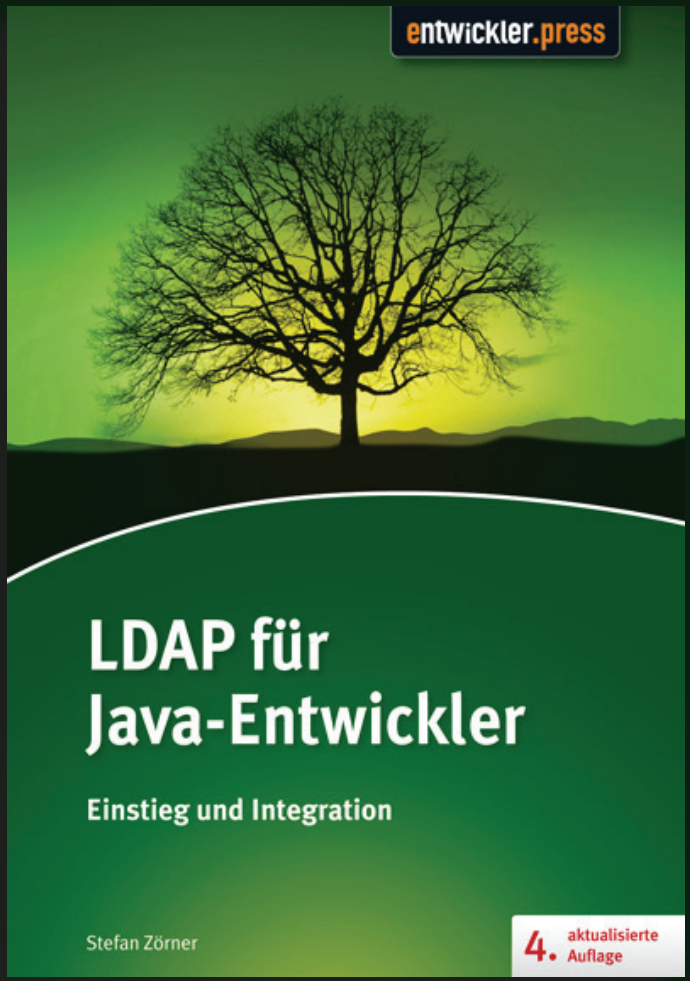
Vielen Dank, dass Sie sich für ein eBook von entwickler.press entschieden haben!

Mehr eBooks und alle Informationen zu unserem Verlagsprogramm finden Sie unter:

www.entwickler-press.de

Viel Spaß,
Ihr entwickler.press-Team



Mit den Pfeiltasten können Sie durch das Dokument navigieren,
mit der ESC-Taste beenden Sie den Vollbildmodus.

Stefan Zörner

LDAP für Java-Entwickler

Einstieg und Integration

entwickler.press

Stefan Zörner
LDAP für Java-Entwickler
Einstieg und Integration

ISBN: 978-3-86802-282-7

Ein Imprint der Software & Support Media GmbH
4. aktualisierte Auflage

Bibliografische Information Der Deutschen Bibliothek

Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über *http://dnb.ddb.de* abrufbar.

Ihr Kontakt zum Verlag und Lektorat:
Software & Support Media GmbH
entwickler.press
Darmstädter Landstraße 108
60598 Frankfurt am Main
Tel.: +49 (0)69 630089-0
Fax: +49 (0)69 630089-89
lektorat@entwickler-press.de
http://www.entwickler-press.de

Projektleitung: Sebastian Burkart
Lektorat: Theresa Vögle
Korrektorat: Frauke Pesch
Satz: Dominique Kalbassi

Inhaltsverzeichnis

Vorwort

Eine LDAP-Einführung für Java-Entwickler. Warum eigentlich?

In vielen Unternehmen und Organisationen haben Verzeichnislösungen zur Speicherung von Informationen eine strategische Bedeutung. Die häufigste Anwendung ist dabei die Verwaltung von Benutzerdaten, Gruppenzugehörigkeiten und Berechtigungen. Insbesondere in diesem Zusammenhang kommen Java-Entwickler regelmäßig in die Verlegenheit, bestehende Verzeichnisdienste in ihre Lösungen einzubinden. Die eigentliche Aufgabe des Projekts besteht in der Regel nicht in dieser Integration, sondern z. B. darin, fachliche Anforderungen in Form einer Anwendung zu realisieren oder ein Content-Management-System (CMS) oder eine Portallösung zu implementieren. Kenntnisse im Bereich Verzeichnisdienste und LDAP (dem Lightweight Directory Access Protocol, einem standardisierten, TCP/IP-basierten Zugriffsprotokoll für solche Dienste) sind im Entwicklerteam regelmäßig nicht ausreichend oder überhaupt nicht vorhanden. Wenn aus diesem Grund innerhalb des Projekts Grundlagenforschung betrieben werden muss, werden Aufwände schwer kalkulierbar und das Vorhaben läuft leicht aus dem Plan. Wer hingegen mit dem nötigen Rüstzeug ausgestattet ist, kann die Komplexität der Integration abschätzen, im Falle der Überforderung der Teamkenntnisse geeignete Maßnahmen ergreifen und sich ansonsten den eigentlichen Zielen widmen.

Ziel und Zielgruppe

In diesem Buch versuche ich, die aus meiner Sicht wesentlichen Kenntnisse zu vermitteln, die ein Java-Entwickler im Zusammenhang mit Verzeichnislösungen und LDAP im Projektalltag benötigt. Neben grundlegenden Inhalten („Was ist eigentlich ein Verzeichnis?“) sind dies vor allem das Ansprechen von Verzeichnissen aus Java mit JNDI und das Einbinden als Benutzerbasis für (Web-)Applikationsserver. Ich behandle dabei die aktuelle Fassung des Standards, LDAP v3.

Demjenigen, dessen Ziel z. B. die Planung, Implementierung und der sichere Betrieb einer unternehmensweiten, verteilten Verzeichnislösung zur Verwaltung zehntausender Benutzer ist, sei ein allgemeiner Titel zu diesen Themen ans Herz gelegt, möglicherweise sogar spezifisch für Ihr konkretes Verzeichnisprodukt, falls die Entscheidung dafür bereits gefallen ist. Falls Sie diese Lösung anschließend in Ihre Java-Anwendung (SE, EE, RCP, …) integrieren wollen, gehören Sie hingegen wieder zur Zielgruppe dieses Buches.

Vierte Auflage

Die erste Auflage dieses Buches erschien im Mai 2004. Damals hatte Jörg Wegener noch Inhalte zum Active Directory beigesteuert, und ich als Hauptautor hatte auch als Herausgeber fungiert. Ursprung war eine Session mit dem Titel „Einführung in LDAP für Java-Entwickler", die ich 2003 auf der JAX Konferenz in Frankfurt/M. gehalten hatte. Der Veranstalter (Software & Support Verlag) und ich selbst waren damals positiv überrascht vom großen Interesse der Teilnehmer. Die Idee, die präsentierten Inhalte als Grundlage für ein Buch zu verwenden, stammte vom Chefredakteur des Java Magazins, Sebastian Meyen.

Bereits ein Jahr nach Erscheinen des Buches zeichnete sich der Abverkauf ab, und der Verlag fragte an, ob ich das Manuskript für eine zweite, aktualisierte Auflage überarbeiten würde. Ich habe das Angebot gerne angenommen. Ebenso wie zu Beginn 2007 für die dritte Auflage.

Auch wenn das Thema LDAP aufgrund der restriktiven Standardisierung recht stabil ist: über die einzelnen Auflagen hinweg hat sich doch einiges getan – insbesondere im Zusammenspiel mit Java. So bereichern ernst zu nehmende, in Java realisierte Server mittlerweile das Angebot an Open-Source-Lösungen. Auch neue Clients stehen zur Verfügung, im Gegensatz etwa zu 2005 gibt es mittlerweile attraktive Eclipse-Plug-ins für LDAP. Skriptsprachen erleben eine Renaissance und beeinflussen auch die Java-Welt. Wie sieht es hier mit der Integration von Verzeichnisdiensten aus? Und nicht zuletzt sorgte die steigende Beliebtheit des Spring Frameworks dafür, dass ich in diesem Buch mittlerweile auch die Frage diskutiere, wie es sich mit LDAP und Spring verhält, und speziell mit Spring Security.

In der nun vorliegenden vierten Auflage sind auch die Auswirkungen des Kaufs von Sun Microsystems durch Oracle eingearbeitet. Der Erwerb wurde im Jahre 2009 angekündigt und 2010 abgeschlossen. Zu den Effekten zählen nicht nur nun ungültige Webadressen oder schlichte Produktumbenennungen, in denen das Wort „Sun" durch „Oracle" ersetzt wurde. Einige Bibliotheken, die Sun vor allem im Kontext mit JNDI zur Verfügung gestellt hat, sind nicht mehr leicht zugänglich; teilweise werden Lösungen nicht weiterentwickelt.

Neben diesen inhaltlichen Ergänzungen und Aktualisierungen motivierten politische Veränderungen die Überarbeitung der Beispieldaten.

Die Beispiele in diesem Buch

Innerhalb des Buches verwende ich ein durchgängiges Beispielverzeichnis. Sie lernen es bereits im ersten Kapitel kennen, inhaltlich geht es um Abgeordnete des Deutschen Bundestages. Die Daten können Sie von der Webseite zum Buch[1] herunterladen und sie z. B. in einen zu Testzwecken aufgesetzten Server importieren. Auf der Seite finden Sie darüber hinaus auch Artikel zu Spezialthemen und auch die Beispielquelltexte zu Kapitel 3, 4, 5

1 *http://www.ldapjava.de*

und 6 und die Ausschnitte aus Beispielkonfigurationen zu Spring Security und Apache Tomcat aus Kapitel 6.

Die Verwendung des Deutschen Bundestages als Fallbeispiel bringt es mit sich, dass die Beispiele im Buch durch das politische Zeitgeschehen veralten. Hierzu zählen vorhersehbare Ereignisse, wie das Ende einer Legislaturperiode und die damit verbundene Bundestagswahl, und unvorhersehbare, wie beispielsweise der Rücktritt eines Bundesministers wegen Plagiatsvorwürfen bei der Doktorarbeit. Während ich das Manuskript 2005 für die zweite Auflage überarbeitete, überraschte mich der damalige Bundeskanzler Gerhard Schröder mit dem Vorhaben einer vorgezogenen Bundestagswahl. Für die Beispiele in der vorliegenden vierten Auflage wurde der 17. Bundestag mit Änderungen bis zum 31.01.2013 als Datenbasis verwendet, mit einer Koalition von CDU/CSU und FDP und Angela Merkel als Bundeskanzlerin. Quelle war die Webseite des Bundestages (Abb. V.1). Auf der Webseite zum Buch werde ich regelmäßig aktualisierte Fassungen der Daten bereitstellen.

Links und Literatur

Links und Literaturhinweise sind jeweils am Ende eines Kapitels versammelt. In diesen Aufstellungen und auch bei Referenzen verwende ich die Notation [x.1], [x.2] … für Hinweise in Kapitel x, was es mir ermöglicht, mich auch in einem späteren Kapitel auf einen Hinweis zu beziehen.

Abbildung V.1: Quelle für die Beispieldaten: Die Webseite des Deutschen Bundestages

Danksagungen

In einem Buchprojekt wie diesem gibt es viele helfende Hände. Folgende Personen haben mich bei der Erstellung der ersten, zweiten, dritten und/oder vierten Auflage besonders unterstützt und ich danke ihnen sehr herzlich dafür:

Christiane Auf

Manuel Burckas

Sebastian Burkart

Nicole Bechtel

Michael Kloss

Christine Koppelt

Wolfgang Korn

Dr. Thomas Lüer

Jan-Piet Mens

Sebastian Meyen

Jutta Werz

Anna Elisabeth Zörner

Zu den vergangenen Auflagen habe ich zahlreiche Rückmeldungen von Lesern erhalten, für die ich mich an dieser Stelle ebenfalls bedanken möchte. Ihre Hinweise haben mich dabei unterstützt, die Stärken dieses Buches, das mittlerweile auf eine für IT-Verhältnisse lange Geschichte zurückblickt, zu bewahren, und die Schwächen auszumerzen. Ich hoffe, dass Java-Entwickler, die sich mit LDAP konfrontiert sehen, dieses Buch weiterhin als nützliches Werkzeug gern zur Hand nehmen.

Stefan Zörner (*stefan@ldapjava.de*) im April 2013.

1 LDAP – Ein erster Kontakt

Ganz allgemein dienen Verzeichnisse (englisch „Directories“) zum Ablegen und Abfragen von Informationen aller Art. Dieses Buch geht speziell auf Directories ein, die LDAP, das Lightweight Directory Access Protocol, für Zugriffe unterstützen. Vor allem zeigt es, wie Sie LDAP-fähige Produkte aus Java heraus ansprechen können. Die Einbindung von Verzeichnissen als Benutzerdatenbasis in (Web-)Applikationsservern ist eine wichtige Anwendung im Java-EE-Umfeld; auch dieses Thema behandle ich im Verlauf des Buches. Zunächst jedoch werden Sie in diesem Kapitel mit einem konkreten Verzeichnis Kontakt aufnehmen, um einen ersten Eindruck zu erhalten, wie in einem derartigen Datenspeicher Informationen abgelegt sind. Den Abschluss des Kapitels bildet eine kurze Vorstellung des Aufbaus des restlichen Buches.

Wo werden Verzeichnisse heute eingesetzt?

Auf den ersten Blick stehen Verzeichnisse als Datenspeicher ähnlich wie andere NoSQL-Lösungen[1] in Konkurrenz zu den etablierten relationalen Datenbanken. Tatsächlich werden in den meisten großen Unternehmen und Organisationen jedoch sowohl Verzeichnisse als auch relationale Datenbanken eingesetzt.

Eine wesentliche Stärke vieler Verzeichnislösungen liegt in der Optimierung auf Lese- und Suchoperationen. Das macht die Produkte besonders als Speicher für Daten interessant, die deutlich häufiger abgefragt als manipuliert werden. Weiterhin bieten die Produkte eine gute Unterstützung bei der Implementierung verteilter Lösungen. Konsequenterweise ist der häufigste Einsatz von Verzeichnissen die unternehmensweite Verwaltung von Benutzerdaten und Berechtigungen, insbesondere bei Organisationen mit vielen Standorten. Verzeichnisse bilden einen elementaren Bestandteil von Single-Sign-on- und PKI-Lösungen[2] und gehen nicht selten über die Verwaltung von Mitarbeiterdaten hinaus. Auch Kunden und Partner greifen heute auf Unternehmensdaten zu; sie und ihre Berechtigungen müssen ebenfalls verwaltet und geprüft werden. Viele Unternehmen legen Beschreibungen ihres Besitzes und ihrer Organisationsstruktur in einem Verzeichnis ab (z. B. Hard- und Software, Niederlassungen, Abteilungen). Eine gängige Anwendung ist auch die Speicherung von Konfigurationsdaten für Applikationen als Alternative zu Property

1 Unter dem Begriff NoSQL, oft „Not only SQL“ ausgesprochen, versammeln sich Alternativen zu den omnipräsenten relationalen Datenbanken, auf die mit SQL zugegriffen wird. Eigentlich ist LDAP kein klassischer NoSQL-Vertreter. LDAP ist älter als die NoSQL-Bewegung und schon lange verbreitet.

2 PKI steht für Public-Key Infrastructure. Eine solche umfasst in der Regel die Ausgabe digitaler Zertifikate an Benutzer, welche in einem Verzeichnis verwaltet werden.

Files. Die Speicherung benutzerbezogener Daten wird in diesem Zusammenhang gerne zur Personalisierung von Anwendungen eingesetzt, etwa in Portallösungen.

1.1 Requisiten

Was brauchen Sie, um loszulegen? Die TCP/IP-basierte Kommunikation mit Verzeichnissen erfolgt nach dem Client-Server-Modell, LDAP ist ein standardisiertes Protokoll dazu. Um die Inhalte des Buches praktisch nachvollziehen zu können, benötigen Sie daher Zugriff auf einen LDAP-Server. Dazu gibt es folgende Optionen:

- Installation eines eigenen Servers zu Testzwecken
- Zugriff auf ein bestehendes Verzeichnis in Ihrem Unternehmen
- Verwendung eines öffentlichen Verzeichnisses im Internet

Der jeweilige Vorbereitungsaufwand für diese Wahlmöglichkeiten nimmt in der Reihenfolge der Auflistung ab, der Spielraum allerdings auch. Wenn Sie umfassende Erfahrungen in diesem Thema sammeln wollen, sei Ihnen daher das Aufsetzen eines eigenen LDAP-Servers zu Testzwecken ans Herz gelegt.

Installation eines eigenen Servers

Da alle gängigen Verzeichnisprodukte LDAP unterstützen, haben Sie zahlreiche Softwarelösungen (freie und kommerzielle) zur Auswahl. Hier soll keine eindeutige Produktempfehlung ausgesprochen und es sollen insbesondere keine Installationsanleitungen reproduziert werden, die Sie ohnehin stets aktuell gemeinsam mit der Software erhalten. Die folgenden Hinweise dienen lediglich als Orientierungshilfe. Am Ende des Kapitels finden Sie Links zu weiteren Produkten.

Falls Sie im Unix-Umfeld zu Hause sind und den Server z. B. unter Linux betreiben wollen, ist OpenLDAP [1.17] recht verbreitet. Das Projekt stellt einen bewährten LDAP-Server als Open-Source-Lösung bereit. Eine weitere Open-Source-Option, die ihre Wurzeln allerdings in einem kommerziellen Produkt hat, stellt insbesondere für Linux der 389 Directory Server [1.14] dar.

Unter Windows ist Active Directory [1.23] zwar eine naheliegende Wahl, jedoch setzt es Windows 2000 Server aufwärts voraus und entzieht sich so ggf. einer Testinstallation ohne großen Aufwand. Mit ADAM (Active Directory Application Mode), später umbenannt in Active Directory Lightweight Directory Services (AD LDS) [1.24] stellt Microsoft eine schlankere Option auch für Workstations bereit. OpenLDAP läuft prinzipiell auch auf Windows. Quellen für geeignete Installationspakete habe ich auf der Webseite zum Buch bereitgestellt und halte sie dort aktuell. Das OpenLDAP-Projekt stellt keine Binärdistribution bereit.

Falls Sie sich für eine freie LDAP-Server-Implementierung interessieren, die vollständig in Java realisiert ist und sich so auf praktisch allen Hard- und Softwareplattformen

gleichförmig darstellt, können Sie z. B. auf Apache Directory Server zurückgreifen. Das Directory-Projekt [1.15] stellt native Installer u. a. für Linux, MacOS und Windows bereit, mit denen sich die Lösung recht leicht in Betrieb nehmen lässt.

Mehr zu diesen Servern erfahren Sie im zweiten Kapitel.

Das Beispielverzeichnis des Buches

In diesem und in den folgenden Kapiteln verwende ich für Beispiele durchgängig ein Verzeichnis, das Daten des Deutschen Bundestages zum Inhalt hat, die dessen Webauftritt [1.1] entnommen wurden. Das Verzeichnis enthält Informationen zu allen Abgeordneten aus den letzten vier Legislaturperioden[3], sowohl textuelle (z. B. der Name) als auch binäre (ein Bild des Abgeordneten im JPG-Format).

Wenn Sie einen eigenen Server aufgesetzt haben, können Sie die Daten des Beispiels in Ihr Verzeichnis importieren. Auf der Webseite zum Buch [1.2] finden Sie diese als Download im Datenaustauschformat LDIF (mehr dazu im zweiten Kapitel). Kurzanleitungen zum Einspielen in gängige Verzeichnislösungen (u. a. Apache, OpenLDAP und 389 Directory Server) stehen ebenfalls für Sie bereit. Abbildung 1.1 vermittelt einen ersten Eindruck des Beispiels.

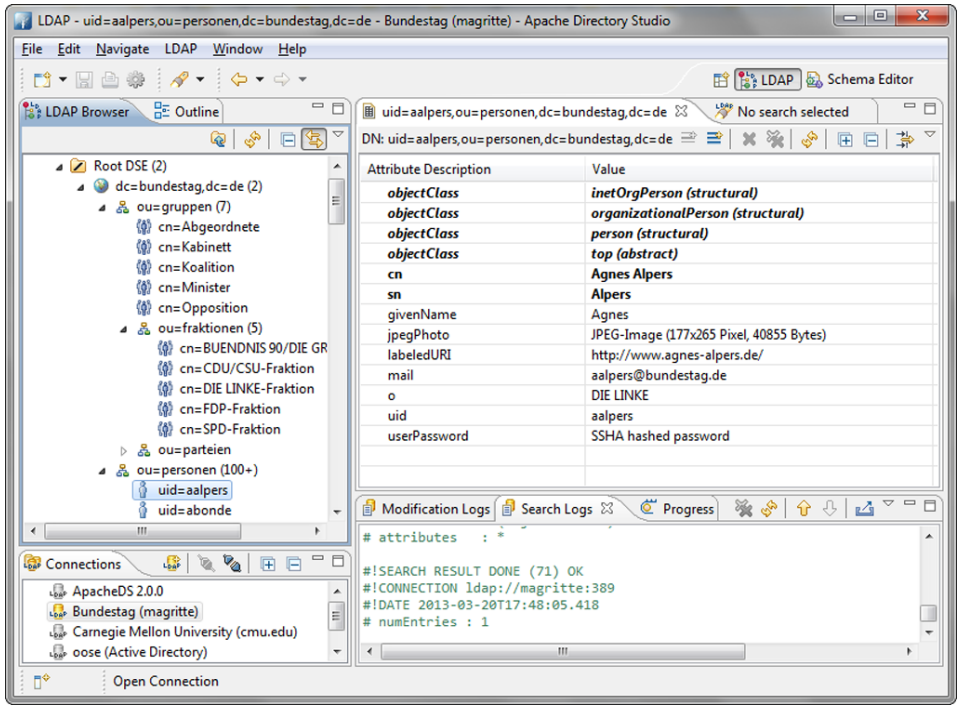


Abbildung 1.1: Das Beispielverzeichnis im Apache Directory Studio

3 Der 14. Deutsche Bundestag wurde am 27.09.1998, der 15. am 22.09.2002, der 16. am 18.09.2005 und der 17. am 27.09.2009 gewählt.

1.2 Kontakt aufnehmen

Im zweiten Kapitel gehe ich genauer auf LDAP ein. Für den Moment genügt es zu verstehen, dass alle gängigen Verzeichnisprodukte dieses Protokoll unterstützen. Sie können daher z. B. einen grafischen LDAP-Client verwenden und sich mit einem beliebigen LDAP-fähigen Server verbinden, egal von welchem Hersteller und auf welcher Hard- und Softwareplattform.

Dieser Ansatz dient hier im Kapitel vorrangig der Veranschaulichung, ist aber bei einer Verzeichnisintegration generell ein guter Startpunkt. Wenn man ein Verzeichnis in eine Softwarelösung einbinden will, erkundet man es meiner Erfahrung nach am besten zuvor auf diese Weise, um die Verbindung zu prüfen und die Struktur des Verzeichnisses und die Form der Einträge kennen zu lernen.

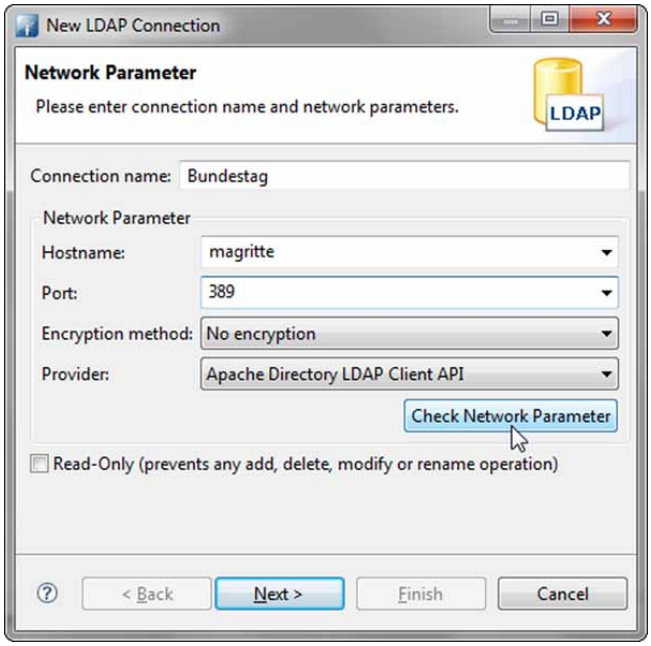


Abbildung 1.2: Ein neues Verbindungsprofil anlegen (Apache Directory Studio)

Als Client wird im Folgenden „Apache Directory Studio“ [1.3] verwendet. Als Eclipse-Plug-in läuft es auf verschiedenen Plattformen und erlaubt u. a. das Lesen, Anlegen und Manipulieren von Einträgen in einem Verzeichnis. Es kann auch als RCP-Anwendung[4] außerhalb der Entwicklungsumgebung installiert werden. Alternative LDAP-Clients finden sich am Ende des Kapitels. Sie haben vergleichbare Funktionalität, nicht alle lassen

4 RCP steht für Rich Client Platform. Hierbei handelt es sich um den Ansatz, beliebige Applikationen mit Eclipse-Technologie zu realisieren (nicht nur Entwicklungswerkzeuge).

allerdings das Editieren im Verzeichnis zu. Auch weitere frei verfügbare, reine Java-Lösungen sind darunter – alternativ auch mit Swing-basierter Oberfläche. Für das Nachvollziehen der folgenden Beispiele installieren Sie idealerweise nun eine der Lösungen.

Nach Installation und Start des Directory Studios muss zunächst eine neue Verbindung angelegt werden (in der Connection View oder im Menü: LDAP | New Connection …). Dazu müssen zumindest der Hostname des Servers und der Port (Standard ist 389) bekannt sein (Abb. 1.2 zeigt den ersten Schritt des Verbindungsassistenten). Für die Authentifizierung kann in vielen Fällen für einen Erstkontakt „anonym" gewählt werden. Eine weitere wichtige Angabe zum Verbindungsaufbau ist der so genannte Base DN (Basisname). DN steht dabei für Distinguished Name und bezeichnet einen Knoten innerhalb der hierarchischen Struktur eines Verzeichnisses eindeutig. Der Base DN bildet den Einstiegsknoten für eine Verbindung. Bei modernen Verzeichnissen können viele Browser mögliche Basisnamen selbst ermitteln („Fetch DNs", „Basis-DNs abrufen"). Andere Tools erwarten die gleichen Angaben zur Konfiguration, Abbildung 1.3 zeigt zur Illustration den Dialog zum Verbindungsaufbau alternativ mit Softerra LDAP Browser [1.6].

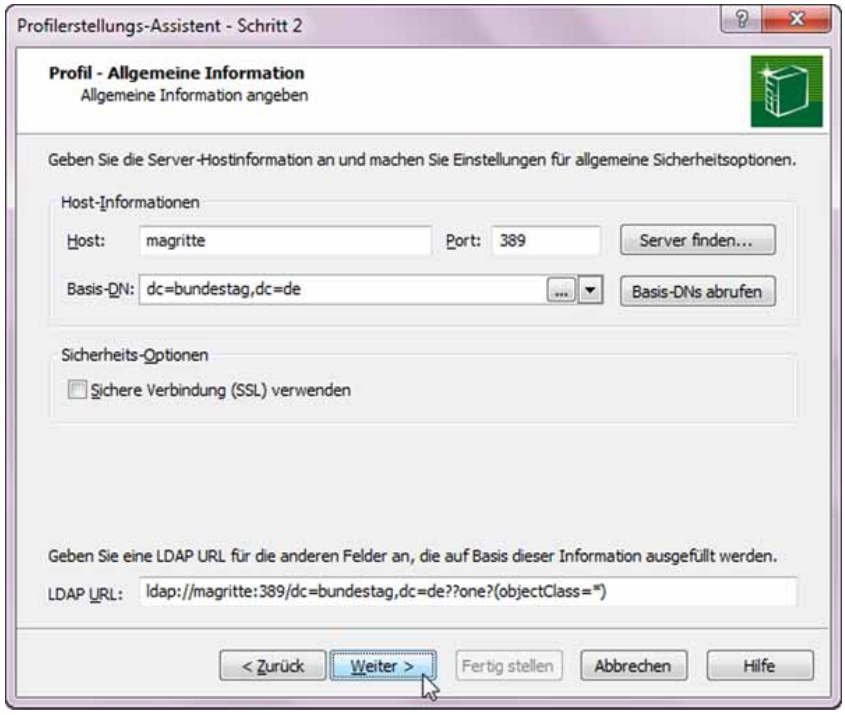


Abbildung 1.3: Ein neues Verbindungsprofil anlegen (Softerra LDAP Browser)

Je nachdem, für welche Möglichkeit Sie sich für Ihren ersten Zugriff auf ein Verzeichnis entschieden haben (eigener Server, vorhandenes Verzeichnis im Unternehmen, öffentli-

ches Verzeichnis im Internet), geben Sie die Daten zum Verbindungsaufbau an. Wenn Sie einen eigenen Server zu Testzwecken installiert haben, werden Sie Hostnamen und Port kennen. Die Beispiele dieses Buches verwenden durchgängig einen Testrechner *magritte* (Port 389) mit Base DN *dc=bundestag,dc=de*. Welches Verzeichnisprodukt konkret verwendet wurde, ist in der Regel unerheblich. Tatsächlich habe ich die Beispiele mit einer ganzen Reihe unterschiedlicher Lösungen getestet.

Hostname	Port	Base DN
db.debian.org	389	dc=debian,dc=org
x500.bund.de	389	o=Bund,c=DE
directory.d-trust.de	389	c=de
ldap.andrew.cmu.edu	389	dc=cmu,dc=edu

Tabelle 1.1: Verbindungsdaten einiger öffentlicher LDAP-Server

Über das Internet sind zahlreiche Verzeichnisse öffentlich erreichbar, die Sie ebenfalls für die ersten Schritte inspizieren können. Tabelle 1.1 listet einige mit den notwendigen Informationen auf. Am Ende des Kapitels finden Sie Informationen zu weiteren öffentlichen Verzeichnissen.

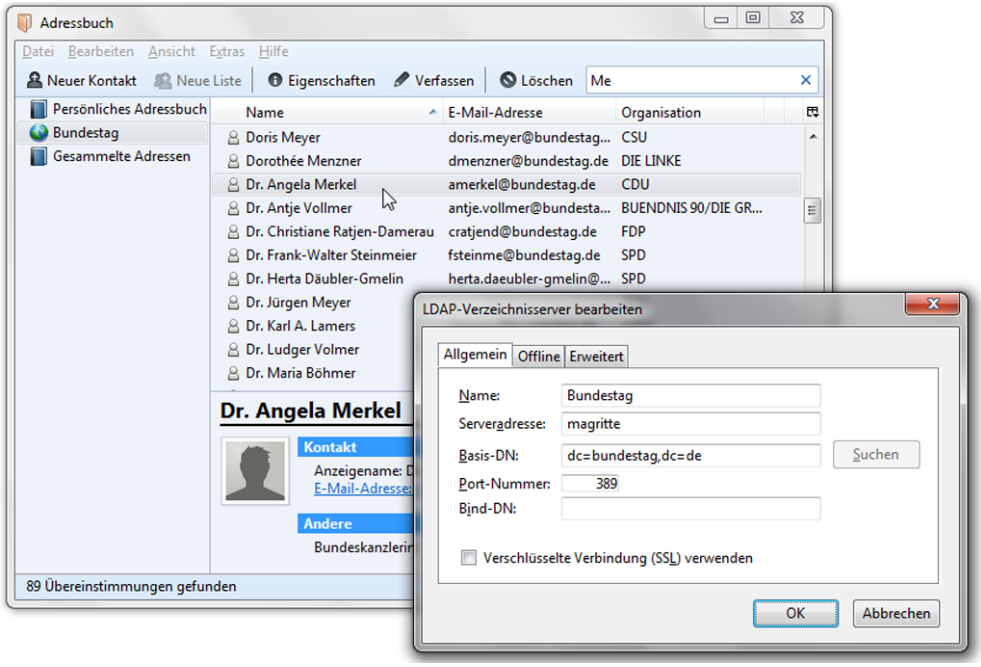


Abbildung 1.4: Das Beispielverzeichnis, eingebunden als Adressbuch (Mozilla Thunderbird)

Wahrscheinlich haben Sie neben diesen öffentlichen Verzeichnissen in Ihrem Unternehmen oder Projektumfeld auch Zugriff auf ein firmeneigenes LDAP-fähiges Softwareprodukt. Vielleicht benutzen Sie es täglich, ohne es zu wissen (z. B. durch das unternehmensweite Adressbuch, zur Illustration siehe Abbildung 1.4). Möglicherweise hat der Wunsch, genau dieses Verzeichnis in Ihre Java-Lösung zu integrieren, Sie veranlasst, dieses Buch zu kaufen. In jedem Fall ist es sicherlich ein attraktives erstes Beispiel für Sie, da Sie bekannte Strukturen und ggf. auch Daten über sich und Ihre Kollegen wiederfinden können. Fragen Sie Ihre Netzwerkadministratoren freundlich nach den vorhandenen Zugriffsmöglichkeiten und Verbindungsdaten.

1.3 Fallstricke

Falls der Versuch des Verbindungsaufbaus zum Verzeichnis Ihrer Wahl fehlschlägt („Can't connect to LDAP Server"), kann dies verschiedene Ursachen haben.

Wenn Sie sich in einem Firmennetz bewegen, kann der Zugriff auf einen externen LDAP-Server im Internet insbesondere durch eine oder mehrere Firewalls versperrt sein. Möglicherweise dürfen Sie lediglich HTTP- und FTP-Dienste des Internets über einen Proxy-Server Ihres Unternehmens nutzen. Die Administratoren werden aus Sicherheitsaspekten vorherrschende Gegebenheiten vermutlich nicht ändern, nur um Ihnen Experimente mit LDAP zu ermöglichen. Als Auswege kommen daher lediglich die Verwendung eines firmeneigenen Verzeichnisses oder die Installation eines eigenen LDAP-Servers zu Testzwecken in Frage.

Auf allen in Tabelle 1.1 aufgeführten Servern ist eine anonyme Anmeldung möglich, d. h., Sie brauchen keinen Benutzer und kein Kennwort anzugeben, um eine Verbindung herzustellen. Ein Verzeichnis in Ihrem Unternehmen kann jedoch auch so konfiguriert sein, dass Zugriffe lediglich nach einer Authentifizierung möglich sind. Ein Active Directory lässt beispielsweise für anonyme Nutzer standardmäßig nur einen sehr eingeschränkten, lesenden Zugriff zu. Wer mehr sehen will, muss sich explizit anmelden. Abbildung 1.5 zeigt den Dialog zur Einstellung von Benutzer und Kennwort in einem Verbindungsprofil in Directory Studio. Für ein bestehendes Profil können Sie die Daten über das Kontextmenü und den Befehl *Properties* ändern, eine Angabe für neue Verbindungen ist in diesem Werkzeug im Assistenten in Schritt 2 alternativ zu „anonym" möglich.

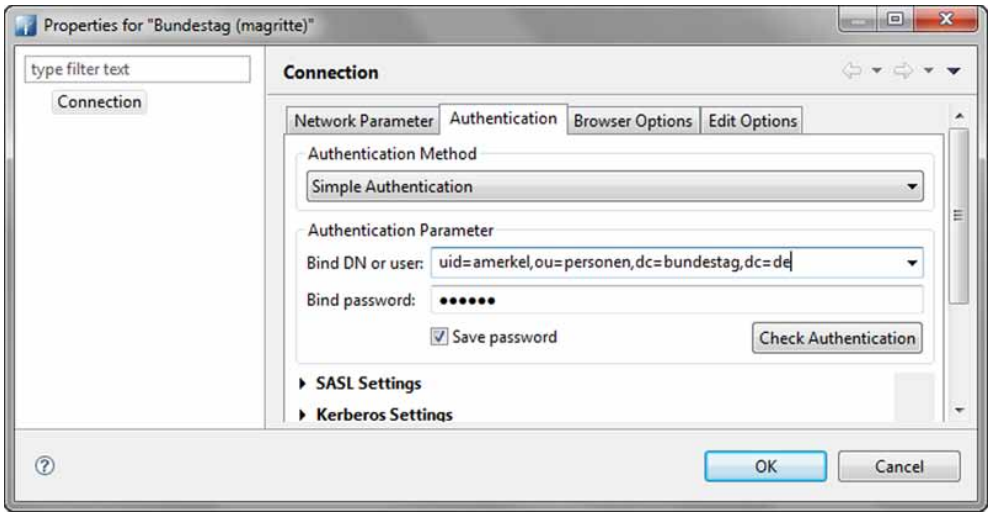


Abbildung 1.5: Anmeldeinformationen bei Apache Directory Studio

Normalerweise muss für den Benutzer der so genannte Distinguished Name (DN) angegeben werden. Falls Sie diesen für Ihre Kennung im Unternehmen nicht kennen, hilft Ihnen Ihr Verzeichnisadministrator sicherlich weiter. Im Fall eines Active Directory können Sie alternativ zum User DN die Zeichenkette <*domain*>\<*user*> verwenden, in der Sie Ihren Domänennamen (<*domain*>) und ihre Anmeldekennung (<*user*>) einsetzen. Die Schreibweise <*user*>@<*DNS-Name*> (z. B. *amerkel@bundestag.de*) wird ebenfalls vom Active Directory akzeptiert. Für den Verbindungsaufbau verwenden Sie das gleiche Kennwort wie beim Anmelden am Rechner.

1.4 Ein wenig Stöbern

Nach erfolgreichem Verbindungsaufbau können Sie das gewählte Directory mit dem LDAP-Browser durchstöbern. Der allgemeine Aufbau von Verzeichnissen, die mit LDAP angesprochen werden können, ist stets gleich. Daten werden in so genannten Einträgen gespeichert, deren Gesamtmenge eine Baumstruktur bildet. Diese Struktur wird in vielen Browseroberflächen wie bei einer grafischen Dateisystemdarstellung visualisiert, in Abbildung 1.1 im linken Bereich (View *LDAP Browser*) zu sehen. Wählt man dort einen speziellen Eintrag aus, werden im rechten Bereich der Oberfläche (View *Entry Editor*) die so genannten Attribute zu einem Eintrag angezeigt. Diese Schlüssel/Wertpaare bilden den eigentlichen Eintrag und enthalten sämtliche gespeicherten Informationen. Unter den Attributen jedes Eintrags ist eins besonders ausgezeichnet. Es legt den relativen Namen innerhalb des übergeordneten Knotens fest und ist in dieser Ebene eindeutig (Re-

lative Distinguished Name, RDN).[5] Im Beispiel-Screenshot in Abbildung 1.1 ist dies für den ausgewählten Eintrag der Abgeordneten Agnes Alpers das Attribut *uid* (User-ID) mit dem Wert *aalpers* (links in der Baumdarstellung geschrieben als *uid=aalpers*). Wandert man im Verzeichnis ausgehend von einem Eintrag schrittweise im Baum nach oben und hängt diese relativen Namen durch Kommas separiert aneinander, entsteht ein innerhalb des Verzeichnisses eindeutiger Name (Distinguished Name, DN). Unter diesem ist der Eintrag direkt ansprechbar, der hier verwendete LDAP-Browser stellt ihn bei ausgewähltem Eintrag in der Werkzeugleiste der Editor-View dar (in Abbildung 1.1 ist dies *uid=aalpers,ou=personen,dc=bundestag,dc=de*).

Grundsätzlich lassen sich in Verzeichnissen alle nur erdenklichen Daten verwalten. Häufig werden Sie in öffentlichen und firmeneigenen Verzeichnissen Personen mit zugehörigen Kontaktdaten und Berechtigungen finden, so auch in unserem Bundestag. In einem Verzeichnis in Tabelle 1.1, nämlich Debian, sind beispielsweise auch Rechner und deren Eigenschaften (Verantwortlicher, Hard- und Softwareausstattung) gespeichert. Um was es sich bei einem Eintrag handelt, wird durch so genannte Objektklassen festgelegt, die sich anhand eines bestimmten Attributs für jeden Eintrag ablesen lassen. Diese Klassen legen wiederum fest, welche Attribute ein Eintrag haben kann und welche er sogar haben muss.

Jeder Eintrag, auf den Sie beim Stöbern im Verzeichnis treffen, genügt diesen allgemeinen Regeln. Vielleicht haben Sie beim Erkunden des Verzeichnisses bereits die Daten entdeckt, die Sie in Ihre Java-Applikation integrieren wollen. Vielleicht planen Sie aber auch, ein Verzeichnis als Datenspeicher für Ihre Java-Anwendungen zu verwenden, d. h., Sie wollen selbst Daten dort ablegen. In jedem Falle sind Sie hoffentlich ein wenig neugierig geworden auf das, was folgt.

1.5 Der Aufbau dieses Buches

Das sich anschließende zweite Kapitel ist eine kurze Einführung in die Welt der Verzeichnisse und in das Informationsmodell, das Verzeichnissen zugrunde liegt. Weiterhin beschreibt es die Operationen, die mithilfe von LDAP auf Verzeichnissen ausgeführt werden können, und es geht auf fortgeschrittenere Themen wie verteilte Verzeichnisse kurz ein.

Das eigentliche Ziel dieses Buches ist es, Ihnen als Entwickler aufzuzeigen, wie Sie LDAP-fähige Verzeichnisse in Java-Anwendungen einbinden können. Dazu stelle ich im dritten Kapitel verschiedene Optionen vor. Auf die Wichtigste, nämlich den Zugriff mit JNDI-Mitteln, geht das vierte Kapitel detaillierter ein. Im Anschluss sind Sie in der Lage, die im zweiten Kapitel kennen gelernten Operationen (z. B. das Suchen nach Einträgen im Verzeichnis) aus Java heraus auszuführen. Das fünfte Kapitel geht auf spezielle Anwendun-

5 Formal darf ein RDN auch aus mehreren Attributwerten des Eintrags gebildet werden („Multi-valued RDN“). In der Praxis wird das sehr selten verwendet.

gen und Fragestellungen rund um LDAP ein, die regelmäßig in Java-Projekten auftreten. Konkret geht es um Stolpersteine im Zusammenspiel mit Microsofts Active Directory, die Integration mit dem Spring Framework und das Verfassen von Skripten gegen LDAP-Server am Beispiel Groovy.

Verzeichnislösungen werden in Unternehmen bevorzugt als zentrale Benutzerverwaltung eingesetzt. Das gilt für das Active Directory ebenso wie für andere Softwareprodukte. Aus diesem Grund stehen Projekte häufig vor der Aufgabe, die Authentifizierung und Autorisierung von Java-EE-Anwendungen auf das unternehmensweite Verzeichnis (oder ein im Rahmen des Vorhabens zu implementierendes) abzustützen. Anstatt das Rad mithilfe der in Kapitel 3 und 4 kennen gelernten Mittel neu zu erfinden, kann man auf deklarative Techniken zurückgreifen, wie sie z. B. die Servlet-Spezifikation vorschreibt. Gängige Applikationsserver unterstützen hierzu LDAP-fähige Verzeichnisse „Out of the Box". Hinweise zu diesem Themenkreis bilden den Inhalt des sechsten Kapitels, das auch die Integration von Verzeichnisdiensten in Spring Security diskutiert.

Das siebte und letzte Kapitel wagt einen kurzen Ausblick, wohin die Reise mit Java und LDAP geht. Wie werden sich Produkte in Zukunft positionieren, und hat die Java-Welt im Bereich der Integration mit Java SE 7 Stabilität erreicht? Und da es sich bei diesem Buch um eine Einführung in die Welt der Verzeichnisse handelt, finden Sie hier auch Hinweise, wo Sie weiterführende Information rund um das Thema Verzeichnisse und LDAP erhalten.

1.6 Links und Literatur zu diesem Kapitel

[1.1] Deutscher Bundestag, *http://www.bundestag.de*

[1.2] Webseite zum Buch, *http://www.ldapjava.de*

1.6.1 LDAP-Browser und -Editoren

Hier finden Sie eine kleine Auswahl interessanter LDAP-Werkzeuge für unterschiedliche Plattformen versammelt. Sie sind allesamt unabhängig von einer konkreten LDAP-Serverimplementierung.

Apache Directory Studio (Open Source)

Eclipse-basierte LDAP-Tools (Browser, Editor, Schema-Browser, LDIF-Editor, …), die sowohl als Plug-in in der IDE, als auch standalone als RCP-Anwendung betrieben werden können (Abb. 1.1). Die Software wird vom Apache Directory Project als grafischer Client für den eigenen Server entwickelt, ist aber problemlos für alle gängigen LDAP-Server einsetzbar. Verfügbar für Linux, Windows, MacOS.

[1.3] *http://directory.apache.org/studio/*

JXplorer (Open Source)

Ursprünglich von Computer Associates (CA) stammender LDAP-Browser (Abb. 1.6) mit reichhaltigem Funktionsumfang (Browser, Editor, Schema-Browser, …). JXplorer ist eine reine Java-Lösung mit Swing-Oberfläche, es werden zahlreiche Plattformen mit nativen Installern unterstützt.

[1.4] *http://www.jxplorer.org*

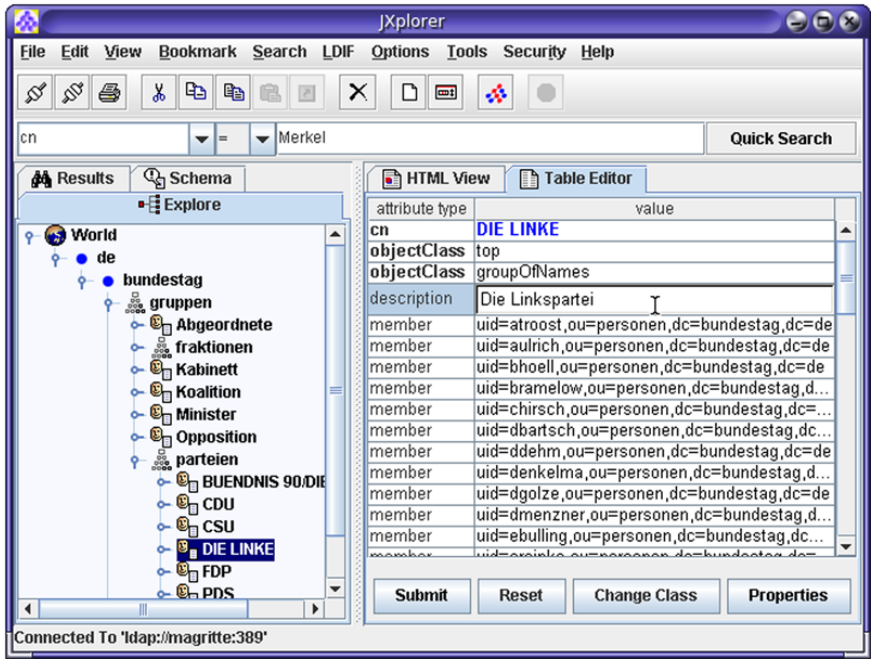


Abbildung 1.6: Der LDAP-Browser JXplorer (hier unter Unix/GNOME-Desktop)

Softerra LDAP Administrator (kommerziell)

Professionelles Werkzeug für die Windows-Plattform. Schema-Browser, Template- und Schema-basiertes Anlegen neuer Einträge und vieles mehr.

[1.5] *http://www.ldapadministrator.com*

Softerra LDAP Browser (Freeware)

Der „kleine Bruder“ des Administrators (Abb. 1.7). Erlaubt lediglich das Browsen von Verzeichnissen und Schemata, nicht aber Änderungen oder LDIF-Imports. Zu empfehlen, wenn Sie die Einschränkung auf die Windows-Plattform nicht stört und Sie nur lesend zugreifen wollen.

[1.6] *http://www.ldapadministrator.com/softerra-ldap-browser.htm*

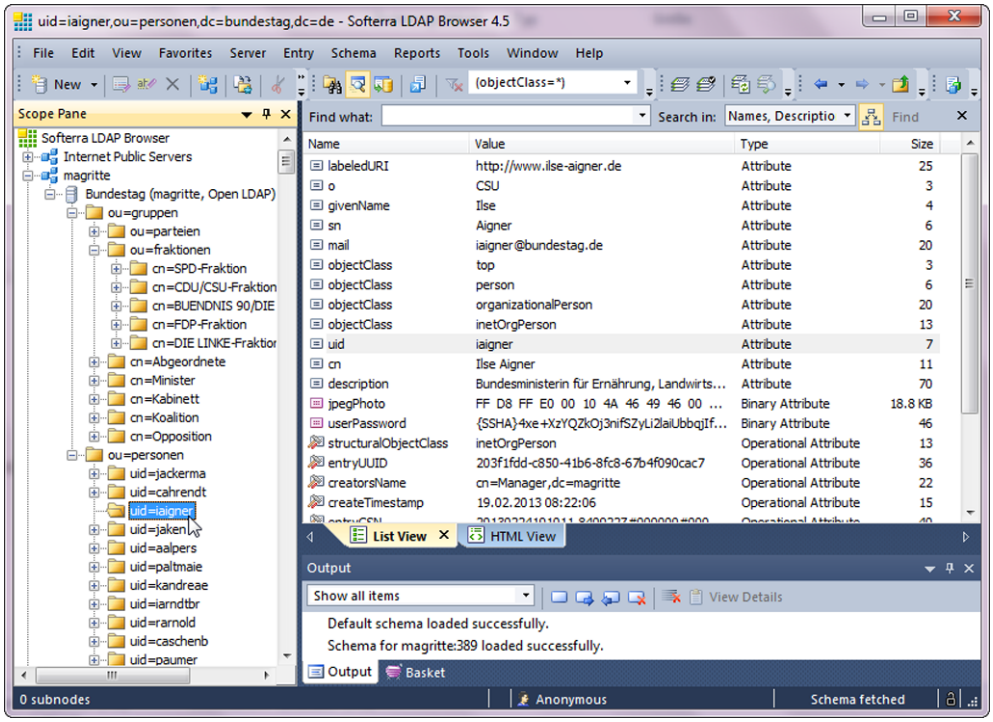


Abbildung 1.7: Softerra LDAP Browser

phpLDAPadmin (Open Source)

Webbasiertes Administrationswerkzeug zum Zugriff auf beliebige LDAP-Server, realisiert in PHP, zum Betrieb z. B. innerhalb von Apache HTTP Server.

[1.7] *http://phpldapadmin.sourceforge.net*

web2ldap (Open Source)

Webbasierter Client zum Zugriff auf beliebige LDAP-Server, realisiert in Python. Großer Funktionsumfang inklusive Schema-Browser.

[1.8] *http://www.web2ldap.de*

LDAP Admin (Open Source)

Grafischer LDAP-Browser und -Editor ausschließlich für Windows-Systeme. Neben der Grundfunktionalität sind u. a. ein Schema-Browser und die Unterstützung von Templates für Einträge enthalten.

[1.9] *http://www.ldapadmin.org*

LAT – LDAP Administration Tool (Open Source)

Mit C# unter Verwendung von Mono und GtkSharp realisierte, grafische Anwendung zum Browsen in LDAP-Verzeichnissen. Weitere Funktionen: die Manipulation von Einträgen, das Untersuchen des Schemas sowie spezielle Unterstützung für das Active Directory.

[1.10] *http://sourceforge.net/projects/ldap-at/*

1.6.2 Informationen zu öffentlichen LDAP-Servern

Die folgenden Seiten stellen Listen mit den Verbindungsdaten öffentlicher LDAP-Server bereit. Eine geeignete Google-Suche fördert weitere zutage.

[1.11] Öffentliche LDAP Server im LDAP Wiki: *http://ldapwiki.willeke.com/wiki/Public%20LDAP%20Servers*

[1.12] Jochen Keutel: *http://keutel.de/directory/public_ldap_servers.html*

[1.13] eMailman: *http://www.emailman.com/ldap/public.html*

1.6.3 LDAP-fähige Serverprodukte

Freie Lösungen (Auswahl)

[1.14] 389 Directory Server: *http://directory.fedoraproject.org*

[1.15] Apache Directory Project: *http://directory.apache.org*

[1.16] OpenDS Project: *http://opends.java.net*

[1.17] OpenLDAP Software: *http://www.openldap.org*

Kommerzielle Produkte (Auswahl)

[1.18] Apple Mac OS X Server Open Directory: *http://www.apple.com/osx/server/*

[1.19] Atos DirX Directory: *http://atos.net/en-us/solutions/identity-security-and-risk-management/*

[1.20] CA Directory (eTrust Directory): *http://www.ca.com/us/ca-directory-services.aspx*

[1.21] IBM Tivoli Directory Server: *http://www.ibm.com/software/tivoli/products/directory-server/*

[1.22] Isode M-Vault: *http://www.isode.com/products/m-vault-directory.html*

[1.23] Microsoft Active Directory: *http://www.microsoft.com/ad/*

[1.24] Microsoft Active Directory Lightweight Directory Services (AD LDS): *http://technet.microsoft.com/en-us/library/cc754361*

[1.25] NetIQ eDirectory: *https://www.netiq.com/products/edirectory/*

[1.26] Oracle Directory Server (früher Sun Java System Directory Server): *http://www.oracle.com/technetwork/middleware/id-mgmt/overview/index-085178.html*

[1.27] Oracle Internet Directory: *http://www.oracle.com/technetwork/middleware/id-mgmt/overview/index-082035.html*

[1.28] Oracle Unified Directory: *http://www.oracle.com/technetwork/middleware/id-mgmt/overview/oud-433568.html*

[1.29] Red Hat Directory Server: *http://www.redhat.com/products/identity-management/directoryserver/*

[1.30] UnboundID Identity Data Platform: *https://www.unboundid.com/products/*