Inventions on using LDAP for different purposes-
Part-3

Umakant Mishra
Inventions on using LDAP for different purposes, Part-3
-A TRIZ based Analysis of US Patents

By- Umakant Mishra, Bangalore, India

http://umakantm.blogspot.in

[This article is divided into 3 parts for convenience of size. The first part of the article includes 6 patents using LDAP for “e-commerce” and “Policy Management”. The second part includes 10 patents using LDAP for “Network Management” and “Telecommunications”. The third part of the article includes 11 patents using LDAP for “World Wide Web” and “Java and CORBA”.]

Contents

Contents .........................................................................................................................................................1
7. Inventions on using LDAP for World Wide Web..........................................................2
  7.1 Accessing web via LDAP........................................................................................................2
  7.2 Implementing smart HTML mail using LDAP .............................................................3
  7.3 LDAP-based distributed cache technology for XML .................................................4
  7.4 Using LDAP to produce composite XML document object model trees ..........5
  7.5 Retrieving style sheets from LDAP based on partial character matching ....6
  7.6 Using LDAP for publishing system for Intranet......................................................7
8. Inventions on using LDAP for Java & CORBA............................................................7
  8.1 Self configurable distributed system............................................................................7
  8.2 Exchanging data between a java system database and LDAP directory ....8
  8.3 Managing the persistence of EJB components via LDAP ..................................9
  8.4 Using LDAP for a CORBA Compliant Name Service: ......................................10
  8.5 Implementing persistence in trader services using LDAP ..........................10
Reference to patents:..............................................................................................................................................10
Other References:................................................................................................................................................11
7. Inventions on using LDAP for World Wide Web

- US Patent 6209036, “Management of and access to information and other material via the world wide web in an LDAP environment”,


- US Patent 6728757, “Smart HTML electronic mail”

- US Patent 6990629, “Publishing system for intranet”,

- US Patent 6463440, “Retrieval of style sheets from directories based upon partial characteristic matching”,

7.1 Accessing web via LDAP

Background problem

Web pages are identified and accessed through URLs. In order to facilitate easy return to a particular page sometimes the user stores the URLs as bookmarks. The bookmark helps the user to revisit the web page at a later period of time without having to retrace the original steps to discover the web page. Although this mechanism of bookmarks is popularly used, it has some drawbacks. For example, an URL visited today may not be valid when re-used at a later period of time for various reasons. The web page might have been shifted to some other location or may be very busy to get access. There is a need for a method to revisit the web pages even if the URL is changed.

Solution provided by patent 6209036

US Patent 6209036 discloses a mechanism of accessing the web through Directory services. According to the invention the web page URLs are stored in an LDAP directory as attribute-value pairs of the directory objects. The access logic to retrieve the URLs is also stored in the directory. When the client wants to access the web page, the invented system finds the desired web page intelligently using the access logic even if the original target web page has been deleted.
This method has the advantage of re-organizing the web sites without invalidating the directory-reference links stored in the bookmarks. This has the advantage of accessing the selected URLs in future. Managing web pages though LDAP also facilitates indexing and searching.

**TRIZ based analysis**

The same link or bookmark should work to revisit Web pages even if the site is moved or the URL is changed (Ideal Final Result).

If the URL is changed to access the web page, it may be possible to get the URL from the contents of the web page (Principle-13: Other way round).

The invention provides an intelligent logic to try various different indirect references for efficient searching of the desired web page (Principle-14: Curve).

### 7.2 Implementing smart HTML mail using LDAP

**Background problem**

Emails use two most common formats, viz., plain text and HTML formatting. There are some email clients, which support only plain text and does not support HTML formatting. If a client sends HTML formatted emails to a recipient that does not support HTML, the emails look garbled and unreadable. On the other hand if the client always sends plain text emails, then it is deprived of the rich formatting features of HTML. There is no way to know whether the recipient has a HTML compatible client or not.

It is desirable for the sender (or sender’s email client) to know the HTML formatting capability of the recipient’s mail client so that the sender can decide whether to send HTML formatted email or plain text email to the recipient.

**Solution provided by patent 6728757**

US Patent 6728757 discloses a method of smart HTML email system that utilizes an intelligent decision algorithm to decide if an addressee is HTML capable or not. Depending on the address of the recipient, the email client will send a query.
to the LDAP server to find out whether the addressee is HTML capable or not. If the addressee is known to be HTML-capable, the document is transmitted as is (in HTML format). On the other hand, if the addressee is not known to be HTML-capable the document is converted to plain text without containing any HTML formatting.

Thus the invention provides a smart HTML e-mail system that can send the same email document to different clients in different formats depending on the capability of the recipients’ email client.

**TRIZ based analysis**

If a client sends HTML formatted emails the receiving client may not be able to read the format. On the other hand, if the client sends plain-text emails he cannot use the formatting features of HTML mail (Contradiction).

The invention stores the HTML capability of the email clients in the LDAP server. Before sending the email, the email client sends a query to the LDAP server to know the HTML capability of the destination client and uses the format accordingly (Principle-10: Prior Action, Principle-15: Dynamize).

If the addressee is not known to be HTML capable, the inventive method automatically converts the document into plain text and sends (Principle-25: Self Service, Principle-36: Conversion).

### 7.3 LDAP-based distributed cache technology for XML

**Background problem**

XML is becoming more and more popular to organize data for the web. But it is not efficient to cache XML data using the conventional caching methods. There is a need for efficient caching of websites based on XML technology, which can improve the performance of data retrieval.

**Solution provided by US Patent 6901410**

US Patent 6901410 discloses a method dealing semi-structured data in the LDAP environment using XML. The invention has three steps, (i) transforming the semi-structured data into LDAP data by converting the semi-structured data as the attributes of individual nodes; (ii) converting a query, written in a semi-structured query language for operation on said semi-structured data, into an LDAP query; and (iii) accessing said LDAP data with said LDAP query.

According to the invention, the combination of internal data representation with the query written in semi-structured query language, can offer substantial advantage when deployed in a distributed caching environment. Since a query, written in a semi-structured language, can be naturally split into a sequence of sub-queries, it becomes easy to cache the results of sub-queries independently.
The process of evaluating sub-queries are simpler than complete queries and easy to manage in the cache.

The invention brings both, the semi-structured data model and the LDAP data model together into a system that provides the ideal characteristics for the efficient processing of XPath queries over XML documents.

**TRIZ based analysis**

The invention converts the semi-structured data to LDAP data and semi-structured query to LDAP query (Principle-28: Mechanics Substitution, Principle-36: Conversion).

The invention uses LDAP directory to store and retrieve the cache of XML data (Principle-24: Intermediary).

The invention splits a query written in a semi-structured language to a sequence of sub-queries as it becomes easy to cache the results of sub-queries independently than to cache the complete queries (Principle-1: Segmentation).

### 7.4 Using LDAP to produce composite XML document object model trees

Conventionally the Document Object Model (DOM) trees are created and updated only by using the DOM APIs upon a specific invocation from an executing application program. An XML parser is used to parse a given XML file to create the DOM tree. The DOM interface is used to change the structure and contents of the DOM tree.

There is a need for a technique that will allow updating DOM trees from the dynamically changing data in the XML document.

**Solution provided by patent 6635089**

US Patent 6635089 discloses a method of producing composite XML document model trees by using LDAP server. According to the invention, the data is stored in an LDAP directory and retrieved using LDAP protocol. It provides a technique to markup the data, which is dynamic in nature. It also marks up the document indicating that the corresponding DOM tree is to be updated dynamically to update the changing information based on a set of conditions.

**TRIZ based analysis**

The invention directly specifies the content of an XML document that this selected content is dynamic in nature, so that the specific values are dynamically obtained when the application uses the XML document (Principle-35: Change Parameter).
The method dynamically refreshes the content of the XML document according to the specified set of conditions, which may include periodic refreshes (Principle-15: Dynamize, Principle-19: Periodic Action).

7.5 Retrieving style sheets from LDAP based on partial character matching

Background problem

A "style sheet" contains information on font, margins and other specifications that is used during presentation of a document. Style sheets can be used for many types of presentation of a document, including printing the document, displaying it on a video display, processing the document by a speech synthesizer, etc. It is desirable to dynamically select one or more style sheets that tailor a document for presentation in a specific target environment. But the question is where to store and how to search the appropriate style sheets?

Solution provided by patent 6463440

Patent 6463440 discloses a method of storing style sheet characteristics as LDAP objects in an LDAP directory. The style sheets may be encoded in Extensible Style sheet Language (XSL), Document Style Semantics and Specification Language (DSSSL), or any other style sheet language having equivalent semantics. The retrieval of style sheets is based on pattern matching of characteristics.

TRIZ based analysis

The document should be automatically formatted with the appropriate style (Ideal Final Result). The appropriate style sheet can be selected to format the document (Desired Result).

The style sheets are stored in LDAP server and accessed dynamically whichever is desired (Principle-15: Dynamize).

The invention provides a pattern matching technique to select the most appropriate style sheets based on partial characteristic mapping for a given document (Principle-15: Dynamize, Principle-16: Partial or excessive action).
7.6 Using LDAP for publishing system for Intranet

US Patent 6990629 discloses a method of using LDAP to provide a publishing system that would facilitate the sharing of project related information in an efficient and easy-to-use manner. According to the invention the LDAP directory server will store the information relating to the rights of authors and readers of the system. The authors and readers can access the desired documents (which could be text, tables, video, graphics, sound etc) according to their access rights.

The invention breaks the HTML page into sections for efficiency in displaying and editing. The HTML page is converted to a plurality of XML files. The author can edit a particular section of the web page and save only that section of the web page on the server instead of transmitting back and saving the whole page on the server (Principle-1: Segmentation).

A reverse method is used to generate the HTML page from the XML files. First the plurality of XML files is combined to generate an XML output file. Then XSL style sheets are applied to generate the HTML output file (Principle-1: Segmentation).

8. Inventions on using LDAP for Java & CORBA

- US Patent 6366954, “Method and data format for exchanging data between a Java system database entry and an LDAP directory service”,
- US Patent 6904602, “Method and apparatus for implementing persistence in trader services associated with a computer system”,
- US Patent 6578050, “Method and apparatus for implementing persistence in name services associated with computer system”.

8.1 Self configurable distributed system

Background problem

With the growing popularity of Web a lot of distributed applications are being developed for the Web. Unfortunately Java applets cannot perform a complete function, which requires to access data distributed on different machines. This is because of the constraints imposed on the Java security model, so that java applets cannot connect to the machines other than the machined they have been downloaded from. This brings a serious limitation to the functionality of java applets as in a distributed system the applets may be distributed throughout the web on different machines.
The problem is resolved when using Common Object Request Broker Architecture (CORBA) mechanism, as a CORBA based applet can open a “virtual” connection to CORBA objects located on other machines. However, it solves only half of the problem, as it cannot open a direct connection to a machine to download contents other than CORBA objects. There is a need to solve this problem.

Solution provided by patent 6553405

Patent 6553405 discloses a self-configurable distributed computer software system and a method of enabling an applet running on a Web browser in the system to obtain needed software code in the form of Java classes from a machine that is not hosting the applet’s originating Web server. According to the invention the java classes are stored in a Lightweight Directory Access Protocol (LDAP) server. The web server is connected to the LDAP server with a mechanism to retrieve java classes as and when necessary.

![Diagram of system elements](image)

TRIZ based analysis

The invention stores all the java classes in an LDAP server. The web server retrieves the java class from the LDAP server as and when necessary (Principle-15: Dynamize).

The java classes are downloaded from a central repository (LDAP server) when required and removed when there is no further need (Principle- Discard and recover).

8.2 Exchanging data between a java system database and LDAP directory

Background problem

In a conventional network configuration, the process of installing new software or new applications is a static process. In such cases the network administrator statically defines each configuration on each client. The configuration information
for each particular client is hard-coded in the particular client. When a major upgrade is installed it requires the network to bring down to perform that operation.

It is desirable to have a system that supports distributed management of client and user configurations by storing such configuration information at a central repository. This would allow a network administrator to manage subsystem configuration from the server. He can also propagate all kinds of changes to applications at client computers from the server.

**Solution provided by US Patent 6366954**

The invention discloses a method of exchanging configuration data between a configuration server schema residing on a configuration server and a network directory service.

According to the invention, the configuration server is a Java System database server containing configuration data for multiple clients and network users. The exchange of data is significantly enhanced through the use of an extension to a network directory service enabling a rapid mapping between a directory service attribute and a configuration server property. The directory service entry includes multiple shadow attributes where each shadow attribute corresponds to a particular directory service attribute. The particular directory service attribute, in turn, has a corresponding property in the configuration server. The extension also includes a correspondence or path-matching file that contains matches between directory service addresses and configuration server location identifier or paths.

**TRIZ based analysis**

The invention addresses configuration of client systems remotely from the server. (Principle-17: Another Dimension).

The invention proposes an extension to the LDAP service (Principle-37: Expansion), which enhances the exchange of data by enabling a rapid mapping between a directory service attribute and a configuration server property (Principle-10: Prior Action).

**8.3 Managing the persistence of EJB components via LDAP**

US Patent 6751797 discloses a method of managing persistence of EJB components via LDAP. As per EJB technology the EJB components are integrated into an EJB server. EJB architecture supports deploying and executing distributed applications. EJB Components are portable, transaction-oriented, protected, multi-user and database-oriented. They are not only machine independent but also platform independent.
According to the invention, the EJB components integrate information into the LDAP entries of the Directory. The EJB server manages persistence of the EJB components via LDAP by mapping and adapting an entity EJB component to a given type of LDAP entry in a given directory (Principle-24: Intermediary).

8.4 Using LDAP for a CORBA Compliant Name Service:
US Patent 6578050 presents a method of using LDAP (or a directory service that is compliant to LDAP) for implementing CORBA compliant name services in distributed object systems (Principle-5: Merging). The name service can be compliant with CORBA CosNaming specification, and directory service is capable of storing persistence information in the directory database. The flexibility of LDAP can provide compliance with CORBA CosNaming Specification and it can provide an interface to interacting with the directory service for storage and retrieval of data.

8.5 Implementing persistence in trader services using LDAP
US Patent 6904602 invents a method of implementing a trader service using LDAP. According to the invention the trader service is compliant with the CORBA CosTrader Specification and where the trader service is arranged to store persistent information using a directory service such as LDAP (Principle-24: Intermediary). A data store interface is used for interacting with the directory service.

Reference to patents:


Other References:


