LDB and the LDAP server in Samba4

Simo Sorce
Samba Team

idra@samba.org
simo.sorce@quest.com
http://www.samba.org/~idra

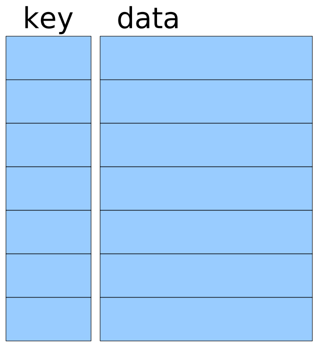
What is LDB ?

- LDB is an LDAP-like database interface
- LDAP-like data model
  - support LDAP like search expressions
  - but it is schema-less
- Modular
  - available backends uses TDB, LDAP, SQLITE3
  - modules stack over backends to provide extended functionality
- Fast and easy to manage indexing (TDB)
- Transactional (TDB,SQLITE3)

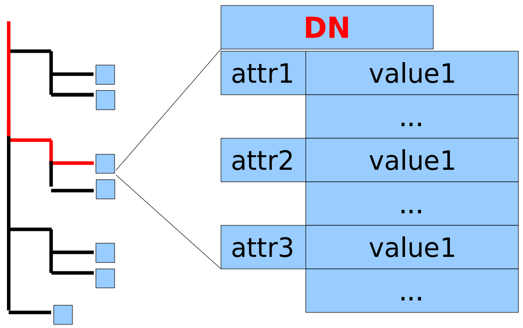
Why LDB ?

- TDB had a number of limitations
  - single key – single value mappings
  - every record is a binary object
  - no indexes, only a traverse function
  - programmers need to manually convert data structures to binary strings
  - programmers need to manually keep indexes if more than one index is needed
  - programmers need to manually check data endianess and handle structure upgrades

TDB

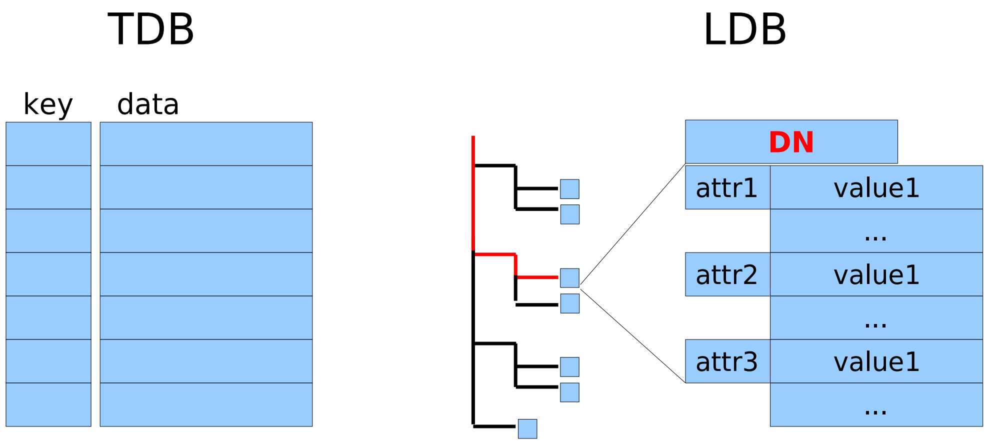


LDB



Why LDB ? (2)

- LDB has the advantages of an LDAP db
  - custom indexes
  - very powerful search strings
  - hierarchical
  - structures are easily modified or extended
- LDB has also the advantages of a TDB
  - fast searches
  - everything is kept in a single file (TDB, SQLITE3)
  - easy to backup



Simplified DB access

- All the complexity of handling hierarchical multivalued structured data in a database has been standardized and concealed behind an LDAP like API
- LDB takes care of building indexes for fast searches
  - when new indexes are added all the db is scanned automatically to rebuild them
- LDB does not need a schema
  - arbitrary attribute-value pairs can be stored in any object

LDB utilities

- LDB has a full set of user space utilities
  - ldbsearch
  - ldbadd
  - ldbdelete
  - ldbrename
  - ldbmodify
  - ldbedit
- Each command has a set of default switches:
  - mandatory:
    - -H ldb_url   choose the database (or $LDB_URL)

ldbsearch

```
An example: ldbsearch

$ ./bin/ldbsearch -H tdb://lib/ldb/test.ldb
'(&(objectclass=organizationalUnit)(ou=Groups))'
# returned 1 records
# record 1
dn: ou=Groups,o=Xsec,c=IT
objectclass: organizationalUnit
ou: Groups
```

- Syntax is quite similar to LDAP utilities
- The -H url defines the backend to be used
  - tdb, ldap, sqlite, ...
- File permission define access controls
  - Authentication is required against LDAP

ldbedit

- ldbedit is a powerful tool
  - it let you explore and change a snapshot of the directory in a text editor
  - it uses the well known ldif representation format
  - you can use it to backup and restore databases
  - you can use the text editor you prefer
  - you can choose to use a filter to edit a subset of objects in the database
  - be careful when editing the objects with option -a, do not touch “internal” objects unless you know exactly what you are doing
  - it works against an LDAP server too !!

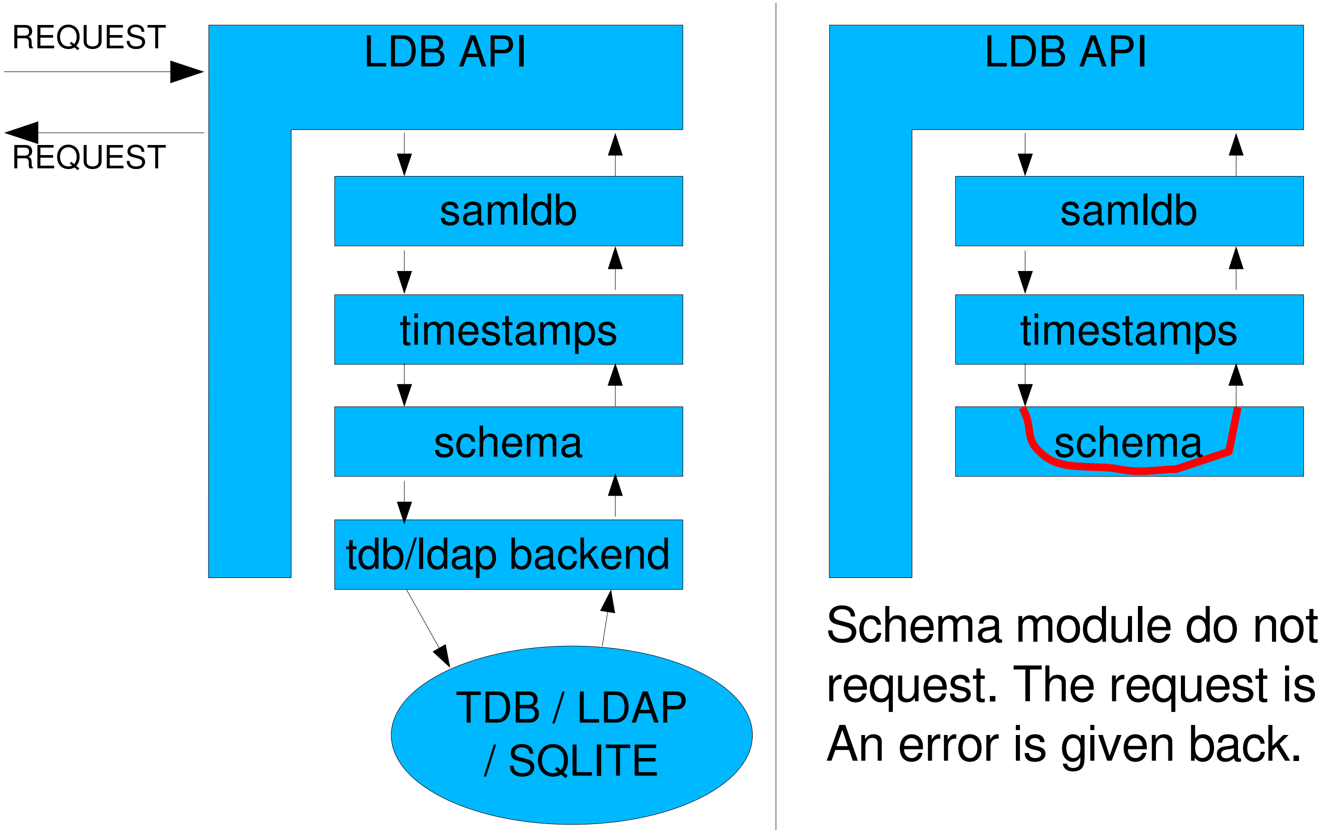
speacial DNs: @<something>

- dn names that start with an @ sign are special
  - the @ sign is used by reserved internal dn names
- you may set useful properties in these objects
  - indexes
    - the special dn @INDEXLIST controls indexing
  - case sensitivity
    - the special dn @ATTRIBUTES controls attributes behavior
  - class hierarchy
    - the special dn @SUBCLASSES is used to define subclasses
  - modules to be loaded
    - the special dn @MODULES set the list of modules to be loaded

Ho do I extend LDB ?

- LDB has a complete module stack
  - modules can intercept every ldb api call
  - modules are stacked, each module call the next one
  - a backend is just the last module that is called in the stack
  - modules can be loaded in the desired order (order often matters)
  - modules can be loaded automatically when opening an ldb file (tdb only)
- Samba4 heavily use ldb modules both internally and as part of the LDAP server

modules stack (simple schema)

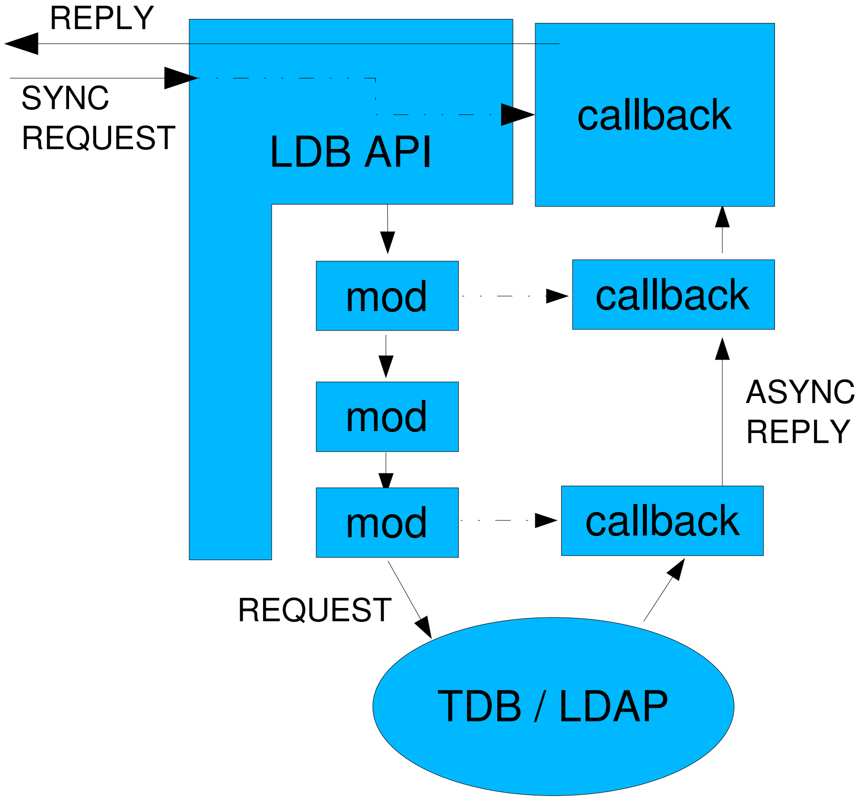


Schema module do not like the request. The request is not forwarded. An error is given back.

LDB async infrastructure

- Recently I have been working on a completely async infrastructure for LDB
  - callback based async calls
  - modules fully asynchronous
  - sync calls uses the async version underneath
  - needs an event system
- The work is mostly done
- To activate it in samba4 we still need:
  - some more testing and polishing
  - finish porting some modules (mostly done)
  - write tests that specifically stress the asynchronous aspect of the API

modules stack (async)



Loading modules

- How to make a module available to ldb once you made one?
  - currently you need to modify ldb_modules.c
  - We are experimenting DSOs so that you are able to load .so objects without any modification to the core LDB library
- How to activate a specific module in LDB?
  - through -o modules:modname,2nd,etc.. option
  - through the @MODULES special DN (TDB)
    - @LIST: samldb,timestamps,schema,...

Available modules

- List of LDB modules
  - asq
  - objectclass
  - operational
  - paged_results
  - rdn_name
  - server_sort
- List of samba4 specific modules
  - extended_dn
  - kludge_acl
  - object_guid
  - password_hash
  - rootdse
  - samldb

Controls

- The addition of modules support permitted us to easily implement controls in the LDAP server
- Controls are also used in the ldb utilities when talking to an LDAP server (-H ldap://...)
- A control is a structure attached to a request
  - used to change the behavior of the request
  - used to return additional info in a reply
- In LDB controls must be coded before use
  - to simplify their usability
  - to not depend on BER/ASN.1
  - this may change in future

why an LDAP server in samba4 ?

- AD has non-standard extensions to LDAP
- LDAP is tightly integrated in AD
  - we need one central DB to provide the same consistent data on all protocols
- why our own LDAP?
  - Building the LDAP server makes it easier to understand the AD LDAP behavior.
  - Modifying your own implementation is much easier than working with an external project
  - The line between LDAP and LDB is evanescent
  - no compatibility issues of sort
  - we can continue to provide bug for bug compatibility :-)

Current Limitations of LDAP srv

- ~~no asynchronous calls~~
  - only client side
- ~~no paged results~~
- ~~no transactions~~
  - not exposed
- not complete (no extended operations)
- still missing some controls
- no sub indexes
- no replication
- no ACLs
- primitive schema support

What do we need to do

- A lot of work on the replication protocols which involve more infrastructure
  - DRSUAPI (uses RPCs)
- Develop better tests to prove our conformance of LDAP to standards and to AD
  - Protocol conformance
  - Schema conformance
  - Authorization (ACLs) conformance
- Add more features in client libraries.
  - Support to follow referrals
  - Better usage of the rootdse informatio

Using LDB

- Can I use it ?
  - The Samba Team encourages people to use LDB in their own projects
- Where can I find it?
  - Currently it is available only by downloading the samba4 source code
  - A project to spin off some basic libraries like talloc, tdb and ldb is planned.
- Do I need to build and install samba4 to use it?
  - No, you can build LDB (with tdb and talloc) alone

Requisites

- What libraries does LDB depends on ?
  - libc
  - tdb
  - talloc
  - ldap libraries if you want to build the ldap backend in stand alone ldb, within samba4 we use the samba4 ldap libraries
  - sqlite3 libraries for the sqlite backend
- What kernel/OS can I use it on ?
  - most of our test has been done on linux kernel 2.4/2.6 using tdb as a backend
  - tdb needs well working locking (don't use it on nfs)

Licenses ?

- My Project has a Funny License, can I use LDB with it?
- Unlike the rest of the code in samba, LDB uses the GNU LGPL license instead of the GNU GPLv2
- This make it possible to:
  - use LDB in any GPL licensed program
  - use LDB with any other free software licensed program
- NOTE: not all modules are LGPLed
  - some modules under /lib/ldb/modules use LGPL
  - modules under other paths use the GPL

Links

- Source
  - samba4 source code:
    - svn co svn://svnanon.samba.org/samba/branches/SAMBA_4_0 samba4
- Developer resources
  - Mailing List:
    - ldb@samba.org
    - samba-technical@samba.org
  - IRC Channel:
    - #samba-technical on freenode.net

Questions ?