



**ORACLE®**

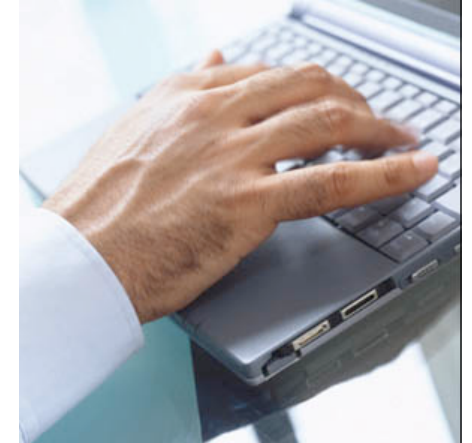


## **FedFS NSDB Draft Open Issues**

Chuck Lever  
Consulting Member of Technical Staff

# NSDB “To Do” Items

- NSDB-2
  - FedFS: Comment about fedfsFsIPort
- NSDB-3
  - FedFS: Question regarding fedfsNfsMajorVer and fedfsNfsMinorVer
- NSDB-4
  - FedFS: security concern regarding the use of time-variant UUIDs
- NSDB-5
  - (ITS#7246) Addition of FedFS schema LDIF to OpenLDAP



## NSDB-2: fedfsFslPort

- Question posted in October 2011
  - Should we keep the optional fedfsFslPort attribute in the fedfsFsl object class, or move it to subclasses?
- For NFS:
  - Language of RFC 3530 does not allow NFSv4.0 servers to communicate the service port in an fs\_locations list
  - Language of RFC 5661 specifically supports this via universal addresses
  - RFC 3530 bis “can be fixed” says Tom
- For others:
  - They may or may not allow specifying a port
- **Next step:** Maybe no NSDB draft change is needed?

## NSDB-3: fedfsNfsMajorVer

- Question posted in October 2011
  - Should we keep fedfsNfsMajorVer and fedfsNfsMinorVer attributes?
- Original intent was to help clients decide what server to choose out of an fs-locations list, but
- Neither fs\_locations nor fs\_locations\_info can convey version information to clients
- Recent list discussion: remove these attributes
- **Next step:** Remove these attributes from NSDB draft

## NSDB-4: fedfsUuid

- Question posted in October 2011
  - Should we drop recommendation to use time-variant UUIDs in favor of random-variant?
- Original intent was to encode provenance in UUIDs, but
  - Security-sensitive deployments may want better privacy
  - MAC address + time does not give good randomness any more
- Recent list discussion: go with type 4 UUIDs
- A change to the NSDB draft has been accepted
- **Next step:** Mark the to-do.txt item CLOSED.

# NSDB-5: General Schema Review

- In April, I asked the OpenLDAP community to consider including the proposed FedFS schema in the OpenLDAP distribution. Their comments:
  - The type of fedfsNcePrefix should be DN, and dispense with the notion of a DN prefix
  - LDAP already has a UUID type
  - Avoid storing XDR blobs in LDAP
  - Not clear why we need to define fs\_locations bit flags
  - Use URL format to specify network service locations

# How To Store fedfsPathName

- Pathnames stored in LDAP as XDR blobs
- Pros:
  - File servers don't need to unmarshal them to send them to clients in an fs-locations list
  - Does LDAP have an “ordered list of UTF-8 strings” attribute type?
- Cons:
  - They are unreadable when stored in LDAP
  - Not scripting-friendly
    - Marshaling is required before storing them in LDAP
    - Unmarshaling is required before displaying them to users

# fedfsNcePrefix

- Chu's recommendation
  - Change type of fedfsNcePrefix to DistinguishedName
  - Then constrain the attribute to be subordinate to the containerInfo entry
- Implementation issues:
  - When initializing an NSDB, administrative client has to construct these strings
    - Usually an LDAP client gets DN strings from the server, does not construct them on its own
  - LDAP server administration varies widely among vendors



# LDAP UUIDs

- We should use existing RFC 4530 UUID type
  - Eliminate fedfsUuid
  - fedfsFsnUuid and fedfsFslUuid can inherit directly from UUID
  - Reorder FedFS OIDs
- Not all LDAP servers support RFC 4530
  - OpenLDAP has UUID, but
  - 389-ds has nsUniqueld

# Storing fs\_locations\_info bit flags

- Chu was not clear why we were defining Trans and Class bit flags
  - Is this simply a matter of more carefully referencing RFC 5661 in the NSDB draft, or
  - Is there a better approach for representing file-system protocol specific data in LDAP?

# Considering An NSDB URL Format

- Should we express file server name and port information in URI format?
- Should we express NSDB service information in URI format?
  - No transport information contained in the currently proposed NSDB name/port
  - However, this *is* specified on the file server end via NSDB connection parameters
  - A URL format can specify an inward-facing NSDB (*i.e.*, `ldapi://` )



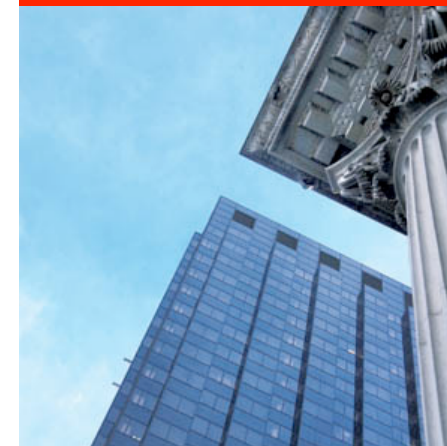
# Next Steps

- Come to consensus on the individual NSDB-5 items
- I volunteer to provide updates to Tom

# Review Process Considerations

- The NSDB-5 issues should have been caught earlier
- Does the WG have an adequate review process for the LDAP pieces of the FedFS NSDB draft?
  - Standards: LDAP experts within the IETF
  - Implementation experience
    - LDAP server and client developers
    - FedFS implementations

# Appendix



# Linux FedFS Implementation Status

- FedFS delivered via separate package from nfs-utils to reduce complexity of dependencies
  - LDAP, XML, sqlite introduced
- nfs-utils updates required:
  - mountd support for resolving junctions
- Kernel updates required:
  - NFSD must recognize junctions and perform an upcall
  - NFS client already handles referrals

# Linux FedFS Implementation Status

- fedfs-utils 0.8 released February 2012
  - Autofs program map to discover and mount domain roots
  - DLL provides mountd support
  - rpc.fedfsd on NFS servers
  - NSDB connection parameter database
  - Solaris-like nfsref tool for managing referrals
  - Scads of admin tools and man pages
- 0.9 planned for 1CQ13
  - LDAP TLS and referral support
  - Tools to convert LDAP service into NSDB
  - Latest DNS SRV format
  - Latest FedFS NSDB schema