Chapter 3
Enterprise Application Characteristics

3.1 Enterprise Data Sources

An enterprise data management system should be able to handle data coming from several different data sources. In the ecosystem of modern enterprises, many applications work on and produce structured data. Enterprise Resource Planning (ERP) systems, for example, typically create transactional data to capture the operations of a business. More and more event and stream data is created by modern manufacturing machines and sensors. At the same time, large amounts of unstructured data is captured from the web, social networks, log files, support systems, and others.

Business users need to query these different data sources as fast as possible to derive business value from the data or coordinate the operations of the enterprise. Real-time analytics applications may work on any of the data sources and combine them. They analyze the structured data of ERP systems for real-time transactional reporting, classical analytics, planning, and simulation. The data from other data sources can be taken into account as well, for example, a text analytics application can combine a customer sentiment analysis on social network data with sales numbers or a production planning application takes sensor data of the RFID sensors into account.

3.2 OLTP vs. OLAP

An enterprise data management system should be able to handle transactional and analytical query types, which differ in several dimensions. Typical queries for Online Transaction Processing (OLTP) can be the creation of sales orders, invoices, accounting data, the display of a sales order for a single customer, or the display of customer master data. Online Analytical Processing (OLAP) is used to summarize data, often for management reports. Typical analytical reports are for example the

H. Plattner, *A Course in In-Memory Data Management*,
DOI 10.1007/978-3-642-55270-0_3, © Springer-Verlag Berlin Heidelberg 2014

sales figures aggregated and grouped by regions, different timeframes and products or the calculation of Key Performance Indicators (KPIs).

Because it has always been considered that these query types are significantly different, the data management systems were split into two separate systems to tune the data storage and schemas accordingly. In the literature, it is claimed that OLTP workloads are write-intensive, whereas OLAP-workloads are read-mostly and that the two workloads rely on “Opposing Laws of Database Physics” [Fre95].

Yet, data and query analysis of current enterprise systems showed that this statement is not true [KGZP10, KKG+11]. The main difference between systems that handle these query types is that OLTP systems handle more queries with a single select or queries that are highly selective returning only a few tuples, whereas OLAP systems calculate aggregations for only a few columns of a table, but for a large number of tuples.

For the synchronization of the analytical system with the transactional system(s), a cost-intensive ETL (Extract-Transform-Load) process is required. The ETL process introduces a delay and is relatively complex, because all relevant changes have to be extracted from the outside source or sources if there are several, data is transformed to fit analytical needs, and it is loaded into the target database.

3.3 Drawbacks of the Separation of OLAP from OLTP

While the separation of the database into two systems allows for workload specific optimizations in both systems, it also has a number of drawbacks:

- The OLAP system does not have the latest data, because the ETL process introduces a delay. The delay can range from minutes to hours, or even days. Consequently, many decisions have to rely on stale data instead of using the latest information.
- To achieve acceptable performance, OLAP systems work with predefined, materialized aggregates which reduce the query flexibility of the user.
- Data redundancy is high. Similar information is stored in both systems, just differently optimized.
- The schemas of the OLTP and OLAP systems are different, which introduces complexity for applications using both of them and for the ETL process synchronizing data between the systems.

3.4 The OLTP vs. OLAP Access Pattern Myth

Transactional processing is often considered to have an equal share of read and write operations, while analytical processing is dominated by large reads and range queries. However, a workload analysis of multiple real customer systems reveals that OLTP and OLAP systems are not as different as expected in classic enterprise

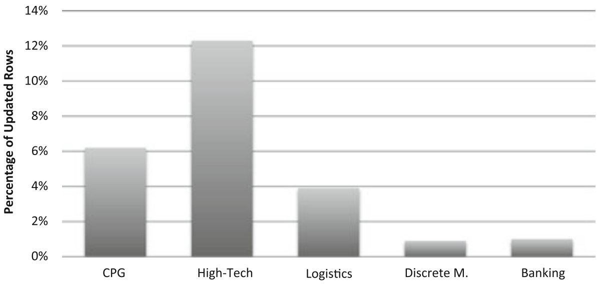

Fig. 3.1 Updates of a financial application in different industries

systems. As shown in [KKG+11], even OLTP systems process over 80 % of read queries, and only a fraction of the workload contains write queries. Less than 10 % of the actual workload are queries that modify existing data, e.g. updates and deletes. OLAP systems process an even larger amount of read queries, which make up about 95 % of the workload.

The updates in the transactional workload are of particular interest. An analysis of the updates in different industries is shown in Fig. 3.1. It confirms that the number of updates in OLTP systems is quite low [KKG+11], and varies between industries. In the analyzed high-tech companies, the update rate peaks at about 12 %, meaning that about 88 % of all tuples saved in the transactional database are never updated. In other sectors, research showed even lower update rates, e.g., less than 1 % in banking and discrete manufacturing [KKG+11].

These results lead to the assumption that updates and deletes can be implemented by inserting tuples with timestamps to record the period of their validity.

The additional benefit of this so called insert-only approach is that the complete transactional data history and a tuple's life cycle are saved in the database automatically, avoiding the need for log-keeping in the application for auditing reasons. More details about the insert-only approach will be provided in Chap. 26.

We can conclude that the characteristics of the two workload categories are not that different after all, which leads to the vision of reuniting the two systems and to combine OLTP and OLAP data in one system.

3.5 Combining OLTP and OLAP Data

The main benefit of the combination is that both, transactional and analytical queries can be executed on the same database using the same set of data as a "single source of truth". Thereby, the costly ETL process becomes obsolete and all queries are performed against the latest version of the data.

In this book we show that with the use of modern hardware we can eliminate the need for pre-computed aggregates and materialized views. Data aggregation can be executed on-demand and analytical views can be provided without delay. With the expected response time of analytical queries below 1 s, it is possible to perform analytical query processing on the transactional data directly, anytime and from any device. By dropping the pre-computation of aggregates and materialization of views, applications and data structures can be simplified. The management of aggregates and views (building, maintaining, and storing them) is not necessary any longer.

The resulting mixed workload combines the characteristics of OLAP and OLTP workloads. A part of the queries in the workload performs typical transactional request like the selection of a few, complete rows. Others aggregate large amounts of the transactional data to generate real-time analytical reports on the latest data. Especially applications that inherently use access patterns from both workload groups and need access to the up-to-date data benefit greatly from fast access to large amounts of transactional data, e.g. dunning or planning applications.

More application examples are given in Chap. 35.

3.6 Enterprise Data is Sparse Data

By analyzing enterprise data in standard software, special data characteristics were identified. Most interestingly, most tables are very wide and contain hundreds of columns. However, many attributes of such table are not used at all: 55 % of all columns are unused on average per company. This is due to the fact, that standard software needs to support many workflows in different industries and countries, however a single company never uses all of them. Further, in many columns NULL or default values are dominant, so the entropy (information containment) of these columns is very low (near zero).

But even the columns that are used by a specific company often have a low cardinality of values, i.e., there are very few distinct values. Often due to the fact that the data models the real world, and every company has only a limited number of products that can be sold, to a limited number of customers, by a limited number of employees and so on.

These characteristics facilitate the efficient use of compression techniques that we will introduce in Chap. 7, leading to lower memory consumption and better query performance.

3.7 Self Test Questions

1. **OLTP OLAP Separation Reasons**
   Why was OLAP separated from OLTP?

   (a) Due to performance problems
   (b) For archiving reasons; OLAP is more suitable for tape-archiving
   (c) Out of security concerns
   (d) Because some customers only wanted either OLTP or OLAP and did not want to pay for both

References

[Fre95] C.D. French, “One size fits all” database architectures do not work for DSS. SIGMOD Rec. **24**(2), 449–450 (1995)

[KGZP10] J. Krueger, M. Grund, A. Zeier, H. Plattner, Enterprise application-specific data management, in *EDOC*, pp. 131–140, 2010

[KKG+11] J. Krueger, C. Kim, M. Grund, N. Satish, D. Schwalb, J. Chhugani, H. Plattner, P. Dubey, A. Zeier, Fast updates on read-optimized databases using multi-core CPUs. Proc. VLDB **5**(1), 61–72 (2011)