Database Archiving
Managing Data for Long Term Retention

Craig S. Mullins
craig.mullins@neonesoft.com
Authors

This presentation was prepared by:

Craig S. Mullins
Corporate Technologist

NEON Enterprise Software, Inc.
14100 Southwest Freeway
Sugar Land, TX  77479
Tel: 888.338.6366
Fax: 281.207.4973
E-mail: craig.mullins@neonesoft.com

This document is protected under the copyright laws of the United States and other countries as an unpublished work. This document contains information that is proprietary and confidential to NEON Enterprise Software, which shall not be disclosed outside or duplicated, used, or disclosed in whole or in part for any purpose other than to evaluate NEON Enterprise Software products. Any use or disclosure in whole or in part of this information without the express written permission of NEON Enterprise Software is prohibited.

© 2006 NEON Enterprise Software (Unpublished). All rights reserved.
Agenda

Emergence of Data Management Functions

The Long Term Data Storage Problem

Long Term Data Storage Solutions

Database Archiving Capabilities
Difference between DBA and DM

**Database Administration**
- Backup/Recovery
- Disaster Recovery
- Reorganization
- Performance Monitoring
- Application Call Level Tuning
- Data Structure Tuning
- Capacity Planning

Managing the database environment

**Data Management**
- Database Security
- Data Privacy Protection
- Data Quality Improvement
- Data Quality Monitoring
- Database Archiving
- Data Extraction
- Metadata Management

Managing the content and uses of data
Data Management Functions

- **Database Security**
  - Authorization Auditing
  - Access Auditing
  - Intrusion Detection
  - Replication Auditing

- **Data Quality**
  - Data Profiling
  - Data Quality Assessment
  - Data Cleansing
  - Data Quality Filtering
  - Data Profile Monitoring

- **Data Archiving**
  - Short term Reference Database
  - Long Term Database Archiving

- **Data Extraction**
  - Maintain privacy
  - Maintain Security

- **Metadata Management**
  - Complete Encapsulation
  - Change History Auditing
Database Administration Functions

- Very well defined tasks
- Very well defined Job Title and Description
- Overwhelming vendor support
- DBMS architectures fully supportive
- Functions fall entirely in IT
- Must be done well to support efficient operational environment
Data Management Functions

- Tasks definitions are emerging
- No standard Job Titles or Descriptions
- More aligned with business units than IT
- IT management has not been supportive (NMP)
- Executive management has not been supportive
- DBMS architectures built without consideration of DM
- Little Vendor Support
- Companies have accrued many penalties for not paying attention to DM requirements
Emerging Data Management Drivers

Recent Regulations:
- Corporate Governance
- Data Privacy
- Data Retention
- Data Accuracy

Increasing Data Quality Costs

Increasing Data Volumes

Increasing uses/users of data

More Emphasis and Spending on Data Management Functions

Significant Tangible Benefits
Drivers Impacts on Functions

- Compliance
- Quality Costs
- Expanded Uses
- Increased Volumes

- Database Security
- Data Quality
- Data Archiving
- Data Extraction
- Metadata Management
Long Term Database Archiving
Trends Impacting Archive Needs

Data Retention Issues:
- Volume of data
- Length of retention requirement
- Varied types of data
- Security issues
Archiving All Types of Data

Physical Documents
- Paper
- Blueprints
- Forms
- Claims

Electronic Documents
- Word
- Excel
- PDF
- XML

Database Data
- IMS
- DB2
- ORACLE
- SYBASE
- SQL Server
- IDMS

Electronic Files
- VSAM
- Programs
- UNIX Files

Email
- Outlook
- Lotus Notes
- Attachments

Multimedia Files
- Sound
- Pictures
- Video

INTELLIGENCE. INNOVATION. INTEGRITY
Database Archiving:
The process of removing selected data records from operational databases that are not expected to be referenced again and storing them in an archive data store where they can be retrieved if needed.
Data Archive and ILM

- **Operational**: Needed for completing business transactions
- **Reference**: Needed for reporting or expected queries
- **Archive**: Needed for compliance and business protection

Create → Reference → Archive → Discard

Mandatory Retention Period
Data Retention: Database Archiving

Data Retention Requirements refer to the length of time you need to keep data:

Determined by laws: external regulations
Determined by business needs: internal needs for analytic applications

We need to keep more data: a lot more data (125% CAGR)
For more years: a lot more years
We need to preserve original content and meaning

Old retention period → New retention period
### Some Sample Regulations Impacting Data Retention

<table>
<thead>
<tr>
<th>21 CFR Part 11 (Life Sciences)</th>
<th>Clinical trials and FDA approval: 35 years</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Food—manufacturing/processing/packing: No less than 2 years after commercial release</td>
</tr>
<tr>
<td></td>
<td>Drugs—manufacturing/processing/packing: No less than 3 years after commercial release</td>
</tr>
<tr>
<td></td>
<td>Manufacturing of biological products: 5 years after end of manufacturing</td>
</tr>
<tr>
<td>HIPAA (Healthcare)</td>
<td>Pediatric medical records: 21 years</td>
</tr>
<tr>
<td></td>
<td>Adult medical records: Up to 2 years after patient’s death</td>
</tr>
<tr>
<td></td>
<td>Documentation related to security rule implementation: 6 years from date of creation</td>
</tr>
<tr>
<td>Sarbanes-Oxley (All Public Companies)</td>
<td>All records related to audit or review: 7 years after the conclusion of audit/review*</td>
</tr>
<tr>
<td>SEC 17a-4 (Financial Services)</td>
<td>All account records: 6 years after closing account</td>
</tr>
<tr>
<td></td>
<td>Financial statements, transaction records, and communications: 3 years, first 2 easily accessible</td>
</tr>
<tr>
<td></td>
<td>Member registration and other corporate documentation: Life of the enterprise</td>
</tr>
</tbody>
</table>

*SOX states a 5 year retention period, but the SEC implementation ruling of 7 years supersedes the SOX mandate*
E-Discovery

Electronic evidence is the predominant form of discovery today.
(Gartner, Research Note G00136366)

Electronic evidence could encompass anything that is stored anywhere.
(Gartner, Research Note G00133224)

When data is being collected (for e-discovery) it is imperative that it is not changed in any way. Metadata must be preserved...
(Gartner, Research Note G00133224)

Gartner Strategic Planning Assumption

- Through 2007, more than half of IT organizations and in-house legal departments will lack the people and the appropriate skills to handle electronic discovery requirements (0.8 probability).
(Gartner, Research Note G00131014)
E-Discovery

Federal Rules of Civil Procedure, Rule 34b

- Took effect December 2006
- A party who produces documents for inspection shall produce them ... as they are kept in the usual course of business..."
- The amended rules state that requested information must be turned over within 120 days after a complaint has been served.

So data stored in database systems must be able to be produced in electronic form.
What Does It All Mean?

Enterprises must recognize that there is a business value in organizing their information and data.

Organizations that fail to respond run the risk of seeing more of their cases decided on questions of process rather than merit.

(Gartner, 20-April-2007, Research Note G00148170: Cost of E-Discovery Threatens to Skew Justice System)
Operational Efficiency

Database Archiving can also be undertaken to improve operational efficiency

- Large volumes of data can interfere with production operations
  - efficiency of transactions
  - efficiency of utilities: COPY, REORG, etc.
  - Storage
    » Gartner: databases copied an average of 6 times!
What Solutions Are Out There?

- **Keep Data in Operational Database**
  - Problems with authenticity of large amounts of data over long retention times

- **Store Data in UNLOAD files (or backups)**
  - Problems with schema change and reading archived data; using backups poses even more serious problems

- **Move Data to a Parallel Reference Database**
  - Combines problems of the previous two

- **Move Data to a Database Archive**
Components of a Database Archiving Solution

Metadata capture, design, maintenance
Archive data query and access
Archive administration

Data & metadata
metadata policies history
Archive data store and retrieve
Data Recall
Data Extract
Databases
What Do You Need to Support Database Archiving?

- Policy based archiving: logical selection
- Keep data for very long periods of time
- Store very large amounts of data in archive
- Maintain Archives for ever changing operational systems
- Become independent from Applications/DBMS/Systems
- Become independent from Operational Metadata
- Protect authenticity of data
- Access data when needed; as needed
- Discard data after retention period
Policy based archiving

**Why:**
- Business objects are archived, not files
- Rules for when something is ready can be complex
- Data ready to be archived is distributed over database

**Implications:**
- User must provide policies for when something is moved

**How:**
- Full metadata description of data
- Flexible specification of policy: “WHERE clause”
- Support accessing data outside archive set
For Example...

Parts Master is the *parent* table to all other tables.

### Parts Master
- Part Number
- Type
- Description
- Unit Type
- Cost
- Price
- Substitute Parts

### Order Info
- Part Number
- PO Number
- Vendor ID
- Quantity Ordered
- Unit Cost
- Date Ordered
- Date Received

### Summary by Quarter
- Part Number
- Year
- Q1 Disbursed
- Q2 Disbursed
- Q3 Disbursed
- Q4 Disbursed

### Disbursement
- Part Number
- Dept. ID
- CHIT ID
- Qty Disbursed
- Date Disbursed

---

INTELLIGENCE. INNOVATION. INTEGRITY
Keep Data for a Long Time

Why: retention requirements in decades

Implications:
- Archive will outlive applications/DBMS/systems that generated them
- Archive will outlive people who designed and managed operational systems
- Archive will outlive media we store it on

How:
- Unique data store
- Application/DBMS/system independence
- Metadata independence
- Continuous management of storage
- Continuous management of archive content
Keep Very Large Amounts of Data

■ Why:
  - Large volumes of data today
  - Increasing rates of data volume growth
  - Long retention periods

■ Implications:
  - Archive won’t fit in DBMS solutions
  - Must partition contents
  - Cannot read all of archive to satisfy queries
  - Must support management functions at partition level

■ How:
  - Unique data store
    - Supports partitioning of data
    - Unlimited number of partitions
    - Manages partitions independently
  - Indexing and scoping
Maintain Archive for Changing Operational Systems

Why:
- Metadata changes frequently
- Applications are re-engineered periodically
  - Change DBMS platform
  - Change System platform
  - Replace with new application
  - Consolidate after M&A

Implications:
- Archive must support multiple variations of an application
- Archive must deal with metadata changes

How:
- Manage applications as major archive streams having multiple minor streams with metadata differences
- Achieve independence from operating environment
Achieve Application Independence

■ Why:
  – Operational applications will not be available
  – Operational systems will not be available

■ Implications:
  – Archive must satisfy all query requirements from within
  – Archive data must include metadata needed for interpretation of data
  – Archive system will be moved to new systems from time to time

■ How:
  – Store metadata and data in archive together
  – Implement archive system on multiple systems
  – Implement archive system on new systems
Achieve Metadata Independence

Why:
- Operational metadata is inadequate
- Operational metadata changes
- Operational systems keep only the “current” metadata
- Data in archive often does not mirror data in operational structures

Implications:
- Archive must encapsulate metadata
- Metadata must be improved

How:
- Metadata Capture, Validate, Enhance capabilities
- Store structure that encapsulates with data
- Keeps multiple versions of metadata
Protect Authenticity of Data

Why:
- Potential use in lawsuits/investigations
- Potential use in business analysis

Implications:
- Protect from unwanted changes
- Show original input
- Cannot be managed in operational environment

How:
- SQL Access that does not support I/U/D
- Do not modify archive data on metadata changes
- Encryption as stored
- Checksum for detection of sabotage
- Limit access to functions
- Audit use of functions
- Maintain offsite backup copies for restore if sabotaged
Access Data Directly From Archive

Why:
- Cannot depend on application environment

Implications:
- Full access capability within archive system

How:
- Industry standard interface (e.g. JDBC)
- LOAD format output for
  - For load into a database
  - May be different from database came from
- Recall format output for
  - Showing original input bit-for-bit
- Requires full and accurate metadata
- Ability to review metadata
- Ability to function across metadata changes
Discard Function

- Why:
  - Legal exposure for data kept too long

- Implications:
  - Data cannot be kept in archive beyond retention period
  - Must be removed with no exposure to forensic software

- How:
  - Policy based discard
  - System level function
  - Tightly controlled and audited
  - True “zero out” capability
  - Discard from backups as well
So Where do we store the archive?

■ NOT a Relational Database!
  – Only supports 1 definition of data
  – Problem with very large amounts of data
  – Cannot protect from unwanted changes
  – Requires excessive administration

■ New Database Archive Structure
  – Stores data and metadata
  – Partitions data by metadata groupings
  – Unlimited number of partitions
  – Does not support INSERT/UPDATE/DELETE functions
  – Manages by partitions
  – Indexed and scoped
Summary Points

- Keeping data in operational systems is a bad idea
- Putting data in UNLOAD files is a bad idea
- Putting data in a parallel references database is a bad idea
- Using a DBMS to store the archive does not work
- Database archiving requires a great deal of data design
  - Establishing and maintaining metadata
  - Designing how data looks in the archive
  - Achieving application independence
- Database archives must be continuously managed
  - Copying data for storage problems (e.g. media rot)
  - Copying data for system changes
  - Copying data for data encoding standard changes
  - Logging, auditing, and monitoring
    - Archive events
    - Partition management
    - Accesses
- Must staff-up for database archival
Craig S. Mullins
NEON Enterprise Software

craig.mullins@neonesoft.com

www.neonesoft.com

www.craigmullins.com

www.DB2portal.com

Intelligent Solutions for Enterprise Data. Depend On It.

http://www.neonesoft.com