# z/OS LDAP Usage and Demonstration (SHARE Session 1722)

Tim Hahn
IBM OS/390 LDAP Development
hahnt@us.ibm.com

# Why use a Directory?

- Provides a place to store information that is accessible from multiple locations
- Provides a place to look up where to find other information or servers
- Provides a place to make information accessible to multiple applications
- If you have information that needs to be managed centrally but used across your enterprise, a directory can help

# What can be stored in a Directory?

- Directories can store just about any type of information
- Basic data types are string, integer, boolean, and binary
- Binary data can range from a few bytes to megabytes in size
- Directories are usually tuned to favor high read rates at the expense of lower write (add/modify/delete) rates
- Store information in the directory that is relatively static but used across your application environment (enterprise, e-business applications, etc.)

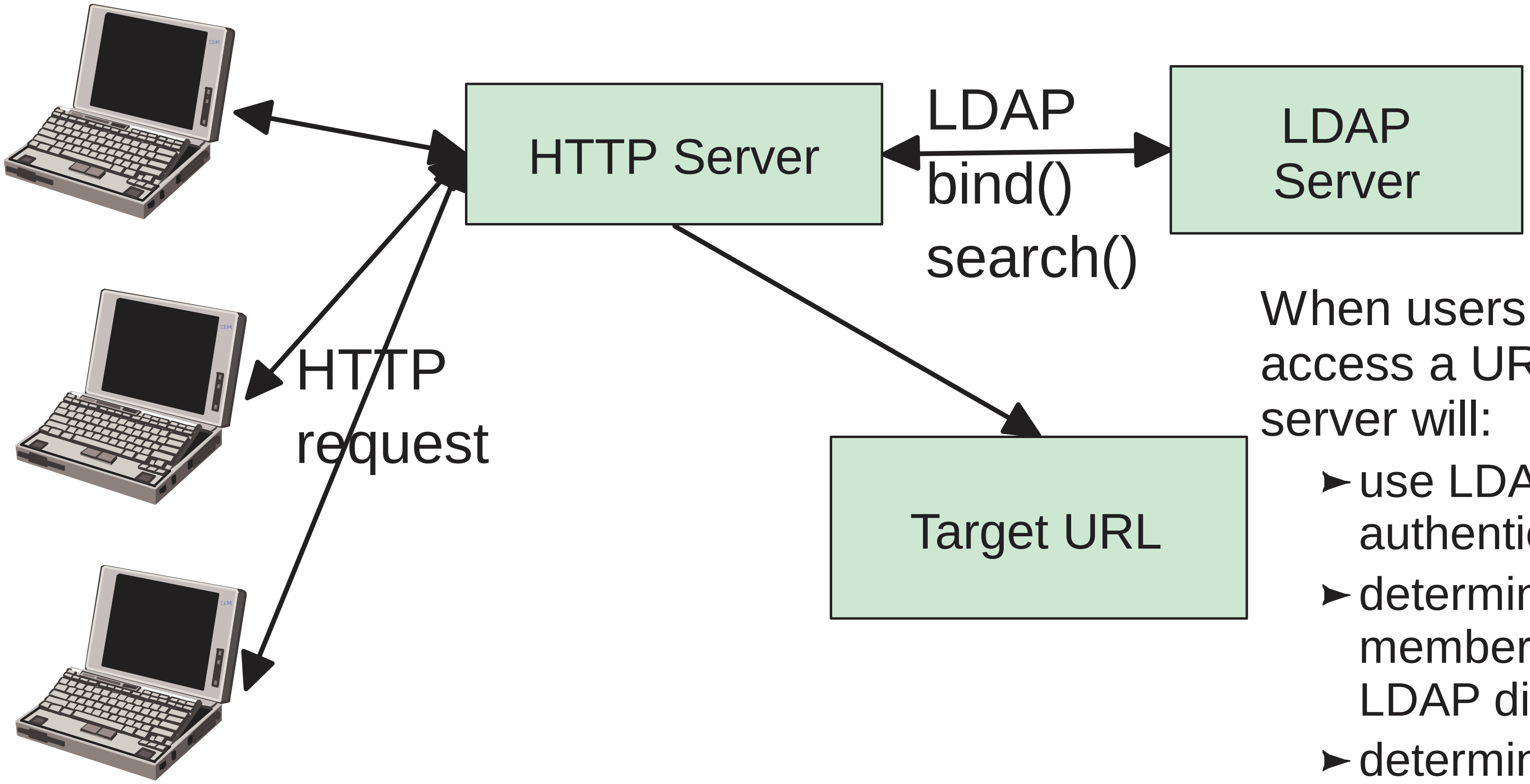
# What types of applications use a directory?

- Single sign-on frameworks
- Enterprise phone books
- Distributed access control checkers
- Centralized configuration database
- Distributed object look-ups
- Web application personalization
- Directory for PKI environments (certificates and CRLs)

# LDAP Usage in the Enterprise

- HTTP server Authentication and Access Control
- Websphere EJB Naming
- Tivoli SecureWay Policy Director User Registry
- IBM "Bluepages" internal phone book

# HTTP Server Authentication and Access Control



HTTP Server

LDAP bind() search()

LDAP Server

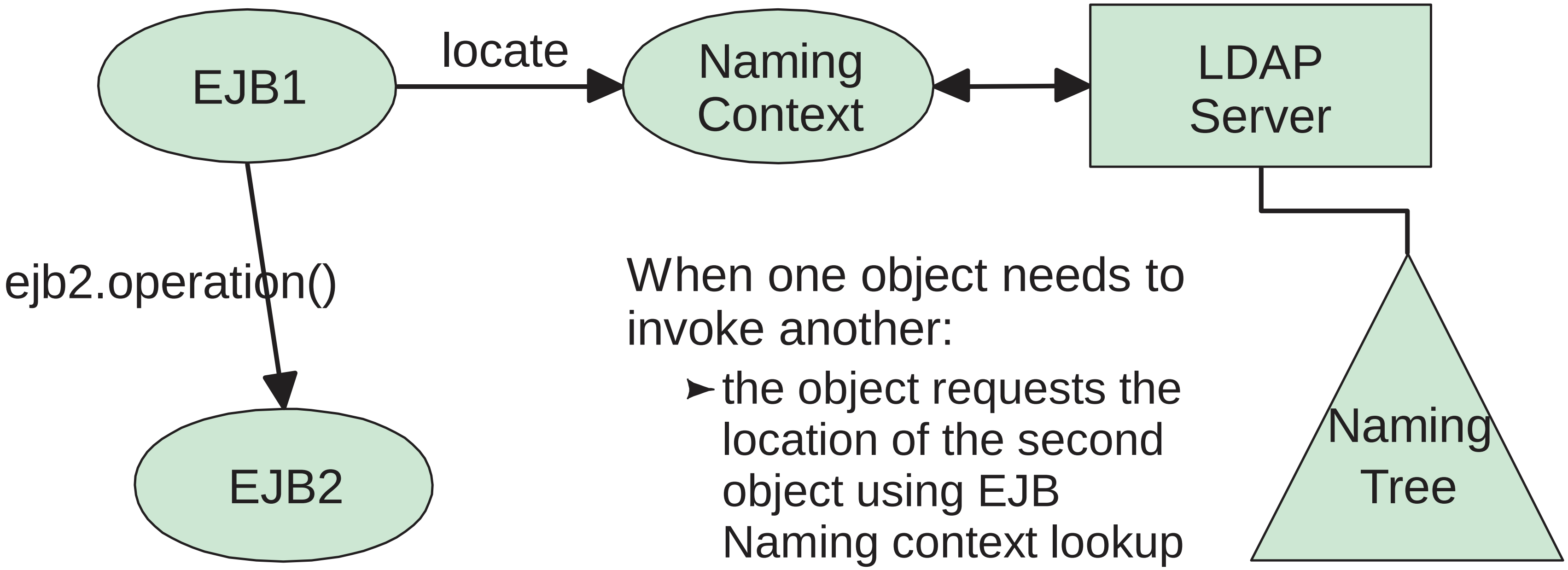
HTTP request

Target URL

When users attempt to access a URL, the web server will:

- ➤ use LDAP directory for authentication
- ➤ determine group membership based on LDAP directory
- ➤ determine access to URL

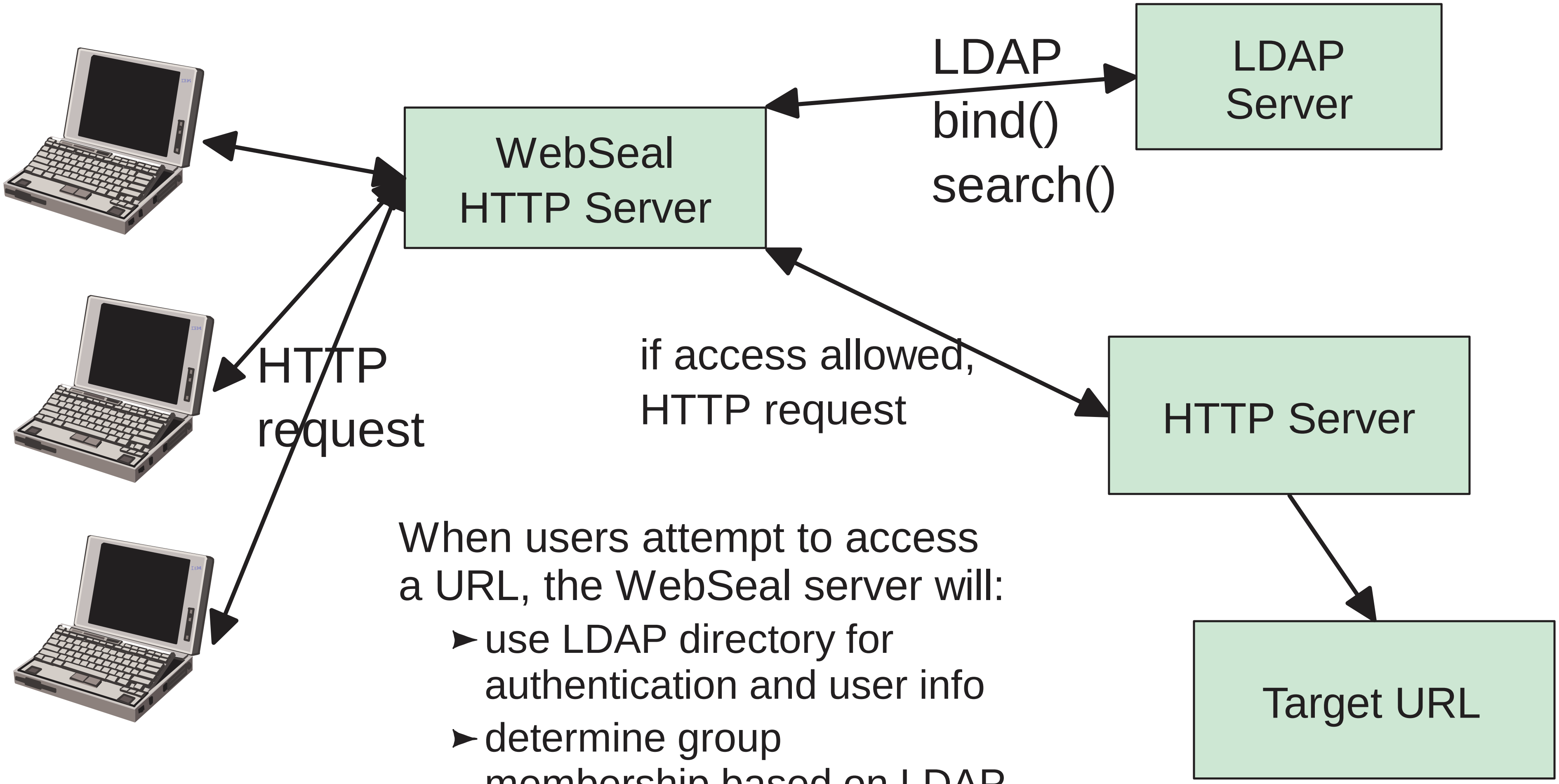
# Websphere EJB Naming



When one object needs to invoke another:

- the object requests the location of the second object using EJB Naming context lookup
- Naming context returns the second object location
- first object calls the second object method

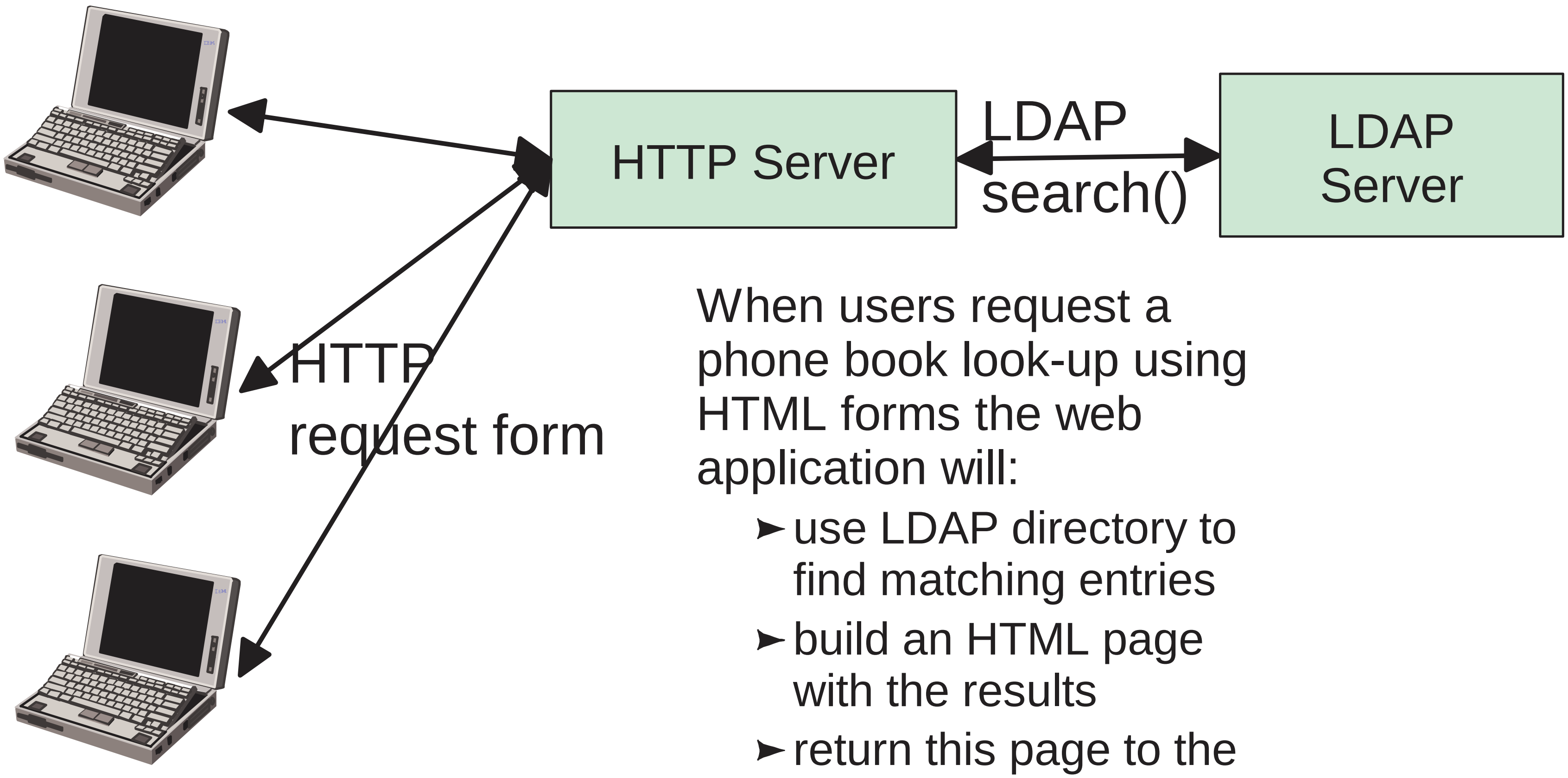
# Tivoli SecureWay Policy Director User Registry



When users attempt to access a URL, the WebSeal server will:

- ➤ use LDAP directory for authentication and user info
- ➤ determine group membership based on LDAP directory
- ➤ determine access to URL

# IBM "Bluepages" Internal phone book



When users request a phone book look-up using HTML forms the web application will:

- ➤ use LDAP directory to find matching entries
- ➤ build an HTML page with the results
- ➤ return this page to the user

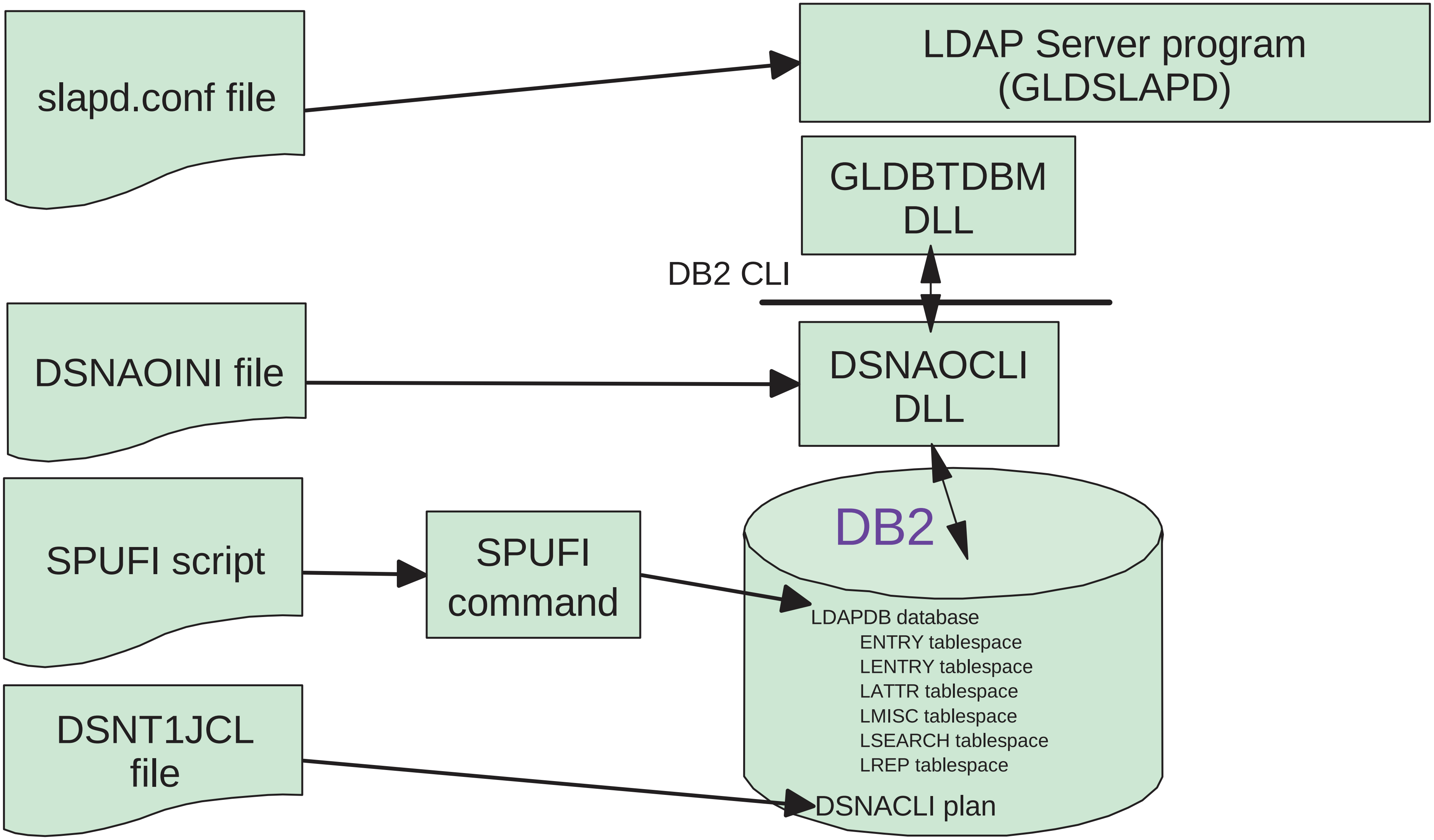
# What are we going to do?

- Create a LDAP server on z/OS
- Start the server
- Add some information
- Query this information using a variety of tools:
  - LDAP Browser
  - Directory Management Tool
  - Netscape Browser
  - Lotus Notes
- Add some new directory schema (data formats)
- Add some more information
  - Use an application to query this new information

# Create a LDAP server on z/OS

- Multiple options available to do this:
  - ldapcnf utility
  - copy and modify the "sample server" in `/usr/lpp/ldap/examples/sample_server`
  - manual modification of SPUFI, LDAP Server started task, LDAP server configuration file
- I'll briefly touch on the the resultant files needed since we're using an already configured server

# Configuring the LDAP server

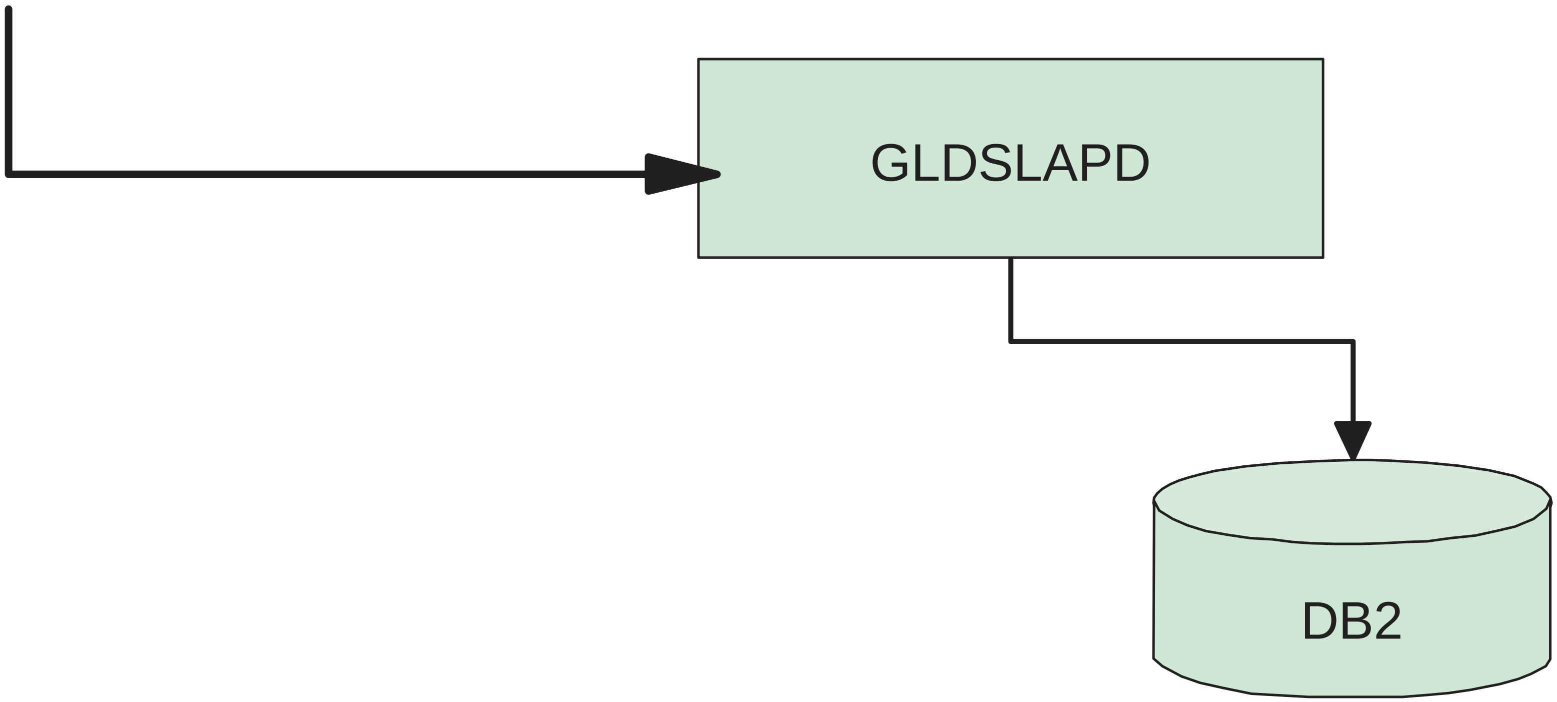


# Starting the LDAP server

- Multiple choices for starting/running the LDAP server:
  - As a started task (place LDAPSRV PROC in PROCLIB)
  - As a long-running batch job
  - As a USS background process
- I'll show a "long-running batch job" since this approximates running as a started task

# Starting the LDAP server

```
READY
submit ( MYJCL(LDAPSRV1) )
```



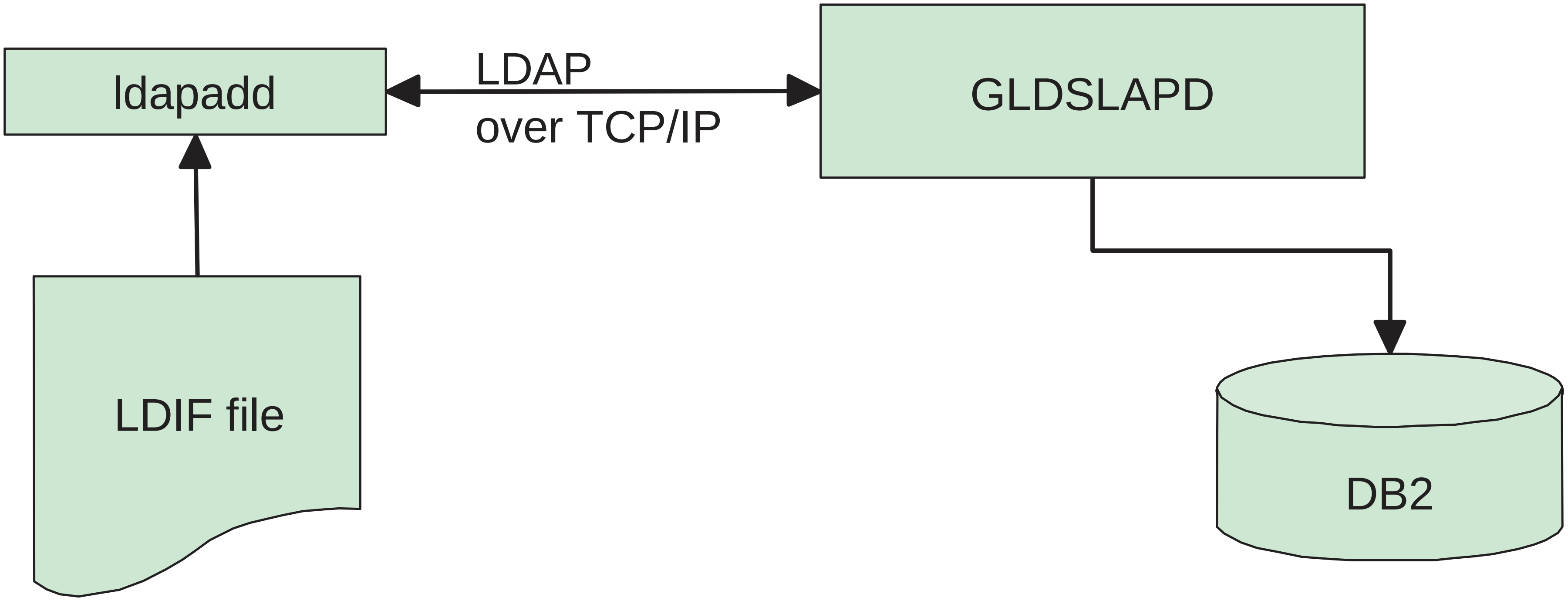
# Starting the LDAP server

- Since this is a new server, some additional information must be added to the now running server:
  - Initial schema information:
    ldapmodify ... -f schema.user.ldif
    ldapmodify ... -f schema.ibm.ldif
  - Initial suffix data:
    ldapadd ... -f suffix.ldif
- Now we can add some information to the directory!

# Adding information to the Directory

- There are a couple of choices for adding information to the directory:
  - bulkload (ldif2tdbm tool) - for adding large amounts of information
  - ldapadd - for adding smaller amounts of information
- I'll use ldapadd from my workstation (this command-line tool is shipped with most "LDAP client" installations). This tool exists on z/OS as well (see the previous slide).

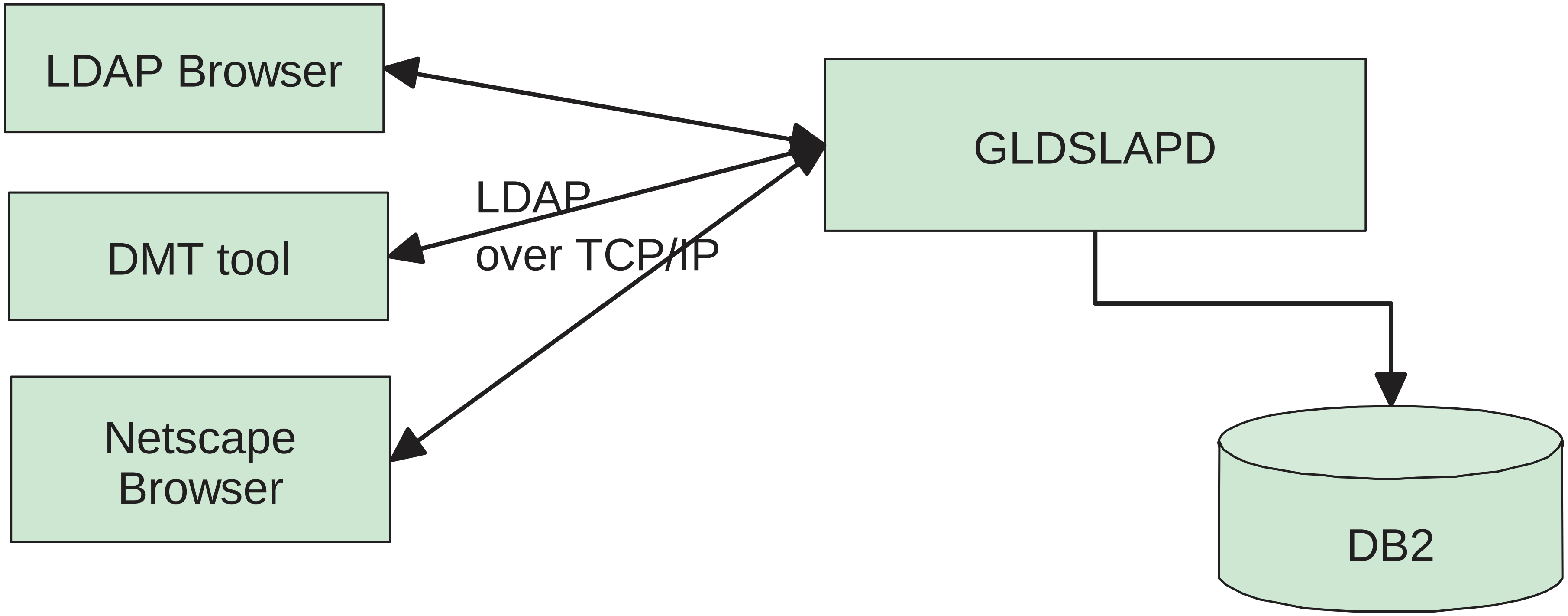
# Adding Information to the Directory



# Querying and Viewing this information

- A variety of tools can be used to view and even update this information:
  - ►LDAP Browser (`http://www-unix.mcs.anl.gov/~gawor/ldap/`)
  - ►Directory Management Tool ( `http://www-4.ibm.com/software/network/directory/downloads/` )
  - ►Netscape Browser (using LDAP URLs)
- I'll show each of these briefly, using my workstation to access the directory server

# Querying and Viewing this Information



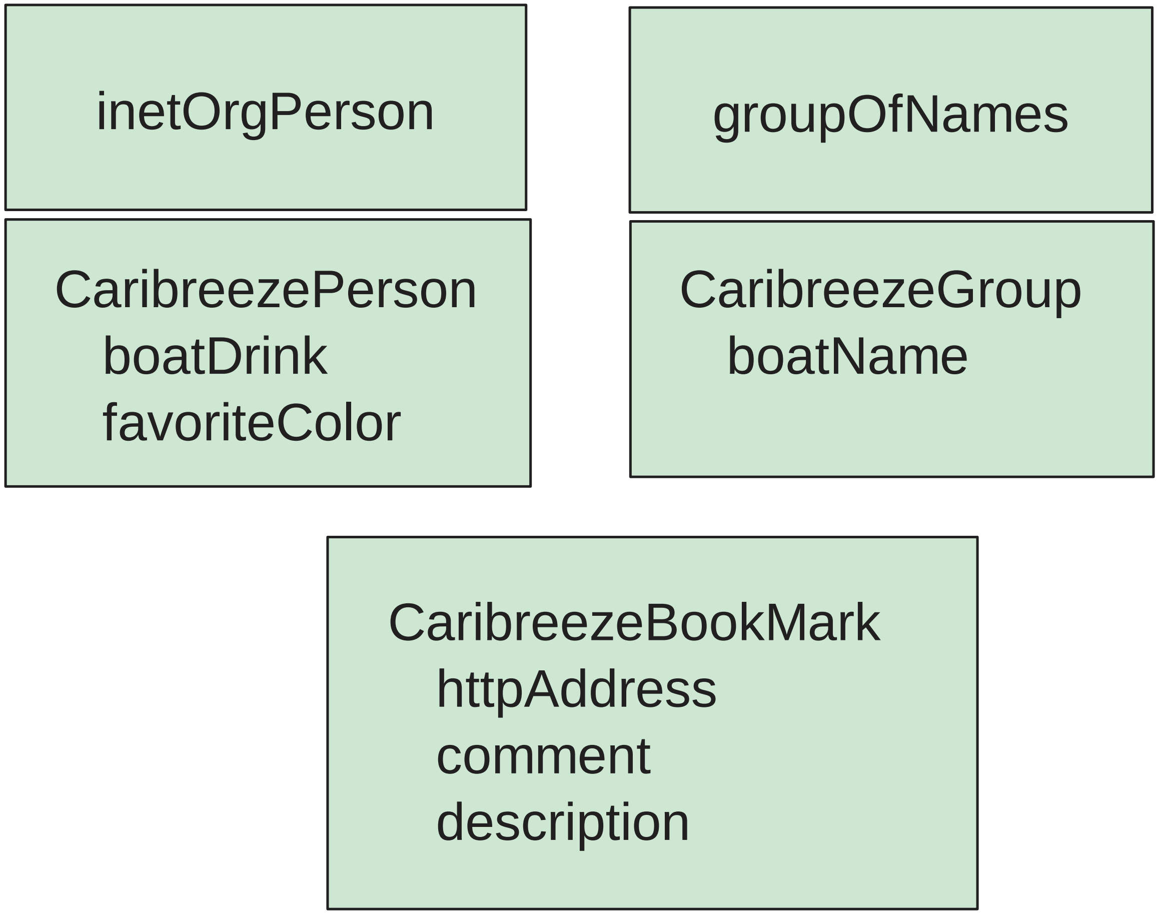
Copyright IBM Corp., 1999

# Add Some New Schema Definitions

- It is possible to add new schema formats to the directory server
- This is done by modifying the "schema entry" using the LDAP modify operation
- Defining new schema allows you to extend existing constructs or define new constructs to be stored in the directory

➤ We'll add a new user definition, a new group definition, and a bookmarks definition

# Adding new Schema Definitions



inetOrgPerson

groupOfNames

CaribreezePerson
boatDrink
favoriteColor

CaribreezeGroup
boatName

CaribreezeBookMark
httpAddress
comment
description

# Adding new Schema Definitions

- Three new Object classes:

```
( 1.3.18.0.2.1000.1.6.1 NAME 'CaribreezePerson'
   DESC 'Attached to inetOrgPerson to add more attributes.'
   SUP top
   AUXILIARY
   MAY ( boatDrink $ favoriteColor )
)
( 1.3.18.0.2.1000.1.6.2 NAME 'CaribreezeGroup'
   DESC 'Attached to groupOfNames to add more attributes.'
   SUP top
   AUXILIARY
   MAY ( boatName )
)
( 1.3.18.0.2.1000.1.6.3 NAME 'CaribreezeBookmark'
   DESC 'Entry that represents HTTP bookmarks for a user.'
   SUP top
   STRUCTURAL
   MUST ( description $ comment $ labeledURI )
)
```

# Adding new Schema Definitions

- Four new Attribute Types:

```
( 1.3.18.0.2.1000.1.4.1 NAME 'boatDrink'
   DESC 'A users favorite boat drink.'
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 EQUALITY caseIgnoreMatch
   USAGE userApplications
)
( 1.3.18.0.2.1000.1.4.2 NAME 'favoriteColor'
   DESC 'A users favorite color.'
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 EQUALITY caseIgnoreMatch
   USAGE userApplications
)
( 1.3.18.0.2.1000.1.4.3 NAME 'boatName'
   DESC 'A users boat name.'
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 EQUALITY caseIgnoreMatch
   USAGE userApplications
)
( 1.3.18.0.2.1000.1.4.4 NAME 'comment'
   DESC 'A short comment for the bookmark.'
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 EQUALITY caseIgnoreMatch
   USAGE userApplications
)
```

# Adding new Schema Elements and more information

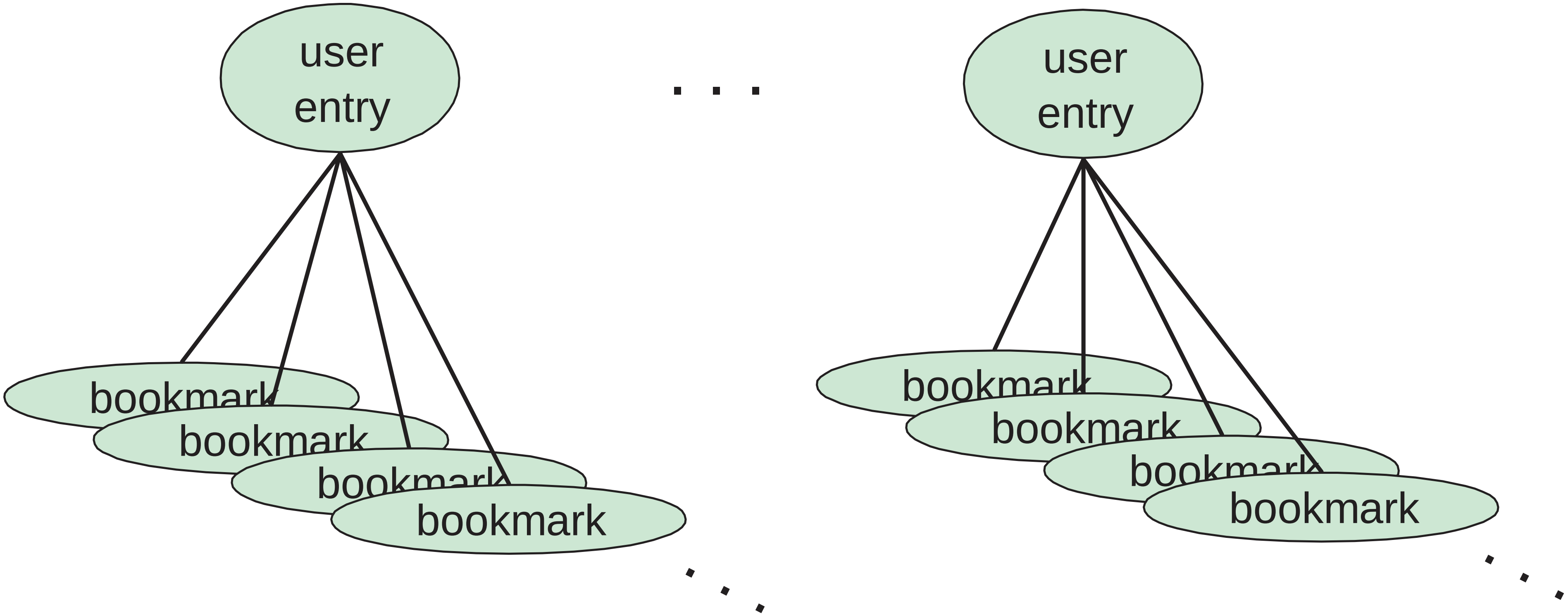
- Finally, use the ldapmodify command to add this new schema to the LDAP server:

  ```
  ldapmodify ... -f caribreezeschema.ldif
  ```

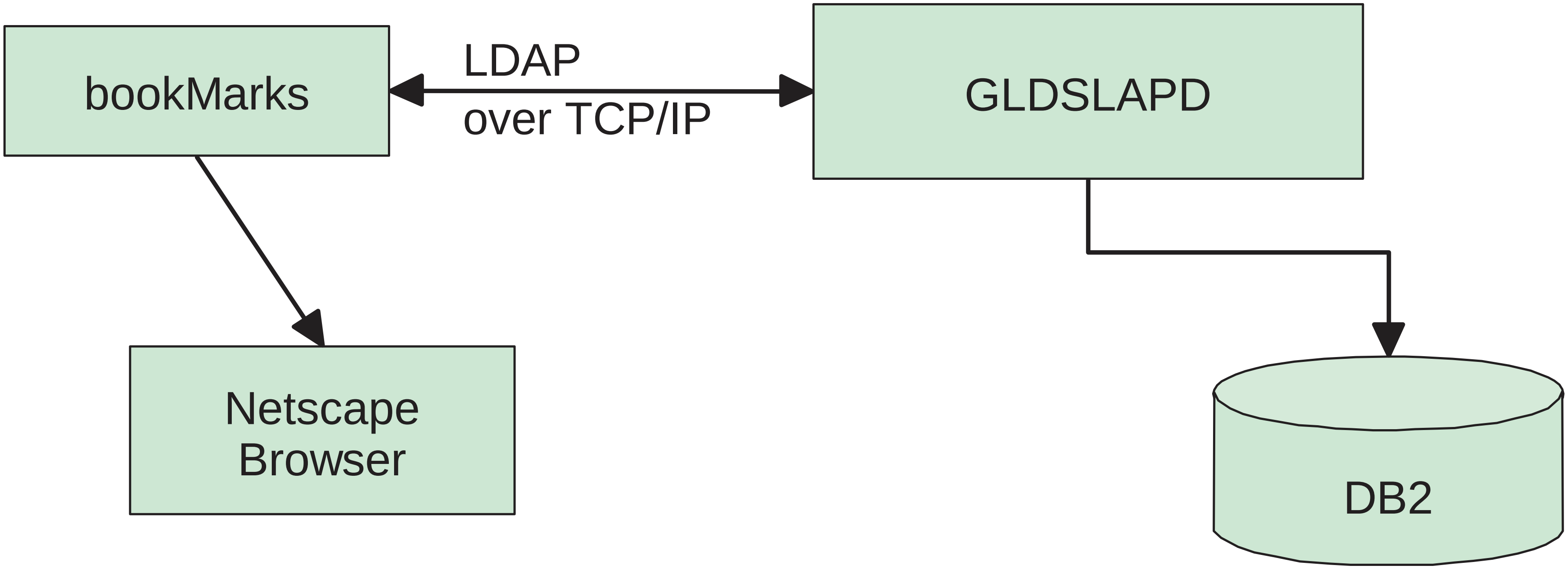
- Now add some more information to the directory using these new schema elements:

  ```
  ldapadd ... -f caribreezeusers.ldif
  ```

# Structure of the Information Added



# Applications to use these new object classes and attributes

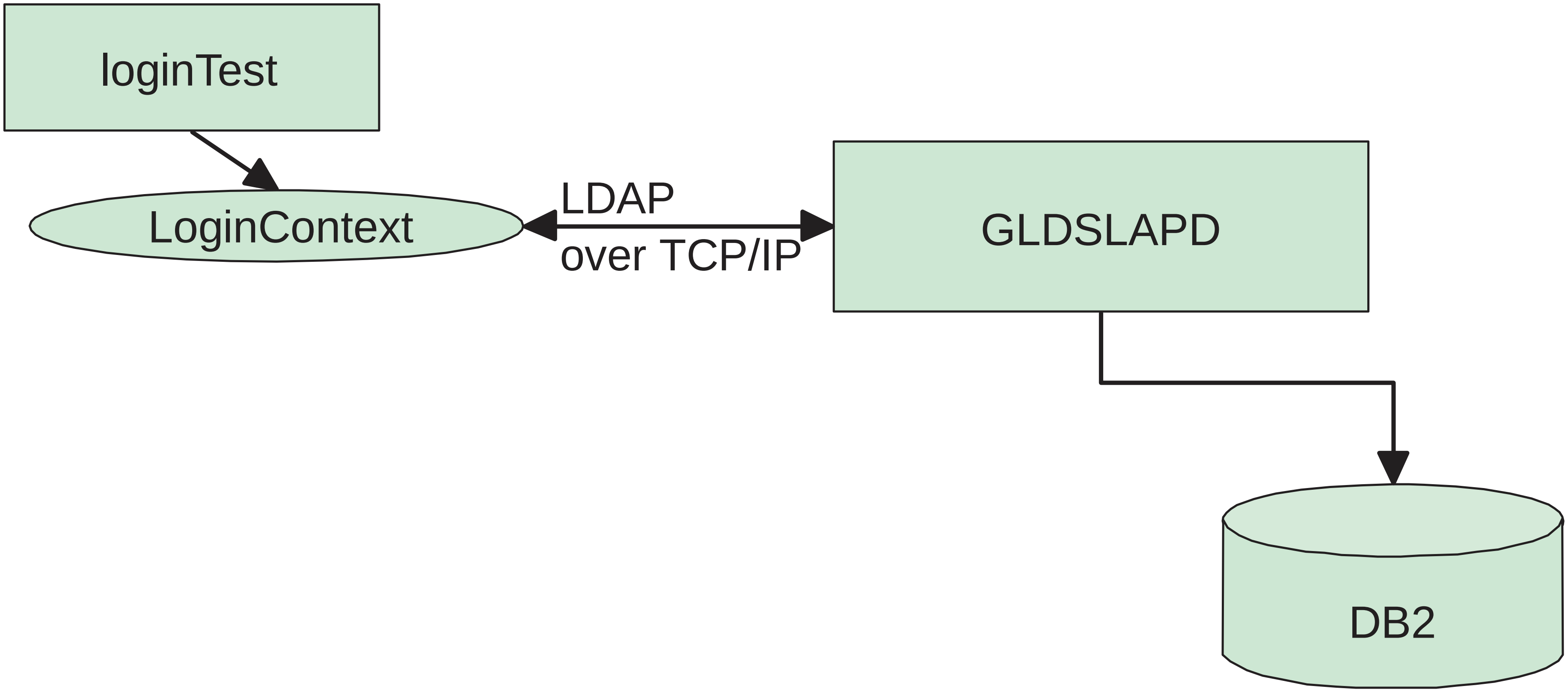


# Login "application"

- This application - really a Java class with a set of wrapper code - shows how LDAP servers can be used to help do authentication across multiple systems
- This algorithm is used in a number of products today
- Java class:

```
class LoginContext {
   LoginContext( String template,
                 String searchBase, String searchTemplate);
   login( String userid, String password );
};
```

# Applications to use these new object classes and attributes



# For More Information

- LDAP RFCs
  - http://sunsite.auc.dk/RFC/rfc/rfc2251.html-rfc2256.html
- OS/390 LDAP Documentation
  - SC24-5861-04 OS/390 Security Server LDAP Server Administration and Usage Guide
    - http://www.s390.ibm.com/ftp/books/os390/pdf/gldaga21.pdf
  - SC24-5878-01 OS/390 Security Server LDAP Client Application Development Guide and Reference
    - http://www.s390.ibm.com/ftp/books/os390/pdf/gld1aa20.pdf