

Die „Softwaretankstelle“ mit integriertem Verleihsystem für mobile Geräte

Stefan Thanheiser, Frederic Toussaint, Hartmut Schmeck

Universität Karlsruhe (TH)
Institut für Angewandte Informatik und Formale Beschreibungsverfahren - AIFB
Kaiserstraße 12
76128 Karlsruhe
<nachname>@aifb.uni-karlsruhe.de

Abstract: Dieser Beitrag stellt ein Konzept vor, wie ubiquitäres E-Learning durch eine zielgerechte Bereitstellung von Hard- und Software unterstützt werden kann. Dazu wird eine automatisierte, benutzerzentrierte Softwareverteilung für unterschiedlichste Hard- und Softwareplattformen und ein Verleihsystem für mobile Geräte vorgestellt. Neben den Grundfunktionalitäten umfasst das Konzept unter anderem einen personalisierten „virtuellen Rechner“, automatisierte Sicherheitsupdates der Systeme oder auch die Sicherung von Benutzerdaten und das Einspielen von Softwareupdates innerhalb des Geräteverleih-Zyklus.

1 Motivation

Ein Fahrzeug, das nicht aufgetankt ist, kann nicht fahren. Genauso kann ein Computer, auf dem keine Software vorhanden ist, nicht benutzt werden. Übertragen auf das hier beschriebene Projekt „Softwaretankstelle“ bedeutet dies, dass die für eine Teilnahme am „mobilen Lehren und Lernen“ notwendigen Anwendungsprogramme möglichst einfach und automatisiert zur Verfügung gestellt werden sollen. Vorher müssen eventuell noch geeignete Endgeräte verliehen werden.

So möchte zum Beispiel ein Professor in einem Seminar einen Satz von zehn Notebooks mit benötigtem Betriebssystem und Software zur Verfügung stellen. Ein Dozent möchte in der Vorlesung ein rechnergestütztes Feedbacksystem einsetzen, das aber vorher - möglichst automatisiert - auf allen Geräten der Teilnehmer installiert werden muss. Ein Studierender benötigt kurzfristig für einen Vortrag ein Notebook mit installierter Spezialanwendung, ein anderer Studierender möchte wissen, ob auf seinem privaten Notebook die installierten Programme noch auf dem neuesten Stand sind oder ob sicherheitsrelevante Änderungen durchgeführt werden müssen.

Alle gemeinsam können die Dienste eines zentralen Verleihsystems mit angeschlossener „Softwaretankstelle“ in Anspruch nehmen, welches den alltäglichen, „ubiquitären“ Lehr- und Lernbetrieb an der Universität Karlsruhe (TH) unterstützt.

2 Anforderungen

Die Grundanforderung an eine „Softwaretankstelle“ für mobile Geräte lässt sich relativ einfach beschreiben: Die benötigte Software soll in einem ersten Schritt benutzerfreundlich an einem beliebigen Ort bereitgestellt werden. In weiteren Schritten soll der weitere Ablauf im Lebenszyklus der Software begleitet und unterstützt werden (siehe hierzu auch [Ca97]).

Von einem Verleihsystem erwartet man, dass Leihnotebooks bzw. beliebige mobile Geräte zur gewünschten Zeit in gewünschter Anzahl zur Verfügung gestellt werden können. Zusätzlich sollen diese Geräte automatisch nach individuellen Vorgaben des Benutzers installiert und konfiguriert worden sein. Für die anfallenden Arbeiten sollen sich die Benutzer von allen Rechnerplattformen aus eindeutig authentifizieren können.

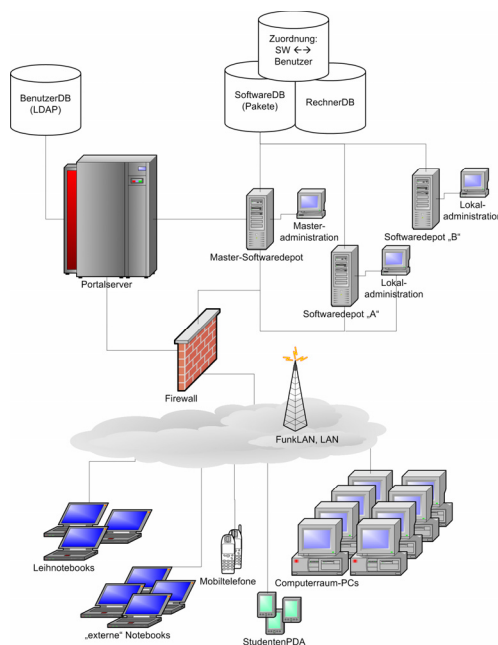


Abbildung 1 - Struktur der „Softwaretankstelle“

Da nicht nur Leihgeräte mit Software versorgt werden sollen, muss die „Softwaretankstelle“ eine große Anzahl heterogener Geräte mit unterschiedlichsten, für den Betreiber zum Teil unbekanntesten Hard- und Softwarekonfigurationen zuverlässig und sicher unter Benutzung unterschiedlicher Netzwerke mit Software bedienen können. Im Unterschied zu den meisten auf dem Markt befindlichen Systemen soll das System zudem keine rechnerbasierte, sondern eine benutzerbasierte Softwareverteilung realisieren. Das bedeutet, dass ein Benutzer, der das Recht hat, eine Software zu installieren, diese an allen Rechnern installieren kann, an denen er sich anmelden kann.

Neben automatisch durchgeführten Updates soll der Benutzer selbstständig von der „Softwaretankstelle“ aktuelle bzw. aktualisierte Software in sein geliehenes oder eigenes System übernehmen können, ohne bisherige Installationen zu stören oder Lizenzbedingungen zu verletzen. Dabei müssen für die Softwareverteilung auf mobilen Clients die speziellen Rahmenbedingungen des WLANs im Sinne von Robustheit und Bandbreitenbeschränkungen berücksichtigt werden.

Im Rahmen des Notebookverleihs soll es möglich sein, zum Ende der Leihzeit private Daten und Einstellungen zur Weiter- und Wiederverwendung zu sichern. Dadurch kann erreicht werden, dass der Benutzer bei jedem Verleih „sein virtuelles persönliches“ Gerät erhält.

3 Vorarbeiten

Vor dem eigentlichen Design der „Softwaretankstelle“ und des Notebookverleihs als Gesamtsystem wurde überprüft, welche Benutzergruppen angesprochen, welche Typen von Geräten unterstützt und welche Software angeboten werden soll. Ein weiterer wichtiger Punkt war zu Beginn die Auswahl des zugrunde liegenden Softwareverteilungssystems, da eine komplette Eigenentwicklung nicht sinnvoll erschien, die Anpassungsfähigkeit an das universitäre Umfeld aber sehr wichtig war. Projektbegleitend wurden die rechtlichen Aspekte einer Softwareverteilung und eines Geräteverleihs untersucht.

3.1 Zielsysteme und Benutzergruppen

Durch Umfragen und eine Auswertung von Webseiten wurde festgestellt, dass ca. 88% der Rechner an der Universität Karlsruhe unter Windows betrieben werden, ca. 10% unter Linux und der Rest unter UNIX oder Apple-Betriebssystemen. Unser Ziel war es, neben Windows auch Linux möglichst gut zu unterstützen, Apple-Systeme aber vorerst zurückzustellen. Die Gruppe der möglichen Benutzer bzw. Zielsysteme einer „Software-tankstelle“ stellte sich als sehr heterogen heraus. Insgesamt wurden vier Benutzergruppen und deren Geräte kategorisiert, die mit einigen Eigenschaften in Tabelle 1 beschrieben werden.

Tabelle 1 – Kategorisierung der Nutzergruppen und deren Geräte an einer „Software-tankstelle“

Benutzer von Leihgeräten	Studierende und Mitarbeiter mit privaten Rechnern	Rechner für Studierenden in Computerräumen	Rechner an Instituten und Lehrstühlen
<ul style="list-style-type: none"> ▪ Mobile Geräte (Notebooks, PDAs) ▪ Momentan: relativ homogener Hardwarepool, Tendenz: Diversifikation ▪ Verleihzeitraum: zwischen zwei Stunden und zwei Wochen ▪ Anzahl: 10-20 Systeme, Tendenz steigend. 	<ul style="list-style-type: none"> ▪ Alle von Extern in die Universität eingebrachten Rechner-systeme ▪ Sehr heterogene Hard- und Software-kombinationen ▪ Netzanbindung: hauptsächlich über WLAN ▪ Anzahl: ca. 200 bis 300 Nutzer, potentiell einige 1000 	<ul style="list-style-type: none"> ▪ Systeme in öffentlichen. Computerräumen ▪ Relativ homogene Rechner ▪ Windows- bzw. Linux-Systeme oder auch Dual-Boot-Systeme ▪ Netzerkanbindung: LAN (100Mbit/s) ▪ Anzahl: 100-1000 PCs 	<ul style="list-style-type: none"> ▪ Rechner im Umfeld der Institute und Lehrstühle ▪ Sehr heterogene Systemlandschaft ▪ Viele Windows-systeme, wenige Linux-systeme

Die sehr unterschiedlichen Benutzergruppen zeigen, dass zum einen sehr viele Clients gleichzeitig versorgt werden müssen. Andererseits können einzelne Clients sehr individuelle Konfigurationen und auch unter Umständen langsame Verbindungen besitzen.

3.2 Auswahl eines Verteilungssystems

Vor Beginn des Systemdesigns wurden die am Markt verfügbaren Softwareverteilungssysteme analysiert. Als erste Testgruppe konnten 19 Produkte identifiziert werden,

welche den Grobanforderungen genügen, indem sie auf unterschiedliche Weise die Verteilung von Software und kompletten Betriebssystemen im Intranet ermöglichen. Anhand von ca. fünfzig Merkmalen wurden sechs dieser Produkte ausgewählt, genauer untersucht und auf Eignung für das universitäre und mobile Umfeld geprüft [CFHHHW98].

Unsere Wahl fiel unter anderem aufgrund des benutzten Installationsprinzips auf ein Produkt, welches neben dem marktüblichen Verteilungsverfahren über so genannte „MSI-Pakete“ auch das „Scripting“ von Hersteller-Installationsroutinen bereitstellt.

Beim „Scripting“ wird die Installation einer Applikation durchlaufen und dabei alle Mausclicks, Texteingaben, etc. aufgezeichnet. Diese werden bei der späteren Installation auf dem Zielrechner „abgespielt“ und somit das Produkt automatisch installiert. Hauptnutzen dieses Verfahrens ist die vom Installationsprogramm an das jeweilige System angepasste Installation, im Vergleich zum so genannten Snapshot-Verfahren¹ – eine auf den jeweiligen Zielrechner abgestimmte Installation. Zudem erfüllt man eine wichtige Bedingung für spätere Supportanfragen, da die meisten Softwarehersteller für den Fall einer Snapshot- oder Imageinstallation keine Unterstützung gewähren.

Ein weiterer Vorteil des gewählten Basissystems ist dessen plattformunabhängige Java-API, über die durch selbst entwickelte Komponenten das Verteilungssystem gesteuert werden kann. Später zeigte sich, dass einige Komponenten nicht wie versprochen funktionierten, eigene Patches geschrieben werden mussten und zahlreiche Updates zu ständigen Anpassungen unserer Schnittstellen führten. Auch die Skripterstellung dauerte wesentlich länger als ursprünglich geplant.

3.3 Softwaredatenbank

Die Gruppe der „Freeware“-Programme stellt eine sehr große Anzahl an kostenlosen Anwendungsprogrammen zur Verfügung. Ein Zusatznutzen dieser Softwaregruppe liegt für unser Projekt darin, dass die meisten Applikationen nicht lizenzrechtlich geschützt sind und somit über die „Softwarebankstelle“ frei verteilt werden können. Um der großen Menge an angebotenen Anwendungen und dem zum Teil erhebliche Qualitätsunterschied gerecht zu werden, entschieden wir uns für den Aufbau einer Softwaredatenbank, in welcher Freeware analysiert, bewertet und kategorisiert wird. Ziel war, die für das universitäre Umfeld am besten geeigneten Produkte herauszufiltern. Bis jetzt wurden ca. 100 Programme in die Datenbank eingepflegt. Über eine Webschnittstelle sind die Daten (im Moment nur intern) am Karlsruher Universitätsrechenzentrum abrufbar.

3.4 Rechtliche Rahmenbedingungen

Solange man über ein eigenes Verteilungssystem nur Freeware verteilen möchte, ist die Betrachtung urheberrechtlicher Aspekte relativ unproblematisch.

¹ Beim „Snapshot“ wird auf einem Referenzrechner eine Differenz zwischen dem Status des Systems vor und nach der Installation einer Anwendung gebildet. Diese Differenz wird auf den Zielrechner „aufgespielt“.

Bei Shareware und besonders bei kommerzieller Software sind die jeweiligen Lizenzbestimmungen speziell unter dem Aspekt der benutzer- oder rechner-spezifischen Nutzungsbedingungen zu prüfen. Eine zentrale Frage ist hierbei die rechtliche Ausgestaltung von Nutzungsbedingungen für die „Softwaretankstelle“ und die technische Realisierung von Zugangs- und Nutzungskontrollen etc., um potentiellen Problemen der Softwarepiraterie und des Lizenzmissbrauchs zu begegnen.

Ein weiteres Thema in diesem Kontext betrifft Datenerhebung und Datenschutz. So stellt sich die Frage, welche Daten für den Betrieb des Systems erhoben werden müssen oder welche Daten bereits an anderer Stelle auf dem Campus vorhanden sind und eingesehen werden können.

Die an dieser Stelle skizzierten rechtlichen Fragestellungen unseres Projektes werden aktuell am Institut für Angewandte Informatik und Formale Beschreibungsverfahren im Rahmen einer Diplomarbeit ausführlich untersucht.

4 Design des Gesamtsystems

Insgesamt gliedert sich das Gesamtsystem in drei Bereiche:

- **Softwareverteilung:** Sie stellt die Grundfunktionalität der Softwareverteilung bereit. Zusätzlich werden Hardware-, Software- und Benutzerinformationen in einer Datenbank zur Verfügung gestellt und über entsprechende Tabellen miteinander in Beziehung gesetzt. Somit wird es möglich, Rechte und Abhängigkeiten zu modellieren.
- **Verleihsystem:** Das Verleihsystem baut auf der Softwareverteilung auf und realisiert ein Verleihverwaltungssystem (mit Scheduling, etc.), das vor Geräteausgabe eine individuelle Konfiguration und Installation ermöglicht.
- **Portal:** Das Portal wird als Benutzeroberfläche eingesetzt, um für alle Teilsysteme einen einheitlichen Zugriff zu ermöglichen. Es ermöglicht via LDAP eine zentrale Benutzerauthentifikation und –autorisierung und stellt den Benutzern ein persönliches Softwaremanagement bereit.

Ein Großteil der Arbeiten bestand darin, zwischen allen Bereichen sinnvolle Schnittstellen zu definieren und diese Schnittstellen mit Skripten zu Verknüpfen. Durch die Vielzahl der beteiligten Einzelsysteme war es nicht möglich, alle Schnittstellen einheitlich zu erstellen:

- Zwischen Portal und Softwareverteilung konnte auf JSP zurückgegriffen werden. Hiermit werden Client-Server-Anfragen des Verteilungsprogramms gesteuert, LDAP-Anfragen gestellt und dem Portal dynamisch Daten geliefert. Ebenfalls mittels JSP erfolgte die Zusammenstellung der benutzerabhängigen Installationskripte innerhalb der Softwareverteilung.

- Die Installationsskripte selbst, welche die Einrichtung der einzelnen Programme auf dem Zielrechner steuern, wurden mit einer proprietären Software erstellt.
- Skripte, die in das System von Windows eingreifen mussten, wurden in Visual-Basic implementiert, da nur damit auf bestimmte Befehlssätze zugegriffen werden konnte.

Im Folgenden werden die drei Hauptelemente Softwareverteilung, Verleihsystem und Portal ausführlich dargestellt.

5 Softwareverteilung

Je nach Benutzergruppe unterscheiden sich die Anforderungen an die Softwaretankstelle (siehe Tabelle 2). Während ein Studierender mit privaten Notebooks eher ein einzelnes Programm installieren möchte, müssen bei Computerpools gleichzeitig für viele Rechner Betriebssysteme und viele Anwendungsprogramme zur Verfügung gestellt werden.

Tabelle 2 - Anforderungsprofile der verschiedenen Nutzergruppen

	Leihgeräte	Geräte von Studierenden und Mitarbeiter	Rechner in Computerräumen	Geräte an Instituten und Lehrstühlen
Betriebssystem:				
Windows	Häufig	Nie ²	Häufig	Selten
Linux	Selten	Nie	Selten	Sehr Selten
Anwendungs-SW				
Windows	Häufig	Häufig	Häufig	Häufig
Linux	Selten	Selten	Selten	Selten
PDA	Selten	Selten	Nie	Selten

Im Folgenden werden die Möglichkeiten beschrieben, die mit dem Prototyp der „Softwaretankstelle“ realisierbar sind.

5.1 Betriebssysteminstallation

Grundsätzlich existieren unabhängig vom verwendeten Betriebssystem zwei Verfahren für die „ferngesteuerte“ Installation eines Betriebssystems:

Zum einen können so genannte Images/Abbilder des Betriebssystems eines Referenz-PCs auf ein Zielsystem kopiert werden, wobei auf evtl. Unterschiede in der Hardware keine Rücksicht genommen wird. Das so kopierte Betriebssystem ist in dieser Form auf dem Zielrechner meist nicht lauffähig. Aus diesem Grund unterstützt Windows XP ein „Minisetup“, das nach dem Verteilen nochmals eine Hardwareerkennung durchführt und das System automatisch auf die neu erkannte Hardware einstellt. Dieses Minisetup

² Für die Betriebssystem-Administration auf externen Notebooks sind die jeweiligen Besitzer selbst verantwortlich.

funktioniert, solange keine zu großen Unterschiede zwischen Referenz- und Zielsystem vorliegen. Trotzdem müssen relativ viele Referenzsysteme vorgehalten werden, um ein funktionierendes System auf unterschiedlichsten Geräten erhalten zu können. Unseres Wissens ist unter Linux diese Möglichkeit der nachträglichen Anpassung an unterschiedliche Hardware „mit Bordmitteln“ nur sehr eingeschränkt vorhanden.

Beim zweiten Verfahren handelt es sich um ein „unattended Setup“ wobei die im Normalfall „von Hand“ zu treffenden Einstellungen bereits in einer „Antwort-Datei“ vorliegen. Dabei wird die Installation automatisch an die vorhandene Hardware angepasst.

Innerhalb der „Softwaretankstelle“ kommt hauptsächlich die „unattended“-Variante in Betracht, weil mit einer nicht einzugrenzenden Menge von unterschiedlichen Hardwarekonfigurationen gerechnet werden muss. Zudem ist es nur mit der „unattended“-Installation bei einem späteren Ausbau der „Softwaretankstelle“ möglich, Benutzerwünsche in die Antwort-Datei einfließen zu lassen.

Das für die „Softwaretankstelle“ gewählte Basissystem unterstützt die unattended-Lösung, kann aber durch Zusatzprodukte auch Komplettimages verteilen³. Zusätzlich unterstützt es sowohl die Installation von Windows-Betriebssystemen als auch von Linux-Distributionen wie SuSE und RedHat.

Technisch wurde für die Netzwerkinstallation eines Betriebssystems folgender Weg gewählt: Der Zielrechner wird über das Netzwerk gebootet und erhält via DHCP einen Boot-Agenten, welcher den Rechner beim Softwaredepot anmeldet. Liegt auf dem Depotserver ein Installationsauftrag vor, werden die Installationsdateien auf den Rechner lokal übertragen. Anschließend wird das Unattended-Setup gestartet und vom Agenten bis zum erfolgreichen Abschluss kontrolliert.

5.2 Installation von Anwendungsprogrammen

Ähnlich wie bei der Betriebssysteminstallation existieren auch bei Anwendungsprogrammen zwei grundlegende Varianten der Softwareverteilung.

Die erste Methode unter Windows ist wiederum die Verteilung via „Snapshot“ bzw. als so genanntes Microsoft-Installer- (MSI)-Paket [MSDNLib04]. Bei Snapshots/MSI-Paketen wird auf einem Referenzsystem der Rechnerzustand vor der Installation mit dem Rechnerzustand nach der Installation des Anwendungsprogramms verglichen. Das „Veränderungsabbild“ wird gespeichert und auf andere Rechner übertragen, wodurch dort die gleichen Systemveränderungen durchgeführt werden.

Daneben können auch bei Anwendungsprogrammen Skripte genutzt werden, um das Originalsetup des Herstellers automatisch ablaufen zu lassen. Dazu werden die originalen Installationsdateien des Softwareherstellers auf den Client übertragen und das Setup gestartet. Ein auf dem Clientsystem installierter Agent erhält von dem Softwarevertei-

³ Komplettimages werden bei bekannter Hardware und zeitkritischen Installationen eingesetzt.

lungssystem ein vorgefertigtes Skript, welches die Installation steuert (vgl. [HHHW97]). Der Agent führt exakt dieselben Schritte aus, die auch ein menschlicher Benutzer bei der Installation durchführen würde.

Für die Erstellung der Skripte wird vom Hersteller der Verteilungssoftware eine Entwicklungsumgebung zur Verfügung gestellt, welche Installationsvorgänge automatisch aufzeichnet und zusätzlich die Bearbeitung der Skripte mit Konstrukten wie Schleifen und Konditionalabfragen unterstützt.

Die „Softwarebankstelle“ unterstützt im Windows-Umfeld sowohl bei der Betriebssystem- als auch bei der Softwareinstallation die Skript- und die Paketversion. Darüber hinaus ist es möglich, gewisse Abhängigkeiten im Softwaredepot zu modellieren. Beispielsweise wird es so möglich, vor Installation eines Plugins den Zielrechner auf das Vorhandensein des jeweiligen Hauptprogramms zu prüfen und dieses gegebenenfalls auch zu installieren.

Im Linux-Umfeld wird im Allgemeinen für die Installation von Software hauptsächlich auf RPM-Pakete zurückgegriffen. Der „Softwarebankstelle“ ist es möglich, RPM-Pakete via Agent auf Linux-Zielsysteme zu verteilen und die RPM-Installation anzuschließen. Ein leider in diesem Bereich für uns noch ungeklärtes Problem ist die bei der RPM-Installation durchgeführte Abhängigkeitsprüfung. Ein generelles Problem im Linux-Bereich ist zudem die Vielfalt an unterschiedlichen RPM-„Distributionen“. Da der Zusatzaufwand der Pflege aller unterschiedlichen Pakete sehr hoch ist, die Anzahl der derzeitigen Linuxbenutzer im Gegensatz dazu aber gering ausfällt, wurde hier eine weniger aufwändige Lösung gesucht. Angedacht ist innerhalb des Softwarebankstellen-Gesamtsystems ein Ausweichen auf die von den Distributionen bereitgestellten Update-Tools. In diesem Fall könnte unsererseits das Update lediglich initiiert werden, die Kontrolle, ob die Software erfolgreich installiert wurde, müsste der Benutzer übernehmen.

Zusätzlich kann noch eine Software zur Verfügung gestellt werden, die sowohl unter Windows als auch Linux die Onlineverbindung zu einem X-Server erlaubt, so dass zumindest auf diesem Wege die Programme zur Verfügung gestellt werden können.

5.3 Erweiterungen

Neben der Installation von Betriebssystem und Anwendungsprogrammen sollten weitere Funktionen zur Verfügung stellen. Durch die Bereitstellung von Bugfixes und Patches für Windows-Software, der Möglichkeit, Software direkt auf einem PDA zu installieren und – zumindest teilweise – der Unterstützung von Dual-Boot-PCs wird die „Softwarebankstelle“ zu einem ganzheitlichen System, das bei allen Situationen genutzt werden kann, die Änderungen an Softwarepaketen nötig machen.

5.3.1 Bugfixes und Patches

Bugfixes und Patches unterscheiden sich bei einem Windows-Betriebssystem in ihrer Installation nicht oder nur geringfügig von der Installation von „normalen“ Anwendungen. Deshalb sind auch sie meist problemlos über die „Softwaretankstelle“ und die beschriebenen Verfahren zu verteilen. Kritisch kann das Einspielen von Patches jedoch in speziellen Konstellationen werden, wie sich beispielsweise anhand des Virus' W32/Blaster alias „Lovesan“ zeigen sollte. Ausgangskonstellation war ein mittels Unattended-Setup neu mit Windows XP versehener Rechner, bei welchem noch keine Updates oder Patches für das System eingespielt waren. Wie sich herausstellte, war eine „ungeschützte“ Zeitspanne von weniger als einer Minute (zwischen Ende der Betriebssysteminstallation und Start der Updates) ausreichend, dass der Zielrechner vom Virus befallen und „außer Gefecht“ gesetzt wurde. Deshalb wurde ein Skript entwickelt, welches direkt nach der Installation die Software-Firewall von Windows XP auf den betroffenen Ports aktiviert, um den Virusbefall zu verhindern. Ein weiteres Skript ermöglicht es, gleichzeitig das in Windows integrierte System-Update zu starten, wodurch umgehend über die entsprechenden Internetseiten von Microsoft oder vom lokalen System-Update-Server (SUS) die aktuellen Updates eingespielt werden.

5.3.2 PDAs und Handhelds

Auch PDAs - oder genauer PocketPCs - können über die „Softwaretankstelle“ mit Anwendungsprogrammen betankt werden. Allerdings ist eine komplette Betriebssysteminstallation nicht möglich, da diese in der Regel durch einen kritischen Flash-Vorgang durchgeführt werden muss.

PDAs können ebenfalls in zwei Varianten mit Anwendungssoftware versorgt werden: Die Installation über den Host-PC des PDA per Synchronisation und die direkte Installation auf dem PDA, sofern dieser beispielsweise über eine Funk-Netzwerkkarte verfügt. Die erste Variante wird analog zu einer „normalen“ Anwendungsinstallation unter Windows auf dem Host-PC initiiert, welcher wiederum die Installation per Active-Sync-Modus mit dem PDA übernimmt. Für die zweite Variante - also den direkten Netzwerkanschluss des Handhelds - existiert ein Agent auf dem PDA, der diesen beim Softwaredepot anmeldet. Analog zur Softwareverteilung unter Windows wird die zu installierende Software per Netzwerk direkt auf den PDA übertragen und vom Agenten installiert.

5.3.3 Dual Boot

Unter „Dual Boot“ verstehen wir die Eigenschaft eines Rechners, zwei installierte Betriebssysteme zu unterstützen. Dabei kann mit einem Bootmanager beim Start des Rechners das gewünschte Betriebssystem ausgewählt werden. Die Variante „Dual Boot“ tritt insbesondere bei Rechnern in Computerräumen auf, wo den Studierenden neben Windows auch eine Linux-Umgebung zur Verfügung gestellt werden soll. Hierbei müssen einige Einschränkungen berücksichtigt werden, da bei einem erforderlichen Reboot des Zielrechners vom Verwaltungssystem nicht dynamisch ausgewählt werden kann, welches Betriebssystem gestartet wird.

Eine komplette Dual-Boot-Umgebung mit allen Betriebssystemen ist auf Basis eines Images verteilbar. Anwendungsprogramme können für das Betriebssystem nachinstalliert werden, welches vom Bootmanager automatisch gestartet wird. Dieser Ansatz wurde mit einigen Änderungen und Einschränkungen bereits vor Entwicklung der „Softwaretankstelle“ von der Fakultät für Wirtschaftswissenschaften und vom Rechenzentrum erfolgreich getestet.

6 Verleihsystem

Neben der „Softwaretankstelle“ als Instrument der Softwareverteilung wurde ein Verleihsystem für mobile Geräte untersucht. Das Verleihsystem realisiert ein Reservierungs- und Konfigurationssystem für mobile Geräte, welches für die eigentliche Softwareverteilung auf die Dienste der Softwaretankstelle zurückgreifen kann. Über das Verleihsystem ist es jedem Universitätsmitglied möglich, nach Angabe seiner Daten ein mit Betriebssystem und Software ausgestattetes Gerät für einen bestimmten Zeitraum auszuleihen. Mobile Geräte umfassen in diesem Zusammenhang vor allem Notebooks unterschiedlicher Qualitäts- und Geschwindigkeitsklassen, Tablet PCs und auch PDAs.

Das Verleihsystem ist als eine integrierte Prozesskette von Informationsverarbeitungs- und Softwareverteilungsprozessen konzipiert. Es basiert auf einem exemplarischen Verleihzyklus (siehe Abbildung 2).

Zu Beginn wählt der Benutzer über die Benutzerschnittstelle im Web (siehe hierzu auch das folgende Kapitel) ein Gerät aus und gibt einen Wunschzeitraum für den Verleih an. Zusätzlich kann der Benutzer für seinen Ausleihzeitraum Betriebssystem und Anwendungsprogramme auswählen. Nach einer Prüfung durch das System (Berechtigung, Verfügbarkeit, Plausibilität, etc.) werden die Daten gespeichert und zum gegebenen Zeitpunkt ein Mitarbeiter der Verleihstelle aufgefordert den Installationsvorgang zu initialisieren. Die Installation selbst wird von der Softwareverteilung durchgeführt, wobei die Vorgaben aus dem Verleihsystem übernommen wurden. Nach der Installation können optional vom Benutzer angegebene Daten seines „virtuellen Notebooks“ auf den Rechner überspielt werden. Beim „virtuellen Notebook“ handelt es sich um Daten, die von einem früheren Notebookverleih stammen oder jetzt zusätzlich zu den Anwendungsprogrammen zur Verfügung gestellt werden sollen (z.B. Vorträge und persönliche Einstellungen). Anschließend kann das Gerät übergeben werden. Während der Nutzungsphase kann der Benutzer über das Portal Software nachinstallieren. Bei der Geräterückgabe können optional die Benutzerdaten von dem mobilen Gerät gesichert werden und stehen somit für den nächsten Leihzyklus wieder zur Verfügung. Die konkrete Realisierung des Verleihsystems wird zurzeit im Rahmen einer Diplomarbeit genauer entwickelt und ausgearbeitet.

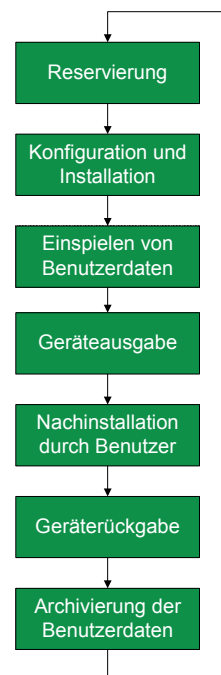


Abbildung 2
Verleihzyklus

7 Benutzerschnittstelle (Portal)

Die „Softwaretankstelle“ soll einen zentralen Dienst auf dem Campus darstellen. Aus diesem Grund wird die Einbindung der Benutzerschnittstelle in eine zentrale Anlaufstelle für alle Universitätsangehörigen angestrebt. Eine solche Anlaufstelle wurde in Form eines Portals zusammen mit dem NUKATH-Projekt „Campus Mobile Communication Center“ entwickelt [NUKATH04]. Das „Softwaretankstellen-Portal“ übernimmt die Authentifizierung und Autorisierung der Nutzer. Anschließend stellt es den benutzerangepassten Zugriff auf die jeweils „persönliche“ Softwaretankstelle zur Verfügung.

Der von uns als Plattform herangezogene Portalserver stellt nicht nur einen zentralen Webserver samt Portal zur Verfügung – er verfügt zudem sowohl über eine Benutzerverwaltung via LDAP-Verzeichnisdienst als auch über einen Application Server für J2EE-Anwendungen. Das von der „Softwaretankstelle“ benutzte Softwareverteilungssystem verfügt über eine passende Java-API, welche das „Fernsteuern“ des Softwaredepots vom Portal (bzw. von einer J2EE-Anwendung innerhalb des Portals) aus gestattet.

Die Ablaufplanung für eine typische Portalsitzung an der „Softwaretankstelle“ sieht folgendermaßen aus: Der Benutzer meldet sich beim Portal mit seinem campusweit gültigen Benutzernamen an und wird authentifiziert. Über eine Schnittstelle zu einem LDAP-Verzeichnis können je nach Bedarf weitere Informationen bezogen werden. Über diese Zusatzdaten kann der Benutzer zu bestimmten Aktionen innerhalb der „Softwaretankstelle“ autorisiert werden. Als Aktionen stehen dem Benutzer zur Verfügung: Reservierung von Notebooks, Neuregistrierung eines Rechners, Installation von Software und Erstellung von Softwareprofilen. Über diese Aktionsfelder hinaus besteht für den Nutzer die Möglichkeit, sich einen Überblick über seine Daten zu verschaffen, insbesondere welche Software auf dem aktuellen Rechner von der „Softwaretankstelle“ installiert wurde. Sofern bereits Software installiert wurde, wird beim Anmelden am Portal überprüft, ob Updates verfügbar sind und dies dem Benutzer gemeldet. Für den Administrator der „Softwaretankstelle“ bzw. für nachgeordnete administrative Stellen ist innerhalb des Portals ein spezieller Bereich vorgesehen, welcher einen erweiterten Zugriff auf das Softwaredepot erlaubt. Über diese spezielle Benutzerschnittstelle können so insbesondere Benutzer und Softwarepakete verwaltet werden.

8 Mögliche Erweiterungen

Zusätzlich zur beschriebenen Funktionalität der „Softwaretankstelle“ sind Erweiterungen denkbar, die den Dienst aufwerten können. Einige Beispiele dazu sind:

- Location Based Services: Im NUKATH-Projekt „Interdisziplinäre Kooperationsmodule in mobilen Netzen“ wurde ein Ansatz entwickelt, mobile Geräte innerhalb des Funknetzes der Universität zu lokalisieren - entsprechendes Einverständnis des Benutzers vorausgesetzt. Somit könnten umgebungsabhängige Dienste wie die Verbindung zum jeweils nächstgelegenen Drucker oder das Herunterladen der gerade in diesem Raum für eine Vorlesung benötigten Software angeboten werden.

- Mobiler Helpdesk, Rechnerkonfiguration über das Netz: Sollte der Benutzer Probleme mit einem Leihgerät bzw. installierter Software haben, wäre es über das Portal per Chat oder ähnlichen Technologien möglich, Hilfe von einem Mitarbeiter der Verleihstelle anzufordern bzw. zu erhalten. Darüber hinaus wäre über das Netzwerk auch eine „Fernwartung“ denkbar.
- Ad-hoc Groupware: Ebenfalls über das Portal realisierbar wäre eine Ad-Hoc-Arbeitsumgebung für Kleinarbeitsgruppen. Denkbar wären hier klassische Tools wie Whiteboard, Chat, etc. bis hin zur Telekonferenz über das Netzwerk. Die benötigte Software kann dabei direkt installiert werden.

9 Fazit

Im vorliegenden Projekt wurde ein Prototyp für eine „Softwaretankstelle“ aufgebaut und das Konzept für ein angeschlossenes personalisiertes Verleihsystem erarbeitet. In zwei noch laufenden Diplomarbeiten werden die bisherigen Ergebnisse noch erweitert, bzw. vertieft. Der klassische Ansatz, Software an einem bestimmten Arbeitsplatz zur Verfügung zu stellen, wurde erweitert um die Möglichkeit, dass ein mit entsprechenden Rechten ausgestatteter Benutzer „seine“ Software selbst an jedem Rechner nachinstallieren kann. Es konnte eine einheitliche Umgebung zur Verfügung gestellt werden, unter der die Betriebssysteme Windows und Linux, sowie Anwendungsprogramme für Windows, Linux und Pocket-PC angeboten werden können. Durch den so realisierten „Basisdienst Softwaretankstelle“ vereinfacht sich nicht nur die „allgemeine“ Softwareverteilung auf dem Campus. Der Dienst schafft für viele Studierende und Dozenten einen einfacheren Zugang zu E-Learning-Angeboten und ermöglicht deren flächendeckendere Nutzung. Er ermöglicht eine Ausstattung mit den „Mitteln zum Zweck“ und erneuert, vervollständigt bzw. optimiert diese Mittel auch lehr- und lernbegleitend.

Literaturverzeichnis

- [Ca97] A. Carzaniga – „A Characterization of the Software Deployment Process and a Survey of related Technologies“, Technical Report, Dipartimento di Elettronica e Informazione, Politecnico di Milano, September 1997, S. 2-4
- [CFHHHW98] A. Carzaniga, A. Fuggetta, R.S. Hall, D. Heimbigner, A. van der Hoek, A.L. Wolf – „A Characterization Framework for Software Deployment Technologies“, Technical Report CU-CS-857-98, Dept. of Computer Science, University of Colorado, April 1998.
- [HHHW97] R.S.Hall, D. Heimbigner, A. van der Hoek, A.L. Wolf – „The Software Dock: A Distributed, Agent-based Software Deployment System“, CU-CS-832-97, Department of Computer Science, University of Colorado, Boulder, Colorado, Februar 1997
- [MSDNLib04] Microsoft MSDN Library Page – „Overview of Windows Installer – Installation Package“ http://msdn.microsoft.com/library/default.asp?url=/library/en-us/msi/setup/installation_package.asp
- [NUKATH04] Projekt Notebook Universität Karlsruhe (TH), <http://www.nukath.uni-karlsruhe.de/>, 2004