



Opsi Features

opsi
Version 3.3

opsi - OpenSource Desktop-Managementsystem

Client- und Depot-Verwaltung / Management-Interface:

- Mehrfachselektion von Clients und gleichzeitige Bearbeitung
- Speichern und Laden von Gruppen, die zur Selektion von Clients verwendet werden können
- Filtermöglichkeit der anzuzeigenden Clients z.B. nach installierter Software, Softwareversion, Hardware:



Abb. 1: opsi-Configed: Gruppe setzen

- Auswahl der Clients nach depot-server
- Wake on LAN mit dem Konfigurationseditor
- Clientliste sortierbar nach Clientname, Beschreibung und letzter Anmeldung bei opsi:

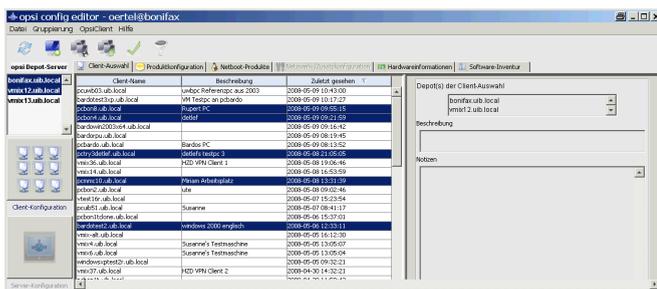


Abb. 2: opsi-Configed Client- und Depotauswahl

- Darstellung des Installationsstatus und der nächsten geplanten Aktion
- Darstellung der installierten und der verfügbaren Softwareversion eines Produktes
- Produktliste sortierbar nach Installationsstatus und Aktionsstatus
- Darstellung des Hardwareinventars
- Einfaches Erstellen und Löschen von Clients
- Konfigurationseditor als Web-Applet im Browser und Java-Applikation

Netbootprodukte: z.B. Betriebssysteminstallation

- Boot per PXE oder über interaktive Client-Boot CD zur Installation neuer oder nicht PXE-fähiger Clients
- Unterstützung von Windows 2000 / XP / 2003
- Paket-basierte unattended Installation
- Einfache Treiberintegration über PCI-Kennungen
- Sichern und Wiederherstellen von NTFS-Images
- Weitere Netbootprodukte wie memtest, wipedisk, ...

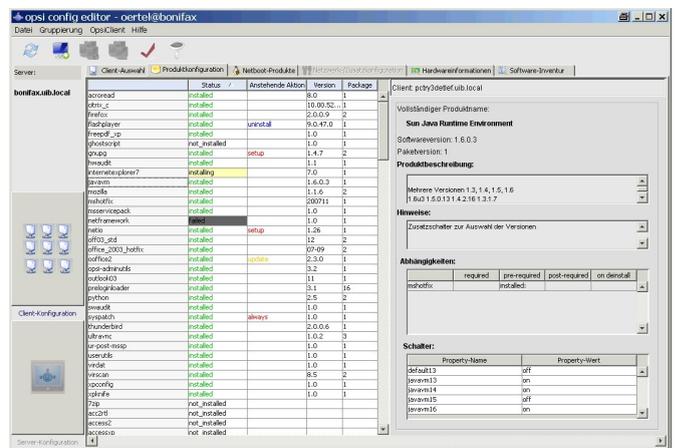


Abb. 3: opsi-Configed: Produktkonfigurationsmaske

Lokalbootprodukte: Softwareverteilung und mehr

- Software-Installation und -Deinstallation mit unterschiedlichen Verfahren wie Silent / unattended Installation, interaktives Setup mit aufgezeichneten Antworten, Differenzsnapshot
- Bearbeitung von Registry, Startmenü- und Desktopeinträgen, Ini-Dateien, XML-Dateien, Textdateien
- 'user defined functions' durch das Ausführen beliebiger Befehle und die Weiterverarbeitung der Ausgaben
- Datei kopieren mit Versionskontrolle
- Reboot- und Shutdownsteuerung
- Logging
- Administrative Tätigkeiten z.B. durch WMI-Aufrufe

Paketformat zur Installation von opsi-Produkten auf einem oder mehreren depot-servern

- Einfache, menügeführte Erstellung
- Installationswerkzeug zur Verteilung auf mehrere depot-server vereinfacht die Verwaltung von mehreren Standorten
- Informationen über die Software- und Paketversion sowie möglicher kundenspezifischer Erweiterungen im Paketnamen, im Installationsverzeichnis und angezeigt im opsi-

Configured zur Unterstützung der Produktverwaltung (Product Lifecycle Management)

- Paketverwaltung ohne root Rechte

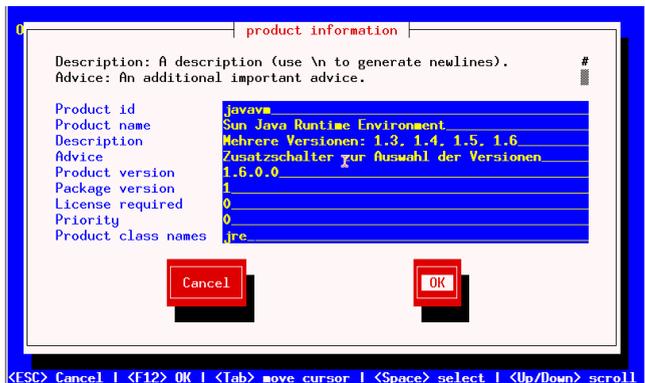


Abb.4: Eingabe der Produktinformationen

Hard- und Softwareinventarisierung

- Hardwareinventarisierung per WMI-Abfrage
- Hardwareinventarisierung aus dem Linux-Bootimage
- Softwareinventarisierung anhand der Registry
- Zentrale Ablage der Inventarisierungsdaten, optional in einem MySQL-Backend mit History-Funktion

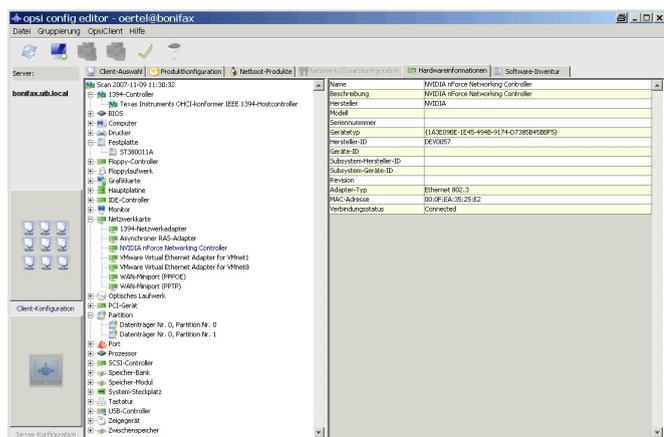


Abb.5: Darstellung des Hardwareinventars

Sonstiges

- Einfaches, initiales Rollout des Client-Agenten
- opsi4ucs: opsi Spezialversion für den Univation Corporate Server (UCS) und Datenhaltung im UCS-LDAP
- opsi4xandros: opsi Spezialversion für den Xandros Server mit Integration in die Xandros Management Konsole

Kontakt

Weitere Informationen zu **opsi** entnehmen Sie unserer Internetpräsenz unter www.uib.de sowie den Projektseiten unter opsi.org.

Gerne beantworten wir Ihre Fragen und führen Ihnen **opsi** vor.



uib gmbh • Bonifaziusplatz 1B • 55118 Mainz
Tel.: 06131 / 27561-0 • Fax: 06131 / 27561-22 • info@uib.de

Konzeptionell

In opsi Version 3 arbeiten alle Komponenten mit dem opsi-Konfigurations-Daemon. Dieser stellt über einen per SSL verschlüsselten Webservice die notwendigen Dienste bereit. Das Lesen und Speichern der Daten aus der Datenhaltung erfolgt durch den Dämonen.

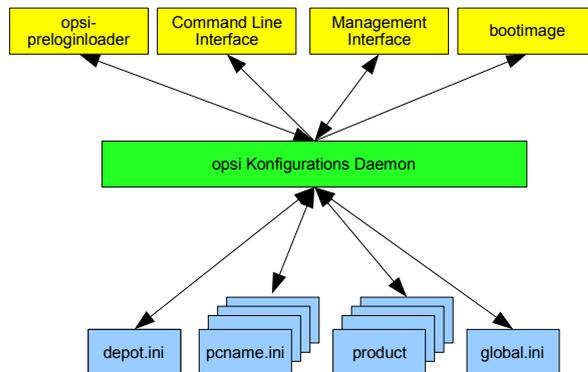


Abb.6: opsi V3: Verwendung eines Webservice zum Datenzugriff

Dadurch wird es relativ einfach möglich, unterschiedliche Datenhaltungen wie z.B. ein dateibasiertes File-Backend oder ein LDAP-Backend als Datenbasis zu bedienen.

Die Umsetzung erfolgt in opsi 3 durch die Bereitstellung einer Python-Library. In dieser sind von der Datenhaltung abstrahierte Aufrufe zur opsi-Konfiguration implementiert. Diese Aufrufe bilden eine allgemeine API zur opsi-Konfiguration.

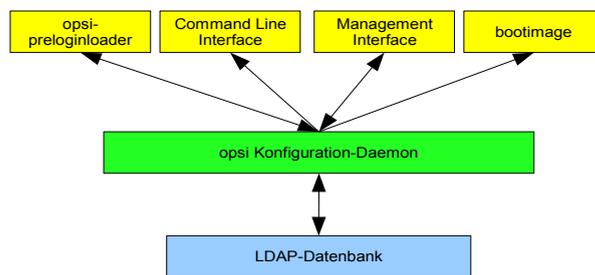


Abb.7: Verwendung alternativer Datenhaltungen durch den Webservice

Diese API wird durch den opiconfigd in einem JSON-basierten Webservice bereitgestellt, der zum Beispiel zur Kommunikation mit dem opsi-Configd verwendet wird. Das Programm opsi-admin stellt wiederum ein Kommandozeilen-Interface zu dieser API zur Verfügung.

Ein weiter Teil der opsi Python-Library implementiert die Zugriffe auf die konkreten Datenhaltungen (Backends), die über den Backendmanager konfiguriert werden können.