Main Architectures Used in ETL as a Tool, Necessary for the Integration of Data in Large Volumes - a Task in the Field of Digital Preservation of Cultural Heritage

Kamen Angelov

Institute of Mathematics and Informatics, Bulgarian Academy of Sciences
Sofia, Bulgaria
Kamen.ANGELOV@raiffeisen.bg

Abstract. The digital recording of Cultural Heritage (CH) and its metadata relates to the concept for data warehousing systems and their lifecycle. It uses models and development tools of ETL packages. The tasks used to extract metadata from raw digitized CH data often include hard computing problems, including NP-complete problems.

This paper describes the current state of digital preservation of Cultural Heritage with an accent on data integration and performance optimization of processes. We show our vision for further solutions for automated optimization of resource utilization in ETL jobs cutting development cost.

In parallel, we describe our experiment that uses quantum computation (in simulation mode) to solve hard combinatorial problems as multiple query optimization.

Keywords: Cultural Heritage, Digitization, Data Integration, Optimization, Performance, ETL, Multiple Queries, Data Warehouse, Data Flow, Data Set, Quantum, Computational, Approach, Simulation, Uncertain, Predicates, Combinatorial, NP-Complete, Graph, Hamiltonian

1 Introduction

The preservation of cultural heritage is a complex self-enriching process - today's consumption and storage of cultural heritage reflect on today's culture. We extend it by adding new digital sources, communication media, and analytics. An obvious consequence is that the digital preservation of the heritage has exponential complexity and data volume. But the preservation of cultural heritage implies that it is (and will continue to be) mass-accessible. Therefore one can expect processing will grow as fast as internet accessibility is growing. All of these are reasons to define strategic goals for development of digital preservation of cultural heritage (DPCH) systems:

- Long term lifecycle of extendable content,
- Scalable and adaptable storage
- Effective integration processes and interfaces,
- High performance query (analytical) processing.

The DCC Curation Lifecycle Model presents core digital preservation activities in wider context [1], including rich development resources, support and training features shown in figure 1.

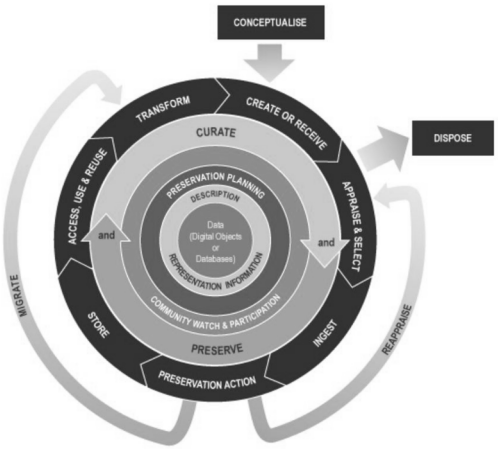

Fig. 1. DCC Curation Lifecycle Model [3]

Often lighter models applied in the archiving domain ISO 14721 (OAIS) are more applicable for low budget projects. OAIS is "a technical recommended practice for use in developing a broader consensus on what is required for an archive to provide permanent, or indefinite Long Term, preservation of digital information." [2]. The technical requirements for libraries, created in the cultural heritage domain, are covered by a simple model of 6 symmetrically related entities and 3 interfaces: Submission Information Package (SIP), to get the information from a producer; Archive Information Package (AIP), which is the information actually stored by the archive; and Dissemination Information Package (DIP), used to transfer data in response to a request by a consumer. [2].

The Metadata and the content format, used in (Archival entity) DPCH systems usually are under standard as Qualified Dublin Core [3], Digital Object Identifier [4], MARK2.The recording of new raw data and/or interchange of information between individual DPCH systems is emergent part of the DCH lifecycle.

2 Current state

The key concept used to provide extendable sets of digitally recognizable objects is metadata extraction and management. It is discussed in details in Chapter 3: Automated Metadata Extraction from Art Images [1] of Access to Digital Culture Heritage. Some metadata are native and come together with the source object, other valuable metadata might be created (extracted) inside of the DPCH system.

If the content resides in an external organization and the data are in an incompatible format, or if the required metadata properties are not trivial to extract from the source, then an interface module(s) has to be developed to deliver the correct data and the required metadata to the SAP of the DPCH system. Optimization of efforts in this task will contribute to the DPCH system as a whole. In terms of DPCH, this relates the digitalization process to it.

In low cost social projects these standards are often not strictly followed, so development of interfaces for integration of content is important for bigger DPCH centers, where DCC Curation Lifecycle Model is supported and digital preservation activities accurately follow digital preservation standards.

In conclusion, the digital preservation of cultural heritage (DPCH) systems keep an extremely huge volume of information in resistant and accessible manner, sometimes in low budget frame. Fundamental characteristics of those systems are

- Long lifecycle
- Extendable content
- Scalable technical solution

Natively, they exploit and enrich Data Warehouse and BI principles and techniques, and that is the reason to do this overview.

The Velocity™ methodology of Informatica® describes the lifecycle defining activities (competencies) in a project timeline visualized in figure 2.

Fig. 2. Data warehouse DLC [5]

Informatica® is one of the top rated data integration vendors, offering a full spectrum of data integration tools, designed for high development productivity and operational resilience.

ETL tools should be a database independent set of services, providing high availability and disaster recovery functionalities [6], [7].

Data integration products are divided by their system architectural approaches. Client utilities for all development, debugging/tracing, monitoring and management (in-

cluding security administration) activities, and support of connectivity with many databases and standards are a must.

The utilization of quantum computation is our second topic. We do not go to present basics of quantum computation here. Also, we do not discuss physical construction of quantum computers. D-Wave sells 2x1000 Q-bit models, used in NASA, Lockheed Martin, Pentagon, Microsoft, Google, achieving progress since 2005 near Moore's low [8]. Detail information about this computer is published on the company's site [9]. Many scientific results confirm adiabatic quantum annealing as Microsoft extends quantum research laboratory and currently shares Language-Integrated Quantum Operations - LIQUi [10].

In May this year, the European parliament approved The Quantum Manifesto [11] programs for strategic investments opening new horizons and challenges for information science and education.

3 Problems to be solved

There are two general approaches of data integration tools depending on where the transformation is done:

1. **Dedicated service** (ETL) - extracted data are passed to the service engine and processed there.
   This approach is applied by Informatica® PowerCenter™ and Microsoft® SSIS™ and many other vendors.
2. **Database service** (ELT) - extracted data are stored into temporary database objects, from where they are transformed using SQL DML queries. This approach is less popular and applied by Oracle® ODI™.

From a more abstract point of view, the actual difference between both approaches is in the focus on classes representing data and possible operations. In ETL the focus is on individual data row as building unit of data flow. Connection managers provide technical functionality and topology properties for transmission. The extraction process is logically covered by source components using metadata to validate input data. The processing mode is by row**.** It is possible to split (route) rows, multicast rows too many destinations. In this kind of processing data buffers are a key factor to involve high degree of parallelism. The number of destination components is not logically limited by this approach. Developer kits for extensions are usually for Java or C#.

In ELT the focus is on data sets containing data rows.

It is native for database developers preferring to analyze SQL code instead of flow diagrams and XMLs. Connectivity is covered by physical and logical topology layers linked by context. Technical functionality of extraction provide knowledge modules using templates and Jyton code (Pyton dialect of Oracle).

In fact, all data are physically written snapshot temporal tables, forming a staging area (but it is not fully equivalent to DWH staging). The data from this staging area are source for the next transformation DML queries. Working with sets is more computationally efficient for many reasons, but splitting, multicasting, and multiple destinations are not allowed at all. The degree of parallelism is manageable using database

hints only. Multi-table-insert is specific and supported for Oracle database technology only. In some scenarios ELT involves more coding and tracing efforts. Deep knowledge in database behavior is a critical prerequisite.

The first approach (data flows) is classical and understandable for junior developers. It is oriented to fast and flexible development "per purpose" of data integration and provides rich functionality, pattern driven design, metadata validation, better visual perspective of data flow and minimal coding efforts.

Two general performance cases may be pointed:

- **Network traffic** is at least doubled in ETL:
- *Source → transformer (first network traffic) source data stream, Transformer → destination (second network traffic) transformed result stream,* and also lookup data: *destination → transformer (third network traffic) lookup data stream.* The proposed solution is to use partial lookup caching, at the cost of more coding efforts and some additional limitations (varying in different products). And here an additional traffic to send each query of partial lookup data is generated *transformer → destination (fourth network traffic) lookup queries stream.*
- Also latency to deliver the data into the cache is unpredictable. So this optimization is very limited, not scalable, and efforts are not always efficient.
- **Memory pressure** may cause very bad performance results. In case, when lookup data occupies a huge amount of memory. For instance – when batch loads data from a particular book issuer, to lookup authors dimension of the target library all existing authors must be extracted from the destination and passed to the transformer.
- Again, partial lookup caching and shared in-memory caches at the cost of more coding efforts and additional limitations may help.
- Generally this approach is preferred where a server area network (SAN) is implemented over optical connections and the integration processes are often or near to real time. Usually "the transformer" is placed on a dedicated HADR server group and needs 24 to 128 GB RAM for lookup caches and 12 to 64 processor cores (data are collected by experience in several banking and manufacturing data warehouse projects). This is a very expensive solution.

All popular ETL tools are adopted to play as ELT using SQL queries instead of in-pipe components, but the development cost is extremely high. Some tools (like Informatica®) support "push down" optimization which corresponds to the generic query approach used in ELT.

The second approach (data sets) is query generic, so it is easier to deploy in different environments. Based on stored metadata, these tools generate SQL queries. The principle is "only database engine can process data". So, even if you just have to do a single variable increment you should send an SQL query to get the result. But this is not important as the network traffic is fully minimized. The extracted data travel on network only once: *source ->target*. Hardware resources (and especially memory) are dynamically shared between database processes in time without any administrative intervention. Integration solutions are predictable and scalable.

Lookups natively use indexes already existing in the database increasing performance without any special development efforts.

The performance bottlenecks relate to processing speed issues:

- **Storage system speed** – I/O operations are at least tripled on destination when ELT is used. If several servers are destinations, all of them have to play transformer role, i.e. to be ready to process data integration challenge. It may cause the storage system to become very expensive.
  The only workaround is to process integration on one server and then to send results to target databases, which is to utilize ETL approach in fact.
- **Processor performance** – a destination database server plays "transformer" role, but often data integration is near to real time process (NRT) causing concurrency with other BI queries. The solution involves additional expert development efforts, administrative actions and process scheduling.

Described problems are similar to inside optimization problems of relational database engines compiling SQL queries. In some scenarios cursor operations are much more effective than set operations, but to take this full optimization decision consumes unpredictable computational resources and time.

No one data integration product provide embedded optimization of processing at all. Actually Informatica's "push-down optimization" is static translation of part of data flow diagram to SQL query, but it is not based on evaluation and do not have optimization goals nor rules. We wish to optimize the overall processing speed, memory utilization, network traffic and IO operations based on collected statistics. This kind of dynamic optimization will help to solve described performance issues automatically or by administrators, providing better utilization of computational assets and cutting development efforts and cost.

4 Conclusions

On the base on analysis of the current state we are working on experimental design of a new data integration tool, utilizing simulation of quantum computation for optimization of transformations.

The optimization of plan involves number of complex tasks as search in graph for Hamiltonian paths. The MQO problem is discussed in details by Sellis [12]. We use ideas and principles shared by the author to realize our goals in context of ETL tools. Inspired by ideas of Koch [13], [14] for Quantum computing applications for Databases, we will use quantum computing simulation with *Language-Integrated Quantum Operations* [15] (LIQUi|>) to define uncertain selection criteria for optimization.

Let our transformation model is $F: A \rightarrow B$ described in XML tree. We decompose F into partial transformations $\{f_i\}$, where for given subsets $\{a_i \mid where\ a_i\ is$ a subtree of $A\}$, $f_i: a_i \rightarrow b_i$, and $\bigcup b_i = B$. Let equivalent pair of implementations for f_i do exist

- S_i is set based implementation of f_i and
- C_i is cursor based implementation of f_i

– $T_{is,}$ T_{ic} we create additional pre-execute procedures for distribution of tasks.

Both S_i and C_i must have relatively well predictable costs measure. The cost |.| should be additive evaluation for monitored resources. Initially the calculation of the cost will be simple metric defined by data volume of information multiplied by constant price of the resource (memory and network traffic). In further development the cost could be subject of statistical and predictive analysis.

We will use very simplified MQO model. We change only implementation of nodes, keeping all the original nodes and logical data lineage concept, but we will distribute tasks to process partial transformations between source, target and dedicated transformation servers. In contrast the MQO in database engines does full reconstruction of the execution tree selecting from many different algorithms on each node. Note that all S_i, C_i and $T_{is,}$ T_{ic} acts and act only on source subset of data a_i.

Let q_i is the selection state for f_i, equal to 1 when S_i is selected to implement f_i, and 0 when C_i is selected to implement f_i. Our goal is to find a implementation $\{q_i\}$ of F using partial implementations for nodes $\{<S_i, C_i, T_{is}, T_{ic}, a_i>, i=1..n\}$ where the total cost function

$$|F|=sum(q_i*(|S_i|+|T_{is}|) + (1-q_i)*(|C_i|+|T_{ic}|))$$

is minimized. For decomposition tree containing n nodes this drives to 2^n possible implementations to be evaluated. We will demonstrate how the quantum computation reduces this complexity $O(2^n)$ to polynomial time $O(n^3)$.

For example in DCH system with 100 dimensions, our optimization task will have 2^{100} combinations for evaluation, but the quantum superposition solves the problem in polynomial time.

Expected results:

1. We will test new ETL tool design enabled for dynamic optimization applicable for data integration of digital preservation of cultural heritage.
2. The ability of quantum computers to solve hard combinatorial problems will help in extraction of metadata from digital content.
3. Demonstration of quantum computational approach (in simulation mode) to solve NP-complete tasks. We believe that quantum computation will take central place in future metadata extraction methods used in DCH too.
4. We hope this will get attention for adequate research investments in quantum computing field and quantum technologies in Bulgaria.

References

1. Access to digital culture heritage, Krassimira Ivanova, Milena Dobreva, Peter Stanchev, George Totkov, Plovdiv University, 2012
2. http://www.oclc.org/research/publications/library/2000/lavoie-oais.html
3. http://www.dcc.ac.uk
4. https://www.doi.org/
5. https://www.informatica.com/content/dam/informatica-com/global/amer/us/collateral/data-sheet/velocity_data-sheet_6091.pdf
6. James Connolly, Data warehouses: Tips for building a disaster recovery plan (http://searchcio.techtarget.com/tip/Data-warehouses-Tips-for-building-a-disaster-recovery-plan)

7. T.T.Lwin, T.Thein, High Availability Cluster System for Local Disaster Recovery with Markov Modeling Approach, IJCSI International Journal of Computer Science Issues, Vol.6, No.2, 2009, ISSN(online) 1694-0784
8. http://www.recode.net/2014/9/25/11631266/d-wave-ceo-our-next-quantum-processor-will-make-computer-science
9. http://www.dwavesys.com/resources/publications
10. https://www.microsoft.com/en-us/research/project/language-integrated-quantum-operations-liqui/
11. http://qurope.eu/manifesto/
12. Sellis, Multiple Query Optimization. TODS, 1988
13. Immanuel Trummer, Christoph Koch, MQO on the D-Wave 2x Adiabatic Quantum Computer, arXiv:1510.06437vl cs.DB
14. http://www.cs.cornell.edu/~sudip/quantumdb.pdf
15. Public Product information, Informatica®, https://www.informatica.com/content/dam/informatica-com/global/amer/us/collateral/data-sheet/velocity_data-sheet_6091