OpenLDAP Development

CN=config and Overlays
Howard Chu  hyc@symas.com
ODD/San Diego August 6, 2004
Status Summary

• Overlays: ABI refactoring from ODD March 21 2003
  – Enhancement of callback mechanism introduced in OpenLDAP 2.1
  – Successfully deployed in OpenLDAP 2.2
• Back-config is coming along slowly
Overlay Status

• Original Goals
  – Reduce redundancy
  – Streamline calling sequences
  – Enhance callbacks
  – Enable stacking/layering of backends
Overlay Status (2)

• Largely Successful
  – backend_group/backend_attribute consolidated, backend API continues to grow
  – Operation and SlapReply blocks yielded a measurable performance gain (~10%)
  – ProxyCache, Chain, Rewrite, and various other overlays have proven the viability of the Overlay concept
Overlay Status (3)

- The Overlay mechanism still has a ways to go
  - Not all the backend entry points are handled
  - None of the tool entry points are handled
  - We’d like layering to be totally transparent, but there are obvious issues in backend design that must be accommodated (e.g., back-ldbm deadlocks)
  - We’d like to reimplement SLAPI in terms of overlays – now that we have a global frontend, this is more feasible
CN=config Status

• Initial Goals
  – Convert config.c to table-driven mechanism
    • Maintain backward compatibility with existing slapd.conf syntax
    • Allow creation of a read-only CN=config tree for viewing the current configuration
    • Duplicates some information from CN=monitor
CN=config Status (2)

- Implement simple LDIF backend (back-ldif)
- Use overlay mechanism (surprise!) to implement modification semantics
- Limited initial support for modifications
  - ACL editing
  - Schema additions
- Modifications will be atomic and take effect immediately
CN=config Status (3)

• Proceeding Slowly
  – back-ldif done
    • Stores one entry per file, using filesystem to maintain tree structure
    • No indexing support
  – liblunicode consolidation
    • uadata hardcoded, so we can bootstrap using LDIF
  – Global variable consolidation
    • New “frontend” mechanism
    • Store remainder of global config items in frontend->be_private
CN=config Bootstrapping

- CN=config overlay stack will be hardcoded
- Migration from slapd.conf:
  - If back-ldif directory exists, slapd.conf is ignored
  - Otherwise, read slapd.conf and write out to back-ldif
  - Alternatively, just write out a flat LDIF file
- Loading a flat LDIF config file
  - Just use slapadd with backend #0
  - Will use a new command line option (e.g. –F) to specify location of back-ldif data directory
CN=config Wishes

- **What about `include`?**
  - May not be necessary, given the back-ldif implementation

- **What about `moduleload`?**
  - It may be feasible to load and unload backends and overlays on the fly since there is a well-defined shutdown API
  - Other modules – syntax, password, etc. will need reworking
CN=config Wishes (2)

• Attribute index reconfiguration?
  – Could be done, by treating the attribute as unindexed until the indexing pass completes

• Backend database swapping?
  – Point a backend at a different filesystem location, on the fly, to allow replacing disks, etc.
  – Probably would require multiple LDAPModify requests; might be a candidate for LDAP Transactions

• Plenty more we haven’t thought of yet
Conclusions

- slapd Overlays are really cool
- back-config will be an Overlay
- We expected back-config to be a big effort
- We’re making progress…