The Integration of LDAP into the Messaging Infrastructure at CERN

Ray Jackson

CERN / IT-IS Group

29 Nov 2000 16:00

CERN IT Auditorium, bldg. 31, 3-005

A bit about me…

- Technical Student Sep 1997–1998 in PS Division working on Timing systems (designing API's in Java and C++)
- Manchester Met. University 1995–1999 studying computer science (main thesis in Java, VRML and HCI)
- Arrived in Internet Services Group of IT in June 1999
- Working on mail service, listbox service, news service and LDAP service (focus for today)

Roadmap

- Introduction to LDAP
- LDAP vs. Traditional Databases
- How we use LDAP today
- Future projects using LDAP

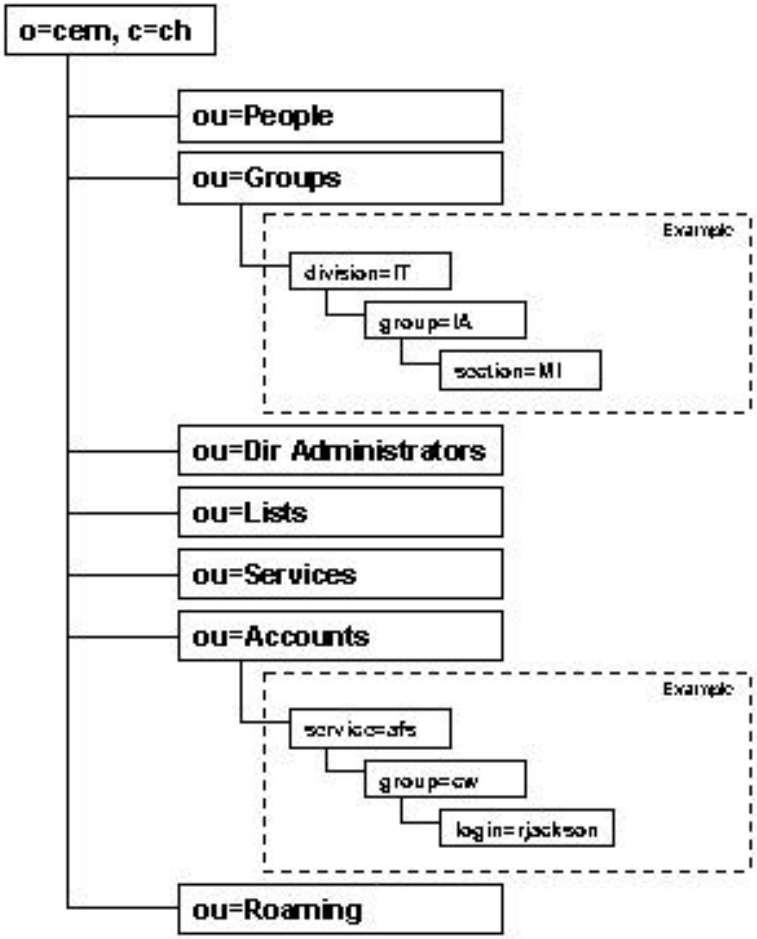
Introduction to LDAP

- "Lightweight Directory Access Protocol"
- Official Internet Standard Protocol for Accessing Directories (IETF)
- TCP/IP implementation of X.500 Information Model (Hierarchical, Attribute-Value)
- V3 Enhancements: Security, Distribution…
- Replaces proprietary protocols with an 'open' protocol (like SMTP & IMAP for e-mail)

How is LDAP organised?

- ‘Root’ (starting place/source of the tree)
- Countries (or TLD e.g. com,org,net)
- Organizations (CERN)
- Organizational units (departments etc.)
- Individuals (includes people, files, and shared resources such as printers)
  - e.g. cn=Ray Jackson,ou=People,o=cern,c=ch

Example of an LDAP tree



Why do we need LDAP?

- Everyone is using it already!
- Wide industry support (Microsoft, Novell, Netscape, Oracle etc.)
- The only successful ‘open’ DAP standard
- Simple, highly scalable, robust
- No viable ‘open’ alternatives

Powerful features of LDAP

- Very fast search/read access (5k+ p.s)
- Flexibility (design & implementation)
- Highly Scalable (using referrals)
- Platform independent
- Secure (v.3+ SSL, Kerberos)
- Broad industry support (MS Act Dir, Oracle, Novell, Netscape etc.)

LDAP operations

- Bind: Identify & authenticate client
- Search: Find entries matching criteria
- Add: Create a new entry
- Delete: Remove an entry
- Modify: Add,remove,modify an entry's attribute
- ModifyDN: Move an entry in the tree
- Others: Application specific operations…

LDAP vs. Relational Databases

- LDAP does NOT have transactions, rollbacks, multi-table queries, views & joins
- Greater speed & lower cost
- Few overheads, simpler data model
- Easier management & implementation
- Hierarchical rather than relational
- LDAP indexed for very fast searches/reads but slower writes (5000 reads, 50 writes p.s)

LDAP Schema definitions

- Objectclasses – A collection of attributes which make up an objectclass
- Attributes – A description of the type of data stored (e.g. givenName = cis / multiple)
- Standard & User defined. e.g. Object: person vs. cernperson & Attribute: cn vs CCID.
- Inheritance from superior objectclass
- Multiple or single allowed attributes
- Require vs. Allowed attributes

Example of an objectclass

```
objectclass person
    oid 2.5.6.6
    superior top
    requires
        sn,
        cn
    allows
        description,
        seeAlso,
        telephoneNumber,
        userPassword
```

LDIF the language of LDAP

- LDIF (LDAP Data Interchange Format)
- Used to create,remove and modify entries in an LDAP directory
- Very simple (text based definitions)
- Can store binaries (e.g. JPEG) in base64 encoding
- Usually used to initially build an LDAP directory and maintain via the command-line

Example using LDIF

```
dn: cn=Fred Bloggs, ou=People, o=exampleorg,c=ch
  objectClass: top
  objectClass: person
  objectClass: organizationalPerson
  objectClass: inetOrgPerson
  cn: Fred Bloggs
  sn: Bloggs
  givenName: Frederic
  mail: Fred.Bloggs@exampleorg.ch
  userPassword: {crypt}KDIE3AL9DK
  ou: Accounting
  ou: people
  telephoneNumber: 54321
  roomNumber: 220
```

Security in LDAP

- Access control information (ACI's usually linked to Group definitions)
- LDAPS protocol running on top of SSL
- Passwords stored in Unix crypt, SHA or text (user defined)
- Certificates (Public key cryptography)
- Plug-ins available (e.g. Kerberos)

LDAP at CERN

- Address Book and White Pages
- Address auto-completion
- Listbox Web Interface (SIMBA)
- Calendar Pilot Service (50+ users)
- Netscape Roaming Pilot Service (40+ users)
- Web authentication (Archives, interface)
- PAM authentication (System Level)
- Message routing in sendmail

CERN Address Books

- 32,000+ people (15,000+ external)
- Mixture of CCDB entries and Listbox users
- Mailing List & Services Addressbooks
- HEP Global addressbook (o=hep)
- Supported by Netscape, Pine, Eudora, Outlook and all major mail clients.
- Web based search engines (Currently test only – possibility of xwho data in future?)

CERN's Address Books

Netscape Address Book Feature

http://cern.ch/whitepages

SIMBA – Listbox Web Interface

- 2,200+ mailing lists stored on LDAP
- 700+ list owners 32,000+ list users
- ALL info (70+ attributes) related to mailing lists now on LDAP (members, configuration information etc.)
- Huge improvement on Mowgli (better security, more functionality etc.)
- Authentication for all 32,000 users using LDAP authentication
- LDAP makes searching for listbox data easy and fast!

SIMBA Listbox Web Interface

https://wwwlistbox.cern.ch

Web authentication & LDAP

- All major web servers can support LDAP for authentication (Apache, IIS, E'prise)
- Based on 'group' ACL's e.g. ou=it-div-is
- Simple to setup and configure (Used extensively in secure web archiving)
- Does not require physical accounts to be created on an OS. (few lines of LDIF only)
- ACL's can be easily created based on data in LDAP from CCDB and HR (e.g. division, group, status, mailing list membership etc.)

PAM authentication

- “Plugable-authentication Modules”
- Available for numerous UNIX platforms (Solaris, Linux, HP etc.) – pam.conf
- Can store most /etc data on LDAP (passwd/shadow, group, fstab, mail alias, protocol, rpc, service, host etc.)
- No duplication of accounts and group data across machines (synchronisation issues)
- Already used in authenticated SMTP service.
- Very useful for clusters of machines with identical configurations… more flexible than Sun’s NIS service as you can restrict individual machines.

Example of PAM data on LDAP

dn: cn=Ray Jackson,ou=People,o=cern,c=ch
objectclass: posixAccount
uid: rjackson
userpassword: {crypt}G51j29jsl09
loginshell: /usr/local/bin/bash
uidnumber: 416
gidnumber: 10
homedirectory: /homedir/r/rjackson
gecos: Ray Jackson
account: mail4 (← Not possible with NIS)

Message routing in sendmail

- Not just sendmail (Sun, Netscape etc.)
- Very fast lookups for mail routing
- Takes CPU load off the mail servers!
- Simple, dynamic and immediate updates
- Single source of routing data rather than distribution to 10+ machines
- Synchronisation and update delays eliminated
- Highly scalable (millions of addresses possible – ISP's using LDAP already for routing)

Example of routing in LDAP

dn: cn=Ray Jackson,ou=People,o=cern,c=ch

mail: Ray.Jackson@cern.ch

objectclass: inetLocalMailRecipient

mailHost: mail4.cern.ch

mailRoutingAddress: rjackson@mail4.cern.ch

mailLocalAddress: Ray.Jackson@cern.ch

mailLocalAddress: rjackson@mail.cern.ch

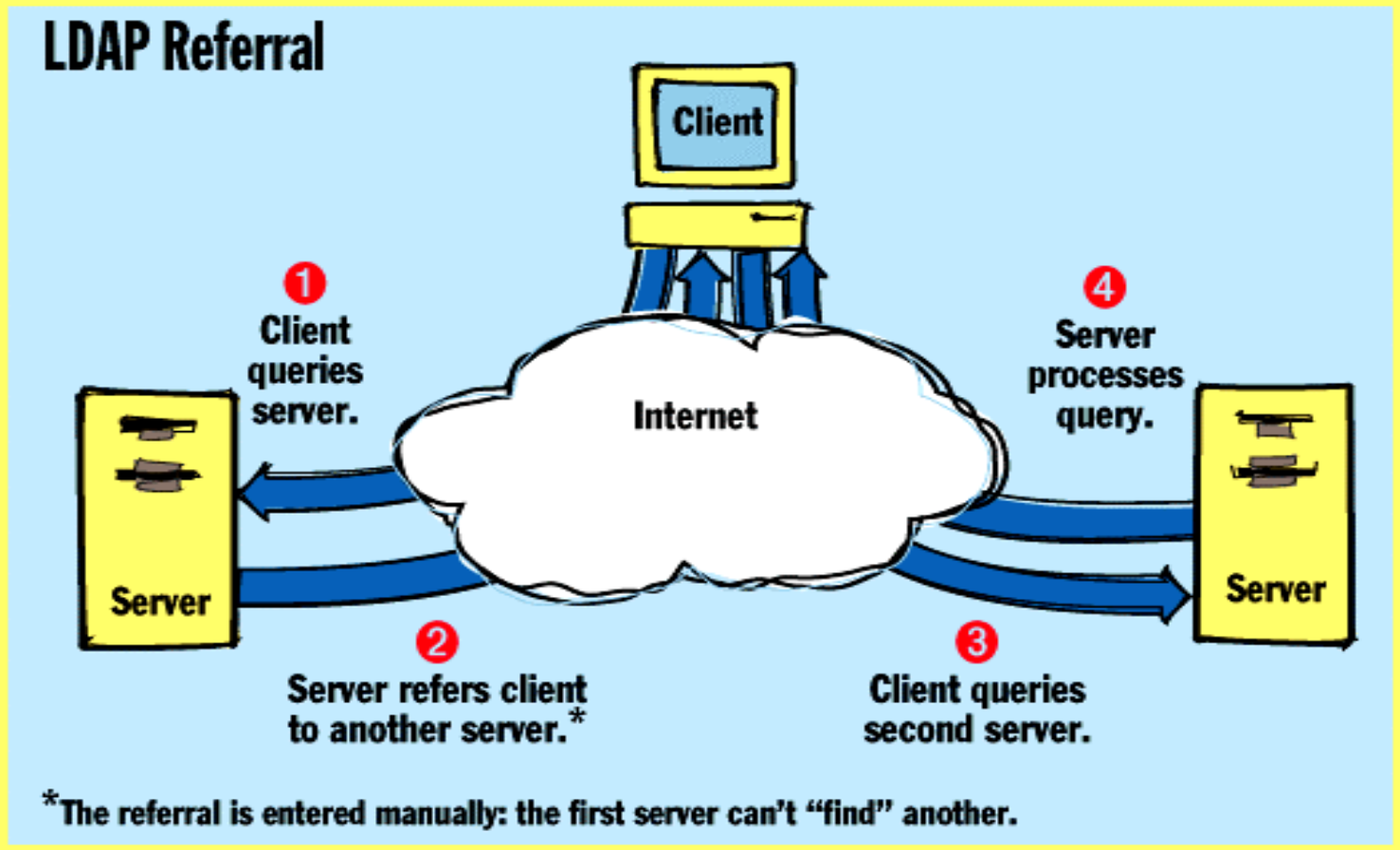
mailLocalAddress: Raymond.Jackson@cern.ch

mailLocalAddress: ldap.support@cern.ch

LDAP Referrals and Scalability!

- Referrals already used in HEP address book.
- All LDAP v3.x clients support referrals. (Netscape, Outlook etc.)
- Referral returns to client the address of another LDAP server to contact to fetch data.
- Completely transparent to user. (Sees single directory not concerned with multiple servers)
- Potentially scalable to millions of objects on dozens of servers.. Searches made in parallel.

Referral illustration



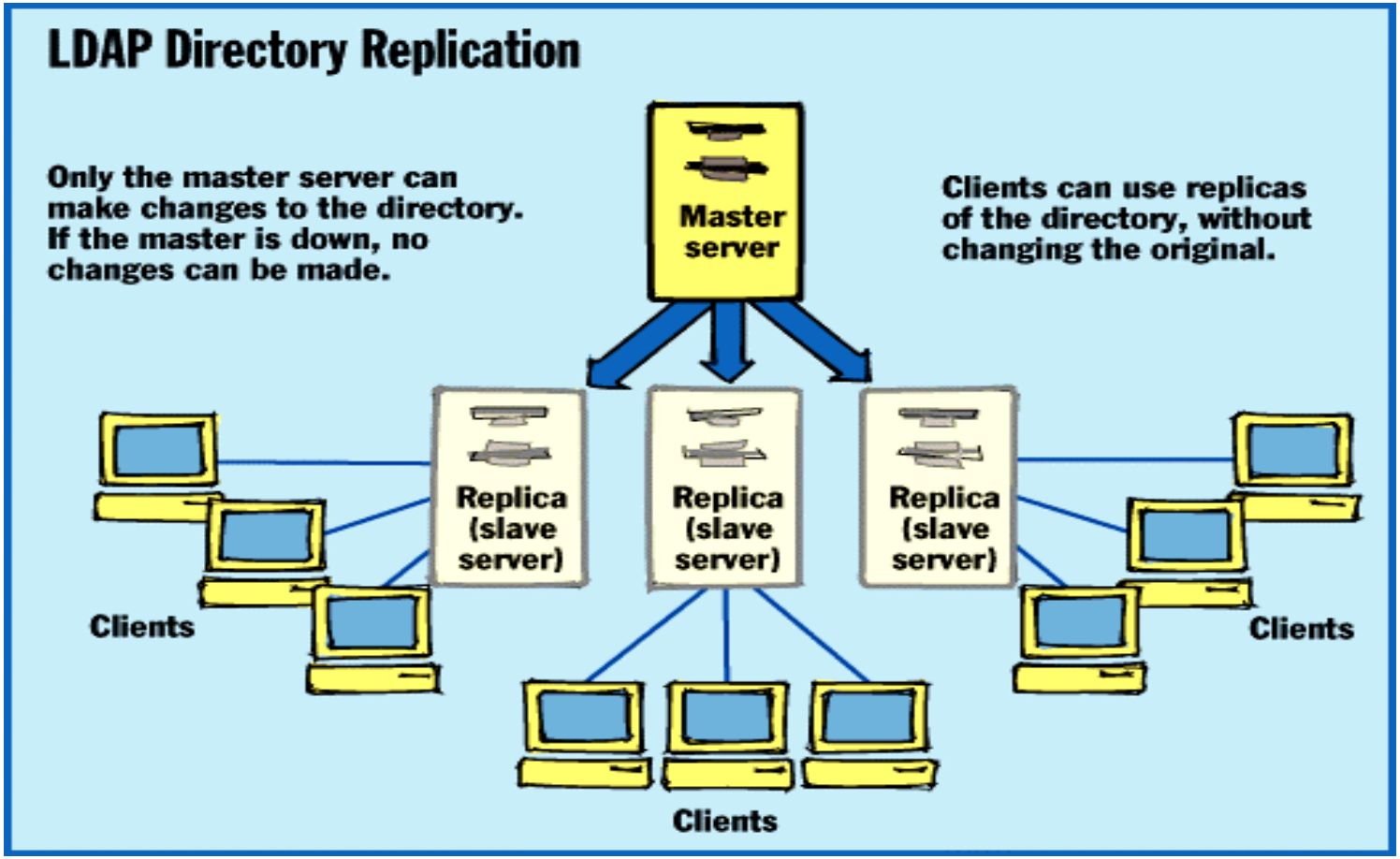
Use of referrals at CERN

```
o=hep
 |
 |------ o=cern
 |        |
 |        |------ ou=People
 |        |          e.g. ccid=49527,ou=People,o=cern,o=hep
 |        |               cn=Ray Jackson
 |        |               telephoneNumber=(+41) 22-767-1413    ...etc.
 |        |
 |        `------ ou=Groups
 |                  e.g. ou=IT,ou=Divisions,ou=Groups,o=cern,o=hep
 |
 |------ o=in2p3
 |          ref: ldap://ldap.in2p3.fr:389/o=in2p3,o=hep,c=net
 |
 |------ o=desy
 |          ref: ldap://mail2.desy.de:389/o=desy,c=de
 |
 |------ o=fnal
 |          ref: ldap://ldap.fnal.gov:389/o=fnal
 |
 : . . . o=slac?
```

Replication (slurpd)

- Replication and Indexing
  - Now standards exist for replicating data between different LDAP servers
- Changes on one server propagated to others. (Master to Slaves mechanism)
- Fault tolerance – Single point of failure so replication provides redundancy, transparancy & reliability
- Used with "DNS round-robin" you can provide a VERY reliable directory service and achieve load balancing.
- CERN – work in progress (LDAP1, LDAP2)

Replication illustration



Other applications of LDAP

- No limits to what can be achieved thanks to API's in Java, C, Perl etc.
- Store serialised Java objects on LDAP
- Hardware – Network routers etc.
- Shared Folders
- Archive Information (Catalog data)
- NT synchronisation with Unix for authentication etc.
- Any search/read intensive application can benefit from the power of LDAP

Conclusions

- LDAP is NOT a database but a protocol to access a directory service (Backend can be anything! – even a normal shell directory)
- LDAP is NOT useful for everything (i.e. cannot have rollback, transactions etc.)
- LDAP is VERY fast for searching/reading thanks to Indexing (MORE indexing means faster READS/SEARCHES but slower WRITES)
- LDAP is VERY useful when you wish to search for OBJECTS without knowing their location.
- LDAP is highly scalable AND very flexible!

The future is LDAP!

- Industry experts believe LDAP is key to any Inter-networked directory infrastructure
- LDAP is the ONLY protocol which interconnects different vendor-driven directory services
- All major vendors are pushing towards LDAP now (MS, Novell, Oracle, Sun, Netscape, IBM, HP etc.)
- Even hardware vendors are using LDAP in their products (Cisco use LDAP for routing)
- Ignore LDAP at your peril!!!

Future of LDAP at CERN

- Separating the service from the data!
- Move all user,listbox,group data OFF the 10+ mail servers and onto LDAP
- Eliminate the need for duplication of data and synchronisation problems.
- Retain backup ‘server’ side flat-files as a backup if LDAP goes wrong!
- Provide simple web access to mail information (inc. web-mail based on LDAP)

Thanks for coming!

Questions?