Inventions on LDAP data storage- A TRIZ based analysis

Umakant Mishra

Available at: https://works.bepress.com/umakant_mishra/54/
Inventions on LDAP Data Storage
-A TRIZ based Analysis

By- Umakant Mishra, Bangalore, India
http://umakanmt.blogspot.in

Contents

Contents ........................................................................................................................................1
1. Introduction ...............................................................................................................................2
2. Data storage mechanism in LDAP ...........................................................................................2
3. Major Concerns in LDAP data storage ....................................................................................3
4. IFR for LDAP data storage .......................................................................................................4
5. Inventions on LDAP data storage ...........................................................................................5
6. Detailed analysis of Patent No 6085188 ................................................................................9
7. Analysis of Other Patents .........................................................................................................9
   7.1 Data security in LDAP service ...............................................................................................9
   7.2 Storing Object Oriented LDAP data in RDBMS .................................................................9
   7.3 Data replication for LDAP ....................................................................................................9
   7.4 Storing sparse hierarchical ACL data in RDBMS for LDAP .............................................10
   7.5 Online directory service with multiple database ...............................................................10
   7.6 Data migration for LDAP ...................................................................................................10
   7.7 Providing extended Access Control to LDAP Protocol ....................................................10
   7.8 Reverse indexing LDAP database for faster searching .....................................................10
   7.9 LDAP cache mechanism .....................................................................................................11
8. Trends of evolution in LDAP storage .......................................................................................11
9. Prediction of future inventions on LDAP data storage ............................................................11
Reference to patents: ..................................................................................................................13
Other references: .......................................................................................................................13
1. Introduction

Lightweight Directory Access Protocol (LDAP) is an IETF open standard to provide directory services in the network. LDAP was initially developed at the University of Michigan with an objective to include most of the features of X.500, while eliminating the burdens and difficulties of the same.

The current Version of LDAP is LDAP V.3 released in December 1997 (RFC 2251). Other specifications of LDAP can be found in different RFCs on the IETF (Internet Engineering Task Force) website.

With the growing use of Internet, LDAP is becoming more and more popular to provide directory services to a wide range of applications. This led to patenting several inventions relating to LDAP operation and application. This study on LDAP data storage is a part of the main study on LDAP based on 60 selected patents on LDAP from US Patent database.

2. Data storage mechanism in LDAP

As LDAP does not provide any specification on the data storage, different vendors can implement different mechanism of data storage as suitable to the vendor or environment. Although underlying data storage system between different LDAP servers can differ, it does not affect the functionality or interaction of LDAP clients. LDAP protocol does not expose this disparity in data storage to the users or an LDAP interface.

For example, an LDAP server may store data in a Flat file, or in RDBMS. The LDAP client applications such as LDAP enabled web browsers like Netscape communicator and Internet Explorer can use LDAP directory interface without having knowledge on the underlying data storage mechanism.

*Umakant Mishra, Inventions on LDAP, http://works.bepress.com/umakant_mishra/52*
LDAP stores the directory information in a database. It provides the capability for directory information to be queried or updated. It offers a rich set of searching capabilities. The client makes a TCP/IP connection and sends requests to an LDAP server. Simple as a web server, the LDAP server responds to the requests of LDAP clients. LDAP servers provide their service using a default port 389.

3. Major Concerns in LDAP data storage

- Storing LDAP data in a flat file is not efficient. There is a need for a better storage mechanism.

- RDBMS is a proven method and can be used for LDAP data. But hierarchical LDAP data is not suitable to be stored in RDBMS format.

- Storing hierarchical LDAP data in RDBMS needs a separate table to be created for each object class, which causes waste of system resources. Besides, each change in the object class requires a modification of database schema.

- Searching hierarchical LDAP data needs recursive queries. Recursive queries work very slow in RDBMS and other databases.

- LDAP being an open system does not provide enough security for its data.

- There is a need for a better method for storing object oriented directory data in Relational Database used as LDAP backing store.

- There is a need to store special types of data, such as semi-structured or unstructured data, in LDAP database.

- There is a need to improve the speed of data access, searching or updating operations.

- There is a need for ensuring security for LDAP data.

- As different LDAP implementations use different database formats, there exists a problem of incompatibility in data transfer, data integration, data replication, data migration and similar operations.

- Each object in LDAP may have different object classes. It is not efficient to use RDBMS to store LDAP objects having different object classes.

- There is a need for easy data maintenance, such as, maintaining data consistency, data duplication, replication, backup etc.
4. IFR for LDAP data storage

According to TRIZ, ideality is a function of its benefits and costs. **Ideality = \( \Sigma \text{Benefits} / (\Sigma \text{Costs} + \Sigma \text{Harm}) \).** So the Ideal Final Result in LDAP storage can be achieved by increasing all useful functions of the data storage and decreasing all harmful functions of the same. An analysis of the LDAP system may find the following as IFR for LDAP data storage.

<table>
<thead>
<tr>
<th>Should have (Positive features)</th>
<th>Should not have (Negative features)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• The LDAP backing store should be able to store all possible types of data for each individual entity in the directory.</td>
<td>• Should not require additional conversions of incompatible data types.</td>
</tr>
<tr>
<td>• The LDAP should accept semi-structured, unstructured and all types of data.</td>
<td></td>
</tr>
<tr>
<td>• The data storage should consume minimum disk space, minimum memory and processor time.</td>
<td>• Should not cause extra burden on the system or cause waste of system resources.</td>
</tr>
<tr>
<td>• The LDAP should provide extreme security of data stored in LDAP database.</td>
<td>• The security mechanism should not increase complicacy of the system.</td>
</tr>
<tr>
<td>• LDAP should be able to store Account Control Lists (ACL) to enable access rights to the users.</td>
<td>• The language to define Access Controls should not be difficult for users.</td>
</tr>
<tr>
<td>• LDAP database should be compatible with other database formats for import, export, transfer, conversion, integration and all other purposes.</td>
<td></td>
</tr>
<tr>
<td>• Should accept all kinds of searches, simple search, conditional search, wild card search and so on.</td>
<td>• Should not take more time for searching the database.</td>
</tr>
<tr>
<td>• LDAP should provide easiest mechanism to update and maintain its data (or automatically update and maintain its data).</td>
<td>• Should not require human involvement for data maintenance or administration.</td>
</tr>
<tr>
<td>• LDAP should be able to use RDBMS as its backing store.</td>
<td>• Should face no incompatibility in storing hierarchical LDAP data in RDBMS structure.</td>
</tr>
<tr>
<td>• Should be able to use standard queries like SQL for data searching.</td>
<td>• Should not require more time to execute recursive queries.</td>
</tr>
</tbody>
</table>
5. Inventions on LDAP data storage

The study finds the following five patents to be dealing with the issues on LDAP data storage.

- “Method of hierarchical LDAP searching with relational tables” (US Patent 6085188)
- “Method for securing sensitive data in a LDAP directory service utilizing a client and/or server control” (US Patent 6339827)
- “Method and system for representing and accessing object-oriented data in a relational database system” (US Patent 6587856)
- “Method and system for data replication” (US Patent 6615223)
- “Method for storing sparse hierarchical data in a relational database” (US patent 6438549).

Besides the following five patents are also related to the same topic.

- “On-line directory service with a plurality of databases and processors” (US Patent 5918227)
- “System, method and computer program product for migrating data from one database to another database”, (US Patent 6915287)
- “Reverse string indexing in a relational database for wildcard searching” (US Patent 6199062).
- “Lightweight directory access protocol (LDAP) directory server cache mechanism and method” (US Patent 6347312).

This article will try to analyze only one patent in detail by using different techniques of TRIZ. The other patents will be analyzed in brief.
6. Detailed analysis of Patent No 6085188

Patent Details:

Problem in the prior art:
All the query mechanisms use recursive queries to handle hierarchical structures of LDAP entries. Recursive queries do not work efficiently with large number of records in RDBMS. There is a need for a better data organization for faster searching.

Ideal Final Result (IFR):
According to TRIZ, an ideal system is a system, which does not materially exist, But all its functions are achieved. An Ideality of searching may be achieved in the following situations.

<table>
<thead>
<tr>
<th>The Ideal Final Result (IFR)</th>
</tr>
</thead>
<tbody>
<tr>
<td>The recursive queries work faster in searching LDAP data.</td>
</tr>
<tr>
<td>Searching is performed without executing recursive queries.</td>
</tr>
<tr>
<td>Searching operation is not necessary at all.</td>
</tr>
</tbody>
</table>

Desired results:
Although our objective is to achieve the IFR it is often difficult to achieve the IFR in full. But we should at-least try to achieve a set of desired results. The desired results in this case are the following.

- LDAP should be able to store and serve large volume of directory data.

- The query for searching the database should work fast even with large number of entries.

- The solution should not make the query language difficult for clients.

- The solution should not make the database complicated for the administrator to maintain.

- The solution should support data integrity, data security and all other necessary database features required for LDAP data.
Alternative Solutions:
We get various different solutions by applying different inventive principles. Some possible ideas of solutions are as below.

- Use a different query language or build an advanced query mechanism to make the query run faster. (**Principle-28: Mechanics Substitution**)

- Organize the data tables differently inside the RDBMS. (**Principle-17: Another Dimension**).

- Use a different file organization instead of using RDBMS (e.g., object database instead of relational database). (**Principle-28: Mechanics Substitution**).

- Keep less number of entries so that the query will run faster. (**Principle-16: Partial or excessive action**).

- Improve the features of RDBMS to support faster searching or recursive queries. (**Principle-38: Enrich**)

Assumptions:
Although each of the above tries to solve the problem in different angles, all of them do not fulfill the basic assumptions. We have to eliminate some solutions, which do not fulfill the assumptions. The assumptions in this case are the following.

- The system assumes that LDAP uses RDBMS as backing store.

- The LDAP would store substantially large number of entries in the database.

Contradiction:
When we want to satisfy the above assumptions we face the following contradiction.

All query languages (such as SQL) use recursive queries to handle hierarchical structures in LDAP entries.

Recursive queries do not work efficiently with RDBMS having large number of records.
Solving the contradiction:
Looking at the principles again, we find that the following principles can solve the above contradiction.

- **Principle-10: Prior Action**: Do a recursive search beforehand and store the search results. Use this search results to answer the query instead of searching afresh for each query.

- **Principle-24: Intermediary**: Use another table or set of tables which store the search results and work as index tables to search the entries.

Solution provided by US Patent 6085188:
US Patent 6085188 provides a solution similar to above. The invention proposes to add two more relational tables in the LDAP database. These two tables store the unique identifier of all the parents and of all the children, and stores entries for parent child relationship. The query searches these relationship tables first instead of doing a recursive search in the whole database.

![Diagram of Relational database, LDAP, and Client](image)

parent child relationship tables for LDAP entries

Evolutionary Potential:
Looking at the trends of software inventions, we find that the solution provides advanced features for “increasing speed”. However it is not very efficient to provide features like “easy to configure and implement”. While plotting different trends for the particular invention we get the following chart on Evolutionary Potential.
The colored portion at the center of the chart indicates the strength of the solution. The brown area at the outer circuit indicates the Evolutionary Potential (EP) for future inventions.

7. Analysis of Other Patents

7.1 Data security in LDAP service

US Patent 6339827 discloses a method of maintaining security in a directory service. As per the invention the lightweight directory access protocol (LDAP) is extended to include two controls, viz., a client-control to be implemented on the client machine and a server-controls to be implemented on a server machine (Principle-1: Segmentation). The client control will include necessary mechanism to provide security to the sensitive data provided from a client application to the directory service. The server side control similarly includes necessary controls for the security of data within the directory service (Principle-3: Local Quality).

However, it is not required to have both client and server controls to be implemented together. A client machine may implement the client control irrespective of whether a server involved in the directory operation is running the server control (Principle-15: Dynamize).

7.2 Storing Object Oriented LDAP data in RDBMS


According to the invention, there will be a table called “attribute store table” containing the information describing objects and object attributes of the system. Secondly, there will be “catalog tables” which store indexes to the “attribute store table” (Principle-10: Prior Action, and Principle-24: Intermediary). A separate catalog table is maintained for each attribute type that is indexed. (Principle-1: Segmentation). These catalog tables improves efficiency in searching.

7.3 Data replication for LDAP

US Patent 6615223 discloses a method of data replication for LDAP servers even in heterogeneous environment. According to the invention, when the first server will receive a change request it will change the data in that site. Then it will generate a change record in a schema independent format (Principle-6: Universality) and send that change record to the second replication site. The second replication site will understand the change record as it is in schema independent format and will implement the corresponding changes in that site.
7.4 Storing sparse hierarchical ACL data in RDBMS for LDAP

US Patent 6438549 discloses a method of securing Access Control List (ACL) for LDAP in a relational database. According to the invention the ACL data is stored in multiple tables (Principle-24: Intermediary, Principle-1: Segmentation), such as an owner table, a propagation table, a permissions table and a source table. The method reduces the activities involved in searching and updating ACL.

7.5 Online directory service with multiple database

US patent 5918227 discloses a method of accessing multiple directory databases though a “gater” or intermediate computer (Principle-24: Intermediary). The first gater receives the requests and passes them to other gaters. The gaters communicate between themselves to access various LDAP databases and reply to the client directly independent of the first gater (Principle-21: Skipping).

7.6 Data migration for LDAP

US Patent 6915287 discloses a method and software tool that is useful to migrate data from LDAP directory to another LDAP directory where the schemas for the two directories are not the same. The tool first compares the schema of a source directory to the schema of the destination directory. Then it updates the schema of the destination directory to be compatible with the source directory’s schema (Principle-33: Homogeneity). Before adding the entries, it verifies the modified schema to ensure that it can take all entries from the source database (Principle-10: Prior Action). In case the destination directory’s schema does not allow any particular entry, the destination directory’s schema is modified again until it ensures that the schema is fully compatible (Principle-20: Continuity of useful action). Finally it adds the entries to the destination directory.

7.7 Providing extended Access Control to LDAP Protocol

US Patent 6633872 provides a method of providing extendible access control to LDAP protocol. The extendible ACL can be implemented as a plug-in within the scope of LDAP server. According to the method, the user is associated with an access control group and the LDAP operation is reformatted based on the access rights of his access control group.

7.8 Reverse indexing LDAP database for faster searching

US Patent 6199062 discloses a method to speed up wildcard searching in an LDAP directory using RDBMS. The RDBMS normally includes a forward index, which is not efficient for wildcard (“*”) searching. The invention uses a reverse index of the character strings in the relational database (Principle-13: Other way round). In case of general searching it uses the forward index, whereas in case of wildcard searching it uses the reverse index to speed up the search (Principle-35: Parameter Change).
7.9 LDAP cache mechanism

US Patent 6347312 discloses a method of faster searching of LDAP data in RDBMS. The invention uses a cache to store the search results retrieved in response to the search queries (Principle-27: Cheap and disposable). The invention prefers to use two sets of caches (Principle-1: Segmentation). The first cache receives a set of identifiers indexed by a filter key of the search query. The second cache stores the search results corresponding to the set of identifiers in the first cache. When the same queries come again and again, the LDAP server serves the results from the cache (Principle-3: Local Quality) instead of sending requests to the backend database. When data is updated in the directory data the corresponding cache is invalidated (Principle-34: Discard and recover) to maintain integrity of the cached information.

8. Trends of evolution in LDAP storage

The following trends are prominent among the inventions on LDAP data storage.

- **Increasing speed in LDAP data access**: Inventions want to store data in such a way that will support faster searching.

- **Increasing controllability in LDAP service**: Inventions tend to increase data integrity, data maintainability, data backup and replication, data security and such other features to improve control of LDAP service.

- **Increasing easiness to “configure and implement” and increasing easiness to “maintain and administer”** are two obvious trends.

- **Increasing standardization**: Inventions are looking for Increasing interoperability to facilitate integration with other systems, databases or services, supporting conversion and migration.

9. Prediction of future inventions on LDAP data storage

Based on the analysis and based on the above trends we can predict a number of possible inventions on LDAP data storage in future. Some of them can be as follows.


- Compatible storage system for hierarchical LDAP data (Principle-33: Homogeneity).

- Converting hierarchical LDAP data to a format suitable for Relational Database (Principle-36: Conversion and Migration).

- Using a Universal format for storing unstructured data in LDAP database (Principle-6: Universality).

- Validating database operations submitted to LDAP server (Principle- Prior Counteraction).

- Distributing LDAP database into multiple databases according to geographical location (Principle- Segmentation).


- Compression and storage optimization of LDAP database (Principle- Mechanics Substitution).

- Automatic backup mechanism for LDAP (Principle- Copying, Principle- Self Service).


- Allowing users to configure access rights to their own data (Principle-3: Local Quality).

- Query optimization in RDBMS for LDAP data (Principle-21: Skipping).

- Providing scalability to LDAP database (Principle-15: Dynamize).

There can be many more possible areas of development. Let’s hope to see more and more inventions on LDAP data storage in future days.
Reference to patents:


Other references:

11. Umakant Mishra, Inventions on LDAP- a Study Based on US Patents, [Link]

12. Umakant Mishra, Inventions on LDAP data storage- A study based on US patents, (Sept 20, 2006), [Link]

13. Umakant Mishra, “Inventions on LDAP data storage- a TRIZ based Analysis”, [Link]


15. Umakant Mishra, Inventions on LDAP Data Management - A TRIZ Based Analysis (August 2006). Available at SSRN: [Link]


About the author

After working for more than 18 years in various fields of Information Technology Umakant is currently doing independent research on TRIZ and IT since 2004. He last worked as Director and Chief Technology Officer (2000-2004) in CREAX Information Technologies (Bangalore). Before that he worked as IS/IT manager (1996-2000) for ActionAid India (Bangalore).

Umakant is a Master in Philosophy (MA), Master in Business Administration (MBA), Bachelor in Law and Logic (LLB), Microsoft Certified Systems Engineer (MCSE+I), Certified Novel Engineer (CNE), Master Certified Novell Engineer (MCNE), Certified Intranet Manager (CIM), Certified Internet Professional (CIP), Certified Software Test Manager (CSTM) and holds many other global IT certifications.