LDAP

RES, Lecture 6

Olivier Liechti

heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud

Agenda

- Introduction
  - LDAP: history, objectives and overview
  - Tools: servers, browsers, APIs and libraries
- LDAP: the data model
  - Hierarchical organization, naming
  - Core concepts: DIT, entry, attribute, class, schema
- LDAP: the protocol
  - Principles, operations and the LDIF data format
- LDAP: the infrastructure
  - Distribution and replication
  - Commands, filters, etc.
- LDAP with Java: Java Naming & Directory Interface (JNDI)
  - Authentication, query, data manipulation

Références

- LDAP for Rocket Scientists (ZYTRAX, Inc.)
  - http://www.zytrax.com/books/ldap/
- Redbook IBM
  - http://www.redbooks.ibm.com/abstracts/sg244986.html
- Tutorials and presentations
  - http://quark.humbug.org.au/publications/ldap/
  - http://www.it-sudparis.eu/s2ia/user/procacci/ldap/
  - http://www.hawaii.edu/its/brownbags-trainings/ldap/
- RFCs
  - http://www.mozilla.org/directory/standards.html
- OpenDJ
  - http://www.forgerock.com/en-us/products/directory-services/
  - http://opendj.forgerock.org/
- LDAP Clients
  - http://directory.apache.org/studio/
  - http://www-unix.mcs.anl.gov/~gawor/ldap/

Introduction

LDAP: a Directory Service

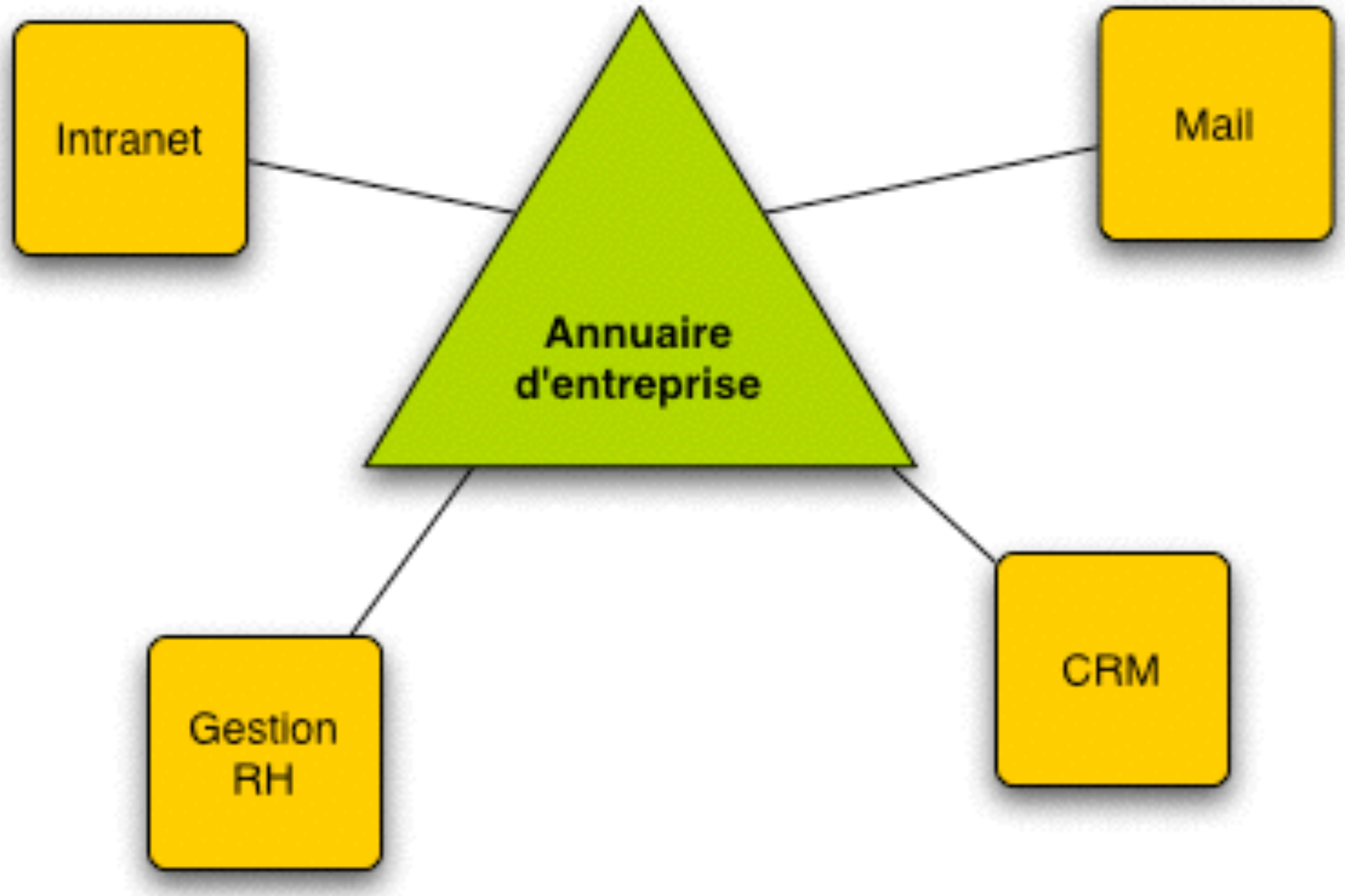
heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud

- Late 70's
  - Standardisation of directory service by the UIT (X.500).
  - Related to the growing adoption of electronic messaging protocols.
  - Directory Access Protocol (DAP).
- Late 90's
  - Lightweight (simplified) version of the protocol, based on the TCP/IP stack.
  - University of Michigan, IETF
- Key functions
  - Fast information lookup
  - Authentication

http://flickr.com/photos/gehmflor/375334958/sizes/m/#cc_license

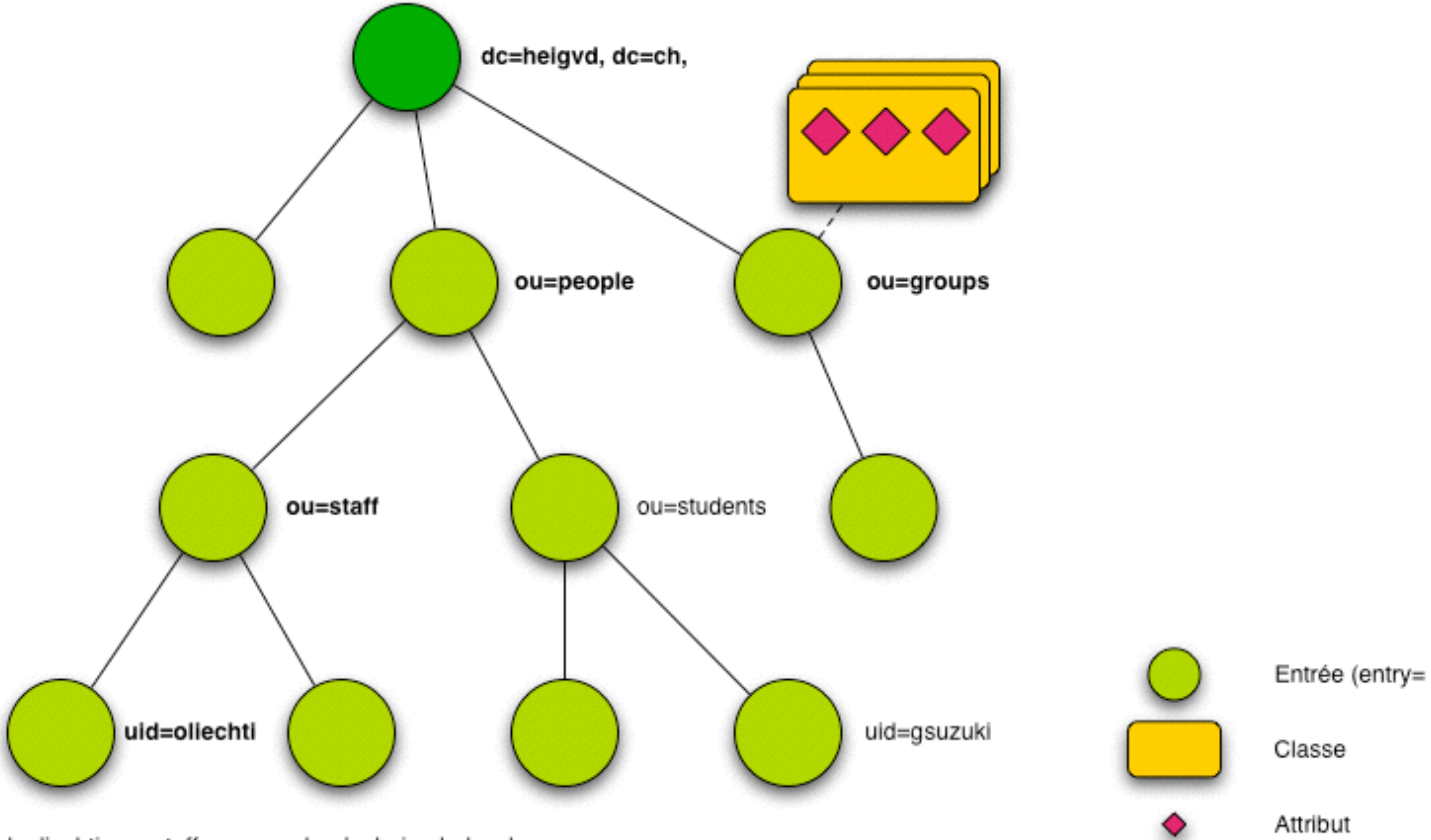
LDAP: Sharing Data in the Enterprise

heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud



LDAP: the Data Model

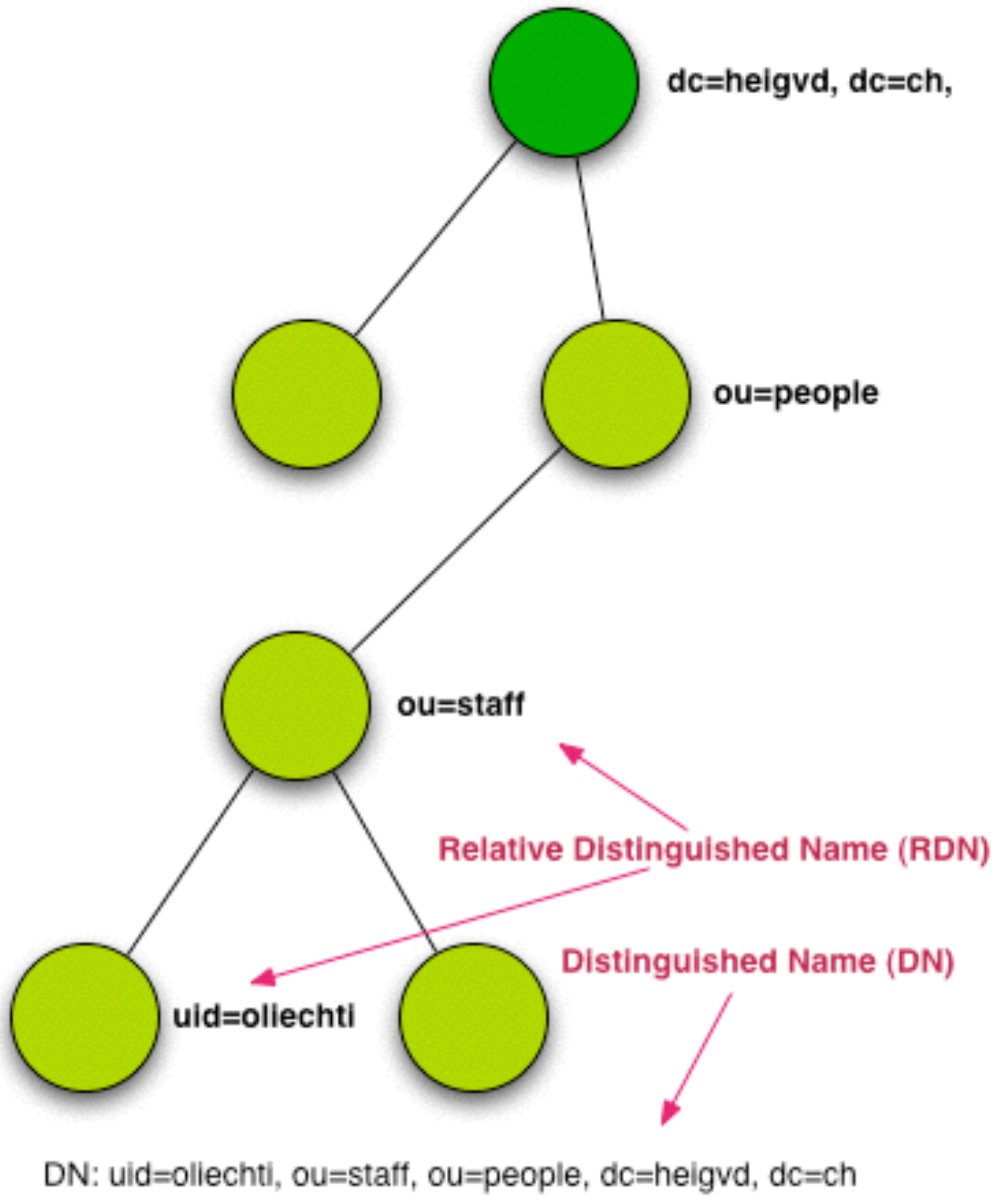
LDAP: the Data Model



DN: uid=oliechti, ou=staff, ou=people, dc=heigvd, dc=ch

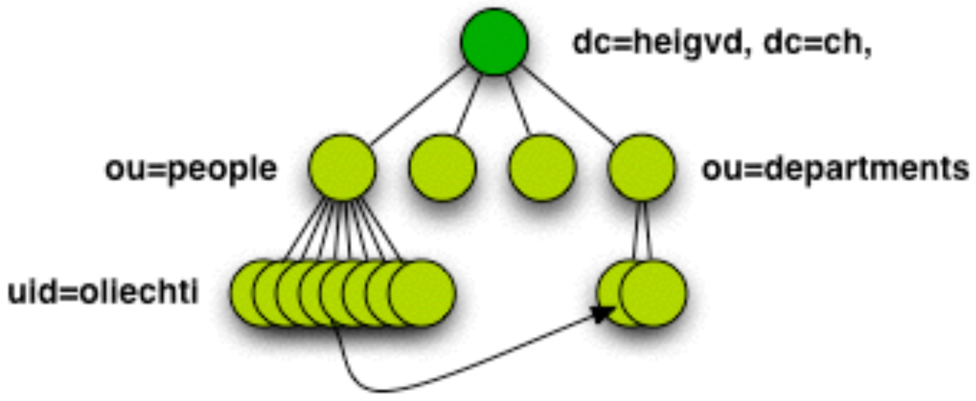
The Directory Information Tree (DIT)

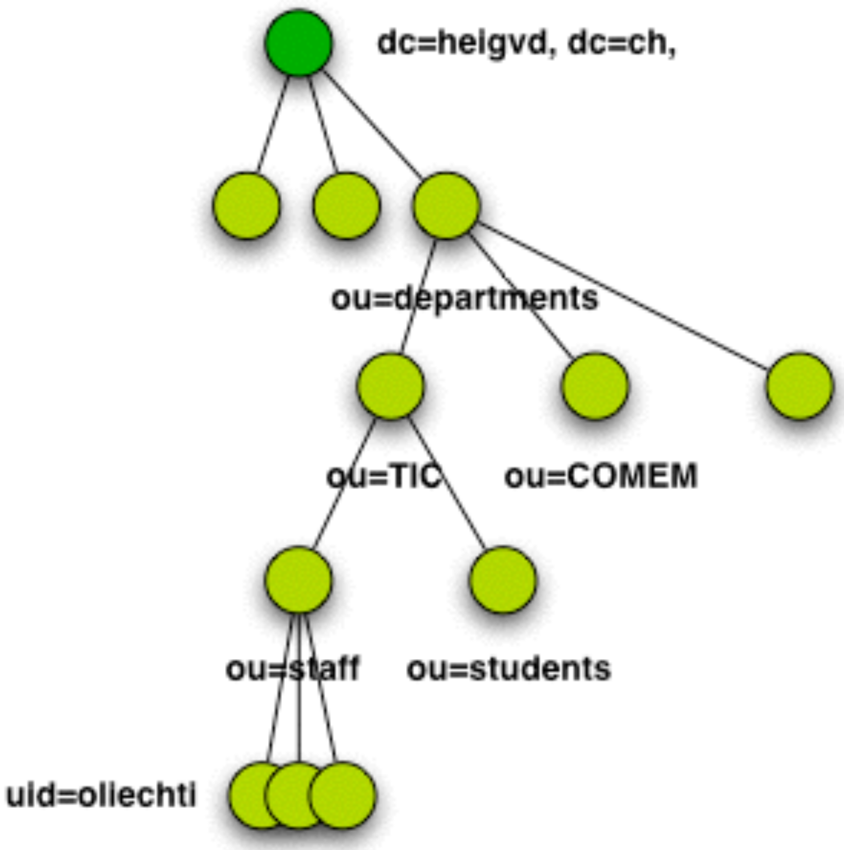
- Data is organized hierarchically, in a tree:
  - The “root” is also called “suffix” or “base”.
  - Each node in the tree is an LDAP “entry”.
  - The intermediate nodes are “container” nodes.
- LDAP entries are named:
  - The **Distinguished Name (DN)** is used to identify and locate an entry in the tree.
  - The DN provides the path from the root to the entry.
  - The **Relative Distinguished Name** (RDN) uniquely identifies an entry among siblings (nodes that are children of the same node)


dc=heigvd, dc=ch,
ou=people
ou=staff
Relative Distinguished Name (RDN)
Distinguished Name (DN)
uid=oliechti
DN: uid=oliechti, ou=staff, ou=people, dc=heigvd, dc=ch

How to Structure the DIT

- What can we store in a directory?
  - People (e.g. employees, customers, partners, etc.)
  - Equipment (e.g. printers, file servers, etc.)
  - Software services (e.g. web services, etc.)
  - Configuration parameters (e.g. of the server itself)
- When storing people, how do we structure the DIT?
  - Do we reflect the org chart, by department?
  - Do we structure by country?
- Recommendation
  - A flat structure is much move convenient, flexible and evolvable than a deep one.





http://docs.sun.com/app/docs/doc/820-2488/fpqzx?a=view
http://docs.sun.com/source/816-6679-10/dit.htm#1015250

The Notion of Entry

> An LDAP “**entry**” LDAP is an object stored in the directory.

> It is a **node** in the DIT.

> An entry is uniquely identified by its **Distinguished Name** (DN)

> An entry is locally (among siblings) identified by its **Relative Distinguished Name** (RDN).

> The state of an entry is defined by a list of **attributes and attribute values.**

> The **structure** of an entry (i.e. the list of attributes) is defined in one or more **classes**. The multi-valued **ObjectClass attribute** of an entry is used to specify which classes it is an instance of.

> Examples of entries:

- A person, a group, a department, a printer, an online service, a configuration parameter, etc.

The Notion of Object Class

- The notion of LDAP object class is similar to the notion of class in an **object-oriented programming language**.
- A class is defined by a list of **attributes**
  - some of which are **mandatory**
  - some of which are **optional**
  - some of which are **multivalued**
- A class can **extend** another one (inheritance)
- The **RFC 2252** (LDAPv3 Attribute Syntax Definitions) provides the **syntax** to define classes.
- Many classes have been standardized and specified in RFCs, for example:
  - `inetOrgPerson, OrganizationalPerson, Person`
  - `organizationalUnit`
  - `groupOfUniqueNames`

Syntax to Define a Class

```
ObjectClassDescription = "(" whsp
    numericoid whsp      ; ObjectClass identifier
    [ "NAME" qdescrs ]
    [ "DESC" qdstring ]
    [ "OBSOLETE" whsp ]
    [ "SUP" oids ]       ; Superior ObjectClasses
    [ ( "ABSTRACT" / "STRUCTURAL" / "AUXILIARY" ) whsp ]
                         ; default structural
    [ "MUST" oids ]      ; AttributeTypes
    [ "MAY" oids ]       ; AttributeTypes
whsp ")"
```

Example: inetOrgPerson

```
objectClasses: ( 2.16.840.1.113730.3.2.2 NAME 'inetOrgPerson'
  SUP organizationalPerson STRUCTURAL MAY ( audio $ businessCategory $
  carLicense $ departmentNumber $ displayName $ employeeNumber $ employeeType $
  givenName $ homePhone $ homePostalAddress $ initials $ jpegPhoto $
  labeledURI $ mail $ manager $ mobile $ o $ pager $ photo $ roomNumber $
  secretary $ uid $ userCertificate $ x500UniqueIdentifier $
  preferredLanguage $ userSMIMECertificate $ userPKCS12 ) X-ORIGIN 'RFC 2798' )
```

The Notion of Attribute

- **Attributes** define that state of an entry.
- Attributes are **referenced in classes**.
- Attributes have a **type** (String, Binary, etc.).
- Attributes can be **multivalued**.

```
attributeTypes: ( 0.9.2342.19200300.100.1.41
  NAME ( 'mobile' 'mobileTelephoneNumber' ) EQUALITY telephoneNumberMatch
  SUBSTR telephoneNumberSubstringsMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.50
  X-ORIGIN 'RFC 4524' )
```

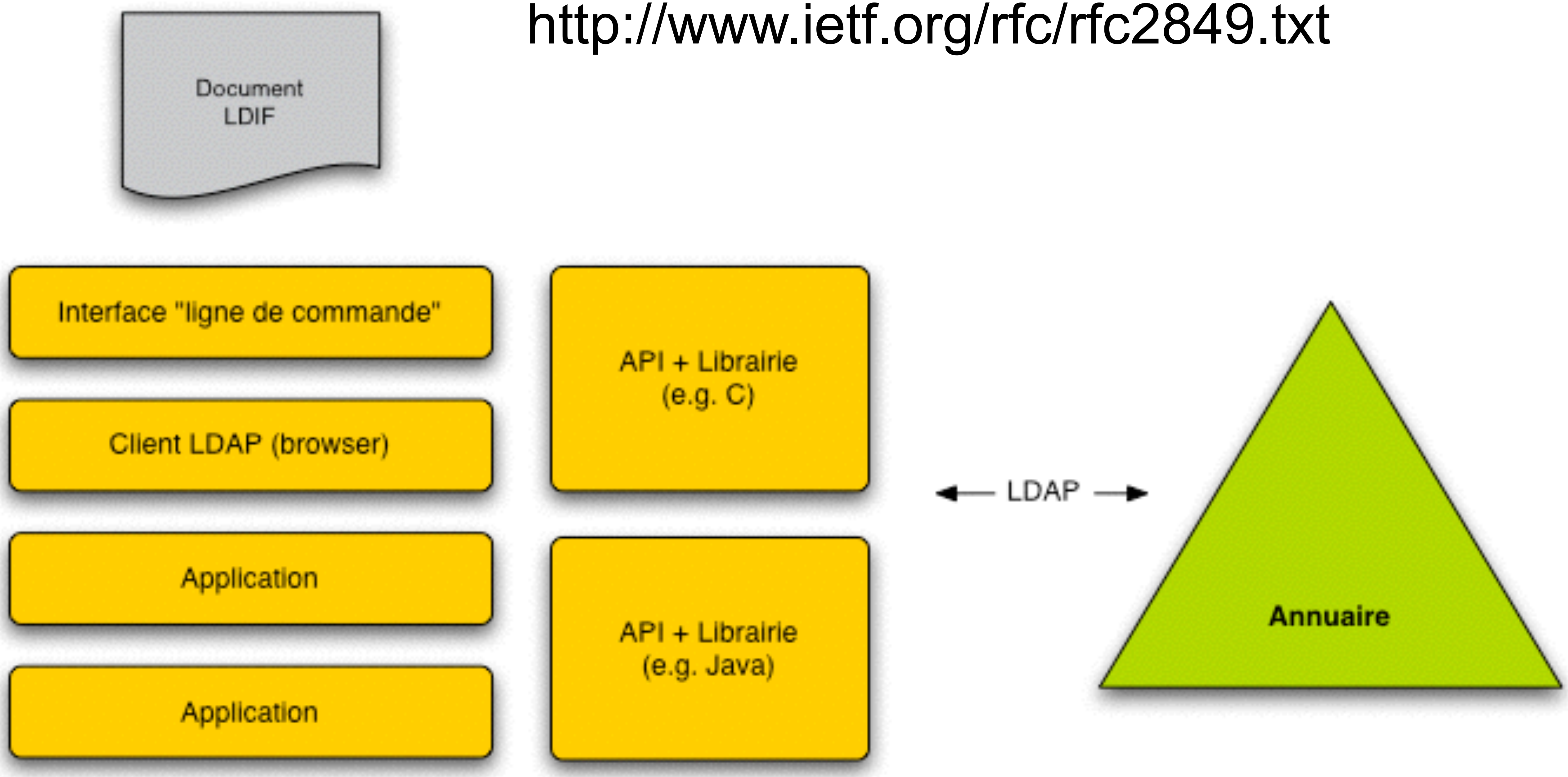
The Notion of Schema

- When deploying an LDAP directory service, one has to specify the **schema** that defines the rules governing the structure of managed data:
  - What are the classes that are supported and that can be used to create entries?
  - What are the attributes that are supported and used to define classes?
  - etc.
- There are **standard schemas** and when you install an LDAP server, a default one is available to you. Very often, you do not need more and can create entries based on the standard classes and attributes (InetOrgPerson, OrganizationalUnit, etc.).
- If you have special needs, then you can **extend the schema** with:
  - custom classes (e.g. heigvdPerson)
  - custom attributes (e.g. gapsIdNumber)
- The procedure for extending the schema depends on the actual LDAP server (OpenDJ, OpenLDAP, Active Directory, etc.)

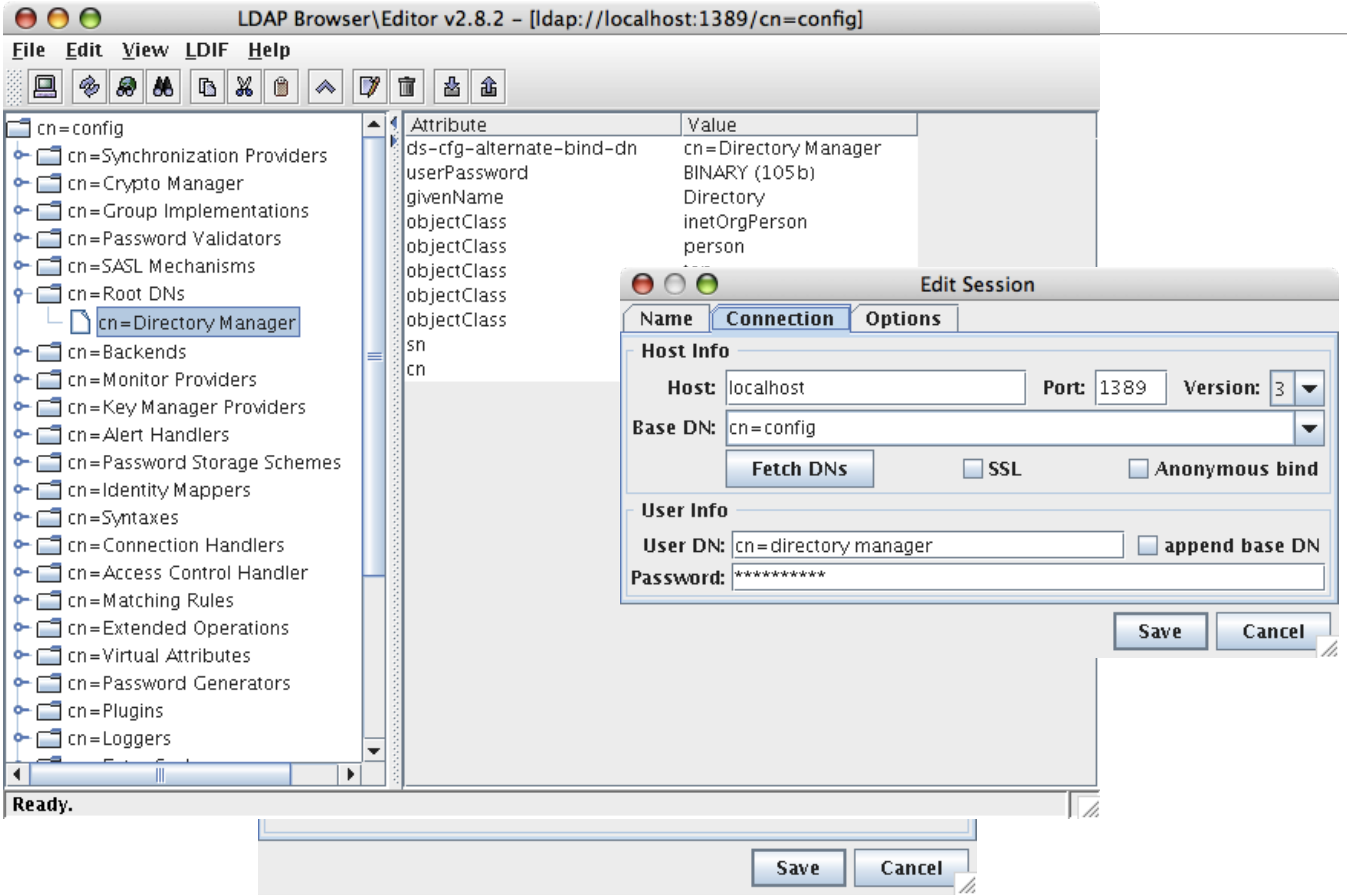
LDAP: the Infrastructure

LDAP: Components

http://www.ietf.org/rfc/rfc2849.txt



LDAPBrowser



LDAP Browser\Editor v2.8.2 – [ldap://localhost:1389/cn=config]

- cn=config
  - cn=Synchronization Providers
  - cn=Crypto Manager
  - cn=Group Implementations
  - cn=Password Validators
  - cn=SASL Mechanisms
  - cn=Root DNs
    - cn=Directory Manager
  - cn=Backends
  - cn=Monitor Providers
  - cn=Key Manager Providers
  - cn=Alert Handlers
  - cn=Password Storage Schemes
  - cn=Identity Mappers
  - cn=Syntaxes
  - cn=Connection Handlers
  - cn=Access Control Handler
  - cn=Matching Rules
  - cn=Extended Operations
  - cn=Virtual Attributes
  - cn=Password Generators
  - cn=Plugins
  - cn=Loggers

Attribute	Value
ds-cfg-alternate-bind-dn	cn=Directory Manager
userPassword	BINARY (105b)
givenName	Directory
objectClass	inetOrgPerson
objectClass	person
objectClass	
objectClass	
objectClass	
sn	
cn	

Ready.

Edit Session

Host Info

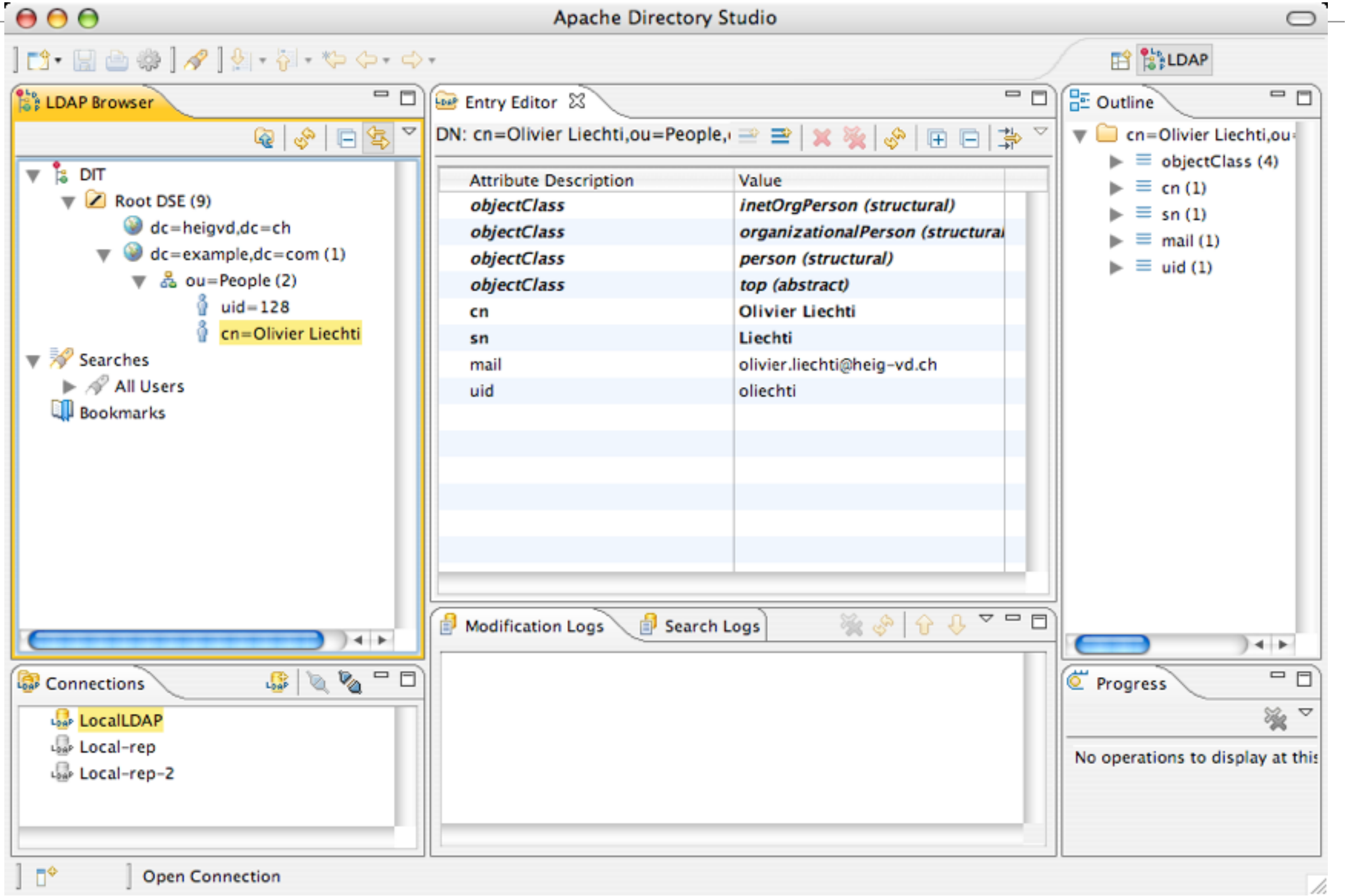
Host: localhost
Port: 1389
Version: 3
Base DN: cn=config

□ SSL
□ Anonymous bind

User Info

User DN: cn=directory manager
□ append base DN
Password: ***********

Apache Directory Studio



DN: cn=Olivier Liechti,ou=People,

Attribute Description	Value
objectClass	*inetOrgPerson (structural)*
objectClass	*organizationalPerson (structural*
objectClass	*person (structural)*
objectClass	*top (abstract)*
cn	Olivier Liechti
sn	Liechti
mail	olivier.liechti@heig-vd.ch
uid	oliechti

No operations to display at this

OpenDS & OpenDJ

LDAP: the Protocol

LDAP: the Protocol

- LDAP is a client-server protocol
  - Operates on top of TCP
  - Standard port: 389
- Main LDAP commands
  - **Bind** (authentication and session establishment)
  - **Search** - search for and/or retrieve directory entries
  - **Add** a new entry
  - **Delete** an entry
  - **Modify** an entry
  - **Modify** Distinguished Name (DN) - move or rename an entry
  - **Unbind** (session termination)

OpenDS

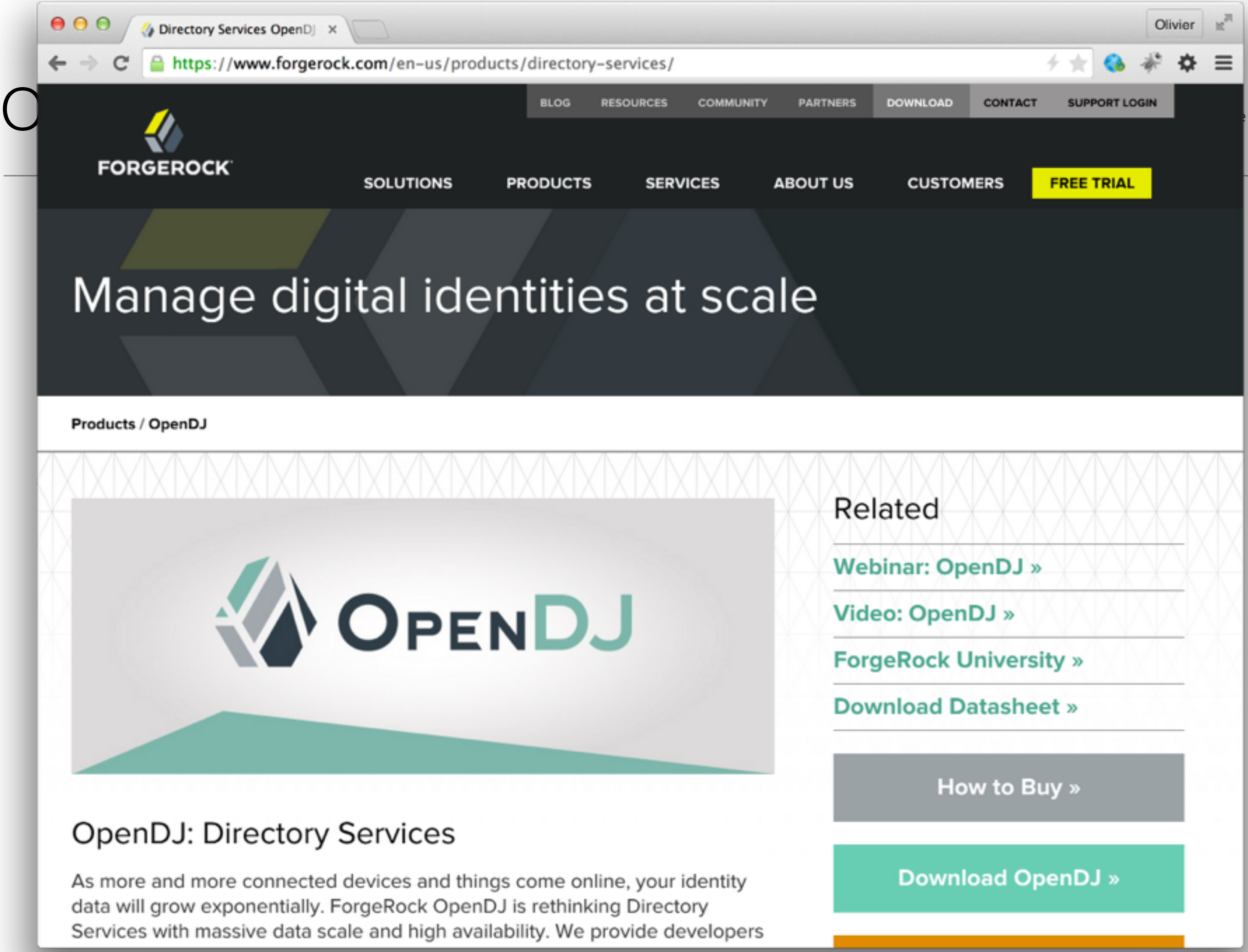
- OpenDS is an LDAP server:
  - developed in Open Source, with the support of Sun Microsystems (now Oracle)
  - 100% Java
  - “Embeddable" in applications
- Installation and setup is very easy
  - via Java WebStart
  - File structure is straightforward
- Yet, OpenDS is “enterprise-ready”
  - replication
  - performance
- OpenDS makes it possible to quickly and easily experiment with LDAP
- http://www.opends.org/

OpenDJ

- OpenDJ is a fork of OpenDS:
  - developed in Open Source, by ForgeRock
  - 100% Java
- Two web sites:
  - Open source project page: http://opendj.forgerock.org
  - ForgeRock product page: https://www.forgerock.com/en-us/products/directory-services/

O

Gestion



Directory Services OpenDJ

Olivier

https://www.forgerock.com/en-us/products/directory-services/

Manage digital identities at scale

Products / OpenDJ

Related

Webinar: OpenDJ »

Video: OpenDJ »

ForgeRock University »

Download Datasheet »

How to Buy »

Download OpenDJ »

OpenDJ: Directory Services

As more and more connected devices and things come online, your identity data will grow exponentially. ForgeRock OpenDJ is rethinking Directory Services with massive data scale and high availability. We provide developers

Installing OpenDJ

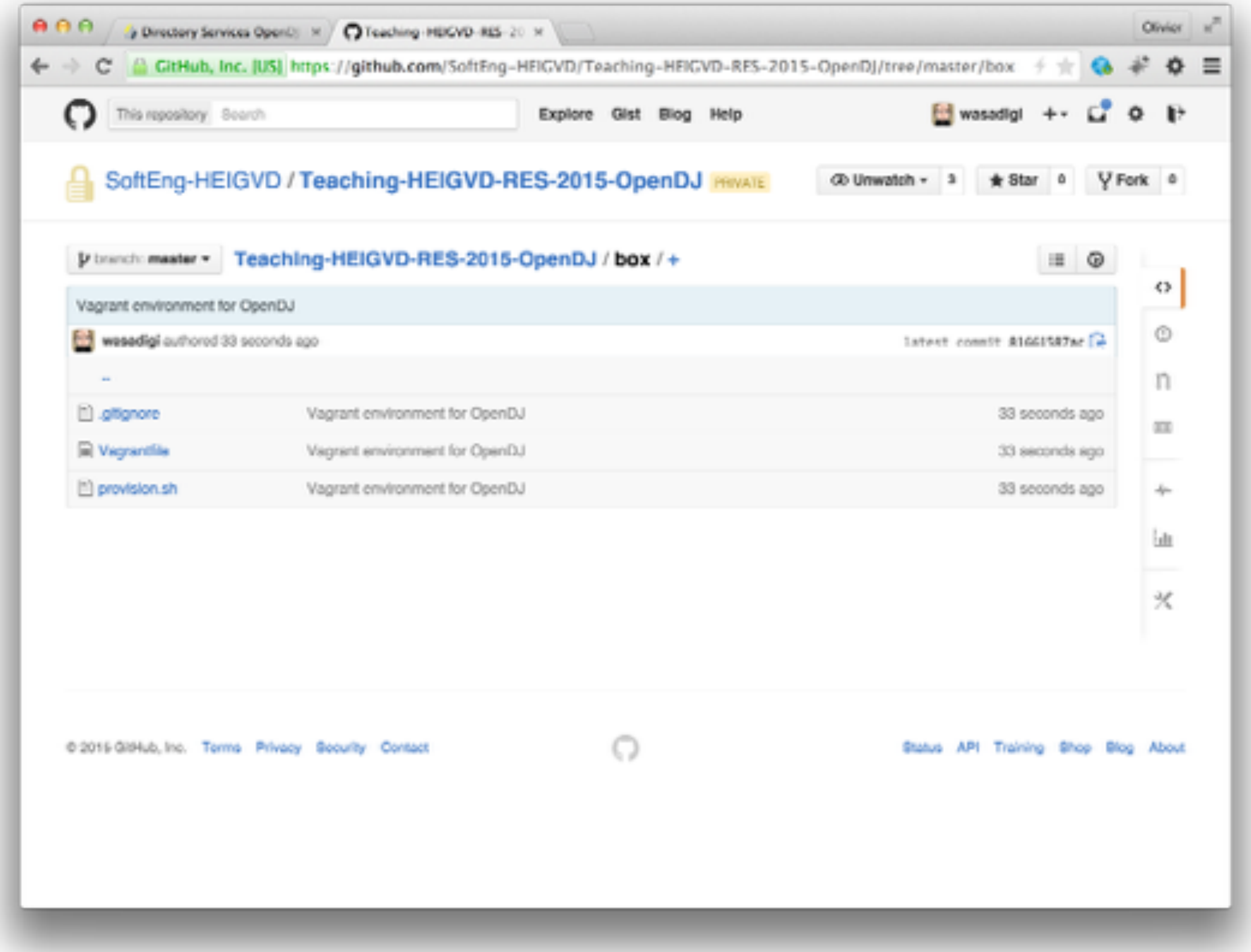
- **Warning**: OpenDJ is quite sensitive to the version of the Java environment. Make sure to read the release notes and requirements:
    - OpenDJ 2.6 does NOT support Java 8.
    - OpenDJ 2.6.2 does support Java 8, but at this time it is only available to paying customers (subscriptions).
- **Note**: OpenDJ 2.6 is free, but you need to register in order to download the package.

Installing OpenDJ - Vagrant

heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud

- We have prepared a Vagrant environment for your experiments with OpenDJ, with a JDK 7.
- We are not allowed to add the OpenDJ package in a public repository, so you will have to register with ForgeRock, download the package and install it in your box.

https://github.com/SoftEng-HEIGVD/Teaching-HEIGVD-RES-2015-OpenDJ



SoftEng-HEIGVD / Teaching-HEIGVD-RES-2015-OpenDJ PRIVATE

Teaching-HEIGVD-RES-2015-OpenDJ / box / +

Vagrant environment for OpenDJ

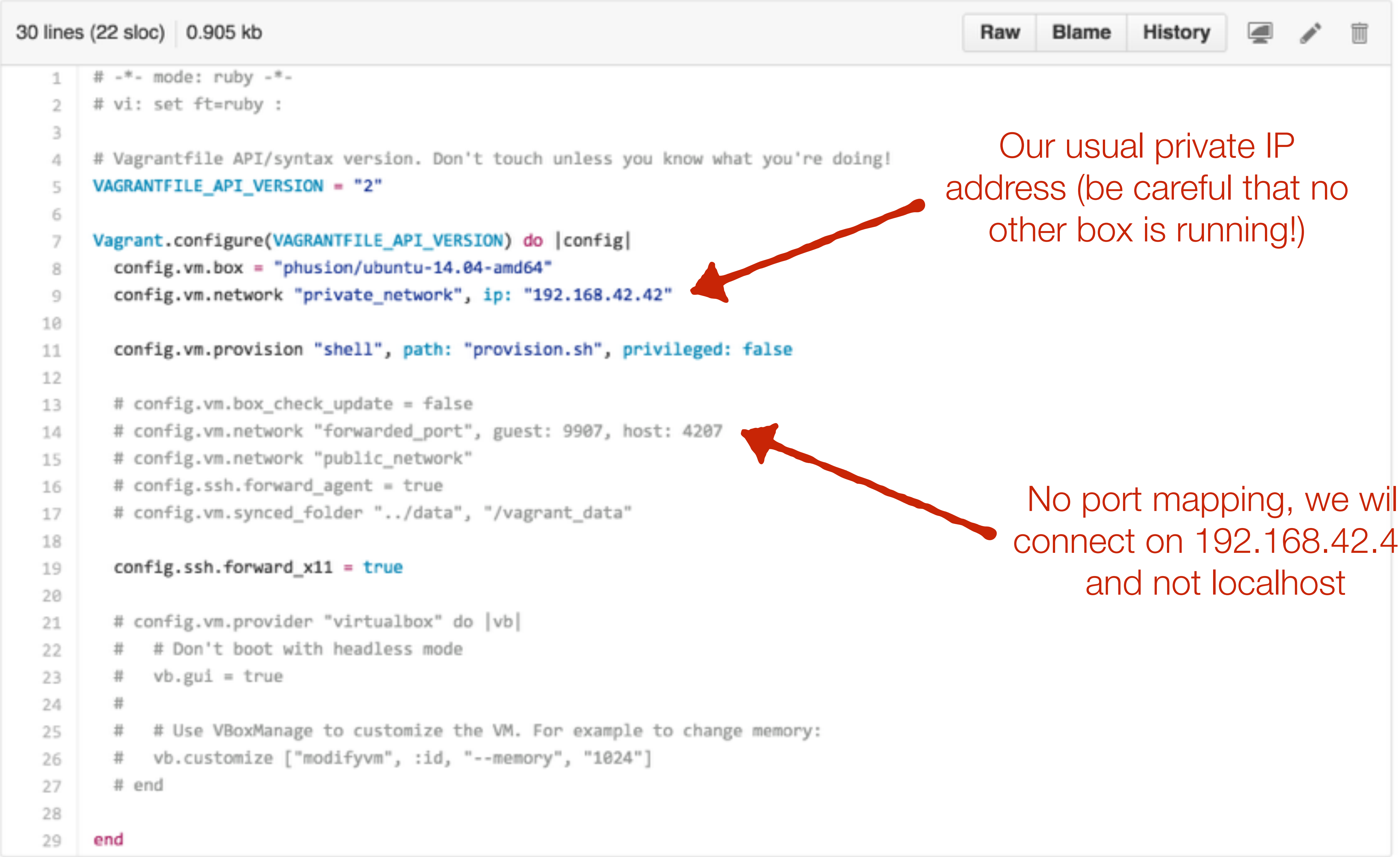
wasadigi authored 33 seconds ago — latest commit 81661587ae

..

.gitignore	Vagrant environment for OpenDJ	33 seconds ago
Vagrantfile	Vagrant environment for OpenDJ	33 seconds ago
provision.sh	Vagrant environment for OpenDJ	33 seconds ago

Installing OpenDJ - Vagrantfile

heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud

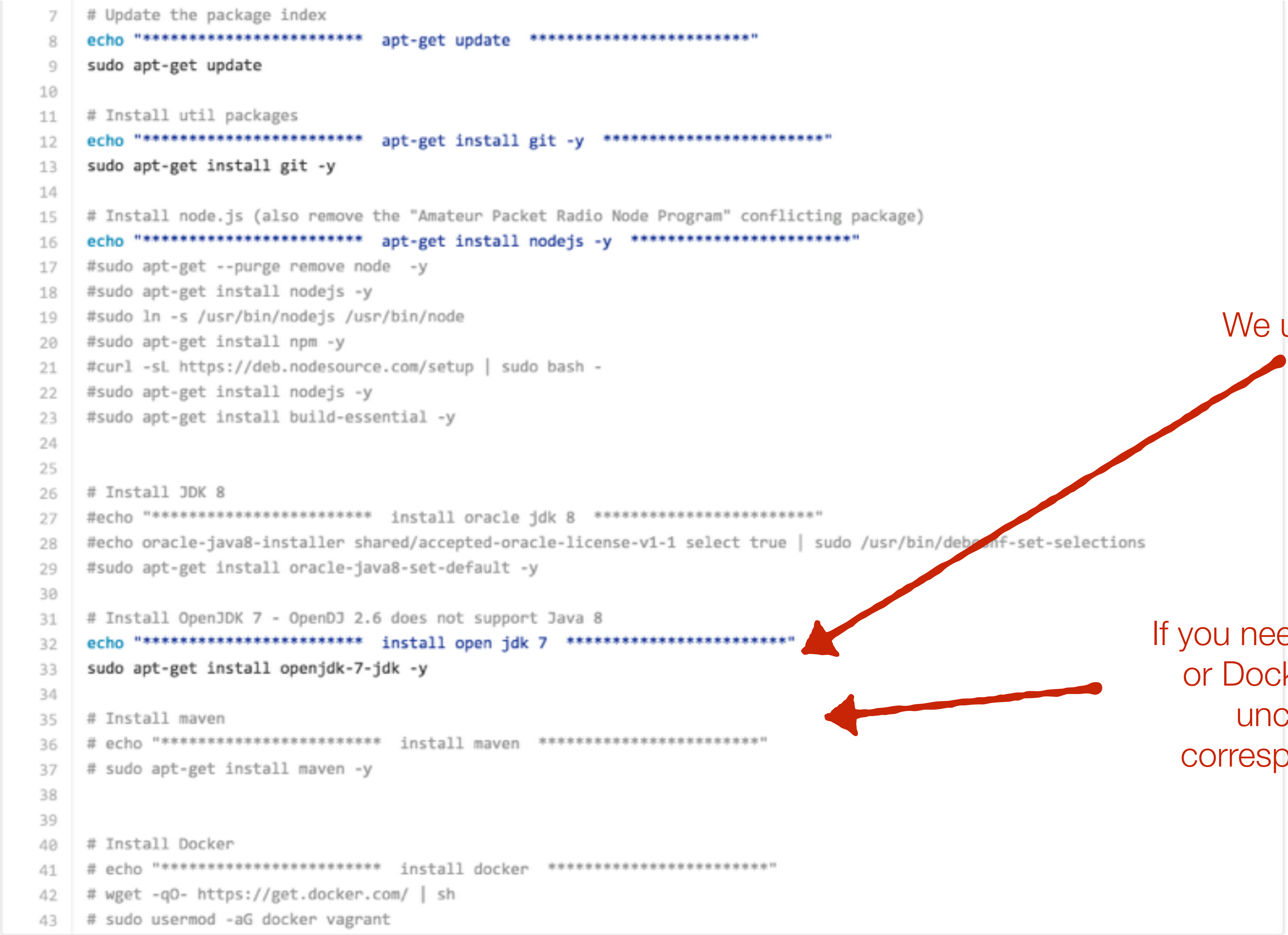


```ruby
# -*- mode: ruby -*-
# vi: set ft=ruby :

# Vagrantfile API/syntax version. Don't touch unless you know what you're doing!
VAGRANTFILE_API_VERSION = "2"

Vagrant.configure(VAGRANTFILE_API_VERSION) do |config|
  config.vm.box = "phusion/ubuntu-14.04-amd64"
  config.vm.network "private_network", ip: "192.168.42.42"

  config.vm.provision "shell", path: "provision.sh", privileged: false

  # config.vm.box_check_update = false
  # config.vm.network "forwarded_port", guest: 9907, host: 4207
  # config.vm.network "public_network"
  # config.ssh.forward_agent = true
  # config.vm.synced_folder "../data", "/vagrant_data"

  config.ssh.forward_x11 = true

  # config.vm.provider "virtualbox" do |vb|
  #   # Don't boot with headless mode
  #   vb.gui = true
  #
  #   # Use VBoxManage to customize the VM. For example to change memory:
  #   vb.customize ["modifyvm", :id, "--memory", "1024"]
  # end

end
```

Our usual private IP address (be careful that no other box is running!)

No port mapping, we will connect on 192.168.42.42 and not localhost

Installing OpenDJ - provision.sh



```
# Update the package index
echo "************************  apt-get update  ************************"
sudo apt-get update

# Install util packages
echo "************************  apt-get install git -y  ************************"
sudo apt-get install git -y

# Install node.js (also remove the "Amateur Packet Radio Node Program" conflicting package)
echo "************************  apt-get install nodejs -y  ************************"
#sudo apt-get --purge remove node  -y
#sudo apt-get install nodejs -y
#sudo ln -s /usr/bin/nodejs /usr/bin/node
#sudo apt-get install npm -y
#curl -sL https://deb.nodesource.com/setup | sudo bash -
#sudo apt-get install nodejs -y
#sudo apt-get install build-essential -y


# Install JDK 8
#echo "************************  install oracle jdk 8  ************************"
#echo oracle-java8-installer shared/accepted-oracle-license-v1-1 select true | sudo /usr/bin/debconf-set-selections
#sudo apt-get install oracle-java8-set-default -y

# Install OpenJDK 7 - OpenDJ 2.6 does not support Java 8
echo "************************  install open jdk 7  ************************"
sudo apt-get install openjdk-7-jdk -y

# Install maven
# echo "************************  install maven  ************************"
# sudo apt-get install maven -y


# Install Docker
# echo "************************  install docker  ************************"
# wget -qO- https://get.docker.com/ | sh
# sudo usermod -aG docker vagrant
```

We use OpenJDK 7

If you need Node.js, maven or Docker on your box, uncomment the corresponding sections

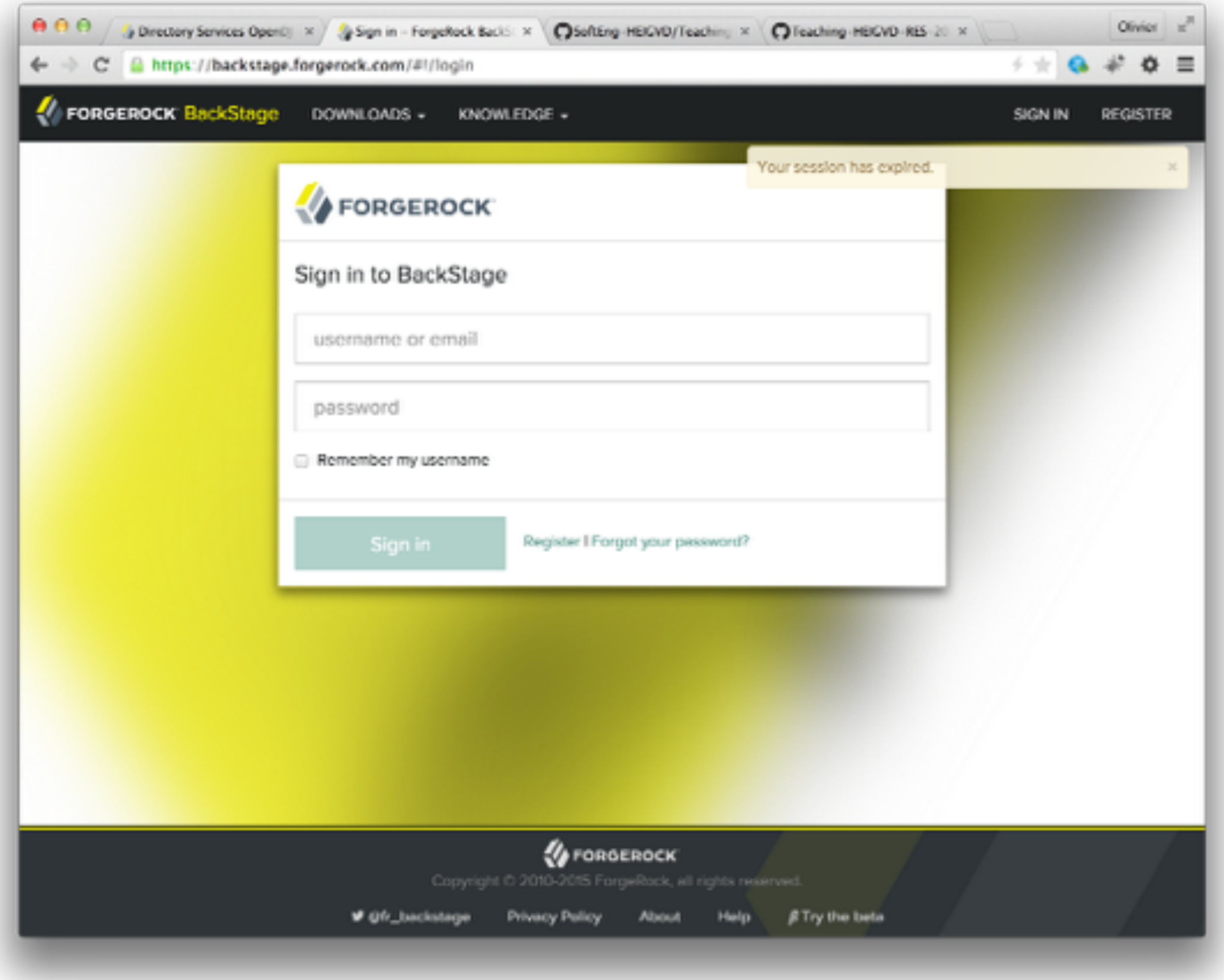
Getting started...

- Check that all your Vagrant boxes are stopped (e.g. use the VirtualBox GUI).
- Clone the repo and fire up the box; this will trigger the execution of the `provision.sh` script (including the download of OpenJDK 7...)

```
$ git clone git@github.com:SoftEng-HEIGVD/Teaching-HEIGVD-RES-2015-OpenDJ.git
$ cd Teaching-HEIGVD-RES-2015-OpenDJ
$ vagrant up
$ vagrant ssh
$ java --version
```

Getting started...

- Go to OpenDJ home page and click on the Download link.
- Register



https://backstage.forgerock.com/#!/login

Your session has expired.

Sign in to BackStage

username or email

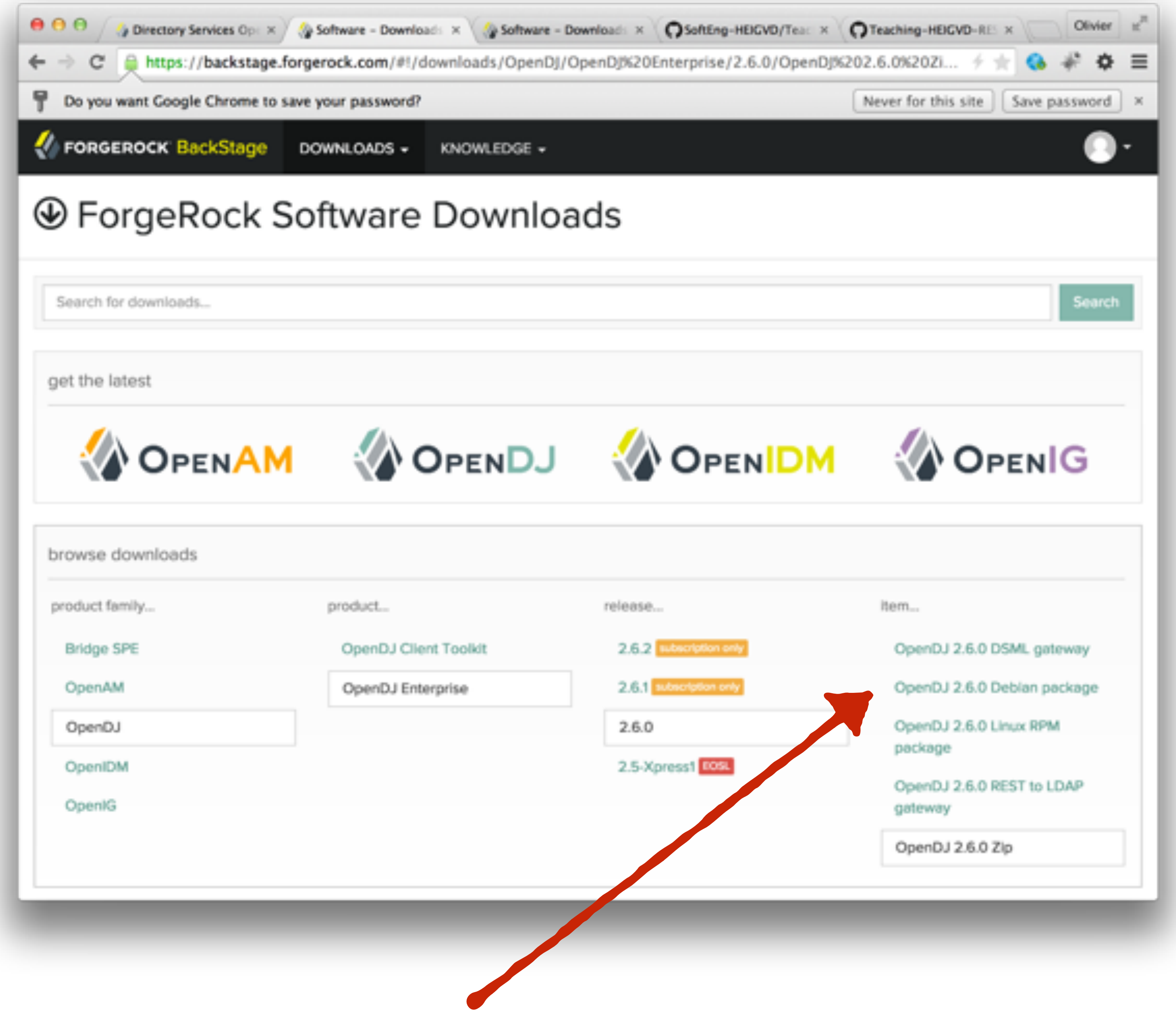
password

□ Remember my username

Copyright © 2010-2015 ForgeRock, all rights reserved.

Getting started...

- Go to OpenDJ home page and click on the **Download** link.
- **Register** and get your ForgeRock account.
- Select OpenDJ / OpenDJ Enterprise / 2.6.0 / **OpenDJ 2.6.0 Debian package**.
- Save the file in your clone, in the directory containing the `Vagrantfile`.



Do you want Google Chrome to save your password?

ForgeRock Software Downloads

Search for downloads...

get the latest

browse downloads

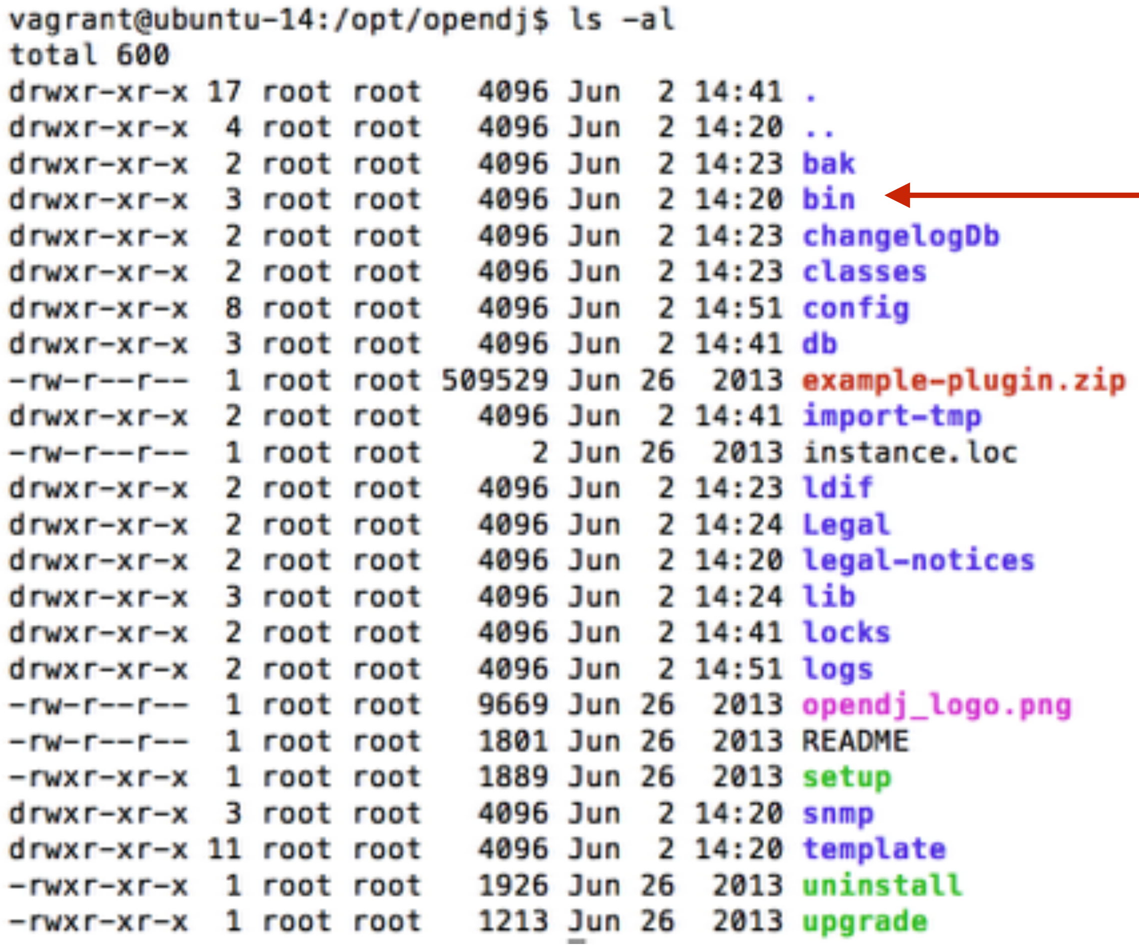
product family...	product...	release...	item...
Bridge SPE	OpenDJ Client Toolkit	2.6.2 subscription only	OpenDJ 2.6.0 DSML gateway
OpenAM	OpenDJ Enterprise	2.6.1 subscription only	OpenDJ 2.6.0 Debian package
OpenDJ		2.6.0	OpenDJ 2.6.0 Linux RPM package
OpenIDM		2.5-Xpress1 EOSL	OpenDJ 2.6.0 REST to LDAP gateway
OpenIG			OpenDJ 2.6.0 Zip

Getting started...

heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud

- Go back to your box. The debian package should be available in `/vagrant`.

```
$ cd /vagrant
$ ls -al
$ sudo dpkg -i opendj_2.6.0-1_all.deb
$ cd /opt/opendj
$ ls -al
```



```
vagrant@ubuntu-14:/opt/opendj$ ls -al
total 600
drwxr-xr-x 17 root root   4096 Jun  2 14:41 .
drwxr-xr-x  4 root root   4096 Jun  2 14:20 ..
drwxr-xr-x  2 root root   4096 Jun  2 14:23 bak
drwxr-xr-x  3 root root   4096 Jun  2 14:20 bin
drwxr-xr-x  2 root root   4096 Jun  2 14:23 changelogDb
drwxr-xr-x  2 root root   4096 Jun  2 14:23 classes
drwxr-xr-x  8 root root   4096 Jun  2 14:51 config
drwxr-xr-x  3 root root   4096 Jun  2 14:41 db
-rw-r--r--  1 root root 509529 Jun 26  2013 example-plugin.zip
drwxr-xr-x  2 root root   4096 Jun  2 14:41 import-tmp
-rw-r--r--  1 root root      2 Jun 26  2013 instance.loc
drwxr-xr-x  2 root root   4096 Jun  2 14:23 ldif
drwxr-xr-x  2 root root   4096 Jun  2 14:24 Legal
drwxr-xr-x  2 root root   4096 Jun  2 14:20 legal-notices
drwxr-xr-x  3 root root   4096 Jun  2 14:24 lib
drwxr-xr-x  2 root root   4096 Jun  2 14:41 locks
drwxr-xr-x  2 root root   4096 Jun  2 14:51 logs
-rw-r--r--  1 root root   9669 Jun 26  2013 opendj_logo.png
-rw-r--r--  1 root root   1801 Jun 26  2013 README
-rwxr-xr-x  1 root root   1889 Jun 26  2013 setup
drwxr-xr-x  3 root root   4096 Jun  2 14:20 snmp
drwxr-xr-x 11 root root   4096 Jun  2 14:20 template
-rwxr-xr-x  1 root root   1926 Jun 26  2013 uninstall
-rwxr-xr-x  1 root root   1213 Jun 26  2013 upgrade
```

This is where you find the interesting stuff!

```
$ export PATH=$PATH:/opt/opendj/bin
```

Getting started...

- You can now run the “quick install procedure”

```
$ sudo /opt/opendj/setup --cli
```

- Make sure to remember the **admin password**!!

- Use default values, except for
  - “Provide the base DN for the directory data”. Enter `“dc=heigvd, dc=ch”`.
  - “Options for populating the database”. Pick `“4) Load automatically...”`

- **Note**: if you need to start from scratch and get rid of the previous config:

```
$ cd /opt/opendj
$ sudo rm -fr db/
$ sudo rm -fr config/
$ sudo rm -fr locks/
$ sudo rm -fr logs/
```

Getting started...

- At the end of the procedure, the directory server should be running and listening on port 389 (if you used sudo and picked the default option).

```
$ telnet localhost 389
```

- Run a few queries (and interpret the results):

```
$ ldapsearch -p 389 -b "dc=heigvd, dc=ch" “objectClass=*"
$ ldapsearch -p 389 -b "dc=heigvd, dc=ch" "objectClass=*" cn mail
$ ldapsearch -p 389 -b "dc=heigvd, dc=ch" "l=Tucson" cn mail l
$ ldapsearch -p 389 -b "dc=heigvd, dc=ch" "!(l=Tucson)" cn mail l
$ ldapsearch -p 389 -b "dc=heigvd, dc=ch" “(&(givenName=Bogdan)(sn=Billing))"
$ ldapsearch -p 389 -b "dc=heigvd, dc=ch" "(|(givenName=Bogdan)
(givenName=Alice))"
```

- Also check from your host

```
$ telnet 192.168.42.42 389
```

Warning!

- Many operating systems (Mac OS, Solaris, Linux, etc.) include LDAP commands natively:
  - Example: Mac OS provides `/usr/bin/ldapsearch`
  - These commands are typically in the path
- LDAP servers, such as OpenDJ, provide their own commands. They may use the same name (e.g. ldapsearch) but accept different options and their own syntax!!!
  - Exemple: `${OPEN_DJ_INSTALL_PATH}/bin/ldapsearch`
  - These commands are not in the path by default
- For that reason, when you use LDAP commands:
  - Be careful of which command you are using:
    - `cd ${OPEN_DJ_INSTALL_PATH}/bin/`
    - `./ldapsearch`
  - Is different from:
    - `cd ${OPEN_DJ_INSTALL_PATH}/bin/`
    - `ldapsearch`

ldapsearch (1)

- This command is used to submit queries and extract data from the directory
- Syntax:
  - `ldapsearch [options] [filter] [attributes]`
- Key options:

– `-h, --host`	à quel serveur veut-on se connecter?
– `-p, --port`	sur quel port écoute-t-il?
– `-D, --bindDN`	avec quel identité veut-on se connecter?
– `-w, --bindPassword`	avec quel mot de passe (à éviter, penser à `ps`!!)
– `-b, --baseDN`	à partir d'où veut-on faire la recherche?
– `-a, --searchScope`	avec quelle profondeur?
– `-T, --dontWrap`	pour éviter les ruptures de lignes (LDIF)
– `--propertiesFilePath`	pour éviter de saisir toutes les options

- Documentation:
  - http://opendj.forgerock.org/opendj-server/doc/admin-guide/index/ldapsearch-1.html

ldapsearch (2)

- Syntax for LDAP filters
  - Defined in RFC 2254
  - Operators for filters: `&, |, !`
- Examples:
  - Entries for which the attribute **cn** is equal to “Babs Jensen”:
    - ➔ `(cn=Babs Jensen)`
  - Entries for which the attribute **cn** is different from “Tim Howes”:
    - ➔ `(!(cn=Tim Howes))`
  - People whose **family name** is “Jensen” **or** whose first name is “Babs” and **family name** starts with “J”:
    - ➔ `(&(objectClass=Person)(|(sn=Jensen)(cn=Babs J*))`

ldapsearch (3)

> Return all entries

– `ldapsearch -h hostname -p 389 -b dc=example,dc=com "(objectclass=*)"`

```
dn: dc=example,dc=com
objectClass: domain
objectClass: top
dc: example

dn: ou=Groups,dc=example,dc=com
objectClass: organizationalunit
objectClass: top
ou: Groups

dn: cn=Directory
Administrators,ou=Groups,dc=example,dc=com
objectClass: groupofuniquenames
objectClass: top
ou: Groups
cn: Directory Administrators
uniquemember: uid=kvaughan, ou=People, dc=example,dc=com
uniquemember: uid=rdaugherty, ou=People,
dc=example,dc=com
uniquemember: uid=hmiller, ou=People, dc=example,dc=com
```

> **Return only some attributes:**

– `ldapsearch -h hostname -p 389 -b dc=example,dc=com "(cn=Sam Carter)" telephoneNumber`

```
dn: uid=scarter,ou=People,dc=example,dc=com
telephonenumber: +1 408 555 4798
```

ldapmodify (1)

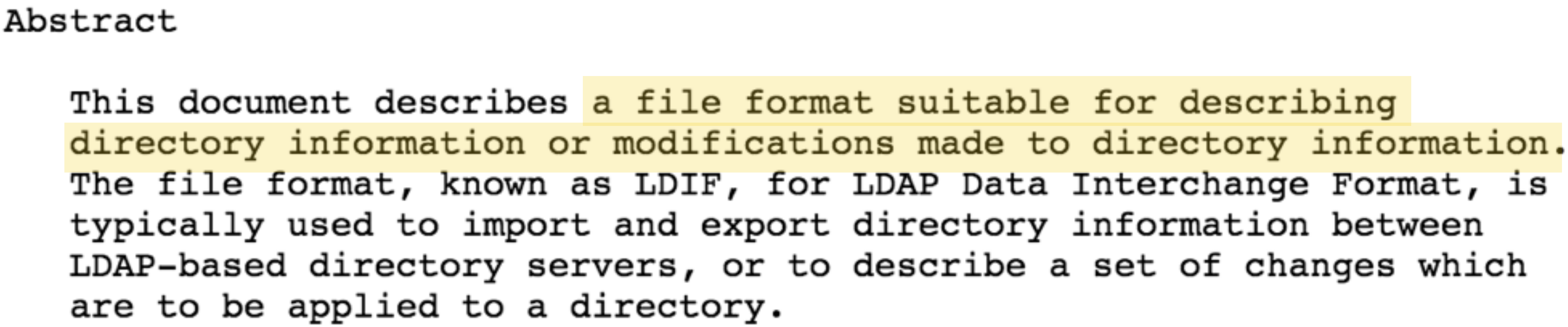
- This command is used to update data in the directory
- Syntax:
  - `ldapmodify [options] [filter] [attributes]`
- Key options:

`-h, --host`	what is the IP address of the server?
`-p, --port`	on which port is it listening?
`-D, --bindDN`	what is the DN of the user connecting to the server?
`-w, --bindPassword`	and his password? (bad practice!! think about the `ps` command!!)
`-f, --filename`	the file containing the LDIF data

- Two ways to provide LDIF data to the server
  - provide LDIF via the command line + CTRL-D (*nix) ou CTRL-Z (Win)
  - Use the `-f` option and capture the LDIF data in a file (strongly recommended)
- Documentation:
  - http://opendj.forgerock.org/opendj-server/doc/admin-guide/index/ldapmodify-1.html

LDIF

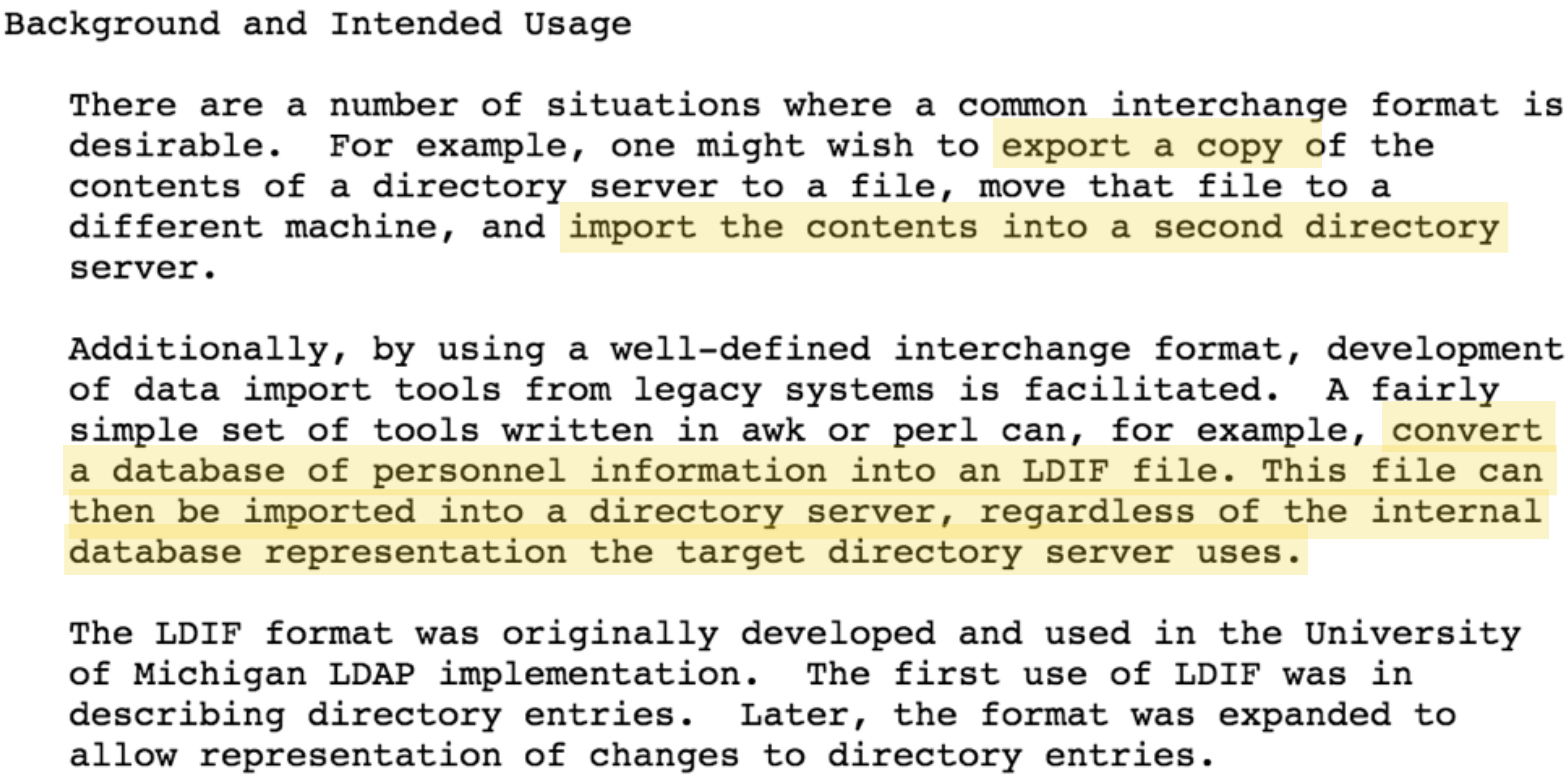
- **L**DAP **D**ata **I**nterchange **F**ormat
- http://tools.ietf.org/html/rfc2849



Abstract

```
   This document describes a file format suitable for describing
   directory information or modifications made to directory information.
   The file format, known as LDIF, for LDAP Data Interchange Format, is
   typically used to import and export directory information between
   LDAP-based directory servers, or to describe a set of changes which
   are to be applied to a directory.
```

LDIF



Background and Intended Usage

There are a number of situations where a common interchange format is desirable. For example, one might wish to export a copy of the contents of a directory server to a file, move that file to a different machine, and import the contents into a second directory server.

Additionally, by using a well-defined interchange format, development of data import tools from legacy systems is facilitated. A fairly simple set of tools written in awk or perl can, for example, convert a database of personnel information into an LDIF file. This file can then be imported into a directory server, regardless of the internal database representation the target directory server uses.

The LDIF format was originally developed and used in the University of Michigan LDAP implementation. The first use of LDIF was in describing directory entries. Later, the format was expanded to allow representation of changes to directory entries.

Example: LDIF to add an entry

```
dn: uid=john.doe,ou=People,dc=example,dc=com
changetype: add
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
uid: john.doe
givenName: John
sn: Doe
cn: John Doe
mail: john.doe@example.com
userPassword: password
```

Example: LDIF to modify an entry

```
dn: uid=john.doe,ou=People,dc=example,dc=com
changetype: modify
replace: description
description: This is the new description for John Doe
-
add: mailAlternateAddress
mailAlternateAddress: jdoe@example.com
```

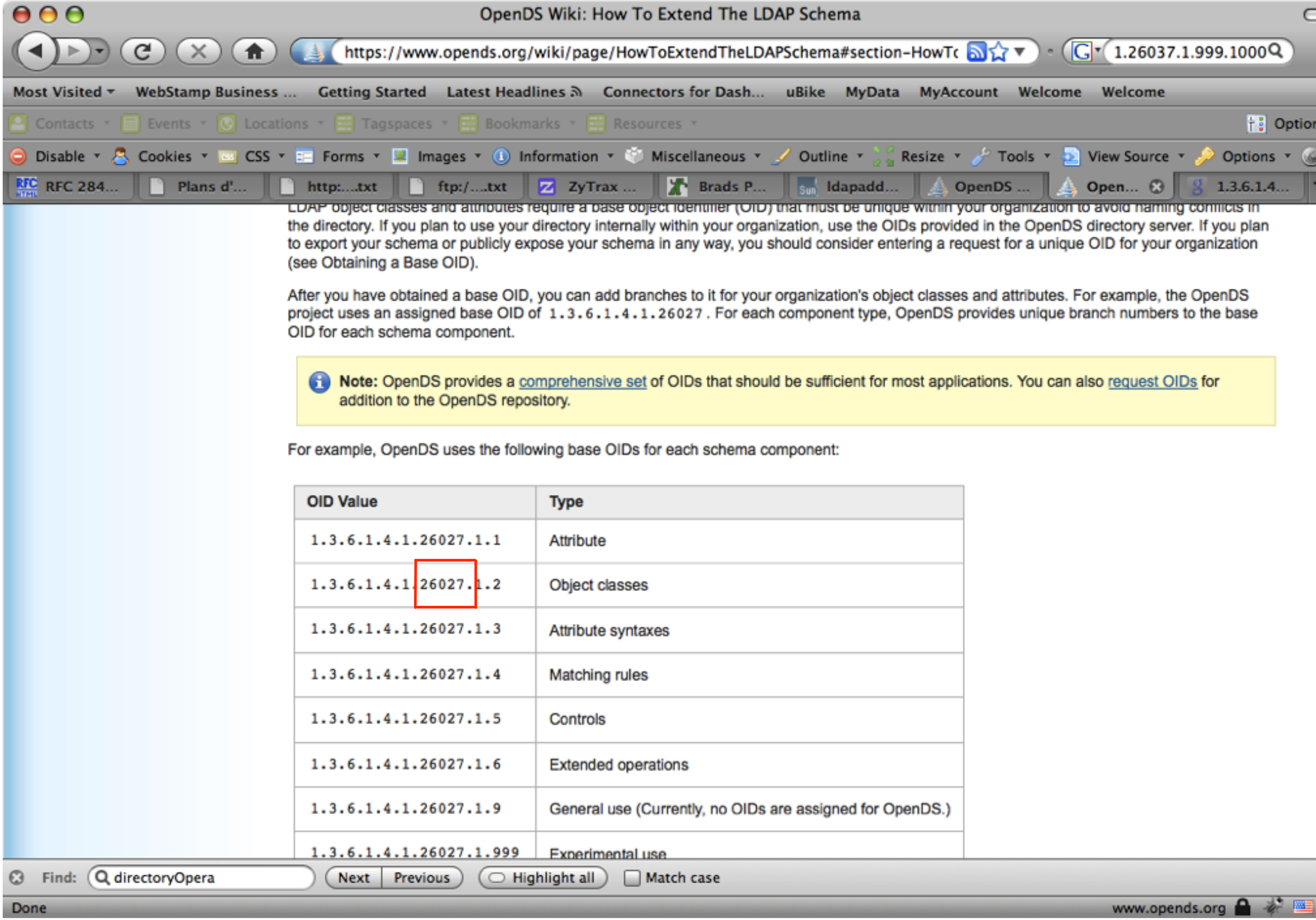
Example: LDIF to delete an entry

```
dn: uid=john.doe,ou=People,dc=example,dc=com
changetype: delete
```

Object IDentifier (OID)

- An OID is an **alphanumeric value** that **uniquely identifies** a particular element in a directory schema, such as a **class** or an **attribute**.
- There are **different ways to obtain an OID** for a schema element:
  - If the schema and directory data is used only for internal purposes, then you can freely define the OID value (as an analogy, think of a private IP network where you can decide for the addressing scheme yourself).
  - If the data is shared with external organizations, then a globally unique OID must be obtained (think of public IP addresses).
- OIDs are managed by the IANA; the procedure to obtain an OID is easy and simple.

https://www.opends.org/wiki/page/HowToExtendTheLDAPSchema#section-HowToExtendTheLDAPSchema-WorkingWithObjectIdentifiersOIDs



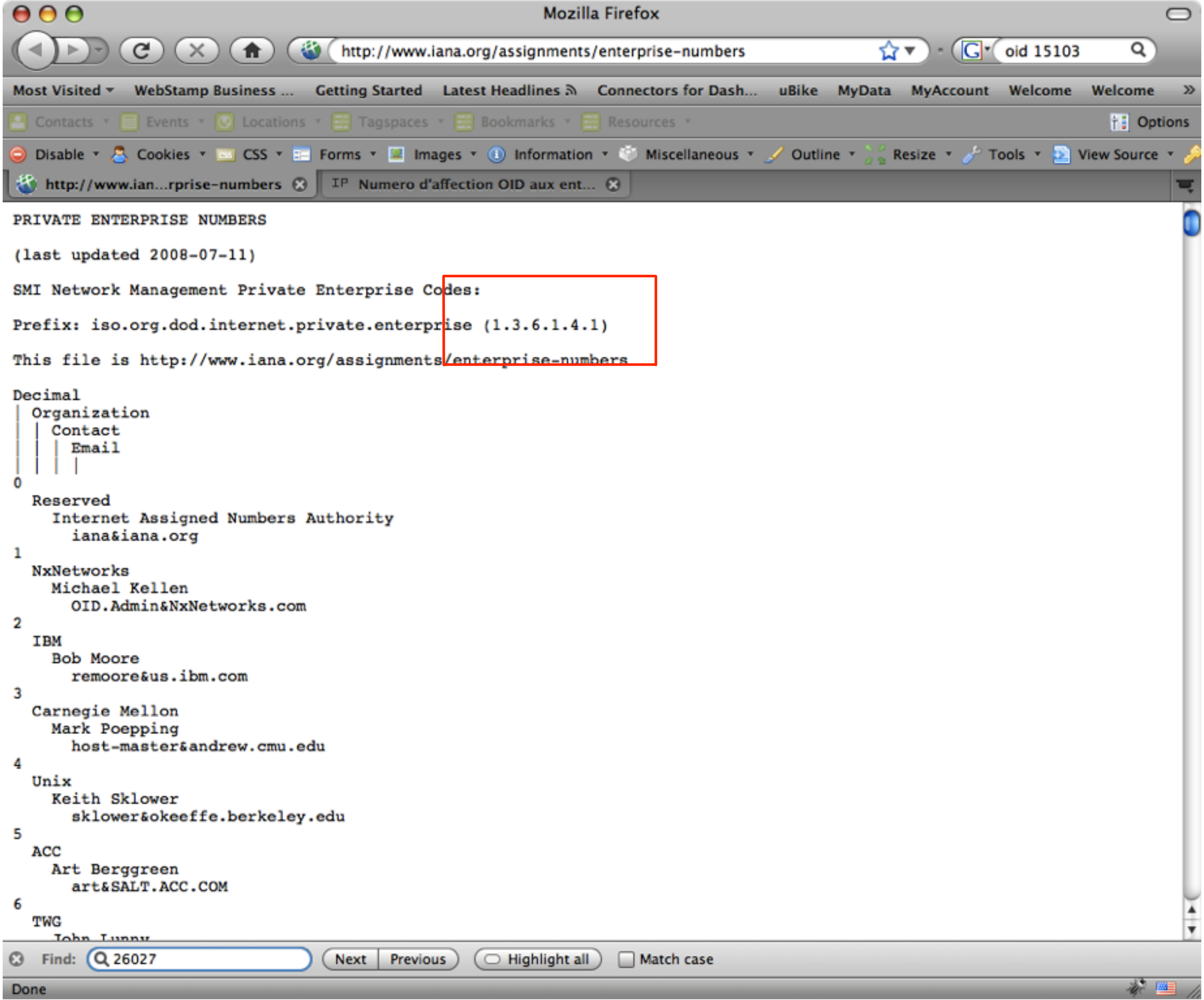
LDAP object classes and attributes require a base object identifier (OID) that must be unique within your organization to avoid naming conflicts in the directory. If you plan to use your directory internally within your organization, use the OIDs provided in the OpenDS directory server. If you plan to export your schema or publicly expose your schema in any way, you should consider entering a request for a unique OID for your organization (see Obtaining a Base OID).

After you have obtained a base OID, you can add branches to it for your organization's object classes and attributes. For example, the OpenDS project uses an assigned base OID of `1.3.6.1.4.1.26027`. For each component type, OpenDS provides unique branch numbers to the base OID for each schema component.

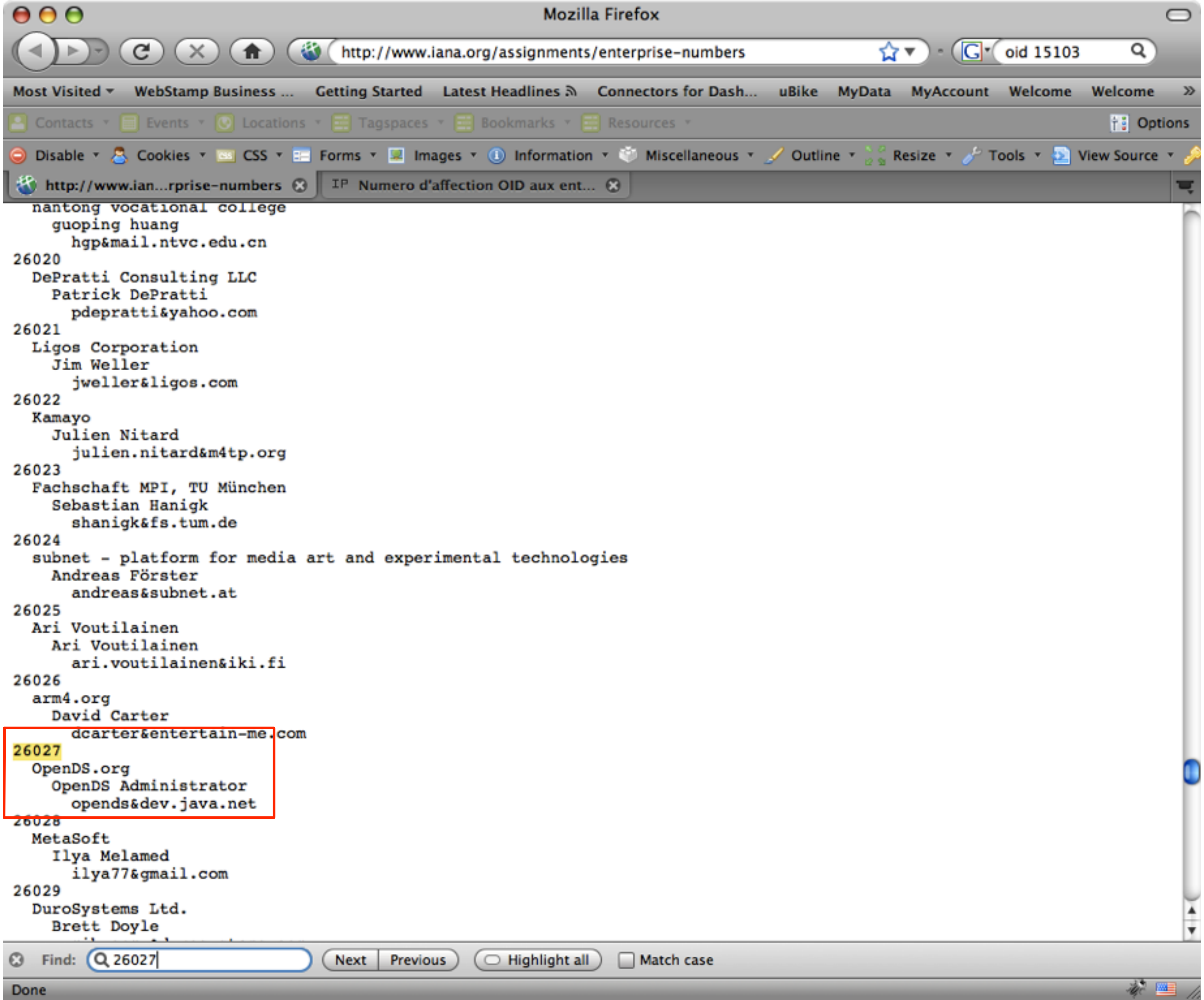
Note: OpenDS provides a comprehensive set of OIDs that should be sufficient for most applications. You can also request OIDs for addition to the OpenDS repository.

For example, OpenDS uses the following base OIDs for each schema component:

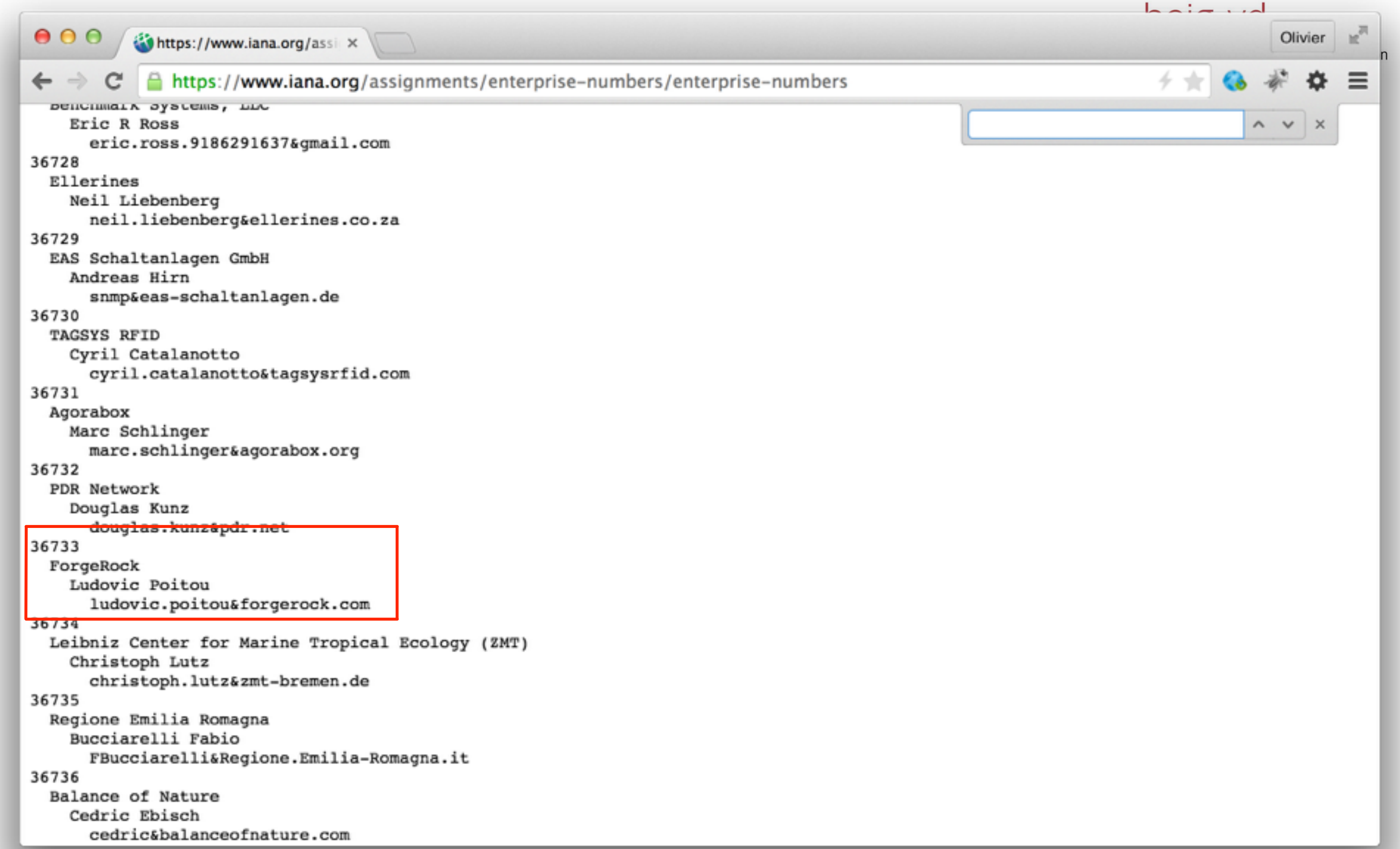
OID Value	Type
`1.3.6.1.4.1.26027.1.1`	Attribute
`1.3.6.1.4.1.26027.1.2`	Object classes
`1.3.6.1.4.1.26027.1.3`	Attribute syntaxes
`1.3.6.1.4.1.26027.1.4`	Matching rules
`1.3.6.1.4.1.26027.1.5`	Controls
`1.3.6.1.4.1.26027.1.6`	Extended operations
`1.3.6.1.4.1.26027.1.9`	General use (Currently, no OIDs are assigned for OpenDS.)
`1.3.6.1.4.1.26027.1.999`	Experimental use



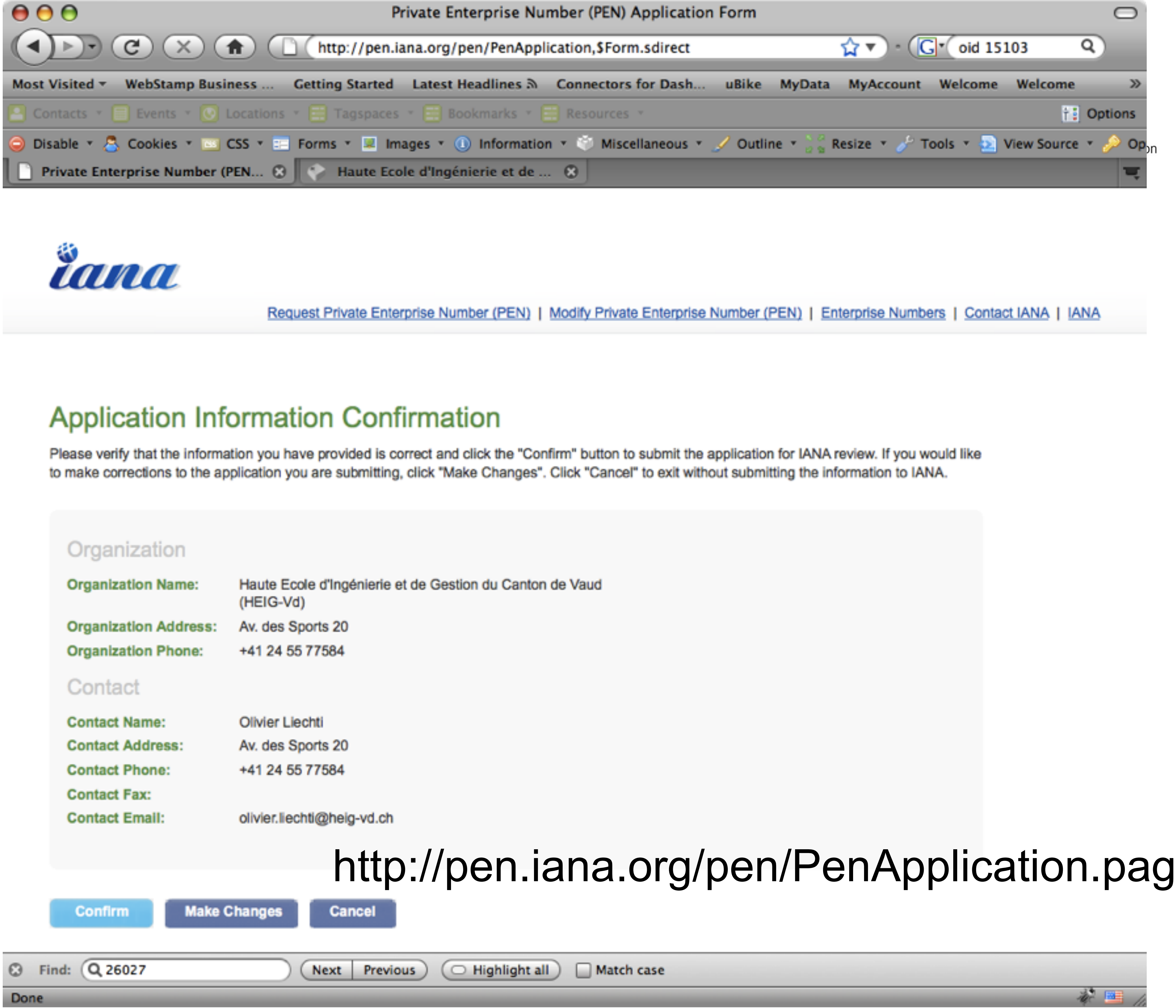
```
PRIVATE ENTERPRISE NUMBERS

(last updated 2008-07-11)

SMI Network Management Private Enterprise Codes:

Prefix: iso.org.dod.internet.private.enterprise (1.3.6.1.4.1)

This file is http://www.iana.org/assignments/enterprise-numbers

Decimal
| Organization
| | Contact
| | | Email
| | | |
0
  Reserved
    Internet Assigned Numbers Authority
      iana&iana.org
1
  NxNetworks
    Michael Kellen
      OID.Admin&NxNetworks.com
2
  IBM
    Bob Moore
      remoore&us.ibm.com
3
  Carnegie Mellon
    Mark Poepping
      host-master&andrew.cmu.edu
4
  Unix
    Keith Sklower
      sklower&okeeffe.berkeley.edu
5
  ACC
    Art Berggreen
      art&SALT.ACC.COM
6
  TWG
    John Lunny
```



```
  nantong vocational college
    guoping huang
      hgp&mail.ntvc.edu.cn
26020
  DePratti Consulting LLC
    Patrick DePratti
      pdepratti&yahoo.com
26021
  Ligos Corporation
    Jim Weller
      jweller&ligos.com
26022
  Kamayo
    Julien Nitard
      julien.nitard&m4tp.org
26023
  Fachschaft MPI, TU München
    Sebastian Hanigk
      shanigk&fs.tum.de
26024
  subnet - platform for media art and experimental technologies
    Andreas Förster
      andreas&subnet.at
26025
  Ari Voutilainen
    Ari Voutilainen
      ari.voutilainen&iki.fi
26026
  arm4.org
    David Carter
      dcarter&entertain-me.com
26027
  OpenDS.org
    OpenDS Administrator
      opends&dev.java.net
26028
  MetaSoft
    Ilya Melamed
      ilya77&gmail.com
26029
  DuroSystems Ltd.
    Brett Doyle
```


https://www.iana.org/assi
Olivier
https://www.iana.org/assignments/enterprise-numbers/enterprise-numbers

```
  Benchmark Systems, LLC
    Eric R Ross
      eric.ross.9186291637&gmail.com
36728
  Ellerines
    Neil Liebenberg
      neil.liebenberg&ellerines.co.za
36729
  EAS Schaltanlagen GmbH
    Andreas Hirn
      snmp&eas-schaltanlagen.de
36730
  TAGSYS RFID
    Cyril Catalanotto
      cyril.catalanotto&tagsysrfid.com
36731
  Agorabox
    Marc Schlinger
      marc.schlinger&agorabox.org
36732
  PDR Network
    Douglas Kunz
      douglas.kunz&pdr.net
36733
  ForgeRock
    Ludovic Poitou
      ludovic.poitou&forgerock.com
36734
  Leibniz Center for Marine Tropical Ecology (ZMT)
    Christoph Lutz
      christoph.lutz&zmt-bremen.de
36735
  Regione Emilia Romagna
    Bucciarelli Fabio
      FBucciarelli&Regione.Emilia-Romagna.it
36736
  Balance of Nature
    Cedric Ebisch
      cedric&balanceofnature.com
```


Private Enterprise Number (PEN) Application Form
http://pen.iana.org/pen/PenApplication,$Form.sdirect
oid 15103
Most Visited
WebStamp Business ...
Getting Started
Latest Headlines
Connectors for Dash...
uBike
MyData
MyAccount
Welcome
Welcome
Contacts
Events
Locations
Tagspaces
Bookmarks
Resources
Options
Disable
Cookies
CSS
Forms
Images
Information
Miscellaneous
Outline
Resize
Tools
View Source
Op
Private Enterprise Number (PEN...
Haute Ecole d'Ingénierie et de ...
iana
Request Private Enterprise Number (PEN) | Modify Private Enterprise Number (PEN) | Enterprise Numbers | Contact IANA | IANA

Application Information Confirmation

Please verify that the information you have provided is correct and click the "Confirm" button to submit the application for IANA review. If you would like to make corrections to the application you are submitting, click "Make Changes". Click "Cancel" to exit without submitting the information to IANA.

Organization

Organization Name:	Haute Ecole d'Ingénierie et de Gestion du Canton de Vaud (HEIG-Vd)
Organization Address:	Av. des Sports 20
Organization Phone:	+41 24 55 77584

Contact

Contact Name:	Olivier Liechti
Contact Address:	Av. des Sports 20
Contact Phone:	+41 24 55 77584
Contact Fax:	
Contact Email:	olivier.liechti@heig-vd.ch

http://pen.iana.org/pen/PenApplication.page

How to Manage Your OIDs?

```
1.3.6.1.4.1.xxx.n.n.n
```

Prefix: iso.org.dod.internet.private.enterprise (1.3.6.1.4.1)

Prefix assigned by the IANA to the HEIG-Vd

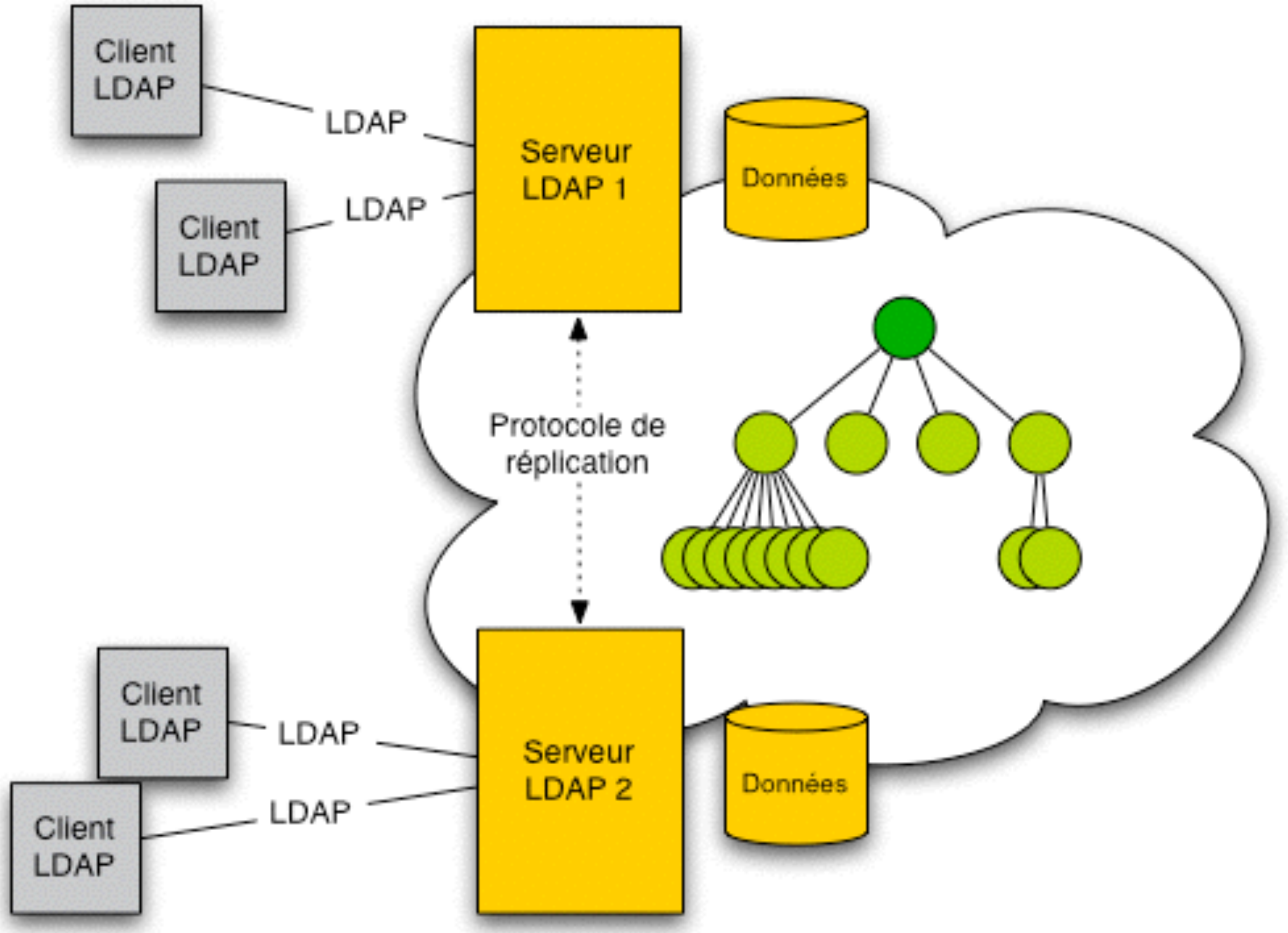
You define the rules for the suffix of the OIDs

```
.1.*: test
.2.*: teaching .2.1.*: PDA  2.2.*: RES
.3.*: research
.4.*: prod
```

Replication

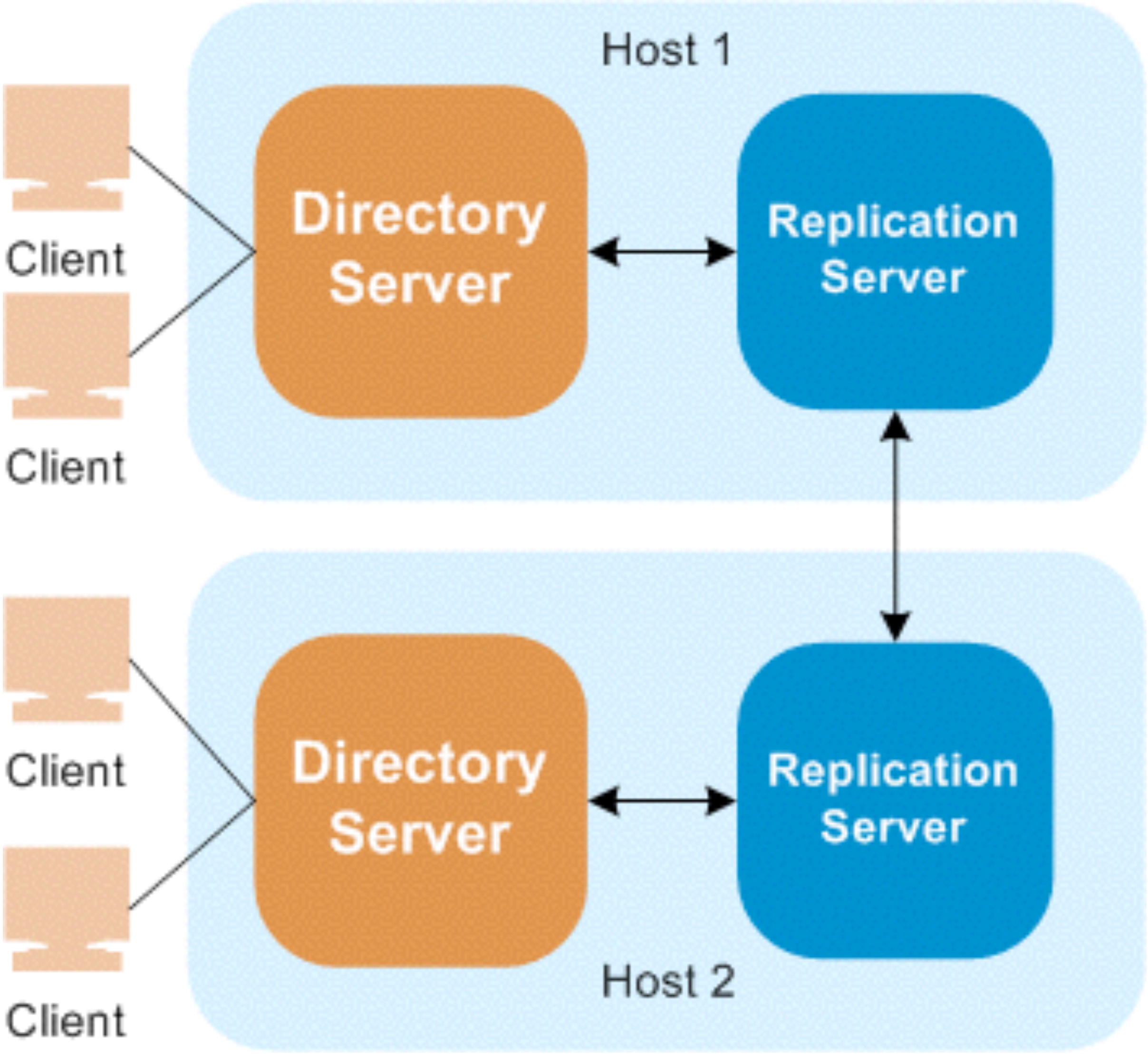
Principles

- What is LDAP "replication"?
  - Several LDAP servers are deployed to provide the directory service:
    - ➔ in the same data center (scalability, availability)
    - ➔ in different data centers, possibly in different countries (performance, latency)
  - Data are replicated (copied when updated) between the servers.
  - Clients can connect to the "most appropriate" server (either directly or via an LDAP proxy)



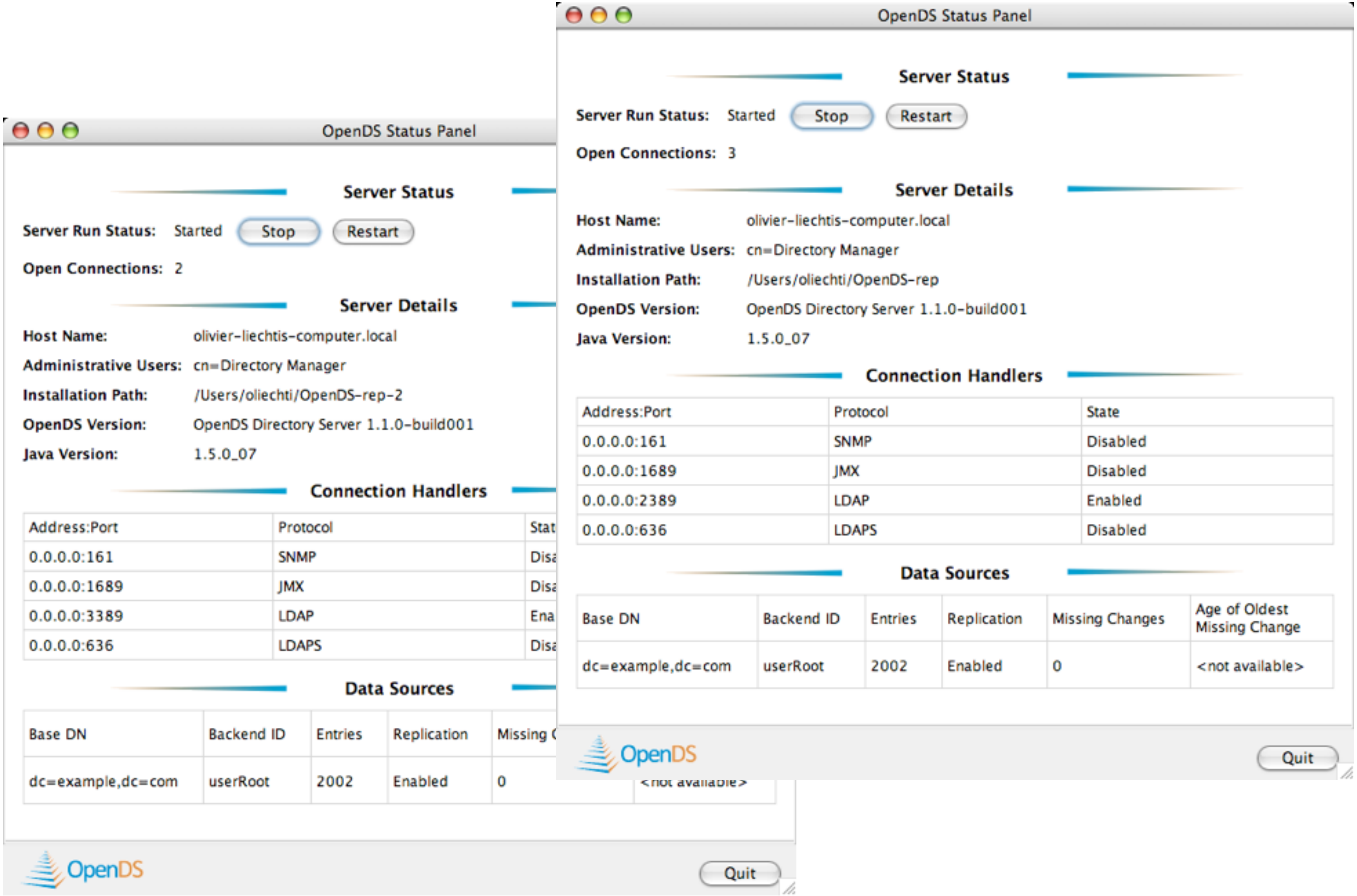
Principles

- Reasons for using LDAP replication
  - To ensure systemic qualities!
  - Performance
  - Scalability
  - Availability
- Different topologies are possible:
  - Single Master (1 server accepts write operations)
  - Muli Master (write operations can be submitted to multiple servers)



https://www.opends.org/wiki/page/SmallTopologies

http://www.sun.com/bigadmin/features/articles/dsee6_multimaster.jsp



OpenDS Status Panel

Server Status

Server Run Status: Started

Open Connections: 2

Server Details

Host Name: olivier-liechtis-computer.local

Administrative Users: cn=Directory Manager

Installation Path: /Users/oliechti/OpenDS-rep-2

OpenDS Version: OpenDS Directory Server 1.1.0-build001

Java Version: 1.5.0_07

Connection Handlers

Address:Port	Protocol	Stat
0.0.0.0:161	SNMP	Disa
0.0.0.0:1689	JMX	Disa
0.0.0.0:3389	LDAP	Ena
0.0.0.0:636	LDAPS	Disa

Data Sources

Base DN	Backend ID	Entries	Replication	Missing (	
dc=example,dc=com	userRoot	2002	Enabled	0	<not available>

OpenDS Status Panel

Server Status

Server Run Status: Started

Open Connections: 3

Server Details

Host Name: olivier-liechtis-computer.local

Administrative Users: cn=Directory Manager

Installation Path: /Users/oliechti/OpenDS-rep

OpenDS Version: OpenDS Directory Server 1.1.0-build001

Java Version: 1.5.0_07

Connection Handlers

Address:Port	Protocol	State
0.0.0.0:161	SNMP	Disabled
0.0.0.0:1689	JMX	Disabled
0.0.0.0:2389	LDAP	Enabled
0.0.0.0:636	LDAPS	Disabled

Data Sources

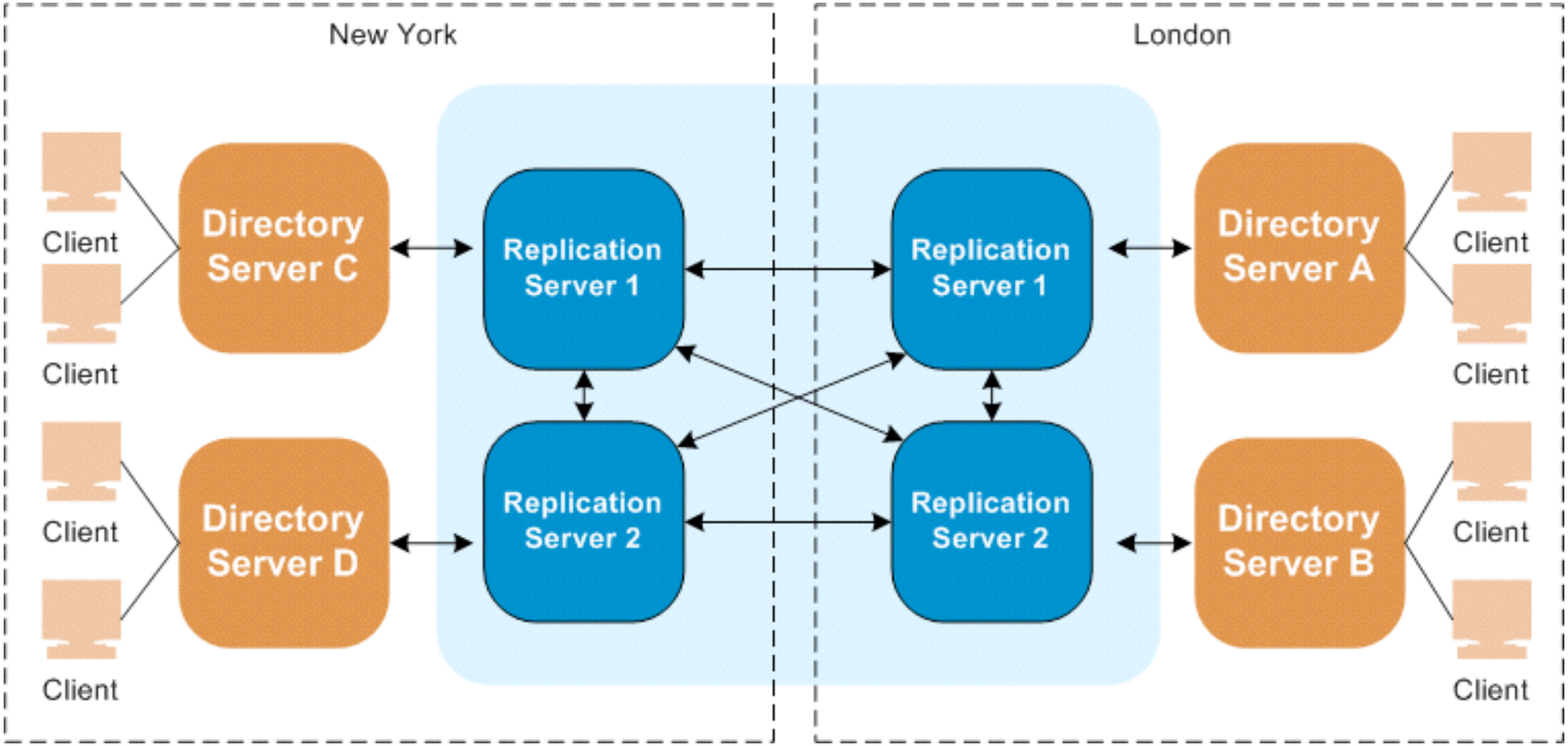
Base DN	Backend ID	Entries	Replication	Missing Changes	Age of Oldest Missing Change
dc=example,dc=com	userRoot	2002	Enabled	0	<not available>

Replication in OpenDS

```
$ lsof -P -i TCP | grep 89 | grep LISTEN
java      6145 oliechti   33u  IPv6 0x7a4c46c      0t0  TCP *:1389 (LISTEN)
java      9527 oliechti   40u  IPv6 0x798da24      0t0  TCP *:2389 (LISTEN)
java      9527 oliechti   46u  IPv6 0x79beaf0      0t0  TCP *:8989 (LISTEN)
java      9617 oliechti   41u  IPv6 0x7a2f174      0t0  TCP *:3389 (LISTEN)
java      9617 oliechti   46u  IPv6 0x7a2dde8      0t0  TCP *:9989 (LISTEN)
```

replication ports

Multi-Site Topology

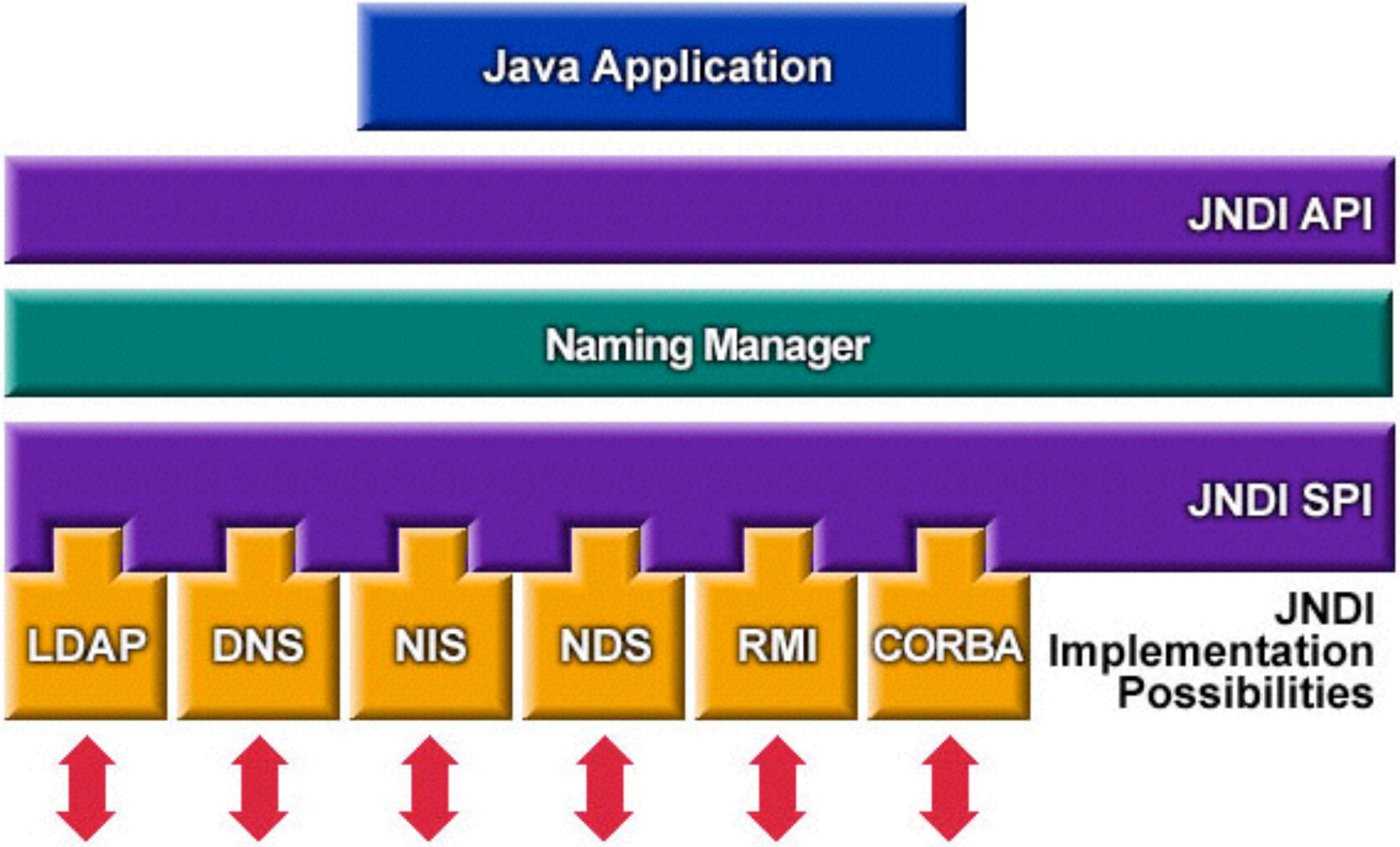


LDAP with Java
Java Naming & Directory Interface (JNDI)

Java Naming and Directory Interface (JNDI)

- LDAP is one of the application-level protocols that deals with data organized in a hierarchical data structure.
- Java developers would like:
  - a standard API that they can use for any protocol used to access hierarchical data (LDAP and others)
  - to be able to use this API to interact with any of the LDAP implementation (Active Directory, OpenDJ, OpenLDAP, etc.)
- In other words, they would like to have **the equivalent of JDBC** (used to talk to different relational database management systems in the same way), but for LDAP servers.
- JNDI is an answer to this need. The API provides a standardized API to interact with naming and directory services.

Java Naming and Directory Interface (JNDI)



http://java.sun.com/products/jndi/tutorial/getStarted/overview/index.html

How do I use JNDI?

- The first step consists of establishing a connection with the directory server.
- This is done with the `InitialDirContext` class:

```
// Set up the environment for creating the initial context
Hashtable env = new Hashtable();

env.put(Context.INITIAL_CONTEXT_FACTORY,
  "com.sun.jndi.ldap.LdapCtxFactory");

env.put(Context.PROVIDER_URL,
  "ldap://localhost:389/o=JNDITutorial");

DirContext ctx = new InitialDirContext(env);
```

How do I use the API?

> Once connected, the API provides abstractions to interact with the naming service.

> It is possible to navigate in the hierarchy, to access the entries and their attributes. It is also possible to submit LDAP filters via the API.

```
// Create the default search controls
SearchControls ctls = new SearchControls();

// Specify the search filter to match
// Ask for objects that have the attribute "sn" == "Geisel"
// and the "mail" attribute
String filter = "(&(sn=Geisel)(mail=*))";

// Search for objects using the filter
NamingEnumeration answer = ctx.search("ou=People", filter, ctls);
```

How do I use the API?

> Here is an example for iterating over all attributes of an entry, and then over all values of each attribute (remember that LDAP attributes can be **multivalued**).

```
// Search for objects using the filter
NamingEnumeration answer = ctx.search("ou=People", filter, ctls);

for (NamingEnumeration ae = answer.getAll(); ae.hasMore();) {
    Attribute attr = (Attribute)ae.next();
    System.out.println("attribute: " + attr.getID());

    /* Print each value */
    for (NamingEnumeration e = attr.getAll(); e.hasMore();
         System.out.println("value: " + e.next()));
}
```