

LiveSystem pro PE Builder

User Manual (v.15)

Version 1.1.0.6 und höher
© www.kare-net.de 2011-2017

Erstellungsdatum: 3. Aug. 2017

Inhaltsverzeichnis

INHALTSVERZEICHNIS	2
1. ERSTE SCHRITTE	4
1.1. ZWECK VON WINDOWSPE	4
1.2. ERSTELLUNG VON WINDOWSPE	4
1.3. UNTERSTÜTZTE HOST-COMPUTERSYSTEME	4
1.4. UNTERSTÜTZTE PE-SYSTEME	4
2. BEFEHLS-SYNTAX	5
2.1. SYSTEMBEFEHLE	5
Convert	5
Echo	5
Exit	6
Extract	6
Halt	6
Math	6
Message	7
Pack	7
Return	7
Run	8
Set	8
Setmacro	8
SetMvar	9
SYSTEM	9
Wait	11
Wim-Befehle	11
Update	12
2.2. PROGRAMMIERBARE BEDINGUNGEN (IF-BEDINGUNGEN)	12
If,...,Then	12
If,...,Then,...,Else-Block	13
2.3. ORDNER-, DATEI- UND INTERFACE-BEFEHLE	14
2.3.1. Ordner	14
DirCopy	14
DirRemove	14
DirMove	14
DirMake	15
2.3.2. Dateien	15
2.3.2.1. Dateibefehle	15
FileCopy	15
FileDelete	15
FileRename	16
FileCreateBlank	16
FileRead	16
FileSetAttrib	16
CopyFile	17
CopyOrExpand	17
Expand	18
GetDistLang	18
ReqFile	19
Require_File	19
Retrieve	19
Search File	20
SetReqFile	20
ShellExecute	21
Start	21
2.3.2.2. Textbearbeitung	21
TXTAddLine	21
TXTReplace	22

StrFormat	22
2.3.2.3. INI-Bearbeitung	22
IniWrite	22
IniWriteSection	23
IniDeleteSection	23
IniRead	23
IniReadPos	23
IniRenameSection	24
2.3.3. Interface-Befehle	24
Interface,Get	24
Interface,GetIndex	25
Interface,Set	25
Interface,State	26
Interface,Update	26
2.4. BEARBEITUNG DER REGISTRY	26
RegLoad	26
RegUnLoad	27
RegRead	27
RegWrite	27
RegDelete	29
2.5. BEARBEITUNG VON ZEICHENKETTEN (STRINGS)	29
String LEFT/RIGHT	29
String TLeft/TRight	29
String FIND	30
String LEN	30
String MID	30
String REPLACE	30
String SPLIT	31
2.6. NETZWERKBEFEHLE	31
WebGet	31
2.7. SCHLEIFENBEFEHLE	32
2.7.1. For / Next - Schleife	32
2.7.2. ExitLoop	33
2.8. VARIABLEN (INTERN)	33
2.9. VARIABLEN (PROJEKT)	34
2.10. SYSTEM,SETPEINIT	36
2.11. SONDERZEICHEN	38
2.11. SHORTCUTS	38
2.12. MAKROS	39
3. SKRIPTSTEUERUNG	40
3.1. STANDARD-ABSCHNITTE EINES SKRIPTS	40
3.1.1. Abschnitt [Main]	40
3.1.2. Abschnitt [Interface]	41
3.1.3. Abschnitt [Process]	41
3.2. AKTIVE SKRIPTSTEUERUNG	42
3.2.1. Auswahl der Windows-Quelle	42
3.2.2. Script-Select	42

Anmerkung:

Optionale Befehle sind in der Schreibweise

[Befehl] bzw. *[<Befehl>]* dargestellt

1. Erste Schritte

1.1. Zweck von WindowsPE

Windows PE (Windows PreinstallationEnvironment) ist hervorragend für folgende Einsatzbereiche geeignet:

- ↪ Computerreparatur-Optionen
- ↪ Durchführung von Datenrettung
- ↪ Vorbereitung von Neuinstallationen des Betriebssystems
- ↪ Entfernung von Viren
- ↪ Wiederherstellung persönlicher Daten (Dokumente und Bilder)
- ↪ Rückspielung von Images (Abbilddateien)

[zurück zum Inhaltsverzeichnis](#)

1.2. Erstellung von WindowsPE

Videoanleitung im Forum **LiveSystem-pro.de**:

<http://www.livesystem-pro.de/showthread.php?92-Videoanleitung-1-quot-LiveSystem-pro-quot-PE-Builder>

[zurück zum Inhaltsverzeichnis](#)

1.3. Unterstützte Host-Computersysteme

Derzeit werden folgende Systeme unterstützt:

- | | | |
|---|-----------------|------------------------------------|
| ↪ | Windows XP/2003 | ab ServicePack 3 (32/64-Bit) |
| ↪ | Windows Vista | mit allen ServicePacks (32/64-Bit) |
| ↪ | Windows 7 | mit allen ServicePacks (32/64-Bit) |
| ↪ | Windows 8/8.1 | (32/64-Bit ohne ServicePack) |
| ↪ | Windows 10 | (32/64-Bit ohne ServicePack) |

[zurück zum Inhaltsverzeichnis](#)

1.4. Unterstützte PE-Systeme

LiveSystempro ist für 4 Projekte verfügbar:

- | | | |
|---|--------|--|
| ↪ | WinXPE | Windows XP, Windows 2003 (32/64-Bit) |
| ↪ | Win7PE | Windows Vista (32/64-Bit)*
Windows 2008 (32/64-Bit)*
Windows 7 (32/64-Bit)*, |

		Windows 2010 (32/64-Bit)* * = alle ServicePacks
⇒	Win8PE	Windows 8 (32/64-Bit) Windows 8.1 (32/64-Bit) Windows 2010 R2 (32/64-Bit), Windows 8.1 Update 1 (32/64-Bit)
⇒	Win10PE	Windows 10 (32/64-Bit)

Alle Projekte sind in LiveSystempro integriert und getrennt auswählbar.

[zurück zum Inhaltsverzeichnis](#)

2. Befehls-Syntax

2.1. Systembefehle

Convert

Konvertiert dezimale und hexadezimale Werte

Syntax:

Convert, <%Variable%>, <Methode>, <Wert>

Methode:	-	<i>Hex2Dec</i>	konvertiert hexadezimalen Wert in dezimalen Wert
		<i>Dec2Hex</i>	konvertiert dezimalen Wert in hexadezimalen Wert
		<i>Dec2chr</i>	konvertiert dezimalen Wert in Ascii-Zeichen (Unicode wird unterstützt)

Beispiele:

Convert, %Var%, Hex2Dec, 401a
Weist der Variablen %Var% den Wert **16410** zu.

Convert, %Var%, Dec2Hex, 16410
Weist der Variablen %Var% den Wert **401a** zu.

Convert, %Var%, Dec2chr, 65
Weist der Variablen %Var% den Wert **A** zu.

[zurück zum Inhaltsverzeichnis](#)

Echo

Gibt einen Nachrichtentext aus

Syntax:

Echo, <Text>

Beispiel:

Echo, Skript wird ausgeführt

[zurück zum Inhaltsverzeichnis](#)

Exit

Beendet die Ausführung des aktuellen Skripts

Syntax:

Exit

Beispiel:

If,%Arch%,x64,Then,Exit

[zurück zum Inhaltsverzeichnis](#)

Extract

Extrahiert einen Anhang aus einer Skript-Datei

Syntax:

Extract.<Skript-Datei>,<Datei>,<Zielpfad>

Beispiel:

Extract,%ScriptFile%,ReBoot-x86.dll,%Target_Sys%\ReBoot.dll

Extrahiert aus dem aktuellen Skript die Datei *ReBoot-x86.dll* und speichert sie als *ReBoot.dll* ins Verzeichnis *System32*.

[zurück zum Inhaltsverzeichnis](#)

Halt

Stoppt die Ausführung des Builds

Syntax:

Halt,<Text>

Text - beliebiger Text (optional)

Beispiel:

Halt,Abbruch durch Benutzer

[zurück zum Inhaltsverzeichnis](#)

Math

Führt eine mathematische Berechnung aus

Syntax:

Math,<%Variable%>,<Wert>,<PARAMETER>

%Variable% - beliebiger Name
Wert - mathematische Berechnung
Parameter (optional) - Round_Up / Round_Down

Beispiele:

*Math,%NeueVar%,45+(7*4)*

Weist der Variablen *%NeueVar%* den Wert **73** zu (Summe aus 45 + 28)

Math,%NeueVar%,16.9-10,Round_Down

Weist der Variablen *%NeueVar%* das abgerundete Ergebnis **6** anstelle von **6.9** zu

[zurück zum Inhaltsverzeichnis](#)

Message

Zeigt eine Nachrichtenbox an

Syntax:

Message, <Text>[, <Titel>], [<Timeout>]

<i>Text</i>	-	Text der Nachrichtenbox
<i>Titel</i>	-	Titel der Nachrichtenbox
<i>Information</i>		Anzeige eines Informationssymbols
<i>Confirmation</i>		Anzeige eines Fragezeichens
<i>Error</i>		Anzeige eines Fehlersymbols
<i>Question</i>		Anzeige von JA/NEIN Buttons
<i>Warning</i>		Anzeige eines Warnsymbols
<i>Timeout</i>	-	Anzeigedauer der Box in Sekunden

Beispiele:

Message, Hallo Welt, Information, 10

Message, Fortfahren?, Question

If, %Exitcode%, =, Yes, Then, Message, Taste JA wurde gedrückt

If, %Exitcode%, =, No, Then, Message, Taste NEIN wurde gedrückt

[zurück zum Inhaltsverzeichnis](#)

Pack

Packt oder entpackt eine Datei

Syntax Packbefehl:

Pack, <Compress>, <Zielordner>, <Datei>

Komprimiert werden können nur Dateien mit Endung `„.zip“`

Syntax Entpackbefehl:

Pack, <Decompress>, <Dateipfad>, <Ordner>[, <Filter>]

Entpackt werden können nur Dateien mit Endung `„.zip“` und `„.7z“`.

Durch Eingabe des Zeichens `*` schaltet der Filter auf Teilsuche um (ähnlich einer Wildcard-Funktion)

Beispiele:

pack, compress, C:\Test, C:\Test.zip

Komprimiert den Ordner `Test` zur ZIP-Datei `Test.zip`.

pack, decompress, C:\RegConfig.7z, C:\MeinOrdner, Win8x86 productoptions.reg

Entpackt aus der 7z-Datei `RegConfig.7z` nur die Datei `Win8x86` und `productoptions.reg`.

*pack, decompress, C:\Test.7z, C:\Test, *.pdf*

Entpackt aus der 7z-Datei `Test.7z` alle Dateien mit Endung `pdf`.

[zurück zum Inhaltsverzeichnis](#)

Return

Beendet den aufgerufenen Abschnitt eines Skripts

Syntax:

Return

Beispiel:

Run, %ScriptFile%, MeinAbschnitt

Message, %Arch%

```
[MeinAbschnitt]
If,%Arch%,x64,Then,Return
Message,Architektur stimmt
```

Die Nachrichtenbox **Architektur stimmt** wird nur bei Architektur x86 angezeigt.

[zurück zum Inhaltsverzeichnis](#)

Run

Führt einen Abschnitt im angegebenen Skript aus

Syntax:

Run,<ScriptPfad>,<Abschnitt>[,<PARAMETER>]

- | | | | |
|-------------------|---|----|--|
| ScriptPfad | - | a. | kompletter Skriptpfad |
| | | b. | bei Aufruf eines Abschnittes im gleichen Skript genügt %ScriptFile% |
| Abschnitt | - | | auszuführender Abschnitt im Skript |
| PARAMETER | - | | 9 Parameter können übergeben werden (#1 bis #9) |

Beispiel:

```
Run,%ScriptFile%,MeinAbschnitt,Hallo
```

```
[MeinAbschnitt]
Message,#1,Information
```

Dieses Beispiel öffnet eine Nachrichtenbox mit dem Text **Hallo**.

[zurück zum Inhaltsverzeichnis](#)

Set

Weist einer Variablen einen Wert zu

Syntax:

Set,<%Variable%>,<Wert>[,<PERMANENT>]

- | | | |
|------------------|---|---|
| Wert | - | beliebiger Wert oder Pfad |
| PERMANENT | - | gültig für das gesamte Build, wird in project.cfg im Abschnitt [variables] gespeichert |

Beispiele:

```
Set,%NeueVar%,myPE
Weist der Variablen %NeueVar% den Wert myPE zu.
```

```
Set,%NeueVar%,myPE,PERMANENT
Weist der Variablen %NeueVar% den Wert myPE zu, gültig für das gesamte Build.
```

[zurück zum Inhaltsverzeichnis](#)

Setmacro

Erzeugt ein Macro

Syntax:

Setmacro,<Macroname>,<Befehl>[,<PERMANENT>]

- | | | |
|------------------|---|--------------------------------|
| Macroname | - | beliebiger Name ohne %% |
| Befehl | - | jeder gültige Builder-Befehl |

PERMANENT - gültig für das gesamte Build, wird in *project.cfg* im Abschnitt [Macros] gespeichert

Beispiel:

Setmacro,require_file,Run,%ProjectDir%\Macro.srt,Process-ReqSys32,#1,PERMANENT
Erzeugt das Macro *require_file* mit dem Befehl
Run,%ProjectDir%\Macro.srt,Process-ReqSys32,#1,PERMANENT

[zurück zum Inhaltsverzeichnis](#)

SetMvar

Erzeugt eine Macrovariable

Syntax:

SetMvar,<%Variable%>,<Wert>

Beispiel:

SetMvar,%MeineVariable%,myPE
Erzeugt eine Macrovariable mit dem Wert *myPE*.
Die Macrovariable ist nur im aktuellen Skript gültig!

[zurück zum Inhaltsverzeichnis](#)

SYSTEM

Befehl für Systemeinstellungen

Syntax:

System,<Systembefehl>,<Systemparameter>

Systembefehl:	<i>Error</i>	schaltet Fehlermeldungen bei Ereignissen ein / aus
	<i>FreeRam</i>	Anzeige des freien Arbeitsspeichers
	<i>TotalRam</i>	Anzeige des gesamten Arbeitsspeichers
	<i>GetDate</i>	Anzeige von Datum und Zeit
	<i>GetEnv</i>	liest Umgebungsvariablen aus: <i>Temp</i> <i>ProgramFiles</i> <i>Number_Of_Processors</i>
	<i>GetFreeDrive</i>	Zeigt den nächsten verfügbaren freien Laufwerksbuchstaben an
	<i>GetFreeSpace</i>	Zeigt den verfügbaren freien Speicher an
	<i>GetDriveType</i>	Zeigt den Datenträgertyp an
	<i>Log</i>	Schaltet das Schreiben der LOG ein / aus
	<i>ProcessMode</i>	Versetzt die GUI in den Prozess-Modus
	<i>SetPEinit</i>	Setzt einen Befehlseintrag für <i>Peinit.ini</i> (siehe Abschnitt 2.10)
Systembefehl (nur für x64-Systeme):	<i>FileRedirect</i>	Schaltet die FileRedirection (Wow64) auf 64-Bit-Systemen ein oder aus
	<i>RegRedirect</i>	Schaltet die RegRedirection (Wow64) auf 64-Bit-Systemen ein oder aus
Systemparameter:	bei <i>Error</i>	On Off
	bei <i>FileRedirect</i>	On Off
	bei <i>FreeRam</i>	%Variable%
	bei <i>TotalRam</i>	%Variable%
	bei <i>GetDate</i>	Format - %yyyy - Jahr (4 Stellen) %yy - Jahr (2 Stellen) %mm - Monat (2 Stellen) %dd - Tag (2 Stellen) %hh - Stunde (2 Stellen)

	%ii	- Minute (2 Stellen)
	%ss	- Sekunde (2 Stellen)
bei <i>GetFreeDrive</i>	%Variable%	
bei <i>GetFreeSpace</i>	%Variable%	
bei <i>GetDriveType</i>	Rückgabewert	
	1 -	kein Root
	2 -	mobiler Datenträger
	3 -	Festplatte
	5 -	CDROM
bei <i>SetPEinit</i>	%Variable%, [<Abschnitt>]	
bei <i>Log</i>	On Off	
	<i>Log, On, 1</i>	Schaltet das Schreiben der LOG ein und löscht eine vorhandene LOG
bei <i>RegRedirect</i>	64 Off	

Beispiele:

System, FILEREDIRECT, On

System, FILEREDIRECT, Off

System, REGREDIRECT, 64

System, REGREDIRECT, Off

System, GETDATE, %yyyy, %Var%

Weist der Variablen *%Var%* das aktuelle Jahr zu (mit 4 Stellen)

System, GETFREEDRIVE, %FreeLetter%

Weist der Variablen *%FreeLetter%* den nächsten verfügbaren freien Laufwerksbuchstaben zu.

System, GETFREESPACE, %Drive%, %Size%

Beispiel:

Set, %Txt%,

For, %Var%, 65, 90

Convert, %Drive%, Dec2chr, %Var%

System, GetFreeSpace, %Drive%:\, %Size%

If, %Size%, >, 0, Then, Set, %Txt%, %Txt% %Drive% - %Size%# \$x

Next

String, %Txt%, TRight, %Txt%, 2

Weist der Variablen *%Txt%* alle Partitionen mit dem jeweiligen freien Speicher zu.

Anmerkung:

65 und **90** sind die ASCII-Werte der Buchstaben **A** und **Z**.

System, GETDRIVETYPE, %Drive%, %Type%

Beispiel:

Set, %Txt%,

For, %Var%, 65, 90

Convert, %Drive%, Dec2chr, %Var%

System, GetDriveType, %Drive%:\, %Type%

If, %Type%, =, 3, Then

System, GetFreeSpace, %Drive%:\, %Size%

If, %Size%, >, 0, Then, Set, %Txt%, %Txt% %Drive% - %Size% MB# \$x

End

Next

String, %Txt%, TRight, %Txt%, 2

Weist der Variablen *%Txt%* den freien Speicher des gewählten Typs zu.

System, GetEnv, %Var%, ProgramFiles

Weist der Variablen *%Var%* den vollständigen Pfad von *%ProgramFiles%* zu.

System, GetEnv, %Var%, Temp

Weist der Variablen *%Var%* den vollständigen Pfad des Ordners *%Temp%* zu.

(z.B.: C:\Users<Benutzername>\AppData\Local\Temp

System, GetEnv, %Var%, Number_Of_Processors

Weist der Variablen *%Var%* die Anzahl der Prozessoren zu.

[zurück zum Inhaltsverzeichnis](#)

Wait

Pausiert das Skript um die angegebenen Millisekunden

Syntax:

Wait, <Millisekunden>

Beispiel:

Wait, 1000

(1 Sekunde = 1000 Millisekunden)

[zurück zum Inhaltsverzeichnis](#)

Wim-Befehle

Befehle zur Bearbeitung von Image-Dateien

Syntax:

Wim, Create, <PackMethode>, <Verzeichnis>, <WimDatei>

PackMethode - none (ungepackt)
LZMS (LZMS-Kompression)

Wim, Extract, <Modus>, <WimDatei>, <%WimIndex%>, <Parm1>[, <...>]

Modus - **File** extrahiert Datei oder Verzeichnis
FileList extrahiert Dateien und Verzeichnisse
aus einer Liste

%WimIndex% - **Image** extrahiert ein komplettes Image
Numerischer Wert beginnend mit 1
(Systemvariable)

Parm1 - bei **File** *<Datei/Verzeichnis>, <Zielverzeichnis>*
bei **FileList** *<FileListe>, <Zielverzeichnis>*
bei **Image** *<Zielverzeichnis>*

Wim, Info, <WimDatei>, <%WimIndex%>, <SkriptDatei>

%WimIndex% - Numerischer Wert beginnend mit 1
(Systemvariable)

Wim, InfoEdition, <WimDatei>, <SkriptElement>, SkriptDatei>

SkriptElement - unterstützt wird nur ‚pScrollBar‘

Pfad zur Image-Datei:

Win7PE, Win8PE+Win10PE: Verwendung der Variablen **%Installwim%** (siehe Abschnitt 2.9)

Beispiele:

Wim, Create, lzms, %TargetDir%, %Target_CD%\sources\Boot.wim
Komprimiert die Datei Boot.wim mit Kompressionsmethode *lzms*.

Wim, info, %SourceDir%\sources\install.wim, %WIM_Index%, %ProjectDir%\project.cfg
Liest den *WIM-Index* aus der Datei *install.wim* und schreibt ihn in *project.cfg*.

Wim, infoEdition, %SourceDir%\sources\install.wim, pScrollBar1, %ProjectDir%\Build\0-PreConfig.srt
Schreibt Informationen aller Images aus *install.wim* in die Datei *0-PreConfig.srt*.

Wim, Extract, File, %SourceDir%\sources\install.wim, %WIM_Index%, #1, %InstallSRC%
Extrahiert die Datei mit Parameter *#1* nach *%InstallSRC%*.

Wim, Extract, Image, %SourceDir%\sources\install.wim, %WIM_Index%, %InstallSRC%
Extrahiert aus *Install.wim* den Inhalt eines bestimmten WIM-Indexes nach *%InstallSRC%*.

*Wim,Extract,FileList,%SourceDir%\Sources\Install.wim,%WIM_Index%,%ProjectTemp%\Instwim.txt,%Basedir%\Wim\OS%%SourceE
dition%%arch%%SP%%Distlang%-I\%WIM_Index%*

Extrahiert auf Grund des Inhalts von *Instwim.txt* Dateien aus der Install.wim in den Zielordner.

Wim,Extract,Image,f:\boot.wim,2,f:\test5

Extrahiert boot.wim in den Ordner *test5*.

[zurück zum Inhaltsverzeichnis](#)

Update

Aktualisiert den Variablenspeicher

Syntax:

Update,<Parameter>[,ScriptDir]

Parameter:

<i>ProjectVar</i>	alle Variablen aus <i>project.cfg</i> , Abschnitt <i>[variables]</i> werden in den Variablenspeicher geladen
<i>InterfaceVar</i>	Variablen im Abschnitt <i>[Interface]</i> eines beliebigen Skripts werden in den Variablenspeicher geladen
<i>Project</i>	alle Skripte werden neu eingelesen
<i>Project,ScriptDir</i>	Skriptverzeichnis wird neu eingelesen

Beispiele:

Update,ProjectVar

Update,InterfaceVar

Update,Project

Update,Project,Apps\%pTextBox111%.srt

[zurück zum Inhaltsverzeichnis](#)

2.2. Programmierbare Bedingungen (If-Bedingungen)

If,...,Then

Syntax:

If,<Bedingung>,<%Variable%>,Then,<Befehl>

Negative Syntax:

If,Not,<Bedingung>,<%Variable%>,Then,<Befehl>

Verfügbare Bedingungen:

ExeRunning
ExistFile
ExistDir
ExistSection
ExistRegKey
ExistWebFile
SrtChecked
= (gleich)
< (kleiner)
> (größer)

Beispiele:

If,%Var%=>,1,Then,Message,Wert ist gleich 1

Die Meldung erfolgt nur, wenn der Wert von *%Var%* gleich *1* ist.

If,3,>,2,Then,Message,3 ist größer als 2

Die Meldung erfolgt immer, da *3* größer als *2* ist.

If,ExistFile,%ScriptDir%\test.exe,Then,Message,Datei existiert

Die Meldung erfolgt nur, wenn *test.exe* existiert.

If,ExistDir,C:\Programme,Then,Message,Verzeichnis ist vorhanden
Die Meldung erfolgt, wenn das Verzeichnis **C:\Programme** vorhanden ist.

If,Not,ExistDir,C:\Intel,Then,Message,Verzeichnis ist nicht vorhanden
Die Meldung erfolgt, wenn das Verzeichnis **C:\Intel NICHT** vorhanden ist.

If,ExistRegKey,HKLM,PE-SYSTEM\ControlSet001\Services\v8042prt\Parameters,EnableWheelDetection,Then,Message,Wert existiert
Die Meldung erfolgt, wenn im RegSchlüssel der Wert **EnableWheelDetection** existiert.

If,ExistSection,%target_win%\TXTSETUP.SIF,SourceDiskFiles,Then,Message,Abschnitt ist vorhanden
Die Meldung erfolgt, wenn der Abschnitt **SourceDiskFiles** in der Datei **TXTSETUP.SIF** vorhanden ist.

If,ExistRegKey,HKLM,PE-SYSTEM\ControlSet001\Services\v8042prt\Parameters,Then,Message,Schlüssel existiert
Die Meldung erfolgt, wenn der komplette RegKey existiert.

If,EXISTREGKEY,HKLM,PE-SYSTEM\ControlSet001\Services\v8042prt\Parameters,,Then,Message,vorhanden
Die Meldung erfolgt, wenn im RegSchlüssel der Wert (**Standard**) eingetragen ist.

If,SrtChecked,Build\8-HW-pnp.srt,Then,Message,Skript ist aktiviert
Diese Bedingung prüft, ob das Skript aktiviert ist.

If,ExistWebFile,http://www.xyz.com/xyz.zip,Then,Message,Datei ist verfügbar
Die Meldung erfolgt nur, wenn **xyz.zip** verfügbar ist.

If,ExeRunning,vmware.exe,Then,Message,VMware wird bereits ausgeführt
Die Meldung erfolgt nur, wenn **VMware** bereits ausgeführt wird.

Hinweis zur mehrzeiligen Schreibweise:

Auch diese Schreibweise ist möglich:

```
If,%Var%,=,1,Then
    Message,Wert ist gleich 1
End
```

Bei dieser Schreibweise ist der END-Befehl **IMMER** erforderlich !

[zurück zum Inhaltsverzeichnis](#)

If,....,Then,....,Else-Block

Alle oben angeführten Bedingungen können auch als **ENTWEDER – ODER Block** verwendet werden

Syntax:

```
If,<Bedingung>,<%Variable%>,Then
    <Befehl_#1>
Else
    <Befehl_#2>
End
```

Beispiele:

```
If,ExistDir,C:\Intel,Then
    Message,Verzeichnis ist vorhanden
Else
    Message,Verzeichnis ist nicht vorhanden
End
```

Diese Bedingung gibt je nach Ergebnis **entweder** die erste **oder** die zweite Meldung aus.

```
If,%Var%,=,1,Then
    Message,Wert ist gleich 1
Else
    Message,Wert ist ungleich 1
End
```

Gleiches Resultat wie in obigem Beispiel.

Die Kombination mit **NOT** ist ebenfalls möglich.

Beispiel:

```
If,Not,%Var%,1,Then
    Message,Wert ist ungleich 1
Else
    Message,Wert ist gleich 1
End
```

[zurück zum Inhaltsverzeichnis](#)

2.3. Ordner-, Datei- und Interface-Befehle

2.3.1. Ordner

DirCopy

Kopiert den Inhalt eines Ordners in einen anderen Ordner

Syntax:

DirCopy,<Quellordner>,<Zielordner>

Beispiel:

DirCopy,C:\MeinOrdner,C:\MeinAndererOrdner
Kopiert den gesamten Inhalt von *C:\MeinOrdner* nach *C:\MeinAndererOrdner*.
Wenn der Zielordner nicht existiert, wird er erstellt.

[zurück zum Inhaltsverzeichnis](#)

DirRemove

Löscht ein Verzeichnis

Syntax:

DirRemove,<ZuLöschendeDateien/Ordner>[,Parameter]

Parameter:	<i>0</i>	Standardwert, löscht alle Dateien im angeführten Ordner
	<i>1</i>	löscht ein Verzeichnis samt eventueller Unterverzeichnisse
	<i>2</i>	löscht nur ein existierendes Verzeichnis

Beispiele:

DirRemove,C:\MeinOrdner,1
Löscht den gesamten Ordner *C:\MeinOrdner* inklusive eventueller Unterverzeichnisse.

DirRemove,C:\MeinOrdner,0
Löscht alle Dateien im Ordner *C:\MeinOrdner*.

DirRemove,C:\MeinOrdner\MeinUnterdordner,0
Löscht die Dateien im Unterdordner *MeinUnterdordner*.

DirRemove,C:\MeinOrdner\MeinUnterdordner,3
Löscht die Dateien im Unterdordner *MeinUnterdordner* falls der Ordner existiert.
(3=Kombination von 1 und 2)

[zurück zum Inhaltsverzeichnis](#)

DirMove

Verschiebt ein Verzeichnis

Syntax:

DirMove,<ZuVerschiebenderOrdner>,<Zielordner>

Beispiel:

DirMove,C:\MeinOrdner,C:\MeinAndererOrdner

Verschiebt den **Inhalt** von *C:\MeinOrdner* nach *C:\MeinAndererOrdner*.
Der Ursprungsordner wird gelöscht.

[zurück zum Inhaltsverzeichnis](#)

DirMake

Erstellt ein Verzeichnis

Syntax:

DirMake,<NeuerOrdner>

Beispiel:

DirMake,C:\MeinOrdner

Erstellt den neuen Ordner *C:\MeinOrdner*.

[zurück zum Inhaltsverzeichnis](#)

2.3.2. Dateien

2.3.2.1. Dateibefehle

FileCopy

Kopiert eine einzelne Quelldatei in einen vorgegebenen Bestimmungsort

Syntax:

FileCopy,<QuellDatei>,<Zielordner>

Beispiel:

FileCopy,C:\MeinOrdner\MeineDatei.txt,C:\MeinAndererOrdner

Kopiert die Datei *MeineDatei.txt* von *C:\MeinOrdner* nach *C:\MeinAndererOrdner*.
Wildcards (*.*) werden unterstützt.

[zurück zum Inhaltsverzeichnis](#)

FileDelete

Löscht eine einzelne Datei

Syntax:

FileDelete,<Datei>[,Parameter]

Parameter: 1 löscht nur existierende Dateien

Beispiele:

FileDelete,C:\MeinOrdner\MeineDatei.txt

Löscht die angegebene Datei.

FileDelete,C:\MeinOrdner\MeineDatei.txt,1

Löscht die angegebene Datei falls die Datei existiert.

[zurück zum Inhaltsverzeichnis](#)

FileRename

Umbenennung einer vorhandenen Datei

Syntax:

FileRename, <AlterDateiname>, <NeuerDateiname>

Beispiel:

FileRename, C:\MeinOrdner\MeineDatei.txt, C:\MeinOrdner\MeineNeueDatei.txt
Umbenennung der Datei *MeineDatei.txt* zu *MeineNeueDatei.txt*

[zurück zum Inhaltsverzeichnis](#)

FileCreateBlank

Erstellt eine leere Datei (eine bestehende gleichnamige Datei wird gelöscht)

Syntax:

FileCreateBlank, <NeueDatei>[, <Kodierung>]

Zeichensatz	(optional)	-	ANSI
			Unicode
			UTF-8

Beispiele:

FileCreateBlank, C:\MeinOrdner\test.txt
Erstellt im angegebenen Ordner die leere Datei *test.txt*

FileCreateBlank, C:\MeinOrdner\test.txt, utf8
Erstellt im angegebenen Ordner die leere Datei *test.txt* im Zeichensatz *UTF-8*

[zurück zum Inhaltsverzeichnis](#)

FileRead

Liest den Inhalt einer Datei in eine Variable

Syntax:

FileRead, <%Variable%>, <Datei>[, <Zähler>]

Zähler: - optionaler Parameter (Anzahl der zu lesenden Zeichen)

Beispiele:

FileRead, %Var%, C:\MeinOrdner\test.txt
Liest den gesamten Inhalt von *test.txt* in die Variable *%Var%*.

FileRead, %Var%, C:\MeinOrdner\test.txt, 20
Liest 20 Zeichen des Inhalts von *test.txt* in die Variable *%Var%*.

Mögliche Vorgehensweise, falls Cursor unerwünschterweise am Beginn einer Leerzeile steht:

```
(%Var%=C:\MeinOrdner\test.txt)
String, %Result%, RIGHT, %Var%, 2
If, %Result%, =, #$, Then, String, %Var%, TRIGHT, %Var%, 2
Setzt den Cursor nach den letzten vorhandenen Buchstaben.
```

[zurück zum Inhaltsverzeichnis](#)

FileSetAttrib

Ändert die Attribute einer Datei

Syntax:

FileSetAttrib,<Datei>,<Attribut>[,<Parameter>]

Verfügbare Attribute:	<i>R</i>	ReadOnly
	<i>A</i>	Archive
	<i>S</i>	System
	<i>H</i>	Hidden
	<i>N</i>	Normal
	<i>O</i>	Offline
	<i>T</i>	Temporary
Parameter:	<i>1</i>	Rekursiv

Wildcards (*) werden unterstützt,
Die Änderung der Attribute erfolgt durch **+** oder **-**

Beispiele:

FileSetAttrib,C:\MeinOrdner\test.exe,+RS
FileSetAttrib,C:\MeinOrdner.doc,+RS*
FileSetAttrib,C:\MeinOrdner.doc,-H*

FileSetAttrib,C:\MeinOrdner\,+H
Ändert das Attribut eines Ordners.

FileSetAttrib,C:\MeinOrdner\,+H,1
Ändert das Attribut des Ordners und inkludierter Unterordner.

[zurück zum Inhaltsverzeichnis](#)

CopyFile

Kopiert eine Datei aus dem Quellordner in den Zielordner.
Dieser Befehl ist geeignet für die Projekte **Win7PE**, **Win8PE** und **Win10PE**.

Hinweis:

Der Befehl **SetReqFile** muss zuvor für Definition der Verzeichnisse u.Parameter gesetzt werden.

Syntax:

CopyFile,<Datei>[,<Modus>]

Modus	0	-	ohne .mui-Datei
	1	-	mit .mui-Datei + Fallback en-US (en-US-Ersatzfunktion)
	2	-	mit .mui-Datei ohne Fallback en-US

Beispiel:

SetReqFile,%boot_sys%,%DistLang%,%Fallback%,%target_sys%
CopyFile,drivers\acpi.sys,1
CopyFile,atmlib.dll
Kopiert die Datei *drivers\acpi.sys* nach *%target_sys%*.
Kopiert die Datei *drivers\acpi.sys.mui* nach *%target_sys%\%DistLang%*.
Kopiert die Datei *atmlib.dll* nach *%target_sys%*.

[zurück zum Inhaltsverzeichnis](#)

CopyOrExpand

Kopiert eine Datei und entpackt sie, falls notwendig:
Wird die Datei nicht gefunden, wird nach Datei-Endung mit Unterstrich oder nach verkürzter Endung gesucht.

Syntax:

CopyOrExpand,<Datei>,<Zielordner>

Beispiel:

CopyOrExpand,%source_win%\SHELL32.DLL,%target_sys%
Kopiert die Datei *SHELL32.DLL* nach *%target_sys%*.

[zurück zum Inhaltsverzeichnis](#)

Expand

Extrahiert Dateien aus einer CAB-Datei

Syntax:

Expand,<CAB-Datei>,<Zielverzeichnis>[,<Filter>]

Filter Extrahiert einzelne oder mehrere Dateien, Trennung erfolgt durch Zeichen :

Das Zielverzeichnis muss physisch vorhanden sein und wird **NICHT** automatisch erstellt.

Beispiele:

Expand,C:\MeinOrdner\MeineDatei.cab,C:\MeinNeuerOrdner
Extrahiert den Inhalt von *MeineDatei.cab* nach *C:\MeinNeuerOrdner*.

Expand,%source_sys%\%SPCabFile%,%target_sys%\drivers,aec.sys:agp440.sys
Extrahiert im Projekt WinXPE aus der CAB-Datei (z.B.: *SP3.cab*) die Dateien *aec.sys* und *agp440.sys* nach *%target_sys%\drivers*.

Expand,%source_sys%\%SPCabFile%,%target_sys%,bthci.dll
Extrahiert im Projekt WinXPE aus der CAB-Datei (z.B.: *SP3.cab*) die Datei *bthci.dll* nach *%target_sys%*.

[zurück zum Inhaltsverzeichnis](#)

GetDistLang

Ergibt bei Eingabe eines Verzeichnisses oder Dateipfads den Verzeichnisnamen zur .besten Sprachdatei. Falls keine Sprachdatei (.mui) existiert, wird ein Leer-String zurückgegeben.

Syntax:

GetDistLang,<%Variable%>,<Verzeichnis/Datei>,<Mui-Liste>[,<Modus>]

Verzeichnis/Datei	-	Verzeichnis oder Mutter-Datei
Mui-Liste	-	Verzeichnisnamen der Mui-Dateien z.B. es-MX:de-DE:en-US (Einträge sind getrennt durch Doppelpunkt)
Modus	-	0 (leeres Ergebnis wird ignoriert) 1 (leeres Ergebnis ergibt Fehlermeldung)

Beispiele:

GetDistLang,%Var%,%boot_sys%\boot,%DistLang%:%FallBack%
Weist der Variablen *%Var%* bei Verwendung eines deutschen Quellmediums den Wert *de-DE* zu.

GetDistLang,%Var%,%boot_sys%\shell32.dll,%DistLang%:%FallBack%
Weist der Variablen *%Var%* bei Verwendung eines deutschen Quellmediums den Wert *de-DE* zu.

GetDistLang,%Var%,%boot_sys%\srms62.dat,%DistLang%:%FallBack%,1
Ergibt eine Fehlermeldung, da für die Datei *srms62.dat* keine .mui-Datei existiert.

[zurück zum Inhaltsverzeichnis](#)

ReqFile

Überprüft, ob die Datei im Ziel existiert: Wenn nicht, wird sie aus dem Quellordner in den Zielordner kopiert. Dieser Befehl ist geeignet für die Projekte **Win7PE**, **Win8PE** und **Win10PE**.

Hinweis:

Der Befehl **SetReqFile** muss zuvor für Definition der Verzeichnisse u.Parameter gesetzt werden.

Syntax:

ReqFile,<Datei>[,Modus]

Modus	0	-	ohne .mui-Datei
	1	-	mit .mui-Datei + Fallback en-US (en-US-Ersatzfunktion)
	2	-	mit .mui-Datei ohne Fallback en-US

Beispiel:

SetReqFile,%Install_sys%,%DistLang%,%Fallback%,%target_sys%

ReqFile,accessibilitycpl.dll,1

ReqFile,control.exe

Kopiert die Datei *accessibilitycpl.dll* nach *%target_sys%*.

Kopiert die Datei *accessibilitycpl.dll.mui* nach *%target_sys%\%DistLang%*.

Kopiert die Datei *control.exe* nach *%target_sys%*.

[zurück zum Inhaltsverzeichnis](#)

Require_File

Überprüft, ob die Datei im Ziel existiert: Wenn nicht, wird sie aus dem Quellordner in den Zielordner kopiert. Dieser Befehl wird im Projekt **WinXPE** benötigt.

Syntax:

Require_File,<Datei>

Beispiel:

Require_File,shell32.dll

Kopiert die Datei *shell32.dll* nach *%target_sys%*.

[zurück zum Inhaltsverzeichnis](#)

Retrieve

Abfrage von speziellen Datei-Informationen

Syntax:

Retrieve,<%Variable%,<Parameter>,<OrdnerOderDatei>,[Param2],[Param3]

Parameter	-	DirSize FileSize FileVersion MD5 FileDate ResStr FileArch	(Ressourcen-String) (nur EXE-Dateien)
Nur bei FileDate	Param2	Modified Created	
Nur bei ResStr:	Param2 Param3	optional [Anzeige der LCID (locale identifiers)] optional [Anzeige der Ressourcennummer]	

Beispiele:

Retrieve,%Var%,DirSize,C:\MeinOrdner

Weist der Variablen **%Var%** die Größe von *MeinOrdner* zu.

Retrieve,%Var%,FileArch,C:\Windows\explorer.exe
Weist der Variablen **%Var%** die Architektur von *explorer.exe* zu.
Es werden **AUSSCHLIESSLICH** EXE-Dateien unterstützt.

Retrieve,%Var%,MD5,C:\boot.ini
Weist der Variablen **%Var%** die MD5-Summe der Datei *boot.ini* zu.

Retrieve,%Var%,ResStr,C:\Windows\system32\shell32.dll, 00000407,21771
Weist der Variablen **%Var%** den Wert *Dokumente und Einstellungen* zu.

Retrieve,%Var%,FileDate,%BaseDir%\Temp\update.srt,Modified
Weist der Variablen **%Var%** das Änderungsdatum der Datei zu.

[zurück zum Inhaltsverzeichnis](#)

Search File

Sucht nach einer Datei. Wildcards werden unterstützt

Syntax:

Search,<%Variable%>,<Parameter>,<Verzeichnis>,<Dateiname>

Parameter	-	File	sucht im Verzeichnis nach allen Dateien mit gleicher Dateiendung
		FileRec	rekursive Suche nach maximal 2000 Dateien

Beispiel:

[Interface.07]
pScrollBar3=,1,0,0,20,270,215,20

...

[Script-Select]
search,%Var%,File,%ISODir%,.iso*
Interface,set,pScrollBar3,%Var%

Sucht nach einer **vorhandenen, ausgewählten** Datei in **%ISODir%** und aktualisiert das Element **pScrollBar3** z.B. mit folgendem Eintrag:

[Interface.07]
pScrollBar3=Win7[SP0]_de-DE_x86.iso,1,0,0,20,270,215,20,Win7[SP0]_de-DE_x86.iso

[zurück zum Inhaltsverzeichnis](#)

SetReqFile

Definiert die Verzeichnisse und Parameter für die Befehle **Copyfile** und **Reqfile**

Syntax:

SetReqFile,<Quellordner>,<%DistLang%>,<%Fallback%>,<Zielordner>

Quellordner	-	jeder beliebige Ordner
%DistLang%	-	Ordner in %WinDir% mit lokalisierter Sprachdatei (MUI)
%Fallback%	-	Mui-Liste (Einträge getrennt mit „;“)
Zielordner	-	jeder beliebige Ordner

Beispiel:

SetReqFile,%boot_sys%,<%DistLang%>,<%Fallback%>,<%target_sys%>
Definiert **%boot_sys%** als Quellordner und **%target_sys%** als Zielordner.

(Siehe auch die Befehle **Copyfile** und **Reqfile**)

[zurück zum Inhaltsverzeichnis](#)

ShellExecute

Startet ein Programm oder eine Datei

Syntax:

ShellExecute, <Aktion>, <Datei>[, <Parameter>][, <Arbeitsverzeichnis>]

Aktion - *Show* Vorgang wird angezeigt
Hide Vorgang wird nicht angezeigt

Parameter - optionale Angabe von Startparametern

Arbeitsverzeichnis - optionale Angabe

Rückgabewert von ShellExecute: %Exitcode%

Exitcode 0: kein Fehler

Exitcode 1-32: unterschiedliche Fehler

Beispiele:

ShellExecute, Show, C:\Windows\system32\cmd.exe

Dieser Befehl öffnet ein Konsolenfenster.

Hinweis:

Der Befehl *ShellExecute* in einem Skript bewirkt ein Pausieren, bis die Datei / das Programm abgearbeitet ist. Erst danach werden die weiteren Zeilen im Skript ausgeführt.

[zurück zum Inhaltsverzeichnis](#)

Start

Startet ein Programm ohne darauf zu warten

Syntax:

Start, <Aktion>, <Datei>[, <Parameter>][, <Arbeitsverzeichnis>]

Aktion - *Show* Vorgang wird angezeigt
Hide Vorgang wird nicht angezeigt

Parameter - optionale Angabe von Startparametern

Arbeitsverzeichnis - optionale Angabe

Beispiel:

Start, Show, %WindowsDir%\Notepad.exe, %ScriptFile%

Dieser Befehl öffnet das jeweilige Skript mit dem Programm *Notepad*.

[zurück zum Inhaltsverzeichnis](#)

2.3.2.2. Textbearbeitung

TXTAddLine

Fügt eine Zeile in eine Datei ein

Syntax:

TXTAddLine, <Datei>, <Text>, <Parameter>

Parameter - *Top* als erste Zeile einfügen
Bottom als letzte Zeile einfügen
Zeilennummer (erste Zeile = 0)

Beispiel:

TXtAddLine,C:\MeineDatei.txt,Meine neue Zeile,Top
 Fügt in *MeineDatei.txt* die Zeile *Meine neue Zeile* als erste Zeile ein.
 Wenn die Datei nicht existiert, wird sie erstellt.

[zurück zum Inhaltsverzeichnis](#)

TXtReplace

Umbenennung eines vorgegebenen Textes

Syntax:

TXtReplace,<Datei>,<AlterText>,<NeuerText>

Beispiel:

TXtReplace,C:\MeineDatei.txt,Meine alte Zeile,Meine neue Zeile
 Ersetzt in *MeineDatei.txt* die Zeile *Meine alte Zeile* durch *Meine neue Zeile*.
 Wenn die Datei nicht existiert, wird eine leere Datei erstellt.

[zurück zum Inhaltsverzeichnis](#)

StrFormat

Gibt Teile eines Pfadnamens zurück

Syntax:

StrFormat,<%Variable%>,<Parameter>,<Datei>

Parameter	-	<i>FILENAME</i>	Dateiname
		<i>FILENAMEEX</i>	Dateiname ohne Dateiondung
		<i>PATH</i>	Dateipfad
		<i>EXT</i>	Dateiondung
		<i>SHORTPATH</i>	Anzeige des Pfades im DOS 8.3-Format

Beispiele:

StrFormat,%Var%,FILENAME,C:\boot.ini
 Weist der Variablen *%Var%* den Wert *boot.ini* zu.

StrFormat,%Var%,FILENAMEEX,C:\boot.ini
 Weist der Variablen *%Var%* den Wert *boot* zu.

StrFormat,%Var%,PATH,C:\boot.ini
 Weist der Variablen *%Var%* den Wert *C:* zu.

StrFormat,%Var%,EXT,C:\boot.ini
 Weist der Variablen *%Var%* den Wert *ini* zu.

StrFormat,%Var%,SHORTPATH,C:\MeineDatei.txt
 Weist der Variablen *%Var%* den Wert *C:\MEINED~1.TXT* zu, sofern die Datei existiert.

[zurück zum Inhaltsverzeichnis](#)

2.3.2.3. INI-Bearbeitung

IniWrite

Schreibt einen Wert in einen bestimmten Abschnitt einer INI-Datei

Syntax:

IniWrite,<Datei>,<Abschnitt>,<Schlüssel>,<Wert>

Beispiel:

IniWrite,C:\LiveSystempro\LiveSyspro.ini,Main,Project,Win7PE

Erstellt in der Datei *LiveSyspro.ini* im Abschnitt *[Main]* den Eintrag *Project=Win7PE*

[zurück zum Inhaltsverzeichnis](#)

IniWriteSection

Erstellt einen neuen Abschnitt in einer INI-Datei, Texteinträge sind optional. Einträge werden untereinander geschrieben.

Syntax:

IniWriteSection,<Datei>,<Abschnitt>[,Text]

Beispiele:

IniWriteSection,C:\LiveSystempro\LiveSyspro.ini,NeuerAbschnitt,

Erstellt in der Datei *LiveSyspro.ini* den Abschnitt *[NeuerAbschnitt]*.

Das abschließende Komma ist optional, daher ist auch diese Schreibweise möglich:

IniWriteSection,C:\LiveSystempro\LiveSyspro.ini,NeuerAbschnitt

IniWriteSection,C:\LiveSystempro\LiveSyspro.ini,NeuerAbschnitt,MeinSchlüssel

Erstellt in der Datei *LiveSyspro.ini* den Abschnitt *[NeuerAbschnitt]* mit dem Eintrag *MeinSchlüssel*.

Wenn die INI-Datei nicht existiert, wird sie erstellt.

[zurück zum Inhaltsverzeichnis](#)

IniDeleteSection

Löscht einen bestehenden Abschnitt in einer INI-Datei

Syntax:

IniDeleteSection,<Datei>,<Abschnitt>

Beispiel:

IniDeleteSection,C:\LiveSystempro\LiveSyspro.ini,NeuerAbschnitt

Löscht in der Datei *LiveSyspro.ini* den Abschnitt *NeuerAbschnitt*.

[zurück zum Inhaltsverzeichnis](#)

IniRead

Liest den Wert eines Schlüssels im angegebenen Abschnitt einer INI-Datei

Syntax:

IniRead,<%Variable%>,<Datei>,<Abschnitt>,<Schlüssel>

Beispiel:

IniRead,%Var%,C:\LiveSystempro\LiveSyspro.ini,Main,Project

Weist der Variablen *%Var%* den Wert von *Project* zu.

[zurück zum Inhaltsverzeichnis](#)

IniReadPos

Liest wahlweise den Wert eines Schlüssels oder den Schlüssel des Wertes

Syntax:

IniReadPos, <%Variable%>, <Datei>, <Abschnitt>, <Parameter>, <Position>

Parameter	<i>Count</i>	liest die Anzahl definierter Einträge in einer INI-Datei
	<i>KeyPos</i>	liest den Schlüssel in einer bestimmten Zeile (Resultat=Wert)
	<i>Value</i>	liest den Wert (Resultat=Schlüssel)
	<i>ValuePos</i>	liest den Wert in einer bestimmten Zeile (Resultat=Schlüssel)

Position Zeilennummer (nur für *ValuePos* und *KeyPos*)

Beispiele:

IniReadPos, %Var%, C:\MeineDatei.ini, MeinAbschnitt, Value, 1
Weist der Variablen **%Var%** den **Schlüssel** zu, dessen Wert **1** beträgt.

IniReadPos, %Var%, C:\MeineDatei.ini, MeinAbschnitt, ValuePos, 2
Weist der Variablen **%Var%** den **Schlüssel** aus Zeile **2** zu.

IniReadPos, %Var%, C:\MeineDatei.ini, MeinAbschnitt, KeyPos, 2
Weist der Variablen **%Var%** den **Wert** des Schlüssels aus Zeile **2** zu.

INI-Datei *C:\Lang.ini* mit folgendem Inhalt:

[Abschnitt]
Key_1=deutsch
Key_2=englisch
Beliebiger Text
Key_3=französisch

IniReadPos, %Var%, C:\Lang.ini, Abschnitt, count
Weist der Variablen **%Var%** den **Wert 3** zu (Anzahl der definierten INI-Zeilen)

[zurück zum Inhaltsverzeichnis](#)

IniRenameSection

Umbenennung eines bestehenden Abschnitts in einer INI-Datei

Syntax:

IniRenameSection, <Datei>, <AlterAbschnitt>, <NeuerAbschnitt>

Beispiel:

IniRenameSection, C:\MeineDatei.ini, Abschnitt_1, Abschnitt_2
Umbenennung von *Abschnitt_1* zu *Abschnitt_2*.

[zurück zum Inhaltsverzeichnis](#)

2.3.3. Interface-Befehle

Interface,Get

Liest den Wert eines Interface-Elements

Syntax:

Interface, Get, <Element>, <%Variable%>

Unterstützte Elementtypen	-	pTextBox pScrollBox pCheckBox pFileBox pTextLabel
----------------------------------	---	---

Die Elementtypen können auch als Variablen definiert werden

Beispiel:

```
[Interface]
pCheckbox1=True,Checkbox,0,<beliebiger_Abschnitt>,50,80,120,18
...
[Process]
Interface,Get,pCheckBox1,%Var%
Weist der Variablen %Var% den Wert True zu.
```

[zurück zum Inhaltsverzeichnis](#)

Interface,GetIndex

Ermittelt die Index-Nummer eines Interface-Elements

Syntax:

Interface,GetIndex,<Element>,<%Variable%>

Unterstützte Elementtypen - pScrollBar

Index-Nummer beginnt immer mit 1

Beispiel:

```
[Interface]
pScrollBar1=ValueB,1,0,0,30,30,60,30,ValueA,ValueB
...
[Process]
Interface,GetIndex,pScrollBar1,%Var%
Weist der Variablen %Var% den Wert 2 zu.
```

[zurück zum Inhaltsverzeichnis](#)

Interface,Set

Setzt den Wert eines Interface-Elements

Syntax:

Interface,Set,<Element>,<Wert>

Unterstützte Elementtypen - pTextBox
pScrollBar
pCheckBox
pFileBox
pTextLabel

Die Elementtypen können auch als Variablen definiert werden

Beispiele:

```
[Interface]
pCheckbox1=True,Checkbox,0,<beliebiger_Abschnitt>,50,80,120,18
...
[Process]
Interface,Set,pCheckBox1,False
Weist pCheckBox1 den Wert False zu.
```

```
Set,%RowCount%,10
Set,%Value%,Test
Interface,Set,pTextLabel%RowCount%,%Value%
Weist pTextLabel10 den Text Test zu.
```

[zurück zum Inhaltsverzeichnis](#)

Interface,State

Ändert das Attribut eines Interface-Elements

Syntax:

Interface,State,<Parameter>,<Element1>[,<Element2>][,...]

Unterstützte Elementtypen - pTextBox
pScrollBar
pCheckBox
pFileBox
pTextLabel

Die Elementtypen können auch als Variablen definiert werden

Parameter: - hide / show

Beispiel:

Interface,State,Hide,pScrollBar1

Verbirgt das Element **pScrollBar1** im Interface

[zurück zum Inhaltsverzeichnis](#)

Interface,Update

Aktualisiert den Wert eines Interface-Elements

Syntax:

Interface,Update,<Element>

Beispiel:

IniWrite,%ScriptFile%,Interface.07,pScrollBar2,Test,1,0,0,100,190,60,20,Val1,Val2,Test

Interface,Update,pScrollBar2

Aktualisiert das Element **pScrollBar2** in Interface.07 mit Wert **Test**.

[zurück zum Inhaltsverzeichnis](#)

2.4. Bearbeitung der Registry

Zur Bearbeitung der PE-Registry müssen die Registry-Zweige geladen, bearbeitet und anschließend entladen werden.

RegLoad

Lädt einen externen Registry-Zweig in die lokale Registry Ihres Systems.

Syntax:

RegLoad,<HivePfad>,<RegistryRoot>

Hivepfad - Pfad zur Hive-Datei

RegistryRoot - *PE-SYSTEM* HKLM/SYSTEM
PE-SOFTWARE HKLM/SOFTWARE
PE-USER HKCU/Software
PE-COMPONENTS HKLM/COMPONENTS

Beispiele:

RegLoad,%RegSystem%,PE-SYSTEM

RegLoad,%RegSoftware%,PE-SOFTWARE

RegLoad,%RegUser%,PE-USER

RegLoad,%target_sys%\config\components,PE-COMPONENTS

Hinweis:

Standardmäßig wird HKLM\COMPONENTS in der PE-Registry nicht angezeigt. Um diesen Registry-Zweig trotzdem anzuzeigen, ist folgender Vorgang notwendig:

1. Im erstellten PE den Registrierungs-Editor öffnen
2. Den Zweig HKEY_LOCAL_MACHINE anklicken
3. Im Menü „Datei“ *Struktur laden* auswählen
4. Als Pfad *X:\Windows\System32\config\components* auswählen, auf ÖFFNEN klicken
5. Im erscheinenden Fenster STRUKTUR LADEN *COMPONENTS* eingeben
6. HKLM\COMPONENTS ist nun verfügbar

[zurück zum Inhaltsverzeichnis](#)

RegUnLoad

Entlädt den externen Registry-Zweig

Syntax:

RegUnLoad,<RegistryRoot>

RegistryRoot-	PE-SYSTEM	(HKLM\SYSTEM =	%RegSystem%)
	PE-SOFTWARE	(HKLM\SOFTWARE =	%RegSoftware%)
	PE-USER	(HKCU\Software =	%RegUser%)
	PE-COMPONENTS	(HKLM\COMPONENTS)	
	All	entlädt alle Registry-Zweige	

Beispiele:

RegUnLoad,PE-SYSTEM

Entlädt den Registry-Zweig *PE-SYSTEM*.

RegUnLoad,All

Entlädt alle Registry-Zweige.

[zurück zum Inhaltsverzeichnis](#)

RegRead

Liest einen Registry-Schlüssel

Syntax:

RegRead,<%Variable%>,<HKEY>,<Schlüssel>,<Wert>

Beispiel:

RegRead,%Var%,HKLM,SOFTWARE\7-zip,Path

Weist der Variablen *%Var%* den Wert von *Path* zu (=C:\Program Files\7-zip)

Bei REG_MULTI_SZ (0x7) wird das Trennzeichen als *#\$x* (Zeilenumbruch) interpretiert

[zurück zum Inhaltsverzeichnis](#)

RegWrite

Erstellt oder verändert einen Registry-Schlüssel

Syntax:

RegWrite,<HKEY>,<Type>,<Schlüssel>[,WertName][,Wert]

HKEY	-	<i>HKLM</i>	-	HKEY_LOCAL_MACHINE
		<i>HKCR</i>	-	HKEY_CLASSES_ROOT
		<i>HKCU</i>	-	HKEY_CURRENT_USER
		<i>HKU</i>	-	HKEY_USERS

		<i>HKCC</i>	-	HKEY_CURRENT_CONFIG
		<i>HKLM64</i>	-	HKEY_LOCAL_MACHINE Wow64
		<i>HKCU64</i>	-	HKEY_CURRENT_USER Wow64
Type	-	<i>0x0</i>		leerer Schlüssel mit Eintrag (<i>Standard</i>)
		<i>0x1</i>	REG_SZ	schreibt Text
		<i>0x2</i>	REG_EXPAND	expandiert Variable in Schreibweise <i>%%</i>
		<i>0x3</i>	REG_BINARY	schreibt einen Binärwert
		<i>0x4</i>	REG_DWORD	schreibt numerischen Wert
		<i>0x7</i>	REG_MULTI_SZ	schreibt mehrfache durch <i>#\$x</i> separierte Zeichenketten
Schlüssel	-			Name des Schlüssels
Wertname	-			optionaler Name des Werts
Wert	-			optionaler Wert

Erweiterte Schreibweise für Variablen

```
Set,%var%,01,a2,03
RegWrite,HKLM,0x3,PE-SOFTWARE\Software,Test,%var%

Set,%var%,01,02,03
RegWrite,HKLM,0x3,PE-SOFTWARE\Software,Test,%var%,04,aa,05
```

Erweiterte Schreibweise für REG_MULTI_SZ (0x7):
als Trennzeichen wird *#\$x* verwendet

```
Set,%Var%,10#$x20
RegWrite,HKLM,0x7,SOFTWARE\MeinKey,Key0x7,%Var%,30,40
```

32-Bit-System:

Schreibt in den REG-Schlüssel *SOFTWARE\MeinKey* die Zeichenfolge *Key0x7* mit den Werten *10 20 30 40*

64-Bit-System:

Schreibt in den REG-Schlüssel *SOFTWARE\Wow6432Node\MeinKey* die Zeichenfolge *Key0x7* mit den Werten *10 20 30 40*

Leere Multistring-Einträge werden übersprungen

Beispiele:

```
RegWrite,HKLM,0x0,PE-SOFTWARE\NeuerKey
Erstellt in der PE-Registry einen neuen Schlüssel NeuerKey.
```

```
RegWrite,HKLM,0x0,PE-USER\Software\Microsoft\Windows\CurrentVersion\MeinRegKey
Schreibt in die PE-Registry in HKCU\Software\Microsoft\Windows\CurrentVersion den neuen Schlüssel MeinRegKey mit Zeichenfolge (Standard) und dem Typ REG_SZ, sowie im Feld Daten den Eintrag (Wert nicht festgelegt).
```

```
Set,%var%,01,a2,03
RegWrite,HKLM,0x3,PE-SOFTWARE\Software,Test,%var%
Schreibt in die PE-Registry in HKLM\Software den Wertnamen Test mit dem Binärwert 01,a2,03.
```

```
Set,%var%,01,02,03
RegWrite,HKLM,0x3,PE-SOFTWARE\Software,Test,%var%,04,aa,05
Schreibt in die PE-Registry in HKLM\Software den Wertnamen Test mit dem Binärwert 01,02,03,04,aa,05.
```

Hinweis zu Typ 0x4:

Es ist sowohl die dezimale als auch die hexadezimale Schreibweise möglich:

```
RegWrite,HKLM,0x4,PE-SYSTEM\ControlSet001\Services\WpsSvc,Start,3
Schreibt in die PE-Registry in HKLM\SYSTEM\ControlSet001\Services\WpsSvc den Wertnamen Start mit einem numerischen Wert von 3.
```

```
RegWrite,HKLM,0x4,PE-SYSTEM\ControlSet001\Services\WpsSvc,Start,0xFF
```

Schreibt in die PE-Registry in *HKLM\SYSTEM\ControlSet001\Services\MpsSvc* den Wertnamen *Start* mit einem numerischen Wert von *255*.

[zurück zum Inhaltsverzeichnis](#)

RegDelete

Löscht einen Registry-Schlüssel

Syntax:

RegDelete, <HKEY>, <Abschnitt>[, WertName]

Beispiele:

RegDelete, HKLM, PE-SYSTEM\ControlSet001\Control\ComputerName\ComputerName, ComputerName
Löscht im angeführten Registry-Schlüssel den **Wertnamen** *ComputerName*.

RegDelete, HKLM, PE-SYSTEM\Test

Löscht im angeführten Registry-Schlüssel den **Wertnamen** (*Standard*).

[zurück zum Inhaltsverzeichnis](#)

2.5. Bearbeitung von Zeichenketten (Strings)

String LEFT/RIGHT

Untersuchung der Buchstabenanzahl von Zeichenketten

Syntax:

String, <%Variable%>, <Aktion>, <Zeichenkette>, <Zähler>

Aktion - *Left* Anzahl der Buchstaben einer Zeichenkette, links beginnend
Right Anzahl der Buchstaben einer Zeichenkette, rechts beginnend

Zähler - Anzahl der Zeichen

Beispiele:

Set, %String%, Sommerurlaub

String, %var%, Left, %String%, 6

Weist der Variablen *%var%* den Wert *Sommer* zu.

Set, %String%, Sommerurlaub

String, %var%, Right, %String%, 6

Weist der Variablen *%var%* den Wert *urlaub* zu.

[zurück zum Inhaltsverzeichnis](#)

String TLeft/TRight

Schneidet beliebige Buchstabenanzahl von Zeichenketten ab

Syntax:

String, <%Variable%>, <Aktion>, <Zeichenkette>, <Zähler>

Aktion - *TLeft* Schneidet Buchstaben einer Zeichenkette ab, links beginnend
TRight Schneidet Buchstaben einer Zeichenkette ab, rechts beginnend

Beispiele:

Set,%String%,Hallo
String,%Var%,TLeft,%String%,1
Weist der Variablen **%Var%** den Wert **allo** zu.

Set,%String%,Hallo
String,%Var%,TRight,%String%,1
Weist der Variablen **%Var%** den Wert **Hall** zu.

[zurück zum Inhaltsverzeichnis](#)

String FIND

Zeigt die erste Position einer Zeichenkette in einer anderen (zu untersuchenden) Zeichenkette an

Syntax:

String,<%Variable%>,FIND,<Hauptstring>,<Teilstring>

Beispiel:

Set,%String_1%,Sommerurlaub
Set,%String_2%,url
String,%Var%,FIND,%String_1%,%String_2%
Weist der Variablen **%Var%** die Position (**7**) der Übereinstimmung des Buchstabens **u** zu.

Hinweis:

Wird keine Übereinstimmung gefunden, dann hat die Position den Wert **0**

[zurück zum Inhaltsverzeichnis](#)

String LEN

Ermittelt die Länge einer Zeichenkette

Syntax:

String,<%Variable%>,LEN,<Zeichenkette>

Beispiel:

String,%Var%,LEN,Teststring
Weist der Variablen **%Var%** die Länge (**10**) des Strings **Teststring** zu.

[zurück zum Inhaltsverzeichnis](#)

String MID

Schneidet einen Teilbereich aus einer Zeichenkette

Syntax:

String,<%Variable%>,MID,<Zeichenkette>,<Position>,<Zähler>

Beispiel:

Set,%String%,Sommerurlaub
String,%Var%,MID,%String%,2,4
Weist der Variablen **%Var%** den Wert **omme** zu.

[zurück zum Inhaltsverzeichnis](#)

String REPLACE

Ersetzt Teilbereich einer Zeichenkette durch einen anderen Bereich

Syntax:

String, <%Variable%>, REPLACE, <Zeichenkette>, <Suchstring>, <Ersatzstring>[, Parameter]

Parameter	-	0	(Standard)	Auflösung aller Variablen
		1		Auflösung aller Variablen bis Level 1 (1. Variable in der Variablen)

Beispiel:

Set, %Zeichenkette%, Franz jagt im total verwaahlrosten Taxi quer durch Bayern

Set, %Suchstring%, Bayern

Set, %Ersatzstring%, Österreich

String, %Var%, REPLACE, %Zeichenkette%, %Suchstring%, %Ersatzstring%

Weist der Variablen **%Var%** die neue Zeichenkette

Franz jagt im total verwaahlrosten Taxi quer durch **Österreich**

zu.

[zurück zum Inhaltsverzeichnis](#)

String SPLIT

Teilt einen String in Teilstrings

Syntax:

String, <%Resultat%>, SPLIT, <%Variable%>, <Trennzeichen>, <Parameter>

Parameter = 0	-	Anzahl der Elemente (Felder) von %Variable%
Parameter > 0	-	Gibt den Teilstring von %Variable% des angegebenen Index zurück

Beispiel:

RegRead, %Var%, HKLM, SYSTEM\CurrentControlSet\Services\NetBIOS\Linkage, Bind

String, %Count%, SPLIT, %Var%, #\$, 0

For, %x%, 1, %Count%

String, %Res%, SPLIT, %Var%, #\$, %x%

Message, %Res%=%x%

Next

Weist der Variablen **%Var%** den Gesamtinhalt der Zeichenfolge **Bind** in obigem REG-Schlüssel zu, während die Variable **%Count%** die Gesamtanzahl der ausgelesenen Felder enthält.

In der folgenden **For/Next-Schleife** wird der Inhalt der Variablen **%Var%** an Hand des jeweiligen Parameters geteilt und in jedem einzelnen Schleifendurchgang der Variablen **%Res%** zugewiesen.

[zurück zum Inhaltsverzeichnis](#)

2.6. Netzwerkbefehle

WebGet

Lädt eine Datei aus dem Internet

Syntax:

WebGet, <DateiImInternet>, <ZuSpeichernderPfad>[, Parameter]

Parameter	-	1	zeigt einen Fortschrittsbalken an
		2	lädt Datei mit unbekannter Größe

Beispiele:

Download ohne Fortschrittsbalken (http://)

WebGet, http://irfanview.tuwien.ac.at/iview433.zip, C:\iview433.zip

Lädt die Datei **iview433.zip** von **http://irfanview.tuwien.ac.at** und speichert sie in **C:**

Download ohne Fortschrittsbalken (ftp://)

WebGet,ftp://ftp.cpubid.com/cpu-z/cpu-z_1.61-32bits-en.zip,C:\cpu-z_1.61-32bits-en.zip
Lädt die Datei *cpu-z_1.61-32bits-en.zip* von *ftp.cpubid.com* und speichert sie in *C:*

Download mit Fortschrittsbalken

Webget,ftp://wa651f4:anonymous@mh-nexus.de/HxDSetupDE.zip,C:\HxDSetupDE.zip,1
Lädt die Datei *HxDSetupDE.zip* und zeigt einen Fortschrittsbalken an

Download einer Datei ohne Größeninformation mit Fortschrittsbalken

Webget,http://googledrive.com.../Firefox_36.0.4.zip,C:\Firefox.zip,3
Lädt die Datei unbekannter Größe *Firefox_36.0.4.zip* und zeigt einen Fortschrittsbalken an

WebGet verfügt über eine Rückgabevariable **%Exitcode%**:

0	-	Ok
1	-	Fehler

Beispiel:

System,Error,Off
WebGet,http://www.a43filemanager.com/download/a43.zip,c:\a43.zip,1
Message,Download: %Exitcode%
System,Error,on

Anmerkung:

Bei Verwendung des WebGet-Parameters **3** (Kombination aus 1 und 2) ist **%Exitcode%** nicht notwendig.

Verwendung von FTP-Downloads mit Passwort:

Beispiel:

WebGet,ftp://wa651f4:anonymous@mh-nexus.de/HxDSetupDE.zip,C:\HxDSetupDE.zip,1

[zurück zum Inhaltsverzeichnis](#)

2.7. Schleifenbefehle

2.7.1. For / Next - Schleife

Führt einen Befehl mehrfach aus

Syntax:

*For,<%Variable%>,<Start>,<End>
<Befehl>*

Next

Start	Anfangswert der Schleife
End	Endwert der Schleife

Beispiel:

*For,%Var%,1,5
Message,Schleife: %Var%*

Next

Diese Schleife wird fünf Mal ausgeführt, wobei die Werte von **%Var%** sich jedes Mal ändern und folgende Meldungen ausgegeben werden:

Schleife: 1
Schleife: 2
Schleife: 3
Schleife: 4
Schleife: 5

[zurück zum Inhaltsverzeichnis](#)

2.7.2. ExitLoop

Verlässt eine For/Next-Schleife

Syntax:

ExitLoop

Beispiel für eine rekursive Schleife:

```
If,a,=,a,then
  For,%x%,1,6
    if,a,=,a,then
      Run,%ScriptFile%,Load-D,mouclass.sys
      if,%x%,=,4,then,ExitLoop
    Else
      Message,Meldungsfenster ohne Fenstertitel
    End
    Message,Schleife: %x%
  Next
Else
  Message,Test 1
End
Message,Schleifenende
```

```
[Load-D]
Message,sub %x%,#1
```

In diesem Beispiel öffnet sich ein Meldungsfenster **mit** dem Fenstertitel **mouclass.sys** und der Meldung **sub1**, anschließend öffnet sich ein Meldungsfenster **ohne** Fenstertitel und der Meldung **Schleife: 1**, anschließend öffnet sich ein Meldungsfenster **mit** dem Fenstertitel **mouclass.sys** und der Meldung **sub2**. Dieser Vorgang wiederholt sich bis zum Meldungsfenster **mit** dem Fenstertitel **sub4**, anschließend öffnet sich ein Meldungsfenster **ohne** Fenstertitel und der Meldung **Schleifenende**

[zurück zum Inhaltsverzeichnis](#)

2.8. Variablen (intern)

Diese Variablen werden vom Builder beim Starten selbst gesetzt

<i>%BaseDir%</i>	-	Builder-Verzeichnis	(z.B.: C:\LiveSystempro)
<i>%BuildMode%</i>	-	Prozessmodus	
		0:	Button „Start“
		1:	Kommandozeile
		2:	Button „>“
		3:	Interface-Element [Abschnitt]
<i>%CPUVM%</i>	-	Prüft, ob der Prozessor virtualisierungsfähig ist (VT-x)	
		0:	nicht virtualisierungsfähig
		1:	virtualisierungsfähig
<i>%ExitCode%</i>	-	Rückgabewert	
<i>%HostAdmin%</i>	-	Admin-Recht des Benutzers	
		0:	Kein Admin-Recht
		1:	Admin-Recht
<i>%HostArch%</i>	-	Architektur des Host-Rechners	(x86, x64)
<i>%HostLang%</i>	-	Spracheinstellung der Ländergruppe (hexadezimal)	(07, 09, ...)
<i>%HostNT%</i>	-	NT-Version des Host-Systems	(5.1, 5.2, 6.0, 6.1, 6.2, 6.3, 10.0)
<i>%HostOS%</i>	-	Betriebssystem des Host-Rechners	(WinXP, Win2003, WinVista, Win7, Win8, Win10)
<i>%ISODir%</i>	-	ISO-Verzeichnis	(z.B.: %BaseDir%\Iso)
<i>%ISOfile%</i>	-	Pfad zur ISO-Datei	(z.B.: %BaseDir%\Iso\winpe3.iso)
<i>%ISOFilename%</i>	-	Name der ISO-Datei	
<i>%PEMode%</i>	-	nur in Win8PE verfügbar (mit Source Windows 8.1)	

		PE:	Build im PE-Modus
		OS:	Build im Betriebssystem-Modus
%ProcessError%	-	0:	kein Fehler
		3:	Halt
		4:	Fehler
			(werden nur in %Projectini%,[ONBUILDEXIT] gesetzt)
%ProgramFilesDir%	-	Programmverzeichnis	(z.B.: C:\Program Files)
%Programs64%	-	64-Bit Programmverzeichnis	
%ProjectDir%	-	Projektverzeichnis	(%BaseDir%\Projects\ <i><Projekt></i>)
%ProjectFolder%	-	Verzeichnisname des Projekts	(Win7PE, Win8PE, Win10PE, WinXPE)
%Projectini%	-	Projektdatei	(%BaseDir%\Projects\ <i><Projekt></i> \project.cfg)
%ProjectName%	-	Projektname	(Win7PE, Win8PE, Win10PE, WinXPE)
%ScriptDir%	-	Verzeichnis der aktuellen Skript-Datei	
%ScriptFile%	-	aktuelle Skript-Datei	
%ScriptLvl%	-	Scriptlevel	
%SourceDir%	-	Quellverzeichnis des Installationsmediums	
%TargetDir%	-	Zielverzeichnis	(z.B.: %BaseDir%\Target\Win7PE)
%TempDir%	-	Temp-Verzeichnis des Benutzers	(z.B.: C:\Users\ <i><Benutzername></i> \AppData\Local\Temp)
%UserName%	-	Benutzername	
%UserProfile%	-	Benutzerverzeichnis	(z.B.: C:\Users\ <i><Benutzername></i>)
%WindowsDir%	-	Windows-Verzeichnis	(z.B.: C:\Windows)
%WinUAC%	-	Benutzerkontensteuerung (ab Windows Vista)	
		Wert	0/1

[zurück zum Inhaltsverzeichnis](#)

2.9. Variablen (Projekt)

Win7PE, Win8PE, Win10PE:

%BootSRC%	-	Boot-Medium (extrahierte Datei boot.wim)	
		z.B.: %BaseDir%\Wim\Win7SP0de-DE-B\2	
%boot_sys%	-	System32-Verzeichnis in %BootSRC%	
		z.B.:	
		%BaseDir%\Wim\Win7x86SP0de-DE-B\2\Windows\system32	
%boot_win%	-	Windows-Verzeichnis in %BootSRC%	
		z.B.: %BaseDir%\Wim\Win7x86SP0de-DE-B\2\Windows	
%boot_wow%	-	SysWow64-Verzeichnis in %BootSRC%	(nur 64-Bit-Medium)
		z.B.:	
		%BaseDir%\Wim\Win7x86SP0de-DE-B\2\Windows\SysWoW64	
%CDDrive%	-	Laufwerk Y:	
%DistLang%	-	Sprache der Quell-DVD	
%Fallback%	-	Mui-Liste der Quell-DVD (Trennzeichen bei mehreren Sprachen „;“)	
%InstallSRC%	-	Install-Medium (extrahierte Datei install.wim)	
		z.B.: %BaseDir%\Wim\Win7-Enterprisex86SP0de-DE-I	
%Install_sys%	-	System32-Verzeichnis in %InstallSRC%	
		z.B.:	
		%BaseDir%\Wim\Win7Enterprisex86SP0de-DE-I\Windows\System32	
%Install_win%	-	Windows-Verzeichnis in %InstallSRC%	
		z.B.:	
		%BaseDir%\Wim\Win7Enterprisex86SP0de-DE-I\Windows	
%Install_wow%	-	SysWoW64-Verzeichnis in %InstallSRC%	(nur 64-Bit-Medium)
		z.B.:	
		%BaseDir%\Wim\Win7Enterprisex64SP0de-DE-I\Windows\SysWoW64	
%Installwim%	-	Pfad zur Image-Datei (Install.wim / Install.esd)	
%ISOfile%	-	z.B.: %BaseDir%\ISO\Win8[SP0]-de_DE_x86.iso	
%PECFG%	-	%TargetDir%\Windows\System32\win7pe.cfg	
%prog_files%	-	Program Files	
%RamDrive%	-	Laufwerk X:	
%SourceEdition%	-	Edition des Windows-Quell-Mediums	
%target_wow%	-	SysWoW64-Zielverzeichnis	(%TargetDir%\Windows\SysWoW64)
%WIM_ARCH%	-	Image-Architektur	
%Wim_BootIdx%	-	Image-Index der boot.wim	
%WIM_BUILD%	-	Teil der Microsoft-Dateiversion	(z.B.: 7600 bei ,6.1.7600.16385')
%WIM_Index%	-	Image-Index	

%WIM_MAJOR%	-	Teil der Microsoft-Dateiversion	(z.B.: 6 bei ,6.1.7600.16385')
%WIM_MINOR%	-	Teil der Microsoft-Dateiversion	(z.B.: 1 bei ,6.1.7600.16385')
%WIM_SPBUILD%	-	Teil der Microsoft-Dateiversion	(z.B.: 16385 bei ,6.1.7600.16385')
%WIM_SPLEVEL%	-	ServicePack-Version	

WinXP PE:

%administrator%	-	Administrator
%admin_tools%	-	Verwaltung
%all_users%	-	All Users
%app_data%	-	Anwendungsdaten
%cookies%	-	Cookies
%common_files%	-	Gemeinsame Dateien
%desktop%	-	Desktop
%doc_and_sett%	-	Dokumente und Einstellungen
%docs%	-	Dokumente
%drCabFile%	-	komprimierte Treiberdatei (im Format CAB)
%favorites%	-	Favoriten
%history%	-	Verlauf
%Keyboard%	-	Tastatur-Einstellungen
%LangID%	-	Sprache des Quell-Mediums
%LangGroup%	-	Sprachengruppe
%local_sett%	-	Lokale Einstellungen
%LCID%	-	Lokale ID (landesspezifisch)
%ms_ie_quicklaunch%	-	Microsoft\Internet Explorer\Quick Launch
%mydocs%	-	Eigene Dateien
%OEMCP%	-	Codepage
%PE_Programs%	-	%Systemdrive%\%prog_files%
%prog_files%	-	Programme
%RamDrive%	-	Laufwerk B:
%sendto%	-	Senden an
%source_sys%	-	%SourceDir%\I386\System32
%source_win%	-	%SourceDir%\I386
%SPCabFile%	-	ServicePack-Version der CAB-Datei
%startmenu%	-	Startmenü
%startmenu_prog%	-	Startmenü\Programme
%startmenu_prog_startup%	-	Startmenü\Programme\Autostart
%startup%	-	Autostart
%target_sys%	-	%TargetDir%\I386\System32
%target_win%	-	%TargetDir%\I386
%templates%	-	Vorlagen
%temp_internet_files%	-	Temporäre Internet Dateien

Hinweis:

Die deutschen Bezeichnungen entsprechen einer deutschsprachigen WinXP-Quelle.

Gemeinsame Variablen:

%AppsDir%	-	Standardverzeichnis für Downloads	(%BaseDir%\Temp\Apps)
%Arch%	-	Architektur des Quell-Mediums	(x86, x64)
%LSPVer%	-	Versionsnummer von LiveSyspro.exe	
%NT%	-	NT-Version des Quell-Mediums	
		5.1	Windows XP
		5.2	Windows XP Professional x64, Server 2003
		6.0	Windows Vista, Server 2008
		6.1	Windows 7, Server 2008 R2
		6.2	Windows 8
		6.3	Windows 8.1
		10.0	Windows 10
%NTVersion%	-	NT-Version des Quell-Mediums	(5, 6)
%OS%	-	Betriebssystem des Quell-Mediums	(WinXP, Win2003, WinVista, Win7, Win8, Win10)
%PE_Programs%	-	entweder X: :\%prog_files% oder Y: :\%prog_files%	
%PreUserdata%	-	%BaseDir%\Temp%\ProjectName%\UserData	
%ProjectTemp%	-	Temporärer Projektordner	(z.B.: %BaseDir%\Temp\WinXPE)
%Ramboot%	-	True/False	
%RegSoftware%	-	Pfad zum Registry-Zweig	PE-SOFTWARE

%RegSystem%	-	Pfad zum Registry-Zweig PE-SYSTEM
%RegUser%	-	Pfad zum Registry-Zweig PE-USER
%SP%	-	Versionsnummer des ServicePacks
%Target_CD%	-	Zielverzeichnis (%TargetDir%_CD)
%Target_Prog%	-	%TargetDir%\%prog_files%
%Target_Sys%	-	System-Zielverzeichnis (%TargetDir%\Windows\System32)
%Target_Win%	-	Windows-Zielverzeichnis (%TargetDir%\Windows)
%Tools%	-	Tools-Verzeichnis (%BaseDir%\Tools)

[zurück zum Inhaltsverzeichnis](#)

2.10. System,SetPEInit

PEInit führt Aufgaben (Programmstarts, Öffnen von cmd-Boxen, etc.) beim Start von **WindowsPE** aus. Diese Befehle werden in die Datei **PEInit.ini** geschrieben. Die Zählweise der Befehle reicht von **001** bis **059**, wobei **001** bis **015** für das jeweilige Projekt reserviert sind.

Syntax:

System,SetPEInit,<ID>,<Abschnitt>,<Befehl>

ID	ID des Eintrags (Buchstaben oder Zahlen verwendbar)
Abschnitt	gültig nur im jeweiligen Skript Init (vor Start des Explorers) Startup (nach Start des Explorers)

Befehle mit Show/Hide-Effekt bei gestarteten Programmen:

CMD	
Run	
RunWait	
Show/Hide-Effekt:	
Show	zeigt das gestartete Programm an (Standard, kann daher weggelassen werden)
Hide	verbirgt das gestartete Programm

Beispiel:

System,SetPEInit,A,Init,RunWait||%PE_programs%\DrvImp\DrvImpe.exe|-d -i:"%UFD%\Drivers\%Archit%"
Erstellt in **PEInit.ini** z.B. diese Einträge:

```
[Init]
20=RunWait||X:\Programme\Drvimp\DrvImpe.exe -d -i:"%UFD%\Drivers\%Archit%"
[Init-ID]
20=A,Apps\my - Driver Import PE\DrvImpe.srt
```

Beispiel für WinXPE:

```
[Init]
003=Ramdisk|25|%\imdisk.exe -a -t vm -m B: -s %Rsize%M -p "/fs:NTFS /q /y /v:Imdisk"
004=CMD||md "B:\Documents and Settings\Default User\Local Settings\Temp"
006=DllInstall|shell32.dll
007=DllInstall|browseui.dll
008=DllInstall|shdocvw.dll
039=BROADCAST
016=RunWait||reg.exe|add HKLM\SYSTEM\CurrentControlSet\Services\usbhub /v ImagePath /t REG_EXPAND_SZ /d system32\DRIVERS\usbhub.sys /f
```

Die Nummern müssen nicht fortlaufend geordnet sein, allerdings werden vorhandene Nummern von gleichlautenden Nummern überschrieben.

Befehlsliste von **peinit.exe**:

CMD	führt einen DOS-Befehl aus
	Beispiele:
	CMD md #<Q>%RamDrive%\%doc_and_sett%\Default User\%local_sett%\Temp#<Q> Cmd-Box wird angezeigt
	CMD Hide md #<Q>%RamDrive%\%doc_and_sett%\Default User\%local_sett%\Temp#<Q> Cmd-Box wird nicht angezeigt

Run startet ein Programm

Beispiel:

```
Run||imdisk.exe|-a -t vm -m %RamDrive% -s %Rsize%M -p #q/fs:%Format% /q /y
/v:%pTextBox2%#sq
```

Runwait startet ein Programm und wartet auf das Ende

Beispiel:

```
Runwait||imdisk.exe|-a -t vm -m %RamDrive% -s %Rsize%M -p #q/fs:%Format% /q /y
/v:%pTextBox2%#sq
```

BROADCAST aktualisiert die Umgebungsvariablen

Beispiel:

```
BROADCAST
```

DllInstall installiert eine DLL-Datei

Beispiel:

```
DllInstall|shell32.dll
```

Auch verfügbar mit zusätzlichem Installations-Parameter

Beispiele:

```
DllInstall|shell32.dll|setup
```

```
DllInstall|shell32.dll|InitMain
```

```
DllInstall|themeui.dll|UserInstall
```

DrvLetter ändert den Laufwerksbuchstaben

Beispiel:

```
DrvLetter|%CDDrive%|I386\ImageFile%
```

FileDelete löscht eine Datei

Beispiel:

```
X:\User\Desktop.ini
```

Ramdisk erstellt eine RamDisk

Beispiel:

```
Ramdisk|%pTextBox1%|%pScrollBar1%|imdisk.exe -a -t vm -m %RamDrive% -s %
Rsize%M -p#q/fs:%Format% /q /y /v:%pTextBox2%#sq
```

Wert von **%pTextBox1%**: Größe der RamDisk

Wert von **%pScrollBar1%**: % oder **MB**

%Rsize%: Ergebnis von %pTextBox1% + %pScrollBar1%

RegDelete löscht einen Registry-Wert

```
RegDelete,<Schlüssel>,[Optionaler Wert]
```

Beispiele:

```
RegDelete\HKLM\SYSTEM\CurrentControlSet\Services\AmdK8
```

```
RegDelete\HKLM\SYSTEM\CurrentControlSet\Services\AmdK8\Start
```

Register registriert eine DLL-Datei

Beispiel:

```
Register|shell32.dll
```

RegWrite schreibt einen Registry-Wert

Beispiel:

```
RegWrite\HKLM\0x1\SYSTEM\ControlSet001\key\ka#-$dre ;'$-d = "|"
```

Für Sonderzeichen Verwendung des Schlüsselworts **\$-d = "|"**

SetRes ändert die Bildschirmauflösung

Beispiel:

```
SetRes|1024|768
```

Shell lädt die angegebene Datei als Shell (nur verfügbar im Abschnitt **Init**)

Beispiel:

```
Shell|Explorer.exe
```

StartService startet einen Dienst

Beispiel:

```
StartService|Ext2fs|fsDrives
```

Mehrfachangaben sind möglich (getrennt durch **|**)

Wait pausiert eine vorgegebene Zeit (in Millisekunden)

Beispiel:

```
Wait|3000
```

Speicherort von Peinit.ini:

%Target_Sys% in allen Projekten

[zurück zum Inhaltsverzeichnis](#)

2.11. Sonderzeichen

Folgende Sonderzeichen werden in LiveSystempro verwendet:

#\$x	-	Wagenrücklauf + Zeilenvorschub
#\$q	-	Anführungszeichen (")
#\$c	-	Komma
#\$p	-	Prozentzeichen (%)
#\$s	-	Leerzeichen
#\$d	-	senkrechter Strich () =Pipe)

[zurück zum Inhaltsverzeichnis](#)

2.11. Shortcuts

Syntax:

Add_Shortcut,[Type],[StartMenuFolder],[FullFileName],[Title],[Workfolder],[Parameters],[IconFile]

Type	-	StartMenu Desktop Quicklaunch
StartMenuFolder	-	a. Eingabe eines speziellen Namens für „Ordner\Unterordner“ b. Eingabe von "." für eine Verknüpfung in <i>Startmenü\Programme</i> c. Leerfeld ebenfalls möglich
FullFileName	-	Nur notwendig, wenn Verknüpfung unterschiedlich zu %ProgramExe% . Standard: %Pe_Programs%\%ProgramFolder%\%ProgramExe%
Title	-	Beliebiger Titel Standard: %ProgramTitle%
Workfolder	-	Beliebiges Arbeitsverzeichnis
Parameters	-	Nur für Start von %ProgramExe% mit Startparametern
IconFile	-	Nur bei speziellem Icon für die Verknüpfung

Beispiele:

Add_Shortcut,Desktop
Verknüpft das auszuführende Programm mit dem Desktop.

Add_Shortcut,StartMenu,%pTextBox1%
Verknüpft den Wert von **%pTextBox1%** mit dem Startmenü.

Add_Shortcut,StartMenu,,MeineProgrammEXE,,,
Erstellt eine Verknüpfung mit dem auszuführenden Programm in **Startmenü\Programme** (ohne Unterordner).

Add_Shortcut,StartMenu,,MeineProgrammEXE,,,exe.ico#\$c7
Erstellt eine Verknüpfung mit dem auszuführenden Programm in **Startmenü\Programme** mit dem Icon #7 der EXE-Datei.

Add_Shortcut,StartMenu,MeinOrdnerMeinUnterordner,MeineProgrammEXE,,,
Erstellt den Ordner **MeinOrdnerMeinUnterordner** in **Startmenü\Programme**, in diesem Ordner liegt

MeineProgrammEXE .

Add_Shortcut,Desktop,,,IrfanView Thumbnails,,/thumbs,%PE_Programs%\%ProgramFolder%\Thumbnails.ico
Verknüpft *IrfanView Thumbnails* mit dem Desktop.
Startparameter für *%ProgramExe%* ist */thumbs*, angezeigt wird Icon der Datei *Thumbnails.ico*

[zurück zum Inhaltsverzeichnis](#)

2.12. Makros

- Makros bieten die Möglichkeit, gewisse wiederkehrende Befehle automatisch ablaufen zu lassen (vgl. Befehl **SetMacro**).
- Manche Dateien (nicht alle sind notwendig) werden von PreConfig aus der Quelle in den Wim-Cache entpackt. Dort befinden sich 2 separate Wim-Unterverzeichnisse (*boot.wim* and *install.wim*)
Makros verwenden diese 2 Wim-Speicher.
Ausnahmen:
Add_Shortcut und *RunFromRam* (alle Projekte)
- Makros werden in *%Projectini% [Macros]* gespeichert.
- **Fallback en-US (en-US Ersatzfunktion):**
Eine englische mui-Datei wird nur kopiert, wenn *%Distlang% mui* nicht existiert.
Bei fehlender Übereinstimmung der mui-Dateien wird keine kopiert. **Keine** Fehlermeldung bei fehlender mui-Datei.

Hinweis:

Die LSP-Projekte verwenden unterschiedliche Makros.

Alle Projekte:

Add_ShortCut

Erstellt Programmverknüpfungen (siehe Abschnitt 2.11)

RunFromRAM,<Wert>

Wert - False | True

Steuert die Programmausführung in *RAM* oder von *CD*

Beispiele:

RunFromRAM,False

Das Programm (*=%PE_programs%*) wird von *CD* gestartet (Laufwerk *Y:*)

RunFromRAM,True

Das Programm (*=%PE_programs%*) wird im *RAM* gestartet (Laufwerk *X:*)

Win7PE, Win8PE, Win10PE:

FileDecBoot

Dekomprimiert Dateien aus der *boot.wim* nach *%BootSRC%*

Beispiel:

FileDecBoot,Windows\System32\Wbem\%DistLang%\Msi.mfl

Extrahiert die Datei *Msi.mfl* nach z.B.:

%BaseDir%\Wim\Win7SP0de-DE-B\Windows\System32\Wbem\%DistLang%

FileDecInst

Dekomprimiert Dateien aus der *install.wim* nach *%InstallSRC%*

Beispiel:

FileDecInst,Windows\System32\Wbem.**

Extrahiert alle Dateien nach z.B.:

%BaseDir%\Wim\Win7Enterprisex86SP0de-DE-I\Windows\System32\Wbem

ReqSys32,<Datei>

Datei - Zu kopierende Datei im Verzeichnis System32 (Install Wim-Cache) + mui (Fallback en-US)

Beispiel:

ReqSys32,mapi32.dll

Kopiert *mapi32.dll* nach *Windows/System32*

Kopiert *mapi32.dll.mui* nach *Windows/System32\%DistLang%*

Wildcards werden nicht unterstützt.

Fehlermeldung, falls Datei nicht existiert.

ReqB32,<Datei>

Datei - Zu kopierende Datei im Verzeichnis System32 (Boot Wim-Cache) + mui (Fallback en-US)

Beispiel:

ReqB32,attrib.exe

Kopiert *attrib.exe* nach *Windows/System32*

Kopiert *attrib.exe.mui* nach *Windows/System32\%DistLang%*

Wildcards werden nicht unterstützt.

Fehlermeldung, falls Datei nicht existiert.

ReqSysWoW64,<Datei>

Datei - Zu kopierende Datei im Verzeichnis SysWow64 (Install Wim-Cache) + mui (Fallback en-US)

Beispiel:

ReqSysWoW64,activeds.dll

Kopiert die 32-Bit-Datei *activeds.dll* nach *Windows\SysWoW64*

Kopiert die 32-Bit-Datei *activeds.dll.mui* nach *Windows\SysWoW64\%DistLang%*

Wildcards werden nicht unterstützt.

Fehlermeldung, falls Datei nicht existiert.

ReqWin,<Datei>

Datei - Zu kopierende Datei im Verzeichnis Windows (Install Wim-Cache) + mui (Fallback en-US)

Beispiel:

ReqWin,explorer.exe

Kopiert *explorer.exe* nach *Windows.*

Kopiert *explorer.exe.mui* nach *Windows\%DistLang%.*

Wildcards werden nicht unterstützt.

Fehlermeldung, falls Datei nicht existiert.

[zurück zum Inhaltsverzeichnis](#)

3. Skriptsteuerung

3.1. Standard-Abschnitte eines Skripts

Im Normalfall erfolgt die Gliederung von Skripten in die Abschnitte

[Main]
[Interface]
[Process]

Erstellung weiterer benötigter Abschnitte ist jederzeit möglich.

[zurück zum Inhaltsverzeichnis](#)

3.1.1. Abschnitt **[Main]**

Bevorzugterweise sollten diese Einträge vorhanden sein:

[Main]		
Title=	<i>Title des Programms</i>	
Description=	<i>kurze Beschreibung des Programms</i>	
Selected=	<i>true</i>	Skript ist im Interface in der Baumansicht aktiviert
	<i>False</i>	Skript ist im Interface in der Baumansicht deaktiviert
	<i>None</i>	Skript wird ausgeführt, ist nicht aktivier- oder deaktivierbar
	<i>Noprocess</i>	Skript wird angezeigt, aber nicht ausgeführt
Level=	<i>Skript-Ebene in der Baumansicht von LiveSystem pro</i>	
	<i>1 - Build</i>	
	<i>2 - Shells</i>	

	3	-	Components
	4	-	Customize + Drivers
	5	-	Apps
	8	-	Finish + Virtual Test
	9	-	Utils
Author=	Name des Skript-Autors		
Version=	Fortlaufende Versionsnummer des Skripts		
Date=	Erstellungsdatum oder Update-Datum des Skripts		

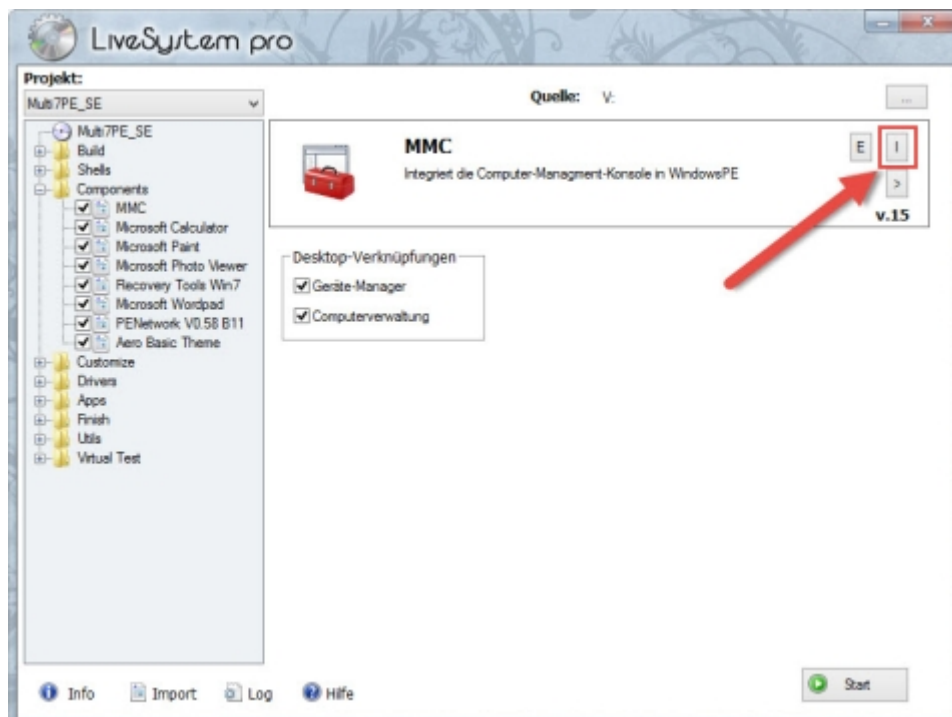
Einträge im Abschnitt [Main] zur Steuerung anderer Skripte:

Disable=	xyz.srt	Deaktiviert das angegebene Skript
Enable=	xyz.srt	Aktiviert das angegebene Skript

[zurück zum Inhaltsverzeichnis](#)

3.1.2. Abschnitt [Interface]

In diesem Abschnitt erfolgt die optische Darstellung der benötigten Texte und Auswahlboxen. Aufruf der Auswahlboxen erfolgt durch Anklicken des Buttons **I** (siehe Bild):



Erstellung mehrsprachiger Interfaces

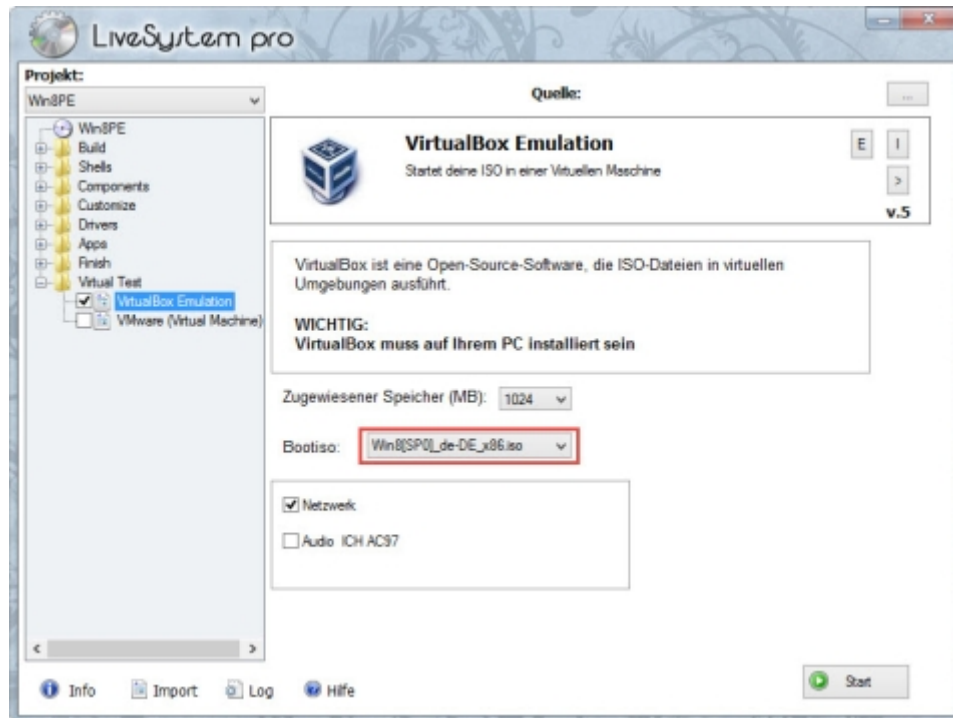
Weitere Interfaces können leicht integriert werden nach diesem Muster:

[Interface]	englische Texte und Auswahlboxen
[Interface.07]	deutsche Texte und Auswahlboxen
[Interface.13]	niederländische Texte und Auswahlboxen

Der Builder verwendet automatisch die korrekte Version.

[zurück zum Inhaltsverzeichnis](#)

3.1.3. Abschnitt [Process]



Beispielcode:

```
[Interface.07]
pScrollBar3=,1,0,0,20,270,215,20
```

```
[Script-Select]
search,%Var%,File,%ISODir%,*.iso
Interface,set,pScrollBar3,%Var%
```

Resultat:

- Bei Auswahl des Skripts in der Baumansicht wird die Scrollbox aktualisiert
- pScrollBar3 wird aktualisiert:

```
[Interface.07]
pScrollBar3=Win7[SP0]_de-DE_x86.iso,1,0,0,20,270,215,20,Win7[SP0]_de-DE_x86.iso
```

[zurück zum Inhaltsverzeichnis](#)