung.0.0.1="2. Unterebene von Anweisung.0, Punkt 2/2"
```

```
Anweisung.0.0="1. Unterebene von Anweisung.0, Punkt 1/1"
```

ergeben folgende Baumstruktur:

```
Hauptebene, Punkt 1/2
```

```
  1. Unterebene von Anweisung.0, Punkt 1/1
```

```
    2. Unterebene von Anweisung.0, Punkt 1/2
```

```
    2. Unterebene von Anweisung.0, Punkt 2/2
```

```
Hauptebene, Punkt 2/2
```

Anmerkungen & Hinweise:

- Jede Unterebene kann (theoretisch) bis zu maximal 65.535 Einträge enthalten. Dabei darf die Länge eines Schlüssels „Anweisung.a.b.c.d.e.f.g.h...“ aber die Länge von 255 Zeichen nicht überschreiten. In dieser Anleitung wird exemplarisch mit drei Ebenen `.x.y.z` gearbeitet. Bei den Anweisungen, wo Unterebenen nicht sinnvoll bzw. möglich sind, ist dies entsprechend vermerkt.
- Die Numerierung muß immer bei 0 beginnen – auch auf den Unterebenen – und darf keine Lücken aufweisen. Bei einer Zahlenlücke würde die Abarbeitung für die entsprechende Sektion abbrechen und im nächsten Abschnitt bei 0 wieder anfangen.
- Die schriftliche Reihenfolge in der Konfigurationsdatei ist egal, lediglich die logische Indexabfolge gibt die tatsächliche Abarbeitungsreihenfolge an.

4.2.3 Abschnitt [<beliebige Bezeichnung>]

Zweck:

Einleitung einer beliebig zu benennenden Gruppe von Anweisungen zur Installation von Software.

Hiermit legen Sie die individuelle sowie grundsätzliche Struktur der Installation fest und bestimmen dadurch das Aussehen der Baumstruktur wie in Abbildung 6 dargestellt.

Beispiele:

[Sun Java VM 1.5]

[DirectX 9]

[Microsoft Office 2003]

[Malprogramm für meine Kinder]

[Kaufmännische Software]

[Hotfixes & Patches für XP]

Anmerkungen & Hinweise:

- a) Die von Ihnen gewählten Sektionsbezeichnungen stellen die oberste Ebene in der Baumstruktur dar. Sie sind in der Wahl der Bezeichnung vollkommen frei, so können Sie beispielsweise ganz konkrete Programmnamen als Bezeichnung verwenden („Sun Java VM 1.5“) wie auch Oberbegriffe („Hotfixes & Patches für XP“).
- b) Den einzelnen Abschnitten müssen noch Anweisungen hinzugefügt werden (siehe ab Kapitel 4.2.3.1), welche WIHU mitteilen, was genau zu tun ist. So bestimmen Sie die Feinstruktur der Installation, d.h. die Unterebenen in der Baumstruktur.
- c) Die Abarbeitungsreihenfolge der Abschnitte untereinander entspricht der Reihenfolge der Niederschrift in der Konfigurationsdatei. Innerhalb der einzelnen Abschnitte bestimmt allein ein Numerierungssystem von Anweisungen die Abarbeitungsreihenfolge.

4.2.3.1 `command.x.y.z=<Befehl>`

Zweck:

Befehl, der ausgeführt werden muß, um eine Software zu installieren. Der Befehls selbst wird im WIHU-Fenster nicht angezeigt.

Beispiele:

```
command.0=..\DirectX9b\DX9NTopk.exe
```

```
command.1=Wmp9\MPSetupXP.exe /Q:A /R:N
```

```
command.2=NetFrmwrk\netfx.msi /qb
```

```
command.2.0=%systemroot%\regedit.exe
```

```
command.2.0.0=cmd.exe /c @echo Hallo Welt!
```

Anmerkungen & Hinweise:

- a) Es handelt sich dabei um eine ganz normale Befehlszeile samt Parametern, etc. Als Befehl ist alles zugelassen, was unter Windows als lauffähiges Programm ausgeführt werden kann, also zum Beispiel MSI-Dateien (ohne Angabe von `msiexec /i`) wie auch simple DOS-Befehle oder Stapelverarbeitungsdateien. Die beiden letzteren sollten aber immer mittels `cmd.exe /c <DOS-Befehl>` eingeleitet werden, um Probleme zu vermeiden.
- b) Die Verwendung von Systemvariablen wie auch der WIHU-spezifische Umgebungsvariablen ist möglich. Außerdem zugelassen sind alle Umgebungsvariablen aus dem Abschnitt [Environment], einschließlich der Sondervariablen.
- c) Eine zusätzliche Angabe von „`start /wait`“ zum Aufruf des Installationsbefehls ist nicht erforderlich, auch nicht empfohlen! WIHU wartet selbständig, bis die entsprechende Installation abgeschlossen ist und fährt automatisch erst im Anschluß daran mit dem nächsten Kommando fort. „`start /wait`“ würde zudem ein (unerwünschtes) DOS-Fenster öffnen.
- d) Mit der Sonderanweisung `command.x.y.z=*` wird das entsprechende Element nicht mehr ausführbar, sondern zu einer reinen Textüberschrift in Fettschrift umfunktioniert. So können Sie beispielsweise kurze textuelle Erklärungen in die Baumstruktur einfügen. Siehe auch

4.2.3.2 workdir.x.y.z=<Pfadangabe>

Zweck:

Legt für `command.x.y.z=<Befehl>` und den dazu gehörigen Unterebenen das zu verwendende Arbeitsverzeichnis fest.

Beispiele:

```
command.0=cmd.exe
workdir.0=.
description.0=DOS-Fenster, Prompt relativ zum WIHU.EXE-Verzeichnis
; hier könnte man die Anweisung auch einfach weglassen [Standardfall]
```

```
command.1=cmd.exe
workdir.1=C:\
description.1=DOS-Fenster, Prompt fest im Verzeichnis C:\
```

```
command.2=cmd.exe
workdir.2=%ThisProfile%
description.2=DOS-Fenster, öffnet sich im jeweiligen Profilpfad
```

Anmerkungen & Hinweise:

- a) Wenn Sie in der Anweisung selbst eine relative Pfadangabe verwenden, bezieht sich diese auf das Verzeichnis, in dem sich WIHU.EXE befindet.
- b) Die Verwendung von Systemvariablen wie auch der WIHU-spezifische Umgebungsvariablen ist möglich. Außerdem zugelassen sind alle Umgebungsvariablen aus dem Abschnitt [Environment].

4.2.3.3 description.x.y.z=<Beschreibungstext>

Zweck:

Fügt dem per `command.x.y.z=<Befehl>` angegebenen Befehl einen erklärenden Beschreibungstext hinzu, der im WIHU-Fenster (siehe Abbildung 6) angezeigt wird.

Beispiele:

```
description.0=DirectX 9 Treiber
```

```
description.1=.NET Framework (mit SP2)
```

```
description.1.0=Ruft den Registrierungseditor auf
```

```
description.1.0.0=Gibt in einer DOS-Box den Text "Hallo Welt!" aus
```

Anmerkungen & Hinweise:

- a) Um den Beschreibungstext als nicht ausführbare, fett formatierte Überschrift erscheinen zu lassen, siehe Unterpunkt d) bei `command.x.y.z=<Befehl>`
- b) Der Standardwert lautet „<missing description>“. Im [Protokoll\(-fenster\)](#) von WIHU erscheint dies, wenn Sie zu einem ausgeführten `command.x.y.z=<Befehl>` vergessen haben, eine Beschreibung einzugeben.

4.2.3.4 `selected.x.y.z=[0|1]`Zweck:

Legt fest, ob die dem Befehl `command.x.y.z=<Befehl>` zugeordnete Auswahlbox in der Softwareliste von WIHU (siehe Abbildung 6) angekreuzt ist (Wert 1) oder nicht (Wert 0).

Damit kann beispielsweise eine Standardauswahl zu installierender Software vorgeben werden.

Anmerkungen & Hinweise:

- a) Standardwert ist „0“, also keine Vorabauswahl des entsprechenden Eintrages.
- b) Für die oberste Ebene der Baumstruktur (siehe Kapitel 4.2.2) ist die Anweisung bei Bedarf pro Abschnitt genau 1x ohne den Nummernindex `.x.y.z` anzugeben. Standardwert für die oberste Ebene ist „1“, also Auswahl des entsprechenden Baumastes.
- c) Der angegebene Wert wird *nicht* an die Unterebenen weitervererbt. Dort muß die Anweisung bei Bedarf erneut angegeben werden.
- d) Soll der Benutzer keine Möglichkeit haben, den vorgegebenen Wert zu ändern, muß dieser mit Hilfe der Option `hidden.x.y.z=[0 | 1]` versteckt werden.

4.2.3.5 hidden.x.y.z=[0|1]

Zweck:

Versteckt (Wert 1) den mit `command.x.y.z=<Befehl>` angegebenen Befehl in der Auswahlliste, so daß er durch den Benutzer nicht mehr an- oder abgewählt werden kann.

Anmerkungen & Hinweise:

- a) Standardwert ist „0“, also Anzeige des entsprechenden Wertes.
- b) Für die oberste Ebene der Baumstruktur (siehe Kapitel 4.2.2) ist die Anweisung bei Bedarf pro Abschnitt genau 1x ohne den Nummernindex `.x.y.z` anzugeben.
- c) Der angegebene Wert wird an die Unterebenen weitervererbt, so gilt also beispielsweise `.0.1=1` auch für `.0.1.0`, `.0.1.1`, `0.1.2`, `0.1.2.0`, `0.1.2.1`, etc. Die Weitervererbung bei dieser Anweisung ist zwingend, d.h. der zugewiesene Wert auf der höchsten Ebene gilt, anderslautende Wertzuweisungen auf Unterebenen werden ignoriert.
- d) Wenn ein Sektionsname mit `selected.x.y.z=[0|1]` oder auf der graphischen Benutzeroberfläche (siehe Abbildung 6) abgewählt wird, werden die dazugehörigen versteckten Einträge ebenfalls nicht installiert.

4.2.3.6 collapsed.x.y.z=[0|1]

Zweck:

Legt fest, ob der entsprechende Zweig des Baumes in der Softwareliste von WIHU (siehe Abbildung 6) standardmäßig aus- (Wert 0) oder eingeklappt (Wert 1) dargestellt werden soll.

Anmerkungen & Hinweise:

- a) Standardwert ist „0“, also Ausklappen des entsprechenden Astes.
- b) Für die oberste Ebene der Baumstruktur (siehe Kapitel 4.2.2) ist die Anweisung bei Bedarf pro Abschnitt genau 1x ohne den Nummernindex `.x.y.z` anzugeben.
- c) Der angegebene Wert wird an die Unterebenen weitervererbt, so gilt also beispielsweise `.0.1=1` auch für `.0.1.0`, `.0.1.1`, `0.1.2`, `0.1.2.0`, `0.1.2.1`, etc. Die Weitervererbung bei dieser Anweisung ist zwingend, d.h. der zugewiesene Wert auf der höchsten Ebene gilt, anderslautende Wertzuweisungen auf Unterebenen werden ignoriert.

4.2.3.7 file.x.n=<Datei>?[=!|<|>]<Versionsangabe>

Zweck:

Liest die in der Datei vermerkte Versionsnummer aus, vergleicht diese mit der in der Anweisung angegebenen Versionsangabe und führt den mit Hilfe der Anweisung `command.x.y.z=<Befehl>` angegebenen Befehl nur bei positivem Ergebnis aus.

So erlaubt Ihnen WIHU, Software abhängig vom Systemzustand zu installieren. Ein Anwendungsbeispiel dieser Funktion wäre, Software nur dann aufzuspielen, wenn nicht schon eine neuere Version derselben Applikation auf dem System vorhanden ist.

Anwendung:

Es existieren vier Vergleichsoperatoren, anzugeben direkt nach dem „?“ . Wenn das Ergebnis eines Vergleichs bejaht wird, gilt es als positiv.

=	prüft, ob beide Versionsangaben übereinstimmen
!	prüft, ob beide Versionsangaben unterschiedlich sind
<	prüft, ob die <Datei>-Version älter/kleiner ist als <Versionsangabe>
>	prüft, ob die <Datei>-Version neuer/größer ist als <Versionsangabe> [Std.]

Tabelle 4: Vergleichsoperatoren

Für den Fall, daß die angegebene Datei keine Versionsinformation enthält, prüft WIHU, ob die Datei am angegebenen Ort überhaupt tatsächlich existiert. Zu den unterschiedlichen Möglichkeiten sehen Sie sich bitte folgende Übersicht an:

Operator	<Datei> hat keine Vers.info & ist ...	Ausführung von <Befehl>
>	vorhanden	nein
>	nicht vorhanden	ja
!	vorhanden	ja
!	nicht vorhanden	Nein

Tabelle 5: Fallunterscheidung bei fehlender Dateiversionsinformation

Beispiele:

```
command.5.3.4=DirectX9b\DX9NTopk.exe  
file.5.0=%SystemRoot%\System32\abc.dll?!9.0.0.3196
```

- ➔ Führt den Befehl “DirectX9b\DX9NTopk.exe” nur aus, falls die in der Datei abc.dll vermerkte Dateiversion nicht 9.0.0.3196 lautet. Falls die Datei keine Versionsinformation enthält, wird der Befehl nur ausgeführt, sofern die Datei vorhanden ist.

```
command.3.0.0=DirectX9b\DX9NTopk.exe  
file.3.0=%SystemRoot%\System32\abc.dll?>9.0.0.3196
```

- ➔ Führt den Befehl “DirectX9b\DX9NTopk.exe” nur aus, falls die in der Datei abc.dll vermerkte Dateiversion größer ist als 9.0.0.3196. Falls die Datei keine Versionsinformation enthält, wird der Befehl nur ausgeführt, sofern die Datei nicht vorhanden ist.

Anmerkungen & Hinweise:

- a) Standardvergleichsoperator ist „>“, d.h. wenn Sie nach dem „?“ keinen Operator angeben, wird automatisch der Standardoperator herangezogen.
- b) Diese Anweisung verwendet ein dem Zweck leicht angepaßtes [Numerierungssystem](#):
Der erste Index .x läuft mit dem der command.x.y.z=<Befehl>-Anweisung gleich, um hier die Zuordnung zwischen Befehl und Vergleich zu kennzeichnen. Der zweite Index .n hingegen dient allein der Numerierung und ist unabhängig von anderen Anweisungen; siehe auch file.comparison.x.y.z=[AND|OR] .
Der zweite Index .n muß immer mitangegeben werden, auch wenn nur ein einziger Schlüssel geprüft wird.
- c) Die Verwendung von Systemvariablen wie auch der WIHU-spezifische Umgebungsvariablen ist möglich. Außerdem zugelassen sind alle Umgebungsvariablen aus dem Abschnitt [Environment].
- d) Bitte streichen Sie für diese Anweisung aus der Versionsangabe führende Nullen:
Aus „4.09.00.0902“ sollten Sie also besser „4.9.0.902“ machen.
- e) Sobald in der Versionsangabe nicht numerische Zeichen wie zum Beispiel „.“ vorkommen, werden die Werte alphanumerisch verglichen.

4.2.3.8 file.comparison.x.y.z=[AND|OR]

Zweck:

Setzt fest, auf welche Weise die Ergebnisse von mehreren zu einer `command.x.y.z=<Befehl>-Anweisung` gehörenden Vergleiche mittels `file.x.n=<Datei>?[=|!|<|>]<Versionsangabe>` zu einem Gesamtergebnis verrechnet werden sollen.

So erlaubt Ihnen WIHU, die Ausführung von `<Befehl>` nicht nur von einem Vergleich abhängig zu machen, sondern von einer Kombination mehrerer.

Beispiel:

```
command.5.3.4=DirectX9b\DX9NTopk.exe
file.5.0=%SystemRoot%\System32\abc.dll?9.0.0.3196
file.5.0=%SystemRoot%\System32\mscoree.dll?1.1.4322.573
file.comparison.5=AND
```

➔ Führt den Befehl „DirectX9b\DX9NTopk.exe“ nur aus, wenn wegen AND ... sowohl die Version der Datei `abc.dll` größer ist als `9.0.0.3196` bzw. die Datei bei Fehler einer Dateiversion auch nicht vorhanden ist als auch die Dateiversion von `mscoree.dll` größer ist als `1.1.4233.573`.

Anmerkungen & Hinweise:

- a) AND (=logisches UND) führt nur zu einem positiven Ergebnis, wenn ausnahmslos alle Vergleiche innerhalb der `.x`-Gruppe bejaht werden. Ein OR-Vergleich (=logisches ODER) hingegen fällt positiv aus, sobald mindestens einer der Vergleiche innerhalb der entsprechenden Gruppe bejaht wird.
- b) Die Anweisung unterstützt ausschließlich die zwei Verknüpfungsoperatoren AND sowie OR. Die anderen Verknüpfungsmöglichkeiten XOR (=logisches exklusives ODER), NOR und NAND können Sie durch eine entsprechende Kombination mit den Vergleichsoperatoren emulieren.

4.2.3.9 key.x.n=[HKLM|HKCU]:<Reg.schlüssel>?[=|!<|>]<Zeichenkette>

Zweck:

Liest den im Registrierungsschlüssel hinterlegten Wert aus, vergleicht diesen mit der in der Anweisung angegebenen Zeichenkette und führt den mit Hilfe der Anweisung `command.x.y.z=<Befehl>` angegebenen Befehl nur bei positivem Ergebnis aus.

So erlaubt Ihnen WIHU, Software abhängig vom Systemzustand zu installieren. Ein Anwendungsbeispiel dieser Funktion wäre, Software nur dann aufzuspielen, wenn nicht schon eine neuere Version derselben Applikation auf dem System vorhanden ist.

Anwendung:

Die Vergleichsoperatoren sind exakt dieselben wie bei der vorab beschriebenen Anweisung `file.x.n=<Datei>?[=|!<|>]<Versionsangabe>`.

Registrierungswerte können aus den beiden Zweigen `HKEY_LOCAL_MACHINE` und `HKEY_CURRENT_USER` ausgelesen werden. Je nachdem geben Sie dafür entweder die Abkürzung `HKLM` oder `HKCU` an. Den Standardwert eines Registrierungsschlüssels lesen Sie aus, wenn Sie an den entsprechenden Astnamen einfach einen umgekehrten Schrägstrich „\“ anfügen.

Beispiel:

```
command.2=Nero\Nero6.msi
key.2.0=HKLM:Hardware\Devices\CDR\Siemens\?=Siemens AG
```

Anmerkungen & Hinweise:

- a) Standardvergleichsoperator ist „>“, d.h. wenn Sie nach dem „?“ keinen Operator angeben, wird automatisch der Standardoperator herangezogen.
- b) Diese Anweisung verwendet ein dem Zweck leicht angepaßtes [Numerierungssystem](#): Der erste Index `.x` läuft mit dem der `command.x.y.z=<Befehl>`-Anweisung gleich, um hier die Zuordnung zwischen Befehl und Vergleich zu kennzeichnen. Der zweite Index `.n` hingegen dient allein der Numerierung und ist unabhängig von anderen Anweisungen; siehe auch `key.comparison.x.y.z=[AND|OR]`. Der zweite Index `.n` muß immer mitangegeben werden, auch wenn nur ein einziger Schlüssel geprüft wird.

- c) Die Verwendung von Systemvariablen wie auch der WIHU-spezifische Umgebungsvariablen ist möglich. Außerdem zugelassen sind alle Umgebungsvariablen aus dem Abschnitt [Environment].
- d) Bei der Prüfung wird nicht zwischen Groß- und Kleinschreibung unterschieden.
- e) Anführungsstriche im Fall von Leerzeichen in der Zeichenkette sind nur nötig, wenn diese auch so im Registrierungsschlüsselwert stehen und in den Vergleich miteinfließen sollen.
- f) Mit dieser Anweisung können Sie nicht nur numerische Versionsangaben vergleichen, sondern jede Art von Zeichenkette, die in einem Registrierungsschlüssel hinterlegt ist.

4.2.3.10 `key.comparison.x.y.z=[AND|OR]`

Analoge Anwendung zur Anweisung `file.comparison.x.y.z=[AND|OR]`.

4.2.4 Abschnitt [Environment]

In diesem Abschnitt können Sie vorübergehend gültige Hilfsumgebungsvariablen für den Installationsprozeß definieren, welche Sie in der Arbeit mit WIHU zur Feinsteuerung des Installationsprozesses einsetzen können, Sie können weiterhin Dauerhafte Umgebungsvariablen pro Benutzer definieren als auch die windowseigenen Standardvorgaben benutzerkontospezifischer Ordner manipulieren.

Auf diese Weise können Sie in WIHU Systeminformationen zur Unterstützung des Installationsprozesses heranziehen und weitere benutzerspezifische Einstellungen gleich während des Installationsprozesses auf einfache Weise, mit sofortiger Wirkung und dauerhaft vornehmen. Dies erspart Ihnen zum Beispiel den Einsatz eines zusätzlichen Optimierungstools nach Abschluß der Betriebssysteminstallation.

Dieser Abschnitt wird übrigens unabhängig von der schriftlichen Reihenfolge in der Konfigurationsdatei immer vor dem Abschnitt [Users] geladen. Falls Sie die benutzerspezifischen Einstellungen per `/Users=<Datei>` in eine andere Datei auslagern, muß der Abschnitt [Environment] in der ursprünglichen Konfigurationsdatei verbleiben.

4.2.4.1 Hilfsumgebungsvariablen für den Installationsprozeß definieren

Zweck:

Definieren Sie frei nach eigenem Gusto Umgebungsvariablen, die Sie überall in WIHU verwenden können, um so den Installationsprozeß zu verfeinern.

Anwendung:

Die zulässige Syntax zur Definition einer neuer Variablen lautet:

```
<Variablenname>=<beliebiger Wert>
```

Sie können Ihre Umgebungsvariablen darüber hinaus auch mit Werten aus der Windowsregistrierung füllen.

```
<Variablenname>=<HKLM|HKCU>:Pfad\{Wert}
```

Registrierungswerte können aus den beiden Zweigen `HKEY_LOCAL_MACHINE` und `HKEY_CURRENT_USER` ausgelesen werden. Je nachdem geben Sie dafür entweder die Abkürzung `HKLM` oder `HKCU` an. Den Standardwert eines Registrierungsschlüssels lesen Sie aus, wenn Sie an den entsprechenden Astnamen einfach einen umgekehrten Schrägstrich „\“ anfügen.

Beispiele (jeweils in einer Zeile zu notieren!):

```
Name meines Hasen=Dr. K. Nickel
; definiert eine Variable mit beliebigem Wert

NUMPROC=HKLM:SYSTEM\CurrentControlSet\Control\SessionManager\
  Environment\NUMBER_OF_PROCESSORS

Moz=HKLM:Software\Mozilla\
; liest aus dem Zweig HKLM und dem Ast „Software\Mozilla\“ den
  Standardwert aus

MozillaFireFoxVersion=
  HKLM:Software\Mozilla\Mozilla Firefox\CurrentVersion

MozillaFireFoxPath=
  HKLM:Software\Mozilla\MozillaFirefox\%MozillaFireFoxVersion%\
  Main\Install Directory
; referenziert die zuvor definierte Variable „MozillaFireFoxVersion“
```

Anmerkungen & Hinweise:

- a) Wie üblich muß an anderer Stelle (außer natürlich der Definition) zum Auslesen des Wertes der Variablenname in Prozentzeichen eingeschlossen werden.
- b) Die einzelnen Variablen dürfen sich auch gegenseitig referenzieren. Im Beispiel ist dieser Umstand farblich hervorgehoben. Variablen, auf die verwiesen wird, müssen zuvor definiert worden sein.
- c) Die maximal zulässige Länge beträgt 260 Zeichen.
- d) Falls ein aus der Registrierung auszulesender Wert nicht existiert, wird die Variable zwar angelegt, bleibt aber ohne Wert (leer). Letztlich kommt dies in diesem Umfeld einer nicht existierenden Variablen gleich.
- e) Die hier definierten eigenen Variablen sind ausschließlich zur Laufzeit und nur innerhalb (nicht an der Befehlszeile) von WIHU gültig. Sie können auch bereits vorhandene Variablenwerte überschreiben – dies umfaßt auch die üblichen Windowsvariablen wie bspw. %path%; neu definierte Variablen verfallen jedoch mit der Beendigung von WIHU.

4.2.4.2 Dauerhafte Umgebungsvariablen pro Benutzer definieren

Zweck:

Entgegen der im Vorkapitel beschriebenen Möglichkeit, nur temporäre Umgebungsvariablen festzulegen, können Sie mit WIHU auch für jeden Benutzer getrennt dauerhaft gültige Variablen vorgeben, welche Ihnen später unter Windows im Rahmen des jeweilig angemeldeten Benutzers jederzeit zur Verfügung stehen.

Nützlich kann dies zum Beispiel bei Programmen wie [GnuPG](#) sein, welche für das korrekte Funktionieren eigene Variablen (meist in der Form zusätzlicher Pfadangaben) benötigen.

Anwendung:

Setzen Sie alle gewünschten Variablen, die Sie dauerhaft dem Benutzerkonto `x` zuweisen möchten, unter den Sonderabschnitt `[environment.x]`.

Der Index `.x` steht dabei für den entsprechenden Benutzer aus dem Abschnitt `[Users]`.⁴

Beispiele:

```
[environment.0]
Nachname des Haustiers=Nickel
```

```
[environment.3]
CryptKeys=C:\GnuPG\
KeyServer=192.168.98.99:1234
```

```
[environment.4]
CryptKeys=C:\GnuPG\%ThisUser%5
```

Anmerkungen & Hinweise:

- a) Die Darstellung in Windows erfolgt wie in Abbildung 10 (oberer Teil) zu sehen.
- b) Die Sondervariablen WIHUs sind zulässig. Siehe auch Fußnote 5.
- c) Dieser Abschnitt braucht nur für diejenigen Benutzer angelegt zu werden, wo diese Funktion benötigt wird. Lücken in der Indexnummernabfolge sind hier unbedeutend.

⁴ Die Einrichtung neuer Benutzerkonten mittels WIHU wird später in dieser Anleitung beschrieben

⁵ Mehr zu dieser besonderen Variable im Folgekapitel 4.2.4.3 bzw. in Tabelle 7

- d) Die Sektionen der Form [environment.x] müssen entgegen dem allgemeinen Abschnitt [Environment] immer in der Datei stehen, wo auch der Abschnitt [Users] zu finden ist. Achten Sie beim Befehlszeilenparameter /Users=<Datei> hierauf.

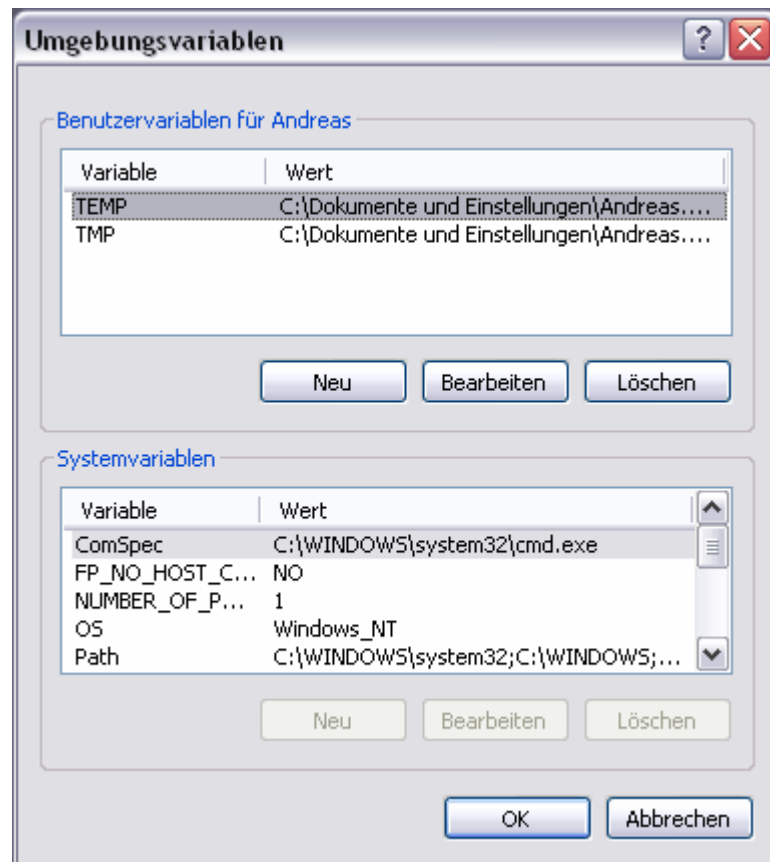


Abbildung 10: Windowsdialog zu dauerhaften Benutzervariablen

4.2.4.3 Standardvorgaben benutzerkontospezifischer Ordner manipulieren

Zweck:

Windows verknüpft mit jedem Benutzerkonto spezielle Ordner auf der Festplatte wie beispielsweise „Eigene Dateien“ oder „Desktop“. Die Standardpfade dieser Verzeichnisse können Sie mit WIHU (dauerhaft) ändern.

Anwendung:

Vorzunehmen im allgemeinen Abschnitt [Environment].

Die zulässige Syntax zur Manipulation eines solchen Ordneintrages lautet:

```
<reservierter Variablenname>=<Pfadangabe>
```

Die verfügbaren reservierten Variablennamen entnehmen Sie der folgenden Tabelle.

reserv. Var.name	Standardwert bei deutschem Windows XP
Administrative Tools	(anfänglich nicht vorbelegt)
Cache	[Profilverzeichnis]\Lokale Einstellungen\Temporary Internet Files
CD Burning	[Profilverzeichnis]\Lokale Einstellungen\ Anwendungsdaten\ Microsoft\CD Burning
Cookies	[Profilverzeichnis]\Cookies
Home	sog. „Basisverzeichnis“ Windowsstandard: „%SystemDrive%\Dokumente und Einstellungen\%UserName%“
Desktop	[Profilverzeichnis]\Desktop
Favorites	[Profilverzeichnis]\Favoriten
Fonts	%WinDir%\Fonts
History	[Profilverzeichnis]\Lokale Einstellungen\Verlauf
Local AppData	[Profilverzeichnis]\Lokale Einstellungen\Anwendungsdaten
Local Settings	[Profilverzeichnis]\Lokale Einstellungen
My Music	[Profilverzeichnis]\Eigene Dateien\Eigene Musik
My Pictures	[Profilverzeichnis]\Eigene Dateien\Eigene Bilder
My Video	(anfänglich nicht vorbelegt)
NetHood	[Profilverzeichnis]\Netzwerkumgebung
Personal	[Profilverzeichnis]\Eigene Dateien
PrintHood	[Profilverzeichnis]\Druckumgebung
Programs	[Profilverzeichnis]\Startmenü\Programme
Recent	[Profilverzeichnis]\Recent

SendTo	[Profilverzeichnis]\SendTo
Start Menu	[Profilverzeichnis]\Startmenü
Startup	[Profilverzeichnis]\Startmenü\Programme\Autostart
Templates	[Profilverzeichnis]\Vorlagen

Tabelle 6: reservierte Variablennamen für benutzerspezifische Ordner

Neben diesen reservierten Variablen gibt es noch weitere sog. „Sondervariablen“, welche Sie in der Wertzuweisung verwenden können. Der Wert der Sondervariablen ist nicht fest, sondern ändert sich je nach Kontext. Sie gelten nicht an der Befehlszeile. Ein Anwendungsbeispiel besteht beispielsweise darin, daß Sie mit Hilfe der Sondervariablen mehrere gleichartige Anweisungen im Abschnitt [Users] zu einer einzelnen Ersatzanweisung im Abschnitt [Environment] zusammenfassen können.

Sondervariable	Bedeutung
%ThisUser%	entsprechender Benutzername
%ThisHome%	entsprechendes Heimverzeichnis Bei WinXP gleicht dies standardmäßig %ThisProfile%, bei Win2000 entspricht dies aber %SystemDrive%
%ThisProfile%	wird durch den entsprechenden Profilpfad ersetzt, in dem alle Benutzerkontodaten gespeichert werden im Windowsstandardfall: “%SystemDrive%:\Dokumente und Einstellungen\ %UserName%“
%NewUserName%	Benutzername des primären Benutzerkontos ; falls dieses nicht verwendet wird (siehe /UseCurrent), Benutzername des aktuell angemeldeten Benutzers
%NewUserPassword%	Paßwort des primären Benutzerkontos in Klartextform; falls dieses Konto nicht verwendet wird (siehe /UseCurrent) enthält die Variable nichts/bleibt leer

Tabelle 7: Sondervariablen für die Konfigurationsdatei

Beispiele:

My Video=C:\Privates\Videosammlung von Familie Normal

; Beachten Sie, daß dieses Verzeichnis für alle Benutzer gilt

Favorites=%systemroot%\Unsere Favoriten

; Beachten Sie, daß dieses Verzeichnis für alle Benutzer gilt

Personal=%SystemDrive%\Benutzerprofile\Daten von %ThisUser%

; Gilt generell auch für alle Benutzer, aber durch Verwendung der Sondervariable %ThisUser% wird für jedes Benutzerkonto getrennt das entsprechende Verzeichnis angelegt. So ersetzen Sie die mehrmalige Angabe der [vergleichbaren Anweisung](#) im Abschnitt [Users].

Home=%SystemDrive%\Benutzerprofile\%ThisUser%

Home=%Personal%\%ThisUser%

; %Personal% kann, muß aber nicht vorher belegt worden sein. Falls nicht, wird einfach der Windowsstandardwert verwendet.

Anmerkungen & Hinweise:

- a) Die im Abschnitt [Environment] vorgenommenen Änderungen an den Einstellungen der benutzerkontospezifischen Ordner gelten für jeden Benutzer! Das schließt auch die mit der graphischen Oberfläche von WIHU erstellen Konten ein (siehe Abbildung 2) sowie die, welche später in Windows nachträglich angelegt werden. Dediziert benutzerkontospezifische Einstellungen hingegen, d.h. nicht für alle Konten gleichermaßen gültig, nehmen Sie im Abschnitt [Users] vor oder in Spezialfällen auch hier wie oben dargestellt mit den Sondervariablen für die Konfigurationsdatei.
- b) Die Änderungen an den Standardverzeichnissen sind unmittelbar nach dem Start von WIHU systemweit aktiv und werden gleichzeitig dauerhaft für die Zeit nach der Installation in der Windowsregistrierung gespeichert.
- c) Die Verwendung von Systemvariablen wie auch der WIHU-spezifische Umgebungsvariablen ist möglich. Auf die reservierten Variablennamen aber kann nur innerhalb von WIHU-Konfigurationsdateien zurückgegriffen werden, nicht jedoch an der Befehlszeile.
- d) siehe auch Abbildung 4: Dialog 1b – Reiter Systemordner; Änderungen an der graphischen Benutzeroberfläche haben Vorrang vor der Konfigurationsdatei
- e) Die einzelnen Variablen dürfen sich auch gegenseitig referenzieren.

4.2.5 Abschnitt [Users]

Hier können Sie zum einen Benutzerkonten neu anlegen und zum anderen für diese neuen Benutzerkonten unabhängig voneinander Einstellungen vornehmen. Dieser Abschnitt kann entweder in der Standardkonfigurationsdatei von WIHU vorkommen (siehe /INI=<Datei>) oder aber in einer separaten Datei mittels /Users=<Datei>. Falls der Abschnitt in beiden Dateien zugleich zu finden ist, wird derjenige aus der per /INI=<Datei> referenzierten Datei ignoriert.

Die Anweisungen in diesem Abschnitt verwenden ein dem Zweck leicht angepaßtes [Numerierungssystem](#): Es gibt nur den Index .x, keine Unterebenen. Alle Anweisungen mit identischem Index beziehen sich auf ein- und dasselbe Benutzerkonto. Bis auf eine Ausnahme sind Anweisungen ohne nachgestellten Index hier nicht zulässig.

Die hier definierten Konten samt Einstellungen werden im entsprechenden Dialog der graphischen Benutzeroberfläche von WIHU ebenfalls angezeigt (siehe Abbildung 2) und die Einstellungen/Änderungen dort haben Vorrang vor der Konfigurationsdatei.

Richten Sie Ihr Augenmerk erneut auf die bereits vorab beschriebenen reservierte Variablennamen für benutzerspezifische Ordner sowie die Sondervariablen für die Konfigurationsdatei, welche auch in diesem Abschnitt angewandt werden können.

Bitte beachten Sie auch die Hinweise in Kapitel 4.1.1! Die Konfiguration der Benutzerkonten kann je nach Aufrufsort von WIHU Probleme bereiten.

Dieser Abschnitt wird übrigens unabhängig von der schriftlichen Reihenfolge in der Konfigurationsdatei immer nach dem Abschnitt [Environment] geladen, überschreibt also damit dort schon vorgenommene Einstellungen mit den neuen Werten, sofern angegeben.

Möchten Sie bereits vor Aufruf von WIHU bestehende Benutzerkonten bearbeiten, sehen Sie sich hierzu den Abschnitt [Users.Operation] an.

4.2.5.1 user.x=<Name>

Zweck:

Legt ein neues Benutzerkonto unter dem angegebenen Namen an.

Beispiel:

```
user.0=Otto Normal
```

```
user.1=PC Guru
```

Anmerkungen & Hinweise:

- a) Der Name entspricht der Bezeichnung, die später am Anmeldebildschirm von Windows angezeigt wird.
- b) Das [primäre Benutzerkonto](#) können Sie auch an der Befehlszeile über `/User=<Name>` erstellen.
- c) maximal zulässige Länge beträgt 20 Zeichen

4.2.5.2 group.x=[0|1|2]

Zweck:

Legt fest, welcher Benutzergruppe der entsprechende Benutzer angehört.

So kann bestimmt werden, ob ein Benutzerkonto zur Systemverwaltung dienen soll (Administratorengruppe, Wert 1) oder hauptsächlich für die tägliche Arbeit eingesetzt wird (Benutzergruppen, Wert 0).

Anwendung:

Wert	Bedeutung
0	fügt den Benutzer der Benutzergruppe hinzu und entfernt ihn ggf. gleichzeitig aus der Administratorengruppe (Standardwert)
1	fügt den Benutzer der Administratorengruppe hinzu und entfernt ihn ggf. gleichzeitig aus der Benutzergruppe
2	fügt den Benutzer keiner Gruppe hinzu

Tabelle 8: Festlegung der Gruppenzugehörigkeit eines Benutzerkontos

Anmerkungen & Hinweise:

- a) Die Gruppe „Benutzer“ verfügt von Haus aus über eingeschränkte Rechte, wohingegen die Benutzergruppe „Administratoren“ über volle Zugriffsrechte verfügt.
- b) Windows bietet insgesamt drei Benutzergruppen zur Auswahl an: Benutzer, Hauptbenutzer (engl. power user) und Administratoren. In WIHU können Sie aber lediglich zwischen der ersten oder der letzten auswählen. Dies gründet auf der Tatsache, daß die Gruppe der Hauptbenutzer von Microsoft nur eingeführt wurde, um bei der Veröffentlichung von Windows 2000 und XP die Kompatibilität zu alten Anwendungen, die noch nicht mit der Rechteverwaltung moderner Betriebssysteme umzugehen wußten, zu gewährleisten; die Rechte eines Hauptbenutzers entsprechen daher weitestgehend bis auf kleine Einschränkungen denen eines Administrators. Prinzipiell wird empfohlen, keine Hauptbenutzerkonten zu verwenden.
- c) Es ist auf Grund von Sicherheitsüberlegungen dringend anzuraten, die tägliche Arbeit am Rechner ausschließlich unter eingeschränkten Benutzerrechten durchzuführen! Diese sind für diese Art von Arbeit vorgesehen. Lediglich für die Fälle, wo volle Zugriffsrechte tatsächlich benötigt werden, d.h. für Installationen von Software und Geräten oder aber für die Vornahme systemnaher Einstellungen, sollte kurzzeitig auf die dann erforderlichen Administratorprivilegien zurückgegriffen werden! Alles andere setzt das System und Ihre Daten einem stark erhöhten Sicherheitsrisiko aus. Siehe auch Kapitel „primäres Benutzerkonto erstellen“.
- d) Falls Sie WIHU zur Installation der Software das sog. „primäre Benutzerkonto“ verwenden lassen (siehe Kapitel 3.1 sowie `default=<Indexnummer>`) und diesem nicht Gruppe „1“ zugewiesen haben, verfährt WIHU automatisch wie in c) dargestellt: Während der Installationsphase verwendet es die hierfür benötigten Administratorprivilegien (Gruppe „1“), nach Abschluß der Installation setzt es das primäre Benutzerkonto auf die empfohlenen eingeschränkten Benutzerrechte (Gruppe „0“) zurück, welche für die tägliche Arbeit vorgesehen und ausreichend sind. Das Zurücksetzen erfolgt ausreichend lange nach den Installationen bei der ersten Anmeldung des primären Benutzerkontos am System.
Das Konto wird nach Abschluß der Installation in seinen Rechten nicht zurückgesetzt, sofern Sie es der Gruppe „1“ zuweisen; von diesem Vorgehen wird aber aus Sicherheitsüberlegungen heraus abgeraten.
- e) Falls Sie die Anweisung nicht angeben, wird für bereits bestehende Benutzerkonten deren Wert übernommen bzw. keine Änderung durchgeführt und bei per WIHU erstellen Benutzerkonten beträgt der Standardwert „0“.

4.2.5.3 status.x=<Kennzahl>

Zweck:

Legt den Status des entsprechenden Benutzerkontos fest.

Anwendung:

Kennzahl	Bedeutung
0x0002	Konto als deaktiv markieren
0x0020	Konto benötigt kein Paßwort
0x0040	Benutzer kann Paßwort nicht ändern
0x10000	Paßwort läuft nicht ab
0x800000	Paßwort als abgelaufen markieren

Tabelle 9: Überblick über die Statuskennzahlen bei Benutzerkonten

Falls Sie mehrere der Möglichkeiten zugleich auf ein Konto anwenden möchten, addieren Sie einfach die gewünschten Kennzahlen.

Beispiel:

```
status.0=0x0002
; entsprechendes Konto wird als deaktiv markiert (nicht gelöscht!)
```

```
status.1=0x10042
; Kombination aus 0x0002, 0x0040 und 0x10000
```

Anmerkungen & Hinweise:

- Abschließende Nullen brauchen nicht angegeben zu werden, d.h. 0x800000 = 0x8
- Weiterreichende Änderungen am Benutzerkonto können Sie mit Hilfe der Anweisungen im Abschnitt [Users.Operation] vornehmen.

4.2.5.4 profile.x=<Pfadangabe>

Zweck:

Weist dem entsprechenden Konto einen neuen Profilpfad zu.

Anmerkungen & Hinweise:

- a) im Windowsstandardfall entspricht dies “%SystemDrive%:\Dokumente und Einstellungen\ %UserName%“
- b) siehe auch Abbildung 3: Dialog 1a – Reiter Benutzerkonto
- c) auf die Pfadangabe wird für den jeweiligen Benutzer durch die Sondervariable %ThisProfile% verwiesen

4.2.5.5 RoamingProfile.x =<Pfadangabe>

Zweck:

Gibt für das entsprechende Konto den Pfad für ein servergespeichertes Benutzerprofil (Gegensatz: lokal gespeichertes Profil) vor.

Anmerkungen & Hinweise:

- a) siehe auch Abbildung 3: Dialog 1a – Reiter Benutzerkonto

4.2.5.6 comment.x=<Beschreibung>

Zweck:

Weist dem entsprechenden Konto eine kurze, erklärende Beschreibung hinzu.

Beispiel:

```
comment.0=Onkel von Mausli mütterlicherseits
```

Anmerkungen & Hinweise:

- a) Die Beschreibung ist in Windows im entsprechenden Dialog der Computerverwaltung zu sehen. Dort kann diese auch abgeändert werden.
- b) siehe auch Abbildung 3: Dialog 1a – Reiter Benutzerkonto
- c) maximal zulässige Länge beträgt 200 Zeichen

4.2.5.7 script.x=<Datei>

Zweck:

Legt für den entsprechenden Benutzer das zu verwendende Anmeldeskript fest.

Anmerkungen & Hinweise:

- a) Ein Anmeldeskript ist in der Regel eine Stapelverarbeitungsdatei (in seltenen Fällen auch eine EXE-Datei), die automatisch bei jedem Anmelden des Benutzers ausgeführt wird.
- b) Der Wert ist in Windows im entsprechenden Dialog der Computerverwaltung zu sehen. Dort kann diese auch abgeändert werden.
- c) maximal zulässige Länge beträgt 260 Zeichen

4.2.5.8 domain.x=<Name>

Zweck:

Legt die Domäne fest, auf der das Benutzerkonto angelegt werden soll

Anmerkungen & Hinweise:

- a) siehe auch Abbildung 3: Dialog 1a – Reiter Benutzerkonto
- b) maximal zulässige Länge beträgt 15 Zeichen

4.2.5.9 password.x=<Paßwort>

Zweck:

Weist dem entsprechenden Konto das gewählte Paßwort zu.

Anmerkungen & Hinweise:

- a) In den Windows-Gruppenrichtlinien (gpedit.msc) kann festgelegt worden sein, daß alle neuen Benutzerkontenpaßwörter bestimmte Mindestsicherheitsanforderungen (alphanumerisch, Mindestlänge, etc.) erfüllen müssen. WIHU muß sich ebenfalls an diese Einschränkungen halten, übernimmt aber keine dementsprechende Prüfung des Kennwortes. Achten Sie daher auf die Übereinstimmung des von Ihnen gewählten Paßwortes mit den diesbezüglichen Sicherheitsanforderungen des Systems.

- b) Das Paßwort wird 1:1 übernommen, sprich Groß- und Kleinschreibung spielt hier eine Rolle!

Unzulässige Zeichen sind: / , \ [] : | < > + = ; ? * und alles kleiner ASCII-Code 32

- c) Das Paßwort für das [primäre Benutzerkonto](#) können Sie auch an der Befehlszeile über /UserPwd=<Paßwort> vorgeben.
- d) maximal zulässige Länge beträgt 100 Zeichen

4.2.5.10 crypted.x=<Wert>

Anmerkungen & Hinweise:

- a) Diese Anweisung wird von WIHU automatisch in der Konfigurationsdatei erstellt, wenn Sie das Programm mittels der Befehlszeilenoption /EncryptINI starten.
- b) Ändern oder löschen Sie diese Anweisung nicht selbständig! Ansonsten wird es zu Problemen mit den in der Konfigurationsdatei festgelegten Benutzerkontokennwörtern kommen.
- c) Zum Entfernen dieser Anweisungen benutzen Sie bitte ausschließlich den Befehlszeilenparameter /DecryptINI.

4.2.5.11 <reservierter Variablenname>.x=<Pfadangabe>

Zweck:

Belegt die in Tabelle 6 aufgeführten Variablen mit einem benutzerkontospezifischen Wert.

Anwendung:

Hängen Sie an den reservierten Variablennamen einfach den Index .x des gewünschten Benutzers an.

Beispiele:

```
My Video.0=C:\Privates\Videosammlung des Vaters
; Beachten Sie, daß dieses Verzeichnis nur für user.0 gilt

Favorites.1=%systemroot%\Favoriten der Mutter
; Beachten Sie, daß dieses Verzeichnis nur für user.1 gilt
```

```
Personal.1=%SystemDrive%\Sonderverzeichnis\Daten von %ThisUser%  
; %ThisUser% erhält hier automatisch den Namen von user.1 und  
überschreibt gleichzeitig nur für user.1 eventuelle anderen Vorgaben
```

Anmerkungen & Hinweise:

- a) Diese Anweisung überschreibt gleichzeitig nur für das angegebene/indizierte Konto den Standardwert oder die im Abschnitt [Environment] (insb. Kapitel 4.2.4.3) angegebene, zunächst für alle Benutzerkonten geltende Wertzuweisung.
- b) siehe auch Abbildung 4: Dialog 1b – Reiter Systemordner

4.2.5.12 default=<Indexnummer>

Zweck:

Legt durch die Indexnummer dasjenige Benutzerkonto fest, welches WIHU als sog. „[primäres Benutzerkonto](#)“ verwendet und zur Softwareinstallation gebraucht.

Beispiel:

```
default=0
```

Anmerkungen & Hinweise:

- a) Mehr Informationen zum primären Benutzerkonto in WIHU finden Sie hier:
 - i. Kapitel 3.1
 - ii. `group.x=[0 | 1 | 2]`, dort insbesondere Anmerkung d)
- b) Wird bei Angabe von `/User=<Name>` an der Befehlszeile ignoriert.
- c) Wird bei Angabe von `/UseCurrent` an der Befehlszeile ignoriert.
- d) Wenn die entsprechende Anweisung `comment.x=<Beschreibung>` nicht definiert wurde, wird der Standardkommentar „Primäres Benutzerkonto für tägliche PC-Arbeit“ für das Konto mit der Kennzahl `<Indexnummer>` verwendet.
- e) Wenn die Anweisung `status.x=<Kennzahl>` für das entsprechenden Konto nicht definiert wurde, wird dem Benutzerkonto automatisch der Status `0x10020` zugewiesen.

4.2.6 Abschnitt [Users.Operation]

Können Sie im Abschnitt [Users] ausschließlich Benutzerkonten bearbeiten, die Sie mittels WIHU innerhalb derselben Sitzung erstellen, erhalten Sie mit diesem Abschnitt die Möglichkeit, zum einen die mit WIHU erstellen Konten als auch die bereits vor Aufruf des Programms bestehenden Benutzerkonten anzupassen.

Dieser Abschnitt muß in derselben Datei vorkommen wie der Abschnitt [Users]. Achten Sie beim Befehlszeilenparameter `/Users=<Datei>` hierauf.

Die Anweisungen in diesem Abschnitt verwenden ein dem Zweck leicht angepaßtes [Numerierungssystem](#): Es gibt nur den Index `.n`, keine Unterebenen. Der Index bezieht sich zudem auf kein spezifisches Benutzerkonto, sondern erfüllt lediglich eine Numerierungsfunktion innerhalb dieses Abschnitts. Es muß zudem für jede der Anweisung getrennt numeriert werden.

Beispiel:

```
delete.0=ASPNET

;neue Anweisung und neuer Index
enable.0=Andreas
enable.1=Benjamin

;neue Anweisung und neuer Index
rename.0=Andreas, Otto Normal
rename.1=Otto Normal, Andreas
rename.2=Benjamin, Andreas
```

Die mit den Anweisungen dieses Abschnitts bearbeiteten Konten werden *nicht* im entsprechenden Dialog der graphischen Benutzeroberfläche von WIHU angezeigt (siehe Abbildung 2).

Beachten Sie bitte, daß WIHU selbst im Kontext eines Kontos mit Administratorrechten ausgeführt werden muß, um an den Benutzerkonten Änderungen egal welcher Art durchführen zu können. Die Erstellung und Verwendung des primären Benutzerkontos hat keinen Einfluß hierauf, ist davon unabhängig.

4.2.6.1 rename.n=<altes Konto>{@Domäne}, <neues Konto>

Zweck:

Benennt <altes Benutzerkonto> in <neues Benutzerkonto> um.

Beispiel:

```
rename.0=Andreas, Otto Normal
```

Anmerkungen & Hinweise:

- a) Die Umbenennung hat natürlich weiterreichende Wirkungen, z.B. werden die Umgebungsvariablen von Windows wie %UserName% natürlich an den neuen Namen angepaßt. Dies bedeutet natürlich überall dort Auswirkungen, wo Sie auf diese Variablen zurückgreifen. Eventuell müssen Sie hier dann Verzeichnisanpassungen vornehmen. Der Profilpfad bleibt aber unverändert und auch falls im Profilpfad der alte Benutzername enthalten sein sollte, ist eine Anmeldung weiterhin möglich.
- b) Die Angabe einer Domäne ist nur bei Bedarf nötig.

4.2.6.2 enable.n=<Benutzerkonto>{@Domäne}

Zweck:

Setzt den Status des angegebenen Benutzerkontos auf „aktiv“.

Dieses taucht dann am Anmeldebildschirm von Windows auf.

Beispiel:

```
enable.1=Gast
```

Anmerkungen & Hinweise:

- a) Neu erstellte Benutzerkonten werden von Windows zunächst generell aktiviert.
- b) Die Angabe einer Domäne ist nur bei Bedarf nötig.
- c) siehe auch `disable.n=<Benutzerkonto>{@Domäne}`

4.2.6.3 `disable.n=<Benutzerkonto>{@Domäne}`

Zweck:

Setzt den Status des angegebenen Benutzerkontos auf „deaktiv“.

Dieses taucht dann am Anmeldebildschirm von Windows nicht mehr auf und kann in Folge auch nicht mehr benutzt werden.

Beispiel:

```
disable.2=Otto Normal
```

Anmerkungen & Hinweise:

- a) Standardmäßig deaktiviert ist in Windows beispielsweise das Gast-Konto.
- b) Die Angabe einer Domäne ist nur bei Bedarf nötig.
- c) siehe auch `enable.n=<Benutzerkonto>{@Domäne}`

4.2.6.4 `delete.n=<Benutzerkonto>{@Domäne}`

Zweck:

Löscht das angegebene Konto dauerhaft.

Beispiel:

```
delete.3=ASPNET
```

Anmerkungen & Hinweise:

- a) Mit der Löschung ist das Konto unwiderruflich verloren! Auf diese Weise verlieren Sie auch den Zugang zu Daten, die Sie mit Hilfe der windowseigenen Verschlüsselungsfunktion EFS (Encrypting File System) für das jeweilige Konto chiffriert haben. Hier hilft es Ihnen auch nicht, wenn Sie wieder ein neues Konto mit demselben Namen erstellen.
- b) Der Befehl löscht auch gleich alle zu dem Konto gehörigen Profildaten auf der Festplatte. Unter gewissen Umständen können aber manchmal Dateien/Verzeichnisse zurückbleiben, die Sie dann eventuell manuell löschen müssten.
- c) Die Angabe einer Domäne ist nur bei Bedarf nötig.

4.2.7 Abschnitt [Settings]

Abschnitt für diverse Einstellungen, u.a. das Aussehen von WIHU betreffend.



Abbildung 11: Anpassung der Kurztexte in den WIHU-Fenstern

4.2.7.1 PageHeaderTitle.x=<Überschrift>

Zweck:

Setzt eine neue Überschrift für den Kurzbeschreibungstext im Fenster Nr. x

Beispiel:

```
PageHeaderTitle.0=Primäres Benutzerkonto erstellen
```

Anmerkungen & Hinweise:

- d) Die Numerierung der WIHU-Dialogfenster beginnt bei 0; es gibt keine Indexunterebenen in der Form .x.y...
- e) Die entsprechenden Nummern der einzelnen Dialogboxen können Sie auch den Abbildungsbeschriftungen aus Kapitel 3 „Programmablauf“ entnehmen.

4.2.7.2 PageHeaderSubTitle.x=<Kurzbeschreibungstext>

Zweck:

Gibt für das Programmfenster x einen neuen Kurzbeschreibungstext vor.

Hiermit können Sie zum Beispiel eigene Erklärungen oder Hilfestellungen zu dem jeweiligen Dialog anbieten.

Anmerkungen & Hinweise:

- a) analog zu `PageHeaderTitle.x=<Überschrift>`
- b) Es gibt keine Maximaltextlänge und auch keine Formatierungsmöglichkeiten.

4.2.7.3 `dimension.width=<Pixelwert>`

Zweck:

Öffnet WIHU mit der angegebenen Programmfensterbreite.

Anmerkungen & Hinweise:

- a) siehe auch `dimension.height=<Pixelwert>`
- b) siehe auch `dimension.update=[0 | 1]`

4.2.7.4 `dimension.height=<Pixelwert>`

Öffnet WIHU mit der angegebenen Programmfensterhöhe.

Anmerkungen & Hinweise:

- a) siehe auch `dimension.width=<Pixelwert>`
- b) siehe auch `dimension.update=[0 | 1]`

4.2.7.5 `dimension.update=[0|1]`

Zweck:

Speichert (Wert 1) bei Beendigung von WIHU die aktuellen Fensterausmaße in der Konfigurationsdatei ab.

Anmerkungen & Hinweise:

- a) siehe auch `dimension.width=<Pixelwert>`
- b) siehe auch `dimension.height=<Pixelwert>`
- c) Zur Abspeicherung der Werte braucht WIHU Schreibrechte bzw. -möglichkeiten für die entsprechende Datei und das entsprechende Medium.

4.2.7.6 OsdText=<Text>

Zweck:

Ändert die zu verwendende Textmeldung, welche bei Verwendung von `/AutoExit=<Sekunden>` oder `/AutoInstall=<Sekunden>` am Bildschirm/Desktop die noch verbleibende Restzeit des Countdowns in Sekunden anzeigt.

Beispiel:

```
OsdText=noch %d Sekunden bis WIHU selbständig fortfährt
```

Anmerkungen & Hinweise:

- `%d` steht als Platzhalter für die noch verbleibenden Sekunden
- Der Countdown kann mittels `/beep=<Sekunden>` auch akustisch unterlegt werden
- Der Standardtext lautet: „WIHU wartet noch %d Sekunden auf Benutzereingaben“

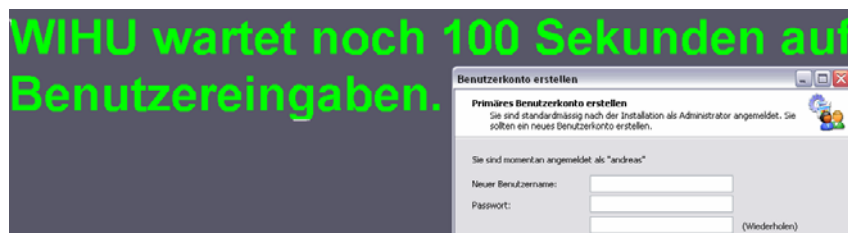


Abbildung 12: Bildschirmanzeige der verbleibenden Countdowndauer

4.2.7.7 AutoExit=<Sekunden>

Analoge Anwendung zum Befehlszeilenparameter `/AutoExit=<Sekunden>`.

4.2.7.8 AutoInstall=<Sekunden>

Analoge Anwendung zum Befehlszeilenparameter `/AutoInstall=<Sekunden>`.

4.2.7.9 beep=<Sekunden>

Analoge Anwendung zum Befehlszeilenparameter `/beep=<Sekunden>`.

4.2.7.10 RestartWait=<Sekunden>

Analoge Anwendung zum Befehlszeilenparameter /RestartWait=<Sekunden>.

4.2.7.11 NoRestartChange=1

Analoge Anwendung zum Befehlszeilenparameter /NoRestartChange.

4.2.7.12 NoCancel=1

Analoge Anwendung zum Befehlszeilenparameter /NoCancel.

4.2.7.13 SkipRestart=1

Analoge Anwendung zum Befehlszeilenparameter /SkipRestart.

4.2.7.14 SkipSettings=1

Analoge Anwendung zum Befehlszeilenparameter /SkipSettings.

4.2.7.15 SkipSoftware=1

Analoge Anwendung zum Befehlszeilenparameter /SkipSoftware.

4.2.7.16 Admin=<Name>

Analoge Anwendung zum Befehlszeilenparameter /Admin=<Name>.

4.2.7.17 AdminPwd=<Paßwort>

Analoge Anwendung zum Befehlszeilenparameter /AdminPwd=<Paßwort>.

4.2.7.18 User=<Name>

Analoge Anwendung zum Befehlszeilenparameter /User=<Name>.

4.2.7.19 UserPwd=<Paßwort>

Analoge Anwendung zum Befehlszeilenparameter /UserPwd=<Paßwort>.

4.2.7.20 AutoLogon=1

Analoge Anwendung zum Befehlszeilenparameter /AutoLogon.

4.2.7.21 Computer=<Name>

Analoge Anwendung zum Befehlszeilenparameter /Computer=<Name>.

4.2.7.22 Workgroup=<Name>

Analoge Anwendung zum Befehlszeilenparameter /Workgroup=<Name>.

4.2.7.23 Owner=<Name>

Analoge Anwendung zum Befehlszeilenparameter /Owner=<Name>.

4.2.7.24 Org=<Name>

Analoge Anwendung zum Befehlszeilenparameter /Org=<Name>.

4.2.7.25 Log=<Datei>

Analoge Anwendung zum Befehlszeilenparameter /Log=<Datei>.

4.2.7.26 verbose=[0|1|2|3]

Analoge Anwendung zum Befehlszeilenparameter /verbose=[0 | 1 | 2 | 3].

5 Unterstützung des Programmmentwicklers

Der Programmentwickler Benjamin Kalytta hat sich viel Mühe bei der Entwicklung von WIHU gegeben, insbesondere ist er immer wieder zügig auf die unterschiedlichsten Wünsche der Programmnutzer während der Programmierphase eingegangen. Ein Blick in den entsprechenden Forumsthread bei MSFN beweist dies sicherlich. Falls jemand diese Anstrengungen belohnen möchte, kann er dies zum einen gerne mit einer kleinen Dankesmail an xmaster@gmx.net oder über das entsprechende Homepageformular tun, genauso gern wird aber auch eine handfestere Belohnung entgegengenommen. ;) Wer will, kann dies jederzeit durch eine kleine und unkomplizierte Überweisung an ...

Name: Benjamin Kalytta

BIC: GENODE51KRE

IBAN: DE83 5609 0000 0002 2928 44

... erledigen! ☺ Benjamin freut sich immer darüber und sagt Vielen Dank dafür!

6 Verweise und Quellen

Hier finden Sie ein paar Links mit weiteren Informationen rund um das Programm und das Thema „Unbeaufsichtigte Installationen“.



<http://www.kalytta.com/>

Homepage des Programmentwicklers



<http://unattended.msfm.org/>

englischer Leitfaden für unbeaufsichtigte Installationen
(dort auch Links zu Übersetzungen des Leitfadens)



<http://www.msfm.org/board/>

Benutzerforum, das sich in einem Teil rund um Fragen und Probleme bei unbeaufsichtigten Installationen kümmert; riesige Wissensquelle! Mit den Ideen der Benutzer dort wurde WIHU weiterentwickelt.

TIP: Erst suchen, dann lesen, und zuletzt im Forum fragen! ;)



<http://www.msfm.org/board/index.php?showforum=90>

Unterforum innerhalb von MSFM, in dem sich alles um WIHU dreht



<http://www.ondemandsoftware.com/FREELE2003/>

kostenlose Software zum Erstellen von Installationspaketen

7 Rechtliches

7.1 zum Computerprogramm

Der Entwickler des Computerprogramms „Windows Installation Helper Utility“ (Abk.: WIHU) weist darauf hin, daß die Benutzung freiwillig und ohne Gewähr erfolgt. Eine Haftung für Verluste oder Schäden jeglicher Art wird ausgeschlossen. Die Benutzung erfolgt auf eigene Gefahr.

Eine kommerzielle Verwertung ist untersagt. Zudem muß der Name des Originalentwicklers im Programm für jeden Benutzer abrufbar sein.

7.2 zur Anleitung

Der Entwickler von WIHU hat der Veröffentlichung dieser Anleitung offiziell zugestimmt und den Inhalt vor Veröffentlichung verifiziert.

Der Autor dieser Anleitung weist darauf hin, daß die Benutzung freiwillig und ohne Gewähr erfolgt. Eine Haftung für Verluste oder Schäden jeglicher Art wird ausgeschlossen. Die Benutzung erfolgt auf eigene Gefahr.

Die Anleitung kann jederzeit weitergegeben werden, auch getrennt von WIHU, solange dies jedoch komplett und nicht nur in Auszügen erfolgt. Eine kommerzielle Verwertung ist verboten, die Weitergabe und Benutzung der Anleitung muß in jedem Fall ohne besonderes Entgelt erfolgen können.

Änderungen am Text, dem Inhalt und der Form der Anleitung sind ausschließlich dem Autor gestattet. Konstruktive Anregungen oder Wünsche können dem Autor natürlich jederzeit vorrangig im entsprechenden Internetforum übermittelt werden.

Eine Übersetzung in andere Sprachen ist jederzeit erlaubt, solange in der (auch auszugsweisen) Übersetzung expliziter und namentlicher Bezug auf diese Originalvorlage genommen wird sowie die sonstigen Urheberrechte gewahrt bleiben.