Enabling Pervasive BI through a Practical Data Warehouse Reference Architecture

An Oracle White Paper
February 2010

NOTE:

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Enabling Pervasive BI through a Practical Data Warehouse Reference Architecture

INTRODUCTION

There is little doubt that 'information' is at the heart of every successful, profitable and transparent business in the world today. Business leaders, industry pundits and academia all agree that information is key to enabling competitive advantage and stakeholder value through Information Technology. However, to achieve this status, it must be delivered pervasively within the business as well as to the wider trading community and customer base, together with the required levels of management excellence to complete the transformation.

Many things have changed in the world of business and IT since Data Warehouses were first popularized. Many studies have suggested that by far the majority of IT spend is on simply running the business rather than changing the business: just keeping the lights on rather than delivering competitive advantage. Within the context of Business Intelligence and Data Warehousing, this is often seen in IT groups spending too much time in ad-hoc integration and data re-engineering, and Business Analysts spending their time collecting and preparing data rather than analyzing it and taking action based on results. To really change the business, Data Warehouses must deliver '*insight*' into the heart of the business process through web services, and the analysis process through Business Intelligence tools.

If we are to truly deliver Business Intelligence pervasively throughout a business and beyond then the architectures on which it is built must be capable of loading and querying data in near real-time with the same level of accuracy and high availability as it is expected from every other operational system. With ever growing data volumes and deeper analysis requirements, it also follows that they must be able to scale out to meet future demands and manage the information lifecycle to reduce costs in a realistic and secure platform.

This white paper discusses the background behind Data Warehousing and reviews some of the dynamics and business trends driving the need for a 'Next Generation' Data Warehousing and Business Intelligence platform. The white paper will present such an architecture, in the form of an Enterprise Reference Architecture, that can be used either to guide new implementation or to develop a roadmap for an existing Data Warehouse to include some or all of the design concepts outlined in this white paper.

BACKGROUND

In this section, we will review some Data Warehousing background and look at the new demands that are increasingly being placed on Data Warehouse and Business Intelligence solutions by businesses across all industries, throughout the world.

Modeling approach

Data Warehouses are not new. First emerging as a solution in the late 1980's they have rightly earned their place in contemporary IT architectures. By their very nature Data Warehouses are highly strategic, both because of the needs and demands placed on them from the business community as well as the capital/operational expense they consume and the complexity of the solution.

In the past, Data Warehouses had to do three things in order to be successful. They had to load data, manage the data for the long term, and to provide access to the data, typically to a limited number of analysts and senior business users.

Two schools of thought prevailed at the time regarding the best approach to the underlying data model used for the Warehouse. One, popularized by Ralph Kimball, was the dimensional approach that simplified the data model to facilitate access to the data for querying by end users using stars or snowflakes. Very much a physical model, drill paths, hierarchy and query profile are embedded in the data model itself rather than the data, and in part at least, this is what makes navigation of the model so straightforward for end users.

Dimensional models place emphasis on information access and not the data management part of Data Warehousing. They simplify user access along well known paths but there are some forms of problem that are not well served by the techniques embodied in the approach, requiring additions such as lookups, helper tables, 'fact-less' fact tables and the like. The approach simplifies access but may limit the depth of analysis possible. This may of course not be an issue for some businesses but will be for others.

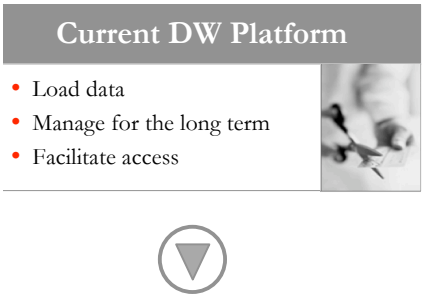
Given the need to manage the information asset over the long term, changes we might expect, such as organizational and reporting hierarchy changes, are often time consuming and cumbersome to achieve, resulting in major amounts of data manipulation. Deeper analysis needs, requiring rich product attribution, are also a challenge for those with millions of products and thousands of product attributes that do not necessarily overlap. One of the real challenges for a dimensional approach is in the development of conforming dimensions across the business that does not restrict analysis. This is a non-trivial thing to achieve if using an iterative development approach but without it the Warehouse will be little more than a collection of Data Mart silos.

The second school of thought was more traditional. Authors such as Barry Devlin and Bill Inmon argued that the devil was in the detail of the data and so the detail must be preserved! Third normal form data structures (3NF) place emphasis on the data management part of the Data Warehouse and trade this off against the information access part. They preserve a detailed record of each transaction without any data redundancy and allow for rich encoding of attributes and all relationships between data

elements, but users typically require a solid understanding of the data in order to navigate the more elaborate structure reliably.

Building a model in 3NF does not guarantee it is immune to business changes of course. For example, a hierarchy might be simply represented with one table in the Warehouse for each hierarchy level, but what happens if the business decides to change the number of levels in the hierarchy or for some levels to be skipped for a certain department or product? Both these changes would require a change to the physical data model and this inevitably also makes retaining a 'what was' reporting perspective problematic. In this case, something simple like a 'bill of materials' structure with a level table would avoid these structural changes.

Whatever your view, it's pretty clear that both model patterns have their merits and their drawbacks. It is also true that from a pure database standpoint Oracle has always been agnostic about the design used as, unlike other vendors, the Oracle database does an outstanding job of supporting both 3NF and dimensional designs with specifically optimized functionality. But the next generation of Data Warehouses will have to do much more than simply load, manage and then deliver up the data to a few casual users following solely a 3NF or the dimensional approach. To be effective, you need to leverage the strengths of both model forms, not just one of them. Anything else is a compromise!



How change is effecting requirements

Business needs as well as the database technologies and tools have moved on considerably since Data Warehousing became popular. No longer isolated, Data Warehouses are increasingly seen as a fundamental element of every business process, and part of the platform that enables business execution by providing insight into operational systems. While efficient business processes must be automated and process centric, effective ones must also be information driven!

The pressing need to deliver additional business value out of the data is driving a number of additional design considerations as organizations look to establish a new foundation for BI and Data Warehousing. These include:

- **Real-Time (R/T) and mixed workload.** The old notion of a single overnight batch load of data is long gone of course, but the operational requirements highlighted already drives the need to be able to load and query data at the same time without complexity around lock-escalations and reading dirty data which can compromise business trust and availability.
- **Pervasive Reach**. Business Intelligence is becoming pervasive, extending to stakeholders within the business as well as outside to trading partners, regulators and customers. It is also being consumed by processes through web services as well as people using more traditional tools. Importantly, once you view the provision of Business Intelligence as a service, you must also understand the impact this has on the platform used to deliver it. It must meet the same stringent performance and high availability requirements that must be satisfied by operational applications; Data Warehousing systems also become mission critical.

- **Changing Tools.** Tools change over time and will probably continue to do so although at a slower rate due to the trend for tools consolidation. Any new Data Warehouse architecture must allow for such tool changes by separating the way the data is stored, from any demands placed on the way the data is presented to the tools themselves.
- **Analytical Requirements.** Businesses are also much more analysis centric than they used to be, and in many companies performing this analysis often involves the use of spreadsheets and other specialist tools. This once again imposes additional platform costs and security risks, isolating users rather than allowing the sharing of insight and introduces latency. What's needed is for the Data Warehouse to facilitate real depth of analysis, not just provide simple dashboards and reports.
- **Realistic Platform.** Data volumes and analysis needs fluctuate along with business cycles and can grow rapidly over time. To manage these demand changes requires a realistic platform – one that can scale out capacity to meet these demands, is affordable by the business, and can be managed with the skills available in the market.
- **Data Quality.** Decisions based on bad data are inevitably going to be bad decisions! If the intention of the Enterprise Data Warehouse is to support strategic, tactical, and operational decisions by providing access to the right information at the right time using the right tools, then data quality and availability become critical concerns. Even if the data is not perfect, and it rarely is, every effort must be made to profile and improve data quality, as well as publicize quality standards and issues so decisions can be made in full knowledge of any limitations.
- **Regulatory Requirements.** Growing regulatory requirements are forcing companies to look at their overall corporate governance, risk, and compliance strategies. This is driving more data to be retained, including unstructured data, and for it to be retained for longer.
- **Roadmap and managing change.** It's one thing to recognize the need to change part of your architecture or tools landscape, and yet another to deliver it whilst still delivering on today's Service Level Agreements! Any new architecture must facilitate change through support for modern iterative design techniques as well as isolating changes to prevent them from rippling through the architecture.
- **Unstructured Data.** The majority of data held within a business as a whole is unstructured. Companies can reduce costs through rationalizing and consolidating this information, make better decisions through distilling meaning from the data, and use the derived values to further enrich existing data.

Another key area that has changed is the rate at which businesses themselves are changing. In the past, it was probably reasonable to base your IT strategy on just running ERP from a single vendor. In many industries, given the rate of consolidation that's occurring, this is probably no longer a viable position to adopt. The Data Warehouse must not only be ready for change – it must embrace it as reliable information in those times of change, making it a real differentiator.

One approach, born out of frustration in the business or as a direct result of limitations in specialist Data Warehouse hardware platforms, has been to push the data downstream into specialized Data Marts. While delivering better access to the data from the business point of view, this fracturing of the information asset reduces the overall business value, increases data latency, increases costs to the business in the longer term and introduces additional risks with regard to security and compliance. It also further limits analysis requiring the integration of the data, encourages definitional differences of common measures between Data Marts. Fragmentation of a Data Warehouse environment should be avoided as it impacts decision accuracy and makes integration into the decision centric business more problematic.

Requirements of a Reference Architecture

If we are to deliver BI pervasively across the business in a practical and cost effective fashion, any modern Data Warehouse design should satisfy the following criteria:

- Balance the demands of information management and information access in a single design concept.
- Allow the information asset to be preserved and enriched over the long term without having to unnecessarily re-architect the data as the business changes, including changes to hierarchy and product attributions or styles, depth and tools used for analysis.
- Provide layers of abstraction so that changes to tools and the type of analysis undertaken, as well as the inclusion of new data sources through business acquisition, do not necessitate in themselves a ripple effect of changes through the rest of the architecture.
- Ensure security of the information asset and be able to prove the provenance of the data and who has queried it in order to satisfy regulatory requirements. This must be the case even for privileged database users and when the database has been lost through hardware and other failures.
- Facilitate the use of future hardware technologies that might further reduce the cost and complexity of managing huge volumes of data.

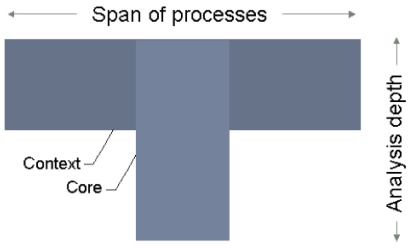


Figure 1 Analysis depth for core and context business processes

Packaged applications and the Reference Architecture

Developing a complete Enterprise Data Warehouse from scratch may guarantee you 100% fit to requirements, but there is always a price to pay regarding costs, effort, elapsed time and organizational focus. Building an Enterprise Data Warehouse can be a big elephant to swallow, especially for smaller organizations or ones with a more dynamic approach who may lack patience or executive support for such an undertaking.

While for the core areas of any business, that serve to differentiate the business in the market place, will undoubtedly benefit from such a bespoke solution, other areas that are more contextual to the way the business operates may not require such depth of analysis or flexibility.

Figure 1 illustrates the difference between core and context business processes by looking at the depth of analysis required and the span of business process covered. It shows how for context processes it may only be necessary to deliver fairly standard reports and dashboards, but core processes will require more complete analysis capabilities and tools such as management by exception, statistical analysis, forecasting, OLAP, text mining, data mining, spatial analysis and others.

For these context based business processes such as those typically served through commercial-off-the-shelf packages (COTS) including HR, Finance, CRM, Sales and Service, the best strategy may be to buy off the shelf analytical packages and adapt / extend them to better meet their needs. Oracle refers to these as BI Applications, and they include ETL, Data Mart, metadata and best practice reports and dashboards.

In many ways this is a repeat of the build versus buy debate that raged for some time around ERP and CRM applications. Given the evidence of the past, it is reasonable for us to conclude that BI Applications are here to stay, and any modern Data Warehouse architecture will need to integrate them into the design.

In order to preserve the high value add and rapid time to value of the BI Applications, wherever possible they should be implemented with the minimum of changes. Some data from the underlying data sources will also be required to give context to broader queries. These are considered as separate flows and a single version of the truth can be preserved through the BI Server component discussed in detail later.

An Enterprise Data Warehouse can be constructed and our "T" shaped analysis needs met by combining a number of BI Applications from COTS with the core data content from transactional systems.

For planning purposes, it is possible to deliver regular value to the business through salami slicing delivery of BI Applications, allowing sufficient time for the more complex task of modeling the core parts of the business to be undertaken. One additional benefit is that as the BI Applications come complete with pre-built ETL and reporting infrastructure, these may serve as a kick-start to understanding the ETL mappings and establishing the reporting layer.

ORACLE'S DATA WAREHOUSE REFERENCE ARCHITECTURE

The goal of Oracle's Data Warehouse Reference Architecture is to deliver a high quality integrated system and information at a significantly reduced cost over the longer term. It does this through recognizing the differing needs for Data Management and Information Access that must both be delivered by the Warehouse, applying different types of data modeling to each in a layered and abstracted approach.

The Reference Architecture is intended as a guide and not an instruction manual. Each layer in the architecture has a role in delivering the analytical platform required to support next generation business execution. The Reference Architecture gives us something to measure back against so we can understand what we compromise by making specific architectural, technical and tools choices. It works equally well for new Data Warehouse deployments as it does for developing a roadmap to migrate any existing ones.

A leading brand retailer in the UK used the Reference Architecture when they implemented their Data Warehouse in 1994. Since then they have seen major changes in the business as well as analysis needs brought about through the introduction of a Loyalty Card system. Even having experienced these sorts of changes, they have never had to go back and re-architect their data. They have been able to just add the new data and tools.

The following sections of this white paper describe straightforward manner each of the major components of the architecture, which is shown in Figure 2 below together with a description of the loading and data provisioning processes.

Although this white paper is new, the Reference Architecture presented is not. It has been developed and refined over many years. In fact, some customers first developed their Oracle based Data Warehouse some 14 years ago using the approach outlined here. What has changed is the capability of the underlying database technology and tools. With each new release we see either Data Warehousing specific or security and availability features added, making it significantly quicker and easier to implement and deploy the architecture.

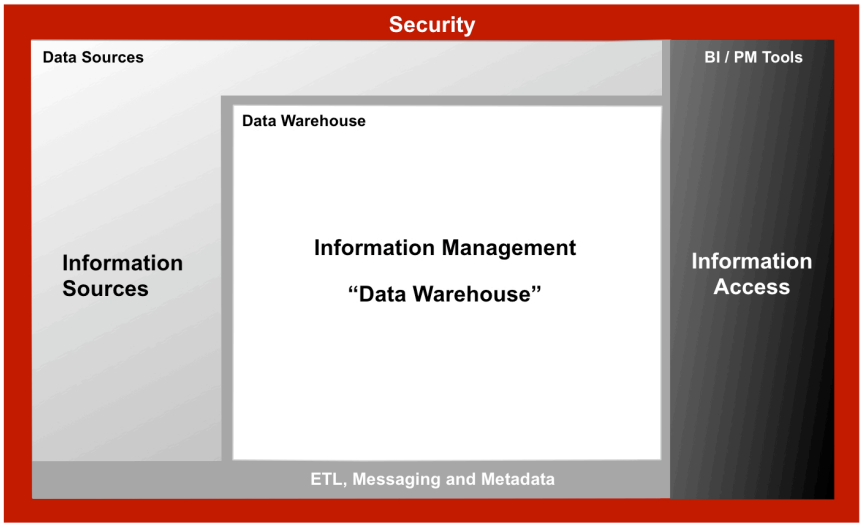


Figure 2 Main components of the Reference Architecture

The Reference Architecture has three main parts as shown in Figure 2. It comprises of the Information Sources, the Data Warehouse itself and the Business Intelligence and Performance Management tools (shown as BI/PM Tools in the diagram) along with ETL, Messaging, Metadata, and Security, which encompass all these parts. These various parts are broken down and described further in the following sections.

Data Sources

These represent all potential sources of raw data for the Data Warehouse, which are required by the business to address its information requirements (See Figure 3).

Sources include both internal and external systems and the data may be provided through a range of mechanisms such as real-time provision via messaging systems, change tracking through log-mining, database replication or simple file transfer. The approach taken will vary depending on source capability, capacity, regulatory compliance and access requirements

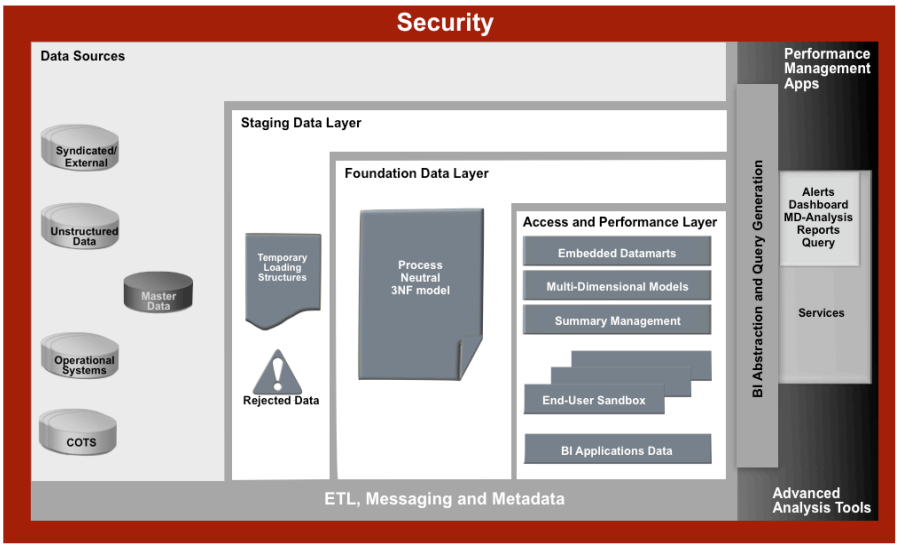


Figure 3 Oracle's Data Warehouse Reference Architecture

While the majority of the data sources for the Data Warehouse are highly structured in nature, increasingly there is also a need to supplement this with unstructured data. This data is typically either used for simple reference purposes - for example a Marketing Analyst may want to see the marketing collateral used for a highly responsive segment of customers - or to enrich the structured data through further processing using Text Mining or classification, clustering or feature extraction techniques.

Master Data Management (MDM) solutions are considered to hold the 'master' for any given business entity. The role of the Data Warehouse in respect of MDM is to preserve a record of changes across time, enable an analysis of these changes and provide the data as context to other analyses. Further data quality related checking may occur in the Data Warehouse as a separate process (not as part of the ETL) which may result in changes to master data. These changes are pushed back to the MDM solution and the new update received back into the Warehouse through the standard flow.

The Core Data Warehouse

The main Data Warehouse can be further subdivided into 3 conceptual (rather than physical) layers: the Staging Data Layer, Foundation Data Layer and Access and Performance Layer. These are shown in figure 3 and further described in the sections below.

Staging Data Layer

The first destination of data that has been extracted from sources is the Staging Layer. This layer acts as a temporary storage area for data manipulation before it enters the Data Warehouse and serves to isolate the rate at which data is received into the Data Warehouse from the frequency at which data is refreshed and made available to end-users. For example, in mobile telephony, UMS devices will typically create files containing call details every 100,000 records or 10-minute interval, whichever is the sooner. These will be loaded into the Staging Data Layer as they are received, but only made available for querying as dictated by business requirements. i.e. this is driven by business requirements and not the whim of the originating switch!

While many of the old rules regarding the static qualities of a Data Warehouse have now gone, it is still true that the Data Warehouse must contain data that is clean, consistent, and complete, as far as is practical. The Staging Data Layer is where business rules are applied to achieve these objectives. Rejected data is retained in this layer for manual or automatic correction. As with all other layers in the design, the Staging Data Layer is exposed to the BI Tools layer so loading performance, including data quality information, can be made available to end users if appropriate.

Foundation Data Layer

The Foundation Data Layer is sometimes referred to as the Atomic Data Layer. As the name implies, this layer records data at the lowest possible level of granularity. It represents the heart of the Data Warehouse and is the layer responsible for managing the data over the long term.

The Foundation Data Layer is modeled in a normalized fashion close to Third Normal Form (3NF) for storage efficiency. As the world of business is changing so rapidly, the data is also recoded in a business neutral fashion. This eliminates the impact of business changes and avoids any unnecessary re-structuring of data. For instance, a logical model may represent each level in a hierarchy in its own table (e.g. Sub-Dept, Dept, Division, Group) with a fixed one-to-many relationship between each level. What if the number of levels changes? What if the organization changes to a network management structure or skips levels for some departments? Using some simple design patterns it is possible to design around these sorts of issues.

The starting point for the logical model design may be a blank sheet of paper and map of existing source systems. More typical is to leverage an Enterprise Information Model sourced from one of the industry bodies or an Enterprise Data Warehousing model from database vendors such as Oracle or a specialist model vendor.

Some element of adaptation of the logical model to meet local needs is typically required and any process oriented representations included in the model must be removed before it can be transformed into a final physical model. As the key relationships and entities are all identified in an enterprise model, it is safe to implement incrementally using any combination of bespoke development and Commercial off the Shelf packages (COTS)/BI Application implementations.

Access and Performance Layer

A previously discussed, the 3NF model approach used to achieve the data management goals of the Foundation Data Layer is not helpful when it comes to providing users access to the data as it is more difficult to navigate. The Access and Performance Layer adds the Information Access component to our architecture. Most critical though is the realization that the tools we use to access the Data Warehouse will change over time. We may have very little control over which tools are adopted by a particular business unit – we perhaps all wish this were not the case! As new tools may impose different requirements in terms of the way that data is structured, it is critically important that we must be able to create these (or re-create them) from the underlying pool of data. That ability to re-constitute the data into different representation, either logically or physically is the entire *'raison d'être'* of the Access and Performance Layer.

Conceptually, this is the subject-oriented representation of a subset of data to simplify the analysis of the business while preserving the common dimensionality - nothing more, nothing less. Physical implementation in views or second tier aggregate structures is typically an implementation detail. They are created as required to facilitate access by a particular module or toolset. All aggregate structures available from Oracle are accessible from standard SQL, and can be added as required, leading to transparent performance improvements for end users and applications.

Population and update of access data layer structures can be highly automated, either using materialized views which keep track of data changes in sources or by extending the scope of ETL into an intra-ETL process.

The subject-oriented representations in this layer are referred to as 'embedded' dependent Data Marts. Several advantages accrue through embedding these structures rather than pushing the data out into a separate downstream Data Marts. These include:

- Reduced platform costs.
  There is no need to buy a completely separate hardware platform on which to run the Data Mart. Additional network bandwidth, power and cooling requirements are also reduced.
- Reduced data and analysis latency.
  Data does not need to be offloaded, transported, re-loaded and aggregated on a 2nd platform. This reduces latency and can improve accuracy as well as a result since the data is less stale.
- Reduced development latency.
  Development is often very rapid. With Oracle's ETL tools you can create a logical Data Mart design and then decide to either implement it relationally or multi-dimensionally by making a small metadata change.

- Improved security and management.
  Embedding Data Marts eliminate the need for administering a 2nd platform. Furthermore, The Oracle database is robust, secure and easily managed, which is often not the case for other platforms.

Specialized areas of analysis such as Data Mining and forecasting often necessitate data to be presented in very specific ways and may also involve data being modified and new data being written back. The notion of an 'Analysis Sandpit' is also included in the design. Sandpits may be created on a project, user or group basis. Data is provided and the analysis performed, reading and writing to the project area only. Once complete any results flow back to the target platform or into the Warehouse (properly) via the Staging Layer as normal. For example, data may be sampled from other structures in the Access and Performance Layer and restructured into a project schema. A data mining tool is then used to create a range of new customer segmentations. The best segmentation scheme is the selected through analysis and the new segment identifiers written back to the Customer Master Data Solution for each customer.

Business Intelligence and Performance Management Tools

This white paper began by outlining some of the forces that are driving requirements for a new breed of Data Warehouses, highlighting in particular the need to deliver BI pervasively, not just within the business but also to external stakeholders such as suppliers and customers. Pervasive BI has significant implications for both the way data is managed in the Data Warehouse as well as the way it is delivered through BI and Performance Management (PM) tools.

There has also been a recent trend towards tools standardization in order to deliver a more pervasive reach, drive up development productivity and drive down overall cost of ownership. Oracle has an excellent suite of BI tools and has many customer proof points that serve to illustrate how tools can gain pervasive reach in an organization given the right technical platform. The focus of this white paper is however not on tools, but on architecture, so the following sections pull out the most important aspects in respect of the Data Warehouse Architecture as a whole.

Other areas more closely associated with BI tools such as clustering, load balancing, caching and the user friendly UI are not covered here. While identified in the Reference Architecture, for the sake of brevity this white paper does not attempt to articulate the role of Performance Management Applications here.

BI abstraction and query generation

Most Data Warehouse solutions implement a range of end-user tools and applications to meet a series of business reporting and analysis requirements. Productivity issues are often associated with the wide range of such tools and the task of synchronizing any changes to data structures in the Data Warehouse with end-user reporting environments. If multiple tools are used to access the Data Warehouse, it is also very common for each tool to encode its own definition of common key performance indicators (KPIs), so that even though the Warehouse provides a single version of the truth, this truth is fractured and potentially changed within the BI tools.

Oracle's BI Server provides a further level of abstraction between the Data Warehouse and reporting tools so that the Data Warehouse data model and the tools can change at different rates. The BI Server is included in the architecture to provide a single enterprise view of the information regardless of the tools used to access the Enterprise Data Warehouse. The BI Server provides a single consistent metadata model, including derived and complex KPIs for consumption by any BI tool with an ODBC interface.

Iterative design methodologies are typically based around the development of a conference room pilot in conjunction with key project stakeholders. For Data Warehousing this can often be problematic as the data required will often not reside within the Warehouse in a suitable form when the project begins. This either means the data has to be artificially generated based on the enterprise model or source system map (a challenge in itself) or a more waterfall approach adopted so the data elements can be put in place first so an understanding of the data can be built before the requirements are explored. This imposed order makes little sense and often results in rework.

The query federation capability of the BI Server offers an alternative development approach by allowing the reporting tools to attach directly to sources while the design is developed. Once requirements are understood, a more rigorous approach to the logical model is taken and the data provisioned through the Data Warehouse in the standard fashion. From the BI tools perspective, this only necessitates a change to the BI Server metadata physical mappings. This kind of query federation capability can be exceptionally useful for relatively modest data volumes and in a development context as described, but it clearly does not address the broader data integration, data quality and production data volume challenges typically experienced in Data Warehousing. This second step of professionally managing the data in a Warehouse is therefore essential and should be included in the project plan from the outset.

From a physical implementation perspective, the BI Server also offers a robust architectural infrastructure in the form of clustering for high availability, load balancing and data caching. This makes it a sound choice when considering a more pervasive BI capability.

Linking insight to action with SOA

Our goal is to make insight actionable in response to insight gained in analysis. This is achieved using the SOA (Service Oriented Architecture) integration capabilities of Oracle's BI Suite. It allows a BPEL (Business Process Execution Language) workflow to be invoked from a dashboard and parameters (including those from user prompts to be) propagated through to the BPEL process orchestrated using SOA. It also allows for BI queries to be embedded in BPEL process workflows.

BPEL integration through SOA results in a highly productive environment with exactly the same query run from dashboards and reports, within an application or the process that orchestrates the components of the process and ties them all together.

ETL, Messaging and Metadata

The scope of ETL extends from data sources through all Data Warehouse layers into the BI Server layer for user representation of physical objects. Extract Transform and Load enables several processes including:

- Load Management
- Information Lifecycle Management
- Warehouse Management
- Data Quality Management

The Reference Architecture shows how ETL Messaging and Metadata act like fingers linking each layer in the Warehouse. Exposing metadata in this fashion allows standard BI tools to be used to report on ETL, data loading and data quality in the Data Warehouse. This can significantly improve the quality, reliability and trust in the data. In addition, Warehouse Management enables activities such as process monitoring and audits, sequence control and execution, job error and re-start handling

Metadata encompasses both technical and business vision of the total environment, such that it can be used to analyze problems on the data or analyze impacts of change, and be usable by end users during reporting and analysis activities.

Security

Data security is of increasing concern to every organization. Managing security is greatly simplified in the Reference Architecture: rather than fracturing the data into multiple downstream Data Marts for analysis, the design consists of a single data store with embedded data marts, representing a single point of administration and security enforcement.

Security extends throughout all layers in the Reference Architecture and is multi faceted. The Data Warehouse and BI tools are subject to any corporate wide security implementations such as LDAP and Active Directory. In addition, data should be protected in the database using role based security at the row level as a minimum. The use of features such as Virtual Private Databases and fine grained label security may also be required depending on the nature of the data and the threat imposed. The use of additional database features such as 'read-only Tablespaces' to prevent accidental changes to data can also be useful.

Best practice also suggests a separation of duties for privileged DBA's such that they are permitted to manage the database but not to access the data it contains. Oracle Database Vault provides this capability.

The ability to fully audit who has actually accessed what data may also be required to fully comply with some regulatory authorities.

Data Loading Process

Data is loaded into the Data Warehouse and made available for querying through the Data Loading process. This is shown in figure 4 below.

Data is received through a variety of mechanisms into the initial Staging Data Layer, both synchronously and asynchronously. It is processed through cleansing, enrichment, validation and integration steps and made ready for loading into the Foundation Data Layer. As previously outlined, the rate at which data is received onto the Warehouse platform and the frequency at which data is refreshed into the Warehouse, is driven by business needs.

Data is typically partitioned using a suitable schema such as date range and region. This allows for a more granular management throughout the data lifecycle. For instance, indexes can be built on each partition rather than the complete dataset and partitions of data loaded and unloaded from the Foundation Data Layer as they enter and leave the data lifecycle.

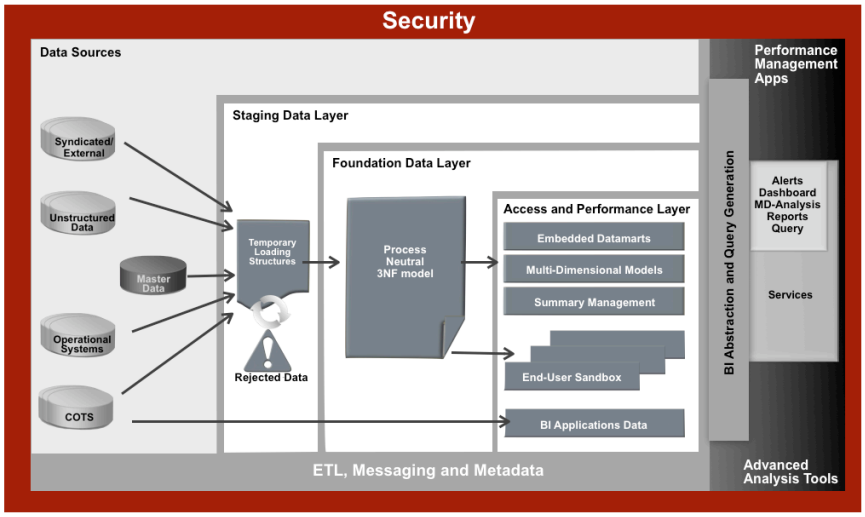


Figure 4 Data loading process

Once prepared in the Staging Data Layer, data can be moved into the Foundation Data Layer, as determined by the business requirement for the given flow of data. In Oracle, this requires a metadata change rather than a wholesale data movement, which is more costly. The majority of the Access and Performance Layer is made up of objects that will refresh automatically. This is true for views, Materialized Views and Cube Organized Materialized Views for instance. In these two latter cases, their definition will also define how they are refreshed, either when the data becomes stale or when a user queries the view, thus deferring the aggregation until some time later. For objects requiring load scripts to run, such as for a Create Table as Select (CTAS), an intra-ETL job will follow the loading of the Staging Data Layer.

In order for business users to have confidence in the Data Warehouse and for it to serve as the basis for regulatory reporting, the quality of the data and accuracy of queries are paramount. Oracle guarantees it will not read or write dirty data through the multi-version read consistency mechanism, which is unique in the industry.

We have previously described how and why there are the two streams from COTS packages into the Data Warehouse. One, required for pre-packaged BI Applications, will have pre-defined ETL as part of the application. The other flow, from COTS to Staging and Foundation Data Layers is processed in the standard fashion as for any other source flow. Any potential data conflict between the two flows is resolved in the BI Abstraction layer to preserve a *'single version of the truth'* for the user.

Information Provisioning Process

Data can be referenced by any BI tool and includes any DW layer as well as ETL and Data Quality process metadata. This allows for a broader analytical capability to be offered, allowing depth of analysis as well as width of business process coverage. That said, the majority of queries will of course be against the Access and Performance Layer, as its entire 'reason of existence' is to simplify user access to the data.

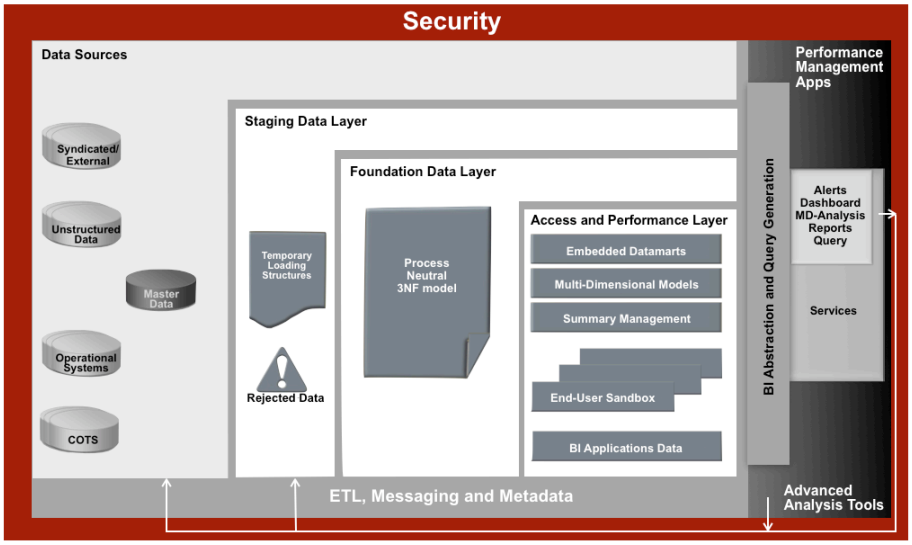


Figure 5 Outbound data provisioning into BI/PM Tools and beyond

Performance Management Applications are also able to query the underlying source systems directly but this is effectively out of scope with respect to the Data Warehouse and so is not covered in more detail in this white paper..

The BI Server can also dynamically map a logical value to multiple sources based on metadata and make this available for querying. For instance, a real-time picture of intra-day sales may be generated by joining the data in the Access and Performance Layer with data in the Staging Data Layer that is yet to be made available.

The additional Web Service capability enables seamless operationalization of the data within the organization and to the wider trading community for solutions such as Master Data Management and technologies such as BPEL.

Advanced Analysis tools and applications such as forecasting and Data Mining may through the analysis process create new data. Under the control of the tool or application, data can be read and written to and from the Analysis Sandpit area of the Access and Performance Layer. When a final set of data has been determined (such as a scored list of customers) a formal flow of data is executed that takes the data and moves

it to the operational system, MDM solution or back into the Data Warehouse via the Staging Data Layer. Data is never loaded directly back from the Access and Performance Layer into the Foundation Data Layer.

Conclusion

Modern businesses are demanding more and more from their Business Intelligence and Data Warehousing solutions. No longer just used for standardized reporting by just a few casual users, modern businesses are demanding faster and more pervasive access to information on which to base critical business decisions. This change to the volume, velocity and reach of the information is in turn forcing changes to the solution architecture and technology that underpins the solutions.

This white paper has sought to outline a practical Data Warehouse Reference Architecture that can enable the delivery of Business Intelligence pervasively in such a manner. This Reference Architecture strikes a balance between the data management and information access requirements of the Data Warehouse in a single design concept to ensure business value can be delivered in a sustainable fashion over time, without major re-engineering or loss of service while data is being re-engineered.

The Reference Architecture is a useful devise, which can be used as both a design template for new Data Warehouses designs, as well as a 'measuring stick' from which you can assess an existing Data Warehouse implementation and upon which roadmap options can be developed. The basic principles of the Reference Architecture are useful regardless of the precise technologies deployed to deliver it. However, Oracle Corporation is uniquely able to deliver integrated components from the disk to the dashboard of the reference architecture. Please contact your local Oracle account team for more information about how Oracle technology can be used to help you deliver Business Intelligence more pervasively through your organization.

Enabling Pervasive BI through a Practical Data Warehouse Reference Architecture
February 2010
Author: Doug Cackett
Contributing Authors: Andrew Bond, Kevin Lancaster & Keith Laker

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com