

SLAC

**Mit „SUSE Linux“
glücklich werden**

Schlomo Schapiro
Principal Consultant
Leitung Virtualisierung & Open Source

12.12.2008

Agenda

- Professioneller Linuxbetrieb
- Software Management mit RPM und Repositories
 - Eigene RPM Pakete
 - Der „SUSE Weg“
 - Novell Subscription Management Tool
- To YaST or Not To YaST
- openSUSE oder SUSE Linux Enterprise Server ?
- Kommerzieller Support – was geht und was nicht geht

Kann man mit OSS-OS glücklich werden?

NEIN

- Never Ending Story
- Nach dem Bug ist vor dem Bug
- Viel zu viele Möglichkeiten

JA

- Herr im Hause sein
- Alles ist möglich
- Freie Auswahl
- Oftmals unkomplizierte und direkte Ansprechpartner

Professioneller Linuxbetrieb



Professioneller Linuxbetrieb – Prozesse

- **Standardisierung der Software, Versionen und Prozesse**
 - **Problemloser Betrieb von vielen Systemen**
 - **Viele Administratoren arbeiten zusammen**
 - **Verwaltung vieler unterschiedlicher Versionen**
- **Dokumentation und Validierung**
 - **Nachvollziehbarkeit über gesamten Lebenszyklus**
 - **Server Logbuch, Changes**
 - **Automatische Dokumentation & Validierung**

Professioneller Linuxbetrieb – Lösungen

▮ Stabile Betriebsumgebung

- ▮ Zertifizierte Hard- und Software

- ▮ Ausgereifte Lösungen

- ▮ Langfristige Updates und belastbarer Support

- ▮ Netzwerk von Unterstützern

▮ NO-NO

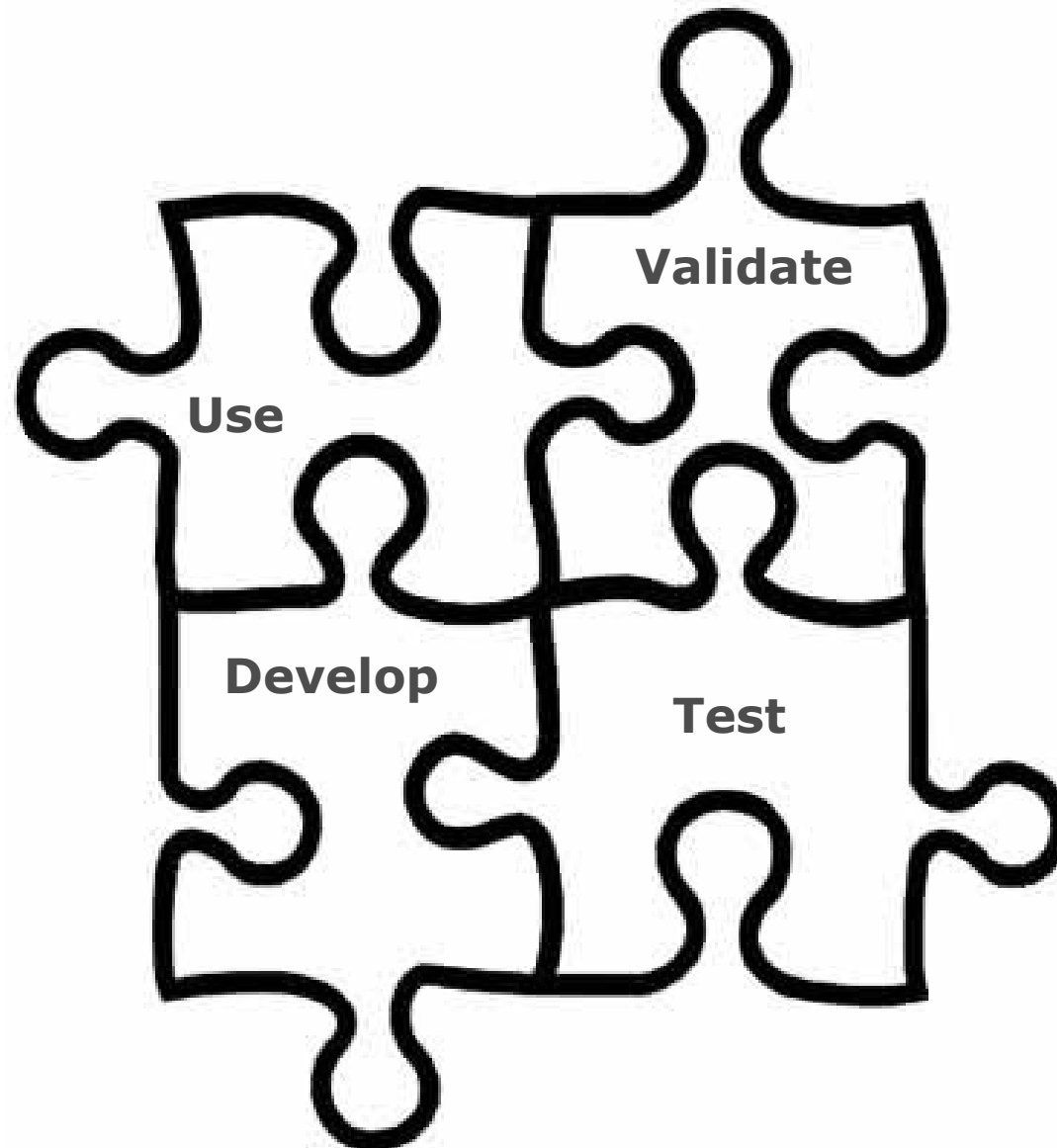
- ▮ Frickellösungen

- ▮ Quick-n-Dirty

- ▮ Hab' ich doch im Kopf

- ▮ Mal eben ein Update einspielen

Softwaremanagement



Software Management mit RPM

- RPM ist die Basis von SUSE Linux
- RPM Features
 - Inventarisierung
 - Validierung der Paketquellen und installierten Systeme
 - Abhängigkeiten & Konflikte
 - Updates und Patche
 - Dateien und Skripte
- Optimale Plattform für Softwaremanagement
- Perfekte Integration in Betriebssystem

Systemmanagement mit RPM

- Konfigurationen und Einstellungen als RPM verteilen
- Einfachster Weg zur Standardisierung
- Lohnt sich meist schon ab wenigen Systemen
- Paketinhalte:
 - Dateien
 - Skripte zur Konfiguration und zur Validierung
- **VORSICHT:**
Paketkonflikte zu Konfigurationsdateien vermeiden!

Softwaremanagement mit Repositories

- Vertrauensstellung über Signaturen
 - Installation nur aus vertrauten Quellen
- Zentrale Verteilung von RPM Paketen
- Zentrale Verteilung von Updates
- Integration eigener Inhalte in die vorhandene Softwareverwaltung des Betriebssystems
 - Korrekte Auflösung von Abhängigkeiten
 - Eigene Aktualisierungen von Distributionspaketen
 - Zusatzsoftware zentral bereitstellen

Repositories (selber machen)

- Relevant: RPMMD/YUM und YaST Repositories

- Inhalt

 - RPMs und Metadaten

 - Patterns und Signaturen

- Repository selber bauen:

 - Verzeichnisbaum erstellen

 - RPMs hineinkopieren

 - RPMs signieren

 - Metadaten generieren

 - Metadaten signieren

 - Veröffentlichen

RPMMD/YUM Repo:

```
/. . . /schlomo10
|-- i386
|-- i586
|-- i686
|-- noarch
|-- repodata
`-- x86_64
```

Repositories bauen (Skripte)

~/rpmmacros

```
%_signature gpg
%_gpg_path /root/.gnupg
%_gpg_name Schlomo Schapiro <schlomo@schapiro.org>
%_gpgbin /usr/bin/gpg --use-agent
```

RPMMD/YUM Repo erstellen und signieren:

```
#!/bin/bash
REPO=/media/install/schlomo10
KEY=9e62229e
rm -Rfv $REPO/{repodata,.olddata}/*
rpm --resign -v $REPO/*/*.rpm
createrepo -v $REPO
gpg -a --detach-sign $REPO/repodata/repomd.xml
gpg -a --export $KEY >
$REPO/repodata/repomd.xml.key
```

Beispiel: Postfix Nullclient RPM

(Nicht vollständiges Beispiel)

```
Name:    schlomo-postfix-nullclient
BuildArch:    noarch
Summary:    schlomo: nullclient configuration for postfix
Requires:    postfix
```

```
%files
```

```
%post
```

```
sed -e 's/^POSTFIX_NULLCLIENT.*$/POSTFIX_NULLCLIENT="yes"/' \
    -e 's/^POSTFIX_RELAYHOST.*$/POSTFIX_RELAYHOST="mail.dmz"/' \
    -i /etc/sysconfig/postfix
```

```
SuSEconfig --module postfix
```

```
/etc/init.d/postfix restart
```

```
%verifyscript
```

```
if grep -q 'POSTFIX_NULLCLIENT="yes"' /etc/sysconfig/postfix && grep -q
'POSTFIX_RELAYHOST="mail.dmz"' /etc/sysconfig/postfix ; then
    echo "Postfix configuration validated." 1>&2
    true # return good exit
```

```
else
```

```
    echo "Postfix configuration is not correct. Check /etc/sysconfig/postfix or
reinstall %{name} package." 1>&2
    false # return bad exit
```

```
fi
```

Eigene RPM – SUSE Linux kompatibel

- Updates der Distributionspakete möglich
 - Keine Paketkonflikte auslösen
 - Abhängigkeiten mit Distributionspaketen beachten
 - Mehrere SUSE Linux Versionen unterstützen
 - Sicherheit durch signierte Pakete
-
- Themen auf dem „SUSE Weg“ angehen
 - Softwareverteilung über Repository
 - openSUSE Build Service nutzen (dummy signiert)
 - Pakete lokal signieren und zur Verfügung stellen

Der „SUSE Weg“

- Nicht in der Konfigurationsdatei patchen, sondern:
 - `/etc/sysconfig/XXX + SuSEconfig --module XXX`
 - Includedateien ablegen (z.B. `/etc/apache/conf.d`)
 - Inputdateien nutzen (z.B. `/etc/syslog-ng/syslog-ng.conf.in`)
- `XXX-kmp-YYY` für Kernelmodule
- `udev`, `hal` usw. nutzen wo SUSE es auch nutzt
- Paketdoku in `/usr/share/doc/packages/XXX` lesen:
 - `README`: allgemeine Hinweise
 - `README.SUSE`: SUSE-spezifische Hinweise (**sehr wichtig!**)

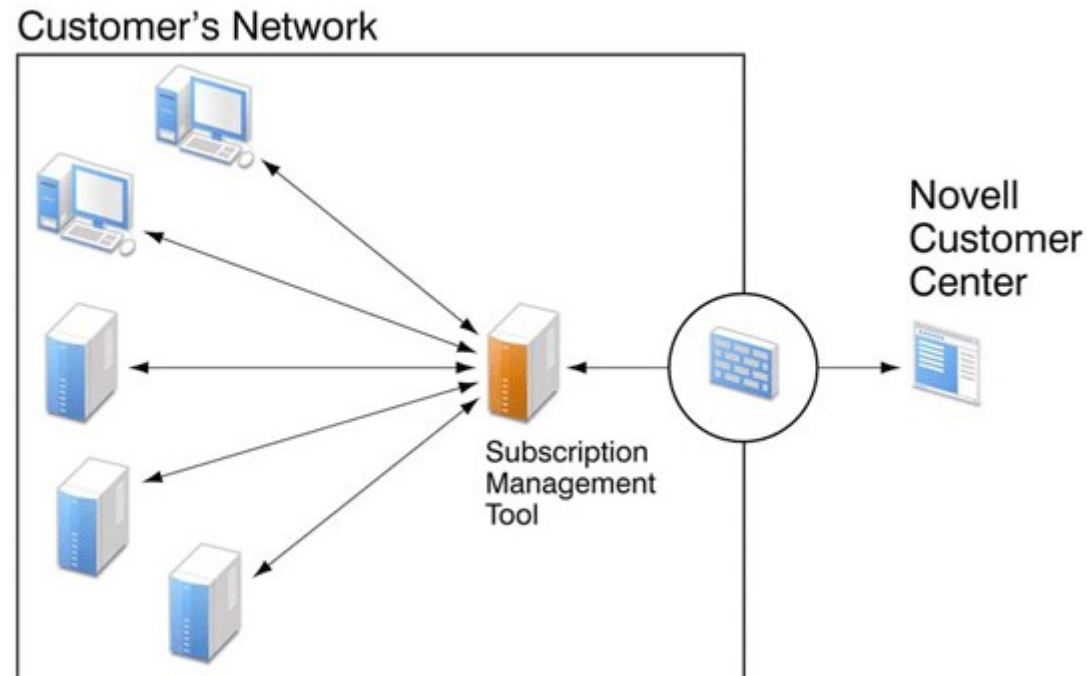
openSUSE Build Service

- build.opensuse.org
- SPEC Datei, Quellen und Patches hochladen
- Pakete werden für SUSE, Red Hat, Mandriva usw. erstellt
- RPMMD Repository für die meisten RPM Systeme
- Pakete auch über software.opensuse.org erreichbar
-
- Lokale Alternative: build, build service lokal installieren

<https://build.opensuse.org/project/show?project=home:sschapiro>

Subscription Management Tool

- Frontend für Novell Customer Center (NCC)
- Für SLE und RHEL
- Lizenzüberwachung
- Offlinemodus
- Lokaler Modus
- Automatisches Zuweisen der richtigen Repositories
- Spiegeln von Repositories
- <http://www.novell.com/linux/smt/>



SMT in der Praxis

- Wichtiges Hilfsmittel
- Nur Standardfunktionen
- Sollte in jedem Netzwerk stehen
- Optimal in Kombination mit einem PXE Installationsserver

- Man muss im NCC aufräumen
- Novell Support hilft kompetent bei Problemen
 - Mirror Credentials funktionieren nicht, sehen nur einen Teil
 - Maschinen werden nicht richtig registriert
- Weiterer Ausbau von Funktionen geplant

Teilautomatisierung mit SMT und RPM

1. Z.B. SLES 10 SP2 installieren

2. SLES im SMT per Skript registrieren:

- System wird im SMT erfasst -> NCC
- Passende Repositories werden aktiviert
 - SLES 10 SP2 Updates
 - Eigene Repositories

**3. Standardkonfiguration mit eigenen RPMs aktivieren
(Metapakete für typische Szenarien nutzen)**

Vollautomatisierte Alternative: AutoYaST

To YaST or not to YaST



Warum YaST?

- **Abstraktion der Systemverwaltung**
 - **Einfache Bedienung für Standardaufgaben**
 - **Einheitliche GUI**
 - **Automatisierbar (AutoYaST)**
 - **Skriptfähig, eigene Module einfach erstellen**
 - **Novell Support**
-
- **YaST ist GUI, /etc/sysconfig und Standardkonfigurationsdateien enthalten die eigentliche Konfiguration**

Persistent Device Names

- Wichtiges Grundprinzip für stabilen Betrieb
 - Netzwerk (bus-pci-..., id-MAC)
 - Datenträger (/dev/disk/by-*)
- Steuerung der Zuordnung per udev
 - SLES: /etc/sysconfig/network/config (PERS*)
 - openSUSE: /etc/udev/rules.d/
- i.d.R. praktisch und sinnvoll, ausser bei VMs
- Hardware in /etc/sysconfig/hardware
 - SLES und openSUSE unterschiedlich
 - Am Besten den SUSE Weg gehen ...

Netzwerkkonfiguration mit YaST

- Bonding (ab SLES10 SP2)
 - NICs „start never“, keine IP
 - Richtigen Typ auswählen
- Bridging (ab SLES10 SP2)
 - NICs „start with system“, IP 0.0.0.0
- /etc/sysconfig/network/ifcfg.template für Beispiele (leider z.T. nicht korrekt bzw. veraltet)
- Bei VM ethX, sonst eth-id-MAC nutzen

Firewall mit YaST

- Standardfirewall
- <http://en.opensuse.org/SuSEfirewall2>
- <http://forgeftp.novell.com/susefirewall2/web/>
- Konfiguration per YaST und `/etc/sysconfig/SuSEfirewall2`
- Vorteile:
 - SUSE konform (Support usw.)
 - Einfach Konfiguration für Standardaufgaben
 - NAT, Masquerading usw. möglich
- Nachteile:
 - Primitiv (kein Antispoofing usw.)
 - Nicht alle Features lassen sich per YaST einstellen, kaum Doku

Paketverwaltung

- YaST packager frontend für zypp
- Rug/zmd: Ist tot, nur noch für ZLM nutzen
- zypper: ab SLES10 SP1 dabei, ab openSUSE 11.0 gut
- PackageKit: ab openSUSE 11.0, nett, nutzt intern zypp
- smart, apt-rpm: Alternativen
- Empfehlung:
 - zypper wenn funktioniert, sonst yast sw_single
 - Zypper hat noch diverse Mängel, reicht aber für Standard aus

Services mit YaST

- Benutzer

- Apache

- DNS

- DHCP

- Samba

- ...

- Man kann, muss aber nicht

- Nur für sehr einfache Umgebungen geeignet

Wichtig: /etc/sysconfig/

- kernel
- displaymanager, windowmanager
- bootloader
- proxy
- postfix, mail
- mdadm
- nfs
- language, keyboard
- ...

openSUSE oder SUSE Enterprise Linux?

Novell®

Gemeinsamkeiten

- **Gemeinsame Codebasis (openSUSE Factory)**
- **Gemeinsames Bugzilla und andere Ressourcen**
- **Gleiches Look & Feel, Bedienung usw.**
- **openSUSE Build Service**

openSUSE

- **Aktuellste Versionen**
- **Grosse Paketauswahl enthalten**
- **Communityprojekt mit starker Steuerung / Kontrolle durch Novell**
- **2 Jahre Updates**
- **Alle 6-8 Monate neues Release**

SUSE Enterprise Linux

- Fork von openSUSE
 - openSUSE 10.1 -> SLES10
 - openSUSE 11.1 -> SLES11
- Reduzierte Paketauswahl, dafür aber mit
 - Hohes Niveau der Qualitätssicherung (Tests)
 - Erweiterte Sicherheitsaudits
 - L2 oder L3 Support für alle Pakete (ab SLES11)
- 5 Jahre Updates & 7 Jahre Security Fixes
- Mindestens 7 Jahre Support, mehr kaufbar
- Zertifizierung für Hardware und Software
- Ausbildungsprogramm und Zertifizierungen

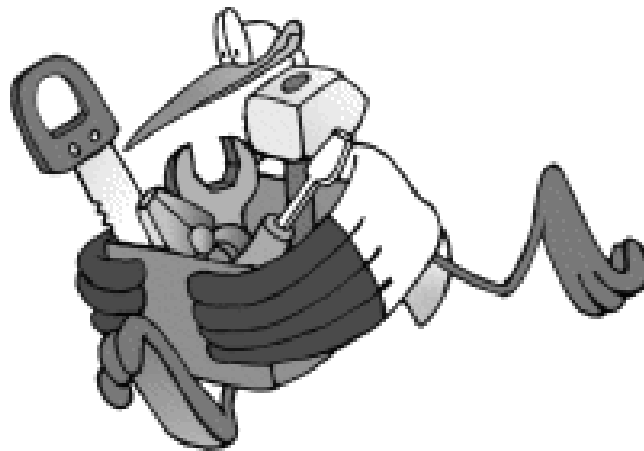
Vorteile SLES

- **Risk Mitigation – Absicherung für den Kunden**
 - **Novell übernimmt Verantwortung für Betriebssicherheit**
 - **Verschiedene Supportmodelle**
 - **Zertifizierung durch Technology Partner**
- **Längerfristige Zyklen entsprechen Abschreibungsfristen**
 - **Betrieb eines Servers mit einer unterstützten Lösung über die gesamte Betriebszeit eines Servers**
- **Konservatives Betriebssystem**
 - **Stabile und erprobte Versionen**
 - **Es funktioniert einfach**

Vorteile durch Support

- **So persönlich wie man möchte**
 - **Websupport, Phone Support, Personal Support Engineer**
 - **Auf Deutsch und auch aus Nürnberg**
- **Relativ direkter Draht zu den Entwicklern**
 - **Kritische Themen werden hocheskaliert**
 - **SUSE Entwickler in Nürnberg**
- **Rückfluss der Lösung in die Distribution (nächstes SP)**
 - 1. PTF Paket, Kunde bestätigt Lösung**
 - 2. Update Paket für alle**

Wie kriege ich schneller meinen Support?



Formale Wege beachten

- **Sofort Ticket aufmachen**
 - **Kritisch: Nur per Telefon**
 - **Sonst: Per Web**
- **Saubere Beschreibung des Problems**
 - **Auf Englisch**
 - **Mit Details**
 - **supportconfig Datei anhängen**
- **„Gutes“ Entitlement benutzen**
- **Sich Zeit für die Bearbeitung des Tickets nehmen**
- **Die Beschäftigung mit dem Support outsourcen**

Beschleunigungsfaktoren

- Per Chat oder Telefon nachhaken
- Sich nicht mit Banalitäten abwimmeln lassen
- Hat der Support Engineer Erfahrung mit dem Thema?
- „meckern hilft“:
 - Bei ausbleibender Reaktion anrufen und vom Dispatch den Fall eskalieren lassen
 - „Did you read my last email? Please respond to the issues!“
- Bei schwierigeren Fällen einen Support Engineer persönlich ansprechen, mit dem man schonmal Erfolg hatte
- Nett bleiben – Support Engineers sind auch nur Menschen
- Hartnäckig bleiben und eine ordentliche Lösung verlangen

Nach dem Supportfall

- **Bewertung ausfüllen – der Support Engineer wird es danken und sich daran erinnern**
- **Nacharbeiten: Doku, TIDs ...**
- **... ist vor dem Supportfall**

Links

SLES - Welche Pakete wie supported werden:

http://support.novell.com/products/server/supported_packages/

Novell Support Lifecycle:

<http://support.novell.com/lifecycle/>

Software Maintenance:

<http://support.novell.com/lifecycle/software-support.html>

SLES - Lebenszyklus und Supportzeiten:

<http://support.novell.com/lifecycle/lcSearchResults.jsp?st=suse+linux+enterprise+server&x=0>

openSUSE - SLES Vergleich:

http://support.novell.com/de-de/products/opensuse/support_overview.html (Support)

<http://www.novell.com/de-de/products/opensuse/comparative.html> (Übersicht)

Fragen & Antworten

Open Source Reklame:

- Linux Disaster Recovery: Relax & Recover (<http://rear.sourceforge.net>)
- VMware VI3 Backup & DR: easyVCB (<http://easvcb.sourceforge.net>)
- OpenVPN Gateway Builder: OGB (<http://ogb.schlomo.schapiro.org>)
- RSYNC BACKUP MADE EASY: RBME (<http://rbme.schlomo.schapiro.org>)

Vorträge:

- <http://www.schapiro.org/schlomo>
- <http://www.probusiness.de>

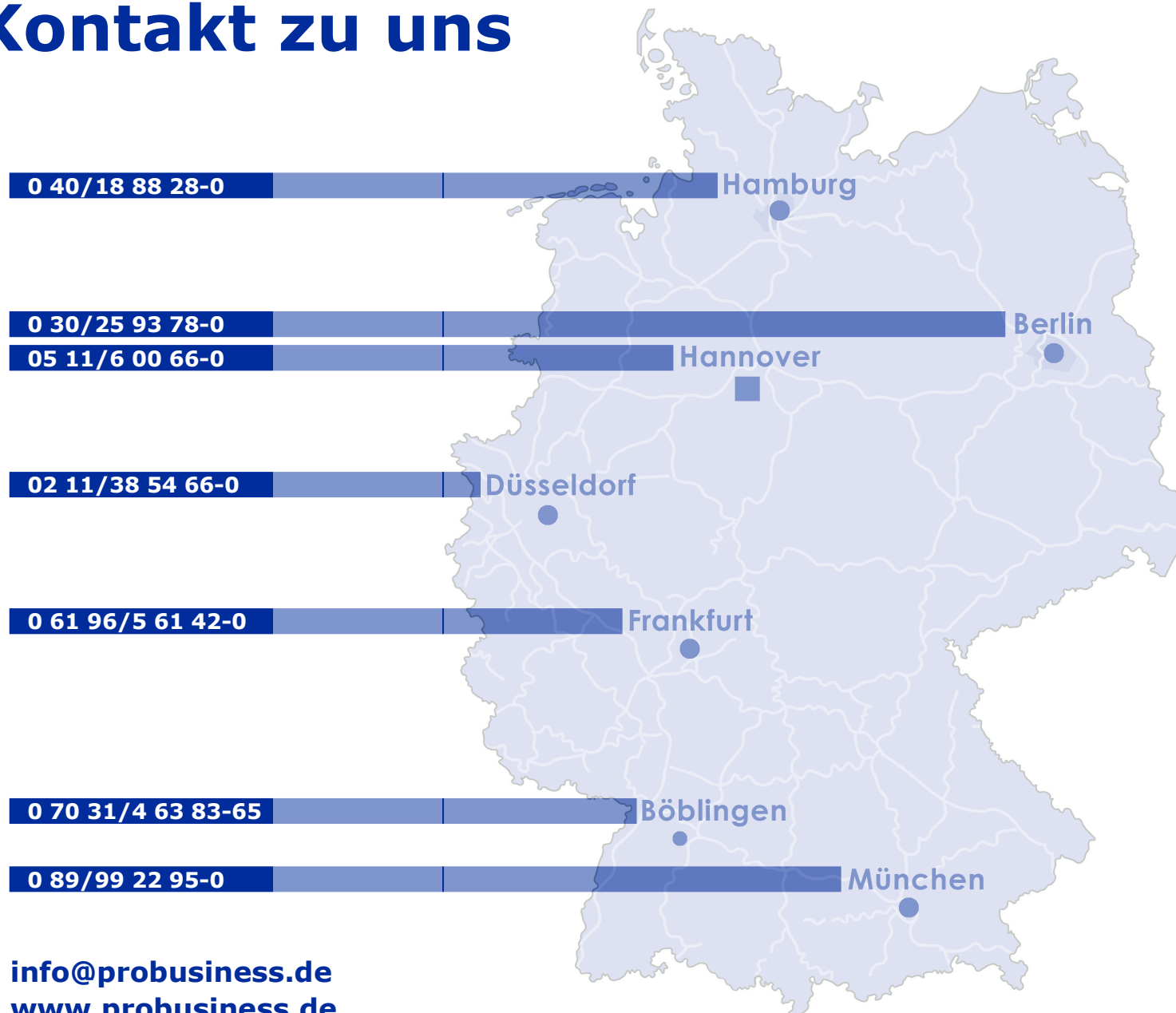


Schlomo Schapiro
Principal Consultant
Leitung Virtualisierung und Open Source

probusiness Berlin AG
Potsdamer Platz 11
D-10785 Berlin

berlin@probusiness.de
+49 30 259378 0

Ihr Kontakt zu uns



info@probusiness.de
www.probusiness.de