

11 Linux-Server für Windows-Anwendungen

Windows-Anwendungen gelten für viele Anwender und Entscheider als so populär, dass sie sich auch für Windows-Betriebssysteme als Arbeitsumgebung entscheiden. Für diese Leser stellt dieses Buch dar, wie Linux-Server einer IT-Landschaft mit Windows-Clients wichtige Serverdienste zur Verfügung stellen können. Doch ist diese Kopplung von Windows-Anwendungen und Arbeitsplatz-Betriebssystem nicht zwangsläufig. Auch Linux-Clients können durchaus Windows-Anwendungen darstellen oder gar laufen lassen. Es gibt bereits mehrere kommerzielle und freie arbeitsplatzbasierte und serverbasierte Lösungsansätze, um Windows-Anwendungen auf Linux-Clients darzustellen. Welcher dieser Lösungsansätze in Frage kommt, dürfte u. a. von der Anzahl der Clients, die Windows-Applikationen nutzen möchten, der Nutzungsintensität und Nutzungsdauer abhängen.

11.1 Windows-Emulatoren am Linux-Arbeitsplatz

Möchte man eine Windows-Anwendung auf einem Linux-Arbeitsplatz ablaufen lassen,

- so kann man entweder ein Windows und ein Linux unabhängig voneinander auf der Festplatte installieren und für die Windows-Anwendung gegebenenfalls dann Windows anstatt Linux booten.
- Oder durch Emulatoren wie VMWare (<http://www.vmware.de>), oder Netraverse Win4Lin (<http://www.netraverse.com>). eine Umgebung einrichten, in der Windows auf Linux läuft. So kann man eine komplette Windows-Umgebung auf einem Linux-Betriebssystem installieren. Beide Betriebssysteme laufen dann gleichzeitig, man kann Fenster beider Betriebssysteme gleichzeitig öffnen und über eine gemeinsame Zwischenablage Daten austauschen. Solche Emulatoren erfordern mehr PC-Ressourcen, da ja zwei Betriebssysteme gleichzeitig laufen. Bei kommerzieller Software erfordert das zusätzliche Lizenzen sowie administrativen Aufwand zum Verwalten beider Betriebssystem-Umgebungen. Dafür spart man zusätzliche Server oder eine Netzwerkinfrastruktur.
- Eine Teilmenge von Windows-Anwendungen wie die Textverarbeitung, die Tabellenkalkulation und die Präsentation von Microsoft Office kann auf Linux-PCs mit installiertem freiem »Wine« (<http://www.winehq.com>) oder stabiler auf dem kommerziellen »CrossOver Office« von Codeweavers (<http://www.codeweavers.com>) direkt ohne ein Windows laufen, als wären sie auf Windows »nativ« installiert. Crossover Office integriert Microsofts Office- und Internet-Programme dabei sehr elegant in KDE.

11.2 Applikations-Server

Statt Windows-Anwendungen auf einzelnen Linux-Clients auszuführen, kann man hierfür eigene Applikations-Server mit Server-Betriebssystemen verwenden. Die Kommunikation zwischen Applikations-Server und Client erfolgt nun über ein Netzwerkprotokoll. So wie man gewohnt ist, von einem Windows-PC mit einem PC X-Server wie Hummingbird Exceed oder Vision Eclipse über den X11-Protokollstandard auf Unix-Anwendungen zuzugreifen, kann man von Linux-Clients aus Windows Anwendungen auf Windows-Applikations-Servern nutzen.

Als Windows-Applikations-Server kommen in Frage:

- Linux-Server mit bis zu 100 Windows 9x-Umgebungen (NeTraverse-Win4Lin TerminalServer) oder
- Linux-Server mit der Server-Edition von Codeweavers CrossOver Office für Mehrbenutzerbetrieb oder
- Microsoft Windows 2000-Server/Windows 2003-Terminal-Server.

Linux-Clients greifen über den X11-Protokollstandard per Display-Umlenkung

- bei der NeTraverse-Lösung auf je ein Windows 9x auf dem Linux-Server zu und
- bei Crossover Office auf die Ablaufumgebung für Windows-Programme.

Microsoft verwendet für die Verbindung der Clients zum Windows Terminal-Server sein proprietäres Remote Desktop Protokoll (RDP).

Da Microsoft leider nur Clients für seine eigenen Windows-Betriebssysteme bietet, entwickeln und pflegen

- das Projekt Opensource-rdesktop einen freien RDP-Client für Linux und
- der Anbieter HOB (<http://www.hob.de>) und ThinSoft (<http://www.thinsoftinc.com>) kommerzielle RDP-Clients für sehr viele Client-Plattformen einschließlich Linux.

Tarantella (<http://www.tarantella.com>) und Citrix (<http://www.citrix.de>) setzen hingegen auf unterschiedliche serverbasierte Lösungen: Bei Tarantella vermittelt ein Tarantella-Server, der auf einem Unix- oder Linux-Server läuft, zwischen dem RDP-Protokoll und einem eigenen Adaptive Internet Protocol (AIP) für die Kommunikation mit einer Vielzahl von Clients, u. a. Linux-Clients.

Bei Citrix ergänzt und modifiziert ein eigener Metaframe-Server auf dem Microsoft Windows-Server dessen Terminal-Server-Funktionalität und kommuniziert mit einer Vielzahl von Clients, u. a. Linux-Clients, über seine ebenfalls proprietäre Independent Computing Architecture (ICA).

11.3 Überblick

- Im kurzen Abschnitt VMWare können Sie das Konzept und die Installation von VMWare sowie Windows XP auf VMWare nachvollziehen.
- Der Abschnitt Tarantella geht auf das Konzept von Tarantella, die Installation, das Einrichten und Verwalten von Benutzern und Anwendungen und Fragen zum Drucken ein.
- Im Kapitel 10.5 konnten Sie bereits miterleben, wie man Netraverse Win4Lin Terminal-Server einrichten und nutzen kann.
- Mit Codeweavers Crossover Office hat die Redaktion dieses Buchs bisher gute erste Test-Erfahrungen gemacht, aber noch nicht lange genug damit gearbeitet, um darüber hier im Detail berichten zu können.

11.4 VMWare

11.4.1 Konzept von VMWare: Windows XP in der Linux-Box

Damit Anwender gelegentlich Windows-Anwendungen nutzen können, reicht es oft aus, eine Windows-Version in einer VMWare-Box auf einem Linux-Server zu installieren. Ist das geschehen, können Anwender von X11-Terminals oder direkt vom Linux-Server aus Windows und Windows-Anwendungen nutzen.

VMWare gibt es mittlerweile in zahlreichen Version für Workstation und Serverbetrieb. Zur Darstellung von Windows-Anwendungen auf Linux-Systemen unterscheidet VMWare zwischen der Arbeitsplatz-Versionen VMWare Workstation und der für Server-Version VMWare GSX/ESX- Server.

VMWare GSX/ESX-Server dient dazu, mehrere Windows-Server auf einem sehr großen Linux-Server zu konsolidieren. Dies spart nicht nur Administrations- und Verkabelungs-Aufwand.

Mit VMWare GSX/ESX-Server kann man viele Instanzen einer Virtuellen Maschine parallel starten und zum Beispiel gleichzeitig mit verschiedenen Versionen von Windows arbeiten. So kann einer dieser Gast-Server die Windows Terminal-Serverdienste anbieten.

Verfolgen Sie im weiteren Text jedoch das Installieren von VMWare Workstation 4 auf einem SuSE 8.2 Professional mit Windows XP als Gastbetriebssystem.

Das Installieren und Konfigurieren ist hier in drei Schritten beschrieben:

1. Zunächst gilt es, VMWare zu installieren,

2. dann dieses als Virtuelle Instanz für Windows zu konfigurieren und
3. schließlich eine Windows-Version, hier im Beispiel Windows XP, einzurichten.

Die Suse-Distribution 8.2 enthält nicht die Packages von VMWare. Man kann Sie als `.tar` oder `.rpm`-Version in der aktuellen Version von <http://www.vmware.com> als zeitlich beschränkte Evaluierungsversion fernladen.

11.4.2 VMWare installieren

Im folgenden Beispiel geht es um die Installation der VMWare Workstation 4.0 als `.tar`.

Dekomprimieren Sie zunächst die Installationsdatei und wechseln Sie dann in das Verzeichnis `vmware-distrib`.

```
boss:~ # gunzip VMware-workstation-4.0.2-5592.tar.gz
boss:~ # tar -xf VMware-workstation-4.0.2-5592.tar
boss:~ # cd vmware-distrib
boss:~/vmware-distrib # ls
. .. FILES bin doc etc installer lib man vmware-
install.pl
boss:~/vmware-distrib #
```

Durch Aufruf der Datei `vmware-install.pl` starten Sie die eigentliche, weitestgehend automatisierte Installation von VMWare. Wenn Sie andere Angaben als die in Klammern gesetzten default-Werte verwenden möchten, können Sie diese durch einfache Eingabe nach der Frage abändern.

Die Installation fragt Sie zunächst nach einigen allgemeinen Daten zu den Speicherorten.

```
boss:~/vmware-distrib # ./vmware-install.pl
Creating a new installer database using the tar3 format.
Installing the content of the package.
In which directory do you want to install the binary files?
↓ [/usr/bin]
In which directory do you want to install the library files?
↓ [/usr/lib/vmware]
The path "/usr/lib/vmware" does not exist currently.
↓ This program is going to create it, including needed parent
↓ directories. Is this what you want? [yes]
In which directory do you want to install the manual files?
↓ [/usr/share/man]
In which directory do you want to install the documentation
↓ files?
[/usr/share/doc/vmware]
```

```
The path "/usr/share/doc/vmware" does not exist currently.
↓ This program is going to create it, including needed parent
↓ directories. Is this what you want? [yes]
What is the directory that contains the init directories
↓ (rc0.d/ to rc6.d/)? [/etc/init.d]
What is the directory that contains the init scripts?
↓ [/etc/init.d]
The installation of VMware Workstation 4.0.2 build-5592
↓ for Linux completed successfully. You can decide to remove
↓ this software from your system at any
time by invoking the following command: "/usr/bin/
↓ vmware-uninstall.pl".
```

```
Before running VMware Workstation for the first time, you need
↓ to configure it by invoking the following command:
"/usr/bin/vmware-config.pl".
boss:~ #
```

Danach folgen einige Optimierungsschritte, welche die Installation ebenfalls für Sie vornehmen kann.

```
Before running VMware Workstation for the first time, you need
↓ to configure it by invoking the following command:
"/usr/bin/vmware-config.pl".
Do you want this program to invoke the command for you now?
↓ [yes].
```

Danach zeigt Ihnen die Installation die Lizenzvereinbarungen, die Sie akzeptieren müssen, damit es weitergeht. Wenn Sie die Vereinbarung gelesen haben, beenden Sie dies durch Eingabe von <Q>

```
You must read and accept the End User License Agreement to
↓ continue.
Press enter to display it.
...
Do you accept? (yes/no)
```

Als Nächstes müssen Sie einige Entscheidungen zur Netzwerkanbindung und zum Zugriff des Windows-Gastsystems auf das Linux-Dateisystem treffen.

```
Trying to find a suitable vmmon module for your running kernel.
Do you want networking for your virtual machines?
↓ (yes/no/help) [yes]
Configuring a bridged network for vmnet0.
Your computer has multiple ethernet network interfaces available
: eth0, eth1. Which one do you want to bridge to vmnet0? [eth0]
The following bridged networks have been defined:
. vmnet0 is bridged to eth0
```

```

Do you wish to configure another bridged network? (yes/no) [no]
Configuring a NAT network for vmnet8.
Do you want this program to probe for an unused private subnet?
↓ (yes/no/help) [yes]
Probing for an unused private subnet
↓ (this can take some time)...
The subnet 172.16.141.0/255.255.255.0 appears to be unused.
The version of DHCP used in this version of VMware Workstation
↓ is licensed as described in the "/usr/share/doc/vmware/
↓ DHCP-COPYRIGHT" file.
Hit enter to continue.
This system appears to have a DHCP server configured for normal
↓ use. Beware that you should teach it how not to interfere
↓ with VMware Workstation's DHCP server. There are two ways to
↓ do this:
1) Modify the file /etc/dhcpd.conf to add something like:
subnet 172.16.141.0 netmask 255.255.255.0 {
# Note: No range is given,
↓ vmnet-dhcpd will deal with this subnet.
}
2) Start your DHCP server with an explicit list of network
↓ interfaces to deal with (leaving out vmnet8). e.g.:
dhcpd eth0
Consult the dhcpd(8) and dhcpd.conf(5) manual pages for details.
Hit enter to continue.
Do you want to be able to use host-only networking in your
↓ virtual machines?
[no]
Do you want this program to automatically configure your system
↓ to allow your virtual machines to access the host's
↓ filesystem? (yes/no/help) yes
The version of Samba used in this version of VMware Workstation
↓ is licensed as described in the "/usr/share/doc/vmware/
↓ SAMBA-LICENSE" file.
Hit enter to continue.
Creating a host-only network on vmnet1.
↓ (this is required to share the host's filesystem).
Configuring a host-only network for vmnet1.
Do you want this program to probe for an unused private subnet?
↓ (yes/no/help) [yes]
Probing for an unused private subnet
↓ (this can take some time)...
The subnet 172.16.113.0/255.255.255.0 appears to be unused.
This system appears to have a DHCP server configured for normal
↓ use. Beware that you should teach it how not to interfere
↓ with VMware Workstation's DHCP server. There are two ways to
↓ do this:

```

```

1) Modify the file /etc/dhcpd.conf to add something like:
subnet 172.16.113.0 netmask 255.255.255.0 {
# Note: No range is given, vmnet-dhcpd will
# deal with this subnet.
}
2) Start your DHCP server with an explicit list of network
↓ interfaces to deal with (leaving out vmnet1). e.g.:
dhcpd eth0
Consult the dhcpd(8) and dhcpd.conf(5) manual pages for details.
Hit enter to continue.
This system appears to have a CIFS/
SMB server (Samba) configured for normal use. If this server
↓ is intended to run, you need to make sure that it will not
↓ conflict with the Samba server setup on the private network
↓ (the one that we use to share the host's filesystem).
↓ Please check your /etc/samba/smb.conf file so that:
. The "interfaces" line does not contain "172.16.113.1/
255.255.255.0"
. There is a "socket address" line that contains only your real
↓ host IP address
Hit enter to continue.
Starting VMware services:
  Virtual machine monitor           [ OK ]
  Virtual ethernet                  [ OK ]
  Bridged networking on /dev/vmnet0  [ OK ]
  Host-only networking on /dev/vmnet1 (background) [ OK ]
  Host-only networking on /dev/vmnet8 (background) [ OK ]
  NAT networking on /dev/vmnet8     [ OK ]
You have successfully configured VMware Workstation to allow
↓ your virtual
machines to access the host's filesystem. Would you like to add
↓ a username and password for accessing your host's filesystem
↓ via network interface vmnet1 at this time?
↓ (yes/no/help) [yes]
Please specify a username that is known to your host: tux
New SMB password:
Retype new SMB password:
root: "/usr/bin/vmware-smbpasswd vmnet1 -a <username>"
The configuration of VMware Workstation 4.0.2 build-5592
↓ for Linux for this running kernel completed successfully.
You can now run VMware Workstation by invoking the following
↓ command:
"/usr/bin/vmware".
Enjoy,
--the VMware team
boss:~/vmware-distrib #

```

VMWare ist nun auf dem Linux-System eingerichtet, und Sie können beginnen, eine Virtuelle Maschine (VM) für das Gastbetriebssystem einrichten zu lassen.

11.4.3 VM konfigurieren

Virtuelle Instanzen richtet man über `/usr/bin/vmware` in einem grafischen Installationsmenü ein.

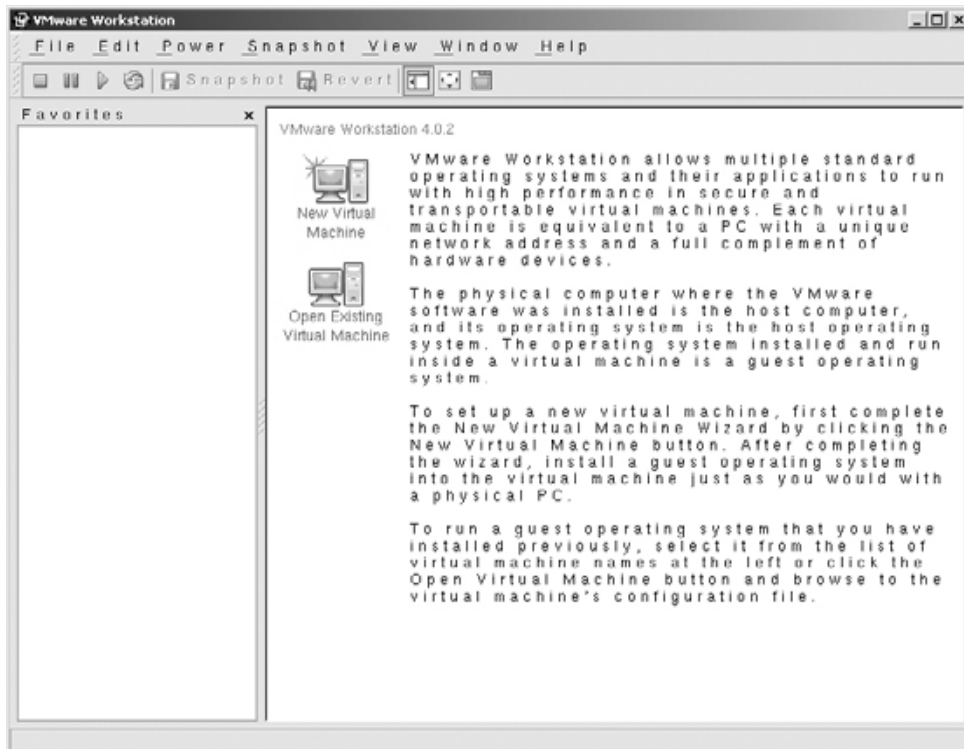


Abbildung 11.1: VMWare Konfigurationsoberfläche

Der Installations-Wizard von VMWare interessiert sich zunächst für den Lizenzschlüssel. Einen 30-Tage-Evaluierungsschlüssel können Sie unter der Adresse <http://www.vmware.de> anfordern und in der Konfigurationsoberfläche unter dem Menüpunkt *Help* • *Serial Number* eintragen.

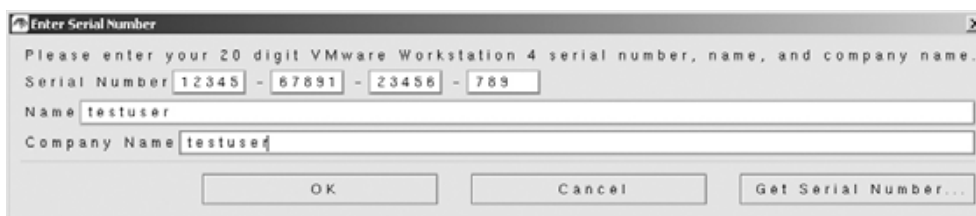


Abbildung 11.2: VMWare Lizenzschlüssel

Drücken der Taste *New virtual Machine* ruft den *Virtual Machine Installation-wizard* auf. Wählen Sie hier bitte *Typical*. Vom Standard abweichende Komponenten können Sie auch im Nachhinein konfigurieren. Wählen Sie anschließend das von Ihnen gewünschte Betriebssystem.

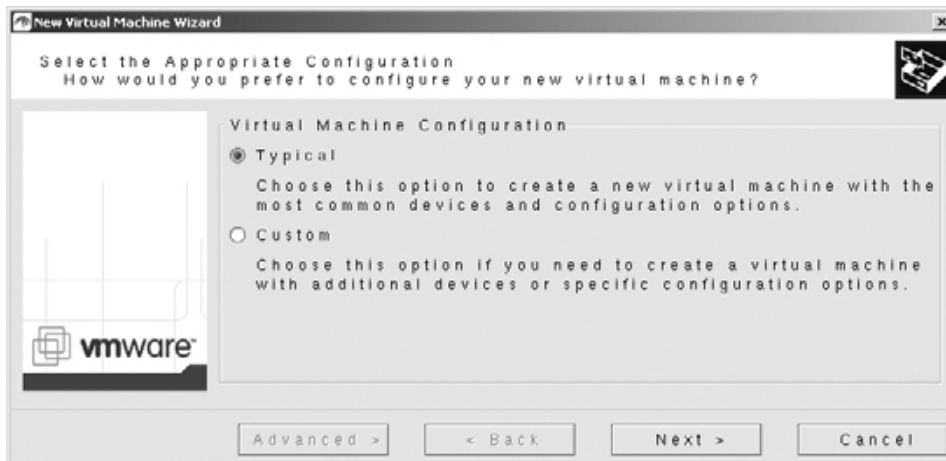


Abbildung 11.3: VMWare Configuration wizard

Nun folgen die Angaben für den Speicherort und die Speichergröße. VMWare nimmt hierfür einen Teil der Festplatte und reserviert ihn für VMWare. Diesen reservierten Bereich nennt VMWare *virtual Disk*. Alternativ können Sie eine eigene Festplatte für die Windows-Installation angeben.

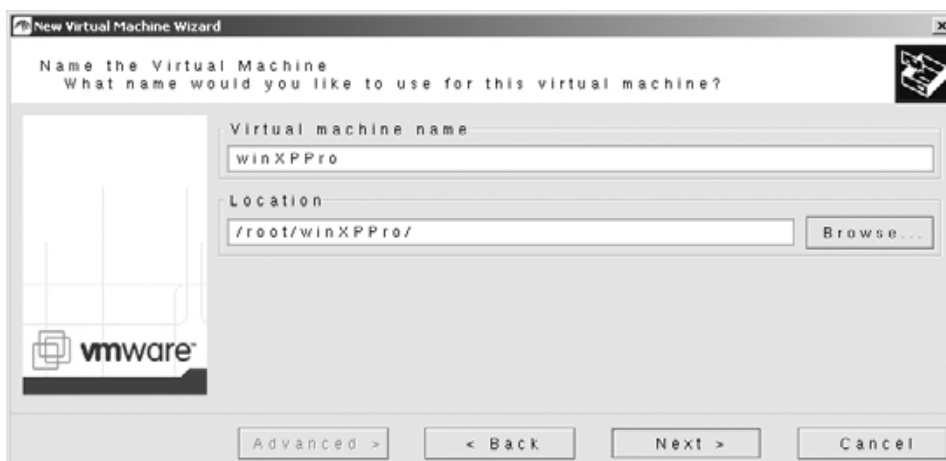


Abbildung 11.4: Name der VM-Instanz und Installationsort

Einer der wesentlichen Punkte für die spätere Arbeit mit Windows ist die Anbindung an das Netz. VMWare bietet Ihnen verschiedene Möglichkeiten zur Anbindung des Gastbetriebssystems an ein bestehendes Netzwerk an. Der Modus des *Bridged Networking* konfiguriert eine komplett eigene Netzwerk-Instanz für das Gastbetriebssystem:

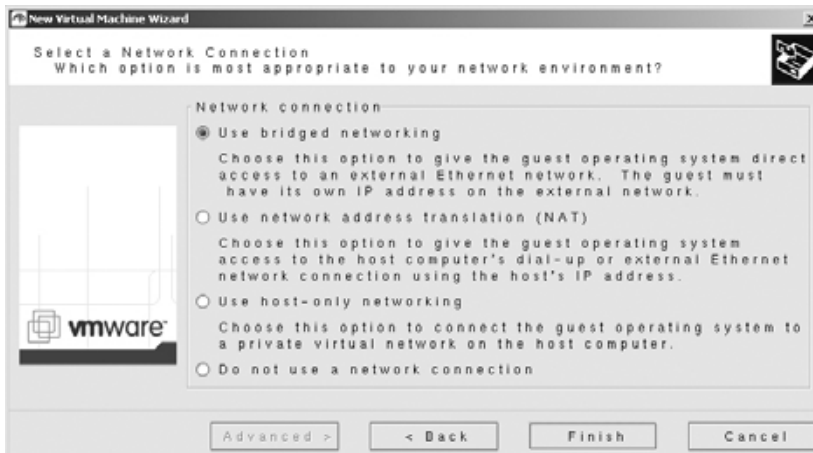


Abbildung 11.5: Netzwerkeinstellungen

Zum Abschluss der Installation der VM-Instanz zeigt diese eine Zusammenfassung an.



Abbildung 11.6: Zusammenfassung der Konfigurationsdaten

Über den Menüpunkt *Edit • Virtuelle Machine Setting* können Sie Ihre bisherigen Einstellungen jederzeit verändern, hier im Beispiel den zur Verfügung gestellten virtuellen Arbeitsspeicher.

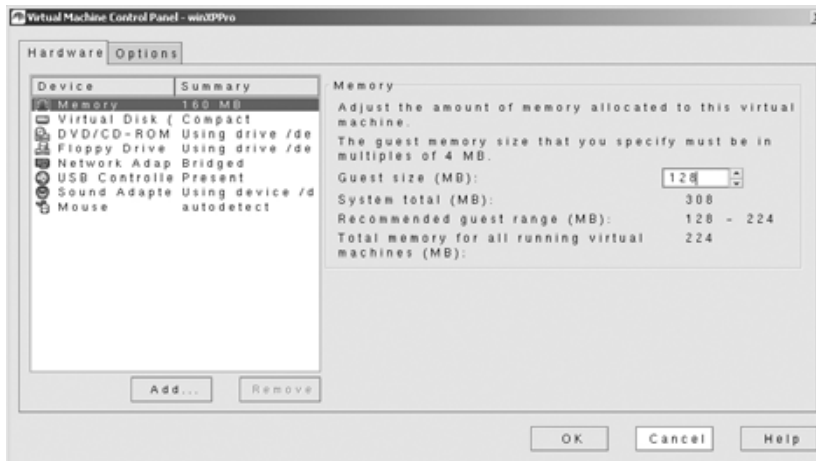


Abbildung 11.7: Einstellungen der virtuellen Speichers

11.4.4 Windows XP Professional auf VMWare installieren

Nach dieser rasant schnellen Vorbereitung kann man Windows XP von CD installieren. Hierzu starten Sie die VM durch den angezeigten Menüpunkt.

Ab diesem Zeitpunkt läuft die Installation von Windows XP wie auf einer physischen Festplatte.



Abbildung 11.8: Windows Setup Begrüßung

Das Setup von Windows stellt eine neue Festplatte fest (Abbildung 11.9). Lassen Sie das Setup-Programm die von ihm erkannte Festplatte einrichten und mit dem Format *NTFS* formatieren.

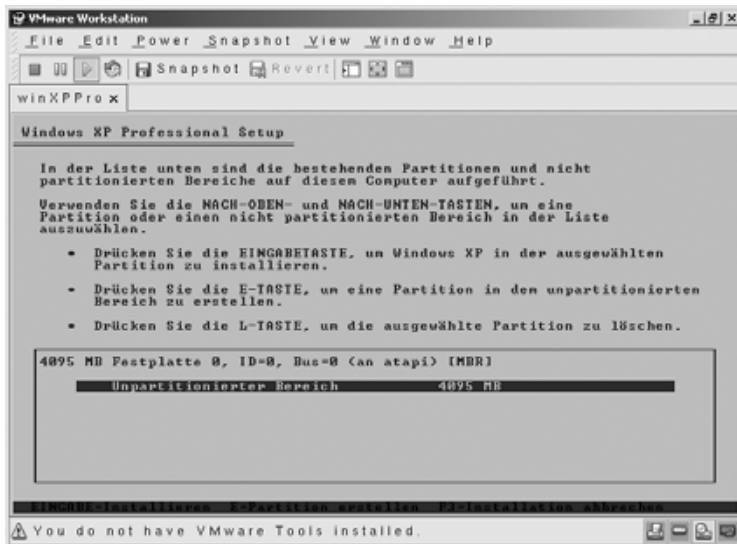


Abbildung 11.9: Festplattensetup

Das Setup von Windows XP kopiert danach die benötigten Dateien in die Installations-Verzeichnisse und initialisiert Windows.

Danach startet Windows XP. Es setzt das Setup im Grafikmodus fort und sucht und installiert Geräte.



Abbildung 11.10: Grafische Windows-Installation

Als Nächstes erfasst das Setup Angaben zum Gebietschema wie Ländereinstellungen, Zeitzone, Sprache und Tastaturlayout.



Abbildung 11.11: Gebietschema wählen

Geben Sie bei den Benutzerinformationen Ihren Namen und den Ihrer Organisation an.



Abbildung 11.12: Benutzerinformationen eingeben

Für Benutzer von Open Source-Software mag es ungewohnt sein, im nächsten Bild eine Seriennummer anzugeben:



Abbildung 11.13: Seriennummer eingeben

Dann müssen Sie dem virtuellen Server einen eigenen Namen geben und den Namen und das Passwort des Systemverwalters eingeben.

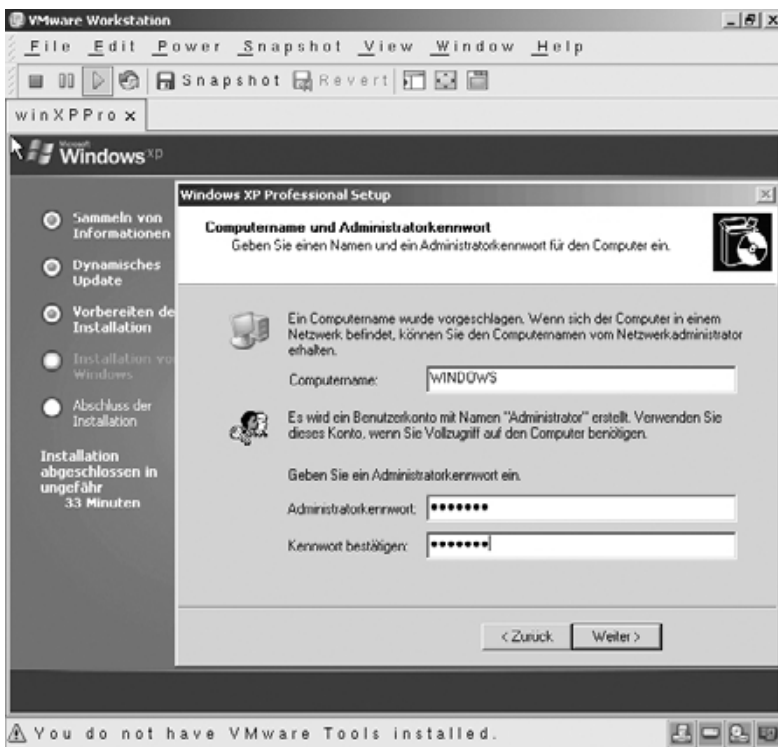


Abbildung 11.14: Namen des Servers und Admin-Kennwort eingeben

Windows XP installiert die virtuelle Netzwerkkarte, die es von VMWare bekommt (diese Netzwerkkarte ist völlig unabhängig von der realen Netzwerkkarte im Linux-Server), und richtet das Netzwerk ein.

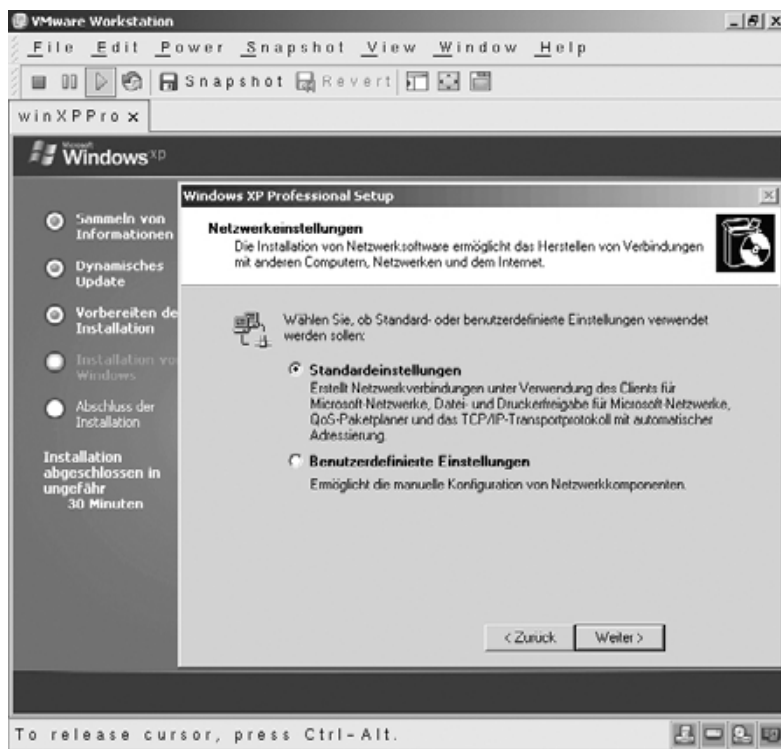


Abbildung 11.15: Microsoft-Netzwerk einrichten

Geben Sie dann an, in welche Arbeitsgruppe oder Domäne Sie den virtuellen Windows XP-PC einfügen wollen, und geben Sie den Namen und das Passwort des Nutzers an, der den PC in die Windows-Domäne einfügen darf.

Die Installation läuft ab jetzt automatisiert.

Sobald die Installation fertig ist, können Sie Windows XP in der Box von VMWare vom Linux-Server und von jedem X11-Terminal aus wie eine eigenständige Installation nutzen und sich an populären Windows-Anwendungen wie Microsoft Office erfreuen. Starten Sie zunächst über `/usr/bin/vmware` VMWare und wählen Sie die Konfigurationsdatei, die Sie zuvor durch VMWare durch die Windows-Installation angelegt wurde.

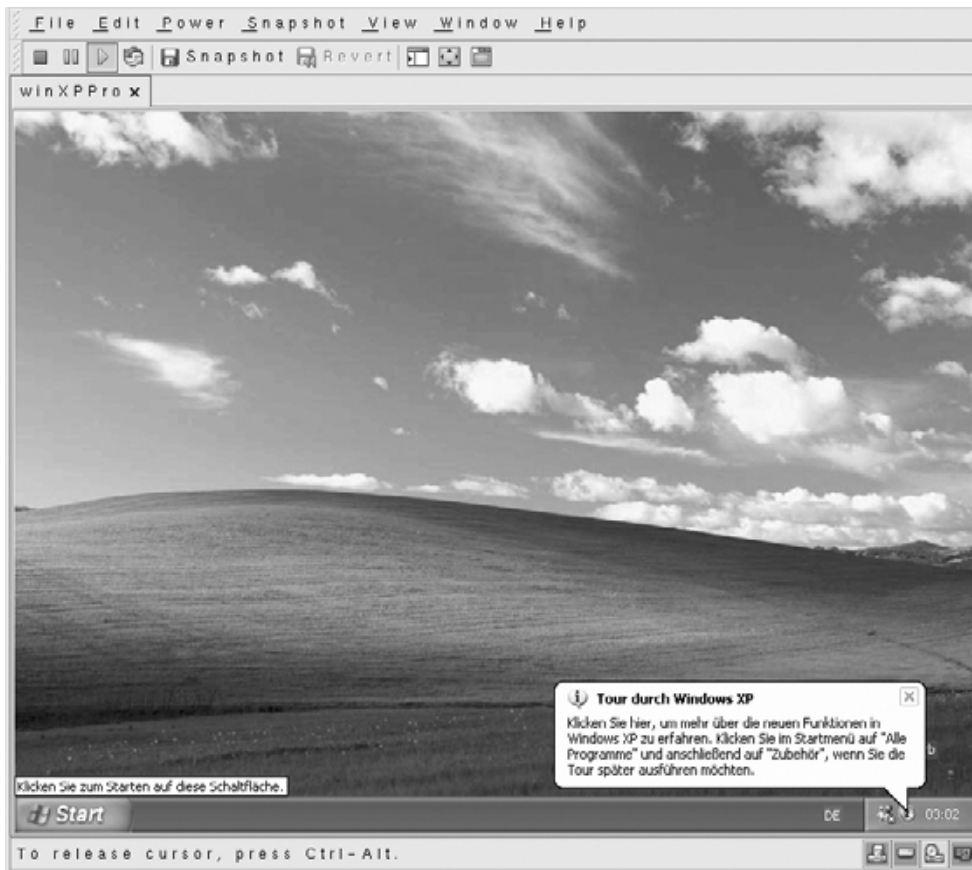


Abbildung 11.16: Windows XP im VMWare-Fenster

Sie können auch zusätzliche Versionen von Windows, z. B. andere Sprachen bzw. andere Windows-Versionen, installieren. Hierzu müssten Sie wiederum die beschriebenen Installationsschritte für das neue, gewünschte Betriebssystem durchlaufen. Wenn Sie mehrere Windows-Sitzungen benötigen, können Sie auch mehrere Instanzen von Windows ausführen lassen. Dies benötigt jedoch sehr großzügig bemessene Hardware-Ressourcen.

11.5 Konzept von Tarantella

Die Firma Tarantella Inc., zuvor SCO Inc., bietet seit Ende 1997 Versionen von Tarantella: Seit dem Verkauf von zwei der drei Sparten der SCO an Caldera hat sich SCOs verbliebene innovativste Sparte, Visionprodukte und Tarantella, Ende 2000 in Tarantella umbenannt und sich auf die ebenso genannten Produkte fokussiert.

Mit dem Zukauf der *New Moon Systems Inc.* im Juni 2003 erweitert Tarantella sein Produktportfolio kontinuierlich im Linux-/Unixbereich als auch auf den Serverbereich der Microsoft Plattformen.

Die Middleware Tarantella Enterprise verbindet nahezu beliebige Clients mit nahezu beliebigen Anwendungs-Servern. Statt Anwendungen lokal auf den Arbeitsplatzrechnern der Anwender zu installieren, stellt man sie diesen über ein LAN oder über ein WAN als Serverdienst zur Verfügung, um Administrations-, Sicherheits- und Supportvorteile zu erzielen und die Gesamtkosten der Datenverarbeitung zu senken.

Tarantella-Server arbeiten in drei Schritten:

- Sie verhalten sich gegenüber Mainframe (TN3270- und TN5250-Protokolle), Unix (X11, VT und ANSI-Protokolle) und Windows-Terminalservern (RDP 4, 5.0, 5.2) wie »native« Clients und fangen deren Benutzersitzungen ab,
- komprimieren und verschlüsseln die Sitzungen in ihr eigenes Tarantella-Protokoll (Adaptive Internet Protocol, AIP) und
- übertragen sie damit an Java-fähige Browser oder spezielle Web-Appliances (Native Clients).

Dies ist weiter unten detaillierter beschrieben.

11.5.1 Zielgruppen für Tarantella

Mit Tarantella kann man Applikationen im Intranet und Internet für gelegentlichen oder häufigen Gebrauch bereitstellen:

- *Serverzentriertes Computing*: Da viele Unternehmen und öffentliche Einrichtungen die ausufernden Kosten der Betreuung von Windows-Arbeitsplätzen nicht mehr tragen wollen, stellen sie vermehrt auf serverzentrierte Datenverarbeitung um. Neue Anwendungen, Patches, Hot Fixes und Service-Packs brauchen sie dann nur noch zentral im Rechenzentrum zu installieren, laufen zu lassen und zu pflegen. Sie verwenden Tarantella als Middleware zwischen Windows-Anwendungs-Servern und Arbeitsplatzgeräten, wenn diese über ein WAN zugreifen oder sie die Last der zahlreichen Arbeitsplatzgeräte auf mehrere Windows-Terminalserver verteilen wollen.
- *PCX-Server oder RDP-Zugriff*: Immer mehr Unternehmen und öffentliche Einrichtungen verwenden aus Sicherheits- und Kostenüberlegungen möglichst Open Source-Programme. Um ausnahmsweise von Java-fähigen Browsern ihrer Linux-Clients auf Windows-Programme zugreifen zu können, ohne auf Clients virenanfällige und instabile Microsoft Produkte zu installieren, stellen Sie mit Tarantella und Windows-Terminalservern die in der Übergangszeit noch gewünschten Win-

dows-Programme bereit. Tarantella ermöglicht Anwendern, von den verschiedensten Clients aus (z. B. Windows CE) auf den verschiedensten Applikations-Servern (z. B. »Gnome« auf Solaris) zu arbeiten. Andererseits ist es auch möglich Unix-/Linuxapplikationen via AIP auf Windows-Clients ausführen zu lassen. So können bestehende Windows-Clients weiter verwendet werden und Anwender bereits auf X-basierende Applikations-Server zugreifen.

- *Sicherheit:* Anwender greifen immer und ausschließlich über den Tarantella-Server auf Anwendungen und Daten zu. Dies ermöglicht es, ein sehr sicheres und sehr einfach zu kontrollierendes Netz aufzubauen. Das Öffnen zusätzlicher Ports zum Zugriff in Rechenzentren entfällt. Jede Kommunikation zwischen Client und Tarantella-Server kann zusätzlich verschlüsselt und protokolliert sowie bei Bedarf aufgezeichnet werden.
- *Webenabler:* Viele Firmen entwickeln nur noch Applikationen, welche in Browsern lauffähig sind (z. B. SAP). Die Bedeutung von Browsern als der universellen zentralen Anwendung wächst. Mit Tarantella kann man verschiedenste Anwendungen innerhalb weniger Minuten »browserfähig« machen und so z. B. X11-Anwendungen über Portale wie iPlanet Usern einheitlich anbieten.

11.5.2 Funktion von Tarantella

Tarantella arbeitet als Middleware zwischen Anwendungs-Servern und Clients.

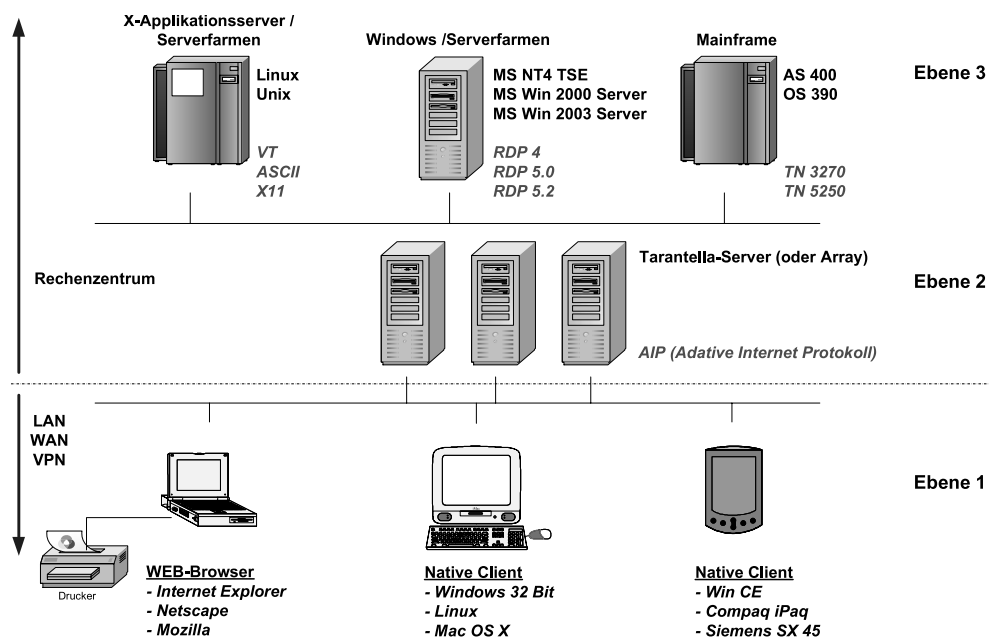


Abbildung 11.17: Die 3 Ebenen aus Tarantella-Sicht

In der Client-Ebene, der Ebene 1, arbeiten die Anwender und in der Ebene 3 die Applikations-Server, typischerweise Windows-Terminalserver, Unix-/Linux- und Mainframesysteme, Plattform-unabhängige Java-Applikations-Server sowie File-, Proxy- und Print-Server (welche manchmal auch als Ebene 4 bezeichnet wird). Zwischen der 1. und der 3. Ebene, in der Ebene 2, fangen Tarantella-Server die Protokolle der 3. Ebene auf, übersetzen diese in ihr eigenes Protokoll, das Adaptive Internet Protokoll (AIP), und stellen dessen Inhalt den Clients zur Verfügung. Tarantella muss nicht dezidiert auf einem eigenen Server laufen, sondern kann auf vorhandenen Servern mitlaufen, zum Beispiel auf einem Samba-Linux-(SMB-) Server.

Tarantella kann Unix-/Linux/Mainframe-Programme über X11-Protokolle oder Terminal-Emulationen auffangen und in sein eigenes Protokoll AIP umsetzen.

Benutzersitzungen von Windows-Programmen im Terminal-Servermodus kann es via RDP abfangen. Windows-Terminalserver kommunizieren mit Clients mit Microsofts proprietärem Remote Desktop Protokoll (RDP). Microsoft bietet jedoch selbst ausschließlich Windows-basierende Clients an. Aufgrund einer Lizenzvereinbarung zwischen Microsoft und Tarantella kennt Tarantella Microsofts Spezifikationen und Code zum Zugriff auf die Terminal-Server-Sitzungen. Der Unix-/Linux-basierende Tarantella-Server verhält sich gegenüber den Terminal-Servern dann ebenfalls wie ein Microsoft Terminal-Server-Client. Obendrein bietet Tarantella zusätzliche Funktionen, die über einen nativen RDP-Client-Zugriff nicht möglich wären. So hat Microsoft über RDP zwar einen Laufwerkszugriff zwischen Terminal-Server und Client zwar von vornherein vorgesehen, jedoch nicht in Windows 2000 integriert (die Schaltfläche hierfür kann nicht aktiviert werden). Tarantella ermöglicht dieses Feature jedoch über einen virtuellen SMB-Server, den die Tarantella-Installation automatisch einrichtet.

Die Terminaldienste von Windows 2000-Servern bieten jedoch nur etwas eingeschränkte Arbeitsmöglichkeiten, da dessen RDP 5.0 nur 256 Farben und keinen Sound übertragen kann.

Diese Einschränkungen gelten zwangsläufig auch für Benutzer, die sich über Tarantella-Server konnektieren. Seit April 2003 hebt die Nachfolgeversion Windows 2003 (Microsoft Windows .NET Serverfamilie) einige diese Begrenzungen auf. Daher empfiehlt es sich, für Windows-Terminalserver auf der Plattform Windows 2003 aufzusetzen.

Tarantella kann mehrere Terminal-Server zu *Serverfarmen* zusammenfassen, um deren Last auszugleichen (Load Balancing via CPU, Speicherauslastung etc.). Dazu sollten die Terminal-Server gleich aufgebaut sein und Anwendern idealerweise über einen gemeinsamen Domain Controller oder über einen Active Directory-Server die gleichen User-Accounts anbieten.

Tarantella-Server können ihr eigenes Protokoll AIP (Adaptive Internet Protokoll) mit 128 Bit SSL verschlüsseln.

Auf Firewalls braucht man für Tarantella keine zusätzlichen Ports zu öffnen, da man es für einen »Single Port Access« über einen Standardport wie Port 80 (http) oder Port 443 (https) konfigurieren kann (siehe <http://www.tarantella.com/whitepapers/firewall/>). Auch Webproxy-Server mit Protokoll Tunneling stellen keine Hindernisse für Tarantella dar. AIP benötigt, falls erforderlich, weniger Bandbreite als Microsofts RDP: Tarantella-Server entscheiden dynamisch nach der dem Client zur Verfügung stehenden Bandbreite, wie weit sie den Datenstrom verdichten.

Während Tarantella-Server in einem 10 Mbit-LAN praktisch nichts komprimieren, verdichten sie den Datenstrom bei langsamen WAN-Verbindungen oder wenn sich mehrere Anwender eine Leitung teilen. GSM-Verbindungen mit 14.400 Baud reichen für einen Benutzer gerade noch aus, 64 Kbit-ISDN-Leitungen bedienen bequem zwei User.

Die Benutzer der Tarantella-Server kann man auf unterschiedlichste Art verwalten: Mit Unix-Accounts auf den Tarantella-Servern (z. B. NIS), Authentisierung gegenüber LDAP- oder Active Directory-Servern, Windows-Domänen, Web-Servern und/oder Secure ID-Servern oder mit einer eigenen Benutzerverwaltung in Tarantella selbst.

Das Session-Resume unterstützt mobile Nutzer: Benutzer können nicht ordnungsgemäß beendete Verbindungen später wieder aufnehmen. Damit sind Tunnel wie auf der ICE-Strecke von Frankfurt nach Göttingen oder Staus im Elbtunnel für mobile GSM-Nutzer nicht mehr ganz so schrecklich.

Auch in mobilen und angeblich papierlosen Büros wollen Anwender drucken. Tarantella-Server kann man als IP-Druck-Server ansprechen und so Druckaufträge ebenfalls über AIP verschlüsseln und komprimiert an Tarantella-Clients schicken. Drucken Anwender, kann man mit Tarantella (oder bei Bedarf auch mit Zusatzprodukten wie ThinPrint) Druckaufträge packen und die von Druckaufträgen beanspruchte Bandbreite mit ThinPrint (<http://www.thinprint.com>) begrenzen.

Da Tarantella automatisiert Druckaufträge in PDF umwandeln und an den Client senden kann, kann man endlich durch die hohe Komprimierungsrate von PDF-Dokumenten auch in WAN-Umgebungen noch vernünftig drucken.

Tarantella selbst nimmt ungefähr 150 MB Platz auf den Festplatten des Linux-Servers in Anspruch. In kleinen Installationen reichen auch ältere Server.

Richtwerte für den Bedarf an Hauptspeicher und Prozessoren für Linux-Tarantella-Server finden Sie in <http://www.tarantella.com/products/e3/requirements.html>. Da Tarantella nicht sehr ressourcenhungrig ist, kann man mit einem Single Pentium III, 1 GHz und 1 GB Hauptspeicher ohne Schwierigkeiten 100 User gleichzeitig anbinden. Bei üppiger Hardwareausstattung sind weit über 1000 User pro Server möglich. Tarantella nennt unter <http://www.tarantella.com/whitepapers/> die Testergebnisse verschiedener Betriebssysteme und Hardwarelieferanten.

Als Client kann man praktisch jedes Gerät mit Java-fähigem Browser verwenden. Für Clients reicht ein langsames Pentium-System ab Pentium 200 MHz und 64 MB Arbeitsspeicher. Beim erstmaligen Verbinden zum Tarantella-Server lädt der Browser 600 KB Javaprogramme und installiert sie. Diese Javaprogramme sollte der Client aus Performancegründen auf der Festplatte speichern können. Ältere Hardware (ab etwa Pentium I mit 32 MB Arbeitsspeicher) kann man mit einem Tarantella-Native-Client verwenden. Der Tarantella-Native-Client ist für verschiedenste Betriebssysteme erhältlich wie: Windows 32 Bit, Windows CE, Linux Kernel 2.2+, Solaris 7+. Eine vollständige Liste der unterstützten Browser und Client-Betriebssysteme hat Tarantella unter der Adresse

http://www.tarantella.com/knowhow/e3.3/help/en-us/base/gettingstarted/clients_supported.html

zusammengestellt.

11.6 Tarantella installieren

Tarantella selbst läuft derzeit ausschließlich auf einem Linux- oder Unix-Betriebssystem. Als Standardinstallation empfiehlt sich eine »normale« Linux-Installation mit installiertem X-Grundsystem, bei Bedarf mit Web-Server-Installation.

Tarantella kann zwar jeden beliebigen Web-Server verwenden, bringt jedoch ebenfalls einen eigenen für Tarantella optimierten Web-Server auf der Basis von Apache 1.3.27 und Tomcat mit. Vor allem in sicherheitskritischen Bereichen erleichtert diese, auf Sicherheit optimierte Version die Konfiguration. Wollen Sie lieber einen bereits existierenden Web-Server nutzen, sollten Sie Tipps für Anpassungen an der folgenden Adresse lesen: http://www.tarantella.com/knowhow/e3.3/help/en-us/base/gettingstarted/websrv_tta_config.html.

Bei der Installation von Tarantella ist das Netzwerk selbst eine Hauptfehlerquelle. Prüfen Sie bitte das Netzwerk einschließlich Namensauflösung (ping, host, nslookup/dig etc.) vor der Installation von Tarantella. Eine nachträgliche Änderung des primären Hostnamens ist sehr zeitaufwändig.

Die folgenden Beispiele basieren auf einer SuSE 8.2 Professional, Installationsmodul *KDE Standard* und zusätzlich *Einfacher Webserver*.

Eine 30-Tage-Demoversion von Tarantella kann man aus dem Internet von <http://www.tarantella.com/download> fernladen. Diese Version ist bis zum zeitlichen Enddatum uneingeschränkt benutzbar und bei Bedarf von Tarantella einmalig um weitere 30 Tage verlängerbar. Nach Ablauf der Evaluierungszeit bedarf es der Eintragung eines Lizenzkeys. Die Lizenzierung von Tarantella ist hierbei sehr modular aufgebaut. Aktuelle Preisinformationen können Sie online bei autorisierten Tarantella-Distributoren beziehen (<http://www.tarantella.com/distributors/>).

Die Tarantella-Dateien gehorchen folgender Schreibweise:

Die ersten drei Buchstaben geben das Programm selbst an: `tta` steht für das Tarantella-Basis-Modul, die nächsten zwei Buchstaben geben den Prozessortyp an, z. B. `i3` für Pentiumsysteme, und die letzten Buchstaben das Betriebssystem z. B. `li` für Linux:

`ttai3li` bedeutet Tarantella Base Pack für Pentiumsysteme mit dem Betriebssystem Linux.

Um die Installation mitzumachen, loggen Sie sich als `root` in Ihr Linux-System ein.

Einige Prozesse auf dem Tarantella-Server laufen unter dem Benutzer »`ttaserv`« als Mitglied der Gruppe »`ttaserv`«. Diesen Benutzer sowie diese Gruppe müssen Sie vor der Installation auf dem Tarantella-Server wie folgt einrichten:

```
boss:~ # groupadd ttaserv
boss:~ # useradd -g ttaserv -s /bin/sh ttaserv
```

Da Tarantella den Benutzer ausschließlich für systeminterne Zwecke verwendet, sollten Sie diesen Benutzer-Account für jeglichen Zugriff sperren:

```
boss:~ # passwd -l ttaserv
Locking password for user ttaserv.
passwd: Success
boss:~ #
```

Mounten Sie nun das CD-ROM-Laufwerk und rufen Sie dann die richtige Datei mit `sh` auf.

Die Installation prüft, ob Sie das richtige Paket verwenden, genügend Festplattenspeicher zur Verfügung steht und ob der Benutzer und die Gruppe `ttaserv` existieren. Sollten während der Installation Fehler auftreten, trägt die Installation die Fehlermeldungen in die Datei `/tmp/ttainst.log` ein:

```
boss:~ # mount /cdrom
boss:~ # cd /cdrom
boss:~ # sh ttai3li.shx
-----
Tarantella Enterprise 3
-----
Setup is verifying the shx file...
Setup is checking for disk space requirements...
Setup is installing package programs in /usr/package...
Tarantella Enterprise 3 for Intel Linux kernel 2.4+
Preparing for installation...
-----
Tarantella Software License Agreement
-----
```

```
To use Tarantella you must agree to be bound by
the terms of the Software License Agreement.
```

```
Y - I have read, and accept the terms of the license agreement
N - I do not accept the terms of the license agreement
R - Let me read the license agreement
```

```
Accept terms of Software License Agreement? [R] y
```

1. Bestätigen Sie wie hier im Beispiel die Lizenzvereinbarungen und geben Sie den Installationsort, den Archivierungszeitraum für Fehlermeldungen und den Lizenzschlüssel (Standard: *Evaluierung*) ein. Wenn wie hier im Beispiel bereits ein Web-Server auf Port 80 läuft, fragt Tarantella, auf welchem Port der Tarantella-eigene Web-Server laufen soll. Wenn Sie Ihren bereits installierten Web-Server nicht deaktivieren möchten, so müssen Sie einen eigenen Zugriffs-Port für Tarantella definieren, zum Beispiel Port 8080. Sollte Tarantella feststellen, dass der Port nicht belegt ist, würde es diesen automatisch für sich reservieren.

```
-----
Setting up Tarantella Enterprise 3
-----
```

```
Tarantella Setup recommends you use the following settings:
```

```
Installation type = install 3.40.827
Installation directory = /opt/tarantella
Peer DNS name = boss.lokales-netz.de
License mode = Evaluation (30-day limit)
HTTP port = 80 [already in use]
Archive logs every week? = yes (Sunday 03:00 hours)
```

```
IMPORTANT: The default HTTP port is in use. Either stop the
└ process
that is using this port, or select a different port when
└ prompted.
```

```
-----
HTTP port
-----
```

```
Tarantella includes a web server. You need to choose the TCP
└ port on which the web server listens for HTTP (unencrypted)
└ connections.
The default port for HTTP is 80/tcp.
```

```
You must choose an unused TCP port for installation to complete.
Port 80/tcp is currently in use.
```

```
HTTP port [80] 8080
```

2. Nach Bestätigung mit `y` beginnt die eigentliche Installation, die je nach Hardware etwa 10 Minuten dauert:

```
Tarantella Setup is recording settings...
Tarantella Setup is installing files...
...
Starting Tarantella server, please wait.....
The Tarantella server is now running.
-----
-----
What's next?
-----
Tarantella is now installed and ready to use.

To get started:
- In a web browser, go to:
  http://boss.lokale-netz.de:8080/tarantella
- When prompted, log in with username "root" and root's
  ↵ password.
- On your webtop, click Admin Guide to learn more about the
  ↵ product (we strongly recommend you read the
  ↵ "Before you begin" section).
  Or click Object Manager to start creating user webtops.

To add license keys, click Array Manager on your webtop or type:
/opt/tarantella/bin/tarantella license add <key>...

Note: Setup has created some application objects.
Setup detected that telnet is not enabled on your system, but
↵ SSH is.
Consequently the application objects have been configured to use
↵ SSH.
However, you might need to configure SSH to work with Tarantella
↵ before you can use these applications through Tarantella.

For more information on Tarantella and SSH, see the
↵ Administration Guide.
boss:~ #
```

Damit ist die Tarantella-Installation abgeschlossen.

Beachten Sie bitte ebenfalls <http://www.tarantella.com/support/updates> für Tarantella-Updates und dort den Unterpunkt Knowhow für besondere Einstellungen und Tuningmöglichkeiten, abhängig vom verwendeten Betriebssystem.

11.6.1 Konfigurieren des vorhandenen Web-Servers für Tarantella

Wenn Sie Ihren bestehenden Web-Server anstatt des von Tarantella installierten Web-Servers verwenden möchten, müssen Sie diesen für eine Zusammenarbeit mit Tarantella konfigurieren.

Zum Glück muss man am installierten Web-Server je nach Typ, Betriebssystem und Tarantella-Version nur einige Kleinigkeiten ändern.

Hier folgen Konfigurationsschritte für den Apache Web-Server 1.3.x. Diese basieren auf der im Kapitel 6 beschriebenen Konfiguration des Web-Servers.

Tragen Sie bitte zusätzlich folgende Zeilen in `/etc/httpd/httpd.conf` ein:

- Alias für Tarantella definieren:

```
#
# Aliases: Add here as many aliases as you need (with no limit).
# Format is
# Alias fakename realname
#
# Eintrag für Tarantella Alias (Verweist ../tarantella Anfragen)
Alias /tarantella /opt/tarantella/var/docroot
#
```

- SkriptAlias für Tarantella definieren:

```
#
# ScriptAlias: This controls which directories contain server
# scripts.
# ScriptAliases are essentially the same as Aliases, except
# that documents in the realname directory are treated as
# applications and run by the server when requested rather
# than as documents sent to the client.
# The same rules about trailing "/" apply
# to ScriptAlias directives as to Alias.
#
ScriptAlias /cgi-bin/ "/usr/local/httpd/cgi-bin/"
#
# Eintrag für Tarantella ScriptAlias (Verweist cgi-bin-Anfragen)
ScriptAlias /tarantella/cgi-bin "/opt/tarantella/var/docroot/
└─ cgi-bin"
#
```

Nach diesen Änderungen sollten Sie den Web-Server (`/etc/init.d/apache restart`) und Tarantella neu starten. Tarantella startet man im Programm Pfad (default: `/opt/tarantella/bin`) mit `tarantella restart`.

Wenn Anwender angemeldet sind (z. B. bei Updates), können diese während des Restarts von Tarantella nicht weiterarbeiten. Sobald der Server jedoch wieder zur Verfügung steht (nach etwa 15 Sekunden), können die Anwender an dem Punkt weiterarbeiten, an dem sie zuvor aufhören mussten. Möchten Sie alle bestehenden Sessions neu aufsetzen, so verwenden Sie bitte `tarantella restart --kill`.

Achten Sie hierbei auf eventuelle Fehlermeldungen:

```
boss:~ # cd /opt/tarantella/bin/
boss:/opt/tarantella/bin # ./tarantella restart
Tarantella services have been stopped.
Starting Tarantella server. Please wait...
Tarantella services are now available on this host.
boss:/opt/tarantella/bin #
```

11.6.2 Erste Verbindung

In den letzten Abschnitten konnten Sie zwei verschiedene Web-Server-Lösungen für den Zugriff auf Tarantella verfolgen. Bei diesen unterscheiden sich auch die Adressen für die Zugriffe:

1. beim Webserver, den das Betriebssystem installiert, reicht ein Aufruf ohne Portangabe: also z. B. `http://boss.lokales-netz.de` und
2. beim Web-Server, den Tarantella einrichtet, müssen Anwender den Port angeben, also z. B. `http://boss.lokales-netz.de:8080`

Tarantella bietet Clients die zwei Zugriffsarten per Browser oder mittels Tarantella Native Client.

Tarantella Native Client

Den Tarantella Native Client für Windows 32 Bit und für Linux können Anwender von Ihrem Tarantella-Server z. B. von `http://boss.lokales-netz.de:8080/tarantella/cgi-bin/install.cgi` laden und für weitere Plattformen direkt von Tarantella (www.tarantella.com/download).

Tarantella hat das Installieren der Clients für Windows als `.exe`-Datei und für Linux als komprimiertes Shellscript `.sh` automatisiert.

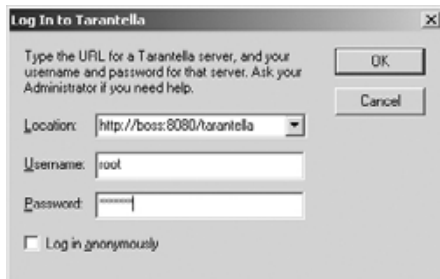


Abbildung 11.18: Anmeldedialog »Native Client«

Anwender können sich in der Voreinstellung mit ihren Linux-User-Accounts und den Passwörtern des Linux-Servers in Tarantella einloggen. Der Systemverwalter *root* des Linux-Servers ist auch Administrator für Tarantella, während die Rechte der *normalen* Anwender eingeschränkt sind. Verwenden Sie als Adresse den Alias, den Sie im `httpd.conf` angegeben haben, hier im Beispiel: `http://boss.lokales-netz.de:8080/tarantella` (oder `http://boss:8080/tarantella`), beim Native Client unbedingt einschließlich der Protokollangabe `http://` oder `https://`.

Tarantella über Browser

Unabhängig vom Betriebssystem des Client können Sie einen Java-fähigen Browser mit dem Tarantella-Server verbinden, indem Sie im Browser die Adresse des Tarantella-Servers eingeben, hier im Beispiel `http://boss.lokales-netz.de:8080/tarantella`. Die erforderliche Java Runtime-Umgebung bzw. eine Java Virtual Machine (Java VM) installieren einige Browser automatisch; sollten sie fehlen, können Sie diese nachinstallieren. Eine aktuelle Version des Java Environment finden Sie zum Beispiel bei `http://java.sun.com`.

Browser installieren die Java-Klassen für Tarantella je nach Bandbreite der Netz-anbindung innerhalb weniger Sekunden. Sie sehen den Abschluss der Installation im Login-Screen von Tarantella. Einige Browser, z. B. die von Netscape, empfehlen nach der Installation aus Sicherheitsgründen einen Neustart des Browsers.

Der Tarantella-Server prüft bei jeder Verbindung zum Client die Aktualität der installierten Java-Plug-Ins auf dem Client. Sind diese veraltet oder nicht installiert, veranlasst Tarantella eine Neuinstallation. Die Installationsroutine fragt dabei die Benutzer, je nach verwendetem Browser, ob sie die erforderlichen Java-Klassen installieren möchten. Anwender brauchen daher die Client-Software für Tarantella auf ihren Computern nicht selbst aktiv zu pflegen.



Abbildung 11.19: Tarantella Bildschirm nach Eingabe des Adress-Links

Nach dem Anmelden sehen Sie im Browser Ihres Clients zunächst den Standardbildschirm von Tarantella (Abbildung 11.20) mit einigen während der Installation angelegten Anwendungen (*xclock*, *Konsole*, *KDE* oder *Gnome* usw.) sowie Hilfetools für Administratoren und Anwender (wie *Administration Guide*, *Object Manager*, *Arraymanager* usw.).

Links im Bild sehen Sie (auf Farbmonitoren gelb/orange hinterlegt) das so genannte Launchpad. Anwender können über diese Leiste alle Anwendungen durch Klicken starten. Im freien Arbeitsbereich sehen Anwender generelle Informationen oder die von ihnen aufgerufene Anwendung.

Hinweis: Da die Tarantella-Client-Darstellung vollständig durch html und Javascript gesteuert ist, kann man diese nach Belieben für den Browserzugriff anpassen. Die folgenden Beispiele gehen immer von den »Standardansichten« aus.

Für den ersten Test mit Anwendungen des Applikations-Servers per Browser reichen die bei der Tarantella-Installation automatisch eingerichteten Anwendungen. Anwender können diese als Vorlage für weitere Applikationen nutzen.

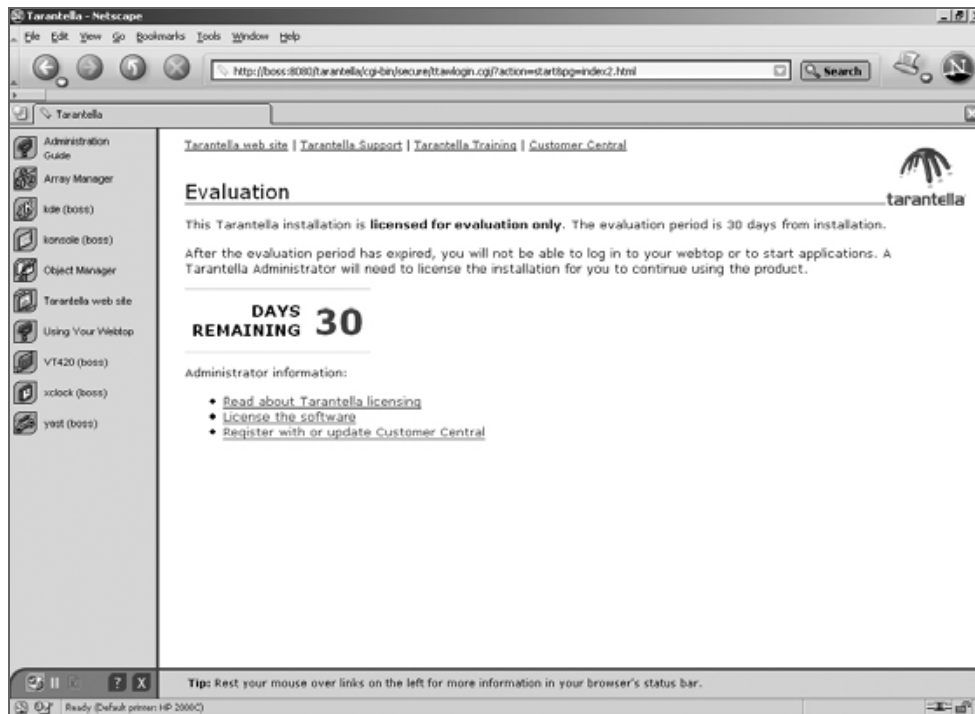


Abbildung 11.20: Tarantella-Standardbildschirm nach dem Einloggen

Tipp: Damit Sie X-Anwendungen starten können, muss im `/etc/ssh/ssh_config` »Forward X11« erlaubt sein!

Rufen Sie die gewünschten Anwendungen über deren Icon auf. Der Benutzer-Client baut dann, automatisiert über den Tarantella-Server, eine Verbindung zu dem Applikations-Server (hier dem Tarantella-Server selbst) auf und durchläuft dabei die Anmeldeprozedur am Applikations-Server. Standardmäßig versucht Tarantella, die Zugangsdaten in Tarantella auf die Applikation anzuwenden. Wenn diese Zugangsdaten von dem Applikations-Server akzeptiert werden, so fragt der Tarantella-Server nicht erneut nach einem Loginnamen/Passwort für den Applikationsstart.

Tipp: Sie können beliebig viele Applikationen gleichzeitig ablaufen lassen. Solange Sie das kleine grüne Zahnrad auf dem Applikations-Icon sehen, wissen Sie, dass die Anwendung läuft.

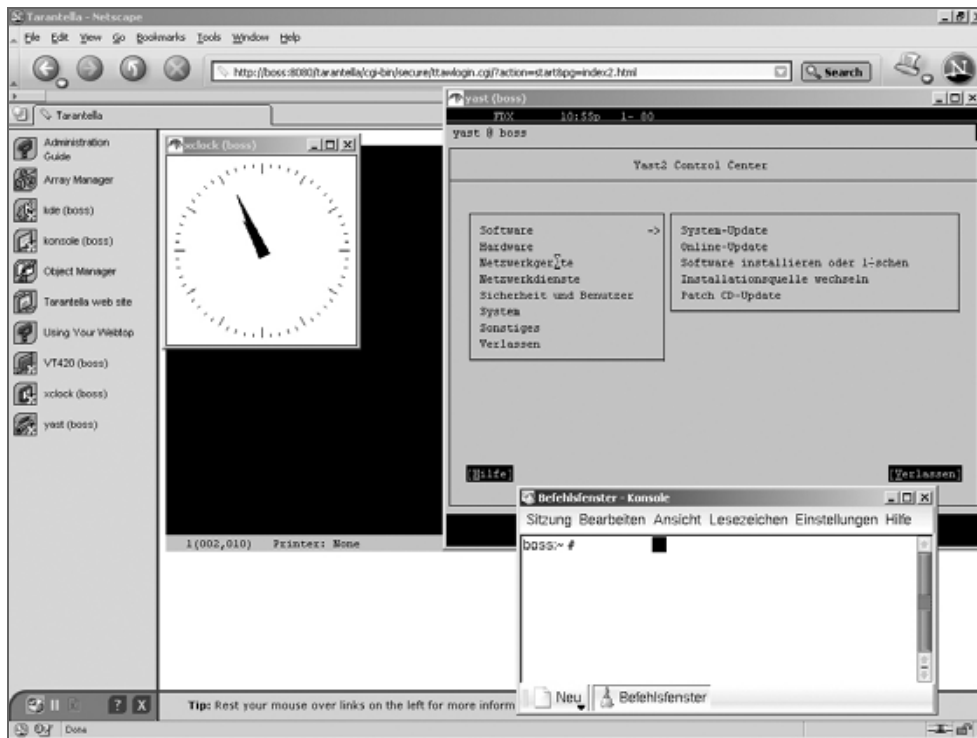


Abbildung 11.21: Tarantella mit verschiedenen Applikationen

Beenden kann man Anwendungen wie bei Windows oder wie bei Linux üblich (z. B. über *Datei • Beenden* bei Windows-Anwendungen oder z. B. über die Eingabe von `exit` im Terminalfenster). Sobald Benutzer sich aus Tarantella ausloggen, beendet Tarantella automatisch deren Anwendungen, falls Sie Tarantella nicht anders konfigurieren.

Tarantella unterscheidet die Darstellung im *Webtop* und die Darstellung als *Independent Windows/Kiosk-Modus* bzw. *Vollbilddarstellung* (Abbildung 11.22). Sie können den Fenstermodus bei Browserzugriff jederzeit wechseln, indem Sie bei gedrückter `[Strg]`-Taste nochmals auf das Icon der Anwendung im Launchpad klicken.

Wenn Sie sich mit bestimmten Login-Daten an einer Applikation anmelden möchten, so müssen Sie den durch Tarantella angelegten lokalen Passwortcache umgehen. Um den Passwortcache während des Applikationsstarts nicht abzufragen, können Sie während des Klickens auf das Anwendungs-Icon (also beim Programmaufruf) die `[⇧]`-Taste drücken. Dann fragt der Tarantella-Server Sie nach einem Usernamen und einem Kennwort für das System. Tarantella wendet dann die Tarantella-Anmeldedaten oder gespeicherten Kennwörter nicht an (Abbildung 11.23).

Anwender können ihre Anmeldedaten speichern, um diese beim nächsten Aufrufen dieser Anwendung nicht erneut eingeben zu müssen. Tarantella speichert das Passwort verschlüsselt (3DES).

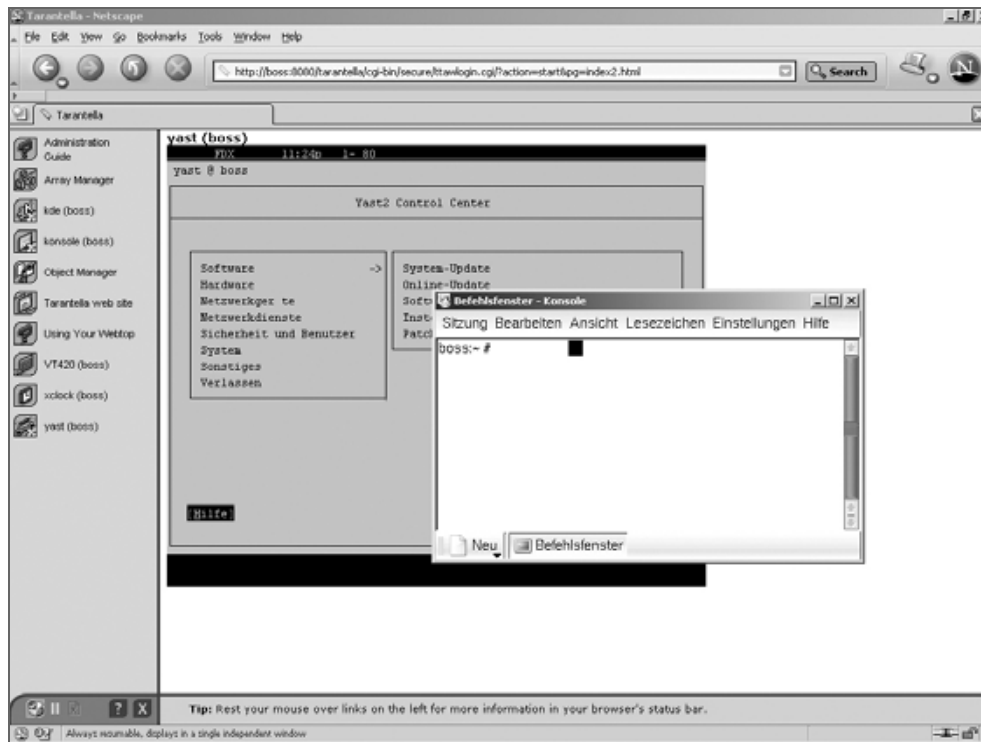


Abbildung 11.22: Tarantella mit YaST (Webtop) und Konsole (Ind. Windows)

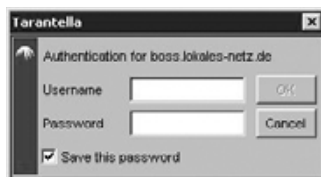


Abbildung 11.23: Username und Kennwort für eine Anwendung auf einem Server

Um sich vom Tarantella-Server abzumelden, können Sie das Kreuz am rechten unteren Rand des Launchpads betätigen. Nach einer Bestätigung gelangen Sie in das Standard-Abmeldefenster von Tarantella. Je nach Konfiguration beendet Tarantella nun die Anwendungen oder versetzt sie in einen Schlafzustand.

11.7 Tarantella konfigurieren und administrieren

Tarantella hat beim Installieren abhängig von Ihren Eingaben ein Installationsverzeichnis angelegt. Dieses gliedert sich in drei Unterverzeichnisse:

`/bin`: Sie können Tarantella über die Command-line (CLI) oder über grafische Administrationstools konfigurieren.

Um Tarantella über die Kommandozeile zu steuern, wechseln Sie in das Installationsverzeichnis und geben dort den Befehl `tarantella` gefolgt von einem weiteren Parameter. Gerade beim Anlegen mehrerer Objekte kann man mit ein wenig Übung sehr viel schneller zum Ziel kommen als über das grafische Hilfetool. Verschaffen Sie sich durch die Eingabe von `tarantella help` einen Überblick:

```
boss:~ # /opt/tarantella/bin/tarantella help

Usage: tarantella <command> [<command-specific args>]

Available commands:

archive          Archives the server's log files
array            Creates and manages arrays of Tarantella
                 ↙ servers
arraymanager     Starts Array Manager
config          Edits array-wide and server-specific
                 ↙ configuration
emulatorsession Lists and controls emulator sessions
help            Displays this list of commands
license         Adds, lists and removes Tarantella
                 ↙ license keys
object           Manipulates objects in the datastore
objectmanager   Starts Object Manager
passcache       Manipulates the password cache
print           Controls Tarantella printing services
query           Examines the server's log files
restart         Restarts Tarantella services
role            Configures role occupants and their
                 ↙ webtop links
setup           Changes Setup options, restores original
                 ↙ objects
start           Starts Tarantella services
status         Shows the current status of Tarantella
                 ↙ array members
stop           Stops Tarantella services
tscal          Lists, frees and returns Terminal Services
                 ↙ CALs
uninstall       Uninstalls Tarantella from this host
version        Displays versions of installed Tarantella
                 ↙ packages
```



```

webserver          Controls the Tarantella Web Server
webtopsession      Lists and controls webtop sessions

Use "tarantella <command> --help" to get help on a command.

boss:~ #

```

/etc: In diesem Verzeichnis finden Sie Konfigurationsdateien für Tarantella.

/var: In diesem Verzeichnis befinden sich die Userdaten, Applikationskonfigurationen, Zuordnungen sowie alle Änderungen am Webtop, Startfenster usw.

/webserver: In diesem Verzeichnis speichert Tarantella alle Dateien zur Tarantella-Version des Apache-Web-Servers einschließlich der httpd.conf.

Um das Risiko zu begrenzen, dass Ihre Konfigurationsmühen vergebens sind, sollten Sie zumindest das Verzeichnis /var regelmäßig sichern, besser das ganze Verzeichnis /opt/tarantella.

Tarantella bietet Administratoren (*root*) im angemeldeten Tarantellafenster zusätzliche Applikations-Icons wie das Tarantella-Hilfe-Dokument Administration-Guide sowie die Tarantella-Konfigurationstools Array Manager und Object Manager (siehe Abb. 11.20). Letztere vereinfachen das Einstellen der Parameter für Applikationen und User erheblich.

Mit dem Array-Manager konfiguriert man einmalig allgemeine Einstellungen wie z. B. Lizenzschlüssel, Farmen (Arrays) von Tarantella-Servern, Portdefinitionen, Anmeldeverhalten usw.

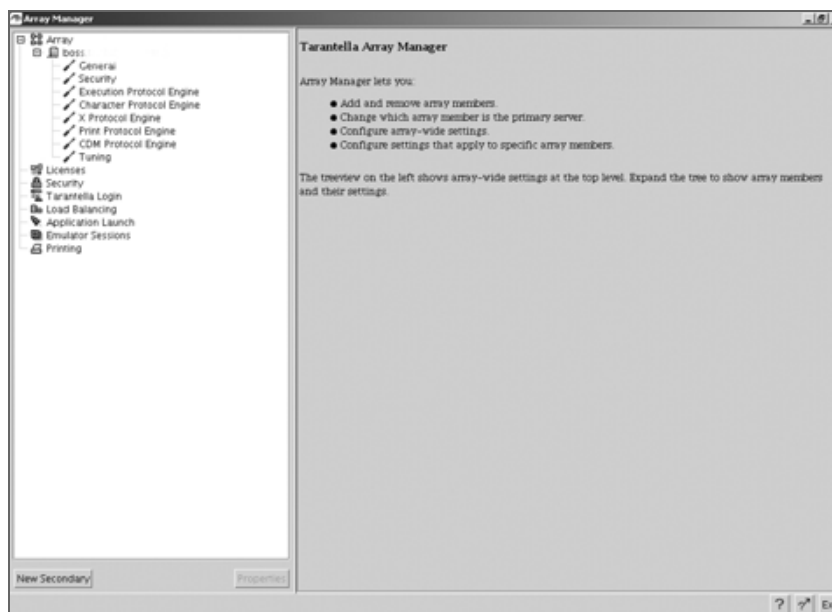


Abbildung 11.24: Array-Manager

Da der Object Manager zum Konfigurieren von Userdaten, Anwendungen, Zuordnungen und Sessions usw. dient, benötigt man ihn sehr häufig.

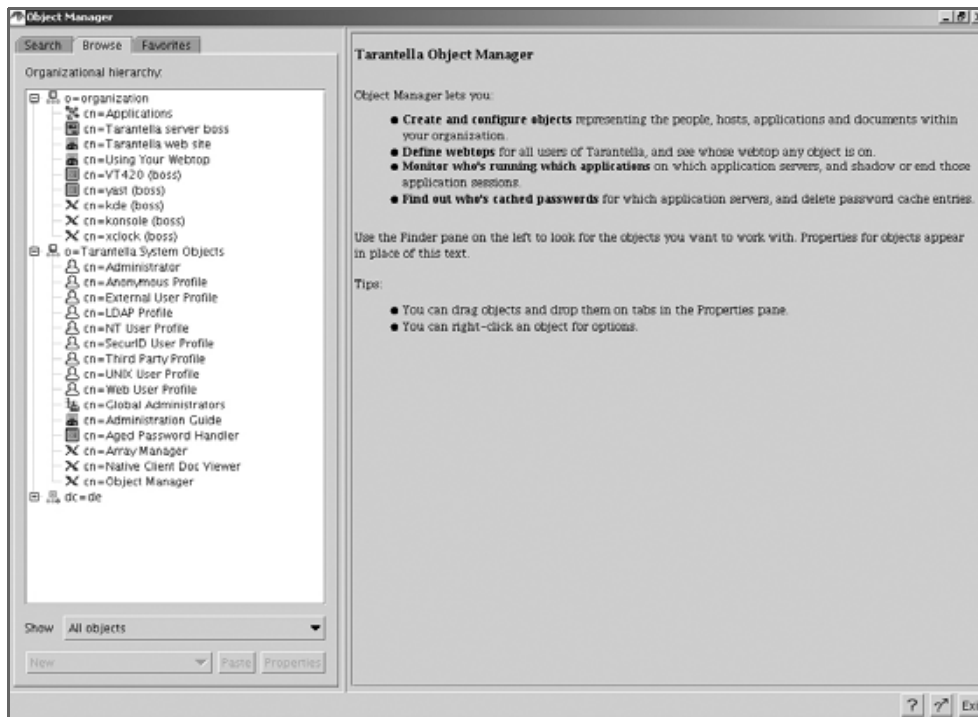


Abbildung 11.25: Object Manager

11.7.1 User für Tarantella anlegen

Die Benutzerverwaltung ist sehr vielschichtig, da sie auch den Zugriff sehr vieler zehntausend User unterstützen kann. Tarantella bietet acht verschiedene Arten einer User-Authentifizierung. Am üblichsten sind die Authentifizierung gegenüber lokal angelegten Unix-Accounts, Windows NT-Domänen, Web-Servern, Secure ID oder gegenüber einem Verzeichnisdienst wie LDAP oder dem Microsoft ADS-Server.

Ein dedizierter Server zur Benutzerverwaltung empfiehlt sich vor allem bei hohen Userzahlen oder wenn User auf mehrere unterschiedliche Applikations-Server zugreifen müssen. Damit legt man die Daten der Benutzer nicht auf allen Applikations-Servern an, sondern man definiert einmalig auf einem Verzeichnis-Server Userprofile, Mailadresse, Fileablage, Rechte usw.

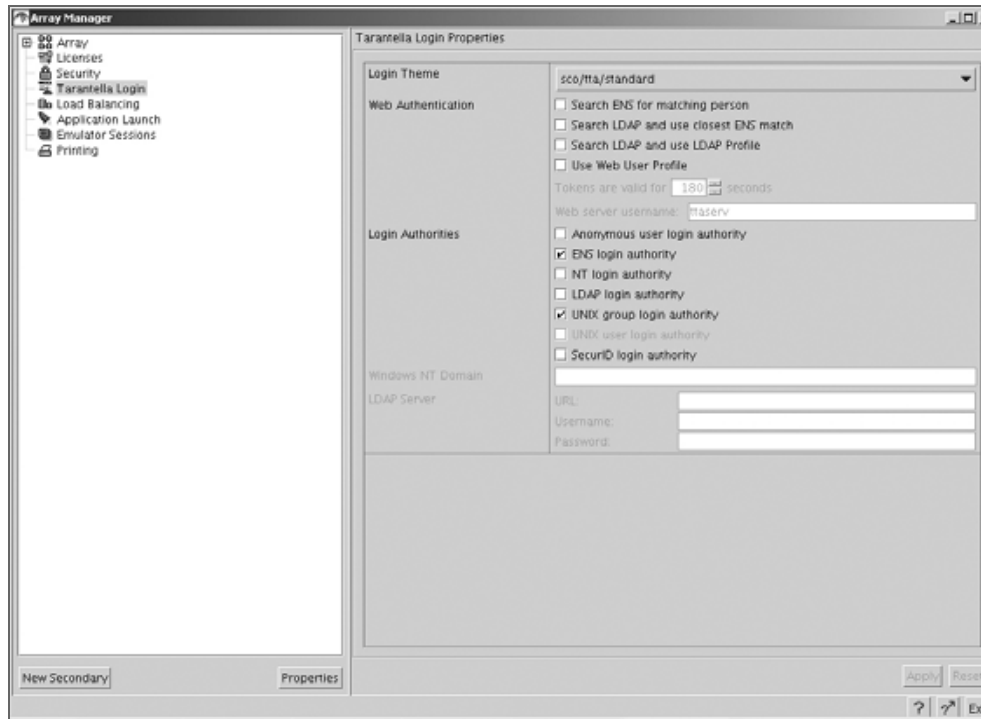


Abbildung 11.26: Userverwaltung in Tarantella

Voreingestellt ist eine Authentifizierung gegenüber einem Unix-Account. Alle User mit einem Unix-Account auf dem Tarantella-Server können sich an Tarantella anmelden. Möchten Sie (für Tarantella) weitere User anlegen, so verwenden Sie am besten die Benutzerverwaltung von YaST oder YaST2. Das Linux-Kennwort ist gleichzeitig das Kennwort für Tarantella. Um damit Erfahrungen zu sammeln, legen Sie bitte in YaST einen User *testuser* an und melden Sie sich mit dessen Namen an Tarantella an.

11.7.2 Applikationen zuordnen

Der Unterschied zwischen der Oberfläche des zuvor verwendeten Users *root* und dem soeben eingerichteten *testuser* ist leicht zu erkennen. Dem *testuser* fehlen die administrativen Tools von Tarantella. »Nicht-root«-Anwender dürfen weder auf den Object-Manager noch auf den Array-Manager zugreifen.

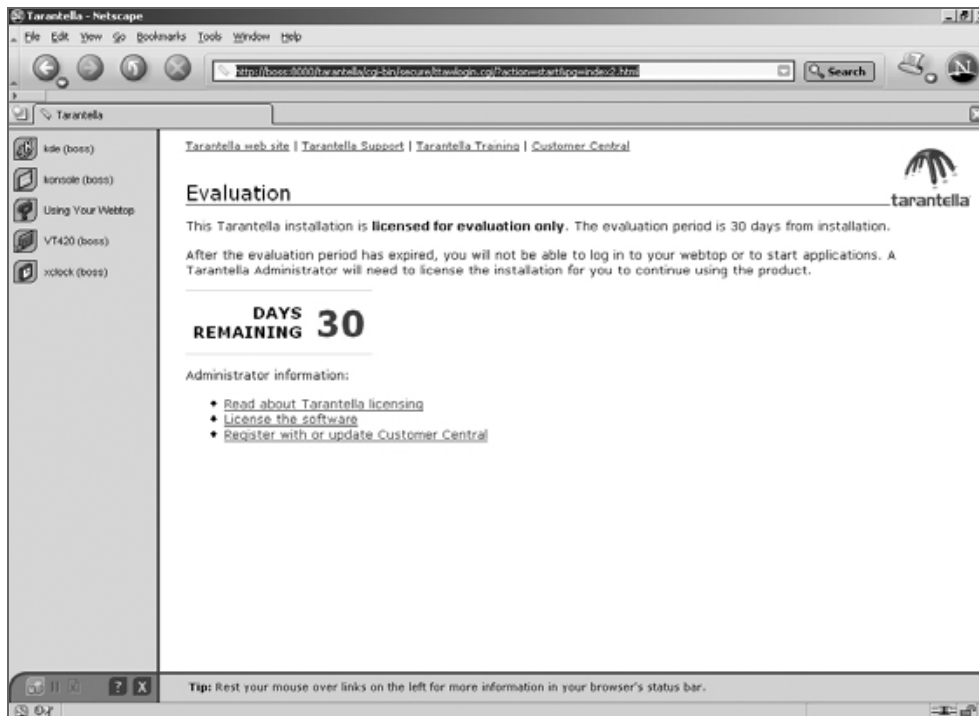


Abbildung 11.27: Anwenderdesktop

Administratoren legen im Object-Manager fest, welche Applikationen bestimmte User/Usergruppen sehen dürfen.

Die Installation von Tarantella legt dazu drei Grundstrukturen an. Diese sind im Object-Manager leicht als Wurzelverzeichnisse zu erkennen (siehe Abbildung 11.28).

Hinweis: Sowohl der Object- als auch der Array-Manger laufen auf dem Server in einem Tarantella-eigenen Java Runtime Environment. Aus diesem Grund gestaltet sich die Arbeit mit diesen stellenweise etwas träge. Achten Sie daher bei der Arbeit mit dem grafischen Administrationstool darauf, alle Änderungen wie eingestellt durch den Mausklick auf *Apply* zu übernehmen.

- Organization: User- und Applikationenobjekte. Administratoren können zum Gliedern eigene Organisationseinheiten (OUs) bestimmen.
- Tarantella System Objects: Hier legt Tarantella Systeminformationen wie den Array-Manager und den Object-Manager ab und verwaltet Administratorrechte. Diesen Zweig kann man nur bedingt ändern.

- `dc=de`: Bei Verwendung von Verzeichnisdiensten können die Strukturen, z. B. Object Units, Container, Gruppen usw., in Tarantella nachgebildet werden. Diese Gruppen kann man dann separiert für Tarantella konfigurieren und in diesem Zweig anlegen.

Öffnen Sie im Object Manager den Verzeichniszweig *Tarantella System Objects*. Dort hat jedes Anmeldeverfahren für Tarantella ein eigenes voreingestelltes Benutzerprofil. Wenn ein Benutzer sich in Tarantella anmeldet, entscheidet der Tarantella-Server selbstständig, abhängig von der gewählten Authentifizierung (Unix-Gruppen, NT-Domänen, LDAP-Profile), welches Profil und somit welche Anwendungen es dem Tarantella-User zuweist.

Wählen Sie das Profil *Unix User Profile* durch Doppelklick aus. Die anzuzeigenden Anwendungen definieren Sie im rechten Fenster auf der Registerkarte *Links*.

Um eine eigene Applikationen zuordnen zu können und die durch Tarantella definierten Standardapplikationen auszublenden, markieren Sie mit der rechten Maustaste die *OU o=organization* im rechten Fenster und wählen *Remove*. Das Anzeigefenster der Registerkarte *Links* ist nun leer. Unix-Anwendern, die sich nun bei Tarantella anmelden würden, stünde keine Applikation zum Starten zur Verfügung.

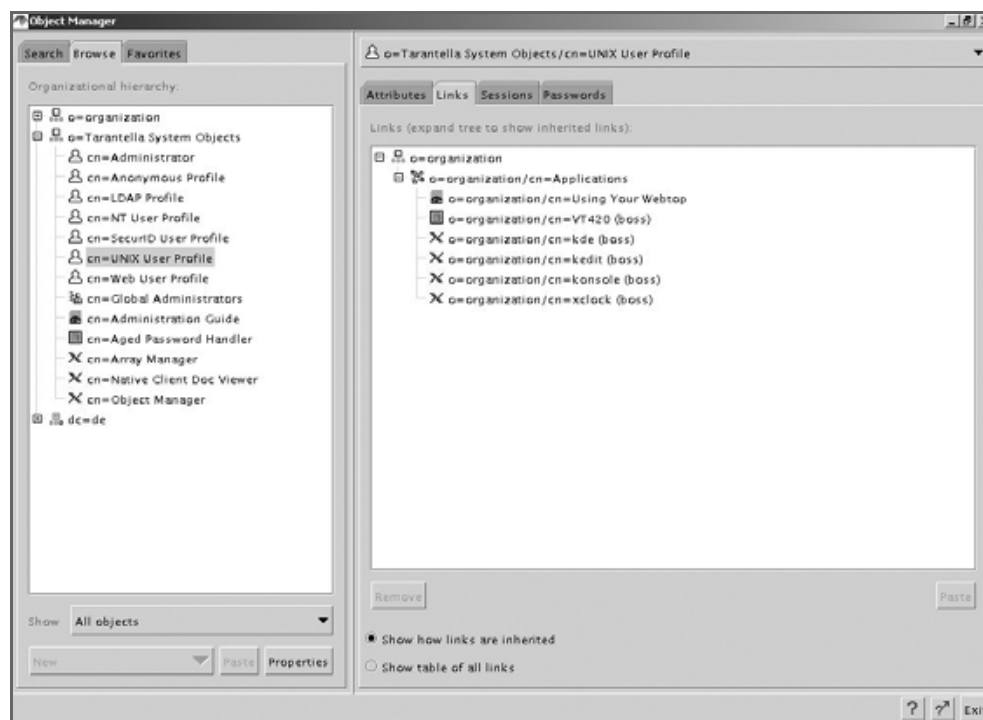


Abbildung 11.28: Profiles und Anwendungszuordnung

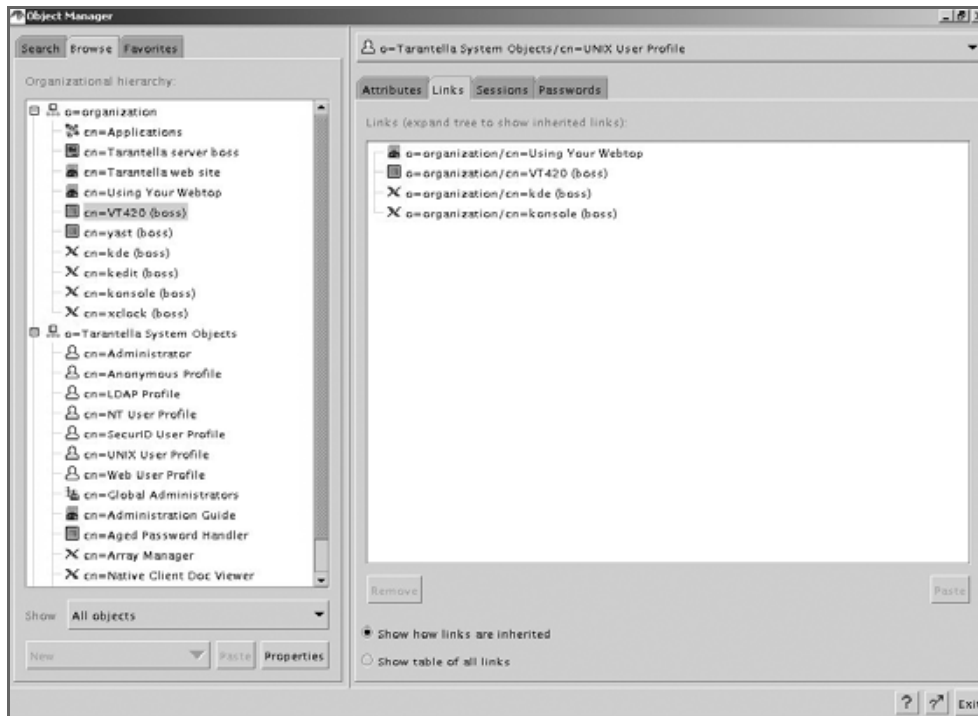


Abbildung 11.29: Definieren von Zuordnungen

Öffnen Sie nun gleichzeitig den Verzeichniszweig `o=organization`. Sie sehen die bei der Installation von Tarantella angelegten Applikationen (z. B. `xclock`, `VT420` usw.). Sie können diese Applikationen nun mit der Maus in die Registerkarte *Links* ziehen und somit die Anwendungen den *Unix Usern* unter Tarantella zur Verfügung stellen. In Abbildung 11.29 sind die Applikationen »Using your Webtop«, `VT420`, `KDE` und die *Konsole* zugewiesen. Sie können über diesen Weg pro Benutzer bis zu 512 Anwendungen bzw. Profile zuordnen.

11.7.3 Unix-Applikationen definieren

Eine neue Applikation legt man in drei Schritten an:

1. Definieren Sie den Host (Applikations-Server), auf welchem die Anwendung ablaufen soll (oder verwenden Sie den bereits angelegten).
2. Definieren Sie die Anwendung und verknüpfen Sie diese mit dem unter 1. definierten Host.
3. Weisen Sie die neue Applikation den Usern/Usergruppen zu.

Beliebige neue Objekte definiert man in Tarantella immer auf dem gleichen Weg:

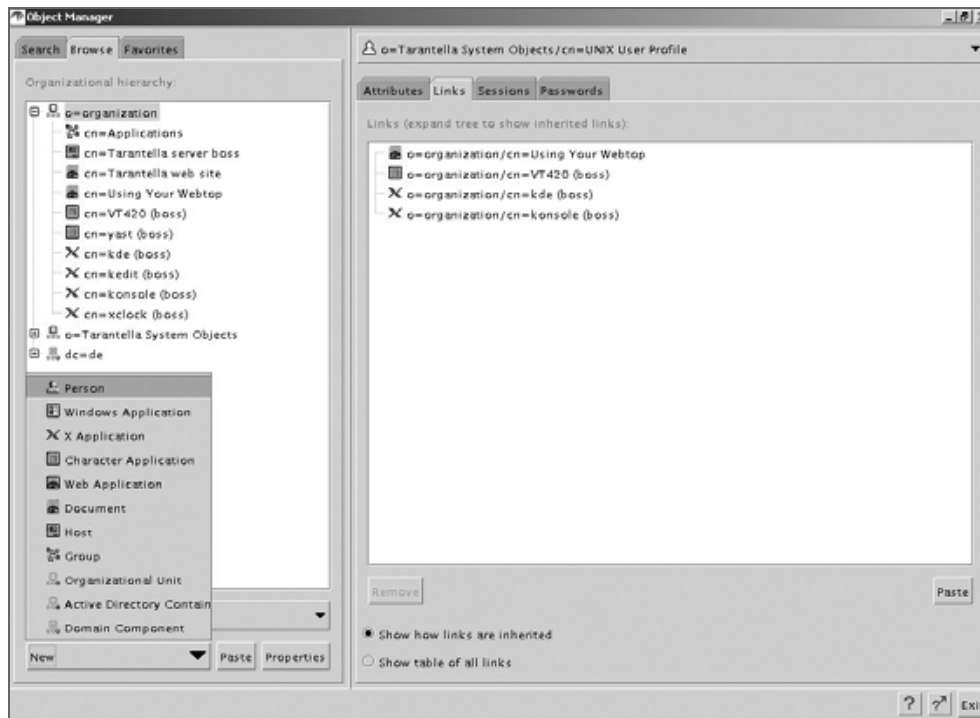


Abbildung 11.30: Neue Objekte erstellen

Wählen Sie zunächst den Verzeichniszweig, in welchem Sie das Objekt erstellen möchten, und wählen Sie dann die Schaltfläche *New*.

Erstellen Sie ein neues Hostobjekt und benennen Sie es z. B. *Linuxserver* und bestätigen Sie Ihre Eingabe. Die rechte Fensterhälfte verlangt nun Einträge:

- *Name*: Geben Sie dem neu erstellten Objekt einen beliebigen Namen.
- *Description*: Aussagekräftige Kurzbeschreibung.
- *Adress*: Tragen Sie hier den vollständigen Hostnamen ein. Erfahrungsgemäß führen hier numerische IP-Adressen zu Schwierigkeiten. Testen Sie den Hostname zuvor am besten mit *ping* und *nslookup*.
- *Windows NT Domain*: Bei Verwendung von Windows-Applikations-Servern tragen Sie die Domäne oder den Rechnernamen ein. Dies entfällt bei Unix-/Linux-Server.
- *Authentication*: Wählen Sie, ob das Tarantella-Login auf den Server angewendet werden soll.

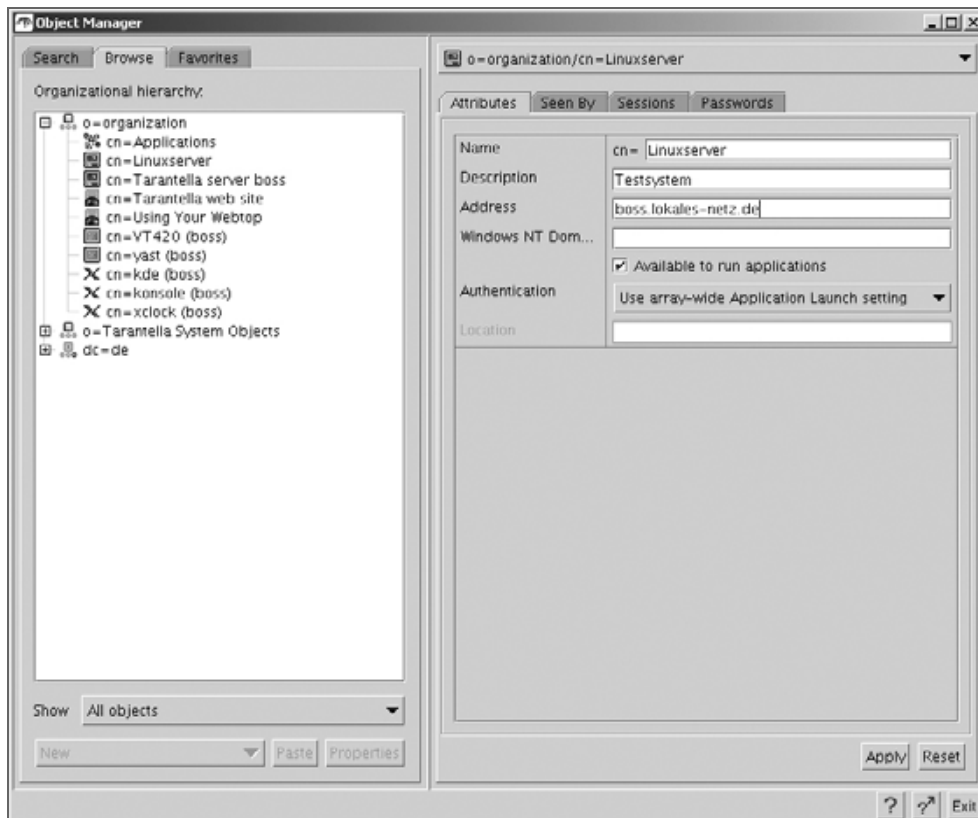


Abbildung 11.31: Neuer Host

Als zweiten Schritt zum Definieren einer Anwendung müssen Sie diese selbst auf dem gleichen Weg wie zuvor den Host definieren. Erstellen Sie ein neues Objekt und verwenden Sie als Objekttyp diesmal die Applikation, welche Sie veröffentlichen möchten. Zum Darstellen von X11-Anwendungen wählen Sie den Typ X-Application und benennen Sie diesen hier.

Tarantella hat erneut die rechte Fensterhälfte verändert. Wenn Sie sich das Bearbeiten der folgenden Einstellmöglichkeiten erleichtern wollen, sollten Sie schon angelegte Objekte als Referenz verwenden:

- *Name*: Geben Sie dem neu erstellten Objekt einen aussagekräftigen Namen.
- *Description*: Aussagekräftige Kurzbeschreibung.
- *Application*: Tragen Sie hier die Anwendung inkl. Pfad ein. Beispiele: /opt/kde2/bin/startkde (für eine KDE-Session), /usr/dt/bin/dtsession (CDE-Session), /sbin/soffice (Staroffice), /sbin/xeyes (Spiel) usw.

- *Connection Method*: Tarantella meldet Benutzer auf dem Applikations-Server über Zugangsprotokolle wie ssh, telnet, rexec, rcmd usw. an. Wenn telnet generell oder für den anzumeldenden User `disabled` ist (Standard bei SuSE 8.2 Professional), wird der Verbindungsaufbau von dritten Rechnern scheitern. Wählen Sie eine andere Konnektierungsmethode (z. B. über ssh oder rexec) oder ermöglichen Sie den Zugriff über telnet (definierbar in `/etc/rc.config`). Beachten Sie, dass für die Verbindungsmethode SSH das X11-Forwarding in der `/etc/ssh/ssh_config` erlaubt sein muss.
- *Resumable*: Hier legen Sie fest, was mit den Prozessen der Applikation geschehen soll, wenn Benutzer diese nicht ordnungsgemäß beenden (z. B. bei Verbindungsverlust einer GSM-Verbindung).
- *Never*: Tarantella beendet die Verbindung bei Verlassen des Screens.
- *Tarantella Webtop Session*: Die Verbindung bleibt so lange erhalten, wie man in Tarantella angemeldet ist.
- *Always*: Die Prozesse der Applikation bleiben so lange erhalten, bis Benutzer sie ordnungsgemäß beenden (z. B. über *Datei • Beenden*).
- *Session Ends When*: Gibt an, wie der Tarantella-Server erkennt, wie ein Anwender die Tarantella-Applikation beendet.
- *Display using*: Hier können Sie festlegen, in welchem Fenstermodus die Anwendung läuft.
- *Webtop*: Tarantella zeigt das Programm im freien Bereich neben dem Launchpad an. Die Größe definieren Sie in den nächsten Zeilen unter *Width* und *Height*.
Dieser Modus eignet sich zur Ansicht kleiner Darstellungen wie `xclock`, `calc`, `yast`, `VT420` und Mainframe-Masken. Beispiel: Abbildung 11.22, YaST.
- *Independent Windows*: Das Programm läuft in einem eigenem Fenster mit Rahmen in den Größen *Width* und *Height*. Dieser Modus eignet sich für Büro-Anwendungen wie `kWrite`, `MS Word`, `MS Excel`, `Star Office`. Beispiel: Abbildung 11.22, Konsole.
- *Client Windows Management*: X-Applikationen können das Fenstermanagement an den Tarantella-Client übergeben.
Die Anwendung definiert die Größe des Fensters selbst vor. Tarantella-User können diese durch kleiner- bzw. größer ziehen des Rahmens frei ändern. Dieser Modus läuft nicht mit Betriebssystemoberflächen (KDE, Gnome, CDE, Windows-Desktop), sondern eignet sich nur für einzelne X11- bzw. Javaprogramme (`KWrite`, `xclock`). Beispiel: Abbildung 11.22, `xclock`.

- **Kiosk:** Tarantella stellt das Programm ausschließlich auf dem Client ohne Rahmen in der maximalen Größe dar und überdeckt das Betriebssystem des Tarantella-Clients vollständig. Dies empfiehlt sich zur Darstellung von Betriebssystemoberflächen.
- **Colour Depth:** Hier können Sie die Farbtiefe bestimmen. Je mehr Farben Sie verwenden, desto mehr Bandbreite benötigt die Übertragung. Der Ressourcenbedarf (Speicher/CPU-Last pro User) auf dem Tarantella-Server ändert sich dabei nur unwesentlich.
- **Webtop Icon:** Wählen Sie hier das gewünschte Icon aus. Tarantella zeigt das hier ausgewählte Icon auf dem Launchpad an und zeigt Anwendern damit die konfigurierte Applikation.
- Definieren Sie nun, wie hier im Beispiel in Abbildung 11.32, die Applikation YaST2 und bestätigen Sie Ihre Angaben mit der Schaltfläche *Apply*:

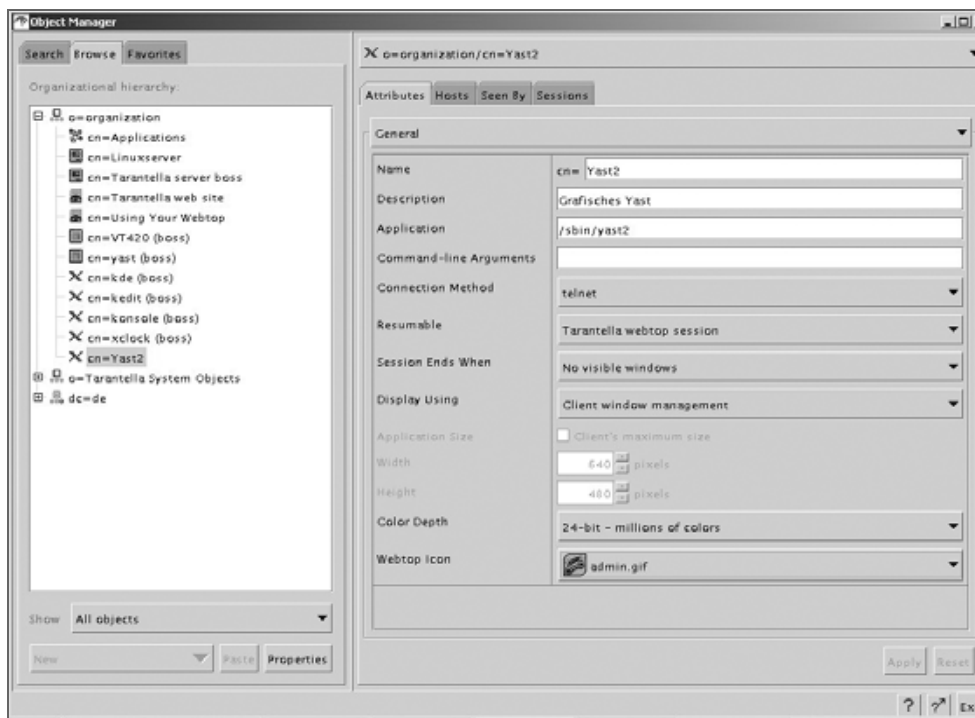


Abbildung 11.32: Neue Applikation »YaST2«

Nun ist die Applikation konfiguriert, jedoch ist noch nicht definiert, auf welchem Server sie ablaufen soll. Wechseln Sie dazu auf den Tab *Hosts* und ziehen Sie das zuvor angelegte Objekt *Linuxserver* von der linken in die rechte Fensterhälfte.

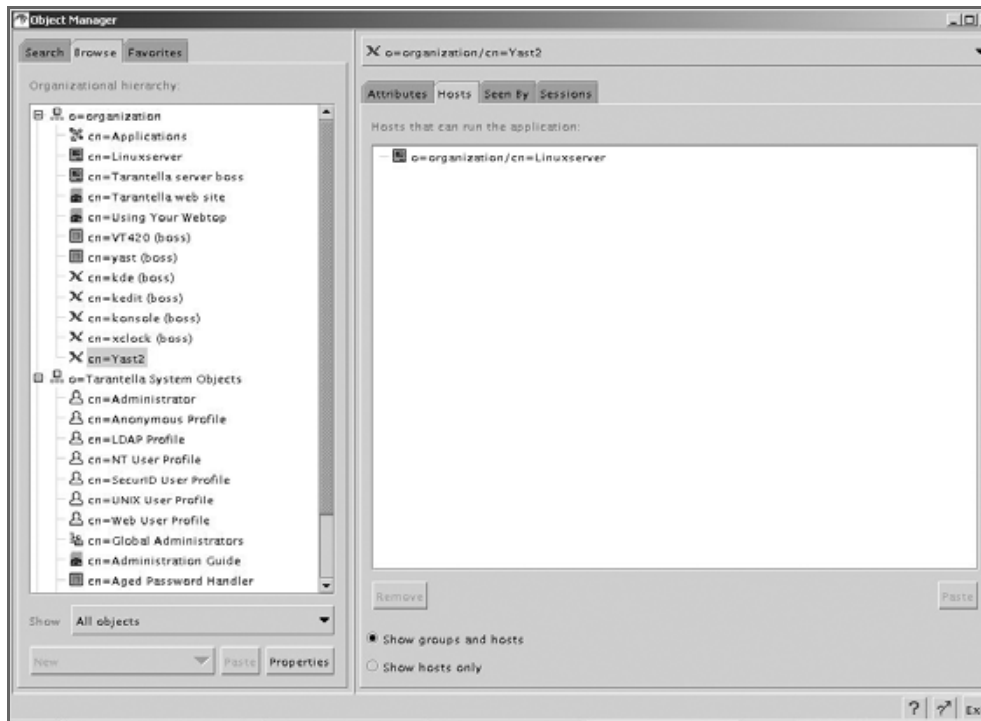


Abbildung 11.33: Zuordnung Applikation »YaST2« zu Host »Linuxserver«

Als dritten und letzten Schritt müssen Sie wie bereits zuvor beschrieben die Applikation für die Gruppe Unix-User verfügbar machen. Dazu ziehen Sie das neue Objekt, hier im Beispiel YaST2, auf die Registerkarte *Links* des Objekts Unix User Profile. Die neue Applikation YaST2 steht danach allen Benutzern zur Verfügung, die sich über die Unix-Gruppen anmelden.

Tipp: Sollten User während der Konfiguration bereits in Tarantella angemeldet gewesen sein, so aktualisiert Tarantella deren Launchpad erst bei erneutem Einloggen.

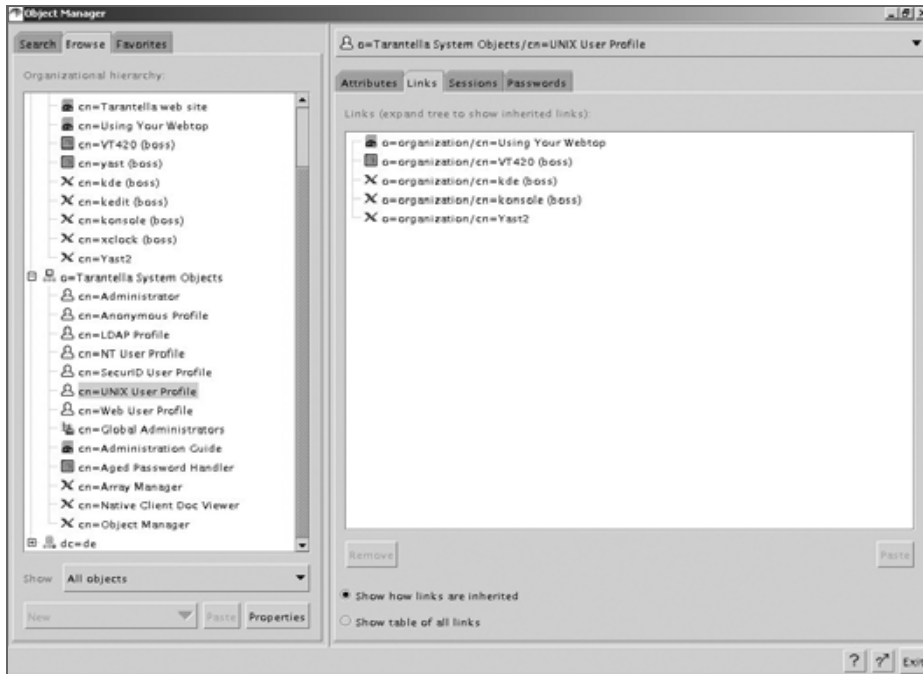


Abbildung 11.34: Zuordnung Applikation »YaST2« zu Gruppe »Unix User Profile«



Abbildung 11.35: Applikation »YaST2«

11.7.4 Windows-Anwendungen definieren

Das Installieren und Konfigurieren von Windows-Terminaldiensten (Windows 2000-Server, Windows 2003) und darauf laufenden Anwendungen gestaltet sich oft aufwändig. Die folgenden Erklärungen gehen daher von einem funktionierenden Windows-Server mit freigeschalteten Terminaldiensten aus.

Für die Zusammenarbeit zwischen Tarantella und Windows-Terminalservern sollte man vorab zwei Grundeinstellungen ändern. Manche Funktionen von Tarantella können auch Microsofts Windows-Terminalserver bieten (z. B. gibt es das Session-Resume auch in Microsoft Windows). Es ist offensichtlich, dass man zusammengehörende Funktionen passend konfigurieren muss.

- So empfiehlt es sich, die Kennwortabfrage der Microsoft Windows Terminal-Server zu deaktivieren, damit Tarantella die Anmeldung durchreichen kann, ohne dass die Benutzer sich ein zweites Mal bei Microsoft Windows anmelden müssen. Hinweis: Diese Einstellung müssen Sie nicht für Windows 2003-Systeme vornehmen, da Microsoft diese Standardeinstellung bereits geändert hat.
- Da man das Resume-Verhalten von Sessions auch auf Microsoft Windows-Terminalservern einstellen kann, sollte man sich entscheiden, welcher Server das Resume-Verhalten kontrollieren soll, Tarantella oder die Microsoft Windows-Terminalserver.
- Sie können beide Einstellungen auf dem Microsoft Windows Terminal-Server im Menüpunkt *Start • All Programs • Administrative Tools • Terminal Services Configuration* ändern. Wählen Sie dieses Programm und hier das Verzeichnis *Connections • RDP*.



Abbildung 11.36: Einstellungen RDP
TCP / Kennwort



Abbildung 11.37: Einstellungen RDP
TCP / Sitzungen

Durch Doppelklick öffnet die *Terminaldienstkonfiguration* vom Microsoft Windows 2003 das Fenster *Eigenschaften RDP-TCP*. Wählen Sie zunächst die Registerkarte *Logon Settings* und deaktivieren Sie den Punkt *Always prompt for password*, damit Tarantella Kennwörter an den Microsoft Windows Terminal-Server weiterreichen kann.

Wechseln Sie nun auf die Registerkarte *Session* und definieren Sie, die Microsoft Windows-Sitzung zu beenden, wenn die Verbindung getrennt wurde. Damit kontrolliert Tarantella die Benutzersitzungen.

Konfiguration eines Windows-Objektes

- Windows-Sitzungen legt man nach dem gleichen Schema wie X-Applikations-Sitzungen an.
- Definieren Sie zunächst ein Hostobjekt mit der (Host-) Adresse des Terminal-Servers.
- Legen Sie ein neues Objekt *Windows Application* an.
- Weisen Sie das Objekt der gewünschte Usergruppe zu.

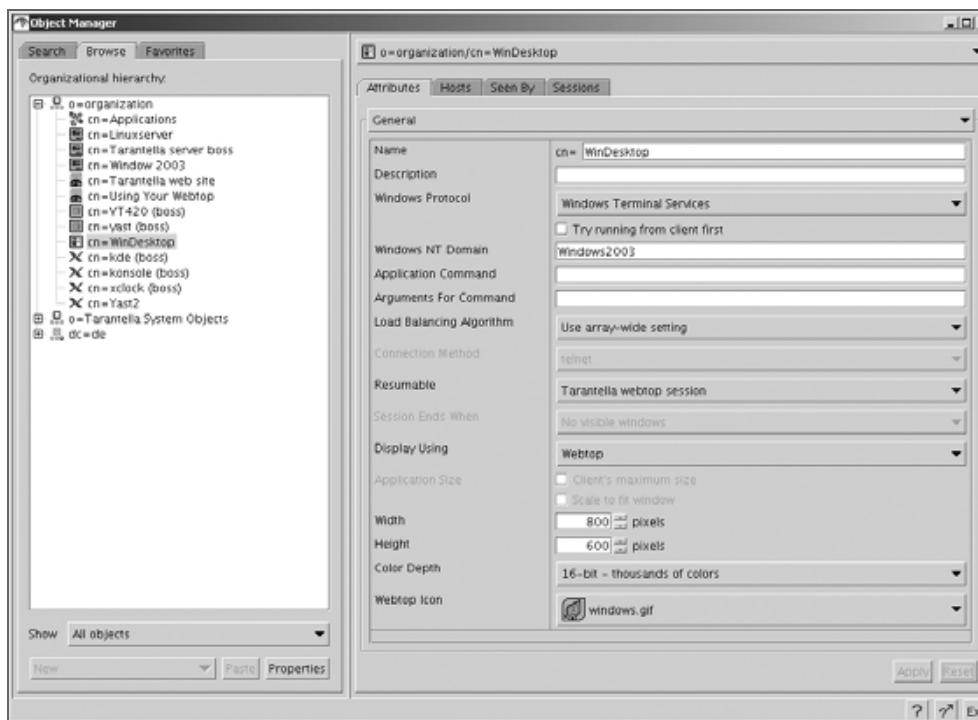


Abbildung 11.38: Konfiguration einer Windows Applikation

Unterschiede zwischen X- und Windows-Applikationen.

Es gibt einige Unterschiede zwischen X- und Windows-Applikationen und deren Konfiguration für Tarantella:

- **NT-Domain:** Tragen Sie hier bitte die NT-Domäne für den Windows-Applikations-Server ein. Sollten Anwender nicht mit einer Domäne arbeiten, so tragen Sie bitte den Hostnamen des Terminal-Servers ein.
- **Application:** Genauso wie bei X-Applikation tragen Sie hier die Anwendungspfade für Einzelapplikationen ein (z. B. c:\Programme\Microsoft Office\Office\winword.exe). Tragen Sie in dieses Feld nichts ein, stellt Tarantella den ganzen Windows Desktop dar.
- **Display Using:** Das Client Windows Management steht leider nicht für Windows-Applikationen zur Verfügung. Die Größe eines Fensters einer Tarantella-Windows-Applikation kann man über eine neue Scaling« Funktion anpassen (Scale to fit into Windows).
- **Connection Method:** Windows verwendet ein eigenes Protokoll. RDP authentifiziert Benutzer selbst.
- **Colour Depth:** Für die Verbindung zwischen Windows 2000-Terminal-Server und Tarantella-Server steht nur eine Farbtiefe von 8 Bit zur Verfügung. Bei Windows 2003-Terminal-Server können Sie zwischen 8 Bit, 16 Bit und 24 Bit wählen.

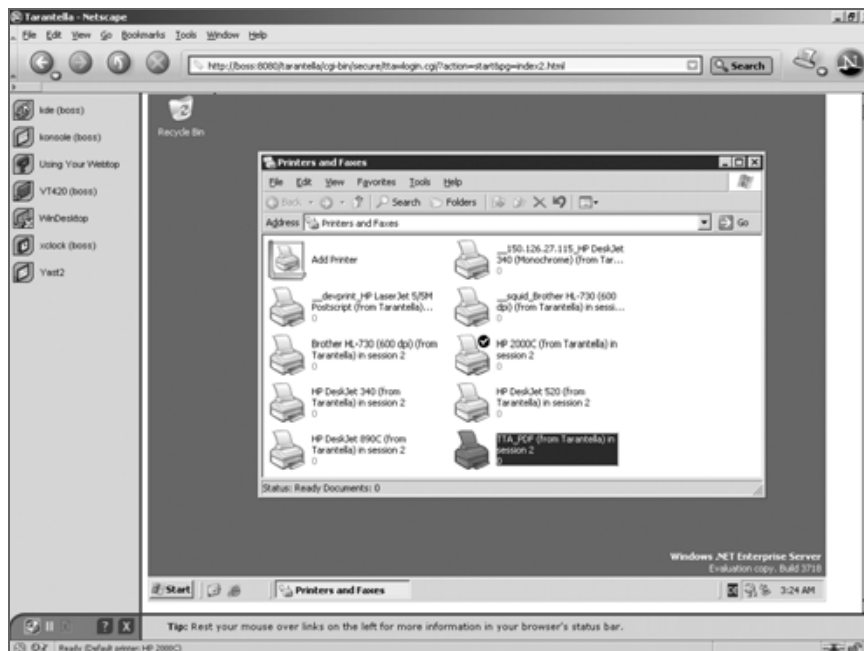


Abbildung 11.39: Windows-Desktop im Tarantella-Webtop

11.8 Drucken unter Tarantella

Beim Vermitteln von Sitzungen durch Tarantella können Sie im LAN herkömmliche Netzwerk-Druckmethoden beibehalten. Wenn Sie jedoch über WAN-Strecken drucken möchten, so empfiehlt sich der Einsatz der Druckdienste von Tarantella. Diese senden die Druckdaten ebenfalls über AIP (bei Bedarf verschlüsselt und komprimiert) an den Standarddrucker des Clients.

Tarantella selbst fungiert hierbei als zusätzlicher Druck-Server, der Druckdaten empfangen und automatisiert an Clients versenden kann.

Der Tarantella-Client erkennt die Clientdrucker automatisiert. Bei einer Anmeldung an Tarantella liest eine Javaklasse (bzw. der Native Client) die Drucker des Clients aus und teilt sie dem Tarantella-Server mit.

Zusätzlich richtet Tarantella einen neuen Drucker TTA_PDF ein. Dieser Drucker erzeugt automatisiert Druckdateien im PDF-Format und überträgt diese an den Tarantella Client. Dieser öffnet dann ein Programm zum Darstellen von PDF-Dokumenten).

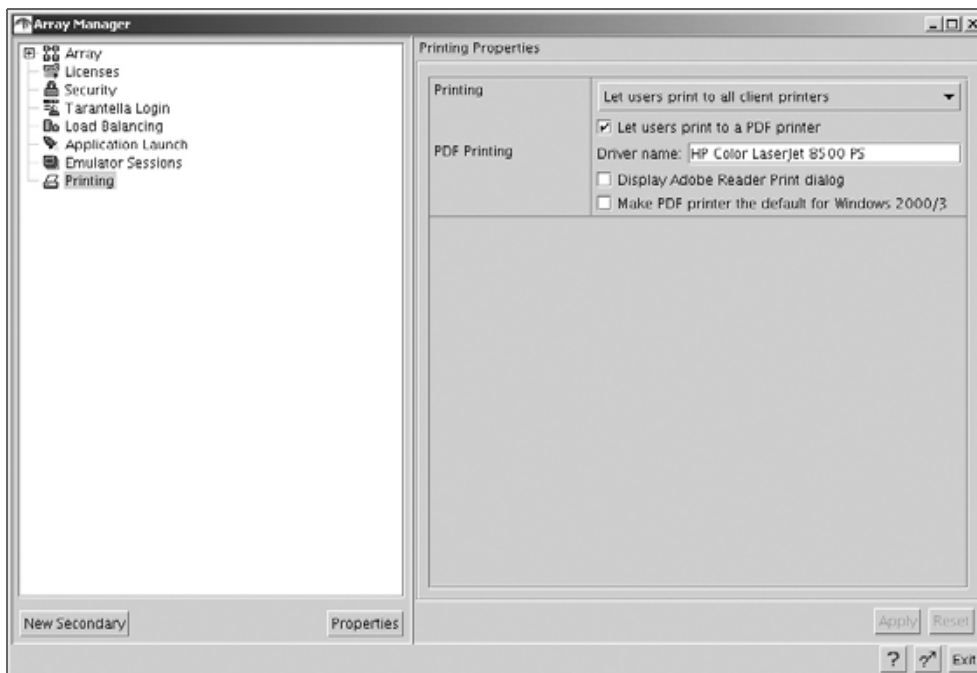


Abbildung 11.40: Einstellungen Tarantella: Drucken

Generell unterscheidet man das Drucken von Microsoft Windows-Terminalservern und von Unix-/Linux-Applikations-Servern: Damit Sie unter Windows-Applikationen drucken können, müssen Sie zunächst in Tarantella nichts zusätzlich konfigurieren. Tarantella ordnet alle Tarantella-Clientseitigen Drucker automatisch den RDP-Druckfunktionalitäten zu. Sie können verschiedene Funktionen für das Drucken umändern. Die Einstellmöglichkeiten für das Drucken sind hierbei sehr vielfältig (Standarddrucker, PDF-Drucker, Anzahl). Beachten Sie die Konfigurationsmöglichkeiten im Array-Manager.

Möchten Sie von einem Unix-/Linux-Applikations-Server drucken, so müssen Sie das Skript `prtinstall.en.sh` von `/opt/tarantella/bin/bin/scripts` auf den Applikations-Server kopieren und dort ausführen. Als einzige Eingabe geben Sie den Namen des Tarantella-Servers an, auf welchem Anwender drucken sollen. Der eingerichtete Drucker heißt dann `tta_printer`.

```
boss:/opt/tarantella/bin/scripts # sh prtinstall.en.sh
DNS name of primary Tarantella server
(press [Return] to accept default [boss.lokale-netz.de]):
Installing printer configuration file...
Installing lp/lpr "wrapper" scripts...
Local system type is Intel Linux.
Informing lp subsystem about local printer tta_printer...
Success.
boss:/ #
```

Die grundlegenden Funktionen von Tarantella haben Sie nun kennen gelernt. Die Funktionen und Konfigurationen von Tarantella sind sehr vielseitig und würden den Umfang dieses Buchs bei weitem sprengen. Wenn Sie weitere Funktionen einrichten oder Tarantella individuell anpassen möchten, blicken Sie bitte in die anfangs beschriebenen weiterführenden Dokumente:

- **Admin Guide:** Den Admin Guide finden Sie als html-Version als Applikation bei Anmeldung als Tarantella-Admin in Tarantella oder im Internet unter http://www.tarantella.com/knowhow/e3.4/help/en-us/admintocs/TOC_FUNC_TYPE.html
- **Newsgroup:** <http://www.tarantella.com/support/newsgroups/>
- **Know how:** <http://www.tarantella.com/knowhow/e3.4/>
http://www.tarantella.com/knowhow/e3.4/help/en-us/admintocs/TOC_FUNC_TYPE.html
- **Newsgroup:** <http://www.tarantella.com/support/newsgroups/>
- **Know how:** <http://www.tarantella.com/knowhow/e3.4/>