

Introduction

Consistent addressing business needs and requirements have driven the development of the current Operational Data Store (ODS), a data structure that is designed specifically around a set of data—all integrated for a specific purpose. The ODS typically consists of current data, which is dynamically updated in real time manner (or almost near-to-real-time manner).

To cut the long story short, the ODS is an architectural construction that contains subject oriented volatile, integrated and current (or near-current), detailed operational information. One of the main confusion every organization faces is during when they set out to deploy an ODS—this question hence arises: should they use the existing DW system in order to store the real time updates for the ODS?

While doing this might work in some situations, however, most clients find conflicting Service Level Agreements (SLA's) to make it more attractive having the ODS and data warehouse (DW) operating on different schemas—even in totally split environments.

Some examples of conflicting SLA's require the need to have data in the ODS faster than it could be transformed into standardized DW format. This, as well as other undetermined business concerns and priorities between ODS use and the demands that are places, between major conflicting requirements and also anticipation of how these requirements are to be taken.

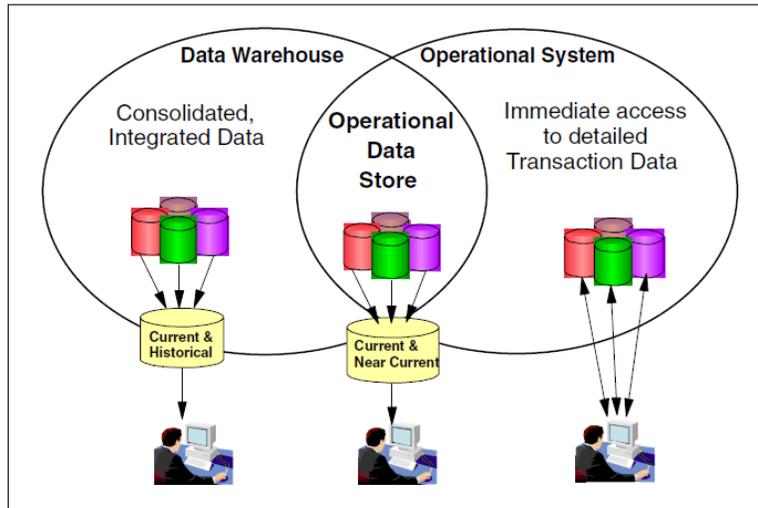
The growth in database sizes

It's a widely recognized fact that database sizes have been growing significantly. This growth is forced by multiple factors, namely companies who require more data access for longer time or increasing data being amount being digitized for storage purposes. The extent of data explosion that's occurred has provided a momentum for business intelligence application too. This intelligence application gathers, as well as store data for evaluating historical, current and prognostic views of business operations. This gathering/storing of data is done by a data warehouse, a data mart or an ODS (Operational Data Store).

Nowadays, the success of any business is heavily influenced by business intelligence applications for better decision making that ultimately rely on data warehouses or ODS. It's therefore essential for every organization to design a highly available data warehouse, which always provides consistent data.

What is ODS (Operational Data Store)?

An ODS is a setting where data from various operational databases is integrated. The main purpose of ODS is to provide the end user community with an integrated outlook of enterprise data. It enables users to address their operational challenges that span over more than one business functions. The principal key differentiators of the ODS are update frequency and direct update paths from applications, as compared to controlled updated paths of data warehouse or datamarts.



Have a look at the following characteristics that relate to an ODS.

- **Subject orientation**—ODS is designed to not only fulfill requirements of a major corporation subject, but also for specific functionality or applications.
- **Integration**—The legacy systems push detailed data through process of transformations in the ODS. This leads towards an integrated understanding of data, that is also corporate-wide.
- **Current data delivery**—Data in the ODS constantly keeps updating. The changing in data requires a higher frequency update. All changes made to the underlying system legacies must reach the ODS as soon as needed. This is so, because the velocity is important. Some data is needed immediately; however other can come in on a daily update.

The Types of ODS

An ODS has three types, type A, B and C.

ODS Type A

The ODS type A has real-time legacy data access, along with localized updates (data modifications aren't fed back to legacy systems). These localized updates typically include new data, not the currently captured in the operational system.

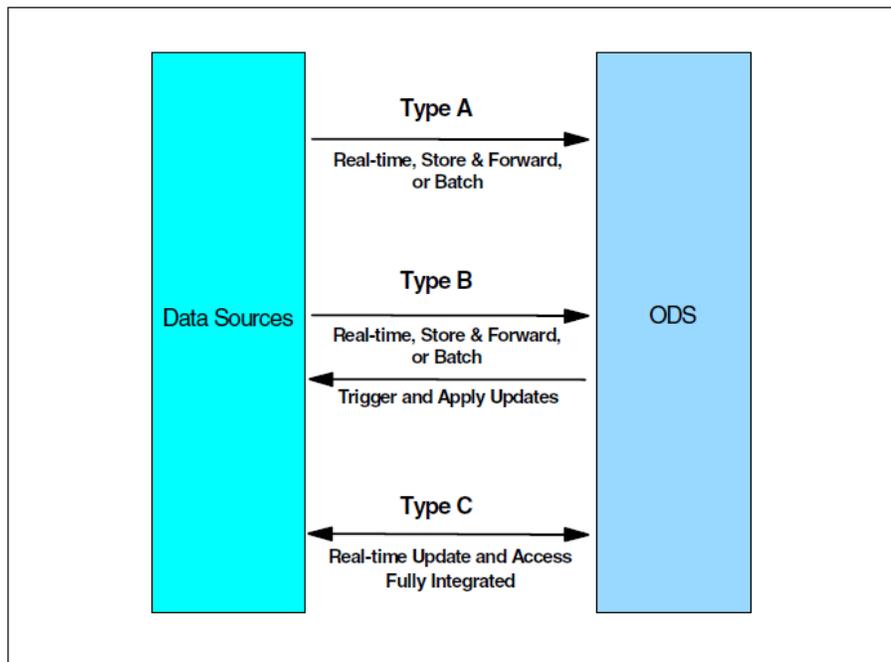
ODS Type B

An ODS type B has all characteristics of the ODS type A, but with a trigger and apply mechanism that feeds data back to operational systems. Normally, these feedback requirements are very specific to have minimized conflicts.

ODS Type C

Type C ODS is either fully integrated with legacy applications or uses real time updating and accessing.

Observe the following table for cross referencing the characteristics of each type of ODS.



Choosing the right technology stack

Exadata

The Oracle Exadata Database machine is one which is easy to deploy, it comes with an out-of-the-box solution for hosting Oracle Database for all applications at the same time of delivering high levels of available performances. The Oracle Exadata Database machine is made up of database servers, Oracle Exadata Storage Servers (Exadata), and InfiniBand fabric for networking storage and all other components needed for hosting an Oracle Database.

Oracle Exadata Storage server is a storage product, which is optimized for usage with Oracle Database applications. It's the building block of Oracle Exadata Database Machine. Exadata is able to deliver an outstanding I/O and SQL processing performance for OLTO (Online Transaction Processing), data warehousing and the consolidation of miscellaneous workloads.

With Exadata, extreme performances are delivered for all kinds of database applications by leveraging massively parallel grid architectures using Oracle RAC (Oracle Real Application Clusters), high speed infiniband connectivity, intelligent storage, Smart Flash Cache, and Hybrid Columnar Compression technology.

Golden Gate

GoldenGate replicates a defined set of new and changed transactions from the source system in the ODS. Conceptually, some of the main components in GoldenGate's decoupled architecture are Transmit, Capture, and Apply. The deploying of GoldenGate's replication involves the setting up of three main processes, which are called Extract, Pump and Replicate. Extract pulls changed data from GSI redo log into transitional "trail" files. The extract setting involves defining subsets of changed data that are to be

replicated. Pump processes extracted trail files and transfers the contents over to the ODS environment in the real time. Finally, replicate takes this trail file and loads its content to the ODS in real time. And finally, Replicate gets the file contents and loads them in the ODS database. The setup includes an sanitation phase to create a one-time primary copy of the source table at the ODS. This can normally be achieved using GoldenGate's initial load procedure or other such physical backup mechanisms.

To straighten out any twists in this whole replication process, Oracle IT completed the setup of these processes and began replicating data to ODS three months ahead of going live. This allowed the fine tuning of replication settings and patched any bugs before the functional go live for all business users.

Although Oracle IT wasn't the first customer to have deployed GoldenGate against a large scale E-business suite environment, they were the first among all the early birds who used this case, hence making them very conservative in the timeline. After ironing out any difficulties that a customer might encounter, they believe a shorter period of advanced replication is sufficient for most deployments. Therefore, Oracle IT is a strong advisor of the best practice of separating technical go live from business go live. This usually enables project team to work out various issues with their data systems and processes, even before their users encounter them. An example of applying this approach is to begin the replication at least a couple of weeks prior to the business go live.

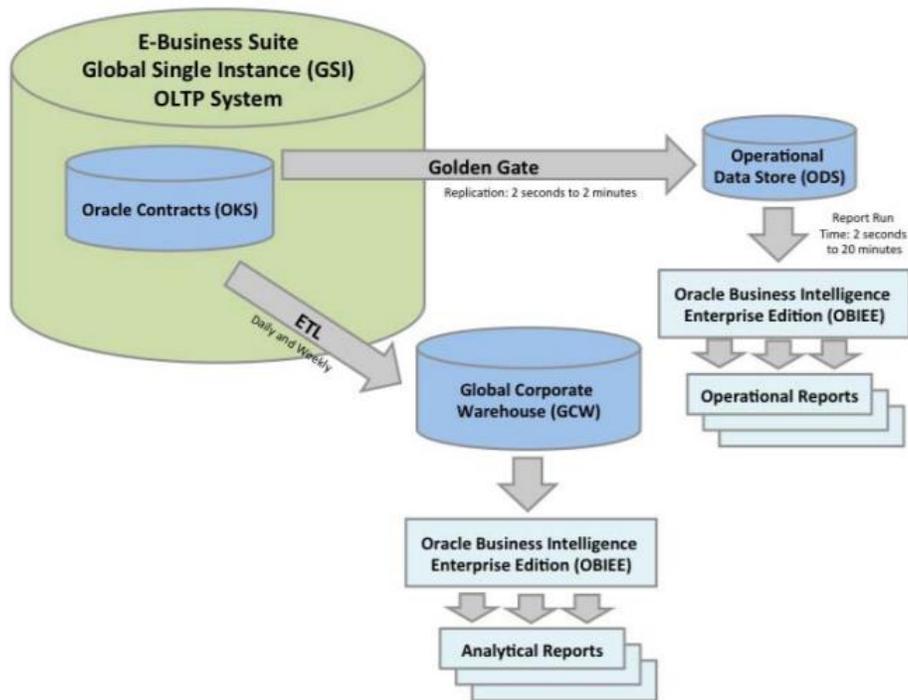
OBIEE

The operational reporting solution makes use of a dedicated instance of OBI EE in order to deliver operational reports to the ODS data. The report footprints consist of—quote inquiries, quoting, ordering, shipping, planning, order holds, service contracts, new product launches and spares management. To create this footprint report, functional “super users” were identified from different business areas. These users typically process owners for functional areas, help design data sets and construct reports based on the templates provided by IT.

Enforcement of data governance policies was also implemented by row level data security that was based on Oracle's Virtual Private Database technology. This feature restricts user access by combinations of product and geographies, and also by roles. For instance, a sales' representative from North America, who is responsible for Exadata, is only permitted to view orders inside his region for that product.

ODS ETL Architecture

A final functional architecture of this solution is shown in the figure below.



This innovative architecture uses GoldenGate for data integration into a whole new database dubbed the ODS Operational Data Store. Next, the Reports are drawn from the ODS using dedicated instance of OBI EE, Oracle Business Intelligence Enterprise Edition.

GoldenGate is real-time data integration software, which essentially replicates only changed data. Instead of doing batch ETL from the source database, GoldenGate scrutinizes database redo (transaction) logs for any change in activity. It then further replicates these changed transactions into a targeted data store. GoldenGate also includes built-in mechanisms for error detection to ensure reliable replication. Because not all transactions are relevant to particular use, GoldenGate can be configured to replicate only certain kinds of changed transactions. The replicated resulting data set can therefore be much smaller than the source's database.

Goldengate has been widely used by Oracle customers against the CRM data sources like Siebel. A small numbers of clients have also deployed GoldenGate against Oracle E-business suite. However, golden gate technology appears to be a good match for Oracle's operational reporting prerequisites. Successful deployment of GoldenGate alongside Oracles' large GSI (Global Single Instance) of e-business suite not only solves the service contracts reporting issues, but is also quickly extensible to various other subject areas.

Data Modeling

Data modeling task is one of the most difficult task in the ODS framework, because the ODS plays 2 different roles. It owns characteristics of an operational system, as well as that of DSS, decision support system. The DSS characteristics are defined through its integration and subject orientation. The operational characteristics or uniqueness is its high availability and fast response, hence the ODS can act as a base for mission critical applications.

An ODS data model could be derived from that of an enterprise data model, when one of them exists. The enterprise data model basically describes all data that is needed for corporate informational purposes. The ODS data model is a division of the enterprise data model. Only the parts which are really needed are derived for the ODS data model.

Two different approaches of data modeling exist, which fulfill the needs of the ODS.

- ERM—Entity Relationship Model
- Dimensional Modeling

ERM—Entity Relationship Model

ERM, Entity Relationship Model is an abstract tool since it's used to understand and simplify the confusing data relationships that are in the complex system environments and the business world. A data model of specific area of interest is produced by ERM modeling, with the use of two concepts.

1. **Entities**
2. **Relationships between two entities**

Entity

Entity can be defined as a person, a place, a thing or an event of interest to the organization or business. An entity is representing a class of objects, which are things in the real world that are observed and classified by their respective characteristics and properties.

Relationships

Relationship depicts structural communication and association between the entities in a model. It shows various dependencies of entities.

Most ODS's implementing the model of 3NF. This model type was originally designed to minimize the data redundancy—models like such minimize the amount of updates, which are required in a database when their values change. This ability explains its value to the OLTP databases and in updating ODS databases as well.

Dimensional modeling

Dimensional modeling is a method for conceptualizing, as well as visualizing the data models as sets of measures described by common aspects of the business. It's also especially useful for shortening and rearranging data, and in presentation of views of data to support the data analysis.

Dimensional modeling keeps a focus on numeric data, such as counts, values, balances, weights and occurrences. Basic concepts of the dimensional model are:

1. **Facts**
2. **Dimensions**
3. **Measures**

Fact

Facts are collection of related data items, which consist of measures and context data. Each fact represents any business item, business event or transaction that could be used in analyzing the business procedures or the business itself.

Dimension

A dimension is a collection of unites or members that explain the fact data from any particular view point. It's usually represented by an axis. In a dimensional model, every data point on the fact table is associated with just one member from each of the many dimensions. Dimensions determine the contextual background for facts.

Measure

Measure is numeric attribute of a fact, representing the performance or behaviors of business relative to dimensions. The actual members are known as variables. For instance, measures are the sale in money, the sale in volume or quality supplied. A measure is normally determined by combinations of the dimension members, and is located on facts.

Let's now shine some light on the best practices of running ODS on Exadata.

Best practices for running ODS on Exadata

In order to be able gain full advantages of the Exadata machine, when you are carrying forward your Exadata migration, remember to be on a lookout for some of the following enhancement related features.

1. Compression
2. Partitioning
3. Parallelism
4. Flash cache
5. Offloading/Smart scan
6. Resource management (DBRM & IORM)

For example: compression of Oracle Exadata will not only reduce your storage footprint, but will also improve your performances. Partitioning will also improve performances, along with providing you maintenance advantages and increased availabilities. Parallel execution will help you in performance, as

same is applied for properly caching tables by Exadata Flash Cache. Even though offloading and smart scan are enabled by default, but you have to make sure that they are happening. The Exadata Machine does a commendable job in managing the resources on its own. However, if you are planning to use Exadata as a consolidated platform, you can implement resource management through DBRM and IORM.

Performance Monitoring Of ODS on Exadata

Performance monitoring for ODS is just as essential for any other Oracle applications, but there is one additional complexity to overcome in the monitoring of the ODS. You will have to monitor all the replication processes to confirm that everything is working just as registered and subscribed.

It's most probable that you will establish a procedure for replication audit/control purpose by using record counts and time stamps. This could still be an initial challenge for you to monitor your replication procedures for ensuring that all is well. This is only one factor of ODS monitoring, and most surely there exist more.

Owing to the dynamism of the ODS environment, ODS easily gets out of balance without any constant adjustments through continual monitoring. The ODS environment is always changing—the data, data size, dormant data, number of the ODS users, any deletion or additions to sources, even the changes in the system records. Each of these changes might require adjustments to be made in the ODS. However, most adjustments that are necessary should be brought forward in a systematic way to the system management, triggered by a constant monitoring for responding to any changes that are made, such as listed below:

- Hardware utilization
- CPU utilization by monthly, weekly, daily or hourly basis
- Disk space
- Memory or buffer utilization
- Network
- Throughput monitoring
- Response time
- Granularity of data and the transactions
- Peak load
- Balances between batch & online processing
- Profiling workloads
- Scheduling and distribution of workloads
- External activity monitoring
- User's activities profiling
- End user query patrols

Building An Operational Data Store (ODS) on Exadata

- Resource usage patterns
- Change requirements
- Problem analysis
- Environmental changes

The end users will be accessing the information in the ODS in two general categories—static apps and dynamic reports. Static applications are usually known quantities from the performance view point. They will typically include operational like transactions, which read and update ODS or applications that produce some sorts of tactical operational report. In this method, typical operational performance monitoring & tuning techniques are applied. However, other trait of the ODS is that it is a DSS system. This typically suggests that users are dynamically producing reports for analytical purposes. Users in this category can also simply create a report that can scan a 100,000,000 table row and bring your ODS to a crushing halt.

Conclusion

Real time reapplication to an operational data store by means of GoldenGate has proven to be winning or Oracle's service contracts reporting—both, with gains in data freshness and report performances in thousands of percent's.