

opsi Version 4.0.2 (Teil 2) Release Notes

Inhaltsverzeichnis

1	Copyright	1
2	Übersicht der Neuerungen	2
3	Hinweise zur Installation	4
4	Werkzeug: Das Managementinterface <i>opsi-configed</i>	5
4.1	Auswahl von Clients	5
5	Erweiterungen bei der Treiberintegration	7
5.1	Vereinfachte Treiberintegration in die automatische Windowsinstallation	7
5.1.1	Allgemeine Treiber Pakete	7
5.1.2	Treiber die zu Ihrer Hardware gehören aber nicht speziell zu geordnet sind	8
5.1.3	Treiber die manuell Rechnern zu geordnet sind	8
5.1.4	Treiber die über die Felder <vendor>/<model> der Inventarisierung automatisch den Rechnern zu geordnet werden.	8
5.1.5	Struktur des Treiber Verzeichnisses und Ablage der Treiber:	9
5.1.6	Abarbeitung der unterschiedlichen Ebenen der Treiberintegration	9
5.1.7	Treiber hinzufügen und prüfen	9
6	Inventarisierung	14
6.1	Hardware Inventarisierung	14
6.2	Software Inventarisierung	17
7	Anpassungen am neuen opsi-linux-bootimage	18
8	Anpassungen am opsiconfd	19
9	opsi PAM Authentifizierung	20
10	/etc/opsi/opsi.conf	21
11	Silent oder Unattended Schalter finden	22
11.1	Skript für Installationen im Kontext eines lokalen Administrators	23

12 opsi-winst 4.11.3	24
12.1 Start und Aufrufparameter	24
12.2 Globale Textkonstanten	24
12.2.1 <i>opsi-winst</i> Pfad und Verzeichnisse	24
12.3 Primäre Sektionen	24
12.3.1 Reaktion auf Fehler	24
12.3.2 Fenster Modus	25
12.3.3 Include Kommandos	25
12.3.3.1 Include Kommandos: Syntax	25
12.3.3.2 Include Kommandos: Library	26
12.3.4 String-Verarbeitung	28
12.3.5 Weitere String-Funktionen	31
12.3.6 Boolesche Ausdrücke	32
12.3.7 String-Listen aus der Registry	33
12.3.8 String-Listen aus Produkt Properties	35
12.3.9 Sonstige String-Listen	35
12.3.10 Fehlgeschlagene Installation anzeigen	36
12.3.11 Kommandos zur Steuerung des Logging	36
12.4 Sekundäre Sektionen	36
12.4.1 Patches-Sektionen	36
12.4.1.1 Aufrufparameter	36
12.4.2 ExecWith Sektionen	36
12.4.3 WinBatch-Sektionen	37
12.4.3.1 Aufrufparameter (Modifier)	37
12.5 Sonstiges	39
13 Sonstiges	40
13.1 Changelogs:	40
13.1.1 Changelog configed	40
13.1.2 Changelog opsi-winst	42
13.1.3 Changelog hwaudit	45
13.1.4 Changelog swaudit	45
13.1.5 Changelog jedit	46
13.1.6 Changelog opsi-adminutils	46
13.1.7 Changelog opsi-template	47
13.1.8 Changelog opsi-template-with-admin	48
13.1.9 Changelog opsi-client-agent	48
13.1.10 Changelog windows netboot products	48
13.1.11 Changelog python-opsi	50

13.1.12 Changelog opsipxeconfd	50
13.1.13 Changelog opsiconfd	51
13.1.14 Changelog opsi-utils	51
13.1.15 Changelog opsi-linux-bootimage	52
13.1.16 Changelog opsi-depotserver	52
13.1.17 Changelog opsi4ucs	52
13.1.18 Changelog opsi-atftp	53

Kapitel 1

Copyright

Das Copyright an diesem Handbuch liegt bei der uib gmbh in Mainz.

Dieses Handbuch ist veröffentlicht unter der creative commons Lizenz
Namensnennung - Weitergabe unter gleichen Bedingungen (by-sa).



Eine Beschreibung der Lizenz finden Sie hier:

<http://creativecommons.org/licenses/by-sa/3.0/de/>

Der rechtsverbindliche Text der Lizenz ist hier:

<http://creativecommons.org/licenses/by-sa/3.0/de/legalcode>

Die Software von opsi ist in weiten Teilen Open Source.

Nicht Open Source sind die Teile des Quellcodes, welche neue Erweiterungen enthalten die noch unter Kofinanzierung stehen, also noch nicht bezahlt sind.

siehe auch:

<http://uib.de/www/kofinanziert/index.html>

Der restliche Quellcode ist veröffentlicht unter der GPLv3:



Der rechtsverbindliche Text der GPLv3 Lizenz ist hier:

<http://www.gnu.org/licenses/gpl.html>

Deutsche Übersetzung:

<http://www.gnu.de/documents/gpl.de.html>

Der Name *opsi* ist eine eingetragene Marke der uib gmbh.

Das opsi-logo ist Eigentum der uib gmbh und darf nur mit ausdrücklicher Erlaubnis verwendet werden.

Kapitel 2

Übersicht der Neuerungen

Der zweite Teil des opsi Service Releases 4.0.2 weist eine Fülle von Neuerungen und Detailverbesserungen auf. Hier eine Übersicht:

- Freigabe von opsi 4.0.2 für folgende Distributionen:
 - Ubuntu 12.04
 - Redhat 6
 - CentOS 6
 - UCS 3.0 (inkl. digitec-opsi-listener)
 - OpenSuse 12.1
 - OpenSuse 12.2
- Freigabe einer aktualisierten vorkonfigurierten opsi-VM auf Basis von Ubuntu 12.04 und mit integriertem opsi-Nagios-Connector

Die entsprechenden Installationsanleitungen finden Sie wie immer im *opsi-getting-started* Handbuch

- opsi-configed
 - Neue Client-Suchmaske mit stark erweiterter Funktionalität
Siehe hierzu auch weiter unten
 - Suchaufrufe können gespeichert und erneut durchgeführt werden.
 - Directory-Baum im Treeview
Unterhalb von *Directory* kann ein hierarchischer Baum zur Clientverwaltung angelegt werden. Im Unterschied zu *Gruppen* kann und muss jeder Client innerhalb von *Directory* an exakt einer Stelle sein.
 - Gruppen können mit Beschreibungsfeld erzeugt werden, dessen Wert als Tooltip angezeigt wird.
 - Abweichungen der Clientwerte von den Serverdefaults bei clientspezifischen Produktproperties und Hostparametern werden durch Fettschrift angezeigt.
 - Farbliche Hervorhebung der Versionsinformation eines Produkt auf einem Client, wenn die Version von der auf dem Server abweicht.
 - *Produktname* als zusätzliche Spalte (konfigurierbar) in den Tabs *Produktkonfiguration* und *Netboot-Produkte*
 - *Localboot-Produkte zurücksetzen* im Kontextmenü des Client-Tabs löscht alle Informationen zu auf diesem Client installierten Lokalboot-Produkten
 - In den Tabs *Produktkonfiguration* und *Netboot-Produkte* lassen sich die Werte der *Report*-Spalte zurücksetzen. Insbesondere lässt sich auf diese Weise der *failed*-Status einer Produktinstallation löschen.
 - Verbindungsabfrage und der Sessioninformations-Abfrage sind in Threads verlagert, so dass Blockieren des opsi-configed bei "hängenden" Verbindungen vermieden wird.

- Über ein Häkchen in der Loginmaske kann die Kommunikation mit dem Webservice auf komprimierte Übertragung umgestellt werden.
- Hardwareinventarisierung
 - Bei *Dell* Rechnern wird nun auch der *Dell Express Code* ermittelt.
 - Die Hardware-Inventarisierung lässt sich nun durch den Aufruf von Kommandos erweitern. Beschreibung hierzu siehe unten.
- Treiberintegration
 - Automatische Zuordnung von Treiberpaketen zu Rechnern auf Basis von den Daten zu Vendor und Model aus der Hardware Inventarisierung. Beschreibung hierzu siehe unten.
- Konfigurationsdatei `/etc/opsi/opsi.conf`
- Windows Netboot Produkte
 - Security fix bei der WinPE Ausführung
 - Anpassung an die erweiterte Treiberintegration
- opsi setup detector
- opsi-winst:
 - Eine Fülle von Erweiterungen, welche weiter unten beschrieben sind.

Kapitel 3

Hinweise zur Installation

Die Produkte welche im Rahmen dieses Releases veröffentlicht werden, sind in etlichen Teilen voneinander abhängig. Sie sollten daher nicht versuchen nur Teile zu installieren.

Die Installation erfordert keine besonderen Arbeiten. Sie erfolgt im Rahmen der normalen Updates ihres Servers und der opsi-Produkte.

Sie sollten dabei zuerst den Server updaten und danach die opsi-Produkte. Dies erledigt der opsi-product-updater:

```
opsi-product-updater -i -vv
```

Sollten Sie eine Multidepot Umgebung haben, so machen Sie zunächst das Upgrade auf Ihrem config-server, bevor Sie die Depots upgraden.

Kapitel 4

Werkzeug: Das Managementinterface *opsi-configed*

4.1 Auswahl von Clients

In der Clientliste lassen sich ein oder mehrere Clients markieren und so für die (gemeinsame) Bearbeitung auswählen.

Die Funktion *Auswahl als Gruppe speichern* existiert nicht mehr. Stattdessen werden Clients mit der Maus zu einer bestehenden Gruppe, die im Treeview angezeigt wird, "gezogen" und so in diese Gruppe zusätzlich aufgenommen.

Mittels des Trichter-Icons bzw. über *Auswahl / Nur ausgewählte Clients anzeigen* kann die Anzeige der Clientliste auf die markierten Clients beschränkt werden

Im Client-Auswahldialog, der über *Auswahl / Auswahl definieren* gestartet wird, können Clientzusammenstellungen anhand wählbarer Kriterien erstellt werden.

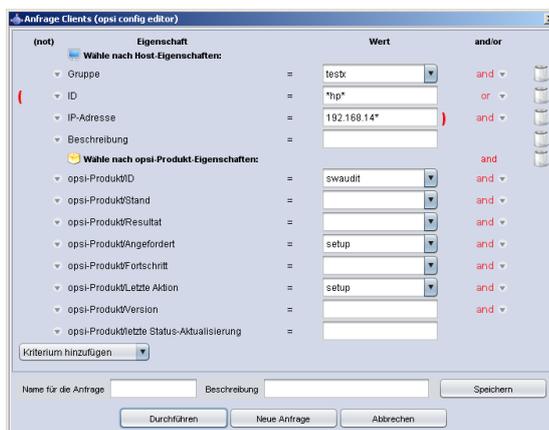


Abbildung 4.1: *opsi-configed*: Auswahldialog

Neben allgemeinen Eigenschaften der Clients existieren Kriterien für (auf dem PC gefundene als auch mit Opsi installierte) Software und Hardware. Bei Texteingaben kann * als Wildcard benutzt werden. Die einzelnen Kriterien können dabei mit logischem "und" oder "oder" verbunden werden, auch die Negation eines Kriteriums ist möglich.

Die so erstellten Suchen lassen sich auch speichern. Anschließend kann man sie über *Auswahl / Gespeicherte Suchen...* erneut aufrufen.

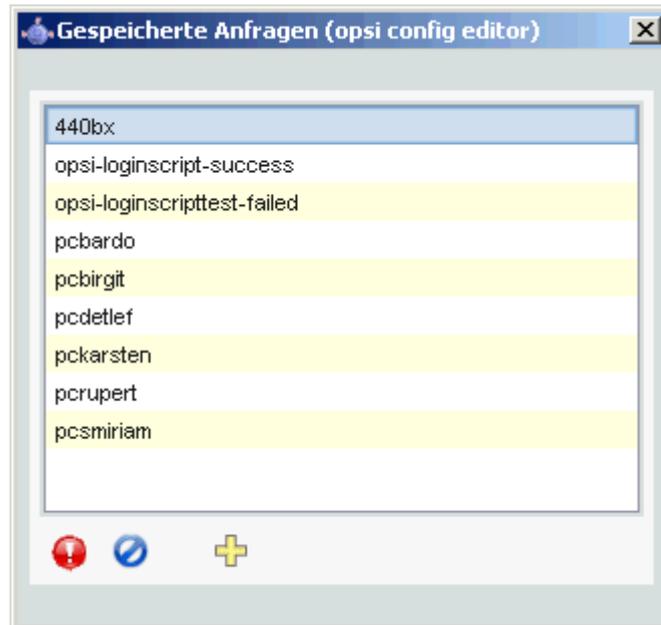


Abbildung 4.2: *opsi-configed*: Gespeicherte Suchen

Zusätzlich ist noch die Abfrage von Suchen über eine Kommandozeilen-Schnittstelle möglich, um sie auch Skripten zugänglich zu machen. Dazu wird *configed* mit dem Parameter "-qs" und anschließend dem Namen einer Suche aufgerufen. Lässt man den Namen der Suche weg, wird eine Liste der vorhandenen Suchen ausgegeben.

Kapitel 5

Erweiterungen bei der Treiberintegration

Das Kapitel zur Treiberintegration im *opsi-getting-started-manual* ist überarbeitet worden. Insbesondere sind die neuen Features für *Treiber die über die Felder <vendor>/<model> der Inventarisierung automatisch den Rechnern zu geordnet werden* beschrieben.

5.1 Vereinfachte Treiberintegration in die automatische Windowsinstallation

Administriert man einen Pool von PCs, die Geräte besitzen, deren Treiber nicht in der Windows-Standardinstallation enthalten sind, so ist es meist sinnvoll, diese Treiber direkt in die Installation zu integrieren. Bei Netzwerkgeräten kann dies teilweise sogar unumgänglich sein, denn ein startendes Windows ohne Netzwerkkarte ist für den Administrator nicht ohne weiteres erreichbar.

Opsi unterstützt Sie durch eine Automatisierung der Treibereinbindung und vereinfacht so die Bereitstellung der Treiber. Dabei müssen die Treiber nur in dem korrekten Verzeichnis abgelegt werden. Durch den Aufruf eines Scripts werden dann die Treiberverzeichnisse durchsucht und ein Katalog erstellt, anhand dessen das Bootimage automatisch die richtigen Treiber erkennen und einbinden kann. Dabei können sowohl Standard-Treiber, USB-Treiber, HD-Audio-Treiber wie auch Treiber für Festplattencontroller (Textmode Treiber) abgelegt und automatisch eingebunden werden.

Damit die Treiber sofort bei der Windowsinstallation mit installiert werden, müssen Sie in einer bestimmten Form auf dem Server hinterlegt werden. Hierzu sind Treiberverzeichnisse geeignet, die eine *.inf-Datei enthalten, die den Treiber für das Windows-Setupprogramm beschreibt. Irgendwelche in `setup.exe`, *.zip oder anders verpackten Treiber sind hier unbrauchbar. Mit dem Programm *double driver* (<http://www.boozet.org/dd.htm>) können Sie von einem installierten Rechner die Treiber im geeigneten Format extrahieren.

Es stehen mehrere Ebenen zur Bereitstellung von Treibern zur Verfügung:

- Allgemeine Treiber Pakete
- Treiber die zu Ihrer Hardware gehören aber nicht speziell zu geordnet sind
- Treiber die manuell Rechnern zu geordnet sind
- Treiber die über die Felder <vendor>/<model> der Inventarisierung automatisch den Rechnern zu geordnet werden.

Wie diese unterschiedlichen Ebenen verwendet werden können ist im folgenden beschrieben:

5.1.1 Allgemeine Treiber Pakete

Wenn die Hardwareausstattung sehr heterogen ist, kann es sinnvoll sein mit allgemeinen Treiberpaketen zu arbeiten. Allgemeine Treiber legen Sie ab unter `./drivers/drivers`.

Solche allg. Treiber Pakete finden Sie <http://driverpacks.net/> .

Laden Sie die gewünschten Treiber Pakete in ein temporäres Verzeichnis herunter und entpacken die Treiberpakete mit:

```
./extract_driver_pack.py <pfad zu dem temporären Verzeichnis mit den komprimierten driverpacks>
```

Hiermit werden die Treiber entpackt und in das Verzeichnis `./drivers/drivers/` abgelegt.

Nachteil dieser Pakete ist, das sich hier auch Treiber finden welche zwar von der Beschreibung zu Ihrer Hardware passen aber nicht unbedingt mit Ihrer Hardware funktionieren.

Treiber welche im Verzeichnis `./drivers/drivers/` liegen, werden anhand der PCI-Kennungen (bzw. USB- oder HD_Audio-Kennung) in der Beschreibungsdatei des Treibers als zur Hardware passend erkannt und in das Windows Setup mit eingebunden.

5.1.2 Treiber die zu Ihrer Hardware gehören aber nicht speziell zu geordnet sind

Haben Sie nur wenige unterschiedliche Hardware zu unterstützen, so können Sie die Treiber bei den Herstellern suchen.

Zusätzliche bzw. geprüfte Treiber gehören in jeweils eigene Verzeichnisse (Name und Tiefe der Verzeichnisstruktur egal) unterhalb des Verzeichnisses

```
./drivers/drivers/preferred.
```

Treiber welche im Verzeichnis `./drivers/drivers/preferred` liegen, werden gegenüber den Treibern in `./drivers/drivers/`` bevorzugt anhand der PCI-Kennungen (bzw. USB- oder HD_Audio-Kennung) in der Beschreibungsdatei des Treibers als zur Hardware passend erkannt und in das Windows Setup mit eingebunden.

Finden sich z.B. zu ein und derselben PCI-ID unterschiedliche Treiber unter `preferred`, so kann dies zu Problemen bei der Treiber Zuordnung führen. In diesem Fall ist eine direkte Zuordnung der Treiber zu den Geräten notwendig.

5.1.3 Treiber die manuell Rechnern zu geordnet sind

Zusätzliche Treiber, die unabhängig von ihrer Zuordnung bzw. Erkennung über die PCI- oder USB-IDs installiert werden sollen, gehören in jeweils eigene Verzeichnisse (Name und Tiefe der Verzeichnisstruktur egal) unterhalb des Verzeichnisses `./drivers/drivers/additional`. Über das Produkt-Property `additional_drivers` von können Sie einen oder mehrere Pfade von Treiberverzeichnissen innerhalb von `./drivers/drivers/additional` einem Client zu ordnen. Im Produkt-Property `additional_drivers` angegebene Verzeichnisse werden rekursiv durchsucht und alle enthaltenen Treiber eingebunden. Dabei wird auch symbolischen Links gefolgt. Dies können Sie nutzen, um für bestimmte Rechner-Typen ein Verzeichnis zu erstellen (z.B. dell-optiplex-815).

Wird in den über `additional_drivers` angegebenen Treiberverzeichnissen ein Treiber für ein vorhandenes PCI-Gerät (oder HD-Audio, USB) gefunden, so wird für dieses Gerät kein weiterer Treiber aus `drivers/preferred/` oder `drivers/` mehr eingebunden. Damit hat `additional_drivers` nicht nur die Funktion Treiber hinzuzufügen, welche über die normale Treibererkennung nicht gefunden würden. Darüber hinaus haben die Treiber welche dem client via `additional_drivers` zugeordnet werden auch Vorrang vor Treibern aus anderen Verzeichnissen (`additional_drivers` ist sozusagen auch *super-preferred*).

5.1.4 Treiber die über die Felder <vendor>/<model> der Inventarisierung automatisch den Rechnern zu geordnet werden.

Der im vorigen Abschnitt beschriebene Mechanismus der direkten Zuordnung von Treibern zu Geräten, kann seit dem 2 Teil des Service Release opsi 4.0.2 automatisiert werden. Dazu wird in dem Verzeichnis `./drivers/drivers/additional/byAudit` nach einem Verzeichnisnamen gesucht der dem bei der Hardwareinventarisierung gefundenen *Vendor* entspricht. In diesem *Vendor* Verzeichnis wird nun nach einem Verzeichnisnamen gesucht, das dem bei der Hardwareinventarisierung gefundenen *Model* entspricht. Wird ein solche Verzeichnis gefunden, so wird diese Verzeichnis genauso behandelt, als sei es über das Produkt-Property `additional_drivers` manuell zugewiesen.

Der Verzeichnisname `byAudit` ist case sensitive. Die Verzeichnisnamen für *Vendor* und *Model* sind nicht case sensitive (Dell oder dELL ist egal)

5.1.5 Struktur des Treiber Verzeichnisses und Ablage der Treiber:

```
/opt/  
  !-pcbin/  
    !-install/  
      !-winxpro/  
        !-drivers  
          |-classes/           (Links auf Treiber über Geräteklassen)  
          |-hdaudioids/       (Links auf HD-Audio Treiber)  
          |-pciids/           (Links auf Treiber über PCI-Kennung)  
          |-pci.ids           (PCI Datenbank)  
          |-usbids/           (Links auf Treiber über USB-Kennung)  
          |-usb.ids           (USB Datenbank)  
          |-vendors/         (Links auf Treiber über Hersteller)  
          !-drivers           (Platz für allg. Treiber Packs)  
            |-additional/     (Für manuell zugeordnete Treiber)  
              |-byAudit/      Modell spezifische Treiber welche  
                |-<vendor>      über die Hardwareinventarisierung  
                  |-<model>      zugeordnet werden  
              |-buildin/      (Daten aus dem i386 Baum)  
              |-preferred/    (geprüfte Treiber)  
              |-exclude/      (ausgeschlossene Treiber)  
              !-mydriverpacks/ (Beispiel Treiber Pack)
```

5.1.6 Abarbeitung der unterschiedlichen Ebenen der Treiberintegration

Als oberste Priorität werden alle Treiber eingebunden, welche über das Property *additional_drivers* bzw. die über die Inventarisierungsdaten in `./drivers/drivers/additional/byAudit` gefunden werden. Im Rahmen der Einbindung von Treibern wird geprüft für welche der Hardware eines Geräts (anhand der PCI-,USB-,HDAudio-Kennungen) hierdurch ein Treiber bereit gestellt wurde. Nur für Geräte für die auf diese Weise noch kein Treiber bereitgestellt wurde wird über die nachfolgenden Methode ein Treiber gesucht.

Für Geräte denen nicht über *additional_drivers* (bzw. *byAudit*) ein Treiber zu geordnet wurde wird anhand der PCI Kennung (bzw. USB-, HDAudio-Kennung) ein passender Treiber gesucht und eingebunden.

Einbindung von Treiber bedeutet dabei:

- Der Treiber wird auf die lokale Festplatte nach `c:\drv\ kopiert.`
- Dem Windows Setup wird in der unattended Datei mitgeteilt, in den Verzeichnissen unterhalb von `c:\drv\` nach passenden Treibern zu suchen.

5.1.7 Treiber hinzufügen und prüfen

Nach jedem Hinzufügen eines Treibers oder jeden anderen Änderung im `./drivers/drivers` Verzeichnis (oder darunter) rufen Sie im Stammverzeichnis des Netboot Produktes Verzeichnis folgenden Befehl auf, um die Rechte korrekt zu setzen:

```
opsi-setup --set-rights ./drivers
```

Danach rufen Sie das Script `./create_driver_links.py` auf. Dieses durchsucht die Verzeichnisse unterhalb von `./drivers/drivers` und erzeugt eine Reihe von Links anhand deren die Zuordnung der Treiber zu bestimmter Hardware (PCI-IDs, USB-IDs, HD-Audio-IDs) zu erkennen ist. Die Treiber aus dem preferred Verzeichnis werden von dem Script bevorzugt verwendet.

Das `setup.py` Script des Bootimages untersucht die Hardware des zu installierenden Computers und identifiziert die notwendigen Treiber. Diese werden dann auf die Platte kopiert und die `unattend.txt` entsprechend gepatcht. Das Script `create_driver_links.py` durchsucht auch bei NT5 Produkten einmalig den *i386* Baum und extrahiert die Inf-Dateien der von Windows mitgelieferten Treiber nach *windows_builtin*. Sollten Sie am i386-Baum eine Änderung vornehmen

(z.B. durch das Einspielen eines Servicepacks) so löschen Sie dieses Verzeichnis und führen `create_driver_links.py` erneut aus. Bei NT6 Produkten werden die Treiber welche sich im WinPE finden als `windows_builtin` erkannt.

Liegt zu einem Client eine Hardware-Inventarisierung vor, so kann über den Befehl:

```
./show_drivers.py <clientname>
```

ausgegeben werden, welche Treiber das Bootimage via PCI-IDs, USB-IDs, HD-Audio-IDs und `additional_drivers` (bzw. `byAudit`) zur Installation auswählen würde und zu welcher Hardware noch kein Treiber bereit steht.

Kontrollieren Sie die Ausgabe von `show_drivers.py` um zu prüfen ob die gewünschten Treiber eingebunden werden.

Es kann vorkommen, das Treiberverzeichnisse von Herstellern Treiber für unterschiedliche Betriebssystemversionen (Vista / Win7) oder Konfigurationen (SATA / SATA-Raid) enthalten. Das `create_driver_links.py` script kann das nicht unterscheiden. Wenn Sie die Vermutung haben, das ein verlinkter Treiber falsch ist, so verschieben Sie diesen Treiber in das Verzeichnis `drivers/exclude` und führen `create_driver_links.py` erneut aus. Treiber die in `drivers/exclude` liegen werden bei der Treiberintegration nicht berücksichtigt.

Beispiel einer `show_drivers.py` Ausgabe:

```
./show_drivers.py pcdummy

PCI-Devices
[(Standardssystemgeräte), PCI Standard-PCI-zu-PCI-Brücke]
  No driver - device directory /opt/pcbin/install/winxpro/drivers/pciids/1022/9602 not found
[ATI Technologies Inc., Rage Fury Pro (Microsoft Corporation)]
  Using build-in windows driver
[(Standard-IDE-ATA/ATAPI-Controller), Standard-Zweikanal-PCI-IDE-Controller]
  /opt/pcbin/install/winxpro/drivers/drivers/D/M/N/123
[Realtek Semiconductor Corp., Realtek RTL8168C(P)/8111C(P) PCI-E Gigabit Ethernet NIC]
  /opt/pcbin/install/winxpro/drivers/drivers/preferred/realtek_gigabit_net_8111_8168b
[IEEE 1394 OHCI-konformer Hostcontroller-Hersteller, OHCI-konformer IEEE 1394-Hostcontroller]
  No driver - device directory '/opt/pcbin/install/winxpro/drivers/pciids/197B/2380' not found
[Advanced Micro Devices, Inc., AMD AHCI Compatible RAID Controller]
  /opt/pcbin/install/winxpro/drivers/drivers/preferred/ati_raid_sb7xx
[(Standard-USB-Hostcontroller), Standard OpenHCD USB-Hostcontroller]
  No driver - device directory '/opt/pcbin/install/winxpro/drivers/pciids/1002/4397' not found
[ATI Technologies Inc, ATI SMBus]
  /opt/pcbin/install/winxpro/drivers/drivers/preferred/ati_smbus

USB-Devices
[(Standard-USB-Hostcontroller), USB-Verbundgerät]
  /opt/pcbin/install/winxpro/drivers/drivers/preferred/brother_844x_pGerb
[Microsoft, USB-Druckerunterstützung]
  /opt/pcbin/install/winxpro/drivers/drivers/preferred/brother_844x_pGerb

Additional drivers
[ati_hdaudio_azalia]
  /opt/pcbin/install/winxpro/drivers/drivers/additional/ati_hdaudio_azalia
```

Beispiel mit `additional_drivers`:

```
./show_drivers.py e5800
Manually selected drivers (additional)
[hp_e5800]
[/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/hp_e5800/sp52852/Vista64/HDXHPAI3.inf]
[/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/hp_e5800/sp52852/Vista64/HDX861A.inf]
[/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/hp_e5800/sp52852/Vista64/HDXHPAI1.inf]
[/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/hp_e5800/sp52852/Vista64/HDXCPC.inf]
[/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/hp_e5800/sp52852/Vista64/HDXHPAI2.inf]
[/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/hp_e5800/sp50134/autorun.inf]
[/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/hp_e5800/sp50134/ibxHDMI/IntcDAud.inf]
[/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/hp_e5800/sp50134/HDMI/IntcHdmi.inf]
[/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/hp_e5800/sp50134/Graphics/kit24890.inf]
[/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/hp_e5800/sp50134/IIPS/Impcd.inf]
[/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/hp_e5800/sp54284/Realtek_64bit/hp64win7\
.inf]
```

PCI-Devices

```
[8086:27C8] Intel : Intel(R) N10/ICH7 Family USB Universal Host Controller - 27C8
/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/preferred/R293337/WIN7
[8086:27DA] Intel : Intel(R) N10/ICH7 Family SMBus Controller - 27DA
/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/preferred/R293337/WIN7
[8086:27C9] Intel : Intel(R) N10/ICH7 Family USB Universal Host Controller - 27C9
/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/preferred/R293337/WIN7
[8086:27DF] Intel : Intel(R) ICH7 Family Ultra ATA Storage Controllers - 27DF
/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/preferred/R293337/WIN7
[8086:27CA] Intel : Intel(R) N10/ICH7 Family USB Universal Host Controller - 27CA
/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/preferred/R293337/WIN7
[8086:2E30] Intel : Intel(R) 4 Series Chipset Processor to I/O Controller - 2E30
/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/not_preferred/x64/C/Intel/1
[8086:27CB] Intel : Intel(R) N10/ICH7 Family USB Universal Host Controller - 27CB
/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/preferred/R293337/WIN7
[8086:2E32] Intel Corporation : Intel(R) G41 Express Chipset
Manually selected [hp_e5800] /opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/hp_e5800/\
sp50134/Graphics
[8086:27CC] Intel : Intel(R) N10/ICH7 Family USB2 Enhanced Host Controller - 27CC
/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/preferred/R293337/WIN7
[8086:244E] Intel : Intel(R) 82801 PCI-Brücke - 244E
Using build-in windows driver
This driver will not be integrated, because same device already integrated in: '/opt/pcbin/install/win7-x64-\
professional-msdn/drivers/drivers/not_preferred/x64/C/Intel/1/dmi_pci.inf'
[8086:27D0] Intel : Intel(R) N10/ICH7 Family PCI Express Root Port - 27D0
/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/preferred/R293337/WIN7
[8086:27B8] Intel : Intel(R) ICH7 Family LPC Interface Controller - 27B8
/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/preferred/R293337/WIN7
[8086:27D2] Intel : Intel(R) N10/ICH7 Family PCI Express Root Port - 27D2
/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/preferred/R293337/WIN7
[8086:27C0] Intel : Intel(R) N10/ICH7 Family Serial ATA Storage Controller - 27C0
/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/preferred/R293337/WIN7
[8086:27D8] Microsoft : High Definition Audio-Controller
No driver - device directory '/opt/pcbin/install/win7-x64-professional-msdn/drivers/pciids/8086/27D8' not found
[10EC:8136] Realtek : Realtek RTL8102E/RTL8103E-Familie-PCI-E-Fast-Ethernet-NIC (NDIS 6.20)
Manually selected [hp_e5800] /opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/hp_e5800/\
sp54284/Realtek 64bit
```

USB-Devices

```
[0461:0010] (Standardsystemgeräte) : USB-Eingabegerät
No driver - vendor directory '/opt/pcbin/install/win7-x64-professional-msdn/drivers/usbids/0461' not found
[0461:4D20] (Standardsystemgeräte) : USB-Eingabegerät
No driver - vendor directory '/opt/pcbin/install/win7-x64-professional-msdn/drivers/usbids/0461' not found
[058F:6366] Kompatibles USB-Speichergerät : USB-Massenspeichergerät
No driver - vendor directory '/opt/pcbin/install/win7-x64-professional-msdn/drivers/usbids/058F' not found
[0461:0010] (Standard-USB-Hostcontroller) : USB-Verbundgerät
No driver - vendor directory '/opt/pcbin/install/win7-x64-professional-msdn/drivers/usbids/0461' not found
```

HD-Audio-Devices

```
[10EC:0662] Realtek High Definition Audio
Manually selected [hp_e5800] /opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/hp_e5800/\
sp52852/Vista64
```

Beispiel mit *byAudit*:

```
./show_drivers.py pctr5detlef
Manually selected drivers (additional)
/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/byAudit/nvidia/awrdacpi]
[/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/byAudit/nvidia/awrdacpi/pctr5detlef/\
Display/Radeon X300-X550-X1050 Series Secondary (Microsoft Corporation - WDDM)/atiilhg.inf]
[/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/byAudit/nvidia/awrdacpi/pctr5detlef/\
Display/Radeon X300-X550-X1050 Series (Microsoft Corporation - WDDM)/atiilhg.inf]
[/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/byAudit/nvidia/awrdacpi/pctr5detlef/\
MEDIA/Realtek AC'97 Audio/oem21.inf]
```

PCI-Devices

```
[1002:5B70] ATI Technologies Inc. : Radeon X300/X550/X1050 Series Secondary (Microsoft Corporation - WDDM)
```

```

Manually selected [/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/byAudit/nvidia/\
awrdacpi] /opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/byAudit/nvidia/awrdacpi/\
pctry5detlef/Display/Radeon X300-X550-X1050 Series Secondary (Microsoft Corporation - WDDM)
Multiple selected [/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/byAudit/nvidia/\
awrdacpi] /opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/byAudit/nvidia/awrdacpi/\
pctry5detlef/Display/Radeon X300-X550-X1050 Series (Microsoft Corporation - WDDM)
[10DE:0053] (Standard-IDE-ATA/ATAPI-Controller) : Standard-Zweikanal-PCI-IDE-Controller
No driver - device directory '/opt/pcbin/install/win7-x64-professional-msdn/drivers/pciids/10DE/0053' not found
[10DE:005D] (Standardsystemgeräte) : PCI Standard-PCI-zu-PCI-Brücke
No driver - device directory '/opt/pcbin/install/win7-x64-professional-msdn/drivers/pciids/10DE/005D' not found
[1022:1100] AMD : AMD HyperTransport(tm)-Konfiguration
Using build-in windows driver
[10DE:0054] (Standard-IDE-ATA/ATAPI-Controller) : Standard-Zweikanal-PCI-IDE-Controller
/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/preferred/evb_potsdam_fsc_esprimo_p625/\
FTS_NVIDIASATAAHCIDRIVERVISTA64V103042MCP78__1026963/NVIDIA_SATA_AHCI_DRIVER_Vista64_V10.3.0.42_MCP78 (textmode \
capable)
[1022:1101] AMD : AMD-Adresszuordnungskonfiguration
Using build-in windows driver
[10DE:0055] (Standard-IDE-ATA/ATAPI-Controller) : Standard-Zweikanal-PCI-IDE-Controller
/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/preferred/evb_potsdam_fsc_esprimo_p625/\
FTS_NVIDIASATAAHCIDRIVERVISTA64V103042MCP78__1026963/NVIDIA_SATA_AHCI_DRIVER_Vista64_V10.3.0.42_MCP78 (textmode \
capable)
[1022:1102] AMD : AMD DRAM und HyperTransport(tm)-Nachverfolgungsmoduskonfiguration
Using build-in windows driver
[10DE:0057] NVIDIA : NVIDIA nForce-Netzwerkcontroller
Using build-in windows driver
[1022:1103] AMD : Sonstige AMD-Konfiguration
Using build-in windows driver
[10DE:0059] Realtek : Realtek AC'97 Audio
Manually selected [/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/byAudit/nvidia/\
awrdacpi] /opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/byAudit/nvidia/awrdacpi/\
pctry5detlef/MEDIA/Realtek AC'97 Audio
[10DE:005E] NVIDIA : NVIDIA nForce4 HyperTransport-Brücke
/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/preferred/ga-ma78-pcbon4/chipset_win7-64/SMBUS
[104C:8025] Texas Instruments : OHCI-konformer Texas Instruments 1394-Hostcontroller
No driver - device directory '/opt/pcbin/install/win7-x64-professional-msdn/drivers/pciids/104C/8025' not found
[10DE:005A] (Standard-USB-Hostcontroller) : Standard OpenHCD USB-Hostcontroller
No driver - device directory '/opt/pcbin/install/win7-x64-professional-msdn/drivers/pciids/10DE/005A' not found
[10DE:0050] (Standardsystemgeräte) : PCI Standard-ISA-Brücke
No driver - device directory '/opt/pcbin/install/win7-x64-professional-msdn/drivers/pciids/10DE/0050' not found
[10DE:005B] (Standard-USB-Hostcontroller) : Standard PCI-zu-USB erweiterter Hostcontroller
No driver - device directory '/opt/pcbin/install/win7-x64-professional-msdn/drivers/pciids/10DE/005B' not found
[1002:5B60] ATI Technologies Inc. : Radeon X300/X550/X1050 Series (Microsoft Corporation - WDDM)
Manually selected [/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/byAudit/nvidia/\
awrdacpi] /opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/byAudit/nvidia/awrdacpi/\
pctry5detlef/Display/Radeon X300-X550-X1050 Series Secondary (Microsoft Corporation - WDDM)
Multiple selected [/opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/byAudit/nvidia/\
awrdacpi] /opt/pcbin/install/win7-x64-professional-msdn/drivers/drivers/additional/byAudit/nvidia/awrdacpi/\
pctry5detlef/Display/Radeon X300-X550-X1050 Series (Microsoft Corporation - WDDM)
[10DE:0052] NVIDIA : NVIDIA nForce PCI-Systemverwaltung
Using build-in windows driver
[10DE:005C] (Standardsystemgeräte) : PCI Standard-PCI-zu-PCI-Brücke
No driver - device directory '/opt/pcbin/install/win7-x64-professional-msdn/drivers/pciids/10DE/005C' not found

```

USB-Devices

```

[1241:1111] (Standardsystemgeräte) : USB-Eingabegerät
No driver - vendor directory '/opt/pcbin/install/win7-x64-professional-msdn/drivers/usbids/1241' not found

```

HD-Audio-Devices

```

No devices installed

```

TIPPS

- Treiberverzeichnisname NDIS1 sind Vista-Treiber ; NDIS2 sind Win7-Treiber
- Manche Chipsatztreiber enthalten Beschreibungsdateien, welche sehr viel Hardware auflisten ohne hier- zu tatsächlich Treiber zu liefern. Ein Beispiel hierfür ist z.B. die cougar.inf von Intel. Wird ein solches

Pseudo-Treiber Verzeichnis per *additional_drivers* (bzw. *byAudit*) zu geordnet, so führt dies dazu, das die hier aufgeführte Hardware von der weiteren suche nach Treibern ausgeschlossen wird. Verschieben Sie solche Treiber Verzeichnisse besser nach **preferred**.

- SATA-Treiber und SATA-RAID Treiber beziehen sich auf die selbe PCI-Kennung. Ein SATA-RAID Treiber wird aber mit einem Einzel-Platten System nicht funktionieren.
- Kontrollieren Sie die Ausgabe von `./show_drivers` genau !

Kapitel 6

Inventarisierung

Zur Inventarisierung stehen die Localbootprodukte `hwaudit` und `swaudit` sowie das Netboot Produkt `hwinvent` zur Verfügung.

6.1 Hardware Inventarisierung

Die Hardwareinventarisierung ist unter opsi über eine Konfigurationsdatei gesteuert. Das bedeutet, dass die Information wie und welche Daten erhoben werden, nicht in den entsprechenden Produkten `hwaudit` und `hwinvent` fest verdrahtet sind. Vielmehr werden diese Produkte über eine Konfigurationsdatei gesteuert. Dazu wird die Konfigurationsdatei bei jeder Ausführung über den opsi Webservice eingelesen und interpretiert. Gleichzeitig steuert diese Konfigurationsdatei auch den Aufbau der Datenbank, so dass eine Erweiterung dieser Konfigurationsdatei auch eine Erweiterung der Datenhaltung nach sich zieht.

Die Konfigurationsdatei ist die `/etc/opsi/hwaudit/opsihwaudit.conf`.

In dieser Datei werden alle zu Inventarisierenden Objekte definiert und beschreiben, wie die zu diesem Objekt gehörenden Daten zu erheben sind (unter Linux und unter Windows). Gleichzeitig wird darüber auch die dazu gehörige Datenstruktur definiert. Zur Vereinfachung enthält diese Konfigurationsdatei Vererbungsmechanismen die an eine Objektorientierung angelehnt sind. Hintergrund hierfür ist die Tatsache, dass viele Objekte identische Datenfelder wie z.B. `Name` und `Vendor` enthalten. Diese allgemeinen Informationen werden so in *virtual* Hardwareklassen definiert. Die eigentlichen Inventarisierungsobjekte sind dann *structural* Hardwareklassen, welche viele Eigenschaften von übergeordneten *virtual* Klassen erben können.

Zur Erläuterung dieses Mechanismus ein Beispiel:

So definiert die Konfigurationsdatei zunächst eine *virtual Class* Namens `"BASIC_INFO"`. Diese definiert die Eigenschaften (*Values*):

- `"name"`
- `"description"`

Als nächstes folgt die *virtual Class* Namens `"HARDWARE_DEVICE"` welche alle Eigenschaften von `"BASIC_INFO"` erbt und folgende zusätzliche definiert:

- `"vendor"`
- `"model"`
- `"serialNumber"`

Als nächstes folgt als erstes Objekt welche wir in der Inventarisierung auch finden, die erste *structural Class* Namens `"COMPUTER_SYSTEM"`, welche alle Eigenschaften von `"HARDWARE_DEVICE"` erbt und folgende zusätzliche definiert bzw. überschreibt:

- "name"
- "systemType"
- "totalPhysicalMemory"

Im Rahmen der Definition einer Klasse und ihrer *Values* werden verschiedene Eigenschaften beschrieben:

- Klassen definition:
 - "Type"
ist "STRUCTURAL" oder "VIRTUAL"
 - "Super"
gibt die Klasse an von der geerbt wird.
 - "Opsi"
gibt den Namen der Klasse an, der auch später in opsi als Anzeigenamen verwendet wird.

Weiterhin können in der Klassendefinition angegeben werden, wie diese Daten erhoben werden. Diese Informationen können aber auch bei der Definition der *Values* stehen.

- Für die Inventarisierung unter Linux:
 - "Linux": "[<command>]<parameter>"
Ausführung des Kommandozeilenprogramms <command> mit dem Argument <parameter>.
 - "Python": "<python code with place holder>"
Ausführung des angegebenen Python codes wobei zunächst der Platzhalte durch die schon ermittelten Werte ersetzt wird.
- Für die Inventarisierung unter Windows:
 - "WMI": "<wmi select statement>"
auszuführende WMI Abfrage
 - "Cmd": "<Python text object with place holder>"
Der Platzhalter ist in diesem Fall der relative Pfad zu einem ausführbarem Programm, dessen Ausgabe den Platzhalter ersetzt.
 - "Registry": "[<registry key>] <value name>"
Aus der Registry wird in <registry key> der Wert von <value name> ausgelesen.
Das Auslesen der Registry erfolgt Architektur spezifisch. Das heißt, auf einem 64 Bit System wird der 64 Bit Zweig der Registry ausgelesen.
- Valuedefinition:
 - "Type": "<MySQL Datenbanktyp>"
<MySQL Datenbanktyp> gibt den Datentyp an in dem dieser Wert in der Datenbank angelegt wird.
 - "Scope": "<scope>"
das Feld <scope> wird folgendermaßen verwendet:
"g" bedeutet: Dieses Attribut ist bei allen Geräten dieses Typs gleich.
"i" bedeutet: Dieses Attribut kann bei Geräten dieses Typs unterschiedliche Werte haben.
 - "Opsi": "<id>"
dabei ist <id> der opsi interne Name des Feldes. Dieser kann zur Ausgabe über die Dateien in /etc/opsi/hwaudit/locales wiederum lokalisiert werden.
 - "WMI": "<id or command>"
dabei ist <id or command> entweder der Name unter dem der in der Klassen definition angegebene WMI Befehl den Wert ausgibt oder ein eigener WMI Befehl.

- "Linux": "<id>"
dabei ist <id> der Name unter dem der in der Klassen definition angegebene Linux Befehl den Wert ausgibt.
- "Condition": "<condition>"
dabei ist <condition> eine Bedingung die erfüllt sein muss, damit der *Value* ermittelt wird. So legt z.B. die <condition> "vendor=[dD]ell*" fest, das der schon erhobene Wert von "vendor" *Dell* oder *dell* enthalten muss.

Hierzu als Beispiel die Klasse "COMPUTER_SYSTEM":

```
{
  "Class": {
    "Type": "STRUCTURAL",
    "Super": [ "HARDWARE_DEVICE" ],
    "Opsi": "COMPUTER_SYSTEM",
    "WMI": "select * from Win32_ComputerSystem",
    "Linux": "[lshw]system"
  },
  "Values": [
    {
      "Type": "varchar(100)",
      "Scope": "i",
      "Opsi": "name",
      "WMI": "Name",
      "Linux": "id"
    },
    {
      "Type": "varchar(50)",
      "Scope": "i",
      "Opsi": "systemType",
      "WMI": "SystemType",
      "Linux": "configuration/chassis"
    },
    {
      "Type": "bigint",
      "Scope": "i",
      "Opsi": "totalPhysicalMemory",
      "WMI": "TotalPhysicalMemory",
      "Linux": "core/memory/size",
      "Unit": "Byte"
    },
    {
      "Type": "varchar(50)",
      "Scope": "i",
      "Opsi": "dellexpresscode",
      "Condition": "vendor=[dD]ell*",
      "Cmd": "#dellexpresscode\\dellexpresscode.exe#.split('=')[1]",
      "Python": "str(int(#{'COMPUTER_SYSTEM': 'serialNumber', 'CHASSIS': 'serialNumber'})#"
```

Besonders interessant ist hier der letzte Value "dellexpresscode":

Dieser ist nur sinnvoll, wenn es sich auch um einen Dell-Rechner handelt, daher die Condition.

Unter Windows wird das Kommandozeilen Programm **dellexpresscode.exe** ausgeführt, welches sich von der **hwaudit.exe** aus gesehen im Unterverzeichnis **dellexpresscode** befindet. Diese produziert eine Ausgabe in der Form: *dellexpresscode=123456789*. Durch den hinter dem Platzhalter befindlichen `.split('=')[1]` wird der Wert hinter dem Gleichheitszeichen verwendet.

Unter Linux wird geprüft in welchem Element (*COMPUTER_SYSTEM* oder *CHASSIS*) bei *serialNumber* ein Wert gefunden wurde, und dieser dann zur Berechnung des Dell expresscodes verwendet.

Die Opsi-Namen der Values werden über die Dateien `/etc/opsi/hwaudit/locales/*` übersetzt. Bsp. `/etc/opsi/hwaudit/locales/de_DE`:

```
COMPUTER_SYSTEM = Computer  
COMPUTER_SYSTEM.systemType = Typ
```

Der Klassenname `COMPUTER_SYSTEM` wird übersetzt in "Computer". Das Opsi-Attribut "systemType" der Klasse `COMPUTER_SYSTEM` wird übersetzt in "Typ". Abschliessend noch der Hinweis: Wenn ein neues Feld erzeugt wird, sollte man dieses in den locale-Dateien anlegen, auch wenn man den Begriff selber nicht übersetzt. Dadurch wird vermieden, dass bei der Laufzeit "Warning" Meldungen produziert werden.

6.2 Software Inventarisierung

Die Softwareinventarisierung findet über das Localbootprodukt `swaudit` statt. Dabei werden die Informationen aus dem Uninstallzweig der Registry erhoben und durch zusätzliche Informationen zu Hotfixes und Lizenzkeys ergänzt.

Der Quellcode dieses Paketes wird hier verwaltet:

<https://svn.opsi.org/listing.php?repname=swaudit>

Kapitel 7

Anpassungen am neuen opsi-linux-bootimage

Das *opsi-linux-bootimage* in der aktuellen Version: 20120728-1 wurde intern so umgebaut, dass nun **dhclient** statt **pump** für den initialen DHCP-Request verwendet wird.

Das Programm **pump** wurde bis jetzt im opsi eingesetzt, da **dhclient** den Flag **next-server** nicht auswertet. Da dieser Flag aber intern im opsi als opsiserver-Adresse verwendet wird wurde bisher das Programm **pump** im *opsi-linux-bootimage* verwendet. Da **pump** nicht konfigurierbar ist und ein "Hardreset" auf der Netzwerkkarte ausführt, um die IP-Adresse zu aktualisieren, kam es immer wieder zu Problemen in diversen Netzwerk-Infrastrukturen. Um diese Probleme zu vermeiden, wurde das neue *opsi-linux-bootimage* nun auf **dhclient** umgestellt.

Aus dieser Anpassung resultiert, dass die opsi-Boot-CD nach dem *opsi-server* fragen muss, auch wenn im Menü "mit DHCP" ausgewählt wird.

Kapitel 8

Anpassungen am `opsiconfd`

Der Webservice von opsi wurde für alle OS-Betriebssystem auf "noarch" umgebaut. Bei einigen RPM-Basierten Distributionen wie CentOS 5 wurde der `opsiconfd` architekturenspezifisch gebaut. Dies ist nicht notwendig, da der `opsiconfd` selbst kein Binary enthält. Dies wurde nun umgebaut.

Das bedeutet, dass Ihr System eventuell anzeigt, dass es kein Paket `opsiconfd-4.0.2.3-1.1.i386.rpm` findet. In diesem Fall müssen Sie das Paket `opsiconfd` manuell neu installieren.

Weiterhin wurden die `init`-Skripte für RPM-basierte Systeme angepasst, so dass der Runlevel 4 nicht mehr verwendet wird.

Kapitel 9

opsi PAM Authentifizierung

opsi verwendet zur Authentifizierung der User diverse PAM-Module. Bisher wurden für verschiedene Distributionen verschiedene PAM-Module verwendet. In der folgenden Auflistung werden die eingesetzten PAM Module aufgelistet:

Standard: common-auth

SLES: sshd

CentOS und RedHat: system-auth

RedHat 6: password-auth

Wie man aus der Liste erkennen kann, wurden diverse PAM-Konfigurationen verwendet, diese können sich aber je nach lokaler PAM Konfiguration wieder ändern. Da für diese Anpassungen immer ein Eingriff in den Code nötig war gibt es nun die Möglichkeit unter: `/etc/pam.d/` die Datei `opsi-auth` an zu legen und für opsi eine eigene PAM-Konfiguration zu hinterlegen. Wenn es diese Datei gibt, benutzt opsi automatisch diese Konfiguration.

Folgendes einfaches Beispiel soll das Verhalten verdeutlichen: Wenn Sie ein Debian/Ubuntu System betreiben und bei der Anmeldung am *opsi-configed* eine PAM-Fehlermeldung bekommen, obwohl mit den selben Benutzerdaten eine SSH Verbindung zum Server geöffnet werden kann, kann man die Datei `/etc/pam.d/opsi-auth` mit folgendem Inhalt erstellen:

```
@include sshd
```

Nach einem Neustart von `opsiconfd` benutzt opsi automatisch das `sshd-PAM-Modul` zur authentifizierung.

Kapitel 10

/etc/opsi/opsi.conf

Seit Version 4.0.2-2

Allgemeine opsi Konfigurationen.

Beispiel:

```
[groups]
fileadminigroup = pcpatch
```

Hintergrund: Die klassische Installationsvariante mit dem Benutzer: `pcpatch` mit der primären Gruppe: `pcpatch` funktioniert nicht mit Samba 4. Da Samba4 den grundlegenden Restriktionen von Active-Directory unterliegt, sind Gruppen mit der gleichen Bezeichnung wie User (wie in Unix/Linux üblich) nicht mehr erlaubt. Aus diesem Grund wurde eine neue Konfigurationsdatei eingeführt: `/etc/opsi/opsi.conf`, über die gesteuert wird, wie die Gruppe für den Samba-Zugriff auf die Freigaben bestimmt wird. So wird z.B. bei UCS 3 nun über diese Datei der Gruppenname `pcpatch` umbenannt und heißt von nun an: `opsifileadmins`. Das bedeutet, dass die User, die Zugriffsrechte für die Freigaben von opsi erhalten müssen (opsi-Paketierer) unter UCS 3 nicht Mitglied der Gruppe `pcpatch` werden können, sondern Mitglied der Gruppe `opsifileadmins` sein müssen. Diese Besonderheit gilt vorerst nur für UCS 3 und unterscheidet sich von den anderen Distributionen (solange bis diese auch Samba 4 einführen, oder Sie dort Samba 4 installieren)

Kapitel 11

Silent oder Unattended Schalter finden

Beim „unattended“ oder „silent setup“ wird das Original-Setup-Programm über Kommandozeilen-Argumente in einen nicht interaktiven Modus gestellt.

Das Problem dieser Installationsmethode ist es die geeigneten Kommandozeilenargumente zu finden.

Ermitteln des Herstellers des Setup-Programms: Die meisten Setupprogramme sind auf Basis von Frameworks wie *Inno*, *NSIS*, *Installshield* oder *Wise* gebaut. Jedes dieser Frameworks hat eigene typische Setupschalter. Um das Framework zu ermitteln kann unter anderem folgende Methode verwendet werden: Mit dem Kommandozeilen Programm `strings` werden die Strings aus der `setup.exe` extrahiert und danach mit `grep` bzw. `findstr` nach den Namen der Frameworks gesucht.

Die selbe Methode verwendet der `opsi-setup-detector`.

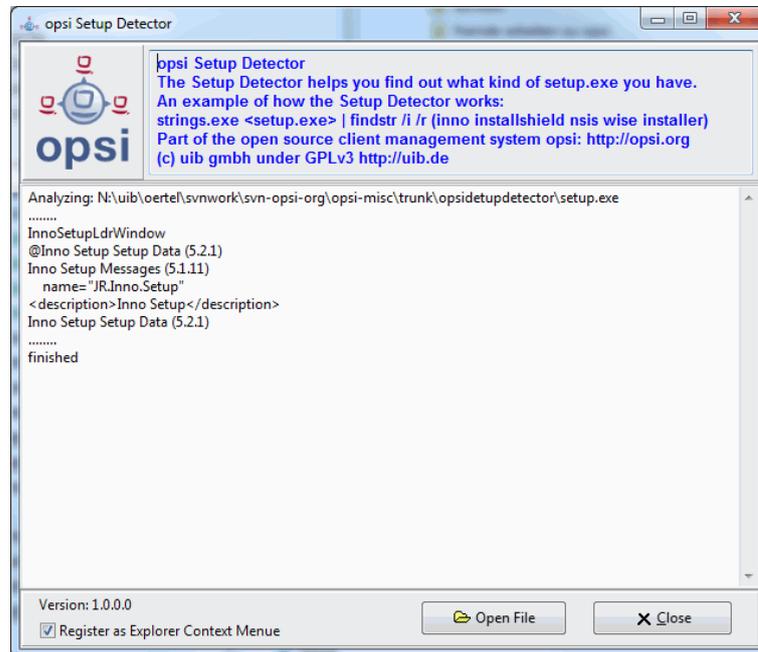


Abbildung 11.1: opsi setup detector

Dieses grafisch-interaktive Programm lässt sich zudem in das Kontextmenü des Explorers Einbinden.

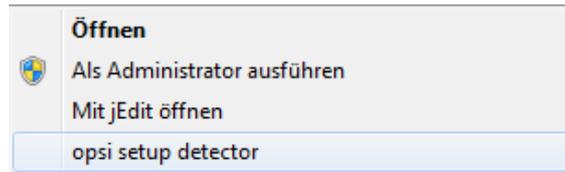


Abbildung 11.2: opsi setup detector im Kontextmenü des Explorers

Der *opsi setup detector* ist Bestandteil des Pakets *opsi-adminutils*. Das Programm kann aber auch alleine von heruntergeladen werden von:

<http://download.uib.de/opsi4.0/helper/opsisetupdetector.exe>

11.1 Skript für Installationen im Kontext eines lokalen Administrators

Das Produkt `opsi-template-with-admin` wurde deutlich überarbeitet:

- Es wird keine externe `psgetsid.exe` mehr benötigt
- Im Rahmenscript müssen nur drei Variablen angepasst werden die jetzt auch deutlich hierfür markiert sind.
- Es gibt nun einen direkten Zugang vom lokalen Skript auf die Produkteigenschaften und das Lizenzmanagement mit den gewohnten Funktionen.
- Auch die Funktion `isFatalError` funktioniert jetzt so wie man es erwartet.
- `LogLevel` und Tastatursperre werden über ein Property gesteuert.
- Das Script benötigt `opsi-winst >= 4.11.3.3`

Kapitel 12

opsi-winst 4.11.3

12.1 Start und Aufrufparameter

Der *opsi-winst* enthält seit Version 4.11.3 ein Manifest mit der Option: `<requestedExecutionLevel level="highestAvailable" />`. Dies bedeutet, dass unter NT6 als Administrator aufgerufen, versucht wird als *elevated* Prozess zu arbeiten. Wird der *opsi-winst* mit User Rechten aufgerufen, so läuft er unter den Rechten dieses Users.

12.2 Globale Textkonstanten

12.2.1 *opsi-winst* Pfad und Verzeichnisse

`%opsiScriptHelperPath%`

Entspricht: `%ProgramFiles32Dir%\opsi.org\opsiScriptHelper`

Pfad in dem Hilfsprogramme, Libraries und ähnliches zur Scriptausführung installiert sein können.

Seit 4.11.3.2

12.3 Primäre Sektionen

12.3.1 Reaktion auf Fehler

- `FatalOnSyntaxError = <Wahrheitswert>`
 - *true* = (default) Bei einem Syntaxfehler wird das Script abgebrochen und *failed* zurückgeliefert. Dem Server wird die Meldung *Syntax Error* übergeben.
 - *false* = Bei einem Syntaxfehler wird das Script nicht abgebrochen.

Der Syntaxfehler wird in jedem Fall als *Critical* in die Logdatei übernommen.

In jedem Fall wird der Errorcounter um 1 erhöht.

Seit 4.11.3.2

In älteren Versionen wird weder gelogged noch abgebrochen.

12.3.2 Fenster Modus

- **NormalizeWinst**
setzt das *opsi-winst* Fenster auf *normal* Modus.
- **IconizeWinst**
setzt das *opsi-winst* Fenster auf *minimierten* Modus.
- **RestoreWinst**
setzt das *opsi-winst* Fenster auf *maximierten* Modus.

12.3.3 Include Kommandos



Achtung

Die Verwendung von Include Kommandos führt schnell zu unübersichtlichen Code. Lassen Sie die Finger davon wenn Sie Anfänger sind.

12.3.3.1 Include Kommandos: Syntax

Mit Include Kommandos kann der Inhalt einer externen Datei dem laufende Script hinzugefügt werden. Dies kann entweder einfügend oder anhängend erfolgen. Die Include Kommandos sind normale Kommandos der primären Sektionen. Die eingefügten Dateien können weitere Include Kommandos enthalten.

Diese Kommandos gibt es seit Version 4.11.3

- **include_insert** <file name>
Fügt den Inhalt von <file name> nach der aktuellen Zeile im laufenden Script ein. Somit ist die erste Zeile der eingefügten Datei die nächste Zeile welche der *opsi-winst* interpretiert.
- **include_append** <file name>
Fügt den Inhalt von <file name> am Ende des laufenden Scriptes ein. Diese Anweisung dient vor allem dazu Sektionen aus z.B. einer Bibliothek hinzu zufügen.

Für beide Funktionen gilt:

<file name> ist:

- Ein kompletter Pfad zu einer Datei.
- Eine Datei in %ScriptPath%
- Eine Datei in %opsiScriptHelperPath%\lib
Entspricht: %ProgramFiles32Dir%\opsi.org\opsiScriptHelper\lib
- Eine Datei in %WinstDir%\lib

Die Prüfung erfolgt in dieser Reihenfolge. Die erste Datei die gefunden wird, wird genommen.

Beispiel:

Wir haben folgendes Script:

```
[Actions]
include_append "section_files_del_tmp_dummy.opsiinc"
include_insert "include-test1.opsiinc"
```

Dabei hat die Datei `include-test1.opsiinc` folgenden Inhalt:

```
DefVar $inctestvar$
set $inctestvar$ = "inctest"
Files_del_tmp_dummy
include_append "section_Files_copy_inctest.opsiinc"
Files_copy_inctest

if fileExists("c:\opsi.org\tmp\dummy.txt")
    comment "passed"
else
    comment "failed"
    set $TestResult$ = "not o.k."
    LogWarning "include test failed"
endif

if fileExists("%scriptpath%\test-files\dummy.txt")
    comment "passed"
else
    comment "failed"
    set $TestResult$ = "not o.k."
    LogWarning "include test failed"
endif
Files_del_tmp_dummy
```

Dabei hat die Datei section_Files_copy_inctest.opsiinc folgenden Inhalt:

```
[Files_copy_inctest]
copy "%scriptpath%\test-files\dummy.txt" "c:\opsi.org\tmp"
```

Dabei hat die Datei section_Files_del_tmp_dummy.opsiinc folgenden Inhalt:

```
[Files_del_tmp_dummy]
del -f "c:\opsi.org\tmp\dummyt.txt"
```

12.3.3.2 Include Kommandos: Library

Mit der Version 4.11.3 werden folgende Includefiles in %WinstDir%\lib ausgeliefert:

insert_check_exit_code.opsiinc:

```
; opsi include file

DefVar $ExitCode$

include_append "section_sub_check_exitcode.opsiinc"
```

insert_get_licensekey.opsiinc:

```
; opsi include file

DefVar $LicenseRequired$
DefVar $LicenseKey$
DefVar $LicensePool$

include_append "section_sub_get_licensekey.opsiinc"
```

section_sub_check_exit_code.opsiinc:

```
;opsi include file

[Sub_check_exitcode]
comment "Test for installation success via exit code"
set $ExitCode$ = getLastExitCode
; informations to exit codes see
; http://msdn.microsoft.com/en-us/library/aa372835(VS.85).aspx
; http://msdn.microsoft.com/en-us/library/aa368542.aspx
if ($ExitCode$ = "0")
    comment "Looks good: setup program gives exitcode zero"
else
    comment "Setup program gives a exitcode unequal zero: " + $ExitCode$
    if ($ExitCode$ = "1605")
        comment "ERROR_UNKNOWN_PRODUCT 1605 This action is only valid for products
that are currently installed."
        comment "Uninstall of a not installed product failed - no problem"
    else
        if ($ExitCode$ = "1641")
            comment "looks good: setup program gives exitcode 1641"
            comment "ERROR_SUCCESS_REBOOT_INITIATED 1641 The installer has
initiated a restart. This message is indicative of a success."
            ExitWindows /Reboot
        else
            if ($ExitCode$ = "3010")
                comment "looks good: setup program gives exitcode 3010"
                comment "ERROR_SUCCESS_REBOOT_REQUIRED 3010 A restart is
required to complete the install. This message is indicative of a success."
                ExitWindows /Reboot
            else
                logError "Fatal: Setup program gives an unknown exitcode unequal
zero: " + $ExitCode$
                isFatalError "Exit Code: "+ $ExitCode$
            endif
        endif
    endif
endif
endif
```

section_sub_get_licensekey.opsiinc:

```
; opsi include file

[Sub_get_licensekey]
if opsiLicenseManagementEnabled
    comment "License management is enabled and will be used"

    comment "Trying to get a license key"
    Set $LicenseKey$ = demandLicenseKey ($LicensePool$)
    ; If there is an assignment of exactly one licensepool to the product the following call
    is possible:
    ; Set $LicenseKey$ = demandLicenseKey ("", $ProductId$)
    ;
    ; If there is an assignment of a license pool to a windows software id, it is possible to
    use:
    ; DefVar $WindowsSoftwareId$
    ; $WindowsSoftwareId$ = "..."
    ; Set $LicenseKey$ = demandLicenseKey ("", "", $WindowsSoftwareId$)
```

```

DefVar $ServiceErrorClass$
set $ServiceErrorClass$ = getLastServiceErrorClass
comment "Error class: " + $ServiceErrorClass$

if $ServiceErrorClass$ = "None"
    comment "Everything fine, we got the license key '" + $LicenseKey$ + "'"
else
    if $ServiceErrorClass$ = "LicenseConfigurationError"
        LogError "Fatal: license configuration must be corrected"
        LogError getLastServiceErrorMessage
        isFatalError $ServiceErrorClass$
    else
        if $ServiceErrorClass$ = "LicenseMissingError"
            LogError "Fatal: required license is not supplied"
            isFatalError $ServiceErrorClass$
        endif
    endif
endif
endif

else
    LogError "Fatal: license required, but license management not enabled"
    isFatalError "No Licensemanagement"
endif
endif

```

12.3.4 String-Verarbeitung

- `contains(<str>, <substr>)`
Boolsche Funktion welche *true* liefert wenn <substr> in <str> enthalten ist. Die Funktion arbeitet case sensitive.
Seit 4.11.3
Beispiel:

```

set $ConstTest$ = "1xy451Xy451XY45"
set $CompValue$ ="xy"
if contains($ConstTest$, $CompValue$)
    comment "passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "failed"
endif
set $CompValue$ ="xY"
if not(contains($ConstTest$, $CompValue$))
    comment "passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "failed"
endif
endif

```

- `stringReplace(<string>, <oldPattern>, <newPattern>)`
Liefert einen String in dem in <string>, <oldPattern> durch <newPattern> ersetzt ist. Die Funktion arbeitet nicht case sensitive.
Seit 4.11.3
Beispiel:

```

set $ConstTest$ = "123451234512345"
set $CompValue$ = stringReplace("1xy451Xy451XY45", "xy", "23")

```

```
if ($ConstTest$ = $CompValue$)
    comment "passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "failed"
endif
```

- `strLength(<string>)`
Liefert Anzahl der Zeichen in <string>
Seit 4.11.3
Beispiel:

```
set $tmp$ = "123456789"
set $ConstTest$ = "9"
set $CompValue$ = strLength($tmp$)
if $ConstTest$ = $CompValue$
    comment "passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "failed"
endif
set $tmp$ = ""
set $ConstTest$ = "0"
set $CompValue$ = strLength($tmp$)
if $ConstTest$ = $CompValue$
    comment "passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "failed"
endif
```

- `strPos(<string>, <sub string>)`
Liefert die Position des ersten Vorkommens von <sub string> in <string>. Wird <sub string> nicht gefunden so liefert die Funktion "0". Die Funktion arbeitet case sensitive.
Seit 4.11.3
Beispiel:

```
set $tmp$ = "1xY451Xy451xy45"
set $ConstTest$ = "7"
set $CompValue$ = strPos($tmp$, "Xy")
if $ConstTest$ = $CompValue$
    comment "passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "failed"
endif
set $tmp$ = lower("1xY451Xy451xy45")
set $ConstTest$ = "2"
set $CompValue$ = strPos($tmp$, lower("xy"))
if $ConstTest$ = $CompValue$
    comment "passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "failed"
endif
```

- `strPart(<string>, <start pos>, <number of chars>)`

Liefert einen Teilstring von <string> beginnend mit <start pos> und <number of chars> lang. Wenn ab <start pos> weniger als <number of chars> vorhanden sind, so wird der String bis zum Ende geliefert.

Die Zählung der Zeichen beginnt mit 1.

Seit 4.11.3

Beispiel:

```
set $tmp$ = "123456789"
set $ConstTest$ = "34"
set $CompValue$ = strPart($tmp$, "3", "2")
if $ConstTest$ = $CompValue$
    comment "passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "failed"
endif
set $tmp$ = "123456789"
set $ConstTest$ = "56789"
set $CompValue$ = strPart($tmp$, strPos($tmp$, "56"), strLength($tmp$))
if $ConstTest$ = $CompValue$
    comment "passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "failed"
endif
```

- `unquote(<string>, <quote-string>)`

Wenn <string> mit dem Anführungszeichen <quote-string> versehen ist so liefert diese Funktion <string> ohne Anführungszeichen

Von <quote-string> wird nur das erste Zeichen berücksichtigt, wobei führende Whitespaces ignoriert werden.

Seit 4.11.2.1

```
set $ConstTest$ = "b"
set $CompValue$ = unquote('`b`', '`')
comment "compare values"
if ($ConstTest$ = $CompValue$)
    comment "passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "failed"
endif
comment "double quote"
set $ConstTest$ = "b"
set $CompValue$ = unquote('`b`', '`')
comment "compare values"
if ($ConstTest$ = $CompValue$)
    comment "passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "failed"
endif
comment "quote string will be trimmed and then only the first char is used"
comment "note: brackets are different chars"
set $ConstTest$ = "b]"
set $CompValue$ = unquote("[b]", "[{ ")
```

```
comment "compare values"
if ($ConstTest$ = $CompValue$)
    comment "passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "failed"
endif
comment "not usable to remove brackets"
set $ConstTest$ = "b]"
set $CompValue$ = unquote("[b]", "[")
set $CompValue$ = unquote($CompValue$, "]"")
set $CompValue$ = unquote("[b]", "]"")
set $CompValue$ = unquote($CompValue$, "["")
set $CompValue$ = unquote(unquote("[b]", "["), "]"")
comment "compare values"
if ($ConstTest$ = $CompValue$)
    comment "passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "failed"
endif
comment "if string not quoted it will be come back without changes"
set $ConstTest$ = "b"
set $CompValue$ = unquote("b", "'")
comment "compare values"
if ($ConstTest$ = $CompValue$)
    comment "passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "failed"
endif
```

12.3.5 Weitere String-Funktionen

- `getDiffTimeSec`
liefert einen String mit dem ganzzahligen Wert der vergangenen Sekunden seit dem letzten Aufruf von `marktime`.
Seit Version 4.11.3.1
- `SidToName(<well known sid>)`
liefert einen String mit dem lokalisierten Namen der mit `<well known sid>` bezeichneten Gruppe. Zum Beispiel für *S-1-5-32-544* je nach Lokalisierung des Betriebssystems *Administratoren* oder *Administrators*.
Seit Version 4.11.3.1
- `GetMyIpByTarget(<target ip addr>)`
liefert einen String mit der IP-Adresse des Interfaces, welches das Betriebssystem verwenden wird, wenn es versucht `<target ip addr>` zu erreichen. Diese Funktion liefert sicherer den korrekten Wert als die Verwendung der Konstante `%IPAddress%`.
Seit Version 4.11.3.1
Beispiel:

```
set $CompValue$ = getMyIpByTarget("%opsiServer%")
```

- `GetIpByName(<ip addr / ip name>)`
liefert die IP-Adresse des mit `<ip addr / ip name>` angegebenen Rechners
Seit Version 4.11.3.2

```
set $ConstTest$ = "%IPAddress%"
set $string1$ = "%IPAddress%"
set $CompValue$ = getIpByName($string1$)
if ($ConstTest$ = $CompValue$)
    comment "passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "failed"
endif
set $CompValue$ = getIpByName("%HostID%")
if ($ConstTest$ = $CompValue$)
    comment "passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "failed"
endif
set $CompValue$ = getIpByName("%PCName%")
if ($ConstTest$ = $CompValue$)
    comment "passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "failed"
endif
```

- `getLastExitCode`

Die String-Funktion `getLastExitCode` gibt den `ExitCode` des letzten Prozessaufrufs der vorausgehenden *WinBatch* / *DosBatch* / *ExecWith* Sektion aus.

Der Aufruf anderer opsi-winst Befehle (wie z.B. einer *Files* Sektion) verändert den gefundenen `ExitCode` nicht.

Bei *DosBatch* und *ExecWith* Sektionen erhalten wir den `Exitcode` des Interpreters. Daher muss in der Regel der gewünschte `Exitcode` in der Sektion explizit übergeben werden.

Beispiel:

```
DosInAnIcon_exit1
set $ConstTest$ = "1"
set $CompValue$ = getLastExitCode
if ($ConstTest$ = $CompValue$)
    comment "DosBatch / DosInAnIcon exitcode passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "DosBatch / DosInAnIcon exitcode failed"
endif

[DosInAnIcon_exit1]
rem create an errolevel= 1
VERIFY OTHER 2> NUL
echo %ERRORLEVEL%
exit %ERRORLEVEL%
```

12.3.6 Boolesche Ausdrücke

- `runningAsAdmin`

Boolesche Funktion welche *true* liefert wenn das laufende Script mit Administrativen Rechten ausgeführt wird.

Seit 4.11.1.1

- **isLoginScript**
Boolsche Funktion welche *true* liefert wenn das laufende Script über die opsi Erweiterung *User Profile Management* als *userLoginScript* läuft.
Seit 4.11.2.1
- **contains(<str>, <substr>)**
Boolsche Funktion welche *true* liefert wenn <substr> in <str> enthalten ist. Die Funktion arbeitet case sensitive.
Seit 4.11.3
- **isNumber(<str>)**
Boolsche Funktion welche *true* liefert wenn <str> einen ganzzahligen Wert (integer) repräsentiert.
Seit 4.11.3

12.3.7 String-Listen aus der Registry

- **getRegistryKeyList32(<regkey>)**
Liefert eine Liste mit dem Namen aller Keys direkt unterhalb von <regkey>.
32 Bit Modus (mit Redirection). Seit 4.11.3
- **getRegistryKeyList64(<regkey>)**
Liefert eine Liste mit dem Namen aller Keys direkt unterhalb von <regkey>.
64 Bit Modus (ohne Redirection). Seit 4.11.3
- **getRegistryKeyListSysnative(<regkey>)**
Liefert eine Liste mit dem Namen aller Keys direkt unterhalb von <regkey>.
Modus abhängig von der Architektur des Betriebssystems. Seit 4.11.3
- **getRegistryVarList32(<regkey>)**
Liefert eine Liste mit dem Namen aller Werte direkt unterhalb von <regkey>.
32 Bit Modus (mit Redirection). Seit 4.11.3
- **getRegistryVarList64(<regkey>)**
Liefert eine Liste mit dem Namen aller Werte direkt unterhalb von <regkey>.
64 Bit Modus (ohne Redirection). Seit 4.11.3
- **getRegistryVarListSysnative(<regkey>)**
Liefert eine Liste mit dem Namen aller Werte direkt unterhalb von <regkey>.
Modus abhängig von der Architektur des Betriebssystems. Seit 4.11.3
- **getRegistryVarMap32(<regkey>)**
Liefert eine Map mit den Namen=Value Paaren aller Werte direkt unterhalb von <regkey>.
32 Bit Modus (mit Redirection). Seit 4.11.3
- **getRegistryVarMap64(<regkey>)**
Liefert eine Map mit den Namen=Value Paaren aller Werte direkt unterhalb von <regkey>.
64 Bit Modus (ohne Redirection). Seit 4.11.3
- **getRegistryVarMapSysnative(<regkey>)**
Liefert eine Map mit den Namen=Value Paaren aller Werte direkt unterhalb von <regkey>.
Modus abhängig von der Architektur des Betriebssystems. Seit 4.11.3

Beispiel:

Wir erzeugen Registryeinträge mit folgender Sektion durch den Aufruf von:

```
Registry_createkeys /32Bit

openkey [HKEY_LOCAL_MACHINE\SOFTWARE\opsi.org\opsi-winst-test]
set "var1" = "value1"
set "var2" = REG_SZ:"value2"
```

```
set "var3" = REG_EXPAND_SZ:"value3"  
set "var4" = REG_DWORD:444  
set "var5" = REG_BINARY:05 05 05 0F 10  
set "var6" = REG_MULTI_SZ:"value6|value7|de"  
openkey [HKEY_LOCAL_MACHINE\SOFTWARE\opsi.org\opsi-winst-test\key1]  
openkey [HKEY_LOCAL_MACHINE\SOFTWARE\opsi.org\opsi-winst-test\key2]  
openkey [HKEY_LOCAL_MACHINE\SOFTWARE\opsi.org\opsi-winst-test\key3]  
openkey [HKEY_LOCAL_MACHINE\SOFTWARE\opsi.org\opsi-winst-test\key4]
```

Dann liefert uns:

```
set $list$ = getRegistryVarList32("hklm\software\opsi.org\opsi-winst-test")
```

folgenden Log:

```
Set $list$ = GetRegistryVarList32("hklm\software\opsi.org\opsi-winst-test")  
Registry started with redirection (32 Bit)  
  retrieving strings from GetRegistryVarList32 [switch to loglevel 7 for debugging]  
    (string 0)var1  
    (string 1)var2  
    (string 2)var3  
    (string 3)var4  
    (string 4)var5  
    (string 5)var6
```

Dann liefert uns:

```
set $list$ = getRegistryVarMap32("hklm\software\opsi.org\opsi-winst-test")
```

folgenden Log:

```
Set $list$ = GetRegistryVarMap32("hklm\software\opsi.org\opsi-winst-test")  
retrieving strings from GetRegistryVarMap32 [switch to loglevel 7 for debugging]  
  (string 0)var1=value1  
  (string 1)var2=value2  
  (string 2)var3=value3  
  (string 3)var4=444  
  (string 4)var5=05 05 05 0F 10  
  (string 5)var6=value6
```

Dann liefert uns:

```
set $list$ = getRegistryKeyList32("hklm\software\opsi.org\opsi-winst-test")
```

folgenden Log:

```
Set $list$ = GetRegistryKeyList32("hklm\software\opsi.org\opsi-winst-test")  
Registry started with redirection (32 Bit)  
  retrieving strings from GetRegistryKeyList32 [switch to loglevel 7 for debugging]  
    (string 1)key1  
    (string 2)key2  
    (string 3)key3  
    (string 4)key4
```

12.3.8 String-Listen aus Produkt Properties

- `getProductPropertyList(<propname>,<default value>)`
Liefert eine Liste mit den aktiven Werten des multivalue Properties <propname>. Im Gegensatz zu der Funktion `GetProductProperty` welche in diesem Fall die einzelnen Werte auf einem Komma separierten String zurück liefert. Diese Vorgehen wird problematisch wenn Kommas auch in der Werten vorkommen.
Seit 4.11.3
Beispiel:

```
;Property "dummymulti" has the values: ("ab", "cd", "ef", "g,h")
set $list$ = GetProductPropertyList ("dummymulti", "True")
if not (" " = takeFirstStringContaining($list$, "g,h"))
    comment "GetProductPropertyList passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "GetProductPropertyList failed"
endif

set $ConstTest$ = "ab,cd,ef,g,h"
set $CompValue$ = GetProductProperty ("dummymulti", "True")
if ($ConstTest$ = $CompValue$)
    comment "GetProductProperty passed"
else
    set $TestResult$ = "not o.k."
    LogWarning "GetProductProperty failed"
endif
```

12.3.9 Sonstige String-Listen

- `getProfilesDirList`
Liefert eine Liste der Pfade zu den lokalen Profilen.
Profile welche die folgenden Worte enthalten werden **nicht** berücksichtigt:

- *localservice*
- *networkservice*
- *systemprofile*

Das Profil des *Default Users* ist Bestandteil der Liste.
All User oder Public sind nicht Bestandteil der Liste.

Beispiel:

```
set $list1$ = getProfilesDirList
```

ergibt folgenden Log:

```
Set $list1$ = getProfilesDirList
Registry started with redirection (32 Bit)
  retrieving strings from getProfilesDirList [switch to loglevel 7 for debugging]
    (string 0)C:\Users\Administrator
    (string 1)C:\Users\Default
```

12.3.10 Fehlgeschlagene Installation anzeigen

Passieren bei einer Installation Fehler, die zum Fehlschlagen der Installation führen, so sollte dies an den Server zurückgemeldet werden.

Um in einem *opsi-winst* Skript, eine Installation als gescheitert zu erklären, gibt es eine Ausdruck namens `isFatalError`

unterbricht die normale Ausführung eines Skripts, an der Stelle, an der er aufgerufen wird. Nach dem der Befehl aufgerufen wurde, werden (außer if-Anweisungen) keine Anweisungen mehr ausgeführt und als Skriptergebnis wird *failed* zurückgeliefert. Wird dieser Befehl nicht aufgerufen, so ist das Skriptergebnis *success*.

Seit 4.11.3.2 ist auch die folgende Variante erlaubt:

- `isFatalError <string>`
wobei `<string>` als kurze Fehlerbeschreibung an den opsi-server als *actionProgress* weitergegeben wird und im opsi-configed angezeigt wird.

12.3.11 Kommandos zur Steuerung des Logging

- `includeLog <file name> <tail size> //since 4.11.2.1`
Fügt die Datei `<file name>` in den aktuellen ein. Dabei werden nur die letzten `<tail size>` Zeilen und nicht die komplette Logdatei eingefügt. Wenn Sie ein anderes Programm (z.B. ein setup Programm) starten das eine Logdatei produziert, können Sie mit diesem Befehl die Informationen aus dieser Logdatei in den Log des *opsi-winst* übernehmen.
Seit Version 4.11.3.2 kann auch eine negative `<tail size>` angegeben werden. Dann arbeitet `includeLog` im *Head* Modus, d.h. ist `<tail size> = -5`, so werden die ersten 5 Zeilen von `<file name>` in den Log übernommen. Beispiel:

```
includeLog "%Scriptpath%\test-files\10lines.txt" "5"
```

12.4 Sekundäre Sektionen

12.4.1 Patches-Sektionen

12.4.1.1 Aufrufparameter

Der Name der zu patchenden Datei wird als Parameter übergeben.

Als optionalen Modifier gibt es:

- `/AllNTUserProfiles`
Wird eine Patches Sektion mit diesem Modifier aufgerufen und der Pfad zur zu patchenden Datei enthält die Konstante `%UserProfileDir%`, so wird diese Patchsektion für alle Profile ausgeführt.
Eine *Patches* Sektion welche in einer `[ProfileActions]` Sektion aufgerufen wird hat im *Machine* Modus den Modifier `/AllNTUserProfiles` implizit. Im Loginscript Modus wird dann `%UserProfileDir%` als `%CurrentProfileDir%` interpretiert.
(Seit Version 4.11.3.2)

12.4.2 ExecWith Sektionen

Seit Version 4.11.3 wird, wenn als Interpreter *powershell* angegeben ist, die temporäre Datei als `.ps1` gespeichert.

12.4.3 WinBatch-Sektionen

12.4.3.1 Aufrufparameter (Modifier)

- `/WaitSeconds [AnzahlSekunden]`
Die Parametrisierung `/WaitSeconds [AnzahlSekunden]` modifiziert das Verhalten dahingehend, dass *opsi-winst* jeweils erst nach `[AnzahlSekunden]` die Skriptbearbeitung fortsetzt. Die angegebene Zeit stoppt *opsi-winst* auf jeden Fall. In der Default-Einstellung wird zusätzlich auf das Ende der angestoßenen Prozesse gewartet. Ist letzteres nicht gewünscht, so kann der Parameter mit dem Parameter `/LetThemGo` kombiniert werden.
- `/WaitForProcessEnding <program name>`
Wartet darauf, das sich der Prozess mit dem Namen `<program name>` beendet.
Kann und sollte mit `/TimeoutSeconds` kombiniert werden.

Erläuterung:

Der *opsi-winst* wartet auf das Ende eines per `winbatch` gestarteten Prozesses bevor mit der nächsten Zeile des Scriptes fortgefahren wird:

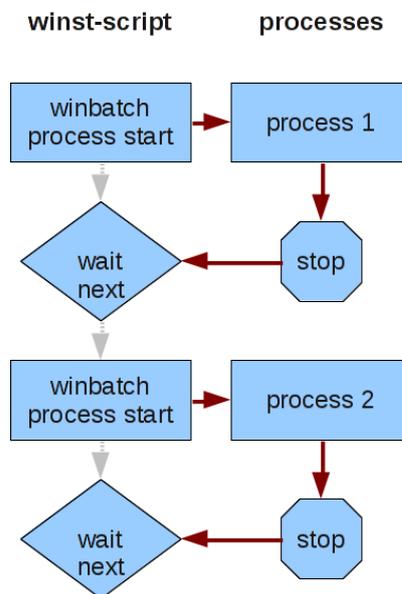


Abbildung 12.1: Sequentielle Abarbeitung des Scriptes mit Warten auf das Ende eines Prozesses

Es gibt allerdings Prozesse, welche einen weiteren Prozess starten und sich Beenden ohne auf das Ende des Kindprozesses zu warten. Aus Sicht des *opsi-winst* ist damit der Weg zur Ausführung des nächsten Befehls frei:

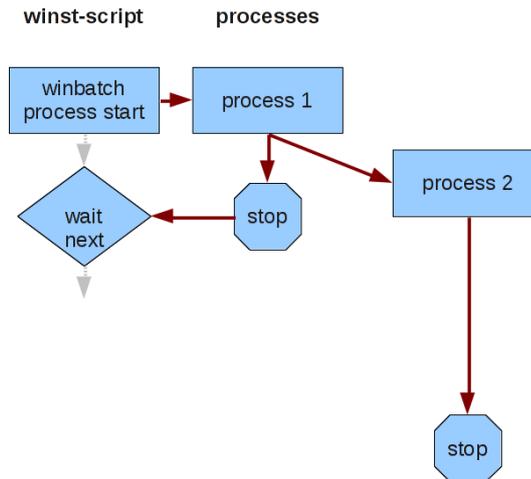


Abbildung 12.2: Ende eines Prozesses mit weiterlaufenden Kindprozess

Werden z.B. hintereinander ein Uninstall und ein Setup Programm aufgerufen und das Uninstall Programm führt die eigentliche Deinstallation in einem Kindprozess aus, so ist das Ergebnis undefiniert, da deinstallation und Installation gleichzeitig laufen:

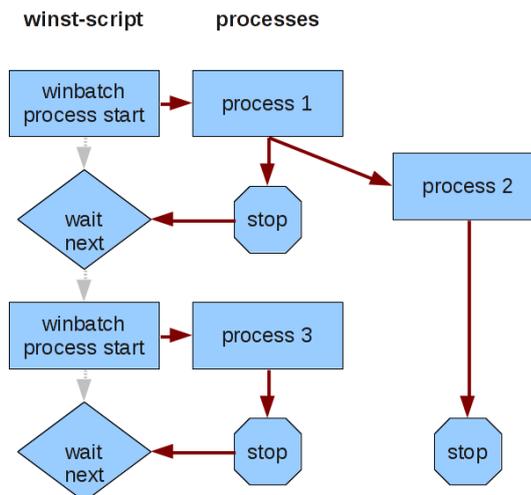


Abbildung 12.3: Überlappung von Kindprozess und nächstem gestarteten Prozess

Mit dem Modifier `/WaitForProcessEnding` kann eine solche Situation vermieden werden.

- `/TimeoutSeconds <seconds>`
Bricht das Warten auf das Prozessende oder eine Wartebedingung (`/WaitForProcessEnding`) nach Ablauf von `<seconds>` ab, auch wenn das Prozessende oder die Wartebedingung noch nicht erfüllt ist. Der Prozess auf dessen Ende gewartet werden sollte wird nicht gestoppt. Kann seit Version 4.11.3 auch alleine (z.B. ohne `/WaitForProcessEnding`) verwendet werden, aber nicht zusammen mit `/WaitSeconds`.
Beispiel:

```
Winbatch_uninstall /WaitForProcessEnding "uninstall.exe" /TimeoutSeconds 20
[Winbatch_uninstall]
"%ScriptPath%\uninstall_starter.exe"
```

- **/RunElevated**
Startet dem Prozess mit einem höheren Security Token (d.h. mit höheren Rechten). Dieser Modifier hat folgende Einschränkungen:
 - Unter NT5 hat er keine Auswirkungen
 - Ein Zugriff auf das Netz ist in dem Prozess nicht möglich. Daher müssen die aufzurufenden Programme von einem Netzlaufwerk in ein temporäres lokales Verzeichnis kopiert werden.
 - Evtl. kann es zu Problemen bei der Nutzung der grafischen Oberfläche kommen. Daher sind echte silent aufrufe hier zu bevorzugen.
 - Funktioniert nur im opsi-service Kontext
- **getLastExitCode**
Die String-Funktion `getLastExitCode` gibt den `ExitCode` des letzten Prozessaufrufs der vorausgehenden WinBatch / DosBatch / ExecWith Sektion aus.

12.5 Sonstiges

- wisynth: `TuibInstScript.doInfo`: optimal fill for `LabelInfo`
- name of installing product shown in initial info block. fixes #367
- wisynth: `executeWith`: use extension `.ps1` if interpreter = powershell, fixes #382
- logfiles now again not locked against reading while opsi-winst is running fixes: <https://forum.opsi.org/viewtopic.php?f=7&t=3655>
- winbatch: test if extension of existsting file is exe or com now not case sensitive. fixes: <https://forum.opsi.org/viewtopic.php?f=7&t=3592#p18127>
- Winbatch: The parameter to `/timeoutseconds` may now be a variable fixes #352

Kapitel 13

Sonstiges

13.1 Changelogs:

13.1.1 Changelog configed

opsi-configed (4.0.2.5.8-1) stable; urgency=low

- fixed: host groups were not reloaded on reload request

-- Rupert Roeder <r.roeder@uib.de> Mon, 15 Oct 2012 12:00:00 +0200

opsi-configed (4.0.2.5.6-1) stable; urgency=low

- fixed: problem when server property is missing
- fixed: a new client must not have the name of the opsi server (or depots)

-- Rupert Roeder <r.roeder@uib.de> Wed, 26 Sep 2012 12:00:00 +0200

opsi-configed (4.0.2.5.5-1) stable; urgency=low

- fixed: some problem when changing between not synchronized depots

-- Rupert Roeder <r.roeder@uib.de> Fri, 07 Sep 2012 18:00:00 +0200

opsi-configed (4.0.2.5.4-1) stable; urgency=low

- fixed: selection by ActionRequest did not work properly
- fixed: in selection dialog, groups are now listed in alphabetic order

-- Rupert Roeder <r.roeder@uib.de> Mon, 03 Sep 2012 18:00:00 +0200

opsi-configed (4.0.2.5.3-1) stable; urgency=low

- fixed: new behaviour of java 1.7.6 stopped location of an applet component

-- Rupert Roeder <r.roeder@uib.de> Fri, 31 Aug 2012 11:00:00 +0200

opsi-configed (4.0.2.5.1-1) stable; urgency=low

- update for reload function; selected pcs are kept

-- Rupert Roeder <r.roeder@uib.de> Tue, 25 Aug 2012 21:38:32 +0200

opsi-configed (4.0.2.5-1) stable; urgency=low

- fixed bug that filter results goes to null when reloading on product tab
- dragging content enabled for hardware info tables

-- Rupert Roeder <r.roeder@uib.de> Tue, 25 Aug 2012 16:38:32 +0200

opsi-configed (4.0.2.4-1) stable urgency=low

- updated FR language file
- eliminated NPEs for not authenticated users

—Rupert Roeder <r.roeder@uib.de> Thu, 16 Aug 2012 11:38:32 +0200

opsi-configed (4.0.2.2-1) testing; urgency=low

- new button indication of existing product property entries
- new client action resetLocalProducts

—Rupert Roeder <r.roeder@uib.de> Thu, 16 Jul 2012 11:38:32 +0200

opsi-configed (4.0.2.1-1) testing; urgency=low

- calls to hostControl_reachable and hostControl_getActiveSessions are made threaded to avoid blocking of the configed if there are problems
- improvements in gui of client search dialogs
- added support of description field for a saved search
- added support of description field for tree groups
- new button indication of existing product property entries
- new client action resetLocalProducts

—Rupert Roeder <r.roeder@uib.de> Thu, 20 Jun 2012 14:38:32 +0200

opsi-configed (4.0.2.0-1) testing; urgency=low

- implementation of DIRECTORY extension of treeview
- coloring of product versioninfo which are not equal to the server version
- report column can now be edited (by transforming the values to values of ActionResult, ActionProgress, LastState
- product properties and host configs are marked by bold font when not being identical with the default values
- an additional product column showing the (full) product names can be used
- if a opsi client is created there now is a check for existing host names which will not be reused without confirmation
- new sorting of depots, tooltip added for depots

—Rupert Roeder <r.roeder@uib.de> Thu, 14 Jun 2012 14:38:32 +0200

opsi-configed (4.0.1.27-1) testing; urgency=low

- improved window handling of client selection dialog

—Rupert Roeder <r.roeder@uib.de> Thu, 03 May 2012 14:38:32 +0200

opsi-configed (4.0.1.26-1) testing; urgency=low

- bugfix: product properties are reloaded for each selected client if a product was selected

—Rupert Roeder <r.roeder@uib.de> Thu, 26 Apr 2012 14:38:32 +0200

opsi-configed (4.0.1.25-1) testing; urgency=low

- improvement: Added a new client selection dialog

—Felix Rohrbach <fxrh@gmx.de> Wed, 21 Mar 2012 14:38:32 +0200

13.1.2 Changelog opsi-winst

opsi-winst (4.11.3.3) stable; urgency=low

- new command `setActionProgress <string>`
- commandline parameter `/productid <string>` This parameter forces the winst to try to start a service connection using the local `opsiclientd.conf`. This will be used in the `opsi-template-with-admin` to run the local script in opsi context
- `Versioninfo`: try to find informations also on other than the stored codepage
- `wisynth`: `doRegistryAllNTUserDats`: use `getProfilesDirList`
- `widatamodul`: `retrieveJSONObject`: JSON Bench Log entry for every service call
- `wimain`: `saveVersionToProfile/readVersionFromProfile` : if there is no access to `Appdata`, we use the profile directory for the `.opsi.org\userLoginScripts.ini`
- `wilog`: `PartOpenForReading`: fix for timing problem *Floatingpoint Underflow* — Detlef Oertel <d.oertel@uib.de> Thu, 13 Sep 2012:15:00:00 +0200

opsi-winst (4.11.3.2) stable; urgency=low

- new command `FatalOnSyntaxError` (default= true)
- every script syntax error is logged
- every script syntax error increases the errorcounter
- if `FatalOnSyntaxError` then every script syntax error is a fatal error and *Syntax Error* is set as `actionProgress`
- now `isFatalError` accepts a string which is stored to the server as `actionprogress` and showed in the `opsi-configed`
- `includeLogFile` now works in head mode if you give a negative number as line count
 - changed in `TLogInfo.includeLogtail`
- new stringlist function: `getProfilesDirList`: All existing Profiles dirs except: *localservice, networkservice, systemprofile*

- Patches sections may now called with %userprofiledir% as part of the file name and /AllNTuserProfiles so it will run on all profiles. If running in Machine mode, the /AllNTuserProfiles is implizit in the [ProfileActions]. If running in LoginScript mode, the %userprofiledir% is the %CurrentProfileDir%.
- new stringfunction: GetIpByName(<ip addr / ip name>) : string
- Files sections changed: wisynth: doFileActions: Modifier /AllNtUserProfiles and /AllNtUserSendto now based on getProfilesDirList
- new stringlist commands to retrieve maps from the registry:
 - getRegistryVarMap32(<regkey>)
 - getRegistryVarMap64(<regkey>)
 - getRegistryVarMapSysnative(<regkey>)
- winbatch: (Startprocess*) catching exceptions if execute bit is missing on samba4 shares

— Detlef Oertel <d.oertel@uib.de> Mon, 30 Jul 2012:15:00:00 +0200

opsi-winst (4.11.3.1) stable; urgency=low

- winbatch: /timeoutseconds now allowed without /waitforprocessending (not with /waitsec)
- Winbatch: The parameter to /timeoutseconds may now be a variable fixes #352
- winbatch /RunElevated
- patchhosts sections: 64 bit compatibility, fixes #327
- manifest added: <requestedExecutionLevel level="highestAvailable" />:
<http://www.lazarus.freepascal.org/index.php?topic=14976.0>
<http://lazarus.freepascal.org/index.php/topic,11659.msg69861.html>
<http://msdn.microsoft.com/en-us/library/bb756929.aspx>
- new stringfunction: getDiffTimeSec (Time in seconds since last marktime)
- name of installing product shown in inital info block. fixes #367
- Exitcode of DosBatch, Execwith is now logged and can be retrieved by getLastExitcode
 - be aware: this is the exitcode of the called program so use something like: exit %ERRORLEVEL% , fixes #161
- on userLoginScript: retry if there no user could be detected
- wisynth: TuibInstScript.doInfo: optimal fill for LabelInfo
- wisynth: executeWith: use extension .ps1 if interpreter = powershell, fixes #382
- new function: NormalizeWinst (set normal window state), fixes #346
- old function IconizeWinst (set minimized window state), fixes #346
- old function RestoreWinst (set maximized window state), fixes #346
- new stringfunction: stringReplace(string, oldPattern,newPattern) : string ;
this function replaces all occurrences of one substring with another in a string
it works not case sensitive , fixes #328
- new stringfunction: GetMyIpByTarget(<target ip addr>) : string
- new boolean function: contains(<str>, <substr>) : boolean ; case sensitive, fixes #365
- old boolean function: isNumber(<str>) : boolean

- new stringlist function: getProductPropertyList(<propname>,<default value>)
- wispecfolder: fix using winapi call GetDefaultUserProfileDirectory for %DefaultUserProfileDir%
- wispecfolder: fix using winapi call GetAllUsersProfileDirectory for %CommonProfileDir%
- new stringfunction SidToName(<well known sid>) : gives localized name of the sid
- wisynt: try-except in doRegistryHackRegeditFormat
- logfiles now again not locked against reading while opsi-winst is running
fixes: <https://forum.opsi.org/viewtopic.php?f=7&t=3655>
- new commands: include_insert "<file name>" : Replaces the command with a empty line and then inserts the content of "<file name>" after this line include_append "<file name>" : Replaces the command with a empty line and then appends the the content of "<file name>" to the end of the script
 - <file name> is either:
 - * a complete valid Path
 - * The name of a file in %WinstDir%\lib
 - * The name of a file in %ScriptPath%
 - should be placed after requiredWinstVersion
- new stringlist commands to retrieve lists from the registry:
 - getRegistryKeyList32(<regkey>)
 - getRegistryKeyList64(<regkey>)
 - getRegistryKeyListSysnative(<regkey>)
 - getRegistryVarList32(<regkey>)
 - getRegistryVarList64(<regkey>)
 - getRegistryVarListSysnative(<regkey>)
- new stringfunction strLength(<string>)
- new stringfunction strPos(<string>, <substr>)
- new stringfunction strPart(<string>, <start pos>, <number of chars>)
- winbatch: test if extension of existsting file is exe or com now not case sensitive. fixes: <https://forum.opsi.org/viewtopic.php?f=7&t=3592#p18127>

— Detlef Oertel <d.oertel@uib.de> Tue, 11 Jan 2012:15:00:00 +0200

opsi-winst (4.11.2.12) stable; urgency=low

- fix for getproductmap widatamodul: TOpsi4Data.initProduct: reset FProductOnClient_aktobject with nil

— Detlef Oertel <d.oertel@uib.de> Tue, 31 Jul 2012:15:00:00 +0200

opsi-winst (4.11.2.11) stable; urgency=low

- wispecfolder: fix for the fix using winapi call GetProfilesDirectory for %profiledir%

— Detlef Oertel <d.oertel@uib.de> Tue, 3 Jun 2012:15:00:00 +0200

opsi-winst (4.11.2.10) stable; urgency=low

- wispecfolder: fix using winapi call GetProfilesDirectory for %profiledir%

— Detlef Oertel <d.oertel@uib.de> Tue, 2 Jun 2012:15:00:00 +0200

13.1.3 Changelog hwaudit

hwaudit (4.0.5-1) stable; urgency=low

- added dell-expresscode to computer_system
- added productProperty loglevel log-file
- changeDirectory added (package requires now winst: 4.11.2.6)
- new temp-Directory C:\opsi.org\tmp instead of old C:\tmp

—Erol Ueluekmen <e.ueluekmen@uib.de> 31 Jul 2012

hwaudit (4.0.4-3) stable; urgency=low

- get timeout from property
- Ticketid 2012021510000025

—Bardo Wolf <b.wolf@uib.de> 23 Feb 2012

hwaudit (4.0.4-3) stable; urgency=low

- get timeout from property
- Ticketid 2012021510000025

—Bardo Wolf <b.wolf@uib.de> 23 Feb 2012

hwaudit (4.0.4-2) stable; urgency=low

- get timeout from property
- Ticketid 2012021510000025

—Bardo Wolf <b.wolf@uib.de> 23 Feb 2012

13.1.4 Changelog swaudit

swaudit (4.0.2-1) stable; urgency=low

- code cleanup
- remove all opsi 3 stuff
- use opsi-winst 4.11.3 methods
- extended search off by default (no data at NT6)
- replace delete by del
- replace c:\tmp by c:\opsi.org\tmp

—detlef oertel <d.oertel@uib.de> Fri, 10 Aug 2012 15:00:00 +0100

swaudit (4.0-15) testing; urgency=low

- new property: readfromfile to switch off audit and use the filebuffer files instead ONLY DEBUG - do not use if you are not shure what you doing! (Do not make a audit but read values from c:\tmp and send them)

—detlef oertel <d.oertel@uib.de> Fri, 11 May 2012 15:00:00 +0100
swaudit (4.0-14) testing; urgency=low

- workaround for bug#354 (set version to " if zero) deleted because it is fixed

—detlef oertel <d.oertel@uib.de> Tue, 26 Apr 2012 16:00:00 +0100
swaudit (4.0-13) testing; urgency=low

- set version to " if zero - workaround for bug#354

—detlef oertel <d.oertel@uib.de> Tue, 15 Mar 2012 16:00:00 +0100
swaudit (4.0-12) testing; urgency=low

- new property: usekeyfinder to switch off keyfinder call

—detlef oertel <d.oertel@uib.de> Tue, 29 Nov 2011 16:00:00 +0100

13.1.5 Changelog jedit

jedit_4.5.2-2 stable; urgency=low

- opsi-winst.xml for version 4.11.3.2

—detlef oertel <d.oertel@uib.de> Tue, 31 Jul 2012 16:01:53 +0200
jedit_4.5.2-1 stable; urgency=low

- update to version 4.5.2
- opsi-winst.xml for version 4.11.3.1
- opsi-winst.xml for **ins**,.opsiscript,*opsiinc

—detlef oertel <d.oertel@uib.de> Thu, 12 Jul 2012 16:01:53 +0200
jedit_4.5.1-1 stable; urgency=low

- update to version 4.5.1
- property gui_language description: jedit localisation values: ["auto", "de", "en", "fr"] default: ["en"]

—bardo wolf <b.wolf@uib.de> 12 Jun 2012

13.1.6 Changelog opsi-adminutils

opsi-adminutils (4.0.2.1-1) stable; urgency=low

- replace ussf by opsi-setup-detector
- update auf InstEd-1.5.15.26.msi
- winscp 4.3.9

—detlef oertel <d.oertel@uib.de> 06 Aug 2012
opsi-adminutils (4.0.1.24-1) stable; urgency=low

- try to fix [2012031610000012] opsi-adminutils webstart funktioniert nicht.
- postinst check if newer configed.jar swingx.jar on server
- winscp 4.3.7
- property javaweb installer or local configed
- swingx.jar
- configed.jar 4.0.1.24
- language selection
- replace %ProgramfilesDir% by %Programfiles32Dir%

—Bardo Wolf <b.wolf@uib.de> 16 Mar 2012

opsi-adminutils (4.0.1.12-4) stable; urgency=low

- corrected linkfolder for opsi-configed

—Rupert Röder <r.roeder@uib.de> 07 Mar 2012

opsi-adminutils (4.0.1.12-3) stable; urgency=low

- putty 0.62
- winscp 4.3.5

—Susanne Ehlers <s.ehlers@uib.de> 13 Dec 2011

opsi-adminutils (4.0.1.12-2) stable; urgency=low

- JXv3.1 replaced by jxplorer-3.2.2-windows-installer.exe

—Bardo Wolf <b.wolf@uib.de> 17 Oct 2011

13.1.7 Changelog opsi-template

opsi-template (4.0.2-2) stable; urgency=low

- requiredWinstVersion >= "4.11.3.2"
- Set \$LogDir\$ = "%SystemDrive%\opsi.org\tmp"
- new: *_short32.opsiscript scripts with includes
- winbatch_install: changed inno-setup example: default without answer file
- winbatch_uninstall: changed NSIS example: /WaitForProcessEnding "Au_.exe"

—detlef oertel <d.oertel@uib.de> Fri, 03 Aug 2012 16:01:53 +0200

opsi-template (4.0.2-1) stable; urgency=low

- requiredWinstVersion >= "4.11.2.6"
- added ChangeDirectory "%SCRIPTPATH%"
- changed delete to del (in delsub)
- changed at msi to ALLUSERS=1

—detlef oertel <d.oertel@uib.de> Mon, 02 Jun 2012 16:01:53 +0200

13.1.8 Changelog opsi-template-with-admin

opsi-template-with-admin (4.0.2-2) stable; urgency=low

- requires opsi-winst \geq 4.11.3.3
- uses /productid switch at opsi-winst call
- code cleanup (use of synative instead of 64bit)
- property debug: switches loglevel and lock keyboard
- code cleanup

—detlef oertel <d.oertel@uib.de> Thu, 19 Sep 2012 15:01:53 +0200

opsi-template-with-admin (4.0.2-1) stable; urgency=low

- requires opsi-winst \geq 4.11.3
- using SidToName instead of psgetsid.exe
- priority set to -20
- first reboot is commented (;ExitWindows /ImmediateReboot). Uncomment it if you have problems with the auto-logon

—detlef oertel <d.oertel@uib.de> Wed, 04 Jul 2012 15:01:53 +0200

13.1.9 Changelog opsi-client-agent

opsi-client-agent (4.0.2.1-2) stable; urgency=low

- no interactive message on copy error if \$INST_SubModus\$="CREATE_CLIENT"
- opsi-winst 4.11.3.2
- use of GetMyIpByTarget()
- use of GetIpByName()
- devided Files_redist_cleanup to Files_redist_cleanup_c and Files_redist_cleanup_d

—Detlef Oertel <d.oertel@uib.de> Mon, 4 Jun 2012 15:00:00 +0100

13.1.10 Changelog windows netboot products

windows (4.0.2-1) stable; urgency=low

- Security fix in net mount while winpe works (NT6)
- set alignment for partitions (NT5)
- show_drivers.py modified to show automated including additional driver over inventory
- create_driver_links for handling with fileadmingroup from opsi.conf-file

—uib GmbH <info@uib.de> Fr, 29 Jun 2012 15:19:15 +0000

windows (4.0.1-11) stable; urgency=low

- fix in multilanguage control file (time zone)
- fix image names windows server

— uib GmbH <info@uib.de> Mon, 2 Apr 2012 15:19:15 +0000
windows (4.0.1-10) stable; urgency=low

- multilanguage control file
- additional locales and keyboard layouts

— uib GmbH <info@uib.de> Tue, 24 Jan 2012 14:19:15 +0000
windows (4.0.1-9) stable; urgency=low

- blockAlignment added.

— uib GmbH <info@uib.de> Wed, 16 Nov 2011 11:06:24 +0000
windows (4.0.1-8) stable; urgency=low

- descriptions now english
- Added new properties for patching unattend.xml Component Microsoft-Windows-International-Core-WinPE winpe_uilanguage winpe_uilanguage_fallback winpe_inputlocale
- added new properties to unattend.xml.template
- show_drivers.py

— bardo wolf <b.wolf@uib.de> Mon, 31 Oct 2011 09:09:50 +0000
windows (4.0.1-7) stable; urgency=low

- unattend.xml patched for postinst-routines like 2008-r2
- show_drivers.py modified to show right list of drivers; now with messages for duplicate drivers.

— uib GmbH <info@uib.de> Thu, 22 Sep 2011 14:19:15 +0000
windows (4.0.1-6) stable; urgency=low

- Remove Display settings from unattend.xml.template

— uib GmbH <info@uib.de> Thu, 25 Aug 2011 12:25:50 +0000
windows (4.0.1-5) stable; urgency=low

- Call auditHardwareOnHost_setObsolete

— uib GmbH <info@uib.de> Tue, 19 Jul 2011 13:17:02 +0000
windows (4.0.1-4) stable; urgency=low

- Modify driver integration

— uib GmbH <info@uib.de> Mon, 18 Jul 2011 15:05:10 +0000

13.1.11 Changelog python-opsi

python-opsi (4.0.2.4-1) stable; urgency=low

- fixes method setProductActionRequestWithDependencies after host_createOpsiClient
- added default dhcp string and text options that the values will be set in double-quotes (fixes#403)
- added method userIsReadOnlyUser()
- WindowsDriverIntegration: do not break when no devices found in txtsetup.oem (corrupted txtsetup.oem)

—Erol Ueluekmen <e.ueluekmen@uib.de> Thu, 27 Sep 2012 10:35:17 +0200

python-opsi (4.0.2.3-1) testing; urgency=low

- Workaround for bootimage: wait if blockfile to partition does not exist.
- Automated additional-driver integration support.
- hostControl-Fix for host_reachable method.
- added opsiFileAdminhandling, added new opsi.conf File.
- dellexpresscode for hwinvent implemented
- licensekey length increased to 1024
- licensekey length increased to 1024
- use opsi-auth pam module if exists:

—Erol Ueluekmen <e.ueluekmen@uib.de> Tue, 17 Jul 2012 13:33:13 +0200

python-opsi (4.0.2.2-1) testing; urgency=low

- Workaround for python 2.7 in jsonrpc-backend: compressed data will send as bytearray
- fix for isc-dhcp-server for oneiric and precise

—Erol Ueluekmen <e.ueluekmen@uib.de> Mon, 11 Jun 2012 13:42:58 +0200

13.1.12 Changelog opsipxeconfd

opsipxeconfd (4.0.2.6-1) stable; urgency=low

- Fix: if always switched to none, don't create new pipe for client.

—e.ueluekmen@uib.de <erol@erollinux.uib.local> Mon, 08 Oct 2012 14:42:45 +0200

opsipxeconfd (4.0.2.5-1) experimental; urgency=low

- Fix: dont lose append entry from template, if no productProperty set

—Erol Ueluekmen <e.ueluekmen@uib.de> Tue, 25 Sep 2012 14:48:08 +0200

opsipxeconfd (4.0.2.4-1) experimental; urgency=low

- write pipe again, if product has set always actionRequest for client.

— Erol Ueluekmen <e.ueluekmen@uib.de> Tue, 18 Sep 2012 18:27:36 +0200
opsipxeconfd (4.0.2.3-1) experimental; urgency=low

- when spaces are in append, does not shuffle productPropertyStates

— Jim Pizagno <j.pizagno@uib.de> Wed, 05 Sep 2012 17:03:22 +0200
opsipxeconfd (4.0.2.2-1) testing; urgency=low

- fixing init.d lsb-header for SUSE based servers. Removing runlevel 4
- fileadmingroup added in /etc/opsi/backends/file.conf

— Erol Ueluekmen <e.ueluekmen@uib.de> Fri, 22 Jun 2012 16:26:42 +0200

13.1.13 Changelog opsiconfd

opsiconfd (4.0.2.4-1) stable; urgency=low

- added readOnly-State in session and worker

— Erol Ueluekmen <e.ueluekmen@uib.de> Thu, 04 Oct 2012 11:18:36 +0200
opsiconfd (4.0.2.3-1) testing; urgency=low

- fixing floatdivision in statistics-class

— Erol Ueluekmen <e.ueluekmen@uib.de> Wed, 18 Jul 2012 17:05:54 +0200
opsiconfd (4.0.2.2-1) testing; urgency=low

- omb: python 2.4 fixing import hashlib
- fixing init.d lsb-header for SUSE based servers. Removing runlevel 4

— Erol Ueluekmen <e.ueluekmen@uib.de> Fri, 22 Jun 2012 15:06:59 +0200

13.1.14 Changelog opsi-utils

opsi-utils (4.0.2.3-1) stable; urgency=low

- opsi-product-updater: Fix for using more than one repositories with different onlyDownload options.

— Erol Ueluekmen <e.ueluekmen@uib.de> Thu, 11 Oct 2012 14:13:35 +0200
opsi-utils (4.0.2.2-1) testing; urgency=low

- Added opsiFileAdminGroup handling.
- opsi-product-updater: Timestamps in Mail
- opsi-newprod: fix for small window start, now has error message

— Erol Ueluekmen <e.ueluekmen@uib.de> Thu, 19 Jul 2012 11:44:37 +0200

13.1.15 Changelog opsi-linux-bootimage

opsi-linux-bootimage (20120728-1) stable; urgency=low

- kernel 3.5
- python-opsi 4.0.2.3-1

— Erol Ueluekmen <e.ueluekmen@uib.de> Mon, 16 Jul 2012 16:42:42 +0200

opsi-linux-bootimage (20120119-1) stable; urgency=low

- kernel 3.2.1
- fix french-localisation
- added gptfdisk toolset
- python-opsi 4.0.1.37-1

— Erol Ueluekmen <e.ueluekmen@uib.de> Thu, 19 Jan 2012 15:01:04 +0100

13.1.16 Changelog opsi-depotserver

opsi-depotserver (4.0.2.3-1) stable; urgency=low

- Config-default-creation by first running --init-current-config fixed

— Erol Ueluekmen <e.ueluekmen@uib.de> Tue, 18 Sep 2012 17:42:23 +0200

opsi-depotserver (4.0.2.2-1) testing; urgency=low

- Added opsiFileAdmins handling.

— Erol Ueluekmen <e.ueluekmen@uib.de> Thu, 19 Jul 2012 11:46:20 +0200

13.1.17 Changelog opsi4ucs

opsi4ucs (4.0.2.3-1) stable; urgency=low

- Join script modified:
 - only patch dhcp if it's installed
 - restart and patch samba4 only, if samba4 is installed
- don't build opsi4ucs-ldap-schema.

— Erol Ueluekmen <e.ueluekmen@uib.de> Tue, 02 Oct 2012 15:14:44 +0200

opsi4ucs (4.0.2.2-1) experimental; urgency=low

- For handling with new pcpatch-group added opsi.conf.
- UCSOverview-page changed from applet to java-webstart.
- Added new opsi-auth pam module for UCS 3

— Erol Ueluekmen <e.ueluekmen@uib.de> Thu, 24 Aug 2012 10:17:21 +0200
opsi4ucs (4.0.2.1-1) testing; urgency=low

- Featurepack-Release 4.0.2 for ucs.
- First Release for UCS 3 Support:
 - univention-ldap-backend have reached his EOL since ucs3 (default backend: file)
- Fixes 99opsi4ucs.inst-Joinscript

— Erol Ueluekmen <e.ueluekmen@uib.de> Fri, 29 Jun 2012 10:13:10 +0200

13.1.18 Changelog opsi-atftp

opsi-atftp (0.7.dfsg-2.1) stable; urgency=low

- fix: don't overwrite use_inetd in /etc/default/atftp if exists

-- Erol Ueluekmen <e.ueluekmen@uib.de> Wed, 17 Oct 2012 11:11:37 +0200

opsi-atftp (0.7.dfsg-2) stable; urgency=low

- fix uncommented entries for tftp-service in inetd.conf

-- Erol Ueluekmen <e.ueluekmen@uib.de> Fri, 12 Oct 2012 13:25:16 +0200

opsi-atftp (0.7.dfsg-1.9) testing; urgency=low

- Replaces univention-tftp package

-- Erol Ueluekmen <e.ueluekmen@uib.de> Fri, 10 Aug 2012 11:21:13 +0200

opsi-atftp (0.7.dfsg-1.8) stable; urgency=low

- LSB Header added in init-script

-- Erol Ueluekmen <e.ueluekmen@uib.de> Tue, 24 Jan 2012 18:01:39 +0100