

# Long-Term Archival and Databases

Peter Mojica

[www.csi1000.com](http://www.csi1000.com)

# Datasets to Databases

- During the past two decades, databases have evolved from simple repositories of 'tabulated data' and number crunching machines into complex systems trying to provide:
  - full internal data life cycle management
  - Support for rich data types (incl. internal procedural code)
  - enable multi-user transaction recording
  - store and organize terabytes and peta-bytes of data efficiently
- They have become integral parts of information systems
- While discussing electronic records archival from the application and archival perspective, databases neglected severe preservation problems arising from the technical basis of modern archival and traditional electronic records management, namely preservation, authenticity and rapid retrieval.

# Importance of Preservation

- Databases often contain “high information value”, e.g. data from financial transactions or statistical surveys:
  - Data that cannot be reproduced (e.g. transient information that is re-calculated on the fly, climate and oceanographic data)
  - Data that was produced at enormous costs (e.g. from scientific experiments, complex numerical calculations)
  - Data that is an asset, producing added-value through re-use (e.g. in a data warehouse)
- Increasingly, databases are used to maintain high evidential value, too.
  - Data from RMS, DMS, E-Mail and Instant Messages
  - Back-end databases recording transactions on a website

Today's archival requirement is complex.  
Access, preservation, authenticity, rapid discovery and retrieval

- Any serious long-term preservation strategy for any kind of digital content must **guarantee continuously**:
  - **Integrity**: protected from unintended and intended harm
  - **Intelligibility**: understandable and comprehensible
  - **Authenticity**: authentication (of authorship and provenance) and reliability (of the contained evidence)
  - **Originality**: "as close to the original as possible"
  - **Accessibility**: technically readable and usable to users
- **In a database environment these are heavily competitive and conflicting goals!**
- **No other technology resist more than databases for meeting these requirements.**

# Specific Problems with databases: An Appraisal

- Since long-term preservation of databases is an extremely costly and laborious task, rigid appraisal is an inevitable prerequisite.
  - In its operational phase, a database has to be considered as an “information system” rather than a “technical artifact”.
  - The value of evidence of the data (contained in the database) is often determinable only through the system’s purpose, design, and context of usage.
- In consequence, reliable appraisal may require an extensive knowledge of information systems, various database architectures, application architecture, use of stored procedures, application development and archival technology.

# Specific Problems: Extraction

- How to determine the scope of the extraction?
  - **Spatial:** Frequently, we encounter federated databases where a single database system refers to data (e.g. time-dependent master data) from other databases through DB links.
  - **Temporal:** Most of the times, only data 'marked for deletion' in the operational database will be archived (rather than 'snapshots' of an entire database). Usually, these records refer to still active ('undeleted') records in the same database.
- How to untie the data from a specific environment?
  - Data to be archived is heavily locked up in a specific database software environment, including vendor-specific and user-defined data types, character encodings, BLOB etc

# Specific Problems: Data structures

- How to preserve the original structure of the data?
  - Large databases consist of dozens or even hundreds of linked database tables. How to preserve the referential integrity of the data (and still keep it usable)?
  - Modern databases don't delete or overwrite any data but record all modifications by using timestamps as multiple primary key components (valid-time state tables, tracking logs, backlogs). How to keep this 'mess' understandable and traceable in the archives? Or shall we preserve 'time slices'?
  - Imposed data types and check constraints, triggers, stored procedures, or logical data views reveal original operational aspects of the data and thus contribute a lot to authenticity and originality of the data concerns.

# Specific Problems: Description

- Databases usually come with very poor documentation!
- How to identify an adequate archival description?
  - Description of tables as archival entities is inadequate and misleading. Usually they are arbitrary technical entities.
  - Databases may be document-centered (i.e. referring to or containing external files) or record-centered (i.e. not referring to documents in the common sense). Mixed forms are widespread.
  - Technical data records do not correspond to records in the archival sense. The latter may span over various tables.
  - Does a subset of multi-table records belong to the same dossier (business case)? This information is often hidden in external applications (which operate on the database).
  - It is often hard to see whether a database should be adequately described as a document, a dossier, or a collection of dossiers.

# Specific Problems: Maintenance

- Data is delivered to the archives from over a dozen DBMS products and various software versions thereof.
- How do you keep archived databases readable and usable in the long term (at acceptable cost)?
  - Archiving many-specific database exports is inappropriate. It would require maintenance of many datasets in the archives and periodic short-time migrations of all archived databases as software versions get de-supported by the source vendors. This would be prohibitively expensive and laborious.
  - Simple 'flat file' exports and de-normalization approaches are error-prone and cause losses of information, authenticity, and originality. In addition, they drastically limit data usability.
  - Migrations of archival database formats to new formats of the future must be automated.

# Specific problems: Access

- Frequently, data from a database is delivered several times throughout the operational phase of the system.
  - Database structure naturally is subject to enhancements. (For example, a typical application may start with 100 data fields per record, then grow to 300 fields overtime, and end-up with over 650 data fields per record).
  - How to provide comprehensive access to an archived database that spans over many sequential accessions from an operational database with an ever-changing data structure?
- How do you provide multi-user online access to hundreds of archived databases containing terabytes or peta-bytes of data and achieve reasonable performance for search and retrieval at an acceptable cost?

# Conclusions

- Long-term databases remain unsuitable for long term archival technologies due to:
  - Lack of procedural standards to cover the entire preservation life cycle: appraisal, extraction, ingestion, description, maintenance, access, and dissemination.
  - Lack of technical solutions to support long-term integrity, intelligibility, authenticity, originality, and accessibility of data
  - Lack of integration of procedural and technical solutions into customizable frameworks to meet the needs of large as well as small archival institutions.
- ‘Long-term record archival and preservation’ is a not a competition between database technologies and archival technologies.