

Oracle Reference Architecture

Business Analytics Foundation

Release 3.1

E24714-04

June 2013

ORA Business Analytics (BA) Foundation, Release 3.1

E24714-04

Copyright © 2011, 2012, 2013, Oracle and/or its affiliates. All rights reserved.

Primary Author: Dave Chappelle

Contributing Authors: Stephen G. Bennett, Bob Hensle, Anbu Krishnaswamy, Mark Wilkins

Warranty Disclaimer

THIS DOCUMENT AND ALL INFORMATION PROVIDED HEREIN (THE "INFORMATION") IS PROVIDED ON AN "AS IS" BASIS AND FOR GENERAL INFORMATION PURPOSES ONLY. ORACLE EXPRESSLY DISCLAIMS ALL WARRANTIES OF ANY KIND, WHETHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. ORACLE MAKES NO WARRANTY THAT THE INFORMATION IS ERROR-FREE, ACCURATE OR RELIABLE. ORACLE RESERVES THE RIGHT TO MAKE CHANGES OR UPDATES AT ANY TIME WITHOUT NOTICE.

As individual requirements are dependent upon a number of factors and may vary significantly, you should perform your own tests and evaluations when making technology infrastructure decisions. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle Corporation or its affiliates. If you find any errors, please report them to us in writing.

Third Party Content, Products, and Services Disclaimer

This document may provide information on content, products, and Services from third parties. Oracle is not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and Services. Oracle will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or Services.

Limitation of Liability

IN NO EVENT SHALL ORACLE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES, OR DAMAGES FOR LOSS OF PROFITS, REVENUE, DATA OR USE, INCURRED BY YOU OR ANY THIRD PARTY, WHETHER IN AN ACTION IN CONTRACT OR TORT, ARISING FROM YOUR ACCESS TO, OR USE OF, THIS DOCUMENT OR THE INFORMATION.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Contents

Send Us Your Comments	vii
Preface	ix
Document Purpose	ix
Audience	x
Document Structure	x
How to Use This Document	x
Introduction to IT Strategies from Oracle (ITSO)	x
Conventions	xi
1 Introduction	
1.1 Business Intelligence and Business Analytics	1-1
2 General Concepts	
2.1 Structured Data Modeling for Analytics	2-1
2.1.1 Normalization	2-1
2.1.2 Dimensional Modeling	2-2
2.1.2.1 Types of Fact Tables	2-2
2.1.2.2 Star Dimensional Modeling	2-3
2.1.2.3 Snowflake Models	2-4
2.1.2.4 Constellation Models	2-5
2.1.2.5 Conformed Dimensions	2-5
2.1.2.6 Alternate Dimensions	2-5
2.1.2.7 Degenerate Dimensions	2-5
2.1.2.8 Junk Dimensions	2-5
2.1.2.9 Slowly Changing Dimensions & Data Versioning	2-6
2.2 OLAP	2-6
2.2.1 Aggregation & Summaries	2-6
2.2.2 OLAP Cubes	2-7
2.2.2.1 Types of OLAP	2-7
2.2.3 OLAP Operations	2-9
2.3 Structured Data Warehouse Strategies	2-9
2.3.1 Storage Components	2-9
2.3.2 Conforming Data Mart Approach	2-10
2.3.3 Centralized Data Warehouse Approach	2-11

2.3.4	Hub and Spoke Approach.....	2-12
2.4	Big Data, Fast Data, & Analytics.....	2-12
2.5	Enterprise Performance Management	2-13
2.5.1	Business & Strategy Planning	2-13
2.5.2	Financial Management & Reporting.....	2-15
2.5.3	Profitability Management.....	2-16
2.6	Pervasive Intelligence.....	2-16

3 Conceptual View

3.1	Overview.....	3-1
3.2	Information Capabilities	3-3
3.2.1	Information Provisioning	3-4
3.2.1.1	Historical Data Management.....	3-4
3.2.1.2	Analytical Data Management.....	3-4
3.2.1.3	Data Movement	3-4
3.2.1.4	Data Processing.....	3-4
3.2.1.5	Insight & Governance	3-5
3.2.1.6	Data Virtualization	3-6
3.2.2	Information Delivery.....	3-7
3.2.2.1	Information Services	3-7
3.2.2.2	Information Access.....	3-8
3.2.3	Information Modelling.....	3-8
3.3	Analysis Capabilities	3-8
3.3.1	Analysis Services	3-9
3.3.2	Analysis Processing.....	3-9
3.3.2.1	Analysis Techniques	3-10
3.3.2.2	Descriptive Analytics.....	3-14
3.3.2.3	Exploratory Analytics	3-15
3.3.2.4	Predictive Analytics	3-17
3.3.2.5	Prescriptive Analytics	3-18
3.3.3	Sense and Response.....	3-19
3.3.3.1	Event Handling.....	3-19
3.3.3.2	Response Actions.....	3-20
3.3.4	Analysis Delivery.....	3-21
3.3.4.1	Presentation Formats	3-21
3.3.4.2	Delivery Channels	3-21
3.3.5	Analysis Modeling.....	3-22
3.3.6	Enterprise Performance Management	3-23
3.4	Logical Data Warehouse Conceptual View	3-23
3.4.1	Staging Layer.....	3-25
3.4.2	Foundation Layer.....	3-25
3.4.3	Access & Performance Layer	3-25
3.4.4	Discovery Layer	3-26

4 Technology Standards

4.1	Dimensional Query & Interface Standards	4-1
4.1.1	OLE DB for OLAP (ODBO)	4-1

4.1.2	Multidimensional Expressions (MDX)	4-1
4.1.3	Open Geospatial Consortium (OGC) Standards.....	4-1
4.1.3.1	Web Feature Service Standards (WFS, WFS-T).....	4-2
4.1.3.2	Geography Markup Language (GML)	4-2
4.1.4	XML for Analysis (XMLA)	4-2
4.2	Programming Standards.....	4-3
4.2.1	MapReduce.....	4-3
4.2.2	R.....	4-3
4.2.3	Continuous Query Language (CQL).....	4-4
4.3	Reporting Standards.....	4-4
4.3.1	eXtensible Business Reporting Language (XBRL)	4-4

5 Interlocking Technologies

5.1	SOA, BPM, EDA, & BAM	5-1
5.1.1	BA & Business Processes	5-2
5.1.2	Real-Time Intelligence.....	5-3
5.1.2.1	BA and Event Processing	5-3
5.1.2.2	BA and Business Activity Monitoring.....	5-4
5.1.3	BA & Service Orientation	5-4
5.2	BA and Security.....	5-5

6 Summary

A Further Reading and References

A.1	Related Documents.....	A-1
A.1.1	Suggested Pre-reading	A-2
A.2	Other Resources and References.....	A-2

List of Figures

2-1	Sample Dimensional Model	2-2
2-2	Star Dimensional Model	2-3
2-3	Snowflake Model	2-4
2-4	Constellation Model	2-5
2-5	Conforming Data Marts	2-10
2-6	Centralized Data Warehouse	2-11
2-7	Hub and Spoke Data Warehouse Pattern.....	2-12
2-8	Performance Management Cycle.....	2-14
2-9	Analysis Supports Strategic Planning.....	2-15
3-1	High Level Conceptual View	3-1
3-2	Detailed Conceptual View	3-3
3-3	BA Information Architecture Capabilities	3-3
3-4	Analysis Architecture Capabilities.....	3-9
3-5	Data Warehouse Conceptual View	3-24
5-1	Business Analytics with SOA, BPM, EDA, & BAM	5-2
5-2	BA & Business Processes.....	5-2
5-3	BA & Event Processing.....	5-3
5-4	BA and Business Activity Monitoring	5-4
5-5	BA and SOA Services	5-5
5-6	BA & Security Architecture	5-6

Send Us Your Comments

ORA BA Foundation, Release 3.1

E24714-04

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this document?

If you find any errors or have any other suggestions for improvement, please indicate the title and part number of the documentation and the chapter, section, and page number. You can send comments to us at its_feedback_ww_grp@oracle.com.

Preface

Oracle Reference Architecture (ORA) is a product-agnostic reference architecture based on architecture principles and best practices that are widely applicable and that can be implemented using a wide variety of products and technologies. ORA does not include any implementation artifacts for the prescribed architecture. Rather, ORA addresses the building of a modern, consistent IT architecture while minimizing the risk of product incompatibilities and obsolescence.

ORA is an extensible reference architecture that describes many facets of IT. It is comprised of several documents that cover core concepts of technology, along with other documents that build upon these core concepts to describe more complex technology strategies.

ORA Business Analytics (BA) is an example of a complex technology strategy. It presents the Oracle Reference Architecture from the perspective of BA, introducing new capabilities and components that are most applicable to BA, extending the core material. This ORA Business Analytics perspective is comprised of two documents:

- *ORA Business Analytics Foundation*: offers a conceptual view of the architecture along with general concepts, capabilities, definitions, and technology standards.
- *ORA Business Analytics Infrastructure*: provides architecture principles, describes several logical (technical) views of the architecture, and maps Oracle technology to the architecture.

Note: The naming and structure of ORA documents is intended to be consistent across technology strategies. Any similarity between document names and Oracle products is purely coincidental. This document describes a product-agnostic reference architecture. It is not related in any way to product documentation.

This document is part of a series of documents that describe IT Strategies from Oracle (ITSO) business analytics strategy. Please consult the [ITSO web site](#) for documents pertaining to BA and other technologies.

Document Purpose

This document is the first of two documents that provide a business analytics perspective to the Oracle Reference Architecture. This document describes important concepts, capabilities, and technologies that help frame the reference architecture. The companion document, *ORA Business Analytics Infrastructure*, offers architecture principles and includes several architecture views including logical, product mapping, and deployment views. It also describes the role of infrastructure in the reference architecture.

Audience

This document is primarily intended for Enterprise Architects or others in a technology leadership position to aid in reference architecture design and roadmap planning. Solution Architects, Information Specialists, Data Scientists, and Business Analysts may also find it useful and informative.

Document Structure

This document is organized into chapters that build upon each other to form the basis of a reference architecture. The chapters are organized as follows:

[Chapter 1, "Introduction"](#) - introduces the subject of business analytics and describes the scope of this material.

[Chapter 2, "General Concepts"](#) - describes important foundational concepts pertaining to business analytics.

[Chapter 3, "Conceptual View"](#) - defines a conceptual view based on a desired set of business analytics capabilities.

[Chapter 4, "Technology Standards"](#) - introduces several technology standards that apply to business analytics.

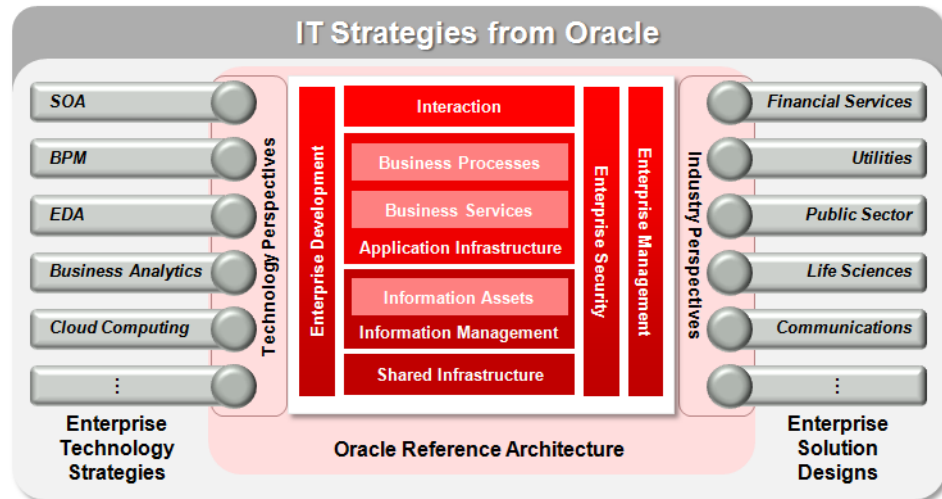
[Chapter 5, "Interlocking Technologies"](#) - provides a conceptual linkage between business analytics and several other technologies.

How to Use This Document

This document is designed to be read from beginning to end. However, each section is relatively self contained and could be read independently from the other sections. Persons familiar with business analytics concepts may wish to skip ahead to [Chapter 3, "Conceptual View"](#).

Introduction to IT Strategies from Oracle (ITSO)

IT Strategies from Oracle (ITSO) is a series of documentation and supporting material designed to enable organizations to develop an architecture-centric approach to enterprise-class IT initiatives. ITSO presents successful technology strategies and solution designs by defining universally adopted architecture concepts, principles, guidelines, standards, and patterns.



ITSO is made up of three primary elements:

- **Oracle Reference Architecture (ORA)** defines a detailed and consistent architecture for developing and integrating solutions based on Oracle technologies. The reference architecture offers architecture principles and guidance based on recommendations from technical experts across Oracle. It covers a broad spectrum of concerns pertaining to technology architecture, including middleware, database, hardware, processes, and services.
- **Enterprise Technology Strategies (ETS)** offer valuable guidance on the adoption of horizontal technologies for the enterprise. They explain how to successfully execute on a strategy by addressing concerns pertaining to architecture, technology, engineering, strategy, and governance. An organization can use this material to measure their maturity, develop their strategy, and achieve greater levels of adoption and success. In addition, each ETS extends the Oracle Reference Architecture by adding the unique capabilities and components provided by that particular technology. It offers a horizontal technology-based perspective of ORA.
- **Enterprise Solution Designs (ESD)** are cross-industry (applicable to many industries) and industry-specific (focused on a single vertical industry) solution perspectives based on the Oracle Reference Architecture. They adhere to the ORA principles and also draw on the best practices and guidelines provided in ETS collateral. They define the high level business processes, business functions, and software capabilities that are required to build enterprise-wide industry solutions. ESDs map the relevant application and technology products against solutions to illustrate how capabilities in Oracle's complete integrated stack can best meet the business, technical, and quality-of-service requirements.

ORA Business Analytics Foundation, along with *ORA Business Analytics Infrastructure*, extend the Oracle Reference Architecture. They are part of a series of documents that comprise the Business Analytics (BA) Enterprise Technology Strategy, which is included in the IT Strategies from Oracle collection.

Please consult the [ITSO web site](#) for a complete listing of BA and ORA documents as well as other materials in the ITSO series.

Conventions

The following typeface conventions are used in this document:

Convention	Meaning
boldface text	Boldface type in text indicates a term defined in the text, the <i>ITSO Master Glossary</i> , or in both locations.
<i>italic text</i>	Italics type in text indicates the name of a document or external reference.
<u>underline text</u>	Underline text indicates a hypertext link.

Introduction

Successful companies are always looking to gain a competitive advantage, to be one step ahead of the competition. Sometimes this occurs by being more efficient and nimble as an organization; able to spot trends and adjust course quickly and easily. Other times it involves correctly predicting the future; targeting the right products to the right people at the right time and price. In addition, companies can create a competitive advantage by quickly recognizing new business opportunities and being first to market with new products and services.

Achieving these objectives requires the development of a comprehensive business analytics strategy. The strategy must provide the ability to harvest all the information available, analyze it to more fully understand trends in the business, mine it for patterns of behavior, detect opportunities, simulate and forecast results, set strategy, monitor key business indicators, and take swift action when anomalies occur or when opportunities arise.

Business analytics should not be a matter of hoarding data to generate reports. Likewise, analysts should not spend their days collecting and preparing data, and IT should not waste time on ad hoc integration and data re-engineering. Rather, information must be made available in a form that allows analysts to understand it, navigate it easily, and take immediate action. Analysis must be able to drive business processes, enable business to react to events quickly, and provide meaningful insight when and where it is needed.

This reference architecture presents a view of business analytics designed to offer companies the advantages they are looking for while minimizing the time and expense often associated with such endeavors

1.1 Business Intelligence and Business Analytics

Business intelligence (BI) is a general term that can be interpreted in many ways. Forrester Research defines it as "a set of methodologies, processes, architectures, and technologies that transform raw data into meaningful and useful information. It allows business users to make informed business decisions with real-time data that can put a company ahead of its competitors."¹

For many years BI was a term used to encompass all aspects of methodology, process, architecture, and technology that gathered and presented information to support business decisions. It encompassed the provisioning of information, (also known as data warehousing), generation of reports, presentation of dashboards, and analysis of historical data. BI also included forward-looking disciplines such as predictive modeling and simulation.

Decision making also applied to the definition and management of business performance. This is referred to as either Performance Management (PM), Enterprise Performance Management (EPM), Business Performance Management (BPM), or Corporate Performance Management

¹ Boris Evelson, "[Topic Overview: Business Intelligence - An Information Workplace Report](#)"

(CPM). For the purpose of this document the term EPM will refer to those aspects of business intelligence that pertain to strategic planning, forecasting, and enterprise performance scoring and evaluation.

The recent emergence of "Big Data", and technologies that support the analysis of less structured data, has greatly promoted the role of advanced analytics. As a result, the term "Business Analytics" has become a hot topic in the industry.

Like BI, Business Analytics (BA) is defined in many ways. Thomas Davenport and Jeanne Harris define analytics as "the extensive use of data, statistical and quantitative analysis, explanatory and predictive models, and fact-based management to drive decisions and actions"². This effectively positions BA and BI as peers, overlapping in some areas such as structured data analytics (e.g. OLAP), predictive modeling, and various applications of statistical analysis.

Lately BA has also become an umbrella term within the industry that encompasses the emerging aspects of analytics as well as the traditional aspects of BI. BI is positioned mainly as a descriptive form of analytics. It complements other forms such as predictive, prescriptive³, and exploratory analytics⁴. The precise number and definition of forms depends on one's source of reference. What is fairly consistent is the encapsulation of BI within a multi-faceted definition of BA.

One reason for aligning with this definition is the recognition that in either case information is being acquired, organized, explored and exploited for business gain. For example, a decision may be based on corporate strategy, historical analysis, sentiment analysis, statistical analysis, some other means, or a combination of techniques. The means may evolve over time although the business objective remains constant - to make a specific informed decision. It may be difficult to say whether BI or BA is used to make a decision given that the line between intelligence and analytics is easily blurred.

From an architecture perspective, it is better to have a unified intelligence/analytics vision, and from an implementation perspective, there are benefits to the organization where standardization and consolidation can be achieved. Also, it is simply easier to adopt one term to address the entire strategy, especially since no officially recognized all-encompassing term yet exists.

Given the industry support and popularity of the term "Business Analytics", this reference architecture will use that moniker to portray both BI and analytics. It will also include the architecture to support Enterprise Performance Management.

From an IT perspective, Business Analytics will involve the acquisition and management of information as well as the infrastructure, platforms, and tools necessary to view, model, analyze, and leverage information appropriately. Given the breadth of discussion around information management (e.g. data movement, cleansing, provisioning, virtualization, master data management), this portion of the architecture is presented in a separate document entitled *ORA Information Management*. Although many aspects of information management apply distinctly to Business Analytics, several may be used by other technology strategies. For this reason information management is positioned within ITSO as a core reference architecture document. The remainder of the architecture, i.e. the use of information for analysis and intelligence purposes, is presented in this document and continues in *ORA Business Analytics Infrastructure*.

² "Competing on Analytics: The New Science of Winning", Davenport, Thomas H., and Jeanne G. Harris, 2007, Harvard Business School.

³ Irv Lustig, Brenda Dietric, Christer Johnson, and Christopher Dziekan, "[The Analytics Journey](#)", Analytics, Nov/Dec 2010

⁴ Manan Goel, "[The Art of the Possible with Business Analytics](#)", Analytics Powered Pursuit of Excellence

General Concepts

This chapter provides an understanding of foundational concepts that underpin the reference architecture. It builds upon concepts presented in *ORA Information Management* by elaborating on various ways information is organized and used to support business analytics.

2.1 Structured Data Modeling for Analytics

This section describes concepts related to various forms of structured data modeling and how they apply to business analytics solutions.

2.1.1 Normalization

Relational Database Management Systems (RDBMS) are designed to store data in tables that support relationships between tables in the form of foreign keys. In order to maximize efficiency and minimize redundancy, data is usually "normalized". Following the normalization process, common repeating data values are stored once in a separate table and linked to non-repeating data values in another table.

As a simple example, two records, each representing the sales of an item to a single customer, would be split into multiple table entries. The customer information would appear once in a Customer table, while each of the sales records would appear as rows in the Sales table. The tables would be linked by a foreign key relationship.

Normalization in this manner not only reduces the total amount of data that needs to be stored, but also promotes consistency and makes it easier to maintain. A change to customer information, for example, would involve changing only one entry in the Customer table vs. many entries in a combined Customer/Sales table.

Based on the method pioneered by Edgar F. Codd there are different degrees to which normalization can be applied. The benefits of various degrees of normalization must be weighed against the added complexity they bring. Sometimes data is intentionally "de-normalized" in order to improve system performance and make the schema easier to understand.

A very widely adopted form of normalization is known as third normal form (3NF). It is often considered the balance point between "too much" and "too little" normalization. Many operational data stores base their schema design on 3NF with some variations and customizations to address specific access and performance needs. However, data marts and data warehouses tend to stray from this practice in order to cater to access patterns and tools that are designed around other forms of modeling, (described later in this section). Detailed information on various forms of normalization can be found in various texts and online sites such as techopedia.com.

2.1.2 Dimensional Modeling

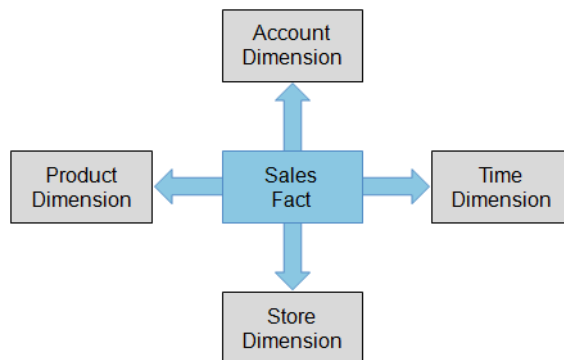
Online transactional processing (OLTP) systems benefit greatly from normalization as they are designed to perform specific functions and are programmed to access data in specific pre-defined ways. Programmers can take time to understand the schema and formulate queries, updates, etc. that link and span tables as needed. The efficiencies of normalization often far outweigh the burden of added complexity.

However, this dynamic changes when data needs to be navigated by business users rather than technology experts. Business users understand how the business is organized, not how a database is organized. Complex relationships, joins, and nesting make sense to a data architect but not so much to anyone else. In order to sift through data efficiently and effectively, the business user needs to view data more simply and hierarchically, in a form that best mimics the way the business operates.

Dimensional modeling is a method of standardizing and organizing data in a way that represents natural business hierarchies. For instance, a sales representative works in a sales district, which is part of a sales region, within a sales geography. Likewise, an item that is sold can belong to a product category, which in turn can belong to a higher level product group. Each of these two hierarchies is considered a dimension.

Business events can be linked to several different dimensions, as shown in [Figure 2–1](#). For example, an item being sold can be linked to dimensions that represent customer account, product, location, and time. The event or measurement of what happened is called a fact. Facts are generally stored in tables that link to the dimension tables via a collection of foreign keys.

Figure 2–1 Sample Dimensional Model



There are several forms of dimensional modeling that are variations or extensions on this theme. They include star models, snowflake models, galaxy models and constellation models. Likewise, there are several different types of fact tables that serve different purposes. The following sections describe these aspects of dimensional modeling.

2.1.2.1 Types of Fact Tables

Fact tables generally represent a measurement of some event that has happened. The measurement can be of a single event, a collection of events, a snapshot in time, other variations of events, and even non-events. Common types of fact tables include:

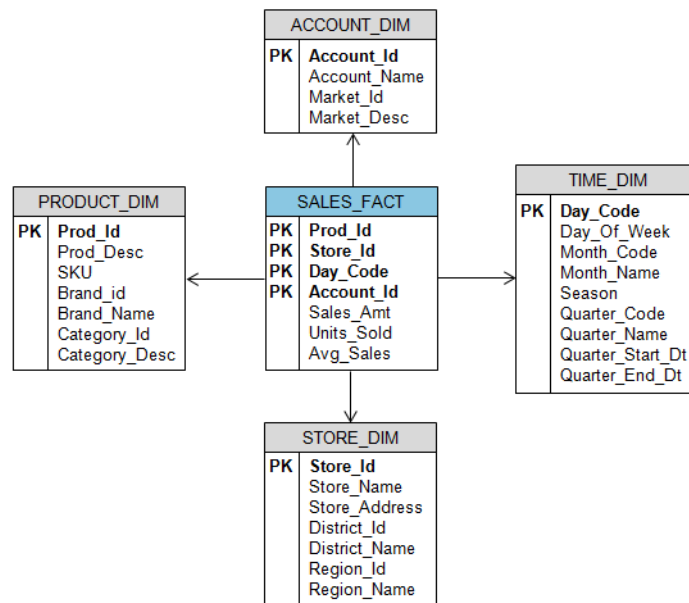
- **Transaction Fact Table.** This fact table records events at an atomic level, i.e. transaction. For example, each unique type of item sold would be recorded in a separate fact table. Fact tables must be aggregated in order to produce a report on how many items of a particular type were sold to a customer over a given period of time.
- **Periodic Fact Table.** Periodic fact tables record aggregate values for an item and a given period of time. These tables represent totals that are updated as transactions occur.

- n **Snapshot Fact Table.** A snapshot fact table can be used to represent facts for a specific point in time. It works by establishing snapshot time frame attributes for a dimension and creating a separate fact table for each snapshot time period. This method can be used to monitor trends over time and handle frequent changes to data in dimensional tables. It can be used as an alternative to the various types of slowly changing dimensions listed in [Section 2.1.2.9](#).
- n **Factless Fact Table.** This table can be used to indicate the occurrence of an event where no measurement data is necessary; for example, a student attending a class. All of the data pertaining to the student, class, date, etc. are stored in dimensional tables. The existence of a factless fact table simply indicates that a correlation between specific dimensional rows is warranted.
- n **Coverage Fact Table.** A coverage fact table is actually a type of factless fact table that is most useful in determining what did not occur. For example, if a student was supposed to attend a class but didn't, it could be difficult to identify this non-event and how often it was happening. A coverage fact table can be created for all students that are enrolled in a class, and an ordinary factless fact table can be created for students that actually attended the class. A report on who didn't attend can be compiled by examining the difference between (count of) the two types of tables.

2.1.2.2 Star Dimensional Modeling

A star model, [Figure 2–2](#), is a dimensional model with one fact table linked to several dimensions. The dimensional tables are not normalized. E.g. data redundancy is allowed to occur in order to minimize the number of tables and relationships. The fact table contains foreign keys to each of the dimensions, but very little else. Over time the fact table will accumulate many rows, however the dimension tables will remain relatively static.

Figure 2–2 Star Dimensional Model



Star models are the easiest to understand due to their simplicity. It is easy for business users, and the tools they use, to navigate this schema because all hierarchies, tables, and relationships are well-known business entities and business relationships.

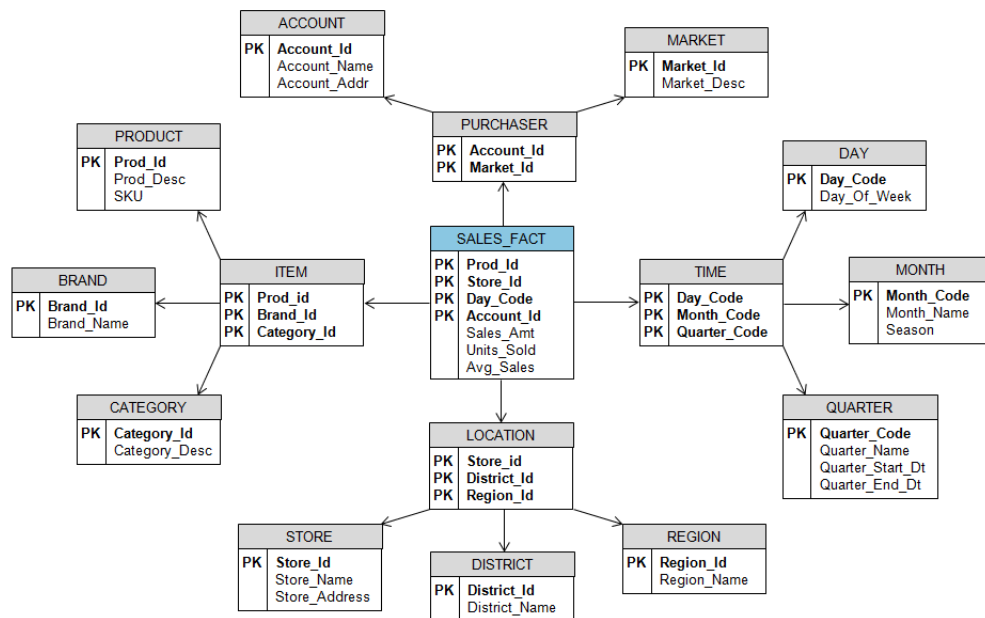
Queries using a star model are fairly simple and straightforward. Queries must join dimensions and facts through foreign key relationships. The lack of 3NF means that the number of tables and joins is minimized.

2.1.2.3 Snowflake Models

Lack of 3NF can be a problem for star schemas, particularly when dimension tables are very large. The number of rows involved in a query has an effect on performance. Likewise, the size of each row (number of attributes/columns) can become an issue for storage capacity as well as performance.

One solution to this is to normalize the dimension tables. Doing so produces a schema with star dimensional form, but with additional reference tables branching off of the dimension tables. It results in what is called a snowflake schema, shown in [Figure 2–3](#) below.

Figure 2–3 Snowflake Model



Snowflake schemas introduce trade-offs between the benefits and complexity of 3NF. They are generally less desirable than the simpler star schemas, but may be preferred for certain cases, such as:

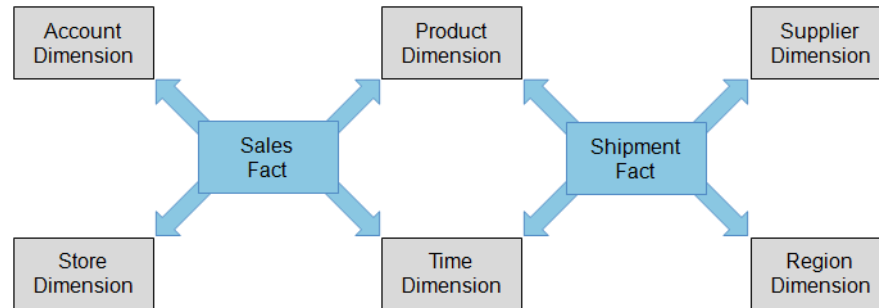
- Handling situations where dimension tables include an attribute that most dimension records have a NULL value for; a.k.a. sparsely populated attributes. These attributes can be moved to a sub-dimension.
- Storing dimensional attributes that have very few unique values; aka low cardinality attributes. These attributes can be moved to a separate table and referenced by the dimensional table.
- Storing attributes that are frequently queried independently. Moving these attributes to a separate table can make the query much more efficient; for example: day, month, quarter, and year attributes of a date hierarchy.

One solution is to normalize only the larger dimensions. This results in a partial snowflake model. If a snowflake schema is used, then access mechanisms or analysis tools should be chosen that insulate end users from the underlying complexity.

2.1.2.4 Constellation Models

Constellation models are dimensional models that contain multiple fact tables that share common dimensions. They appear as a collection of star dimensional models, shown in Figure 2-4.

Figure 2-4 Constellation Model



Constellation models make it possible to compare facts accurately by supporting queries based on shared or conformed dimensions.

2.1.2.5 Conformed Dimensions

A serious challenge with business analytics solutions is the distribution of data across an organization. This often results in silos of data that are similar, but have slightly different representations. Any difference in the way an entity is represented makes it very difficult to combine data accurately.

For example, if sales volumes are stored in multiple locations and the time dimension in each location is represented differently, then it is hard to query and aggregate, or query and compare sales by time (week, sales quarter, fiscal year) and get meaningful results.

Conformed dimensions are dimensions that are either exactly identical, or where one dimension is a perfect subset of another. This allows query results (row headings and hierarchy levels) to match perfectly. In addition, if fact tables conform to identical units of measure, then measures can be mathematically combined to produce accurate totals.

2.1.2.6 Alternate Dimensions

Alternate dimensions are dimensions that provide different ways to view data. For example, if a group wanted to view sales by calendar time (week/month/quarter/year) instead of fiscal time, then an alternate time dimension can be added to the model.

2.1.2.7 Degenerate Dimensions

As a consequence of dimensional modeling, there are often data attributes that can be used to group multiple facts, but are not part of any dimensions. An invoice number, for example, spans multiple invoice item facts. All attributes of the invoice are either included in a dimension or fact table. Rather than create a dimension table for invoices, one can choose to include the invoice number as a degenerate dimension attribute of the fact entries. It resembles a key to a dimension, but there is no actual dimension table for it to reference since there are no remaining attributes associated with the invoice to record.

2.1.2.8 Junk Dimensions

Junk dimensions are dimension tables that hold attributes, indicators, etc. that do not belong to any of the existing dimensions. They actually belong to the fact entry and can be stored as attributes of the fact. However, in order to reduce the size of fact tables these miscellaneous

fields can be extracted into separate table(s) that the fact table links to, mimicking the relationship to ordinary dimension tables.

2.1.2.9 Slowly Changing Dimensions & Data Versioning

Data in dimension tables act as reference data for facts. However dimension data values are not always constant. They do change over time, which can have an effect on historical consistency.

For instance, a customer that changes addresses could affect sales metrics if they move from one sales district to another. However, there will likely be no impact at all if a customer changes their phone number. Depending on the type of change, updates to dimension data may need to be handled differently.

There are several well-recognized methods for managing changes to dimensional data:

- **Type 1: Replace the value.** In the case of a phone number change, the old value is simply overwritten with the new value. This method makes sense when changes have no effect on the value of historical consistency.
- **Type 2: Add a record with an effective start date and effective end date.** This solution preserves history by maintaining both values in separate records. The date fields are used to determine which value is in effect at a particular time.
- **Type 3: Store the old value.** This solution provides some backward reference by maintaining historic values in "previous value" fields. Only one record is maintained, and the number of historic values is often limited to one. It is most useful when historic values provide informational value, but are not necessary for computational value.
- **Type 6: A combination of Types 1, 2, and 3.** In this case the current value is replaced, a previous value is maintained, and a new record is added with start and effective dates. This option provides the most information, but has the highest cost in terms of storage and maintenance.

2.2 OLAP

Online Analytical Processing (OLAP) is an approach to provide business insight by using dimensional models to satisfy multi-dimensional queries. The output of a query is often represented in a matrix or pivot table form.

Multi-dimensional queries define the subset of dimensional data to be analyzed. They define the dimensions and levels from which to results sets are returned. For example, a query might ask for all sales data from store 123 pertaining to brand X for the previous six months. The query might also ask to compare results with the same data from the previous year.

2.2.1 Aggregation & Summaries

Data in a warehouse is often stored at the lowest level of granularity. E.g., a sales transaction that includes multiple line items will result in fact table entries for each line item. As a result, fact tables can grow enormously large over extended periods of time. Queries aimed at summarizing data may need to scan thousands or millions of records in order to compute the requested totals.

Given the frequent need to summarize data for meaningful comparisons, it makes sense to consider pre-calculating some of the totals. This can drastically improve query performance and reduce the processing power needed to handle OLAP functions. The summarized values are called aggregations.

Aggregations can be created and stored for any or all of the dimensional hierarchy levels. For example: sales by store, district, or region; sales by product, brand, or category; and sales by

day, month, quarter, or year. Aggregations can also combine dimensions, e.g. sales by product, quarter, and region.

In terms of architecture, the system must be able to determine when the aggregates are available to help satisfy a query, even when they are not the full answer to a query. For instance, if a user requests sales figures for the year, and aggregates are available at the month level, then month aggregates should be summarized to produce a result. The system must recognize the most efficient means to produce a result and automatically alter the query path accordingly.

Aggregations are very powerful, but there is a trade-off between run-time performance and "load-time" preparation and storage. Whenever any new data is added to the model, all (affected) pre-summarized data values must be recalculated. In addition, summary data must also be stored, thus increasing the overall storage capacity requirements.

It is important to consider several factors when developing an aggregation strategy, such as:

- The size of fact tables and distribution of facts across dimensions. Aggregation may be necessary when a large number of facts are concentrated on specific dimension elements.
- The duration of a query without summarization.
- The frequency a given summary is typically used.
- The frequency that data is added to the model.
- The time available to perform aggregations. This takes into account factors such as data loading time, frequency, and the uptime requirements of the system (e.g. 7x24 vs. 5x9).

An alternative to pre-aggregation is post-aggregation. With this method summaries are not created in advance, rather they are calculated when the query is first performed. They are subsequently stored or cached for use on future queries.

2.2.2 OLAP Cubes

An OLAP cube is a way of representing dimensional data along with aggregations. Building on the star dimensional model, facts can be aggregated at each level of each dimension. Data can also be aggregated at the intersection of each level and dimension.

Given two dimensions, the result of all possible aggregations would resemble a matrix, i.e. a spreadsheet. Given three dimensions, the result would resemble a cube. An OLAP cube can contain more than three dimensions; although geometrically inaccurate, the model is still referred to as a cube.

2.2.2.1 Types of OLAP

OLAP cubes are a way of representing (presenting) data to business intelligence consumers. The manner in which data is actually physically stored and managed can vary. A cube can be based on different underlying technologies and architectures, and draw data from a variety of sources.

Several common types of are described below.

2.2.2.1.1 Relational Online Analytical Processing (ROLAP)

ROLAP is a type of OLAP solution that is based on a relational database management system. It uses star or snowflake schemas to hold facts and dimensions, and may also include summary tables to hold aggregates. Data is usually accessed via standard SQL queries.

The use of standard RDBMS technology allows ROLAP to scale well for very large data sets. It also allows the data to be modeled in ways that deviate from strict dimensional model formats, adding tables, relationships, and normalization to enrich the way data is stored; which may add

value where data happens to be used for other purposes. In addition, standard database features common with OLTP solutions, such as row level security, high availability, etc. can also be leveraged for OLAP.

ROLAP, however, has generally been shown to perform worse than implementations based on dimensional databases. Also, summary tables usually must be computed as a function of data loading rather than being handled automatically by the OLAP solution itself.

2.2.2.1.2 Multidimensional Online Analytical Processing (MOLAP)

MOLAP solutions are designed specifically for multidimensional queries. They generally store data in optimized multidimensional arrays, and automatically compute summaries when data is loaded.

MOLAP, in general, tends to perform better than ROLAP due to storage, indexing, aggregation, and caching implementations that are tailored to dimensional data models. It also tends to require less overall storage space than ROLAP.

Disadvantages of MOLAP differ from product to product. Architecturally, a pure MOLAP solution can introduce additional overhead in terms of having another data source to manage. Where ROLAP offers the flexibility to use data for other purposes, MOLAP is designed solely for analytics. Therefore with this type of solution data must be copied from a traditional data warehouse or staging area in order to be used for analytics.

A MOLAP solution can be built on top of a relational database, as is the case with Oracle OLAP. The OLAP cubes are multidimensional objects stored within the Oracle database. This eliminates data integration issues between relational and multidimensional data sources. It also brings several RDBMS benefits such as SQL access, high availability, backup and recovery, and security. Oracle OLAP can also be implemented in a HOLAP fashion.

2.2.2.1.3 Hybrid Online Analytical Processing (HOLAP)

HOLAP is a combination of ROLAP and MOLAP where some data are stored relationally and others are stored using MOLAP. The intent is to combine technologies to get the best of both worlds.

The way data is partitioned between ROLAP and MOLAP can vary. In a vertical partitioning approach, aggregations are handled in MOLAP while detailed data is persisted in ROLAP. This combines the speed of MOLAP with the flexibility and bulk data handling capabilities of ROLAP.

Data can also be partitioned horizontally, based on hierarchical concerns such as time, or cubically, based on a combinations of dimensions that may contain large (dense) or small (sparse) amounts of data.

The disadvantage of HOLAP is that data must now be maintained in more than one place. The ROLAP and MOLAP data stores must be kept synchronized in order to avoid inconsistent results, and the partitioning strategy must be well managed.

2.2.2.1.4 Extended Online Analytical Processing (XOLAP)

Another variation on the theme is XOLAP. This model is similar to ROLAP in that data is stored relationally, however it adds a multidimensional front end processor that can efficiently handle dimensional queries. It uses metadata to map dimensional queries, (using MDX), to relational queries, (using SQL).

The front end processor appears to users as a MOLAP solution and provides capabilities to handle data calculations. However, data is stored and aggregated in a relational database.

2.2.3 OLAP Operations

With a dimension representation of data in place, one can begin to view data from a number of vantage points. Each combination of dimensions, and level of dimensional hierarchy, can offer clues as to what has transpired. The data can be observed in order to evaluate current performance. It can also be compared to similar data sets in order to spot trends, similarities, and differences across dimensions.

OLAP tools are meant to enable the consumption of data in ways that maximize business value while minimizing the time and effort required achieving that value. Common OLAP operations include:

- **Slicing** - observing a subset of multidimensional data where one or more dimensional values are fixed. For example, if store number is set to a fixed value, then the resultant data set would be considered a slice of the original cube. The store dimension/value would constrain the results but would not be included in the results set.
- **Dicing** - further slicing of data by fixing one or more additional dimensional values.
- **Drilling (up or down)** - observing lesser or greater detail by navigating up or down the dimensional hierarchies. The top-most level of a hierarchy provides the greatest degree of summation, while the bottom-most level provides the greatest level of detail.
- **Rolling-up** - computing all of the data relationships for one or more dimensions. To do this a computational relationship or formula might be defined.
- **Pivoting** - rotating, or re-orienting the layout to observe data from a different dimensional point of view. A matrix or spreadsheet can easily represent two-dimensional data sets, however a cube often has three or more dimensions of data. Pivoting allows the user to change the dimensions being displayed so that data can be viewed from alternate vantage points.

2.3 Structured Data Warehouse Strategies

Data warehousing is not a new concept; organizations have been doing it in one form or another for some time. Traditionally, data warehousing has been used for well structured data, such as data generated from operational systems. Lately, more and more unstructured, or less structured data has been included. This section introduces some of the architecture building blocks of structured data warehousing and describes some of the more well-known architecture strategies. Later, in [Section 3.4](#), the structured data warehouse is combined with less structured data, data processing, and virtualization capabilities to form what is termed a logical data warehouse.

2.3.1 Storage Components

A data warehouse strategy for business analytics can involve the use of several types of storage components, (aka architecture building blocks), including:

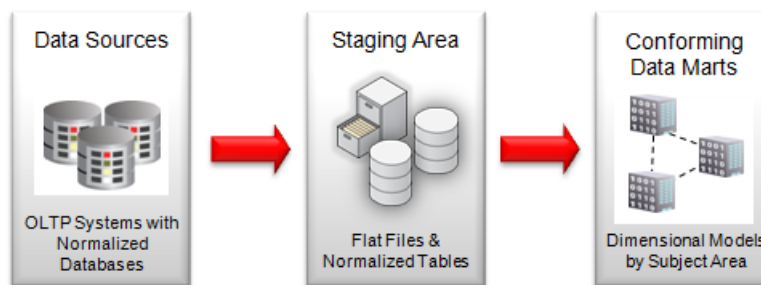
- **Operational Data Store (ODS)**. An ODS is a copy of an operational database that is most often used for reporting purposes. The reasons an ODS exists are generally twofold. First, the OLTP system database can't handle the additional load placed on it by users running reports or doing ad hoc queries. A separate system with a copy of OLTP data is needed. Second, a more universal reporting and analysis solution is not available to manage the data, so a tactical decision is made to deploy a solution for the requirements at hand. An ODS usually contains incremental snapshots of OLTP data, often in a form that closely resembles the schema(s) of operational databases. An ODS can also be useful to support the data warehouse, such as when the warehouse does not maintain data to the lowest level of granularity to support user queries. It can also be used as an intermediate collection point for loading data into the data warehouse.

- n **Data Warehouse.** The data warehouse is a primary storehouse of historical data for an organization. Though it may be built out over time, its purpose is to be the primary online archive for an organization's data. Ideally, the warehouse is not constrained by business process or OLTP system boundaries. In order to avoid making changes to the schema whenever the structure or hierarchy of the business changes, it is advantageous to model it in a business-normalized form as opposed to a dimensional form.
- n **Data Mart.** A data mart is a storehouse of data, generally collected from a number of systems that make up a business process. For example, finance, customer service, warehouse management, etc. Data marts can be thought of as data warehouses with a limited scope. They are often created to satisfy the reporting and analysis needs of a particular department. Since data marts maintain a collection of data that are used for analysis purposes, they often follow dimensional modeling schemes.
- n **Cubes.** Cubes are described in [Chapter 2.2.2](#) as multi-dimensional representations of business data. Their purpose is to enable rapid analytical data operations. Cubes may contain any number of dimensions as well as measures that pertain to intersections of dimensions at various levels.

2.3.2 Conforming Data Mart Approach

This approach to data warehousing, depicted in [Figure 2–5](#), is based on the use of data marts with conformed dimensions. It has been described by Ralph Kimball and the Kimball Group as a "bus architecture" for data marts. This approach is intended to produce results quickly by dividing the scope of the warehouse effort into several subject areas.

Figure 2–5 Conforming Data Marts



Each subject area can relate to a business process, such as sales, support, operations, etc. Data pertaining to a subject area are collected from operational systems into a temporary staging area for cleansing. Once cleansed, the data are loaded into a data mart following dimensional modeling techniques.

Data marts can be provisioned and maintained at departmental levels, as opposed to creating a single enterprise-wide data warehouse. This reduces the effort required to prioritize, approve, and fund the project and allows each department to advance their BA program at their own pace.

The use of dimensional models makes data easily consumable by a wide variety of BA tools and applications. Tools are generally capable of interpreting the dimensions and provide a platform for analysis without the need for software programming and DBA expertise. This approach builds the data warehouse structure to maximize usage efficiency while minimizing the scope of data collection.

Since the analysis needs of an organization will span multiple subject areas, the architecture must provide the ability to perform queries across data marts. Therefore it is critically important to ensure that data is compatible across subject area data models. Conformed dimensions are

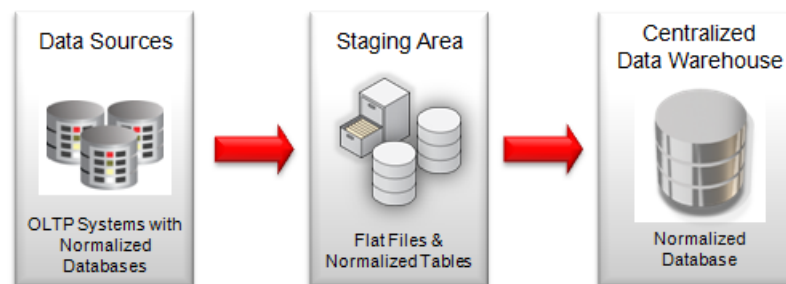
required for the success of a federated approach. Unfortunately, since data are being managed by separate data mart DBMS systems, conformance is often a matter of proper governance, (people and process), as opposed to technology.

In addition to the challenges of defining, implementing, and maintaining conformed dimensions across disparate systems, the federated approach has certain drawbacks due to its heavy reliance on dimensional models. Dimensional models are well suited to navigation but are not the most effective way to model business data for historical purposes. For instance, over time as changes occur to the business, dimensions will need to change, and likewise business entities and their attributes will also change. The dimensional model can remain clean and simple if it represents either the current hierarchical state or a previous state, but can become quite convoluted if it tries to represent different versions of hierarchies over time.

2.3.3 Centralized Data Warehouse Approach

In contrast to the federated approach, the centralized data warehouse, depicted in [Figure 2–6](#), offers a single point of historical data management. A centralized (enterprise) data warehouse is a vital component of Bill Inmon's original data warehouse architecture. This approach is considered "top-down" because it attempts to view data holistically across the entire organization. Like the conforming data mart approach, it can be implemented incrementally, one subject area or business process at a time. However, rather than using separate data marts to provide a historical record for each subject area, all subject areas are managed in a single data warehouse.

Figure 2–6 Centralized Data Warehouse



Data are collected from operational systems into a temporary staging area where cleansing can occur. Once all related data are collected and cleansed, the warehouse is updated. Updating can occur in small "drip-feed" increments (preferred) or larger batch-feed loads. Data history and versioning concerns are handled in the data warehouse.

The data warehouse is designed using relational schema patterns that easily allow for changes to business objects and hierarchies. This allows it to evolve in support of business changes.

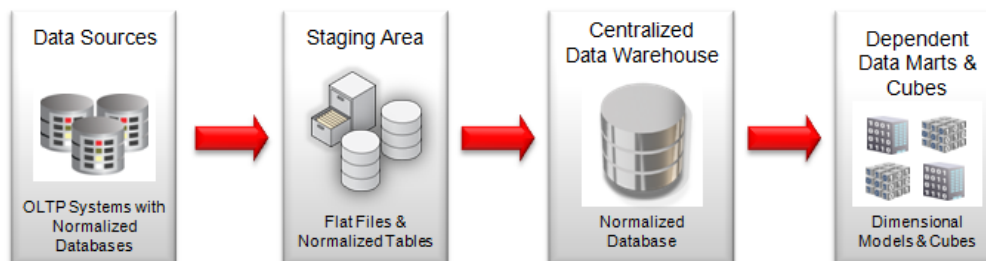
The use of a single DBMS system helps to promote consistency of structured data by eliminating the need to manually enforce conformity across multiple data marts. It also helps to reduce complexity by eliminating many of the integration concerns that can be attributed to federated data.

Although the consolidated data warehouse is superior in terms of historical data management, it is not nearly as easy to use for business analytics. Lacking dimensional models, the warehouse is not easy to navigate. Third party analysis tools designed for dimensional models and cubes are unable to work with these models. Database and software development expertise is often required in order to produce even the most basic analysis, dashboards, and reports. For these reasons, the centralized data warehouse is often combined with data marts and cubes in order to support business analytics.

2.3.4 Hub and Spoke Approach

The intent of a hub and spoke approach is to combine the historical data management benefits of the consolidated approach with the access and navigation benefits of dimensionally modeled data marts. This hybrid approach, depicted in [Figure 2-7](#), is presented by *ORA Information Management* as the recommended architecture for a structured data warehouse.

Figure 2-7 Hub and Spoke Data Warehouse Pattern



The key concept of this approach is the separation of concerns between historical data management and support for analytics. In short, it leverages the consolidated data warehouse to manage historical data in a business process-neutral data model, and exposes data for analysis operations via dimensional models and cubes. The data warehouse maintains history, while the dimensional models and cubes change over time to reflect current business processes and hierarchies.

Another key aspect to the hub and spoke approach is that the data marts are built from data in the centralized data warehouse. Although their schemas are modeled differently (dimensionally), they are dependent upon the warehouse as a source of data. By operating as dependent data marts they can more easily be conformed and their facts will be based on a shared view of reality.

The hub and spoke approach is expanded upon in [Section 3.4, "Logical Data Warehouse Conceptual View"](#) and in the *ORA Business Analytics Infrastructure* document. For more information on data warehousing, data movement, data quality, and master data management, please consult *ORA Information Management*.

2.4 Big Data, Fast Data, & Analytics

Gartner defines Big Data as "high volume, velocity and variety information assets that demand cost-effective, innovative forms of information processing for enhanced insight and decision making"¹. While this definition avoids any specifics on how high the volume, velocity, and variety is, it does infer a paradigm shift on the technologies required to handle it.

Previous information management and processing systems typically relied on relational databases and structured data modeling. The volume, velocity, and/or variety aspects of Big Data make this approach impractical. From an information management standpoint, Big Data tends to be handled with less structure. Records are no longer decomposed into well-defined fields that are normalized and indexed. Rather, records are treated as a single unit that may or may not have any descriptive metadata attached. Records tend to be persisted in distributed file structures or low cost NoSQL databases.

Given the lack of structure, Big Data doesn't lend itself well to traditional forms of analysis. Tools and applications that use SQL or MDX to query, process, and present relational or

¹ [Big Data Definition, Gartner IT Glossary](#)

dimensional data cannot introspect this less-structured form of data. Instead, analysis must be performed using tools that are designed for non-relational data.

Big Data can however be processed in a way that makes it compatible with structured data. A common technique is to filter and reduce the set of Big Data into a much smaller set of records that can be combined or associated with structured data. The intent is to extract what is considered useful, valuable, and identifiable. By being identifiable, data records have the ability to be related to, and correlated with, other forms of data. By being useful and valuable, data records are deemed worth the added cost of long term relational database management and storage. Once this correlation takes place, analysis can be performed across different types of data using a single set of tools.

Fast Data tends to be defined as a subset of Big Data that is of high velocity. It too demands special handling, but primarily in terms of a system load and performance. Fast Data is often associated with streaming data that needs to be handled in real-time. Therefore the architecture tends to be event-driven - capable of accepting a continuous data stream and taking action on specific patterns. It is also usually memory-resident since reading and writing to disk is not feasible on a record by record basis.

Big Data and Fast Data are further described in *ORA Information Management*. The components required to analyze these forms of data are presented in *ORA Business Analytics Infrastructure*.

2.5 Enterprise Performance Management

According to Gartner, Enterprise Performance Management (EPM) is the process of monitoring performance across the enterprise with the goal of improving business performance.² This set of capabilities enables the organization to integrate IT information and intelligence capabilities with strategic business performance activities such as business strategy and planning, financial management, and profitability management.

2.5.1 Business & Strategy Planning

EPM can be used to both help define business strategy and to measure its success. The latter helps determine where adjustments need to be made, which factor into the next strategy planning sessions. Strategy planning then becomes part of an iterative cycle that is based on performance data and is actively managed.

Depicted in [Figure 2-8](#), a typical performance management cycle consists of four phases: Strategy, Planning, Monitoring & Analysis, and Corrective Action.³

² Gartner, [Enterprise Performance Management Definition](#)

³ [Business Performance Management: One Truth](#), by Mark N. Frolick and Thilini R. Ariyachandra, *Information Systems Management*, Winter 2008

Figure 2–8 Performance Management Cycle

Strategy & Planning functions involve the definition of business strategy and objectives. In the Strategy phase, an organization will determine what it wants to achieve and how it will define and measure success. For example, the strategy may be to become a global supplier of widgets; expanding sales outside of the home country. Success is defined by generating a certain percentage of total sales from foreign markets. Strategy can draw on existing intelligence to help set meaningful, yet realistic goals.

Strategy can be represented in many forms. Among them, Strategy Maps, are widely used. Strategy Maps, evolved from the Balance Scorecard, capture corporate objectives across four perspectives: financial, customer, internal business, and learning & growth. Strategy Maps offer a means to describe strategy as a set of related objectives from which the business can plan a course of action.

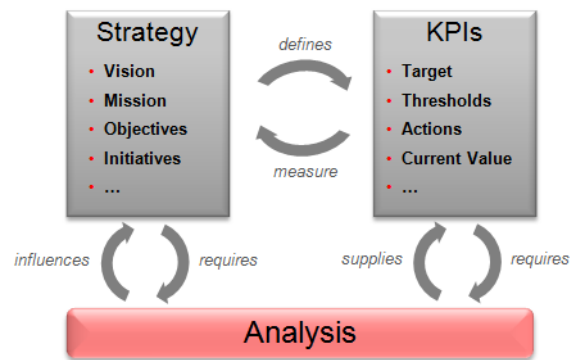
In the Planning Phase plans are devised to accomplish what is required in order to achieve the strategy. Groups within the organization determine their initiatives - what they need to accomplish, when it needs to be done, and who is responsible. Schedules are created and metrics are defined in order to measure each plan's success.

A linkage exists between strategy and plans in that the successful execution of plans should foster the achievement of the organization's strategy. The linkage can be modeled as a hierarchy where individual plans feed up into departmental plans, and so on, converging on a smaller set of high level plans that directly link to the organization's strategy. This form of model allows the business to drill down to the level of detail needed to understand precisely what needs to be done.

Once the plans are in place, the participants must chart and report their progress on achieving goals. Metrics, a.k.a. key performance indicators (KPIs), defined within the strategy and plans, must be monitored, analyzed, and reported. KPIs should be linked to accurate, well-defined, and timely operational data in order to ensure that proper measurements are being taken.

After proper evaluation of KPIs, the business may decide that corrective action must be taken in order to improve business performance. Actions may include the adjustment of business processes, business rules, governance processes, operational decisions, and/or a change to one or more affected business plans.

Figure 2–9 illustrates the link between analysis and business strategy. Strategy and KPIs are both defined by the business and refined as part of the performance management cycle. Strategy drives the definition of KPIs along with their characteristics: target values, threshold values, and actions that need to be taken when certain threshold values are reached. KPIs provide management an ongoing measure of success.

Figure 2–9 Analysis Supports Strategic Planning

Analysis influences strategy and supplies the current KPI information. As a result, strategic planning and KPIs impose requirements on the organization's business analytics capabilities. They help determine what information is collected, how it needs to be viewed and presented, and what types of actions the system must be able to perform in order to respond to current conditions.

2.5.2 Financial Management & Reporting

Financial planning, budgeting, and reporting are an integral part of the EPM strategy based on the need to accurately compile, manage, and report on financial information. Every publicly owned company needs to issue financial reports, and there are strict regulations regarding the accuracy and management of such information.

In order to avoid the chaos of management via disparate spreadsheets, EPM solutions are used to establish a single version of truth with respect to financial information. They gather source data from various systems across the enterprise and provide the official version of financial information that can be formatted and reported based on widely accepted accounting principles.

Financial management involves a number of activities which include:

- Financial and operational planning processes - requires a framework to integrate financial and operational planning processes and models. It links financial plans and metrics to operational plans in order to support decision making and provide insight on how business operations impact financials.
- Workforce planning. Planning for head count, salary, and compensation across the enterprise. This capability links workforce planning to financial planning in order to account for compensation in financial plans.
- Capital asset planning. Planning for assets, maintenance, transfers, and depreciation, with linkage to assess impacts on expenses and finances.
- Managing margins and cost of goods sold (COGS). The architecture supports detailed cost modeling based on operational drivers and assumptions.
- Management of period-end close activities. Processes include ledger and sub-ledger close, financial consolidation, 10K/10Q creation, reconciliations, etc. The architecture centralizes management of all period-end close activities using dynamic interactive dashboards. Users can also manage tasks and track status using notifications and alerts.
- Financial consolidation and reporting. Formal reports are frequently used to communicate financial information about the organization to outside parties. Given the strict need for accuracy, driven in part by Sarbanes-Oxley financial reporting standards, financial statements must be based on an accurate aggregation of financial data. The architecture supports this via information aggregation and governance capabilities as well as financial

reporting tools. Financial reports can be linked together to create consistency across them as well as to maintain a group of reports (e.g. Blue Book or Controller's Book). Reports may also be output as XBRL documents for filing with agencies such as the U.S. Securities and Exchange Commission.

2.5.3 Profitability Management

Profitability management is a capability that enables organizations to better understand the profitability and cost drivers within their business. It provides users with the visibility to improve resource alignment, increase margins and drive profitability.

This capability helps organizations identify sources and attributes of profitability via multi-dimensional analysis. It also includes scenario modeling to help improve profitability-based decisions.

2.6 Pervasive Intelligence

Traditionally, intelligence solutions have been designed to provide insight to the business in a standalone manner. That is, managers are given access to reports and KPIs which can be used for analysis and planning outside of the normal flow of business processes. Knowledge workers, in general, would need to gather information to make informed decisions as a separate activity. Although the necessary information might be readily available, it is not integrated into the normal flow of a process.

The natural evolution of BA involves the integration of intelligence into everyday processes and activities. Intelligence becomes pervasive throughout IT, available where and when it is needed. It can take the form of automated decisions, e.g. real-time business rules, automated recommendations, or simply context-based heads-up displays that provide a decision maker with all the information they need at the time a decision needs to be made.

Pervasive intelligence involves integrating information, charts, graphs, etc. into applications, business processes, and portals in a manner that makes IT more effective as well as more efficient. It extends BA from the realm of managers and executives to all types of knowledge workers in the organization. For example:

- A customer service representative can be made aware of opportunities to up-sell or cross-sell based on recent customer purchases.
- A customer facing portal can target certain advertisements based on previous purchases and/or recent browser activity.
- Risk analysis can be based on a comparison of current and recent transactions with historical averages and trends.
- Inventory levels of certain types of goods may be automatically adjusted based on weather forecasts.
- Shipping methods may be influenced by fluctuations in shipping prices and dynamic inventory models.

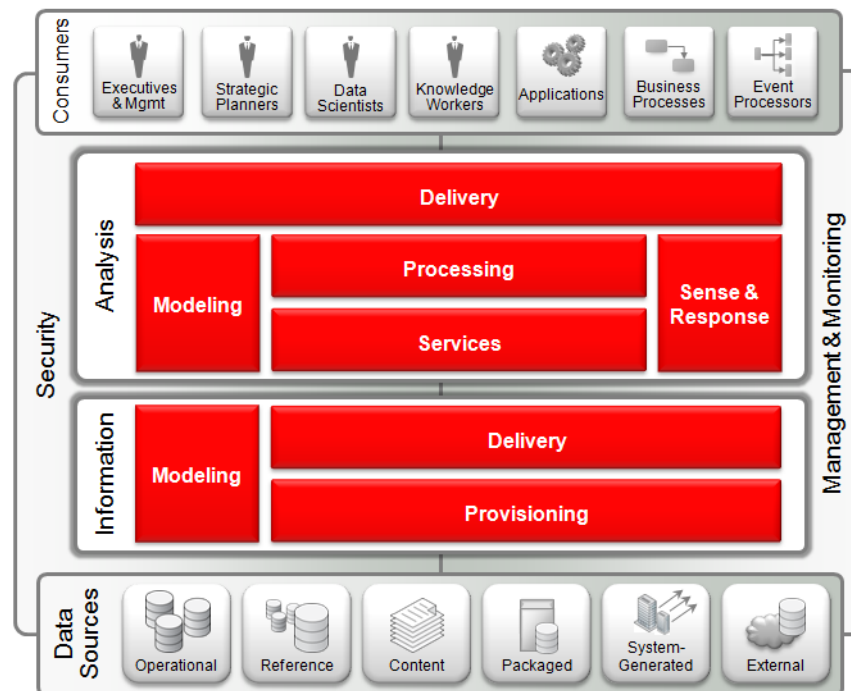
Conceptual View

This chapter presents a conceptual view of the business analytics architecture. It consists of both a high level view and a more detailed view that position BA within the larger realm of IT. It also describes the capabilities of the architecture and how they map to layers of the architecture.

3.1 Overview

As illustrated in [Figure 3–1](#), the business analytics architecture consists of a set of analysis capabilities that leverage business information. Analysis and information are the primary components of the architecture. They draw on all the various sources of data across the enterprise, and they produce intelligence for many types of consumers. All layers of the architecture are protected by security architecture and are supported by management and monitoring architecture.

Figure 3–1 High Level Conceptual View



Information is provisioned in various forms and for various consumers. It is not tied to a specific solution or consumer, rather it is designed for broader use and multiple purposes. It represents an aggregation or federation of data that has been organized in a way that best

represents business constructs. The information layer includes modeling, provisioning, and delivery components.

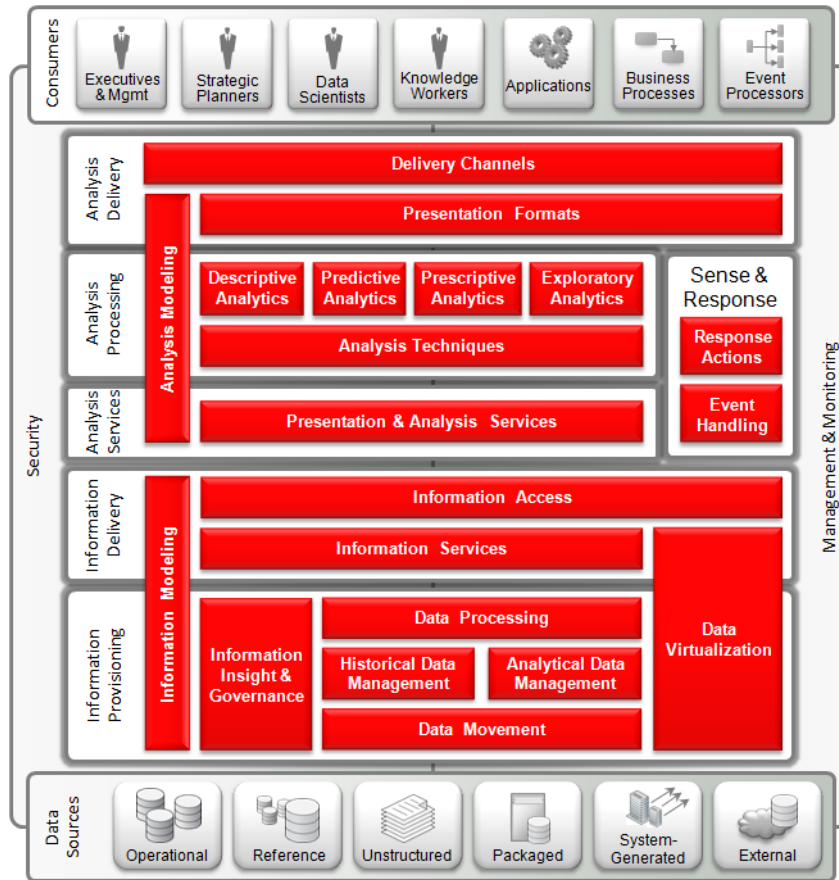
The analysis layer leverages the information layer to produce accurate and meaningful intelligence. It adds reusable services that organize, manipulate, and present information in ways that are insightful to the user. It includes processing that enables the discovery of business value and insight from data, as well as the ability to simulate and predict trends. It delivers insight in various forms depending on the type of consumer and access device. It functions as the mechanism for users to interact with information as well as a means to notify consumers when conditions or events occur and take action based on those events. The analysis layer also provides a means to define, model, and reuse analysis artifacts.

Consumers include executives, managers, and planners, which represent traditional users of BA. It includes data scientists to represent the types of users that perform advanced discovery and predictive modeling activities. It also includes a general category labeled knowledge workers, which refers to any type of user that can benefit from the insight that BA offers but might not actively define or model BA constructs (dashboards, reports, etc.). In addition, applications, business processes, and event processors are depicted as programmatic consumers of BA. These programmatic consumers may invoke BA services or may in turn be invoked when events or conditions occur.

The ORA Business Analytics architecture perspective builds upon several core ORA architecture constructs including information management, security, and management and monitoring. Please consult *ORA Information Management*, *ORA Security*, and *ORA Management & Monitoring* for more detail in these areas. This document will elaborate on the analysis layer as well as specific aspects of information that are most applicable to business analytics.

A more detailed view of the architecture is provided in [Figure 3–2](#). Here the information and analysis layers have been expanded to drill down one more level of detail. The following subsections describe each layer of the architecture in terms of the capabilities it provides.

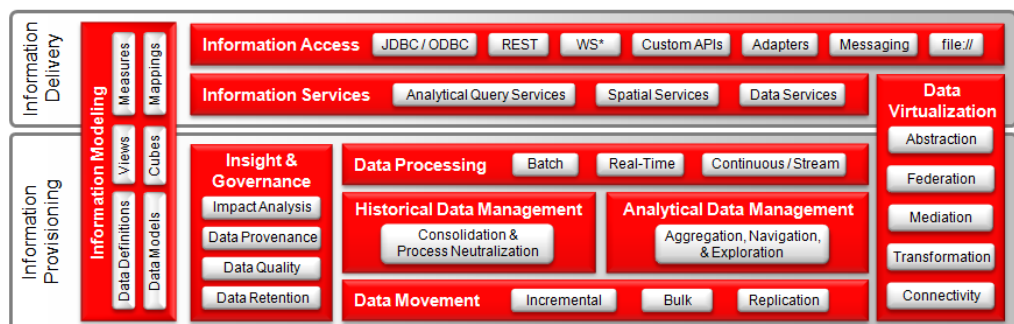
Figure 3–2 Detailed Conceptual View



3.2 Information Capabilities

The information portion of the architecture is composed of two high level layers - provisioning and delivery. These layers include capabilities pertaining to data movement, data processing, historical data management, analytical data management, insight & governance, data virtualization, information services, information access, and information modeling. Each is described in the following sections.

Figure 3–3 BA Information Architecture Capabilities



In addition to a brief introduction below, many of the information provisioning capabilities are further described in the Logical Data Warehouse Conceptual View section of this document.

3.2.1 Information Provisioning

The following architecture capabilities are provided by the Information Provisioning Layer.

3.2.1.1 Historical Data Management

Information provisioned for BA purposes often has one of two main qualities - it represents an accurate version of history, or it is provisioned in a way that best supports analytics. As previously described in "[Structured Data Warehouse Strategies](#)", these qualities can be at odds with each other. In this architecture they are defined as two separate components.

Historical Data Management is designed to accurately maintain a history of data. Data are gathered, consolidated, versioned, and normalized in a process-neutral form. This allows the structure to support change to business processes and hierarchies without the need for schema changes. Analytical performance and the ability for end users to navigate data in this form are of less concern. Versatility, consistency, depth, and accuracy are paramount.

3.2.1.2 Analytical Data Management

This capability of the architecture enables current and historical information to be easily navigated, explored, and efficiently queried and summarized. For structured data, it offers a multi-dimensional representation of data that is designed to reflect the current state of business process, hierarchies, and entities. For less structured data, analytical data management provides a persistence area for snapshots, samplings, or subsets of data that can be used for exploration or analysis that does not need to be executed across the entire history of data.

Analytical data management may also manage user-defined data such as forecasts, segments, attributes, and simulation variables that may factor into analysis reports, quotas, projections, etc. These data may be managed centrally, by an IT department, or locally, by end users.

3.2.1.3 Data Movement

Data movement is used to get data from an original data source into a target data store. For the purpose of BA, the target data store is considered to be within the realm of the Information Provisioning layer. Data movement may also be used to move data from one source within this layer to another. It is often associated with batch style extract, transform, and load (ETL/ELT) processes or MapReduce jobs. Data movement also includes incremental movement such as change data capture (CDC) as well as replication and synchronization technologies.

Data can be moved either physically or logically. Logical data movement refers to capabilities such as partition exchange, which has the effect of moving data simply by swapping definitions between two data partitions. Data appears to have moved even though no transmission has occurred.

3.2.1.4 Data Processing

Data processing represents a group of capabilities that pertain to the processing of data within the information management architecture. Processing, for the sake of analysis, may be used to transform data as it moves from one domain to another, i.e. as part of an ETL/ELT process or a MapReduce job. It may also refer to the execution of analysis algorithms on data within or "close to" the data persistence layer. For instance, a statistical analysis routine or a data mining algorithm may be executed within the database in order to avoid the overhead of transferring large amounts of data from the database to a separate application server or end user tool.

Data processing can occur in batch, real-time, and stream modes. Batch processing involves the processing of data at specific intervals, asynchronous to when results are needed. Batch jobs are often initiated by a scheduler or via human interaction. The results are maintained and made available to users, processes, or applications that need them. As such, they can be considered "long-lived", e.g. the results will not change until the next batch processing run occurs.

Batch processing is often used when working with very large data sets or when results need not reflect current, frequently changing conditions. For example, the results of a Web search can only be provided quickly by performing most, if not all, of the data processing beforehand. Data processing to populate OLAP cubes can also be done in batch mode, particularly when the underlying data is refreshed at specific intervals.

Real-time processing, in this context, refers to processing that occurs "on demand". Each request will be processed when it occurs and the results will be returned once processing completes. Real-time processing supports in-line analytical operations, such as recommendation engines, real-time pricing algorithms, and decision logic. Results are typically "short-lived". Although they may be cached for (immediate) future use, there is generally a direct correlation between when results are needed and when processing occurs.

Continuous / Stream processing is used when data needs to be processed constantly. A continuously running process operates on data as it is received. The process may be designed to take specific actions depending on data content or context. This form of stream processing is also referred to as event processing since the data in specific form or context can be considered an event. Users, applications, or other processes may subscribe to events in order to be notified when an event occurs.

Continuous / Stream processing can be used in conjunction with data movement as a means to collect data in (near) real time. It can be used to update analysis on constantly changing data, e.g. sensor data, weather data, and stock tickers. It can also be used as a means to trigger some form of action when an anomaly is detected.

3.2.1.5 Insight & Governance

Insight and governance encompasses a set of capabilities that either enrich the understanding of data, or govern the management of data. The capabilities include impact analysis, data provenance, data quality, and data retention.

Impact analysis provides the ability to see dependencies across the architecture. For instance, when a database table is changed or moved, it helps one understand the effect on other components of the architecture such as data movement processes, virtualization definitions, and information services.

Without impact analysis changes may result in a cascade of failures. This is particularly true when dependencies are not managed by a single technology. Each technology has its own 'understanding' of the architecture, and works on the basis that changes are not being made. Impact analysis requires active introspection or sharing of knowledge in a way that provides an end-to-end view of interdependencies.

Data provenance, sometimes called "lineage" or "pedigree", is the description of the origins of a piece of data and the process by which it arrived in a database¹. Provenance is particularly important when multiple copies, versions, or variations of similar data reside across the organization.

In order to ensure the accuracy of information produced via BA, one must know where the underlying information came from. Without provenance, BA might be derived from stale, incomplete, or inaccurate versions of data. Given the circuitous route that data may travel before being analyzed, the architecture must provide capabilities to understand provenance without reverse-engineering ETL processes, database procedures, and software programs.

A critical function of data governance is to ensure data quality. Data must be fit for purpose and accurately reflect the real world entities and facts for which they represent. Many factors can adversely affect the quality of data, including incomplete or inaccurate data entry, inconsistent representation of data fields across solutions, and duplication of data within or across solutions.

¹ Why and Where: A Characterization of Data Provenance, Peter Buneman, et al, University of Pennsylvania Departmental Papers. 1 Jan 2001.

Data quality fits into the architecture as processes, services, and procedures to correct these types of errors and arrive at a representative data set that meets the accuracy requirements for the organization's needs. Data quality involves several capabilities such as:

- n **Data Profiling** - examining and tagging data to determine and advertise its characteristics with respect to quality, usefulness, conformance to standards, etc.
- n **Data Validation** - validity checks may include type checking (numeric vs. alphanumeric), range checking, improper use of null fields, incorrect values where a set of valid values must be used, invalid entries (e.g. not a valid address), etc.
- n **Data Cleansing** - the act of fixing bad data. Cleansing is often performed as a set of automated steps that can correct common errors, along with manual intervention as needed.
- n **Data Standardization** - adjusting data formats and values so that records of diverse representations are all represented using common terms. For instance, standardizing the format of a telephone number or the units of measure.
- n **Data Enrichment** - adding additional value to existing data. Examples include fully completing address fields, supplying missing information, and tagging data with profiling metadata.
- n **Data Match / Merge** - matching and combining like records. This capability can be challenging when no fields are available with which to perform a match. It may involve aspects of validation, profiling, cleansing, standardization, and enrichment in order to deduce a possible match.
- n **Exception Management** - handling all data that cannot be automatically processed for quality. Data are often staged for manual processing as part of exception management.

Lastly, data retention refers to the period of time data should/must be made available. Organizations must take into account both the usefulness of data as well as government regulations pertaining to retention policies. Information management architecture supports the governance processes around data retention by providing metadata on the age of data as well as capabilities to purge data from the system based on age.

3.2.1.6 Data Virtualization

A cornerstone capability in an information provisioning architecture is data virtualization. It includes many important functions that enable information to be consumed without direct knowledge of, or connection to, underlying data sources.

Data virtualization acts as the intermediary between the information delivery layer and all physical sources of data that may contribute to the interaction. Data may come from sources outlined in the information provisioning layer, or from sources associated with specific point solutions across the organization.

In order for intelligence to be reliable, the underlying information must be reliable and consistent. Another goal of data virtualization is to deliver a "Single Version of the Truth" (SVoT). To achieve this, virtualization must operate on, and expose, a common business information model. This is a logical semantic data model that is backed by one or more physical data sources. The purpose of this model is to provide consumers with a clear semantic view of information irrespective of physical data model characteristics. Data virtualization is used to perform logical to physical mediation.

Data virtualization includes several capabilities such as:

- n **Abstraction & Mediation** - enables loose coupling between consumers and providers via an intermediary that abstracts location, storage structure, API, and technology differences.
- n **Transformation** - allows information to be consumed in a form that is different than how it is stored.

- n **Federation** - provides the ability to combine data from multiple sources into a single operation.
- n **Connectivity** - supports the various integration requirements necessary for efficient data access.

3.2.2 Information Delivery

The following architecture capabilities are provided by the Information Provisioning Layer.

3.2.2.1 Information Services

In order to best provide, manage, and advertize information to a diverse range of consumers, it is advantageous to deliver information as a service. Information Services logically encapsulate all underlying architectural concerns, e.g. provisioning and virtualization, and present information via a clearly defined interface.

Information as a service, at a minimum, offers the advantages of service-oriented integration, as described in *ORA Integration*. In addition, Information Services can be designed to meet the specifications of enterprise-class SOA services. This level of robustness enables them to be discovered and reused across the organization, and provides the level of management and autonomy needed to support performance requirements and quality of service agreements.

A fully-defined information management architecture will include many types of information services. For the purpose of BA, three general types of services have been listed: Analytical Query Services, Spatial Services, and Data Services.

Information Services (aka Data Services), as defined in *ORA SOA Foundation*, are used to access data from various sources using many different technologies, and present data in a business-friendly form. Data may originate in various databases, flat files, XML files, and legacy systems. Information Services offer a way to aggregate, transform, and synchronize data from multiple sources. This creates an abstraction between the users of data and the sources of data. Users of data do not need to be concerned with where data is stored, how it is stored, or how it is represented in its native form.

By definition, Information Services tend to represent simple create, read, update, and delete (CRUD) functions. They have no awareness of the context in which they are being used, which makes them quite generic and universally applicable. They also tend to lack the capability to introspect or manipulate data according to formulas, rules, relations, or hierarchical constructs. This creates a design trade-off between simplicity and robustness. A simple Information Service is easy to design and build, and more applicable to a large consumer base. In contrast, a more complex and robust form of Information Service can offer more powerful and complex features, however, it may take longer to build, it may not apply to as many consumers, and it may require more frequent revisions in order to maintain its feature set.

Spatial Services are a type of Information Service that pertain to geospatial data. They provide data for geographical and location-based purposes such as mapping, routing, and geocoding. The services can be used to query geospatial data and to query, create, delete, and update geographic features within the data. The Open Geospatial Consortium (OGC) has defined specifications for a standard set of Spatial Service interfaces. See [Section 4.1.3.1](#) for more information on the Web Feature Service (WFS) interface standard.

Analytical Query Services are a type of Information Service designed specifically for BA. In addition to virtualization features they are aware of dimensional constructs such as hierarchies and aggregations. This combination of features enables them to satisfy queries based on data from multiple physical sources, either standard normalized (via SQL) or multi-dimensional (via MDX). Analytical Query Services are also "aggregate-aware", which means that queries that involve aggregations or calculations can use tables of pre-aggregated values if they exist, or perform aggregations and calculations "on the fly".

Analytical Query Services can support operations that go beyond the realm of standard SQL queries. For example, to determine market share changes versus a year ago, a service may need to perform functions that involve row to row comparisons, time-based reporting, and derived measures (ranks, Ntiles, standard deviations, moving averages, etc.). Complex business measures such as these add value over standard Information Services in terms of capability, efficiency, and performance.

3.2.2.2 Information Access

Information is accessed using standards-based protocols and APIs, such as Open Database Connectivity (ODBC), Java Database Connectivity (JDBC), and Web Services (WS* and REST) interfaces. This permits the use of information and Information Services by many types of consumers, thus maximizing value while minimizing vendor lock-in.

Information can also be accessed using custom APIs, adapters, and lower level protocols. These access mechanisms can be useful for information that is stored in distributed files, legacy systems, or databases that do not support standards-based interfaces.

3.2.3 Information Modelling

Information modeling enables information sharing and consistency. It supports the definition of a common logical information model and associated semantics. This enables users to define measures, calculations, analysis, and reports that are based on well-defined business data, thereby driving consistency of analysis as well as efficiencies of reuse.

Information modeling applies to many capabilities of the information layer. Models are not only used to organize information, they are also used across the architecture for introspection and mapping between components and data sources. [Table 3–1](#) describes a number of ways that modeling can be applied across the information layer.

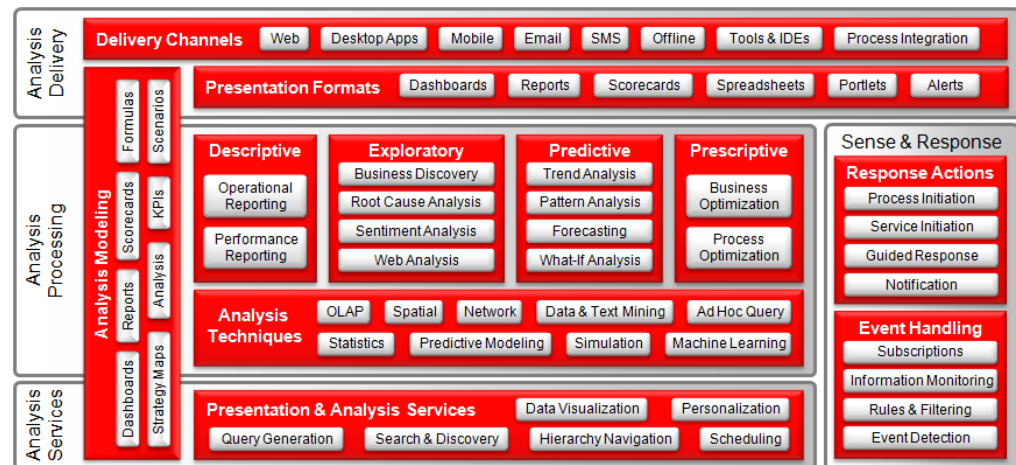
Table 3–1 Information Modeling

Architecture Component	Modeling Constructs
Data Movement	Data movement processes, transformations, mappings, metadata
Insight & Governance	Data definitions & profiles, relationships, data quality processes, rules, mappings, transformations, metadata
Historical Data Management	Data definitions, schemas, relationships, views, and metadata
Analytical Data Management	Data definitions, schemas, aggregations, views, cubes, metadata
Data Virtualization	Common information model, semantic definitions, queries, measures & calculations, physical to logical mappings, transformations, metadata
Information Services	Service models, metadata
Information Access	Interfaces

3.3 Analysis Capabilities

The analysis portion of the architecture, shown in [Figure 3–4](#), is composed of four layers: Services, Processing, Delivery, and Sense and Response. These layers include capabilities pertaining to presentation & analysis services, analysis techniques, descriptive analysis, predictive analysis, prescriptive analysis, exploratory analysis, presentation formats, delivery channels, event handling, response actions, and modeling. Each is described in the following sections.

Figure 3–4 Analysis Architecture Capabilities



3.3.1 Analysis Services

Presentation and Analysis Services transform information into a visual experience. They enable objects to be rendered in tabular or graphical formats that the end user can manipulate. They provide the ability to present a graphical representation of the information model that the user can navigate in order to perform functions such as querying and reporting. Presentation and Analysis Services ultimately translate user interactions into queries that can be fulfilled via the Information Services.

Presentation and Analysis Services leverage some of the analysis components that are represented by Analysis Modeling. Components include pre-defined charts, graphs and views, which can be searched and embedded into dashboards, reports, and applications, and reused across the organization.

This layer also provides a personalized experience for BA users. It acts as the home page, offering a common means to provide and secure access to analysis, with common tooling to support component modeling, sharing, and discovery (search). It also offers the capability to personalize content based on users' roles and preferences.

Presentation and Analysis Services act as a universal platform for various forms of intelligence gathering and analysis. They can support functions such as OLAP operations, reporting, enterprise performance management, simulation, forecasting, and data mining. They can also be used to pre-fetch information or run queries and reports based on user-defined schedules.

3.3.2 Analysis Processing

As the subject of Business Analytics has matured the definition and scope have coalesced around a generally accepted set of categories. These categories describe types of analytics that can be performed to benefit the business in different ways. The four categories of Business Analytics that are reflected by this reference architecture are:

- Descriptive Analytics - uses data to understand past and current performance. It aims to answer questions such as: what happened, what is happening, how many, how often, and why? This is the most common use of traditional business intelligence tools and applications. Descriptive Analytics is most often used to support decision making, performance monitoring, and planning.
- Predictive Analytics - uses data, tools, and mathematical techniques to determine what is most likely to happen in the future. It applies methods such as trend analysis, pattern matching, and simulation to base future predictions on current and historical data.

Businesses use this to target products, services, and marketing campaigns, and to set forecasts for sales and earnings.

- Prescriptive Analytics - applies mathematical analysis to optimize business processes and operations. It prescribes the best alternative to a specific problem. Prescriptive Analytics helps answer questions such as how can we achieve the best outcome?
- Exploratory Analytics - is used to investigate complex and varied data, whether structured or unstructured, for information discovery. This style of analysis is particularly useful when the questions aren't well formed or the value and shape of the data isn't well understood. It can be used to discover new business opportunities, to test hypothesis regarding cause and effect, and to perform advanced forms of root cause analysis.

These four categories of analytics rely on various analysis techniques. The techniques refer to the way in which information is organized and presented, e.g. OLAP, spatial, and network analysis, as well as the forms of processing that can be applied, e.g. statistical analysis, data mining, and simulation. In order to avoid instituting unnecessary restrictions, the architecture does not attempt to draw an explicit link between the four types of analytics and the different analysis techniques.

3.3.2.1 Analysis Techniques

Business analytics may require the analysis of many forms of information using many different techniques. This architecture describes several common techniques that represent an "example set". Each organization should determine which techniques are important to them and which are not. Additional forms, beyond this list, may also be required. Regardless of which techniques are embraced, the architecture should be capable of handling them in a consistent and reliable manner, ideally with the fewest number of technology implementations and component redundancies.

3.3.2.1.1 OLAP

OLAP interactions include the operations of slicing, dicing, pivoting, and rolling up and down sets of dimensional data, often managed or represented as a cube. See [Section 2.2.3, "OLAP Operations"](#) for a description of these operations. Users determine which dimensions and measures to include, and the system creates and populates the representative cube view. Presentation Services provide all the navigation, rendering, and back end information query capabilities. This form of analysis is frequently used for performance analysis, operational reporting, and planning. BA users can track progress of their objectives and perform time-based comparisons using current and historical data.

3.3.2.1.2 Spatial

Spatial analysis is used to portray data over physical dimensions. It enables end users to understand geospatial relationships and trends most efficiently. The architecture must support the processing and rendering of data in this manner. Spatial analysis can be used to gain insight on how geospatial properties affect business performance and how processes may be optimized to account for spatial properties.

3.3.2.1.3 Network

Network or graph analysis is used to understand the direct and indirect relationships (connections) between entities. It also helps to understand the implications of those relationships (or lack thereof). Network analysis can be used for several purposes such as:

- **Network optimization** - finding the shortest path or fewest hops from source to target destination. This information can be used for purposes such as supply chain optimization (in a physical/spatial application) or navigating relationships (in a social/media application).

- n **Network robustness** - determining the robustness of a network by examining how it will perform under various failure scenarios. It helps determine if one or more entities are removed will the network survive via alternate connections or will some entities be isolated from the larger network.
- n **Social network analysis** - navigating relationships between people, organizations, groups, etc. Organizations are beginning to leverage information that is gathered from social web sites such as Facebook, LinkedIn, and Twitter. Relationships between people, often with similar likes and characteristics, can represent a target customer base for a particular product. Likewise, relationships can be used to understand social behavior such as the emergence of trends or spread of diseases.
- n **Link analysis** - understanding the links (associations and types) between entities. For example, people can be linked via common addresses, familial ties, phone calls, business transactions, etc. Link analysis is quite useful in investigative analysis to determine if a relationship between entities exists and how strong that relationship might be.

3.3.2.1.4 Data Mining

While several techniques offer the ability to perform analysis by observing data in different ways, organizations may also want to detect patterns in data programmatically through bulk data analysis.

Data Mining is a term used to describe an automated form of analysis where algorithms, statistics, and/or artificial intelligence are applied to large sets of data in an attempt to uncover hidden patterns or anomalies. The user defines a data model and writes (or selects) algorithms to apply. The data model is usually a subset of historical or analytical data that pertains to the subject a user wants to examine. Algorithms are then applied and the results are analyzed. Results can be used to help drive business strategy, or they can be applied directly to current business processes, workflows, and applications. Such techniques can help profile consumer buying habits, predict peaks in service usage, or spot patterns of fraud or waste.

Data Mining processes are generally performed on data within large data stores since they tend to contain the breadth and depth (history) of data necessary to produce meaningful results. The processes involve several established methods including:

- n **Classification** - mapping of entities into one of several pre-define classes. This can be useful to classify customers based on a known set of attributes. This can help predict customer behavior.
- n **Regression** - mapping of measured entities against a control variable to measure the accuracy of the control. A regression can be used to rate the quality of an assumption or business rule, or to predict or estimate the control value.
- n **Attribute Importance** - finding the attributes that have the most influence on a target attribute. This helps to improve decision making by focusing attention to the factors that are most critical.
- n **Anomaly Detection** - trains itself on "normal cases" and scores unusual cases on their probability. This is useful in finding rare events and/or fraudulent activity.
- n **Clustering** - identifying natural groupings of entities based on the "relative closeness" of their relationships and overall cluster density. This can be used to better understand or profile (segment) customers, suppliers, products, etc.
- n **Association Rules** - looking for common associations between entities e.g. market basket analysis and discovery of frequently co-occurring items in a shopping cart.
- n **Feature Extraction** - creating new attributes that represent the same information using fewer attributes.

- n **Summarization** - producing a relatively small set of data to summarize a larger data set, e.g. mean, median, and standard deviation.

The architecture supports data mining by providing tools to model data, define and select algorithms, and run analysis. Given the large amount of data that can be included in a model, the architecture operates on a two-tier approach. One tier consists of end user tools to perform modeling, algorithm design, and analysis. The other consists of a database that is capable of natively performing all of the mining operations.

3.3.2.1.5 Text Mining

Text mining is similar to data mining, however it is performed on unstructured text documents such as blogs, Word documents, comments fields, and emails. The intent is to support objectives such as sentiment analysis and social analysis on data that has not been captured in a consistent or structured format.

The unstructured nature makes text mining far more difficult to perform. Consequently, text mining involves several steps used to overcome the nature of freeform written language. The steps may include:

- n **Filtering** - to extract text from various file formats into a common workable output format.
- n **Sectioning** - to preserve the hierarchy of text in a manner that is searchable.
- n **Lexical processing** - to eliminate the uniqueness of text caused by low value words or punctuation.
- n **Personalization** - to understand the context of a word based on the user or author, (e.g. socket as an electrical receptacle vs. socket as form of wrench).
- n **Semantic mapping** - to link words or phrases with similar meaning.
- n **Progressive query relaxation** - to allow fuzzy matches in a manner that is most likely to produce meaningful results.
- n **Dynamic rendering** - to adjust the way results are presented based on the ability to categorize, combine, or map results together.

The architecture for text mining may combine capabilities of text processing, as listed above, with methods of data mining.

3.3.2.1.6 Machine Learning

Similar to data mining, machine learning also applies algorithms and statistical analysis to data. The principle difference with machine learning is the intent to develop a learned algorithm, based on sample training data, which can be applied to influence or predict future behavior.

With machine learning, the emphasis is on accuracy. Learning can be a process of refinement where previous findings are carried forward and new data points are continually applied. Once data has been applied to the learning algorithm, it is no longer required for processing. This differs from data mining, where an emphasis on scalability and efficiency must be maintained in order to process large data sets.

Common methods used for machine learning include:

- n **Supervised Learning.** Similar to classification in data mining, this method uses training data that has been labeled in such a way that the system will understand the context of the classifications. For example, a voice recognition system will be able to associate voice patterns that are labeled with words or phrases. It can classify similar patterns with a specific meaning.
- n **Unsupervised Learning.** Much like clustering in data mining, this method must group similar objects together without the use of a label. Objects are grouped according to

characteristics, which may or may not produce accurate and distinct groupings. Also, since there is no label, the system will not know the semantic meaning of the groups.

- n **Active Learning.** This method involves users in the learning process. Users are asked to label examples in order to refine the learning process. The goal is to create a model that is as accurate as possible given the amount of labeling that can be obtained from users.

Machine learning can apply these algorithms using a number of specialized techniques, such as:

- n **Neural networks** - using a network of independent processors to produce conclusions that are collectively "weighed and measured" in order to produce an outcome or decision.
- n **Genetic programming** - evaluating a population of independent programs and optimizing their usage based on which is the most fit for purpose for given tasks.
- n **Inductive learning programming** - attempting to define logic that encapsulates all positive results and no negatives results from a given data set.

The architecture to support machine learning should combine tools that offer machine learning techniques and algorithms with data that can act as a representative sample set.

3.3.2.1.7 Statistics

Statistics can be defined as the science of collecting, analyzing, presenting, and interpreting data.² It involves aspects of information management such as data acquisition, cleansing, classification, and ranking (for quality purposes), as well as appropriate mathematical processing and effective visual rendering (i.e. table or graph style, scale, and density).

Common Statistical methods include:

- n Summarization and frequency distribution, such as computing the sum of expense items by expense type and graphing the results by type in a bar chart.
- n Computing various descriptive measures, such as mean, median, mode, percentiles, quartiles, standard deviation, and variance.
- n Calculating probabilities, e.g. numerical measures of the likelihood of an occurrence.
- n Regression, hypothesis testing
- n Inferencing: drawing conclusions about a large set of data based on analysis of a smaller set.

Statistical analysis can be used to directly support all forms of analytics (e.g. descriptive, predictive, exploratory, and prescriptive), as well as to support other analysis techniques, such as data mining, text mining, and machine learning.

The architecture supports statistical analysis very much the way it supports data mining. It provides an environment to define statistical programs and the ability to efficiently run those programs on large data sets.

3.3.2.1.8 Predictive Modeling & Simulation

Modeling and simulation are used to support predictive analytics. They combine aspects of data mining and statistical analysis, used to understand and model attribute relationships, with randomized simulation, i.e. Monte Carlo simulation, in order to better understand the probabilities of an outcome.

For example, a sales prediction or a cost estimate may be based on several factors (attributes) that are deemed most relevant to the outcome. Each attribute may be fixed or variable. Variable attributes may have characteristics that describe their variation, i.e. some may vary uniformly

² Anderson, Sweeney, and Williams, "Statistics for Business and Economics", © 1990 by West Publishing Co.

across a known spectrum while others cluster in a manner referred to as a bell curve. A model can be created to reflect the attributes and their variation characteristics. The model can then be used in a simulation that would randomly choose variable values based on expected variation characteristics. The result will reflect the probability of a range of outcomes, i.e. a probability curve.

Predictive modeling and simulation can play a large role in strategic planning, risk analysis, and other business use cases. The architecture includes tools designed for modeling and simulation.

3.3.2.1.9 Ad Hoc Query

An important technique to support all forms of analytics is ad hoc query. It enables analysts to drill down into descriptive data in order to understand how measures and statistics have been achieved, to investigate anomalies, to perform root cause analysis, and other investigative use cases that are not fully satisfied by other analysis techniques.

Ad hoc queries are queries that a user defines "on the fly" (versus those that have been pre-defined). It is an important capability from an analysis standpoint in that intelligence gathering often is a journey of discovery. It requires one to produce and execute a query, examine the results, and formulate another query based on what has been learned so far. The new query may be a variation that includes new data, different filters, and more or less detail.

The architecture supports this capability in two ways. First, by leveraging Presentation Services that enable users to visually navigate the information model and select columns, tables, fields, and constraints to define a query. And second, by allowing users to manually enter queries for execution. Both methods result in a query against the same information model.

In order to provide ad hoc query in a useful way, the architecture must be able to query and search all sources and forms of data, e.g. structured, semi-structured, and unstructured data. Otherwise the analysis could end when the trail leads to another data source or from structured to semi-structured data. The information management system must provide a way to correlate information of different types into a unified, "faceted" data model. In addition, virtualization technologies can be used to federate queries across data sources such that the end user does not need to query different sources or even know where specific forms of data are physically being managed.

Performance of the query must be acceptable with the form of analysis being undertaken. Some analysis may be performed incrementally over a long period of time. In these cases longer running batch query processes could suffice. However, if multiple queries are done interactively, in succession, i.e. following a trail of evidence to a conclusion, then the query must perform quickly enough as to not interrupt the train of thought. A popular buzzword for this is "speed of thought" queries, where query results are provided as quickly as the end user can formulate the next query.

3.3.2.2 Descriptive Analytics

Descriptive Analytics uses data to understand past and current performance. It aims to answer questions such as: what happened, what is happening, how many, how often, and why? It represents a collection of capabilities that are most commonly associated with traditional business intelligence. It includes operational reporting, performance reporting, and other functions that provide business insight and enhance decision support.

Descriptive Analytics often relies on analysis techniques such as OLAP, statistics, and ad hoc query, (although any techniques can be used). They offer the greatest insight to answer common business problems using tools and skills that are readily available and easy to learn.

The architecture supports Descriptive Analytics via a number of means. It provides reliable information and virtualization, (via the Information Management layers), that enables analysis to be performed across many data sources without requiring knowledge about how and where

information is stored. It offers Presentation Services that provide visualization and navigation capabilities. And, it provides a modeling layer that allows analysis and KPIs to be defined and shared by multiple users.

3.3.2.2.1 Operational & Performance Reporting

Operational and performance reporting includes both the daily "fact-based" analysis of what is happening as well as the "opinion-based" analysis of how well the business is performing. Both are vital to running a company. Operational reporting provides facts such as number of units sold, mean time to repair, and customer satisfaction metrics. The facts are often provided by a single source system and the architecture must be capable of providing these data points accurately and timely. The definition of data points and mapping to physical data sources must be properly managed in order to produce consistent and useful results.

Performance reporting describes the type of analysis used to form opinions of how well the business is operating. For example, meeting or missing various targets, operating under budget, etc. It includes most of the reporting capabilities associated with EPM, to support strategy, financial, and profitability management. Performance reporting will often measure data points from multiple systems against KPI targets that have been defined to represent optimal performance conditions.

The architecture supports reporting for multiple purposes, from ad hoc intelligence capture to formal report publishing. Ad hoc reporting is built upon the common information model as well as common semantic business object definitions, and visual Presentation Services. This set of capabilities enables users to define and select business objects such as "Gross Revenue", "Net Revenue", and "Net Revenue Rank". Objects are combined to create and define calculations, such as "revenue for each region during the current month". Presentation Services translate calculations into queries, which are performed using Information Services. Reports, calculations, and business objects can be saved, shared, and personalized across the user base.

Formal reports are defined in much the same way, except greater attention is placed on quality formatting and visual appeal. The tooling offers more control over editing and layouts in order to achieve a professional look and feel. Tooling options may include custom editors, document editors, and spreadsheets. Reports are generated using the same underlying architecture capabilities as ad hoc reporting, including Presentation Services, Information Services, and the BI object catalog.

3.3.2.3 Exploratory Analytics

Exploratory Analytics applies ad hoc exploratory activities to discover why, or if, something is happening. It can be used for new business discovery, root cause analysis, and to influence business strategy by supporting or refuting various hypothesis regarding cause and effect.

The architecture includes capabilities that enable the business to perform investigations to uncover hidden opportunities. These capabilities make it possible to analyze data to develop new potential revenue streams or to gain insight that will improve existing processes and activities. Investigation generally involves the analysis, filtering, distillation, classification, and correlation of data in order to create a useful form of knowledge or understanding.

Since activities such as data and text mining are very data-intensive, processing is best done where the data resides. This avoids performance problems associated with extracting data from the database (or other persistence mechanism) for processing, or executing an extremely large number of queries across architectural layers.

In general, exploration capabilities often will not leverage intermediate layers of the architecture such as data virtualization. In addition, many forms of exploration will take place on data that has not been incorporated into the data models that are managed and governed for enterprise use. In fact, one purpose for doing investigation might be to uncover applications for

data with sufficient value that will make it worth the effort to define, model, virtualize, service-enable, and govern said data.

3.3.2.3.1 Business Discovery

A business may perform some form of exploratory analytics in order to discover or justify new business opportunities. The business may have an idea for a new product or service and need data to determine the likelihood of achieving a favorable ROI. If the opportunity is related to existing products and services, then historical data may be examined for trends or evidence to support or reject the new venture. Otherwise, if the new opportunity is dissimilar to existing products and services, then data from outside the organization may need to be obtained and examined.

Business discovery can involve many analysis techniques depending on the type of opportunity. For example, if a business is considering opening a new location or a new type of store in another geographical area, then analysis may take several forms. Statistics and data mining may be used to calculate sales expectations. Network and spatial analysis may factor into the estimate based on location, business connections, and supply chain variables. Text mining may be beneficial to help understand existing customer perceptions. The final outcome may reflect a weighted estimation of many forms of analysis.

3.3.2.3.2 Root Cause Analysis

Root cause analysis (RCA) is a process of analyzing data, processes, and events in order to understand precisely why something is happening rather than reacting to the outcome or symptoms. The objective is to identify the source(s) of a problem, or anomaly, and recommend solutions that will prevent recurrence at lowest cost, and in the simplest way.

There are many forms of RCA including:

- n Failure analysis
- n Quality control / production-based analysis
- n Accident & safety-based analysis
- n Process-based analysis

The RCA process is defined as a series of steps that begin with problem definition and end with a solution implementation. The exact number of steps varies across sources of RCA information, but all processes tend to follow this common theme:³

1. Identify the problem. Define what is happening and what symptoms are present.
2. Collect data about the problem, e.g. proof that the problem exists, how long it has existed, impacts, locations, conditions, etc.
3. Identify all possible causal factors.
4. Trace causal factors back to the root cause(s) and why they occurred.
5. Recommend and implement a solution.

Historical data plays a key role in RCA. It contains the facts, at the lowest level possible, to support the data collection portion of this process. RCA is generally not a prescriptive process in that an investigation can go in different directions as new evidence comes to light. Ad hoc query across various forms of information is an important capability to facilitate such investigations. In addition, more advanced forms of analysis such as data mining and statistical analysis may be required in order to validate correlation and/or causality.

³ Root Cause Analysis - Problem Solving from MindTools.com

3.3.2.3.3 Sentiment Analysis

Sentiment analysis is an emerging trend that looks at ratings, comments, blogs, etc. to understand how a product or service is being perceived in the marketplace. It helps companies with product pricing, placement, advertising, and evolution strategies so they can be more competitive. The architecture for sentiment analysis often revolves around the processing of text data, e.g. text mining, since sentiment is typically expressed in written language.

3.3.2.3.4 Web Analysis

Given the amount of business conducted over the Internet, companies today can benefit from analyzing how customers interact with their Web sites. Web analysis represents the ability to track online activity, page clicks, product interest, referrals, and buying habits in order to measure the effectiveness of the online experience.

This information can be used to measure the ROI of advertising campaigns, influence product placements, and influence additions and modifications to the Web site. It can also drive efforts to optimize business processes in order to avoid circumstances where potential sales are abandoned before they are completed.

3.3.2.4 Predictive Analytics

Predictive Analytics is a set of capabilities that apply algorithms and user-defined models to current and historical data in order to more effectively predict future results. It bases predictions on techniques such as statistical analysis, data mining, and simulation. Businesses use this to target products, services, and marketing campaigns, and to set forecasts for sales and earnings.

Predictive Analytics is useful for both strategic and tactical decision making. It is used in strategic planning to generate financial and profitability forecasts. Users run simulations based on historical data, market trends, and current influences in order to realistically project results into the future. This information might be used to alter business strategy and processes in order to improve profitability and reduce costs. It can also be used in tactical decision making to learn how to optimize business rules, operational processes, marketing plans, etc.

The architecture to support predictive analytics includes specialized end user tools in addition to the information and intelligence layer capabilities previously mentioned in this chapter. User-defined modeling and simulation tools differ from ordinary BI tools in that they allow end users to create their own information models. The models are backed by common information services, however, in order to be flexible they are often created and maintained by the end user community. Models allow data to be altered to support simulations based on various assumptions and variables.

Optimization may be determined based on manual simulation runs, e.g., several iterations of simulating, analyzing, and adjusting data variables in order to produce optimum results. Tooling can also provide the capability to automate simulations and determine optimizations automatically.

3.3.2.4.1 Trend Analysis, What-If Analysis, & Forecasting

A simple form of predictive analysis involves projecting current trends. It offers predictions that project the current trend forward without any changes in dynamics that might otherwise change the current trajectory. Such analysis might be used to estimate performance for the immediate future before any operating conditions can be altered.

A higher form of analysis involves introducing variables and predicting results based on how those variables are adjusted. It is used to model alternate outcomes based on different combinations of input parameters, e.g. "what-if" analysis. This simulation makes it possible to set policies or define initiatives to maximize performance by making business adjustments according to the predictive models. Such analysis can be used to help define the strategic initiatives of an organization.

Predictive analysis that involves simulation, hence adding or altering data values, is often performed on a snapshot of historical data, where changes can be easily made and alteration will not compromise the integrity of the data warehouse. Organizations can attempt to create forecasts based on projecting current trends, by modeling simulations where various data points are purposefully altered, or by detecting hidden patterns that affect operations.

3.3.2.4.2 Pattern Analysis

Pattern analysis, (a term used here in support of predictive analytics), refers to the exploitation of previously discovered patterns for their predictive qualities. The discovery or recognition of patterns is considered an exploratory analysis endeavor. Once a pattern is detected, the business can use this knowledge to help predict future results and behaviors.

Pattern analysis supports predictive analytics by mathematically establishing a most likely outcome for a scenario based on a given set of input parameters. It differs from trend analysis in that instances of the pattern have no temporal qualities. E.g. the likelihood of an output does not change based on when it occurs. Further weighting would need to be performed in order to account for trends.

In addition to providing the most likely outcome, the pattern analysis algorithm may also calculate the probability of that outcome occurring. It includes a confidence factor to rate how well past events of a similar nature can be accurately classified to provide an answer. This is referred to as a probabilistic algorithm. The system may also provide several outcomes along with probabilities for each. Correspondingly, the system may abstain from making a prediction when the confidence factors are too low.

Pattern analysis, discovery, or recognition relies heavily on statistical analysis, data mining, and machine learning. Like other forms of predictive analytics it draws from historical data.

3.3.2.5 Prescriptive Analytics

Prescriptive Analytics applies mathematical analysis to optimize business processes and operations. It prescribes the best alternative to a specific problem. Prescriptive analytics helps answer questions such as "how can we achieve the best outcome?"

Capabilities pertaining to Prescriptive Analytics can be divided into two parts - analysis that produces insight into how optimization can occur, and analysis of how business processes are performed in order to apply optimizations.

For example, pattern recognition can be used to produce a probabilistic prediction of outcomes along with associated confidence factors. This was discussed in the previous section on Predictive Analytics. Once the results have been calculated, the business must determine how it will take action. The business must analyze its operations and processes and decide which need to be changed, and how the changes need to be implemented.

3.3.2.5.1 Business Optimization

Business optimization can be achieved by making operational or organizational changes to the business in response to analysis. Changes can occur at different levels of the business and in various granularities.

Changes to the way a business operates may be desired if analysis suggests that cost savings or new opportunities are likely. Cost saving may be achieved through means such as consolidation of resources or standardization of processes. Statistical analysis might yield an estimate on the costs and savings of different operating models and the best course of action to optimize the business.

Optimization can also occur at lower levels of the organization and for very narrowly focused concerns. For example, analysis can be used to determine where kiosks, drop boxes, or vending machines should be located in order to generate the most sales.

3.3.2.5.2 Process Optimization

Optimization may be done to both manual and automated processes. Manual processes can be optimized by informing the participants in the process of how, when, and why changes need to be made. For instance, spatial or graph analysis might determine the best shipping options or delivery routes. Changes can be implemented by informing the persons involved in these activities.

Analysis may also be supplied to guide users toward the best course of action. For example, an employee travel Web site can provide information on the pricing and location of hotels for a travel destination. It can be used to either encourage or force the user to choose from a set of "preferred" options. Likewise, data pertaining to the likelihood of flight delays for a list of flights might be considered when one has a tight travel schedule.

Automated processes can be optimized via dynamic business rules or data-driven process decision logic. For example, a stock trading algorithm can rely on certain variables to determine when or how a trade is initiated. Ongoing statistical analysis can be used to calculate values for these variables, which in turn dynamically alter the trading process.

The steps, tasks, and decisions within a process can be optimized as well. Machine learning can be used to determine the best way to achieve a result, i.e. which course of action statistically yields the best results. This can factor into how a business process is defined or re-defined.

3.3.3 Sense and Response

Sense and Response is represented by two sets of capabilities: Event Handling and Response Actions.

3.3.3.1 Event Handling

Event handling represents a collection of capabilities that enable the out-of-band detection and handling of conditions or events. It allows users to determine what conditions they are to be alerted of, and how they are to be notified. It also enables analysis criteria to automatically trigger business processes and/or affect the rules by which they operate.

Event handling starts with the definition of conditions that are important to the business. KPIs and other important data can be measured on a scheduled basis in order to quantify current business conditions. When measurements exceed predefined tolerances, then the system automatically initiates the defined [Response Actions](#).

In addition to scheduled measurements, the architecture includes the capability to monitor real-time information feeds it receives from various sources. Sources may include operational systems, syndicated data feeds, message queues, flat files, etc. The data are stored, cached, analyzed, and reported on, based on predefined rules. The results can be pushed out to users as live display reports, or sent via channels such as SMS and email.

This monitoring feature of the architecture is often referred to a business activity monitoring (BAM) since the information being monitored and analyzed generally pertains to business level activities, (as opposed to operational monitoring which focuses on technical matters). The audience for reports and notifications is a business audience that needs to be informed of current business conditions as well as exceptions that may occur. BAM provides insight that is up to the minute, which helps businesses react to issues faster than if they had to wait for daily, weekly, or monthly reports. Since the information requirements are current, the information feeds tend to originate from operational sources as opposed to historic or dimensional sources.

Organizations that have embraced Event-Driven Architecture (EDA) will want to leverage those capabilities in concert with the BA capabilities. The architecture supports the consumption of (subscription to) events as a data feed to its BAM capability. This provides a link between events from EDA, BAM, and the information analysis and notification

capabilities of BA. Events can trigger the collection of content for dashboards, displays, and user notifications.

BA can also act as a publisher of information to event processors. This enables intelligence to factor into complex events, and to be pushed to all types of consumers that the EDA system is able to support. Analysis data that is fed into EDA can come from several layers of the BA or information management architecture. For instance, it can originate from a data movement (i.e. ETL) process, a trigger from the data warehouse, a Big Data feed, or the event handling layer, to name a few. This allows events to be based on data of various degrees of processing, composition, and refinement.

The architecture also includes subscription and notification capabilities that allow users to subscribe to events and indicate the types of alerts they will receive, e.g. email, SMS, etc. Events, in this case, may or may not be synonymous with events defined within an event-driven architecture. As described in *ORA EDA Foundation*, events may have a formal definition and representation. The BA architecture may leverage this standardization, or may evolve independently. There is an obvious synergy between these two capabilities with apparent overlaps in terms of publish/subscribe and subscription/notification. However, the BA implementation is often geared towards business users such that end users can define and manage their own events. Whereas EDA event definition, subscription, and implementation may be more IT-managed, and may not offer such end user versatility.

Another aspect of event handling is the ability to adjust business rules based on analysis information. It is a sensory capability that does not require human notification or intervention - an autopilot of sorts. Current state information is queried by a rules engine, which makes decisions based on predefined business rules. The rules engine can draw information from operational systems, analytical data stores, or persisted BAM and event data. Rules can also factor contextual data into a decision. Process behavior is automatically adjusted to account for trends and events in real time.

3.3.3.2 Response Actions

Response actions are various means to close the loop between analysis and intelligent action. These capabilities make intelligence actionable by connecting BA architecture to the rest of the business.

The obvious ways to accomplish this include the initiation of business process and services as the result of [Event Handling](#). Details of what needs to happen and how it should happen are contained within the processes or services. The BA architecture invokes them when conditions are right. For instance, if the stock of an item drops below a certain threshold, an ordering process can be initiated. The ordering process is configured to perform all the steps necessary to place an order including how many items to order and from where the order should be placed. Analytics may play a role by factoring in variables such as normal sales figures for this store over time, optimum stocking levels, etc. These data may be included in the context of process initiation or factored into business rules that the ordering process uses.

In many cases action must be taken by end users. The system must be able to notify certain users, in certain ways, under certain conditions. The system must provide insight about the current situation and facilitate a range of responses.

Insight can be relayed via dashboards and reports that are custom designed to present information pertinent to the condition at hand. This provides an optimal starting point for analysis, allowing the user to interactively drill down into the problem in the most efficient manner. The information may also be streamed live to the dashboard or report in order to allow the user to monitor the situation.

Guided response is an additional capability whereby the user is directed to certain screens where action needs to be taken. This takes the guesswork out of how to respond by pre-configuring the system navigation that a user should take.

3.3.4 Analysis Delivery

Analysis Delivery includes capabilities that pertain to the way analysis is delivered and presented to consumers.

3.3.4.1 Presentation Formats

Analysis can be delivered to consumers in many ways, and in many formats. The most common formats include dashboards, reports, scorecards, spreadsheets, and portlets. Each of these formats represents a pre-designed arrangement of graphs, tables, indicators, etc., designed to deliver a collection of analysis in a manner best suited to the end user. In addition, the architecture supports delivery of information contained within these formats. For instance, an individual graph can appear on a user's web page or a company portlet. Likewise, measurements can be factored into the execution of a business process.

Dashboards and scorecards both offer an interactive user experience. They present information to the user and allow the user to perform investigative operations in order to better understand the situation at hand. Dashboards essentially are an entry point to analysis. The initial content is populated based on the user's preferences. Once displayed, the user can click through elements on the display to various levels of detail. This enables users to realize and investigate common situations without having to work at the data query level.

Dashboards represent a collection of predefined content such as reports, graphs, tables, tickers, etc. Users create content and use visual tools to assemble dashboards to suit their needs. Dashboards can be shared among users, personalized to an individual's preferences, and designed to adjust automatically to roles and data level security restrictions.

Dashboards can be interactive, allowing users to drill down into information when desired. They may also interact with other IT assets, e.g. passing context from one graph to another, (i.e. master-detail behavior), or passing information to event handlers, services, and business processes. They may also be rendered statically in order to capture information for a report, spreadsheet, document, or presentation.

Scorecards present information about strategy and KPIs, and are designed to allow users to drill into and analyze the data behind the measurements. The linkage between strategy, KPIs, and underlying data provides full line of sight to the business. It enables the business to watch what is happening and understand the cause and effect of daily operations on business strategy.

Alerts are included as a format for providing information to users about conditions or events as they are happening. Alerts can contain analysis information of interest, information identifying the nature of the alert, and/or embedded links that enable end users to address the nature of the event.

3.3.4.2 Delivery Channels

The architecture supports delivery of analysis via multiple delivery channels. The most predominant channel is the Web browser. This channel is used by most modern applications and portals making it a must-have for BA tools and applications. It is a natural choice when analysis needs to be embedded into operational system displays, aggregated with content from other tools, or made available across public and private networks.

As a natural extension to web delivery, the modern enterprise is increasingly mobile oriented. Therefore, mobile delivery is also an important channel of the architecture. Users must be able to access information while away from their office or home. Mobile delivery works well for both accessing information and receiving alerts (SMS messages). Due to physical limitations the mobile platform is not as conducive to in-depth analytical analysis or strategic planning, but is vital for monitoring while on-the-go, and for real-time sense and response functions.

Another important delivery channel is depicted in the architecture as desktop applications. This channel includes all PC applications such as Microsoft Word, Excel, PowerPoint, etc. It also

includes similar applications that run on desktop alternatives, such as tablets. It is often necessary to have information available for these devices using spreadsheets, graphs, or custom applications, therefore the architecture must support this form of access.

Given the desire to protect the confidentiality and integrity of information, it is recommended that desktop applications access information that is managed centrally (via proper information management and security processes) rather than having information copied to each users' desktop applications. This helps avoid problems that arise with data that is out of sync, spreadsheets that are shared without considering security implications, or data that is exposed as the result of lost laptops, USB drives, etc.

Analysis may need to be made available to users that are not online or do not have direct access to the information. The architecture includes a disconnected delivery channel for these circumstances. It supports the snapshot of dashboards, reports, or analysis in a printable form as well as the ability to email the information so it can be viewed while disconnected from the network.

Some advanced forms of analytics, particularly those pertaining to exploratory analysis, will require the use of specialized tools or an integrated development environment (IDE). They enable the user to access and manipulate data using relatively low level constructs such as MapReduce job programming, the R statistics programming language, and machine learning algorithms.

Finally, process integration is included in the architecture in order to represent the link from analytics back to operations. It provides the ability to optimize processes and automate the reaction to analysis-related events as they occur. Process integration links information monitoring and event generation capabilities in this architecture with business process management (BPM) and/or service oriented architecture (SOA) that underpins the operational aspects of the business. For information on BPM and SOA, please consult *ORA BPM Foundation* and *ORA SOA Foundation* respectively.

3.3.5 Analysis Modeling

Modeling serves two primary purposes for the analysis layer: it enables users to design how they want to represent information, and it facilitates standardization and sharing of these representations across the user community.

As described throughout this section, information can be represented in many forms. Graphs, charts, reports, dashboards, and scorecards are among the most widely used. Modeling includes the design and creation of these objects as well as the underlying KPIs, calculations, formulas, and analysis they are derived from.

While it is important to enable the creation of these objects, what becomes more important is the governance and sharing of objects. It offers the ability to define the information required to perform a calculation, define the precise semantic meaning of a calculation, and define a consistent manner in which it is represented. This leads to what is termed 'Single Version of Question' (SVoQ).

SVoQ is a critical aspect of BA in that it helps to ensure that time and money spent to acquire insight is not wasted on inconsistent queries. Even with flawless information available, different people within an organization can interpret data models differently and arrive at different conclusions. While an organization should strive for Single Version of Truth with respect to information, it must also make sure that the queries being performed, (the questions), are equally consistent.

KPIs, formulas, and calculations are important to standardize, (define, share, and govern), since they determine how the question will be asked, e.g. what information is needed and how it will be manipulated. Graphs, charts, reports, and dashboards can also be standardized in order to ensure information is presented clearly and consistently to various audiences. In addition, the

organization may produce standards around look-and-feel for corporate branding, accessibility, useability, and so forth.

Modeling also provides the capability of making information available to users without the time, knowledge, or skills to perform analysis or to design their own representations. Many knowledge workers can benefit from analysis that is designed by experts for various roles within the organization. These end users are recipients of reports, users of dashboards, and receivers of alerts, but are not themselves analysts.

3.3.6 Enterprise Performance Management

EPM, as described in [Section 2.5](#), includes activities such as strategic planning, financial reporting, and profitability management. These activities are complex in nature, often involving applications that are designed to present data and support specific types of processes, reporting, and analysis. Hence, the activities do not map directly to a specific layer of the architecture, rather they involve capabilities that span the entire architecture.

The architecture supports EPM by providing infrastructure required to:

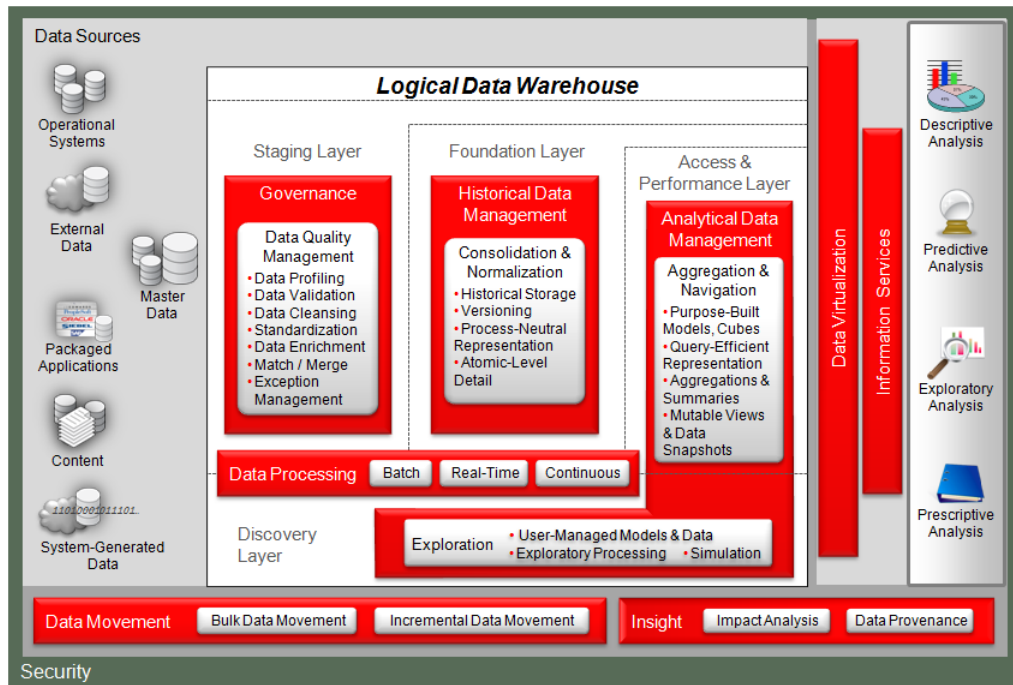
- Capture goals, strategy, and plans, in support of common industry approaches such as Balanced Scorecard, Six Sigma, and Total Quality Management (TQM).
- Establish and monitor KPIs based on a common semantic information model.
- Support drill down from high level KPIs to low level operational data via hierarchical KPI definitions.
- Communicate status to stakeholders and facilitate collaboration.
- Invoke Sense and Response capabilities.
- Associate recommended actions with business exceptions.
- Visualize strategy and causal relationships between KPIs, objectives and initiatives.
- Support trend analysis and forecasting.
- Provide dashboard and reporting capabilities.

3.4 Logical Data Warehouse Conceptual View

[Figure 3–5](#) offers a perspective of the architecture from the standpoint of a logical data warehouse (LDW). It draws on conceptual layers and capabilities that are described earlier in this chapter and organizes them in a way that highlights the function of the data warehouse.

Since this view includes the provisioning of all information to support business analytics, (outside of operational data sources), it is highly likely that implementation will involve more than one product and database technology. Therefore the data warehouse is referred to as a logical data warehouse, implemented as a set of components that together satisfy all required capabilities.

Figure 3–5 Data Warehouse Conceptual View



The LDW is positioned in the architecture between the operational data sources and the consumers of BA information. The warehouse includes capabilities of governance, data processing, historical data management, and analytical data management. It is supported by data movement and insight, and accessed via data virtualization and information services. These groups of capabilities were described earlier in [Section 3.2.1](#), and are further described in *ORA Information Management*.

Data Sources represents all potential sources of "raw" data for the data warehouse. Raw data is assumed to be unclean, unfiltered, and potentially incomplete. Operational data can originate from both internal and external sources. It is generally structured data, but may include unstructured/semi-structured content such as images, documents, sound files, spatial data, etc. It can also contain system-generated data such as log file data, sensor data, Web logs, click streams, etc. The LDW is an ideal place to associate unstructured data with structured data in order to establish linkage across data types. For instance, structured customer records from operational systems can be linked with content (documents and images), along with product ratings and reviews obtained from Web logs.

Data Sources also includes master data and reference data. It is assumed that master data management (MDM) solution(s) will maintain the master copy of these data. The LDW will be a consumer of master and reference data. It may need to maintain a history of these data in order to maintain referential integrity of historical data. It may act as a historical reference point for such data in the event that MDM solutions are unable to provide this capability.

Data Processing is shown to span all layers of the LDW. This illustrates that processing can occur in any layer, and can also occur during the movement of data from one layer to another.

Consumers of information, from this perspective, include the virtualization layer, information and presentation services, as well as BA tools and applications. The primary focus of this view is the LDW, therefore consumption is highly generalized. Just enough detail is provided to illustrate touch points between the LDW and other layers of the architecture described earlier in this section.

3.4.1 Staging Layer

Structured data, and some unstructured or semi-structured data, enter the LDW via the Staging Layer. This layer acts as a temporary storage area for data manipulation and cleansing. It provides a level of isolation between data that are received by the warehouse and data that are generally available to consumers. By providing this layer of isolation, data can be received at various rates, (e.g. in small frequent increments or large bulk transfers), asynchronous to the rate at which data are refreshed for consumption.

The Staging Layer is where capabilities of data quality management are applied in order to achieve clean, consistent, and complete data. Once cleansed, data are moved into the Foundation Layer for permanent storage, or into the Discovery Layer for processing.

3.4.2 Foundation Layer

The Foundation Layer represents the heart of the LDW. It is responsible for managing data for the long term. It may also be called the Atomic Data Layer since it maintains data at the lowest level of granularity.

It is in the Foundation Layer that data from all originating sources is maintained for historical reference. Structured data should be consolidated, to the extent possible, into a common schema. Although actual consolidation may take place during data movement, processing, or staging, the master version resides in this layer.

In order to most easily adapt to changes in the organization over time, structured data are maintained here in a business neutral form. Changes to organizational structures and dimensions should not impact the way data are structured in this layer. Versatility takes precedence over navigation and performance. Likewise, in order to efficiently store large volumes of structured data, this layer is normalized in a manner similar to 3NF. Versioning of records and reference data is also managed in this layer.

3.4.3 Access & Performance Layer

The Access and Performance Layer (APL) is used to represent data in ways that best serve the BA community. Since it does not have responsibility for historical data management, it is free to represent data for most efficient access and ease of navigation. It is also where user-defined data can be introduced for simulation and forecasting purposes.

The structure of the APL is dependent upon the tools and applications using the data. Often it will consist of dimensional models and/or cubes. In addition, it may contain aggregations of facts (rollups) for rapid query responses. APL structures can be instantiated and changed at will in order to conform to the business hierarchies of the day. When changes occur within the business, the APL can be reoriented and repopulated from the Foundation Layer.

Performance can be enhanced in a number of ways, including:

- The creation and tuning of models to be most efficient for the tools and queries being performed.
- Pre-aggregation of dimensional data to populate cubes.
- The exclusion of data that are not needed, e.g. time ranges, levels of detail, and previous versions.
- The deployment of specialized hardware or software including caching mechanisms and in-memory databases to meet query response time requirements.

The separation of duties between the Foundation Layer and APL provide a best case scenario for the business. Historical data is managed in a way that is most tolerant of business changes, yet BA users, tools, and applications can operate on the most efficient data representations.

3.4.4 Discovery Layer

The Discovery Layer provides a separate location where exploration activities can take place. It parallels the other layers of the LDW in that data can move between it and all of the other layers. Data movement is described further in ORA Business Analytics Infrastructure.

There are three common themes to how the Discovery Layer is used:

1. As a means to maintain data that will be used for analysis but not stored in the Foundation Layer or Access and Performance Layer. Bulk data loads, for example, may be kept in this layer in support of Exploratory Analytics. It may remain in the Discovery Layer unless it is later determined that governance or historical data management capabilities are required.
2. As a location for user-managed data sets and models (i.e. sandboxes). User-managed models allow the system to quickly evolve in support of new end user requirements. Data can be copied from the other layers and organized according to users' needs. This is especially useful for Predictive Analytics where simulations need to be run on custom data models, and for Exploratory Analytics operations such as machine learning. It is also a convenient place to perform experimental modeling, such as modeling relationships between different sources and types of data.
3. As a location to process data for exploratory purposes. Data can be loaded into this layer directly from the Data Sources, from other layers of the LDW, or from a combination of both. Processing can be performed in this layer in order to justify new business opportunities, to help solve specific problems, or to simply determine if new sources of data have value and should be maintained in the Foundation Layer.

Technology Standards

Business analytics builds upon many facets of technology that are described in other ORA documents, including *ORA Information Management*, *ORA Service Orientation*, *ORA User Interaction*, and *ORA Security*. Many technology standards presented in those documents apply directly to BA. In addition, the following standards for BA are introduced in this section.

4.1 Dimensional Query & Interface Standards

4.1.1 OLE DB for OLAP (ODBO)

OLE DB for OLAP is a specification for multidimensional data processing published by Microsoft. It is the standard API for exchanging metadata and data between an OLAP server and a client on a Microsoft Windows platform. The ODBO specification is specific to the Microsoft Windows platform, although other vendors can use this specification to support integration with their OLAP products.

ODBO introduced MDX as the multidimensional query language.

Information on ODBO can be found at:

<http://msdn.microsoft.com/en-us/library/ms714903%28VS.85%29.aspx>

4.1.2 Multidimensional Expressions (MDX)

MDX provides a rich and powerful syntax for querying and manipulating the multidimensional data stored in OLAP server cubes¹. It is similar to SQL in language and structure, but includes capabilities that are beneficial for working with multidimensional data. For example, MDX includes data types for dimensions, levels (within a dimension), members (of dimensions), tuples (collection of members from different dimensions), and sets (collection of tuples).

MDX was first introduced by Microsoft in 1997. Although it is not officially an open standard, many OLAP vendors on both the server side and client side have adopted it.

MDX documentation can be found at <http://msdn.microsoft.com/en-us/library/ms145514.aspx>

4.1.3 Open Geospatial Consortium (OGC) Standards

The Open Geospatial Consortium (OGC) is an international industry consortium of 482 companies, government agencies and universities participating in a consensus process to develop publicly available interface standards. OGC® Standards support interoperable

¹ Carl Nolan. "Manipulate and Query OLAP Data Using ADOMD and Multidimensional Expressions". Microsoft. Retrieved 2008-03-05.

solutions that "geo-enable" the Web, wireless and location-based services and mainstream IT. The standards empower technology developers to make complex spatial information and services accessible and useful with all kinds of applications.²

The OGC maintains a number of standards related to geospatial computing, including the Web Feature Service (WFS), the transactional Web Feature Service (WFS-T), and the Geography Markup Language (GML).

4.1.3.1 Web Feature Service Standards (WFS, WFS-T)

The Web Feature Service defines a standard interface for querying geographic information on the Internet. It offers direct fine-grained access to geographic information at the feature and feature property level. This enables consumers of WFS-based services to work with features within the data set.

Prior to WFS, geographic information was often obtained as an image file. This made editing and spatial analysis difficult or impossible. The WFS standard represents data using an XML-based language called Geography Markup Language (GML), which makes it easy to access and manipulate features within the data.

A transactional Web Feature Service (WFS-T) has also been defined that supports creating, updating, and deleting geographic features. Information on WFS and WFS-T can be obtained at: <http://www.opengeospatial.org/standards/wfs>.

4.1.3.2 Geography Markup Language (GML)

The Geography Markup Language (GML) is an XML grammar for expressing geographical features. GML serves as a modeling language for geographic systems as well as an open interchange format for geographic transactions on the Internet. As with most XML based grammars, there are two parts to the grammar - the schema that describes the document and the instance document that contains the actual data. A GML document is described using a GML Schema. This allows users and developers to describe generic geographic data sets that contain points, lines and polygons. GML is also an ISO standard ([ISO 19136:2007](#))³.

Users can extend GML-based schemas to refer to specific application-level constructs such as roads, highways, and bridges. Provided these extensions are recognized across the user base, they are even more convenient than working with generic features such as points, lines, and polygons.

4.1.4 XML for Analysis (XMLA)

XML for Analysis is a standard that allows client applications to talk to multi-dimensional or OLAP data sources⁴. The communication of messages back and forth is done using web standards - HTTP, SOAP, and XML. The query language used is MDX. XMLA specifies a set of XML message interfaces over SOAP to define data access interaction between a client application and an analytical data provider. Using a standard API, XMLA provides open access to multi-dimensional data from varied data sources - any client platform to any server platform - through web services that are supported by multiple vendors.

XML for Analysis is designed to provide the interactive capabilities of ODBO, with support for MDX, but without the platform-specific limitations. The XML for Analysis specification is available at: <http://news.xmlforanalysis.com/docs/xmla1.1.doc>

² [About OGC, OGC Web site](#), 15 January 2013

³ [Geographic Markup Language, OGC Web site](#), 15 January 2013

⁴ [XML for Analysis.com](#)

4.2 Programming Standards

4.2.1 MapReduce

MapReduce is a framework for processing parallelizable problems across huge datasets using a large number of computers (nodes), collectively referred to as a cluster (if all nodes are on the same local network and use similar hardware) or a grid (if the nodes are shared across geographically and administratively distributed systems, and use more heterogeneous hardware). Computational processing can occur on data stored either in a file system (unstructured) or in a database (structured). MapReduce can take advantage of locality of data, processing data on or near the storage assets to decrease transmission of data.

MapReduce is comprised of two steps:

1. **"Map" step:** The master node takes the input, divides it into smaller sub-problems, and distributes them to worker nodes. A worker node may do this again in turn, leading to a multi-level tree structure. The worker node processes the smaller problem, and passes the answer back to its master node.
2. **"Reduce" step:** The master node then collects the answers to all the sub-problems and combines them in some way to form the output - the answer to the problem it was originally trying to solve.

MapReduce allows for distributed processing of the map and reduction operations. Provided each mapping operation is independent of the others, all maps can be performed in parallel - though in practice it is limited by the number of independent data sources and/or the number of CPUs near each source. Similarly, a set of 'reducers' can perform the reduction phase - provided all outputs of the map operation that share the same key are presented to the same reducer at the same time, or if the reduction function is associative. While this process can often appear inefficient compared to algorithms that are more sequential, MapReduce can be applied to significantly larger datasets than "commodity" servers can handle - a large server farm can use MapReduce to sort a petabyte of data in only a few hours. The parallelism also offers some possibility of recovering from partial failure of servers or storage during the operation: if one mapper or reducer fails, the work can be rescheduled - assuming the input data is still available.⁵

4.2.2 R

R is a language and environment for statistical computing and graphics. It is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form. It compiles and runs on a wide variety of platforms. It is also packaged into several commercially available product suites such as Oracle Advanced Analytics.

R provides a variety of statistical and graphical techniques, such as: linear and nonlinear modeling, classical statistical tests, time-series analysis, classification, and clustering.

The R environment includes:

- a data handling and storage facility
- a suite of operators for calculations on arrays
- an integrated collection of tools for data analysis
- graphical facilities for data analysis and display
- a programming language for data analysis and presentation

⁵ [MapReduce - Wikipedia](#), 12 December 2012

The R project website is located at: <http://www.r-project.org/index.html>.

4.2.3 Continuous Query Language (CQL)

CQL is an SQL based language for maintaining continuous queries over long periods of time. Such queries are suitable for real-time and reactive programs - those where keeping an up-to-date view and latency are important (such as collaborative editing, traffic or safety or security monitoring, command and control, robotics, etc.). The original CQL was developed at Stanford University between 2002-2006 as part of a project named 'STREAM'. That implementation is now available under the BSD license. In 2007 Oracle Corporation introduced 'Oracle CQL', as part of its Complex Event Processing (CEP) framework and application server. This language varies from Stanford's but serves a similar role and purpose.⁶

Oracle CQL is designed to be:

- Scalable with support for a large number of queries over continuous streams of data and traditional stored data sets.
- Comprehensive to deal with complex scenarios. For example, through composability, you can create various intermediate views for querying.

Documentation on Oracle CQL is available at: http://docs.oracle.com/cd/E16764_01/doc.1111/e12048/intro.htm.

4.3 Reporting Standards

4.3.1 eXtensible Business Reporting Language (XBRL)

XBRL is a global standard language for the electronic communication of business and financial data. It uses XML syntax to model information and to define the semantic meaning of data.

The idea behind XBRL is simple⁷. Instead of treating financial information as a block of text - as in a standard internet page or a printed document - it provides an identifying tag for each individual item of data. This is computer readable. For example, company net profit has its own unique tag. This allows automated processing of business information. XBRL documents can be created, analyzed, presented, and stored programmatically.

Companies can use XBRL to save costs and streamline their processes for collecting and reporting financial information. Consumers of financial data, including investors, analysts, financial institutions and regulators, can receive, find, compare, and analyze data much more rapidly and efficiently if it is in XBRL format.

The XBRL specification can be found on the XBRL International site at: <http://www.xbrl.org/SpecRecommendations/>

⁶ [Continuous Query Language](#), 17 December 2012

⁷ XBRL International, [An Introduction to XBRL](#)

Interlocking Technologies

Business analytics is an important function of any business, and it relies heavily on capabilities that are provided by technology. This document and *ORA Business Analytics Infrastructure* focus mainly on technologies that are paramount to BA. They combine to provide a view of architecture from the perspective of BA.

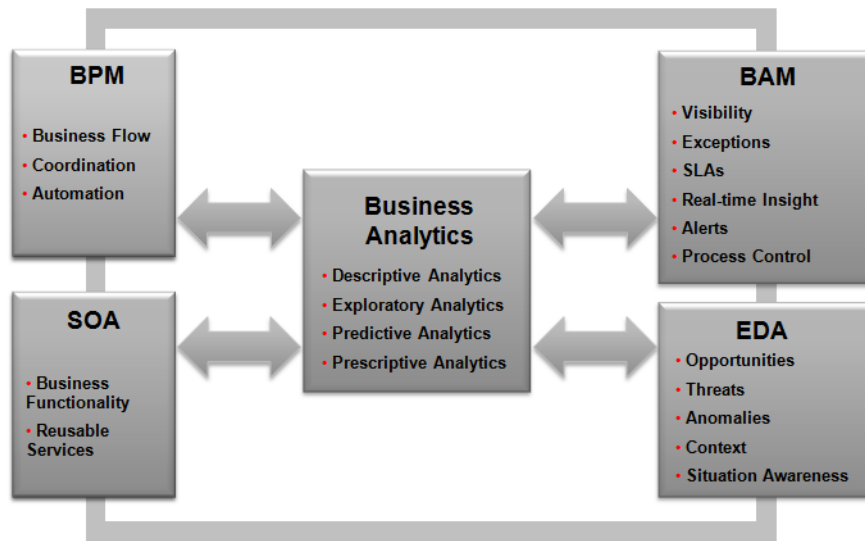
BA builds upon many other technologies, such as application infrastructure, integration, security, monitoring, management, etc. It also works well in conjunction with technology strategies such as Business Process Management (BPM), Service-Oriented Architecture (SOA), and Event-Driven Architecture (EDA).

Given the breadth of technologies available today, the ORA documentation has been divided into documents that each focuses on a specific technology. The architecture views can be combined at will to create composite views of an overall architecture. This chapter illustrates some of the conceptual relationships between BA and other technologies. It is useful for organizations that want to benefit from the synergies of multiple technologies, or want to conceptually understand how technologies might relate to each other. Further information on other technologies can be found in ORA documents listed in each section.

5.1 SOA, BPM, EDA, & BAM

SOA, BPM, EDA, and Business Activity Monitoring (BAM), are very popular technology strategies. It is highly likely that an organization that uses business analytics will also use one or more of these technologies. As depicted in [Figure 5-1](#), each technology offers a set of capabilities, and BA can interact with the other technologies in order to combine capabilities for maximum benefit.

Figure 5–1 Business Analytics with SOA, BPM, EDA, & BAM



The following sections describe some of the interactions between BA and other technologies.

5.1.1 BA & Business Processes

BPM coordinates and automates the flow of activities for a business process. Activities may be performed manually, or they may be automated. Processes can spawn other (sub)processes to perform complex tasks. In addition, the process flow may fork in different paths, depending on certain variables and decision points, and rejoin at a later time.

Figure 5–2 BA & Business Processes

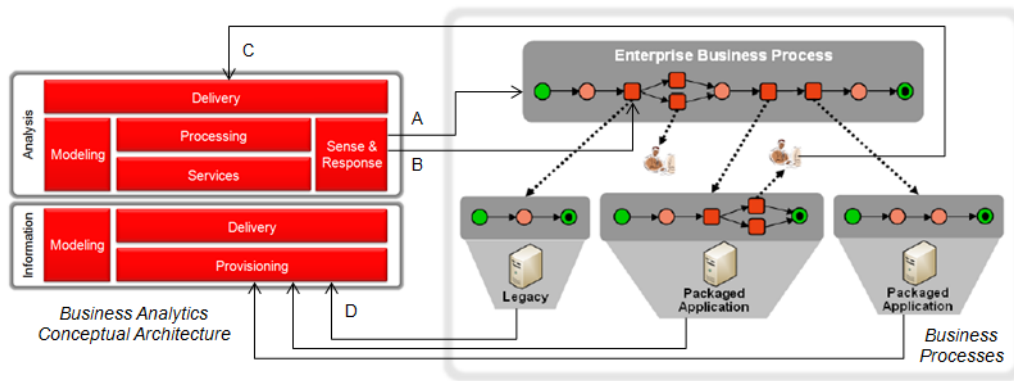


Figure 5–2 illustrates several touch points between BA and business processes. Via sense and response capabilities, BA can be configured to trigger business processes (A) in order to automatically handle situations that arise, such as unusually high or low sales volumes. BA can factor into business rules (B), which in turn are used by business processes to make automated decisions. Decisions may affect the flow of a process, such as whether or not a stock item should be reordered. Users might rely on BA to provide insight (C) required for them to perform their manual tasks. Conversely, data affected by business processes are collected and provisioned to provide future intelligence (D).

5.1.2 Real-Time Intelligence

Business analytics must be capable of drawing from both historical and real-time information. The latter enables up-to-the-minute insight and trend analysis, while the former provides long term context.

Real-time data can be drawn from operational systems in a passive query-response model. This assumes that users know what data to look at and when it needs to be viewed. It also means introducing data for analysis that has not been cleansed and normalized via the Information Provisioning architecture layer.

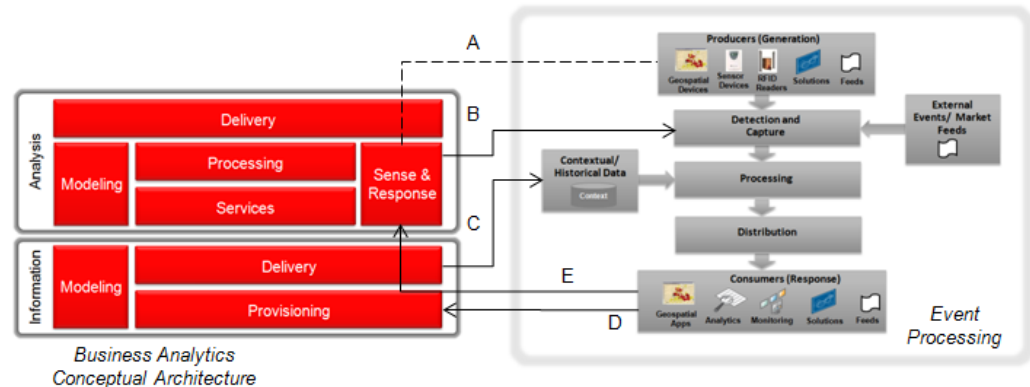
Another way to provide real-time insight, where and when it is most meaningful, is to allow events and business activities to trigger such analysis. The BA architecture is designed to react to meaningful real-time events and take action in its own way. Conversely, the BA architecture can generate events for other IT components to use. This model is described using two similar, but unique real-time architecture patterns: event processing, and business activity monitoring.

5.1.2.1 BA and Event Processing

Event processing, a key capability of Event-Driven Architecture (EDA), is designed to handle business opportunities, threats, and anomalies as they occur. The purpose is to provide situation awareness and visibility into business operations.

Event processing detects and captures events from across the IT landscape, both internal and external to the business. It may correlate multiple events, and detect conditions based on event sequences, elapsed time periods, and various rules. When pre-defined conditions are detected, it will alert all consumers that have subscribed to the event.

Figure 5–3 BA & Event Processing



As shown in Figure 5–3, BA architecture can interact with event processing architecture at multiple levels. First, the Sense & Response capability of BA can be classified as a producer of events (dotted line A). When it detects that certain conditions exist, it can send events to the event capture components (B). Event processing may draw on historical data in order to determine when an event needs to be acted upon. This historical data may be provisioned and delivered via the same Information Layer that BA uses (C).

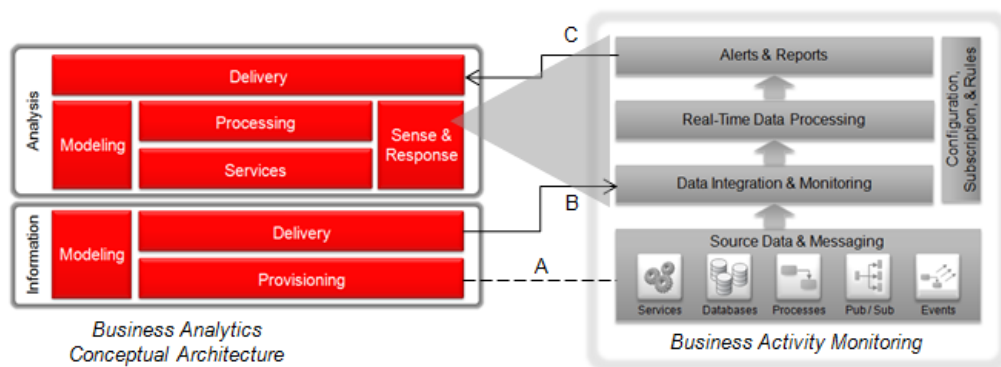
As a consumer of events, the BA architecture may either store event information for ongoing or later analysis (D), or it may use events to trigger actions in real time (E). Real-time actions can include user notifications, real-time report generation, dashboard updates, and guided interactive user sessions.

5.1.2.2 BA and Business Activity Monitoring

Business activity monitoring (BAM) is similar to event processing in that it works with real-time data, however it is more focused on providing insight to users about current business conditions. Where the output from EDA is event notification; the output from BAM is business insight, which often involves aspects of information gathering, analytics, and presentation delivery.

BAM operates by collecting information from a variety of sources, either by receiving messages (or events), by querying data sources, or by invoking data services. BAM correlates and analyzes the information, looking for pre-configured conditions. It then either sends notifications to users, or generates interactive reports.

Figure 5-4 BA and Business Activity Monitoring



BAM works in conjunction with BA by providing robust real-time analysis and interaction capabilities. The two fit together so well that BAM is positioned in Figure 5-4 as a subset of the BA architecture. The capabilities are included in Sense & Response from the BA Conceptual View, and BAM components and product mapping are included in *ORA Business Analytics Infrastructure*.

BAM capabilities tie into other aspects of BA in several ways. Although most source data for BAM will be driven by operational systems, the Information Layer of BA may comprise some of the BAM source data (A). BAM can integrate data from operational systems as well as systems contained within the Information Provisioning layer (B). Delivery of BAM content such as interactive reports is included in the Analysis Delivery layer (C).

5.1.3 BA & Service Orientation

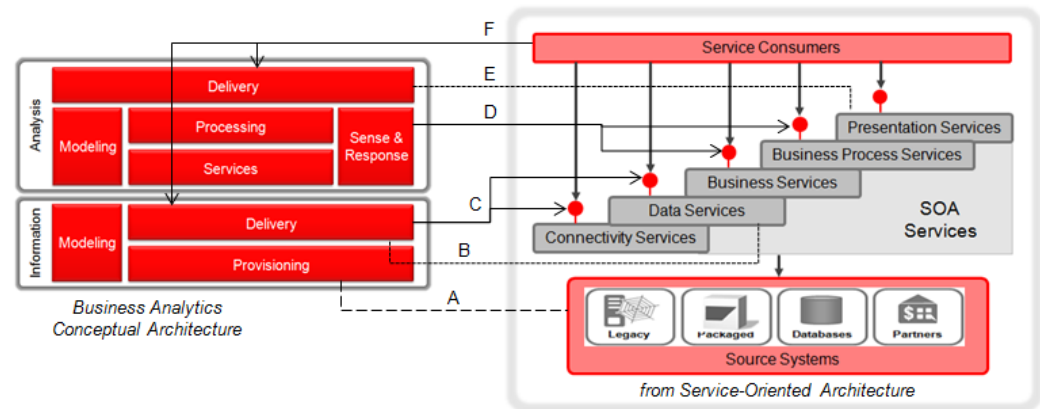
Service Oriented Architecture (SOA) enables encapsulation of business functionality and data as reusable services. SOA Services provide a well-defined interface and contract, which promotes discovery, autonomy, and reuse across the organization. SOA is designed to help organizations be more efficient and flexible by decomposing massive applications into reusable building blocks (SOA Services), and by integrating the building blocks via open standards-based protocols and message formats.

ORA SOA Foundation defines five basic types of SOA Services:

- **Connectivity Services** which provide standardized access to legacy assets that are not able to provide such access on their own.
- **Data Services** which offer virtualized access to data sources.
- **Business Services** which encapsulate business logic.
- **Business Process Services** which provide access to multi-step business processes.

- **Presentation Services** which expose UI components as services.

Figure 5–5 BA and SOA Services



The relationship between BA and SOA Services is bi-directional, e.g. BA can act as a consumer as well as a provider of SOA Services. In addition, data provisioned for BA can be one of the many types of source systems in a SOA architecture (A). The dotted lines in [Figure 5–5](#) indicate where BA acts as a service provider. Information can be delivered from the BA architecture as a form of Data Service (B). Also, analysis can be delivered in various formats as Presentation Services (E). SOA service consumers invoke Data Services and Presentation Services provided by BA via their respective delivery capabilities (F).

As a service consumer, BA Information Delivery can invoke Connectivity Services and/or Data Services (C) in order to obtain some data required by the Analysis Layer. In addition, the Sense and Response capabilities may invoke Business Services or Business Process Services (D) when certain conditions arise.

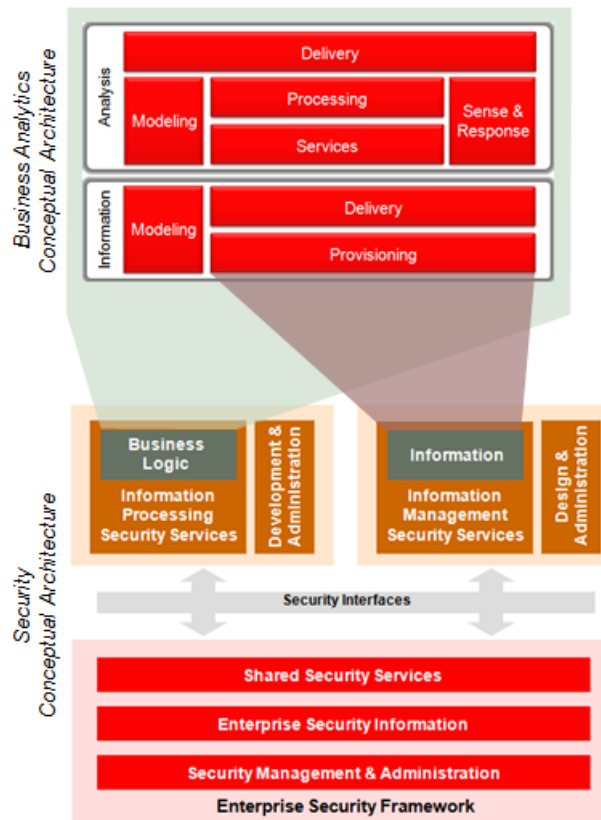
5.2 BA and Security

The conceptual view of security architecture, as described in *ORA Security*, promotes the ideal of secure application infrastructure platforms that leverage common security services. These platforms facilitate information processing and information management and provide local security services such as authentication, authorization, auditing, and cryptography. Local security services are either directly invoked via application business logic, or are indirectly (automatically) invoked via information processing and information management platform logic.

Local platform security services, in turn, are clients of enterprise security services. Services at the enterprise level enable consistent definition of users, credentials, access policies, etc., as well as universal identity management, auditing, attestation, and governance.

Having both sets of security services allows each platform the ability to enforce and administer security locally, using the most efficient and effective platform-specific mechanisms, whilst giving the enterprise a holistic management viewpoint, a common set of security information, and a consistent means to implement shared security services.

Figure 5–6 BA & Security Architecture



Architecturally, BA consists of both information management and information processing capabilities. The BA Information Layer contains a set of capabilities labeled Provisioning, which includes historical and analytical data management. These platforms can be mapped directly to the information management platforms of the security architecture, as illustrated in Figure 5–6.

Information management platforms offer local security services to protect the confidentiality, integrity, and availability of information. These services provide access control, cryptography, data redaction, data masking, auditing, monitoring, etc. The intent is to establish a defense-in-depth capability for maximum data protection.

All other aspects of the BA architecture can be classified as information processing capabilities. They are mapped directly to the business logic that is protected by information processing security services. In this case business logic can be applications or tools that an organization buys, or those that an organization builds for itself. In both cases the logic must be secured, and a common set of security services is recommended.

Information processing security services include authentication, authorization, auditing, and cryptography. They protect business logic that operates on information. They leverage enterprise-wide services for single sign-on (SSO), identity federation, and policy management. They can also defer to enterprise-wide authentication, authorization, and auditing services.

For more information on security, please consult the *ORA Security* document.

Summary

In "Competing on Analytics: The New Science of Winning", Harvard Business School Press, Thomas Davenport and Jeanne Harris define analytics as "the extensive use of data, statistical and quantitative analysis, explanatory and predictive models, and fact-based management to drive decisions and actions". This combination of information, modeling, and analysis, combined with action, is what makes business analytics valuable. And it is this form of business analytics that can give organizations a competitive advantage in the marketplace today.

Decision-making requires the right information delivered to the right people at the right time. All of which depends on the ability of IT to provide what the business needs as quickly as possible, when the business needs it. This is no easy task, especially considering the sheer volume of data that are often involved. The BA initiative must be well planned and executed, otherwise it can turn from a differentiating asset into a burden on the business.

A BA reference architecture must be defined in order to ensure that IT consistently implements what the business needs, and that IT can remain flexible to quickly adapt to new and changing needs. Such an architecture must be rooted in sound principles and proven technologies. The architecture must have clearly defines capabilities that are in line with business requirements.

This document comprised many important concepts, capabilities, and technology standards for business analytics. It organized capabilities into a conceptual architecture, which can be used as a starting point for the definition of a logical reference architecture. The conceptual architecture can be used as-is, or can be tailored to fit the needs of a specific organization. The companion document, *ORA Business Analytics Infrastructure*, builds on the content of this document to provide a logical reference architecture. It also illustrates how Oracle's BA products map to the logical architecture.



Further Reading and References

The following references provide more information on business analytics and related IT architecture topics.

A.1 Related Documents

The IT Strategies from Oracle series contains a number of documents that offer insight and guidance on many aspects of technology. In particular, the following documents from the ORA series may be of interest:

ORA Service-Oriented Integration - This document examines the most popular and widely used forms of integration, putting them into perspective with current trends made possible by SOA standards and technologies. It offers guidance on how to integrate systems in the Oracle environment, bringing together modern techniques and legacy assets.

ORA Security - This document describes important aspects of security including identity, role, and entitlement management; authentication, authorization, and auditing (AAA); and transport, message, and data security required to secure the modern IT environment.

ORA Monitoring & Management - A common thread running through many applications, services, and systems is the ability to monitor and manage assets in a consistent and efficient manner. ORA Monitoring and Management offers a framework for OA&M to rationalize these capabilities and help optimize the operational aspects of enterprise computing.

ORA Service Orientation - The promise of cost savings and agility derived from a service oriented approach to architecture has garnered widespread attention within the IT industry. This document describes how Oracle Reference Architecture embraces service orientation to connect disparate technologies into a unified reference architecture.

ORA SOA Foundation - This document describes the key tenets for SOA design, development, and execution environments. Topics include: service definition, service layering, service types, the service model, composite applications, invocation patterns, and standards.

ORA SOA Infrastructure - Properly architected, SOA provides a robust and manageable infrastructure that enables faster solution delivery. This document describes the role of infrastructure and its capabilities. Topics include: logical architecture, deployment views, and Oracle product mapping.

ORA BPM Foundation - This document defines the core concepts of modern BPM, provides a conceptual architecture depicting the key capabilities required, and identifies the architectural principles for successful BPM.

ORA BPM Infrastructure - This document connects the conceptual architecture with a logical architectural view and includes the functional components necessary. Topics include: logical architecture, deployment considerations, and Oracle product mapping.

ORA EDA Foundation - This document describes the concepts and business benefits of EDA, provides a conceptual architecture depicting the key capabilities required, identifies the architectural principles for successful EDA, and identifies and describes the relevant industry standards.

ORA EDA Infrastructure - This document describes the infrastructure and its capabilities necessary to process complex events. Topics include: logical architecture, deployment views, and Oracle product mapping.

A.1.1 Suggested Pre-reading

The following documents are suggested pre-reading for those that would like to more fully understand the concepts this document builds upon:

ORA Information Management - This document describes important aspects of information management that provide the capabilities necessary to organize, persist, manage, and retrieve multi-structured enterprise information. This includes the capabilities to ensure accuracy, integrity, and consistency of the information as well as data integration capabilities.

A.2 Other Resources and References

In addition, the following materials and sources of information relevant to BA may be useful:

- Doug Cackett, Andrew Bond, Kevin Lancaster, and Keith Laker. [Enabling Pervasive BI through a Practical Data Warehouse Reference Architecture](#). An Oracle White Paper. February 2010.
- [Oracle Enterprise Performance Management & Business Intelligence Whitepapers](#) available online.
- Thomas H., and Jeanne G. Harris, *Competing on Analytics: The New Science of Winning*, Davenport, 2007, Harvard Business School.
- Boris Evelson, Forrester Research. [Topic Overview: Business Intelligence](#). November 21, 2008. Online.
- Mark N. Frolick and Thilini R. [Business Performance Management: One Truth](#). Ariyachandra, Information Systems Management, Winter 2008. Online PDF.
- Peter Buneman, Sanjeev Khanna, and Wang-Chiew Tan. [Why and Where: A Characterization of Data Provenance](#). University of Pennsylvania Departmental Papers. 1 Jan 2001. Online PDF.
- Carl Nolan. [Manipulate and Query OLAP Data Using ADOMD and Multidimensional Expressions](#). Microsoft. Retrieved 2008-03-05. Online Article.
- Ralph Kimball, et al. *The Kimball Group Reader, Relentlessly Practical Tools for Data Warehousing and Business Intelligence*. Wiley Publishing. © 2010 Ralph Kimball and Margy Ross.
- Doug Cackett, et al. [Information Management and Big Data, A Reference Architecture](#). An Oracle White Paper. February 2013.