Licensing Support Network (LSN)

Physical Design Document

Final (Revision 1)
April 30, 2001

Contract Number GS-35F-4507G

Prepared for
Nuclear Regulatory Commission (NRC)
Matt Schmit, Project Officer
11545 Rockville Pike, M/S T-3F23
Rockville, MD 20852-2738

Prepared by
GRC International, an AT&T Company
1900 Gallows Road
Vienna, Virginia 22182

REVISIONS			
Letter	**Description**	**Date**	**Approval**
Draft	Initial Submittal of the Licensing Support Network (LSN) Physical Design Document	1 March 2001	L. Bradley
Final	Submittal of the Licensing Support Network (LSN) Physical Design Document	March 22, 2001	L. Bradley
Final (Revision 1)	Submittal of LSN Physical Design Document with Changes per NRC Physical Design Feedback (email) dated March 29, 2001 3:22 PM	April 30, 2001	L. Bradley

Table of Contents

Appendices

1.   INTRODUCTION

The Nuclear Regulatory Commission (NRC) has contracted with GRC International (GRCI) to develop the Licensing Support Network (LSN). The objective of the LSN is to reduce the time needed to prepare for the licensing hearing and give NRC the chance of meeting the Congressionally mandated three-year licensing process time frame. It is generally acknowledged that although the system does not guarantee the licensing time frame will be met, without the LSN it will not be possible to meet the mandated time frame. The LSN will achieve this time saving by:

*   Replacing classic "discovery" exchanges among parties by making all parties' relevant documents publicly accessible before docketing;
*   Establishing an electronic docket that will be publicly accessible; and
*   Transforming the motions practice a fully electronic process.

The Motions Practice and the Electronic Docket, while key parts of the LSN overall, are outside the scope of this particular effort. The focus of this project is on the design, development, operation, and maintenance of a web portal capable of working with the other components that make-up the complete LSN system. Despite the LSN being composed of other components when the term LSN is referred to in this document it will refer to only that portion that is the web portal. The LSN web portal effort is focused on two of the four LSN functional requirements:

*   Establishing an effective Internet-based method of accessing (search and retrieval) the record collections of the parties and potential parties to the high-level waste repository licensing proceeding.
*   Providing an audit/compliance subsystem, including the automated tools and policies and procedures needed to monitor Participant compliance with the availability and document integrity submission requirements found in 10 C.F.R. Part 2, Subpart J.

LSN Participants include the NRC, Department of Energy (DOE), the State of Nevada, counties in Nevada, Nuclear Energy Institute (NEI), other organizations and citizen groups, as well as the general public.

Each Participant, including NRC, must meet core requirements for making available on the web their documentary materials and for providing the computerization necessary to comply with the Subpart J provisions for document production and service. These include requirements for providing web accessible bibliographic headers (structured data) and searchable text (unstructured data) in its documentary material and a description of where an authenticated image of the document may be obtained. Where text is not available (e.g., topographical maps, engineering drawings, etc.), the image will be made available online in lieu of the text file. Structured data bibliographic headers are required for items not suitable for image or text. Similarly, structured data bibliographic headers are required for privileged, confidential, safeguards, and other types of limited access documents.

1.1   Background

Section 114(d)(2) of the Nuclear Waste Policy Act of 1982 (NWPA) requires the Commission to issue a final decision approving or disapproving issuance of the construction authorization for a mined geologic repository to store high-level radioactive waste at Yucca Mountain, NV. This final decision must be reached within three years of the U.S. Department of Energy (DOE) license application. The Licensing Support Network (LSN) is a critical tool to ensure that document accesses, and associated hearing agenda, can all be handled in an expeditious manner. As outlined in 10 C.F.R. Part 2, Subject J, it will establish a system to provide shared document discovery and facilitate electronic motion practice for the hearings on DOE's license application for the repository. Having all parties to the LSN decision share their documents via the LSN web portal greatly improves the discovery process, reduces costs, and levels the playing field among all Participants.

1.2   Objectives

The objective of implementing the Licensing Support Network (LSN) is to reduce the time needed for the licensing hearing and give the Agency some chance of meeting the Congressionally mandated three-year licensing process time frame.

The LSN web portal is the primary search tool for retrieval and dissemination of the electronic documents concerning the Yucca Mountain Project and will be used by both Participants in the Licensing Activity as well as the general public. The system will provide full support of the National Archives and Records Administration guidance for record and record-keeping systems. The portal will also be in compliance with Section 508 of the Rehabilitation Act.

The core Internet functions that will be provided by the LSN web portal are:

- Provides shared access to documentary material;
- Provides timely, effective access, search, and retrieval for large collections of diverse documents;
- Identifies where associated images are easily located, if not available on the system;
- Provides a unique document ID across the enterprise;
- Provides priority access (to approximately 500 users) during key phases of the licensing process;
- Allows the LSN Administrator (LSNA) to document integrity of the Participant collections; and
- Ensures uninterrupted performance over at least a three-year licensing time frame.

The LSN as a web portal is not a central repository, but the central source for information discovery for the LSN. Documents are not copied over or stored on the LSN from Participant sites. The LSN will instead store pointers or links to where the actual documents are on an individual Participant's web server. When users find the document they seek and request it, the document is served up from the Participant's site to the requester.

1.3   Scope

The LSN is being developed for the U.S. Nuclear Regulatory Commission (NRC) and is based on the Project Charter dated January 5, 2001. The LSN is being developed to comply with the provisions of 10 C.F.R. Part 2, Subpart J. The various Participants in the LSN must provide access to their document repositories in order for the LSN to provide the data and information necessary for the licensing process to succeed.

The Agency-wide Document Access Management System (ADAMS) contains publicly accessible docket files that will meet the docket requirements. Similarly, NRC's existing Electronic Information Exchange (EIE) infrastructure will meet the 10 C.F.R. Part 2, Subpart J motions practice requirements. Therefore, the LSN project is focused on the search and retrieval component and the audit capability.

When constructing the LSN, the Development Team will rely heavily on commercial off-the-shelf (COTS) software tools and applications for development as compared to creating custom software. Visually, the scope of this project is shown in Figure 1.3-1 below.
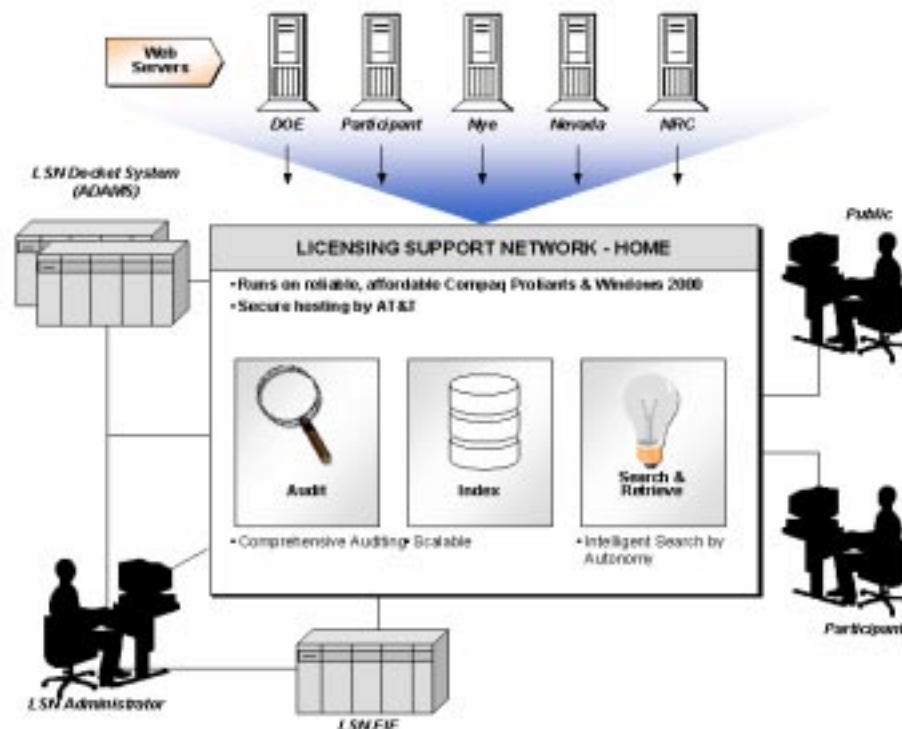

Figure 1.3-1. LSN High-Level Architecture.

1.4   Assumptions

The LSN Physical Design has been developed using the following Assumptions:

1. Several COTS software product selections have already been made.
2. The volume of data the LSN will contain is between the High Estimate of Total

Cumulative Relevant Pages for the year 2010 of 15,776K and the Low Estimate of Total Cumulative Relevant Pages for the year 2010 of 1,331K. These estimates (as of February 5, 2000) were provided by the Government in Appendix B of the NRC LSN Statement of Work.

3. The archiving of the data (documents) is the responsibility of the Participant site and the Participant site is responsible for making the data available to the LSN as described in the "Rule".

1.5   Applicable Documents

a. Statement of Work, U.S. Nuclear Regulatory Commission, and Contract Number GS-35F-4507G dated December 12, 2000.
b. GRC International (GRCI) Nuclear Regulatory Commission (NRC) Licensing Support Network (LSN), Volume 1 – Technical Documentation, dated August 25, 2000.
c. 10 C.F.R. Part 2, Subpart J.
d. Technical Documents/Plans/Standards:
    1. Workforce Investment Act of 1998, Rehabilitation Act Amendments of 1998, Section 508, Implementation.
    2. Computer Security Act of 1987.
    3. System Development and Life-Cycle Management Methodology (SDLCM), Procedures Standards and Forms, Version 1.2, dated December 1999.
    4. System Development and Life-Cycle Management Methodology (SDLCM), Handbook, Version 2.2, dated December 1999.
e. LSN Logical Design Document dated March 20, 2001.
f. LSN Project Action Plan dated March 15, 2001.
g. LSN Project Definition and Analysis Document dated March 22, 2001.

1.6   Overview

This document discusses the design activities that preceded the creation of the Physical Design. The sections herein include:

a. Section 1 defines the project objectives, scope, assumptions, reference documents used and the organization of this document.
b. Section 2 identifies the approach used to develop the LSN physical design.
c. Section 3 presents the Physical Model.
d. Appendix A contains the Requirements-to-Design Traceability Matrix.
e. Appendix B contains the Logical to Physical Mapping.
f. Appendix C contains Class and Object Model Tailoring information.
g. Appendix D contains the Figures referred to in Section 3   (Note: These figures are contained in an electronic PowerPoint file due to WP limitations).
h. Appendix E contains the Data Definition Language (DDL).

2.   LSN DESIGN APPROACH

This Design Document is a model of key LSN implementation details. It defines the custom objects, web pages, scripts, system objects and COTS objects that will compose the system. Because the implementation will be primarily object-based, this model is expressed in UML, the Unified Modeling Language, an industry standard, object-based modeling language.

Figures 3-1 through 3.6.2-1 are located in Appendix D of this document. Presenting the diagrams in a separate appendix enables the user to place the diagrams beside the text as they are reading.

2.1   Audience

Although the primary audiences of this document are the developers and maintainers of the system, general reviewers may gain an understanding of the LSN in several ways. To understand how different parts of the LSN will work, locate a package of interest, review the corresponding object interaction diagrams (or sequence diagrams) and the accompanying text. Second, to see what the different building blocks are, review the corresponding class models (or class diagrams) and text.

2.2   An Overview of the LSN Models

The LSN design is expressed as a collection of different UML models. The top-level model is a:

- Package diagram: Figure 3-1 (in Appendix D) shows the six "packages" that represent the major functional capabilities of the system.

    1. (1) Participant Document Fetch
    2. (2) Participant Site Audit
    3. (3) LSN Search
    4. (4) LSN Administration
    5. (5) LSN General Web Content
    6. (5.1) Participant Login

Each package is then defined in greater detail with at least two models:

- One *Class Model*
- One or more Object Interaction Models expressed as sequence diagrams

Each contains both diagrams and descriptive text. Although not part of a pure object-oriented approach, the relational data stored in SQL Server is also documented with an additional type of model:

- Entity Relationship Diagrams (ERDs)

The relational data in the Class Model will have a corresponding ERD that describes its physical database structure.

This modeling approach is "use-case driven, architecture-centric", meaning that the focus is on discovering the class and object structures (architecture) that will implement the major functional behaviors (use cases) of the system. The following sections provide more detail on the content of these core models, the Class and Object diagrams.

2.3   Class Models

Each package shown in Figure 3-1: Top Level LSN Packages Diagram, will have a Class Model that decomposes it.

Each Class Model includes a diagram of all the classes and class relationships in the package. A narrative describes the roles and responsibilities of the classes. The class models represent the static, structural aspects of LSN. Thus, this part of the design doesn't reveal how things work; it only defines the building blocks. A class represents a discrete, cohesive chunk of software and data that will be either coded during implementation or assembled from existing packages.

Some of these classes are shown as components or Component Object Model (COM) objects. These are special types of language-independent objects that implement Microsoft's COM interfaces. Other entities, pictured as classes, are Commercial-Off-the_Shelf (COTS) products and web pages. These are, classically speaking, not true objects, but are shown here for completeness. See Section 2.5 below for a discussion of tailoring.

See Figure 3.4.2-1, Participant Administration, for a representative sample of a class model. Each class is pictured as a box. Where appropriate, key methods and data are shown inside the box. The types of relationships between classes are shown on the connecting lines between the boxes. For example, *Find Participant Form* (top right in Figure 3.4.2-1) is a class representing an HTML form. It has fields (or data) in it for Last, First Name and User ID. It has two relationships with the Participant List class (1) an Active Server Page (ASP) where the form is physically contained within page, and (2) it has an HTML "submit" association with the page. More information on UML class modeling is presented in Appendix C.

2.4   The Object Interaction Models

In turn, each Class Model will have one or more Object Interaction Models that describe how things work. These include a sequence diagram representing how the objects (instances of classes from in the class model) interact at run-time to provide the major functionality (or "behaviors") of the system. A text narrative describes each interaction.

See Figure 3.4.3-1, Administer Participants Sequence Diagram, for a representative sample of a sequence diagram. The objects are shown horizontally across the top of the

diagram: *Participant Org List, Participant List, Find Participant Form*, for example. Each object is a member of one of the classes shown in the corresponding Class Model. A sequence of interactions between objects is shown vertically down the page: *Link, submit, get values*, for example. The accompanying text describes this step-by-step, down to the level of pseudocode where applicable. An arrow in the diagram indicates that one object is calling a method (or function) in another object. The arrow points to the object being called. Please note that this indicates the flow of commands (or "messages") through the system, *not* the flow of data. Data tends to remain inside the objects themselves, or is sometimes returned back to the caller (in the *opposite* direction of the arrow) as a result of issuing a command. This is the reverse direction of arrows in other modeling techniques, such as data flow diagrams.

2.5 Tailoring the UML Models for LSN

LSN is not a "classic" object-oriented system for the following reasons: it is web-based; and, large parts of it are assembled from existing COTS and system products. However, UML is a *customizable* industry standard for expressing object-based designs. Thus, we have tailored and extended UML to clearly express the design of LSN. This tailoring includes the following:

- Showing COTS products as if they were classes. For example, many of the Autonomy products are pictured as "class utilities", with a shadow on the lower right part of the class box. They are also labeled with a class stereotype: «COTS Product» for example.
- Showing Autonomy and SQL Server data stores as class utilities. Generally, these data stores are never directly manipulated by LSN custom code, and are, therefore, immaterial to the design. The LSN code will use vendor-supplied components to indirectly access this data. Regardless, these data stores are shown for completeness so the reviewer knows where the data is physically stored on the disk.
- Showing web pages and other HTML elements such as forms as classes and objects in the UML models. Sample stereotypes are «Client Page», «HTML Form».
- Showing the relationships between web pages, such as hypertext links and form submissions as class relationships in the UML models. Sample stereotypes used are «Link», «Submit».

This tailoring is described in detail in Appendix C.

2.6 An "Iterative and Incremental" Approach

System building is an iterative and incremental process. The design presented here represents the current implementation approach; however, the LSN development team will continue to refine and optimize this design, feeding lessons learned from coding and testing each iteration of the system back into the design. These iterations are defined in the Project Action Plan. Thus, this document represents a snapshot of the design at one given point in time. Future document versions will document the final form of the completed product.

3.   THE LSN OBJECT-BASED DESIGN

LSN TOP-LEVEL PACKAGES (SUBSYSTEMS)

This section describes the details of the LSN object-based design. The top-level of the design consists of six packages or subsystems, pictured in Figure 3-1:

1. (1) Participant Document Fetch
2. (2) Participant Site Audit
3. (3) LSN Search
4. (4) LSN Administration
5. (5) LSN General Web Content
6. (5.1) Participant Login

These packages represent major capabilities of the system, and may be thought of as LSN Subsystems. After some high level introductory material, sections 3.1 through 3.6 will present class and object models for each package.

MAJOR LSN REPOSITORIES

There are six primary stores of information presented in detail in this design but summarized first here. Note that significant amounts of LSN data are stored *outside* of a relational database. Autonomy, for example, manages it own store of information, optimized for free text searching. Also note that some of these repositories are "work areas" where data is staged for processing, not for end-user searching. The major LSN data repositories are as follows:

- *Autonomy Staged Content Data* – an Autonomy work area where document content is initially staged as it is retrieved from Participant sites. This area is *not* used for end-user searches. (See the Autonomy Production Content Data below.)
- *Autonomy Production Content Data* – the primary searchable repository. It contains both searchable content and header information merged into one Autonomy record. It also contains older records, deleted by Participants.
- *SQL Header Database* – a database work area where headers are stored. It contains current headers plus older changed and deleted headers.
- *SQL Audit Data* – an audit database of add/change/delete transactions for both documents and headers.
- *SQL Participant Site and Representative Database* – contains information on Participants, especially IDs and passwords.
- Site Log files and availability audit statistics – are maintained within two COTS products, *WebTrends* and *What's Up Gold*.

INFORMATION CAPTURED DURING THE FETCH AND AUDIT

One of the major goals of LSN is to capture the *content* of added, changed and deleted headers and documents, and to create an audit transaction that records each add, change, delete *event*. The following table shows where this is captured in the design. The numbers

in parentheses refer to a documented point in the Sequence Diagram narrative where each item is captured. Any duplicate captures will be used to validate the algorithms.

Information Captured:	Added Records	Changed Records	Deleted Records
Header record saved	(1)	(3)	(17)
Header audit transaction captured	(2)	(4)	(18)
Document content saved	(5 hdr only rec) (9)	(11) (14)	(7) (16)
Document audit transaction captured	(6 hdr only rec) (10)	(12) (13)	(8) (15)

3.1   Participant Document Fetch (1)

The Participant Document Fetch subsystem (package) creates a searchable LSN repository of unstructured document content and structured headers or metadata. Because this is the most customized and most complex part of the LSN system, we cover it first at a high level here and then in more detail below. The "Fetching" process happens in two phases, described in sections 3.1.1 and 3.1.2. The following paragraphs provide a summary. Figure 3.1-1, Participant Document Fetch, A Conceptual View, illustrates this. This is not a formal model, merely a visual aid. Unlike the class and sequence diagrams that follow, it pictures the flow of data, not commands.

The first phase of the "Fetching" process uses a working area or staging repository to prepare the data. The Autonomy HTTP Fetch retrieves the textual content of all documents and stores them in the Autonomy Staged Content Data. The Header Importer, a custom Visual Basic (VB) component, fetches all headers, parses them, and stores them in the SQL Header Database staging area.

In the second phase of the "Fetching" process, the *Header-Content Merger*, another custom VB component, pulls each staged header from the *SQL Header Database* and matches it by URL to the corresponding staged document content from Autonomy.   The *Header-Content Merger* component then combines both into one Autonomy record and writes it to the *Autonomy Production Content Data* repository. Header data is added as custom fields to the Autonomy record, with any multiply occurring fields concatenated together.

Users will search headers and content in the *Autonomy Production Content Data* repository. (This will be described in Section 3.3, LSN Search.) Combining headers and content into one record allows users to write advanced Boolean searches that combine both content keywords and header terms into one query. Note that the *SQL Header Data* in the staging area is not searched, but remains available for future uses if needed.

As the fetching process runs, it also provides information needed to detect added, changed and deleted documents. This will be described in Section 3.2, Participant Site Audit.

3.1.1 Participant Document Fetch (Class Model)

This section is a brief definition of each class in the Participant Document Fetch. See Figure 3.1.1-1 for a graphic, UML representation of this Class Model.

AUTONOMY PROVIDED CLASSES

This section provides a brief description of the classes used in the Participant Document Fetch. The following sections, 3.1.2 and 3.1.3, will describe how these classes interact. Discussed first are the Autonomy COTS products and components:

- The *Autonomy HTTP Fetch Service* (shown in the top center of the diagram) is a component supplied by Autonomy. It "crawls" each Participant site, retrieves documents, strips out the text content and stores it. This is sometimes called a "spider".

The *Autonomy HTTP Fetch Service* writes to two data storage areas:

1. The *Autonomy Staged Content Data* (top right) is a working area repository for content retrieved from Participant sites. It is not searchable by end-users. The actual format of this data to a user is a black-box, it is not directly visible to the outside.
2. The *Autonomy Fetch Results* or log file (top left). As Autonomy retrieves the text of each document, it places an entry in its log. These entries will be used later in the Participant Site Audit package.

The major output of this step is the searchable data:

- The *Autonomy Production Content Data* (bottom right). This contains both headers and content in searchable form. The "Merge Headers and Content" interaction diagram shown in Figure 3.1.3-1 describes its creation.

Both the Autonomy Staged Data and Production Content Data stores are black boxes. Their internal structure is not and need not be visible to LSN software because the following Autonomy-supplied components provide a programmatic, COM interface that manipulates the data:

- The *Autonomy Agent* component (upper right) reads from the *Autonomy Staged Content Data.*
- The *Autonomy DRE Component* (lower right) reads from and writes to the *Autonomy Production Content Data* store.

SQL SERVER PROVIDED CLASSES

See section 3.7 for Physical Data Model Information.

A SQL Server database and a set of ADO (Active Data Object) components also play a key role in the *Participant Document Fetch*. There are three SQL Stores:

- The *SQL Header Data* (bottom center right) is a set of relational tables containing all header data (the "metadata") retrieved from Participant sites. The single-value fields (such as Title, Participant Accession Number, and LSN Accession Number) are in one central table (Header); the multi-valued fields are in subsidiary tables (one each for Authors and Organizations, Addressees, and Related Records, for example). This table is populated by the *Header Importer* component (middle center).
- *SQL Header Audit Data* (bottom left) contains a record of the header audit of Participant sites. Events such as added, changed and deleted headers are detected and recorded here by the *Header Importer* component (middle center).
- *SQL Document Audit Data* (bottom left) contains a similar log of added, changed, and deleted documents detected by the *Header-Content Merger* (middle right) and the *Fetch Auditor* (top right in the following "Participant Site Audit" class diagram).

There are three corresponding sets of ADO objects that programmatically manage the above SQL data. These are system-provided components. Each set contains an ADO Connection, Command, and Recordset:

- *ADO Header Data* (lower center)
- ADO Header Audit Data (lower center)
- *ADO Document Audit Data* (lower left)

CUSTOM VISUAL BASIC OBJECTS

The core of this process is the three custom-built Visual Basic components:

- The *Header Importer* (middle center) retrieves headers from Participant sites. Its key responsibilities are as follows: (1) It uses FTP (File Transfer Protocol) to read each Participant's directory of headers and transfers them over the Internet to LSN. (2) It parses each header using Microsoft's XML parser component. (If XML is not ultimately used, some other suitable parsing component will be used.). (3) It generates a unique LSN Accession Number if this is a new header; otherwise (4) it compares this header to the previous copy. And, finally (5) it adds the header to the *SQL Header Database*. This process is shown in detail in Figure 3.1.2-1, Assemble Documents and Headers.
- The *Header-Content Merger* (middle right) matches up three separate pieces of information: (1) SQL Header Records, (2) Staged Autonomy Content Records and (3) Production Autonomy Content Records. It builds a Production record containing both searchable content and searchable header fields. It also audits added, changed and deleted documents.
- The *Fetch Auditor* (middle left) is discussed in detail in section 3.2, Participant Site Audit.

3.1.2   Fetch Participant Documents and Headers (Object Interaction)

See Figure 3.1.2-1 for a graphic, UML representation of this object interaction model and a step-by-step description.

The above section has described the static classes that are the building blocks of the Participant Document Fetch part of the system, but not how they work together to provide some useful function. The following sections will describe these interactions, at a high level below, and step-by-step in the diagrams. There are two parts to the fetch:

FETCH DOCUMENT CONTENT (PART A)

Fetching documents from Participant sites is essentially a COTS operation performed by Autonomy. Initially, a Knowledge Management Analyst configures the Autonomy spiders for each Participant site. Then, the Analyst monitors the fetch results for unusual conditions. While fetching is done during production, it is also part of the Maintenance activities necessary to ensure that what is expected is in fact returned.

Once configured, Autonomy fetches documents, stores the ASCII text content of each (stripped of formatting) in the *Autonomy Staged Content Data* repository, and logs the fetch results. If Autonomy detects a changed record, it updates the content in the Staged area. If it detects a new record, the document content is added to the Staging area. If unchanged, nothing happens.

IMPORT HEADERS (PART B)

These objects build an SQL database of both Header Data and Header Audit Data. It loops over each header for each Participant site, retrieving and parsing each one. It attempts to find a corresponding header in the SQL database, matching by the Participant Accession Number (or the URL of the header file). If there is a match, it compares the current and stored header to detect any changes. If there is no match, it generates a new LSN Accession Number, stores the new header, and logs the event for audit purposes.

3.1.3   Merge Headers and Content (Object Interaction)

See Figure 3.1.3-1 for a graphic, UML representation of this object interaction model and a step-by-step description. Note that the numbers in parentheses in the diagrams refer to information captured.

At this point the staging area contains separate headers and document content. This step will merge the two together into the *Autonomy Production Content Database*, collecting audit statistics along the way. This step also looks in both the Autonomy Production and Staging repositories for each header record, and deduces auditable conditions from this, as will be shown below.

For each Participant site and header, the steps are as follows. Format the header information for storing as custom fields in Autonomy. Authors, for example, will be concatenated together into one Autonomy field. Now, using the URL of the document (recorded in the Header), attempt to find a matching document in the *Autonomy Staging* and the *Production Content Data*. There are four possible outcomes:

Staged Doc Found?	Production Doc Found?	**Action:**
No	No	This is a new, header-only record with no matching document: Add a "Header only" record to the Production Data. Record a "new document" audit transaction. Check the header fields to make sure this is really a document that is not available from the Participant. It might be a lost record, or have a bad URL in the header.
No	Yes	This is a deleted record: Flag the Production Record as "deleted" (in a custom field). Note that we still keep the old record content, even though it is flagged as deleted. Record a "deleted document" audit transaction.
Yes	No	This is a new record: Add the record content and header to the Production Data. Record a "new document" audit transaction.
Yes	Yes	Check the date of the staged record. If it is today's date the record was just replaced. Thus, take the following actions: Flag the Production Record as "changed"; keep the old copy. Add the new record (header and content) to Production. Record a "changed document" audit transaction. If its not today's date, then do nothing; this is a normal condition.

Note that the Record ID from the Staged Document is also stored as a custom field in the Production Document. This information is needed later.

3.2   Participant Site Audit (2)

In terms of class and object models, the title of this section is somewhat inaccurate. First, several COTS products provide much of the auditing capability, so there is not much to model here. Second, most of the document and header auditing has already been fulfilled in the Participant Document Fetch package as a normal part of the fetching and merging process. Thus, this section merely documents the COTS products used and ties up some loose auditing ends that were not covered earlier in section 3.1.

3.2.1   Participant Site Audit (Class Model)

See Figure 3.2.1-1 for a graphic UML model of these classes.

The purely COTS solution "classes" are shown in this diagram for completeness:

- *WebTrends* (lower left) retrieves log files and builds reports for each Participant site.
- *What's Up Gold* (bottom left) monitors Participant site availability and response time.

Both of these classes interact with external web sites:

- *Participant Web Site* (lower center) represents one of the "eleven" monitored Participant sites.

The customized part of this subsystem is a Visual Basic component:

- *Fetch Auditor* (top right) examines each entry in the Autonomy Fetch Results log and attempts to match it to a corresponding Staged and Production record. This locates changed and deleted documents. The *Fetch Auditor* also identifies deleted headers.

The *Fetch Auditor* is driven by the content of the log file created by the Autonomy HTTP Fetch:

- Autonomy Fetch Results Log (top left).

The *Fetch Auditor* is built using the following pre-existing components:

- The *Autonomy Search Agent* (upper middle) provides a COM interface (API) to the Autonomy data. One instance of this class searches production data; another, staged data.
- The *ADO Fetch Audit Data* (middle right) provides a COM interface to the SQL Server database of audit events.

The above components provide read/write access to the following stored data. (These data stores are pictured only for completeness. They are not manipulated directly by custom code.)

- *Autonomy Production Data* (middle left)
- Autonomy Staged Content Data (middle center)
- *SQL Fetch Audit Data (*middle right)

3.2.2   Audit Site Availability and Participant Log Files

Because COTS products perform these functions, there are no object models. A brief synopsis of the actions performed by the COTS products is presented here for completeness.

PARTICIPANT LOG FILES

The first major audit performed provides statistics on Participant site page usage. *WebTrends* imports the Participant log files across the web. A configuration file defines frequency, URL and access information (for example, password where required) for the import process. WebTrends

analyzes the log file statistics and stores the aggregated results in a Participant Site Log File Report where they will be viewed later.

PARTICIPANT SITE AVAILABILITY

What's Up Gold periodically monitors key pages on each Participant site to determine if they are available and responding in a timely manner. A configuration file defines the locations to monitor. The results are stored for viewing later.

3.2.3   Audit Document Changes (Object Interaction Model)

See Figure 3.2.3-1 for a graphic, UML representation of this object interaction model and a step-by-step description.

This interaction captures changed and deleted documents. Note that copies of changed and deleted document content remain in the *Autonomy Production Data*.

The *Fetch Auditor* loops over the each record in the fetch log (created during the Participant Document Fetch in section 3.1). However, it processes only the "delete" log entries. Note that if a document is changed, Autonomy writes a "delete" followed by an "add". In other words, there is no "change" entry per se. Strictly by examining the log content, it is not possible to reliably determine if a "delete-add" sequence is a true "change" to one document or a "delete" to one document, followed by an "add" to a different document. This problem is resolved, however, by storing the Autonomy Document ID for the *staged* record in a custom field in the *production* record, and then searching for that ID in the Production data. If there is no matching record, ignore the log entry because this is just some non-document web page that was encountered by the spider.

On the other hand, if the document is found in the production repository, also search for it by Participant URL in the staging repository. If the document *exists* in both places, the "delete" was part of an add-delete pair:

- Record a document change event in the Audit Data.
- Flag the production record as an old copy of a changed record.

If the document *does not exist* in the staged data, the "delete" was part of a true delete transaction:

- Record a document delete event in the SQL Audit Data.
- Flag the production record as deleted.
- Notify the LSNA.

3.2.4   Audit Header Deletes (Object Interaction Model)

See Figure 3.2.4-1 for a graphic, UML representation of this object interaction model and a step-by-step description.

This interaction diagram will detect deleted headers, the final audit item required. This is implemented with a simple SQL query: Select all header records where the "last found" date is yesterday (or the previous spidering date). Since they were not found on the most recent visit, these are headers that were just deleted from the Participant site. For each record found:

- Update the header record status to "deleted".
- Record a header-deleted event in the SQL Audit Data.

This is the end of the auditing process. All required audit statistics have been captured.

3.3   LSN Search (3)

At this point, the *Autonomy Production Content Data* contains a complete set of document content and header fields indexed for searching. This part of the LSN design describes, in software terms, how a search is programmed. This and the remaining parts of the system are relatively straightforward. They follow well-understood Active Server Page (ASP) patterns and are good candidates for iterative design.

3.3.1   LSN Search (Class Model)

See Figure 3.3.1-1 for a graphic UML model of these classes.

Unlike earlier models, this Class Model focuses on web pages. The top part of the model pictures a simple and an advanced search each with its own Hyper Text Markup Lanaguage (HTML) search form:

- *Basic Search* (top left) is an Active Server Page (ASP) page containing general LSN content (navigation links, for example), several search hints and an HTML form (below).
- *Basic Search Form* (upper left) is an HTML form that has basic search input fields.
- *Advanced Search* (top right) is the equivalent ASP page for an advanced search. It contains the Advanced Search Form.
- *Advanced Search Form* (upper right) has an input field for each searchable item. For example: header fields, dates, Participant sites, and content search terms.

Other combinations of searches and form fields are, of course, possible; this section just focuses on a coding approach. Both search forms submit their search criteria to the *Search Summary* page:

- *Search Summary ASP* (middle center) is an ASP server-side page that contains Visual Basic Script code to formulate and execute the search. It scripts an Autonomy-supplied component (below) to execute the search and format the results for display in HTML.
- *Autonomy Search Agent* (middle left) is the scripted component that provides Autonomy search services.
- *Search Summary HTM* (middle center) is a client side page generated by the corresponding ASP page.

The user may click on a link on the summary page to see more details (*Search Details ASP*) or go directly to the document on the Participant site (*Participant Page Document*).

- *Search Detail ASP* (lower center) also contains the Autonomy Search Agent component and uses it to run a detailed query on the clicked-on document.
- *Search Detail HTM* (bottom center) is generated by its corresponding ASP page.
- *Participant Doc Page* (lower right) is a notional representation of a document page on a Participant site. It is outside the scope of LSN, but is shown here for completeness. LSN does not directly deliver up documents or images. It merely provides a page that will hyperlink the user to them.

There is one additional class shown on this page:

- *Autonomy Production Data* (lower left) is accessed indirectly through the *Autonomy Search Agent* component, discussed above, and is only shown here for completeness.

A COMMENT ON "ASP" VERSUS "HTM" PAGES

An explanatory comment on *Search Summary ASP* and its corresponding *Search Summary HTM* class is warranted here. The following is the basic concept behind Active Server Page technology: An ASP page contains script code that is executed by Microsoft's web server, IIS (Internet Information Server). Typically, this script generates HTML output that will be sent to the user's browser. In essence, the server-side ASP page builds or generates the client-side HTML page. Thus, in class diagrams, where relevant you will see the *«Builds»* stereotype on the arrow that points from the ASP page to the HTML page.

This comment applies to *Search Detail ASP* and *HTM* pages, and other pages modeled here. If there is any significant HTML generation on the ASP page it is modeled using the ASP – *«Builds»* – > HTM class pattern.

3.3.2   LSN Search (Object Interaction Model)

See Figure 3.3.2-1 for a graphic, UML representation of this object interaction model and a step-by-step description.

This sequence illustrates only one of many possible paths through the hyperlinks on these HTML pages. The user fills out the form on the Advanced Search page and presses the "submit" button. This sends the contents of the form to the *Search Summary ASP* page. This page contains the *Autonomy Search Agent* component and some VB Script subroutines to format the search and run the Search Agent. Behind the scenes, the component searches the *Autonomy Production Data* and assembles the results. Then, the *Search Summary ASP* page generates the *Search Summary HTM* page. VB Script retrieves each search field from the component and wraps HTML tags around the fields to format the output. The web server delivers this generated HTML, *Search Summary HTM,* to the user's browser.

From the summary page, the user may jump directly to the document on a Participant site or view the details of any one document on LSN. A similar sequence of events occurs to generate a *Search Details* page if the user selects that link.

3.4   LSN Administration (4)

This section describes the types of administration activities available to the LSN Administrator. Additional, system-related maintenance activities, outside the scope of the administrator, are performed locally at the hosting site or remotely via the *LSN Maintenance* package.

3.4.1   The Administration (Class Model)

See Figure 3.4.1-1 for a graphic UML model of these classes.

All access to administration functions start here:

- *Administration* (top left) is an ASP page that acts as a menu of administrative functions, providing links to the four major administrative areas. This page and all other administration pages first check the ASP Session object to assure that the user in indeed an administrator.
- *ASP Session* (top right) is a system component that stores any user-specific information. There is one unique Session object in memory for each current user of the system. Here is how the Session object is used in LSN: It controls what different types or classes of users can see. When a Participant logs on (discussed later in section 3.6), the logon scripts store a *Participant type* flag in the Session object indicating (a) public users, (b) Participant representative users or (c) NRC administrative users. The Administration page and all other relevant pages check this flag to prohibit access to any one page. If a user is not authorized to view a page, he or she is redirected away. Every page will perform this check as its first action; however, it is only shown for the *Administration* page on the class diagram.

COTS products perform many of the functions in the Administration package, specifically the Document/Header Audit Report Details. The administrative user navigates to them by clicking on a link on the Administration page:

- *Participant Performance Report* (middle left) is a set of web pages that display the information assembled by *What's Up Gold*.
- *What's Up Gold (*middle left) collects statistics on each Participant web site.
- *Participant Log file Report* (middle left) is a set of web pages that displays log file reports compiled by *WebTrends*.
- *WebTrends* (middle left) builds reports from the Participant site log files.

Four Document and Header Audit reports are shown first here and detailed in a sub-package covered in section 3.4.4. These reports are hypertext linked together, letting the administrator either point-and-click to drill down into details, or select a report from menu and fill in a search form.

- Summary Doc/Hdr Report lists a count of add/change/deletes by each Participant totaled across a selected date range.

- Daily Doc/Hdr Report lists a count of add/change/deletes by each Participant by each day in the date range.

- Doc Content Detail Report shows the document content for one or more add/change/deleted document.

- Header Detail Report shows the header content for one or more add/change/deleted header.

- Doc/Hdr Audit Reports is the sub-package that shows more detail in section 3.4.4.

Finally, Participant Administration is shown here as a sub-package:

- The *Administer Participants* (middle right) package provides a mechanism for maintaining Participant site and representative information, including logon information. It will be decomposed into more detail in section 3.4.2.

As discussed above for the *Administration* page, all other pages here check the *ASP Session* object to make sure the user is logged on as an administrator. If not, it redirects him or her away.

Most system-level maintenance is performed at the hosting site. There is one other package shown here that provides additional access:

- *LSN Maintenance* (middle right) provides supplemental contractor access to the operating system, database and Autonomy software over a frame relay.

3.4.2   Participant Administration (Class Model)

See Figure 3.4.2-1 for a graphic UML model of these classes.

These classes provide a mechanism for maintaining a database of Participant organizations and Participant representatives. The primary purpose is to let the LSN Administrators control the users who are authorized to log on to the Participant-only servers, and to authenticate users by ID and password when they do log on. The starting point is the Administration menu:

- *Administration* (upper left) is a web page that lets an LSN administrator select "maintain Participants" or some other administrative function.

This brings the Administrator to a list of Participant sites. (Because there are only 11 or so Participant sites, there is no need to provide a way to search for just one.) The following classes manage this list:

- *Participant Org List* (upper center) is a page that lists all Participant sites. There are several links on this page: The user may click on one organization name (or click on "unknown") to jump to a Participant search page. Or, the user may click an "add organization" or "change organization" button to update organization data. This page scripts ADO objects to query the database.
- *Add/Change Org* (middle center) is an ASP server-side page containing an edit form that lets administrators update information on a Participant organization, or add a new organization. It uses ADO objects to update the SQL database.
- *Edit Org Form* (middle left) is the HTML form on the above page.
- *ADO Participant Data* (middle lower) is a set of system-supplied components that manage the SQL database.
- *SQL Participant & Org Data* (middle bottom) is the SQL Database.

The next page is a list of Participants:

- *Participant List* (upper right) has a search form and an ADO component that lets the administrator search for a Participant by name, partial name or ID.

- *Find Participant Form* (upper right) provides the search criteria. The form submits back to the Participant List page where the SQL search is executed.

- *Add/Change/Delete Participant* (middle right) is an ASP page that supports maintaining Participant information. It has a Participant edit form and a set of ADO objects to update the database.

- Edit Participant Form (middle right) is the edit form containing an input field for each value in the Participant's SQL record.

- ADO Participant Data (lower center) are components that provide programmatic access to SQL.

Not shown: each page here checks the *ASP Session* object to make sure the user is logged on as an administrator. If not, it redirects him or her away.

Note: The software activity of adding or updating Participant information, or resetting a password must have a corresponding *manual procedure* to assure that the system is not compromised. For example, to reset a password the administrator should call the Participant back on a known phone number stored in the database and/or mail the password to a known e-mail account and/or ask the Participant some "shared secret" to authenticate him. Furthermore, the system will force the user to change his or her password on logon to a new password not known to the administrator.

3.4.3   Administer Participants (Object Interaction Model)

See Figure 3.4.3-1 for a graphic, UML representation of this object interaction model and a step-by-step description.

This sequence describes searching for, selecting and updating one LSN Participant representative. This is a typical web page interaction. The code itself is written in VB Script (as are all the ASP pages). As is typical, the script manipulates several built-in objects: ADO objects (the *Connection, Command and Recordset*), the Find Participant form (*Request.Form*), and the HTML output (*Response*).

This sequence starts after the administrator clicks on a Participant organization on the *Participant Org List* page. This links or jumps to the *Participant List* page. The administrator then enters the Participant name search criteria into the HTML form and presses the "submit" button. He may enter any combination of fields. The *Participant List* page submits its form back to itself. (In other words the form's Action = "ParticipantList.ASP"). The ASP scripting code

then gets any non-blank values from the request form, constructs an appropriate where clause ("Where Last_Name like 'Gra%' and Participant_Organization_ID='NRC'", for example) for the SQL query. It then connects to the database and executes the SQL Select command using the ADO objects.

ADO returns a Recordset object containing one row for each person matching the search criteria. The script then loops over each row in the Recordset object, wrapping HTML tags around the values of each field to generate the client-side HTML page: "<tr> <td>" & *Recordset.Value(0)* & "</td> <td>"…, for example. Typically, one of the fields has a hypertext link wrapped around it so the administrator can click on it to select it: " … ".

When the administrator clicks on the link, the *Participant List* page links to the *Add/Change/Delete Participant* page where the administrator can edit and update all Participant data. As its name implies, the *Add/Change/Delete Participant* page may also be used to add a new Participant or delete an old one.

3.4.4   Document/Header Audit Report Details (Class Model)

This is a set of administrative reports on Participant site's added, changed and deleted headers and documents. These are built in ASP using ADO to report from the *SQL Fetch Audit Data* and the *SQL Header Data*. The administrator may either select a report from a menu and fill in a form, or click on a report line to drill down to a more detailed report. Here are the reports and report forms, in ascending levels of detail:

- *Summary Doc/Hdr Form* lets the administrator select a date range to span across. This may be one day, a week or any desired period of time.
- *Summary Doc/Hdr Report* lists a count of added, changed, deleted headers and documents by each Participant totaled across the selected date range. Clicking on one report field drills down to the Daily Report. Filling out the form re-runs this report. A sample report may appear as follows:

LSN Summary Add Change Delete Report for 1 Dec 2001 to 31 Dec 2001:

Date	Added Headers	Added Docs	Changed Headers	Changed Docs	Deleted Headers	Deleted Docs
NRC	126	126	2	1	1	2
DOE	25	25	2	2	0	0
Nevada	10	10	0	0	1	0
…						

- *Daily Doc/Hdr Form* lets the administrator select a date range and one or more Participant sites.

- *Daily Doc/Hdr Report* lists a count of added, changed, deleted documents and headers by each Participant by each day in the date range. Each Participant site starts on a new page. Clicking on one report field drills down to the *Doc Content* or *Header Content Detail Report*. Filling out the form re-runs this report. A sample report may appear as follows:

LSN Daily Add Change Delete Report for NYE from 1 Dec 2001 to 31 Dec 2001:

Date	Added Headers	Added Docs	Changed Headers	Changed Docs	Deleted Headers	Deleted Docs
1 Dec 2001	101	101	0	0	1	2
2 Dec 2001	25	25	2	1	0	0
3 Dec 2001	0	0	0	0	0	0
…						

- *Doc Content Detail Form* lets the administrator select any or all of the following fields: LSN Accession Number; Participant Accession Number; Participant; From-To date range. (Bypassed on drill-down.)

- *Doc Content Detail Report* shows the document content for one or more added, changed, deleted documents from the Autonomy Production Data.

- *Header Detail Form* lets the administrator select fields, as described above.

- *Header Detail Report* shows the header content for one or more added, changed, deleted headers from SQL Server Header Data.

These reports use the following components to access SQL Header content, SQL Header Audit events and Autonomy content.

- *ADO Header Audit Data*
- *ADO Header Data*
- *Autonomy Agent*

3.5   LSN General Web Content (5)

3.5.1   General Web Content (Class Model)

See Figure 3.5.1-1 for a graphic UML model of these classes.

This class model portrays several ASP implementation details in the general web content of LSN.

- *Participant Login* (top left) is an ASP page where Participant representatives enter their user IDs and passwords. This is covered in section 3.6.

After logging on, a Participant is automatically redirected to the LSN Home page. There are several implementation details of interest here:

- *LSN Home* (top center) is the home page. It includes in its HTML two other page fragments, search and navigation, described below.
- *Simple Search* (top right) is a page fragment that includes a simple search form. Since the primary purpose of LSN is to provide a search capability, a simple search should be commonly available on most LSN pages. Thus, it will be implemented as an "include" file. (It is only modeled as included on the Home page here to reduce line clutter.)
- *Navigation Options* (upper center) is another page fragment that will be included on most pages (but only modeled on one, again to reduce line clutter). It will be a dynamically constructed "navigation bar" containing hyperlinks (<a href=…>) to all common LSN pages. The *Navigation Options* page fragment uses information in the ASP Session to dynamically configure itself for each user type. For example, only administrators will see the "Administration" link; guests will not be able to save agents; etc. This is covered in detail in sequence diagram 3.5.2 below.
- *ASP Session* (lower right) is a built-in ASP object that will contain the Participant type for the logged on user. The current types are "Participant", "administrator" and "guest" (anyone not logged on).

Fanning out from *Navigation Options* is the set of general content pages: About LSN, FAQs, Privacy/Security Statement, Help, Searches, etc. This is straightforward web content, not requiring any object-oriented modeling. There is, however, one additional detail worth noting:

- *Administration* (lower center) is the administrator's menu. This and all other administrative pages will have NT security permissions set on them, requiring the administrator to have an NT account on the server. This will provide an extra measure of security for administrative functions.
- *NT Security* (lower right) represents the Windows 2000 (NT-5) system-provided challenge/response security.

3.5.2 Configure Participant Page (Object Interaction Model)

See Figure 3.5.2-1 for a graphic, UML representation of this object interaction model and a step-by-step description.

This sequence shows how information in the ASP Session system-supplied object is used to configure the navigation bar options and to keep non-administrators out of administrative areas.

Script code in the *Navigation Options* "include" file gets the Participant type flag from the *ADO Session* object. If the user is a Participant or administrator, it generates any additional Participant links. If the user is an administrator, it generates administration-only links. Thus, public users never see or have access to the links. This alone is not sufficient, however, because any user could get to any page by typing in the URL directly into his browser. For example:

- http://www.lsnnet.gov/admin/AddChangeDelParticipant.ASP

Thus, each restricted page will also get the Participant type from the ASP Session and redirect non-administrators away as needed.

3.6 Participant Representative Login (5.1)

3.6.1 Participant Representative Login (Class Model)

See Figure 3.6.1-1 for a graphic UML model of these classes.

- *Participant Login* (top center) is the login page containing an HTML form and ADO objects.

- *Login Form* (upper left) is the HTML form containing the login fields: User ID Password, New Password (for changing a password). The password is not displayed on the screen as it is typed.

- *LSN Home* (top right) is the standard home page. Participants are automatically redirected here after successfully login.

- *Login Help* (middle center) provides additional help and an LSN Administrator contact point for login or password problems.

- *ADO Login* (lower left) are the system provided ADO database manipulation objects.

- *SQL Login Data* (bottom left) is the SQL database.

- *ASP Session* (bottom center) is the system provided session object.

3.6.2   Participant Representative Login (Object Interaction Model)

See Figure 3.6.2-1 for a graphic, UML representation of this object interaction model and a step-by-step description.

This page performs straightforward ASP Form and ADO object scripting. The user enters his or her ID and password. The script constructs the SQL query using the form fields to build an appropriate WHERE clause (for example: WHERE UserID='xxx' and Password='yyy'), connects to the database, executes the query and receives an ADO Recordset back from SQL.

If there are no rows in the Recordset, either the ID or the Password or both were incorrect. The script generates an appropriate error and counts the failed login attempt. If there is exactly one row in the Recordset, the login was successful. The script stores the Participant type (role) and Participant ID in the ADO Session object.

Optionally, if the failed login attempt exceeds a threshold, the script disables the Participants account and generates an e-mail to the administrator.

If the user is changing his or her password, the page checks for a minimally-acceptable password, prompts the user to re-enter the password, and records it in the database.

3.7   Physical Data Model

This section and Appendix B highlight the LSN physical data model and related mappings to the logical data model. The physical data model is presented in Figures 3.7-1 through 3.7-3 located in Appendix D. This physical data model derives from the logical data model previously documented in the LSN Logical Design Document. As per the NRC SDLCM, this section discusses how that derivation took place. Specifically, the following are discussed:

- the mapping of logical objects to physical objects
- the mapping of logical objects to schema objects
- the completion of the physical data model (PDM)

Finally, the data definition language (DDL) for the LSN database is provided in Appendix E.

3.7.1   General Mapping of the Logical Model Objects to the Physical Model Objects

ERwin version 3.5.2 was used to generate the logical data model, and therefore was used to generate the physical data model. Table 3.7.1-1 below shows how the logical data model objects map to the physical data model objects.

Logical Design Logical Data Model Objects	Physical Design Relational Schema
Entity Types	Relational Tables
Attribute Types	Column

Table 3.7.1-1 Logical Data Model Objects

The LSN data model is fairly simple, therefore information types and sub-set types were not required to properly model the data.

3.7.2   Specific Mapping of the Logical Model Objects to the Relational Schema

This section illustrates the mapping between the logical model attributes and entities, and the relational schema tables and fields. Refer to Appendix B, Table 3.7.2-1 for the Entity and Attribute mapping to the Tables and Columns of the LSN physical data model. Refer to Appendix B, Table 3.7.2-2 for the Logical to Physical relationships mappings.

3.7.3   Completing the Physical Design

After the logical model was translated into the physical model, the physical model was augmented to provide the connectivity between tables required for LSN (see Appendix D, Figures 3.7-1 through 3.7-3). Primary key indexes were created for each table. The Participant_Participant_Organization table was created to handle the many-to-many relationship between the Participant_Organization table and the Participant table. Foreign key indices were created to speed access from the Header table to its sub-ordinate tables. None of the indexes are clustered.

The LSN physical data model uses no views, data structure re-definitions, or column name changes.

3.7.4   Data Definition Language

See Appendix E for the Data Definition Language.

Appendix A. Requirements to Design Traceability Matrix

The "LSN Req. No." and "Description" columns in the tables below correspond to the requirements received in the NRC LSN Statement of Work. The last column contains either the software Component Number or the hardware Configuration Item Name. The five (5) major software components are:

4. (1) Participant Document Fetch

5. (2) Participant Site Audit

6. (3) LSN Search

7. (4) LSN Administration

8. (5) LSN General Web Content (includes Participant Representative Login)

6. (5.1) Participant Login

Functional Area 1: General Characteristics of Central LSN Site System

LSN Req. No.	Description	Component / CI
LSN DR B-1 (new)	The LSN shall be configured with hardware sufficient to store and serve all information (e.g., sufficient disk space, RAM (Random Access Memory), processing power, network interfaces, etc.) at estimated usage levels, and to be easily upgradeable, should estimates fail to adequately characterize usage. Concurrent usage is estimated at 150 users during peak periods.	5
LSN DR B-1.1 (new)	The system shall be capable of limiting access to only priority users in the event that spikes in usage occur.	5
LSN DR B-2 (formerly LSN 1.02)	LSN components shall be integrated using modular design techniques and well-documented interfaces that allow individual components of the system to be replaced without significantly impacting other components.	1,2,3,4,5
LSN DR B-3 (new)	The system shall be configured with software, licensed at sufficient levels, to store and serve all documentary and associated materials (e.g., networking-capable operating system), web server software, HTML authoring and site maintenance software, database server, etc.) at anticipated usage levels.	5
LSN DR B-4 (new)	The system shall be designed to maintain the security of the collection and the system itself including the ability to deny unauthorized access or update privileges, detect and defeat compromise attempts, and defend against denial of service attempts.	4,5

LSN Req. No.	Description	Component / CI
LSN DR B-5 (new)	The system shall be designed to maintain the integrity of the collection and provide for timely recovery in the event of a hardware or software failure with complete restoration of the central LSN site within three working days, or earlier if so specified within the parameters of the NRC's disaster recovery plan.	5
LSN DR B-6 (formerly LSN 1.03)	The LSN shall adhere to established Federal Government, international, and/or industry hardware and software standards appropriate to meet the intent of the Rule.	1,2,4,5
LSN DR B-6.1 (formerly LSN 2.03.01)	The standard for network access shall be HTTP/1.1 [http://www.faqs.org/rfcs/rfc2068.html] over TCP (Transmission Control Protocol, [http://www.faqs.org/rfcs/rfc793.html]) over IP (Internet Protocol, [http://www.faqs.org/rfcs/rfc791.html]).	1,2,3,4,5
LSN DR B-6.2 (formerly LSN 2.03.02)	The standard for associating server names with IP addresses shall be the DNS (Domain Name System), [http://www.faqs.org/rfcs/rfc1034.html] and [http://www.faqs.org/rfcs/rfc1035.html].	1,2,3,4,5
LSN DR B-6.3 (formerly LSN 2.03.03)	The standard for web page construction shall be HTML version 4.01 [http://www.w3.org/TR/REC-html40/].	3,4,5
LSN DR B-6.4 (formerly LSN 2.03.04)	The standard for electronic mail (e-mail) exchange between e-mail servers shall be SMTP (Simple Mail Transport Protocol, [http://www.faqs.org/rfcs/rfc821.html]).	5
LSN DR B-6.5 (formerly LSN 2.03.05)	The standard for the format of an electronic mail message shall be per [http://www.faqs.org/rfcs/rfc822.html] optionally extended by MIME (Multimedia Internet Mail Extensions) per [http://www.faqs.org/rfcs/rfc2045.html] to accommodate multimedia e-mail.	5
LSN DR B-7 (formerly LSN 1.04)	The LSN shall provide for electronic exchange of information. This function shall allow users to identify and receive electronic documents (e.g., motions, filings, orders, decisions, etc.).	3
LSN DR B-7.1 (formerly LSN 1.05)	The LSN shall utilize an electronic information exchange function that provides for an electronic acknowledgment that a motions practice document has been transmitted to a requester. The acknowledgment shall include, as a minimum, the name and electronic address of the recipient and the date the document was delivered.	NA
LSN DR B-8 (formerly LSN 1.06)	The LSN shall be capable of electronically storing and retrieving bibliographic headers in the system.	1
LSN DR B-8.1 (formerly LSN 2.06.01)	Bibliographic headers will contain all fields as described in Table A.	1

LSN Req. No.	Description	Component / CI
LSN DR B-8.2 (formerly LSN 2.06.02)	Bibliographic headers will be stored in a manner that they can be retrieved through reference to any field as designated in Table A.	3,4
LSN DR B-8.3 (formerly LSN 2.06.03)	Bibliographic headers will be stored in a manner that the contents of their fields can be searched for specific data.	3,4
LSN DR B-9 (formerly LSN 1.07)	The LSN shall be capable of storing electronic indexes for use in searching and retrieving document texts made available by the Participants.	1,3
LSN DR B-10 (formerly LSN 1.08)	The LSN shall allow identification of text through queries of the occurrence of text content through all document texts. Specific query options include the ability to search for the occurrence of a phrase in the full text of documents, perform proximity searching (i.e., search for phrases near each other or near the beginning or end of the document), perform wild card searching, perform root searching, perform frequency searching, and to arbitrarily combine any search strategy through the use of Boolean operators.	3,5
LSN DR B-11 (formerly LSN 1.09)	The LSN shall be capable of storing electronic indexes for use in searching and retrieving digital images of each page of graphic-oriented documentary material made available by the Participants.	3,5
LSN DR B-12 (formerly LSN 1.10)	The system shall allow users to easily view (by clicking) and print (by selecting "file/print" from the menu) documents of the following acceptable formats: ASCII, native word processing (Word and WordPerfect versions as requested by Participants), PDF Normal, or HTML.	5
LSN DR B-13 (formerly LSN 1.11)	The LSN shall provide tools to assist users in identifying documentary material consistent with the technology.	3,5
LSN DR B-13.1 (new)	Documentation shall be prepared and published on the central LSN site describing how to use the features of the website, specifically the search and retrieval functions.	3,4,5
LSN DR B-13.2 (new)	The system must provide finding aids to users to include online help on use of the site, usage guidelines, and contact information for further assistance.	5
LSN DR B-13.3 (new)	The LSN shall be capable of displaying bibliographic header information from the hit list resulting from a search.	3,5
LSN DR B-13.4 (new)	The system shall default searches based on relevancy ranking. The system shall state that relevancy ranking is machine-generated.	3,5
LSN DR B-14 (formerly LSN 1.12)	The LSN shall provide a user interface compatible with current browser technologies including access using both graphical and text-only browsers to documentary collections.	5

LSN Req. No.	Description	Component / CI
LSN DR B-14.1 (formerly LSN 2.13.02)	The system must be designed to be accessible to Internet users with web browsers currently supporting HTML 4.0 or higher (i.e., not just Netscape or Internet Explorer). The system shall deliver a consistent organization and style (including but not limited to colors, fonts, spacing) throughout the central LSN site independent of the browser used to access the LSN.	3,5
LSN DR B-15 (formerly LSN 1.13)	The LSN shall support non-interactive access to the central LSN by web "crawlers."	1,2
LSN DR B-15.1 (formerly LSN 2.13.01)	Web pages must be authored in compliance with the Web Content Accessibility Guidelines for access by individuals with disabilities [http://www.w3.org/TR/WAI-WEBCONTENT/].	5
LSN DR B-16 (formerly LSN 1.14)	The LSN shall be designed to allow the LSN Administrator to coordinate the availability and the integrity of the information available via the LSN.	4
LSN DR B-16.1 (formerly LSN 1.15)	The system shall provide the LSN Administrator with access to Participant collections necessary and sufficient to allow the LSN Administrator to independently verify the integrity of data available via the LSN.	1,2,4
LSN DR B-16.2 (formerly LSN 2.15.01)	The system shall have the capability of non-interactively "crawling" Participant LSN websites, and fetching a subset or the entire website for analysis.	1,2
LSN DR B-16.3 (new)	The system shall be able to access an electronic log of all retrievals of documentary material from each Participant's LSN website. This log will contain the IP address or DNS host name of the recipient's computer and the date and time of delivery. The log shall be in either the web standard "Common Log Format" or "Combined Log Format."	2,4
LSN DR B-16.4 (formerly LSN 2.14.01)	A monitoring/audit station shall be established to allow the LSNA to obtain, store, and report information on the availability and integrity of LSN information.	4
LSN DR B-16.5 (formerly LSN 2.14.04)	The monitoring/audit station shall have the capability of tracking changes on Participant LSN websites, monitoring Participant LSN website responsiveness and other performance characteristics, and reporting this information to the LSNA.	4
LSN DR B-16.6 (new)	The LSN shall provide a means for the LSNA to post announcements and notices on the central LSN site.	5
LSN DR B-17 (new)	The LSN shall not use "persistent cookies" (i.e., ongoing capture of data that tracks a user's pattern of use and preferences) without the approval of the Chairman of the NRC.	5

Functional Area 2: LSN Administrator (LSNA) Related Requirements

LSN Req. No.	Description	Component / CI
LSN DR C-1 (formerly LSN 2.14.01)	The LSNA shall obtain, store, and report information on the availability and integrity of LSN information.	2,4
LSN DR C-1.1 (new)	The LSNA shall identify any problems experienced by Participants regarding LSN availability, including the availability of individual Participant's data, and provide a recommendation to resolve any such problems to the Participant(s) and the Pre-License Application Presiding Officer relative to the resolution of any disputes regarding LSN availability, including disputes on the availability of an individual Participant's data.	4
LSN DR C-1.2 (new)	The LSNA shall post announcements on the central LSN site about the overall LSN program, items of interest (hour of availability, scheduled outages, etc.), and procedures for a user to acquire authenticated image copies from a Participant.	5
LSN DR C-1.3 (new)	The LSNA shall identify any problems regarding the integrity of documentary material certified in accordance with the LSN Rule by the Participants to be in the LSN, and provide a recommendation to resolve any such problems to the Participant(s) and the Pre-License Application Presiding Officer relative to the resolution of any disputes regarding the integrity of documentary material.	4
LSN DR C-1.4 (new)	The LSNA shall post notices that contain listings of changes, if any, to each Participant's collection, identified by LSN accession number, with a description of what the change was and why it was necessary.	5
LSN DR C-2 (new)	The LSNA shall review all Participant LSN website designs to ensure that they meet the design standards.	NA
LSN DR C-2.1 (new)	The LSNA shall have the authority to allow variances from the design standards to accommodate changes in technology or problems identified during initial operability testing of the individual Participant LSN websites or the central LSN site.	NA
LSN DR C-2.2 (new)	The LSNA may develop and issue guidance for LSN Participants on how best to incorporate the LSN standards in their system.	NA

Functional Area 3: General Capabilities of Participant LSN Site Systems

LSN Req. No.	Description	Component / CI
LSN DR D-1 (formerly LSN 1.20)	Each LSN Participant must obtain the computer system necessary to comply with the requirements for document production and service.	NA
LSN DR D-1.1 (formerly LSN 2.20.01)	The Participant's system must provide the function of HTTP service. HTTP service may be provided by a dedicated computer, a virtual server (dedicated computer hosting multiple web sites), or be provided by a commercial web hosting service (that can comply with requirements).	NA
LSN DR D-1.2 (formerly LSN 2.20.02)	The Participant's system must be configured with hardware sufficient to store and serve all documentary and associated materials (e.g., sufficient disk storage, RAM (Random Access Memory), processing power, network interface, etc.) at estimated usage levels and to be easily upgradeable should estimates fail to adequately characterize usage.	NA
LSN DR D-1.3 (formerly LSN 2.20.03)	The Participant's system must be configured with software, licensed at sufficient levels, to store and serve all documentary and associated materials (e.g., networking-capable operating system, web server software, HTML authoring and site maintenance software, database server, etc.) at anticipated usage levels	NA
LSN DR D-1.4 (formerly LSN 2.20.04)	The Participant's system must be designed to maintain the integrity of the Participant's discovery collection documentary material and provide for timely recovery in the event of hardware or software failure with complete restoration of the Participant LSN website within three working days, or earlier if so specified within the parameters of the Participant's disaster recovery plan.	NA
LSN DR D-1.5 (formerly LSN 2.14.02)	The Participant's system shall allow monitoring of various parameters by a monitoring station established by the LSNA to track changes on the Participant LSN website, website responsiveness, and other performance characteristics. Specific access shall include SNMP monitoring of network utilization and ICMP access for determination of certain performance characteristics as well as access to the normal web distribution facility.	2,4
LSN DR D-1.6 (formerly LSN 2.14.03)	The Participant's system shall allow LSNA access to their logs of electronic transactions in raw and summary formats to enable tracking of site usage.	2,4

LSN Req. No.	Description	Component / CI
LSN DR D-1.7 (formerly LSN 2.20.05)	The Participant's system must be designed to maintain the security of the Participant's discovery collection documentary material and the system itself including the ability to deny unauthorized access or update privileges, detect and defeat compromise attempts, and defend against denial of service attempts	4,5
LSN DR D-1.8 (formerly LSN 2.20.06)	The Participant's system must be connected to the Internet with the capability of being accessed by any Internet user. The Participant's system will allow Internet users the ability to retrieve documentary material from the Participant's LSN website without utilizing a proxy from the LSN server. This connection shall be sufficient to provide reasonable responsiveness during periods of normal usage.	NA
LSN DR D-1.9 (formerly LSN 2.20.07)	Participants shall make textual (or, where non-text, image) versions of their discovery collection documents available on an Internet accessible server which is able to be canvassed by web indexing software (i.e., a "robot," "spider," "crawler") and the Participant's system must make both data files and log files accessible to this software.	1,2
LSN DR D-1.10 (formerly LSN 2.20.08)	Non-LSN-related information may be maintained on the same Participant website as LSN-related material, but must be kept logically separate. All LSN materials on a Participant's site must be maintained together within a single hypertext sub-tree. The entire LSN-related sub-tree must be able to be navigated under a single URL (Uniform Resource Locator) reference. If a Participant LSN website contains site navigation links on its LSN-related pages (e.g., a "home" button, or the result of a search), these links must point exclusively within the LSN-related sub-tree and not to another part of the World Wide Web site or off site.	5
LSN DR D-2 (formerly LSN 1.29)	Each LSN Participant must design and implement their web facility to ensure acceptable access and responsiveness consistent with performance specifications.	2
LSN DR D-2.1 (formerly LSN 2.29.01)	Sites must be provisioned to be able to satisfy not less than 500 web page requests per minute.	5
LSN DR D-2.2 (formerly LSN 2.29.02)	Sites must be provisioned to be able to deliver a web page or image page on average in not more than five seconds to a web browser located on the same LAN segment.	5
LSN DR D-2.3 (formerly LSN 2.29.03)	Communications between the server and the Internet must be provisioned to be able to deliver interactive response.	5
LSN DR D-3 (new)	To facilitate data exchange, the Participant's system shall adhere to established hardware and software standards appropriate to meet the intent of the LSN Rule.	NA

LSN Req. No.	Description	Component / CI
LSN DR D-3.1 (new)	The standard for network access shall be HTTP/1.1 [http://www.faqs.org/rfcs/rfc2068.html] over TCP (Transmission Control Protocol, [http://www.faqs.org/rfcs/rfc793.html]) over IP (Internet Protocol, [http://www.faqs.org/rfcs/rfc791.html]).	NA
LSN DR B-3.2 (new)	The standard for associating server names with IP addresses shall be the DNS (Domain Name System), [http://www.faqs.org/rfcs/rfc1034.html] and [http://www.faqs.org/rfcs/rfc1035.html].	NA
LSN DR B-3.3 (new)	The standard for web page construction shall be HTML version 4.01 [http://www.w3.org/TR/REC-html40/].	NA
LSN DR B-3.4 (new)	The standard for electronic mail (e-mail) exchange between e-mail servers shall be SMTP (Simple Mail Transport Protocol, [http://www.faqs.org/rfcs/rfc821.html]).	NA
LSN DR B-3.5 (new)	The standard for the format of an electronic mail message shall be per [http://www.faqs.org/rfcs/rfc822.html] optionally extended by MIME (Multimedia Internet Mail Extensions) per [http://www.faqs.org/rfcs/rfc2045.html] to accommodate multimedia e-mail.	NA
LSN DR D-4 (formerly LSN 1.17)	Each LSN Participant must establish its own procedures to make its own documentary material available.	NA
LSN DR D-4.1 (formerly LSN 2.17.01)	Procedures shall cover all aspects of the production and web publication process including (as applicable on a per-document basis) authorship, content guidelines, stylistic guidelines, distribution guidelines, maintenance and revision guidelines, format conversion, quality assurance, uploading to the web server, accessing documentary material, and removal from the web server.	NA
LSN DR D-5 (formerly LSN 1.18)	Each LSN Participant must train its own staff on how to make its documentary material available	NA
LSN DR D-5.1 (formerly LSN 2.18.01)	Staff shall be trained in document production, conversion, and web publication in compliance with acceptable procedures.	NA
LSN DR D-5.2 (formerly LSN 2.18.02)	Staff shall be trained in operation and maintenance of the web server system.	NA
LSN DR D-5.3 (formerly LSN 2.18.03)	Staff shall be trained in operation and maintenance of the Participant LSN website.	NA
LSN DR D-5.4 (formerly LSN 2.18.04)	Staff shall be trained in performing the user assistance or help desk function, if applicable.	NA

LSN Req. No.	Description	Component / CI
LSN DR D-6 (formerly LSN 1.22)	DOE must provide electronic access to the central LSN site at DOE Headquarters and at all DOE Local Public Document Rooms (LPDRs) in the vicinity of the proposed site for a geologic repository, including Las Vegas, Reno, Carson City, Nye County and Lincoln County.	NA
LSN DR D-6.1 (new)	Access must be provided no later than eight months in advance of submitting its license application to receive and possess high-level radioactive waste at a geologic repository operations area.	NA
LSN DR D-6.2 (formerly LSN 2.22.01)	Access to the central LSN site is to be provided to the public on computers equipped with a web browser.	NA
LSN DR D-6.3 (formerly LSN 2.22.02)	Access to the central LSN site is to be provided to visually impaired and otherwise disabled individuals as needed through appropriate hardware and software or by provision of user assistance.	NA
LSN DR D-7 (new)	The NRC must provide electronic access to the central LSN site at the NRC Public Document Room no later than eight months in advance of DOE's submitting its license application to receive and possess high-level radioactive waste at a geologic repository operations area.	NA
LSN DR D-7.1 (new)	Access to the central LSN site is to be provided to the public on computers equipped with a web browser.	NA
LSN DR D-7.2 (new)	Access to the central LSN site is to be provided to visually impaired and otherwise disabled individuals as needed through appropriate hardware and software or by provision of user assistance.	NA
LSN DR D-8 (new)	Participants should provide items of interest about their LSN website (hours of availability, scheduled outages, etc.) to the LSNA to post on the central LSN site. These items may also be posted on the Participant's LSN website.	NA

Functional Area 4: Document Production and Service

LSN Req. No.	Description	Component / CI
LSN DR E-1 (formerly LSN 1.16)	Each LSN Participant must designate an official who is responsible for the administration of making its documentary material available.	NA
LSN DR E-1.1 (formerly LSN 2.16.01)	Each designated official must provide contact information to the LSNA including a telephone number, e-mail address, and postal address.	NA
LSN DR E-1.2 (formerly LSN 1.19)	Each LSN Participant must have the designated official certify to the Pre-License Application Presiding Officer, in accordance with §2.1009, that procedures have been implemented and that documentary material has been made electronically available, or as otherwise indicated in the LSN Rule.	NA
LSN DR E-2 (formerly LSN 1.23)	Each LSN Participant must prepare and publish its documentary collections in a manner that allows access by Internet users who have access to the LSN search and retrieval capabilities.	NA
LSN DR E-2.1 (formerly 2.23.01)	All participating entities must provision their web server with enough storage to accommodate all headers plus text and/or images of their entire collection of relevant documents as specified in the LSN Rule	NA
LSN DR E-2.2 (formerly LSN 2.23.02)	All participating entities shall ensure access to their documentary collection through electronic means. Participant capabilities should be such that any Internet user using a web browser and the LSN search and retrieval capabilities will be able to locate, identify, and retrieve documents of interest in relevant formats (header, text, and/or image).	1,2,5
LSN DR E-2.3 (formerly LSN 2.23.03)	Each Participant must provide documents in a format that allows their presentation through a web server, including a header plus text and/or image portion**.**	NA
LSN DR E-2.4 (formerly LSN 2.23.04)	All participating entities must provide the LSN with access to its full text or image files as required by the LSN Rule.	NA
LSN DR E-2.5 (formerly LSN 2.23.05)	Documents presented on a Participant web server must be an accurate representation of the source document.	NA
LSN DR E-2.6 (formerly LSN 2.23.06)	All participating entities must store each TIFF document image in a page per file format**.**	NA
LSN DR E-2.7 (formerly LSN 2.23.07)	All participating entities must provide the capability to retrieve and deliver documents identified through searching or browsing performed at the LSN portal site.	NA
LSN DR E-3 (formerly LSN 1.24)	LSN Participant must ensure that each document on their system has a unique ID (Participant Accession Number).	1

LSN Req. No.	Description	Component / CI
LSN DR E-3.1 (formerly LSN 2.24.01)	Each representation of a document (text and/or image) must be index-accessible through its Participant Accession Number. Participants shall programmatically link the bibliographic header record with the text and/or image file it represents to provide for file delivery and display from Participant machines using the central LSN site. The bibliographic header must contain fielded data identifying its associated object (text and/or image).	3
LSN DR E-3.2 (formerly LSN 2.24.02)	It must be possible to retrieve a document by providing its Participant Accession Number.	3,5
LSN DR E-3.3 (formerly LSN 2.24.03)	It must be possible to retrieve an individual document which is part of a compound document (package) by providing each individual Participant Accession Number.	3,5
LSN DR E-4 (formerly LSN 1.25)	Each LSN Participant must follow data format standards to facilitate electronic exchange and transfer.	NA
LSN DR E-4.1 (formerly LSN 2.25.02)	Textual material shall be formatted to comply with the ISO/IEC 8859-1 character set and be in one of the following acceptable formats: ASCII, native word processing (Word and WordPerfect versions as requested by Participants), PDF Normal, or HTML. As a goal, textual documents should be accurately represented with an overall error rate of no more than 0.5% based on character accuracy and a per page error rate of no more than 1.5%. Documents converted through means other than OCR should have an error rate of less than 0.05%.	NA
LSN DR E-4.2 (formerly LSN 2.25.03)	Image file format shall be TIFF CCITT G4 for bi-tonal images or PNG (Portable Network Graphics) per [http://www.w3.org/TR/REC-png- multi.html] format for grey scale or color images, or PDF (Portable Document Format). TIFF, PDF, or PNG images will be stored at 300 dpi (dots per inch) or greater, grey scale images at 150 dpi or greater with eight bits of tonal depth, and color images at 150 dpi with 24 bits of color depth. Images shall be stored as single image-per-page to facilitate retrieval of no more than a single page. Alternatively, images may be stored in a page-per-document format if software is incorporated in the web server that allows single-page representation and delivery.	NA
LSN DR E-5 (formerly LSN 1.26)	Each LSN Participant must provide a bibliographic header with each document and with other material submitted (e.g., videotape or photograph). This includes submissions for which no text or image is available (e.g., rock sample) and for privileged, confidential, safeguards and other types of limited access documentary material as specifically identified.	NA

LSN Req. No.	Description	Component / CI
LSN DR E-5.1 (formerly LSN 2.25.01)	Bibliographic header data shall be available in an HTTP accessible, ODBC (Open Database Connectivity) and SQL (Structured Query Language)-compliant (ANSI IX3.135-1992/ISO 9075-1992) database management system (DBMS). Alternatively, the structured data containing the bibliographic header may be made available in a standard database readable (e.g., XML Extensible Markup Language http://www.w3.org/xml/), comma delimited, or comma separated value (.csv) file.	1
LSN DR E-5.2 (new)	Bibliographic headers shall contain all fields as described in Table A, as applicable to Participants (i.e., all fields except the LSN Accession Number which is generated by the LSN).	NA
LSN DR E-5.3 (formerly LSN 2.26.01)	Headers for limited access documentary material will be as those for full access documentary material.	NA
LSN DR E-5.4 (formerly LSN 2.26.02)	Headers for limited access documentary material shall be logically organized on the Participant LSN website in a list, index, or table of contents separate from documentary material that is publically accessible, with user instructions for reviewing the limited access material.	NA
LSN DR E-6 (new)	Participants may correct or revise documentary already made available on their LSN websites.	NA
LSN DR E-6.1 (new)	Changes to documentary material previously provided are permitted if (1) a corrected or updated version is noted as superseding a previously provided version; and (2) the previous version is not removed.	NA
LSN DR E-6.2 (new)	The Participant must notify the LSNA of the change, identified by LSN Accession Number, with a description of what the change was and why it was necessary, so it can be posted on the central LSN site. Notification may also be posted on the Participant's LSN website.	NA
LSN DR E-7 (formerly LSN 1.27)	NRC, DOE, and each other potential party, interested governmental Participant or party shall provide a statement that indicates where an authenticated image copy of the document can be obtained for those where an image is not required to be made available online.	NA
LSN DR E-7.1 (formerly LSN 2.27.01)	Documents for which the electronic image is not available on the Participant LSN website may be made available by means of authenticated image copy distribution. The website shall contain the document's header record and its text representation.	NA
LSN DR E-7.2 (formerly LSN 2.27.03 and LSN 2.27.04)	Requests and delivery for an authenticated image copy will be via procedures and processes established and operated by the Participants.	NA

LSN Req. No.	Description	Component / CI
LSN DR E-7.3 (formerly LSN 2.27.02)	Each Participant's procedures and processes for a user to acquire images shall be submitted to the LSNA for posting on the LSN home page.	5
LSN DR E-8 (formerly LSN 1.21)	All documentary material not provided to other Participants in electronic form (i.e., not included in the LSN) must be identified in an electronic notice. Any item requested from that list must be provided to the requestor within five business days.	NA
LSN DR E-8.1 (formerly LSN 2.21.01)	Notice of availability may be achieved by publication on the relevant Participant LSN website.	NA

Functional Area 5: Timeliness

LSN Req. No.	Description	Component / CI
LSN DR F-1 (formerly LSN 1.28)	DOE must make its documentary material available no later than eight months before submitting its license application. The NRC must make their documentary material available 30 days after the DOE initial certification of compliance in accordance with §2.1009. All other Participants must make their documentary material available 90 days after the DOE initial certification of compliance.	NA
LSN DR F-1.1 (formerly LSN 2.28.0)	The computer system providing document production and service must be designed, specified, acquired, integrated and installed sufficiently in advance of the specified date to meet the availability criteria. Customary funding and procurement lead times must be considered when scheduling these actions.	NA
LSN DR F-1.2 (formerly LSN 2.28.02)	Document conversion and Participant LSN website page authoring and document collection population must be begun sufficiently in advance of the specified date to meet the availability criteria. Collection size and resource availability must be considered when scheduling these actions.	NA
LSN DR F-1.3 (formerly LSN 2.28.03)	Each participating entity must ensure their site availability and integrate it into the overall LSN sufficiently in advance of the specified date to meet the availability criteria. The availability of other Participant staff must be considered when scheduling these actions.	2,4
LSN DR F-1.4 (formerly LSN 2.28.04)	Each participating entity must complete site and LSN integration testing acceptable to the LSNA sufficiently in advance of the specified date to meet the availability criteria.	1,2,3,4,5
LSN DR F-1.5 (formerly LSN 1.19)	The responsible official shall make the initial certification to the Pre-License Application Presiding Officer in accordance with §2.1009.	NA
LSN DR F-1.6 (formerly LSN 1.19)	The responsible official for the DOE shall also update this certification at the time of submission of the license application.	NA
LSN DR F-1.7 (new)	Documentary material created after the initial certification of compliance must be made available reasonably contemporaneous with its creation, so Participants may have timely access to this material in order to prepare for the licensing proceedings.	NA

Functional Area 6: Docket Related Requirements

LSN Req. No.	Description	Component / CI
LSN DR G-1 (formerly LSN 1.30)	The NRC must provide an electronic docket that receives, stores, distributes, and maintains docket material no later than at the time of the docketing of the DOE license application to receive and possess high-level radioactive waste at a geologic repository operations area pursuant to 10 C.F.R. Part 60.	NA
LSN DR G-1.1 (formerly LSN 1.33)	The NRC, in its management of the electronic docket, must provide the computer system necessary to comply with service requirements.	NA
LSN DR G-1.2 (formerly LSN 1.35)	The NRC must provide an electronic docket that contains a list of all exhibits, showing where in the transcript each was marked for identification and where it was received into evidence or rejected.	NA
LSN DR G-1.3 (formerly LSN 1.40)	The NRC must identify a means by which the unavailability of the electronic docket for more than four hours in any day is communicated to the Presiding Officer so that the day is not counted in the computation of time.	NA
LSN DR G-2 (formerly LSN 1.49)	DOE must submit the license application to the docket in electronic form using the same header, text, and image standards and formats as for evidentiary collection materials	NA
LSN DR G-3 (formerly LSN 1.31)	The NRC must provide a Protective Order File.	NA
LSN DR G-4 (formerly LSN 1.32)	The NRC must deliver all pleadings, orders, and decisions per §2.1013(c) (e.g., electronically, using secured process).	NA
LSN DR G-5 (formerly LSN 1.34)	The NRC must maintain the docket.	NA
LSN DR G-5.1 (formerly LSN 1.36)	The NRC must enter hearing transcripts into the docket on a daily basis in order to provide next-day availability at the hearing.	NA
LSN DR G-6 (formerly LSN 1.37)	The NRC must establish a mechanism whereby all filings are able to be submitted/received electronically and to require a password security code for transmission of these documents.	NA
LSN DR G-7 (formerly LSN 1.38)	The NRC, in its management of the electronic docket, must provide a mechanism by which all Presiding Officer and Commission issuances and orders are transmitted electronically.	NA
LSN DR G-8 (formerly LSN 1.39)	The NRC must provide a mechanism by which the Presiding Officer and all counsel of all parties have access to the electronic docket (including Protective Order File) during the hearing.	NA

LSN Req. No.	Description	Component / CI
LSN DR G-9 (formerly LSN 1.41)	The NRC must provide a mechanism to receive electronically transmitted depositions(including questions, cross-questions, and answers) and enter them into the docket file	NA
LSN DR G-10 (formerly LSN 1.42)	The NRC, in its management of the electronic docket, must identify a means by which only a part or parts of a deposition may be offered into evidence.	NA
LSN DR G-10.1 (new)	The NRC must be able to deliver from the docket a single image from a multi-page document.	NA
LSN DR G-11 (formerly LSN 1.43)	Each LSN Participant must demonstrate substantial and timely compliance with §2.1003 in order to be granted party status.	NA
LSN DR G-12 (formerly LSN 1.44)	Absent good cause, each LSN Participant must ensure that any exhibit is available before the commencement of that portion of the hearing where it will be offered.	NA
LSN DR G-13 (formerly LSN 1.45)	For parties and interested governmental Participants, service is completed when the sender receives electronic acknowledgment ("delivery receipt") that the electronic submission has been placed in the recipient's electronic mailbox.	NA
LSN DR G-13.1 (formerly LSN 2.45.01)	All participating entities shall ensure that they can receive and reply to Internet-standard electronic mail by arbitrary Internet users using Internet standard e-mail MUAs (Mail User Agents) and MTAs (Mail Transfer Agents).	NA
LSN DR G-13.2 (formerly LSN 2.45.02)	All participating entities shall maintain an electronic log of all deliveries of LSN documents to their EIE or Internet mail server. This log will contain the IP address or DNS host name of the recipient's computer and the date and time of delivery. The log shall be in either the web standard "Common Log Format" or "Combined Log Format."	NA
LSN DR G-13.3 (formerly LSN 2.45.03)	All participating entities shall maintain an electronic log of all motions practice-related electronic mail transactions. This log will contain the IP address or DNS host name of the recipient's computer, the sender and recipient's user names, and the date and time of exchange.	NA
LSN DR G-13.4 (formerly LSN 2.45.04)	All participating entities shall maintain a log of all non-electronic LSN-related deliveries. This log will contain recipient identification, date of delivery, and method of delivery.	NA
LSN DR G-13.5 (formerly LSN 1.46)	Each LSN Participant may utilize an electronic acknowledgment ("delivery receipt") as proof of service.	NA

LSN Req. No.	Description	Component / CI
LSN DR G-14 (formerly LSN 1.47)	Deponents must submit an electronic index of all documentary material in his/her possession relevant to the subject matter of the deposition to all parties and interested government Participants, identifying which were already made available electronically, ten days before the scheduled date of the deposition.	NA
LSN DR G-14.1 (formerly LSN 1.48)	Deponents must bring to the deposition a paper copy of all documents that are included in the electronic index that the deposing party or interested governmental Participant requests that have not already been provided electronically.	NA
LSN DR G-14.2 (new)	Parties may request that any or all documents on the index that have not already been provided electronically be made electronically available by the depondent.	NA

Functional Area 7: Electronic Information Exchange (EIE) Related Capabilities

LSN Req. No.	Description	Component / CI
LSN DR H-1 (formerly LSN 1.50)	During the pre-license application phase, each LSN Participant must use the procedures specified in §2.1013(c) for service of all pleadings, answers, orders, and decisions.	NA
LSN DR H-2 (formerly LSN 1.51)	During the licensing proceeding, each LSN Participant must use the procedures specified in §2.1013(c) for service of all pleadings, answers, orders, and decisions.	NA
LSN DR H-3 (formerly LSN 1.52)	Absent good cause, parties and interested government Participants must submit all filings electronically using a secure transaction process for transmission of documents to the electronic docket.	NA
LSN DR H–4 (formerly LSN 1.53)	Parties and interested government Participants must transmit depositions to NRC's Secretary (SECY) in electronic form for entry into the docket.	NA

Appendix B. Logical to Physical Mappings

Table 3.7.2-1 contains the Entity and Attribute mapping from the Logical Design to the Tables and Columns of the LSN Physical Data Model (PDM).

Table 3.7.2-1 Entity and Attribute Mapping

Logical Design Entity and Attribute Types	Physical Design Rational Tables and Columns
Participant Organization	Participant_Organization
Name	Name
Address Line 1	Address_Line_1
Address Line 2	Address_Line_2
City	City
State	State
Zip Code	Zip_Code
Participant ID	Participant_ID
url	url
Participant	Participant
First Name	First_Name
Middle Initial	Middle_Initial
Last Name	Last_Name
Organization	Organization
Address Line 1	Address_Line_1
Address Line 2	Address_Line_2
City	City
State	State
Zip Code	Zip_Code
Phone Number	Phone_Number
FAX Number	FAX_Number
E-Mail Address	E_Mail_Address
Encrypted Password	Encrypted_Password
Participant Type	Participant_Type
POC Flag	POC_Flag
Administrator Flag	Administrator_Flag
Header	Header
LSN Accession Number	LSN_Accession_Number
Participant Accession Number	Participant_Accession_Number
Title	Title
Document Date	Document_Date
Comments	Comments
Non Digital Media	Non_Digital_Media
QA Record	QA_Record
Descriptors	Descriptors
Number of Images	<< Calculated on Demand>>
Physical Location	Physical_Location
Participant Code	Participant_Code

Logical Design Entity and Attribute Types	Physical Design Rational Tables and Columns
Addressee	Addressee
Name	Name
Organization	Organization
Audit Table	Audit_Table
Transaction Date	Transaction_Date
Transaction Type	Transaction _Type
Item_Type	Item_Type
Author	Author
Name	Name
Organization	Organization
Document Number	Document_Number
Document Number	Document_Number
Document Type	Document_Type
Type	Type
Access Control	Access_Control
Control Info	Control_Info
Related Records	Related Records
Code	Code
Document Type	Document_Type
Document Type	Document_Type
Packages	Packages
Package ID	Package_ID
Traceability	Traceability
Code and Number	Code_And_Number
Number	Number
Versions	Versions
Version	Version
Web Location	Web_Location
Image URL	Image_URL

APPENDIX C. CLASS AND OBJECT MODEL TAILORING

UML is designed to be an extensible modeling language. This section documents the extensions used in the LSN class and object diagrams. These extensions include a consistent set of relationship stereotypes, class stereotypes and class naming conventions for modeling the COTS products and web pages used in LSN. These extensions provide the reader with additional, useful information about the implementation of the system:

- *Class stereotypes* are shown in guillemets (double angle brackets, such as *«some stereotype»*) at the top of the class box.

- *Class names* are shown next in the class box. In the text, class names are *italicized*. (They are not shown in Italics in the figures, however, because Rational Rose modeling tool does not support this.)

- *Relationship stereotypes* are shown in guillemets on the lines connecting the class boxes.

See Appendix C, Figure 3.4.2-1, Administer Participants Class Diagram, for an example. These stereotypes and naming conventions are described in detail below.

C.1   Modeling Autonomy and Other COTS Products as Class Utilities

COTS products such as Autonomy, WebTrends and What's Up are pictured as "class utilities", with a shadow on the lower right part of the class box. These are not classes in the traditional sense, but are modeled as such in LSN for completeness. The product name is used as part of the class name. For example:

- *Autonomy Fetch* as a class name

- *What's Up* as a class name

Each is labeled with an identifying class stereotype:

- *«COTS Product»*

C.2   Modeling Autonomy and SQL Server Stores as Class Utilities

Generally, data stores are never directly manipulated by LSN custom code, and are, in theory, immaterial to the design. However, to indicate where data is ultimately stored, these data stores are modeled here with stereotypes.

- *«System Product»* for SQL Server

- *«COTS Product»* for others

The class name describes the type of data:

- *Autonomy Production Data*

- *SQL Header Audit Data*

C.3   Modeling Pre-Built Components

The LSN code will use vendor-supplied components that indirectly access these data stores. These are pre-written objects that are provided by Autonomy or Microsoft and used by the VB or VB Script code to read and write to the data stores. These components use the following stereotypes:

- *«COTS Component»* for Autonomy Supplied components

- *«System Component»* for SQL Server supplied ADO components. IIS, the Microsoft Web Server also supplies pre-built ASP (Active Server Page) components used in the code.

The class name indicates the provider of the component. For example:

- *Autonomy Agent*

- *ADO Header Audit Data*

- *ASP Session*

C.4   Modeling Web Pages and Other HTML Elements as Classes

Because they contain both code ("methods") and data ("state") and have a unique identity, we model web pages and other web elements as if they were classes. The stereotypes used are as follows:

«Server Page» is an ASP page that runs on the server. Typically the ASP page builds or generates the HTML that makes up a client page.

- *«Client Page»* is either a general purpose client-side HTML page, or an ASP-generated, customized client page.

- *«HTML Form»* is an HTML form on a web page.

C.5   Class Relationships Used in LSN

In an object-oriented design, classes have different types of relationships that exist in the code and are modeled in the diagrams on the lines connecting classes. See Figure 3.4.2-1, Administer Participants Class Diagram, for an example. The LSN class model use the following subset of the typical UML relationships, presented here from strongest to weakest:

- *Aggregation* is a whole-part or containment or "has a" relationship between one class and its parts. It is pictured as a line with a diamond at the owner end. For example, the *Participant List* ASP web page has a set of *ADO Participant Data* components and the *Find Participant* HTML form physically embedded as parts of the page.

- *Association* is a relationship between coequal classes that use each other's services. It is pictured as a solid line connecting both classes. If the association is one way, an arrow points in the navigable direction. For example, the *Participant List* page is associated with the *Add/Change/Delete Participant* page.

- *Dependency* is a "using" relationship pictured with a dashed line and an arrow. The ADO Participant Data components uses the SQL database, for example.

Generalization or inheritance is not used in the LSN class model.

C.6 Modeling Relationships between Web Pages

The LSN design also extends the class relationships (the lines that connect the class boxes) with the following stereotypes. These are customized for modeling web-based applications:

- *«Link»* denotes a page that has a hypertext link association to another page. For example the *Participant List* Page has a link to the *Add/Change/Delete Participant* page, expressed in HTML as: …

- *«Targeted Link»* is also a page with a hypertext link to another page, but the second page opens in a different frame or new window. This would be used in LSN, for example, when a user clicks on the LSN search page to view a document on a Participant site. The document would open in a separate window so that the search list is still available to the user. In HTML: <a *target*="_blank" href= "*http://www.doe.gov/some doc.htm*"> …

- *«Submit»* is a relationship between an HTML Form and a Page. The form submits its data to a page that is then activated. For example: <form name="FindParticipant" action="*ParticipantList.ASP*"> … </form>

- «Redirect» is a relationship where one page automatically runs another page without the user clicking on it. For example, non-administrators will be automatically sent away from all administrative pages. In the code, this is done using a built-in ASP object: If Session("Admin") = "NO" Then Response.Redirect = "LSNHome.ASP"

- *«Include»* is a dependency relationship where a page includes a fragment of another file physically on it. For example, most LSN pages will include a common navigation bar built once and used everywhere. In ASP code: <!--#include file="NavigationBar.ASP"-->

- *«Builds»* is the relationship between an ASP page and the HTML page it generates. For example, an ASP page may run an SQL query, wrap HTML tags around the resulting values in the Recordset and return it to the user's browser. The resulting HTML page is built by the ASP page.

These relationship stereotypes are focused on the web-based aspects of the system. In addition to a stereotype, sometimes a relationship role name is also shown on the line. The reader, for example, can follow all the link stereotypes to see the key hypertext links modeled here.