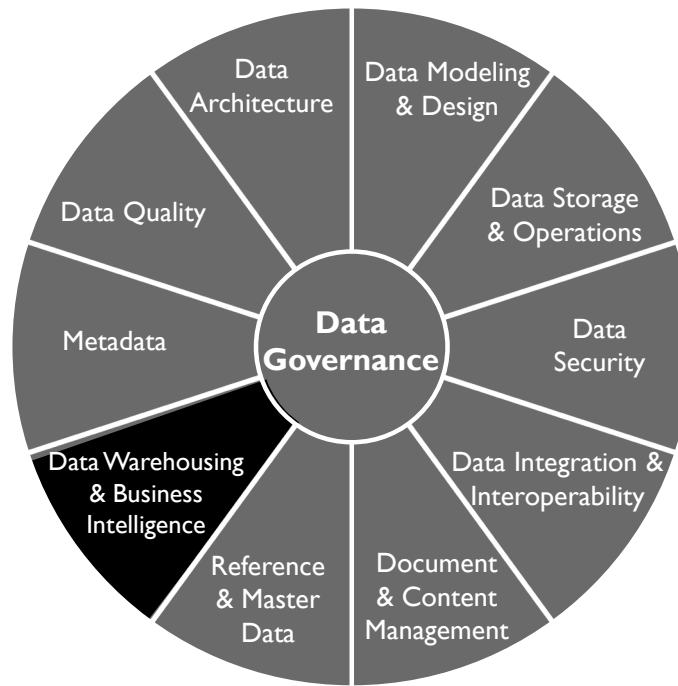


Data Warehousing and Business Intelligence



DAMA-DMBOK2 Data Management Framework

Copyright © 2017 by DAMA International

1. Introduction

The concept of the Data Warehouse emerged in the 1980s as technology enabled organizations to integrate data from a range of sources into a common data model. Integrated data promised to provide insight into operational processes and open up new possibilities for leveraging data to make decisions and create organizational value. As importantly, data warehouses were seen as a means to reduce the proliferation of decision support systems (DSS), most of which drew on the same core enterprise data. The concept of an enterprise warehouse promised a way to reduce data redundancy, improve the consistency of information, and enable an enterprise to use its data to make better decisions.

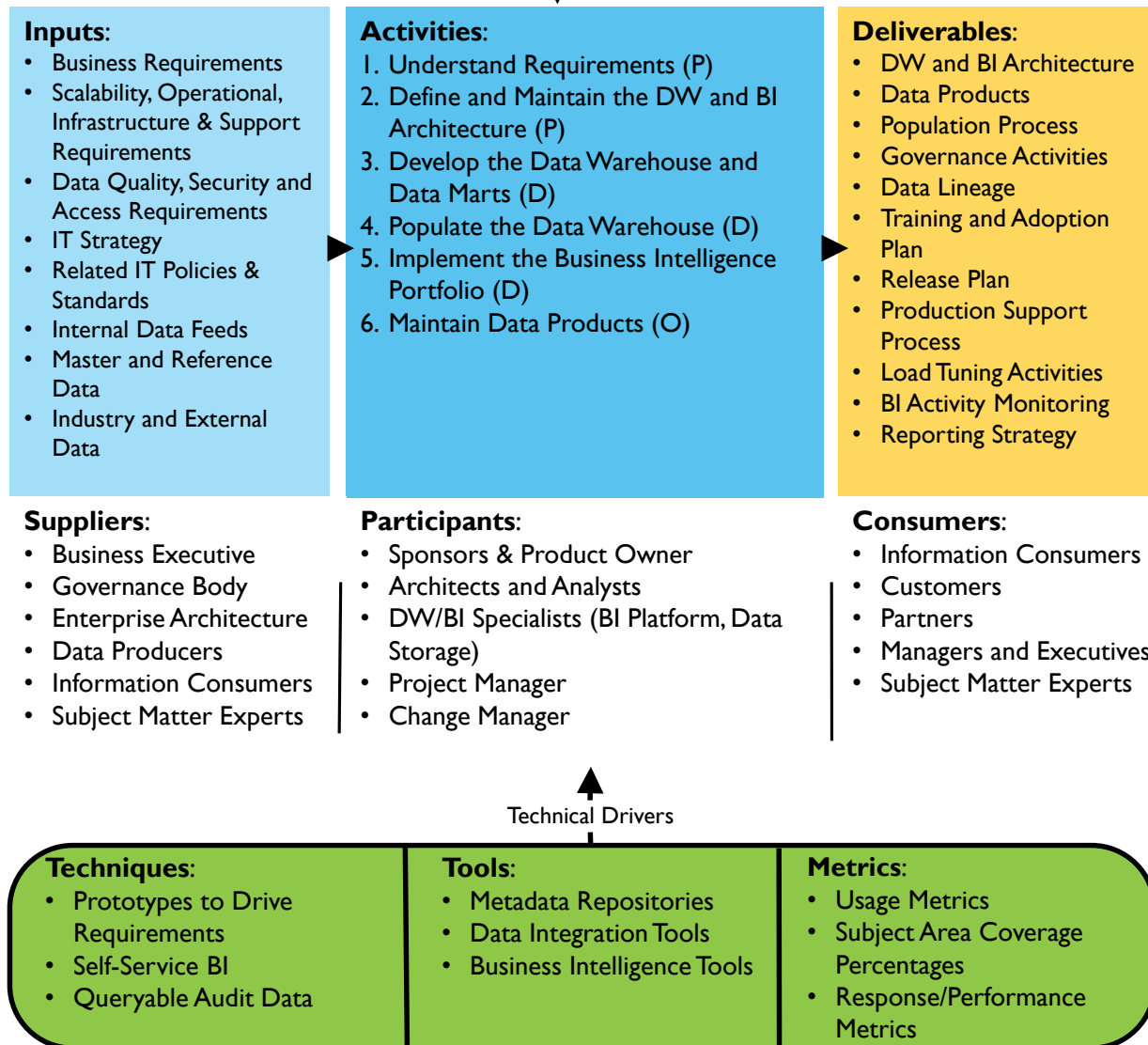
Data Warehousing and Business Intelligence

Definition: Planning, implementation, and managing an integrated data system to support knowledge workers engaged in reporting, query, and analysis.

Goals:

1. To build and maintain the data system with the technical and business requirements needed to deliver integrated data that supports operational functions, compliance, and business intelligence.
2. To create insights to support and enable effective business analysis and decision making.

Business Drivers



(P) Planning, (C) Control, (D) Development, (O) Operations

Figure 79 Context Diagram: DW/BI

Data warehouses began to be built in earnest in the 1990s. Since then (and especially with the co-evolution of Business Intelligence as a primary driver of business decision-making), data warehouses have become

‘mainstream’. Most enterprises have data warehouses and warehousing is the recognized core of enterprise data management.⁶³ Even though well established, the data warehouse continues to evolve. As new forms of data are created with increasing velocity, new concepts, such as data lakes, are emerging that will influence the future of the data warehouse. See Chapters 8 and 15.

1.1 Business Drivers

The primary driver for data warehousing is to support operational functions, compliance requirements, and Business Intelligence (BI) activities (though not all BI activities depend on warehouse data). Increasingly organizations are being asked to provide data as evidence that they have complied with regulatory requirements. Because they contain historical data, warehouses are often the means to respond to such requests. Nevertheless, Business Intelligence support continues to be the primary reason for a warehouse. BI promises insight about the organization, its customers, and its products. An organization that acts on knowledge gained from BI can improve operational efficiency and competitive advantage. As more data has become available at a greater velocity, BI has evolved from retrospective assessment to predictive analytics.

1.2 Goals and Principles

Organizations implement data warehouses in order to:

- Support Business Intelligence activity
- Enable effective business analysis and decision-making
- Find ways to innovate based on insights from their data

The implementation of a Data Warehouse should follow these guiding principles:

- **Focus on business goals:** Make sure DW serves organizational priorities and solves business problems.
- **Start with the end in mind:** Let the business priority and scope of end-data-delivery in the BI space drive the creation of the DW content.
- **Think and design globally; act and build locally:** Let end-vision guide the architecture, but build and deliver incrementally, through focused projects or sprints that enable more immediate return on investment.
- **Summarize and optimize last, not first:** Build on the atomic data. Aggregate and summarize to meet requirements and ensure performance, not to replace the detail.
- **Promote transparency and self-service:** The more context (Metadata of all kinds) provided, the better able data consumers will be to get value out of the data. Keep stakeholders informed about the data and the processes by which it is integrated.
- **Build Metadata with the warehouse:** Critical to DW success is the ability to explain the data. For example, being able to answer basic questions like “Why is this sum X?” “How was that computed?”

⁶³ <http://bit.ly/2sVPIYr>.

and “Where did the data come from?” Metadata should be captured as part of the development cycle and managed as part of ongoing operations.

- **Collaborate:** Collaborate with other data initiatives, especially those for Data Governance, Data Quality, and Metadata.
- **One size does not fit all:** Use the right tools and products for each group of data consumers.

1.3 Essential Concepts

1.3.1 Business Intelligence

The term *Business Intelligence* (BI) has two meanings. First, it refers to a type of data analysis aimed at understanding organizational activities and opportunities. Results of such analysis are used to improve organizational success. When people say that data holds the key to competitive advantage, they are articulating the promise inherent in Business Intelligence activity: that if an organization asks the right questions of its own data, it can gain insights about its products, services, and customers that enable it to make better decisions about how to fulfill its strategic objectives. Secondly, *Business Intelligence* refers to a set of technologies that support this kind of data analysis. An evolution of decisions support tools, BI tools enable querying, data mining, statistical analysis, reporting, scenario modeling, data visualization, and dashboarding. They are used for everything from budgeting to advanced analytics.

1.3.2 Data Warehouse

A *Data Warehouse* (DW) is a combination of two primary components: An integrated decision support database and the related software programs used to collect, cleanse, transform, and store data from a variety of operational and external sources. To support historical, analytical, and BI requirements, a data warehouse may also include dependent data marts, which are subset copies of data from the warehouse. In its broadest context, a data warehouse includes any data stores or extracts used to support the delivery of data for BI purposes.

An *Enterprise Data Warehouse* (EDW) is a centralized data warehouse designed to service the BI needs of the entire organization. An EDW adheres to an enterprise data model to ensure consistency of decision support activities across the enterprise.

1.3.3 Data Warehousing

Data Warehousing describes the operational extract, cleansing, transformation, control, and load processes that maintain the data in a data warehouse. The data warehousing process focuses on enabling an integrated and historical business context on operational data by enforcing business rules and maintaining appropriate business data relationships. Data warehousing also includes processes that interact with Metadata repositories.

Traditionally, data warehousing focuses on structured data: elements in defined fields, whether in files or tables, as documented in data models. With recent advances in technology, the BI and DW space now embraces semi-

structured and unstructured data. Semi-structured data, defined as electronic elements organized as semantic entities with no required attribute affinity, predates XML but not HTML; an EDI transfer could serve as an example. Unstructured data refers to data that is not predefined through a data model. Because unstructured data exists in a range of formats and encompasses items such as e-mail, free format text, business documents, videos, photos, and web pages to name a few, defining a feasible storage construct that sustains analytic workloads within warehousing governance has been a challenge yet to be overcome.

1.3.4 Approaches to Data Warehousing

Much of the conversation about what constitutes a data warehouse has been driven by two influential thought leaders – Bill Inmon and Ralph Kimball – who have different approaches to modeling and developing warehouses. Inmon defines a *data warehouse* as “a subject-oriented, integrated, time-variant and non-volatile collection of data in support of management’s decision-making process.”⁶⁴ A normalized relational model is used to store and manage data. Kimball defines a warehouse as “a copy of transaction data specifically structured for query and analysis.” Kimball’s approach calls for a dimensional model. (See Chapter 5.)

While Inmon and Kimball advocate different approaches to building warehouses, their definitions recognize similar core ideas:

- Warehouses store data from other systems
- The act of storage includes organizing the data in ways that increase its value
- Warehouses make data accessible and usable for analysis
- Organizations build warehouses because they need to make reliable, integrated data available to authorized stakeholders
- Warehouse data serves many purposes, from support of workflow to operational management to predictive analytics

1.3.5 Corporate Information Factory (Inmon)

Bill Inmon’s Corporate Information Factory (CIF) is one of the two primary patterns for data warehousing. The component parts of Inmon’s definition of a *data warehouse*, “a subject oriented, integrated, time variant, and nonvolatile collection of summary and detailed historical data,” describe the concepts that support the CIF and point to the differences between warehouses and operational systems.

- **Subject-oriented:** The data warehouse is organized based on major business entities, rather than focusing on a functional or application.
- **Integrated:** Data in the warehouse is unified and cohesive. The same key structures, encoding and decoding of structures, data definitions, naming conventions are applied consistently throughout the warehouse. Because data is integrated, Warehouse data is not simply a copy of operational data. Instead, the warehouse becomes a system of record for the data.

⁶⁴ <http://bit.ly/1FtgeIL>, last accessed 2/27/2016.

- **Time variant:** The data warehouse stores data as it exists in a set point in time. Records in the DW are like snapshots. Each one reflects the state of the data at a moment of time. This means that querying data based on a specific time period will always produce the same result, regardless of when the query is submitted.
- **Non-volatile:** In the DW, records are not normally updated as they are in operational systems. Instead, new data is appended to existing data. A set of records may represent different states of the same transaction.
- **Aggregate and detail data:** The data in the DW includes details of atomic level transactions, as well as summarized data. Operational systems rarely aggregate data. When warehouses were first established, cost and space considerations drove the need to summarize data. Summarized data can be persistent (stored in a table) or non-persistent (rendered in a view) in contemporary DW environments. The deciding factor in whether to persist data is usually performance.
- **Historical:** The focus of operational systems is current data. Warehouses contain historical data as well. Often they house vast amounts of it.

Inmon, Claudia Imhoff, and Ryan Sousa describe data warehousing in the context of the Corporate Information Factory (CIF). See Figure 80. CIF components include:

- **Applications:** Applications perform operational processes. Detail data from applications is brought into the data warehouse and the operational data stores (ODS) where it can be analyzed.
- **Staging Area:** A database that stands between the operational source databases and the target databases. The data staging area is where the extract, transform, and load effort takes place. It is not used by end users. Most data in the data staging area is transient, although typically there is some relatively small amount of persistent data.
- **Integration and transformation:** In the integration layer, data from disparate sources is transformed so that it can be integrated into the standard corporate representation / model in the DW and ODS.
- **Operational Data Store (ODS):** An ODS is integrated database of operational data. It may be sourced directly from applications or from other databases. ODS's generally contain current or near term data (30-90 days), while a DW contains historical data as well (often several years of data). Data in ODS's is volatile, while warehouse data is stable. Not all organizations use ODS's. They evolved as to meet the need for low latency data. An ODS may serve as the primary source for a data warehouse; it may also be used to audit a data warehouse.
- **Data marts:** Data marts provide data prepared for analysis. This data is often a sub-set of warehouse data designed to support particular kinds of analysis or a specific group of data consumers. For example, marts can aggregate data to support faster analysis. Dimensional modeling (using denormalization techniques) is often used to design user-oriented data marts.
- **Operational Data Mart (OpDM):** An OpDM is a data mart focused on tactical decision support. It is sourced directly from an ODS, rather than from a DW. It shares characteristics of the ODS: it contains current or near-term data. Its contents are volatile.

- **Data Warehouse:** The DW provides a single integration point for corporate data to support management decision-making, and strategic analysis and planning. The data flows into a DW from the application systems and ODS, and flows out to the data marts, usually in one direction only. Data that needs correction is rejected, corrected at its source, and ideally re-fed through the system.
- **Operational reports:** Reports are output from the data stores.
- **Reference, Master, and external data:** In addition to transactional data from applications, the CIF also includes data required to understand transactions, such as reference and Master Data. Access to common data simplifies integration in the DW. While applications consume current master and Reference Data, the DW also requires historical values and the timeframes during which they were valid (see Chapter 10).

Figure 80 depicts movement within the CIF, from data collection and creation via applications (on the left) to the creation of information via marts and analysis (on the right). Movement from left to right includes other changes. For example,

- The purpose shifts from execution of operational functions to analysis
- End users of systems move from front line workers to decision-makers
- System usage moves from fixed operations to ad hoc uses
- Response time requirements are relaxed (strategic decisions take more time than do daily operations)
- Much more data is involved in each operation, query, or process

The data in DW and marts differs from that in applications:

- Data is organized by subject rather than function
- Data is integrated data rather than 'siloed'
- Data is time-variant vs. current-valued only
- Data has higher latency in DW than in applications
- Significantly more historical data is available in DW than in applications

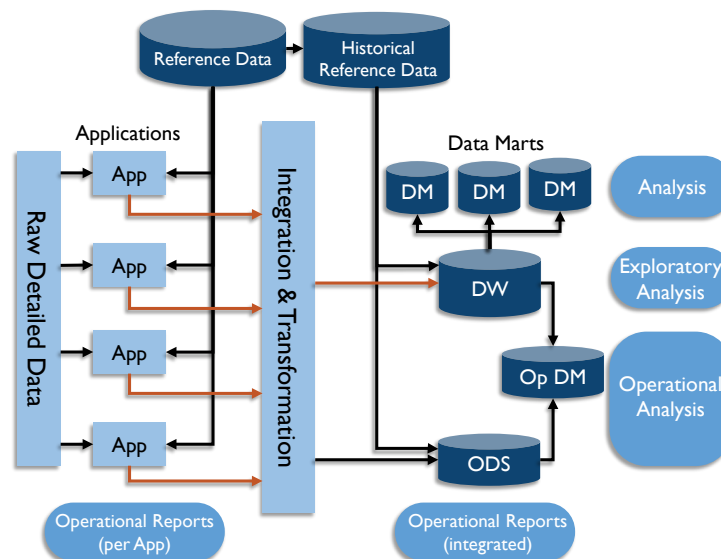


Figure 80 The Corporate Information Factory

1.3.6 Dimensional DW (Kimball)

Kimball's Dimensional Data Warehouse is the other primary pattern for DW development. Kimball defines a data warehouse simply as "a copy of transaction data specifically structured for query and analysis" (Kimball, 2002). The 'copy' is not exact, however. Warehouse data is stored in a dimensional data model. The dimensional model is designed to enable data consumers to understand and use the data, while also enabling query performance.⁶⁵ It is not normalized in the way an entity relationship model is.

Often referred to as *Star Schema*, dimensional models are comprised of *facts*, which contain quantitative data about business processes (e.g., sales numbers), and *dimensions*, which store descriptive attributes related to fact data and allow data consumers to answer questions about the facts (e.g., how many units of product X were sold this quarter?) A fact table joins with many dimension tables, and when viewed as a diagram, appears as a star. (See Chapter 5.) Multiple fact tables will share the common, or conformed, dimensions via a 'bus', similar to a bus in a computer.⁶⁶ Multiple data marts can be integrated at an enterprise level by plugging into the bus of conformed dimensions.

The DW bus matrix shows the intersection of business processes that generate fact data and data subject areas that represent dimensions. Opportunities for conformed dimensions exist where multiple processes use the same data. Table 27 is a sample bus matrix. In this example, the business processes for Sales, Inventory, and Orders all require Date and Product data. Sales and Inventory both require Store data, while Inventory and Orders require Vendor data. Date, Product, Store, and Vendor are all candidates for conformed dimensions. In contrast, Warehouse is not shared; it is used only by Inventory.

Table 27 DW-Bus Matrix Example

Business Processes	Subject Areas				
	Date	Product	Store	Vendor	Warehouse
Sales	X	X	X		
Inventory	X	X	X	X	X
Orders	X	X		X	
Conformed Dimension Candidate	Yes	Yes	Yes	Yes	No

The enterprise DW bus matrix can be used to represent the long-term data content requirements for the DW/BI system, independent of technology. This tool enables an organization to scope manageable development efforts. Each implementation builds an increment of the overall architecture. At some point, enough dimensional schemas exist to make good on the promise of an integrated enterprise data warehouse environment.

Figure 81 represents Kimball's Data Warehouse Chess Pieces view of DW/BI architecture. Note that Kimball's Data Warehouse is more expansive than Inmon's. The DW encompasses all components in the data staging and data presentation areas.

⁶⁵ <http://bit.ly/1udtNC8>.

⁶⁶ The term *bus* came from Kimball's electrical engineering background, where a bus was something providing common power to a number of electrical components.

- **Operational source systems:** Operational / transactional applications of the Enterprise. These create the data that is integrated into the ODS and DW. This component is equivalent to the application systems in the CIF diagram.
- **Data staging area:** Kimball's staging includes the set of processes needed to integrate and transform data for presentation. It can be compared to a combination of CIF's integration, transformation, and DW components. Kimball's focus is on efficient end-delivery of the analytical data, a scope smaller than Inmon's corporate management of data. Kimball's enterprise DW can fit into the architecture of the data staging area.
- **Data presentation area:** Similar to the Data Marts in the CIF. The key architectural difference being an integrating paradigm of a 'DW Bus,' such as shared or conformed dimensions unifying the multiple data marts.
- **Data access tools:** Kimball's approach focuses on end users' data requirements. These needs drive the adoption of appropriate data access tools.

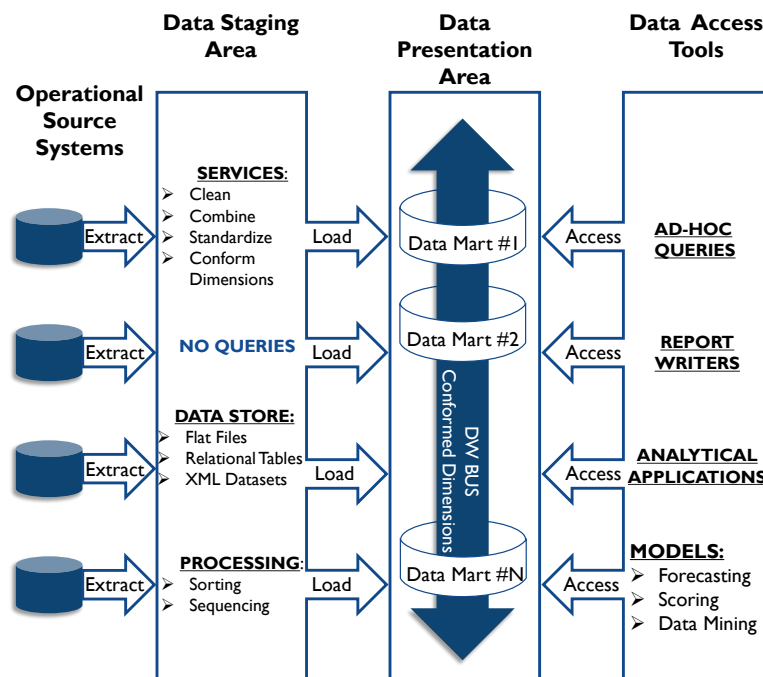


Figure 81 Kimball's Data Warehouse Chess Pieces⁶⁷

1.3.7 DW Architecture Components

The data warehouse environment includes a collection of architectural components that need to be organized to meet the needs of the enterprise. Figure 82 depicts the architectural components of the DW/BI and Big Data Environment discussed in this section. The evolution of Big Data has changed the DW/BI landscape by adding another path through which data may be brought into an enterprise.

⁶⁷ Adapted from Kimball and Ross (2002). Used with permission.

Figure 82 also depicts aspects of the data lifecycle. Data moves from source systems into a staging area where it may be cleansed and enriched as it is integrated and stored in the DW and/or an ODS. From the DW, it may be accessed via marts or cubes and used for various kinds of reporting. Big Data goes through a similar process but with a significant difference: while most warehouses integrate data before landing it in tables, Big Data solutions ingest data before integrating it. Big Data BI may include predictive analytics and data mining, as well as more traditional forms of reporting. (See Chapter 14.)

1.3.7.1 Source Systems

Source Systems, on the left side of Figure 82, include the operational systems and external data to be brought into the DW/BI environment. These typically include operational systems such as CRM, Accounting, and Human Resources applications, as well as operational systems that differ based on industry. Data from vendors and external sources may also be included, as may DaaS, web content, and any Big Data computation results.

1.3.7.2 Data Integration

Data integration covers Extract, Transform, and Load (ETL), data virtualization, and other techniques of getting data into a common form and location. In a SOA environment, the data services layers are part of this component. In Figure 82, all the arrows represent data integration processes. (See Chapter 8.)

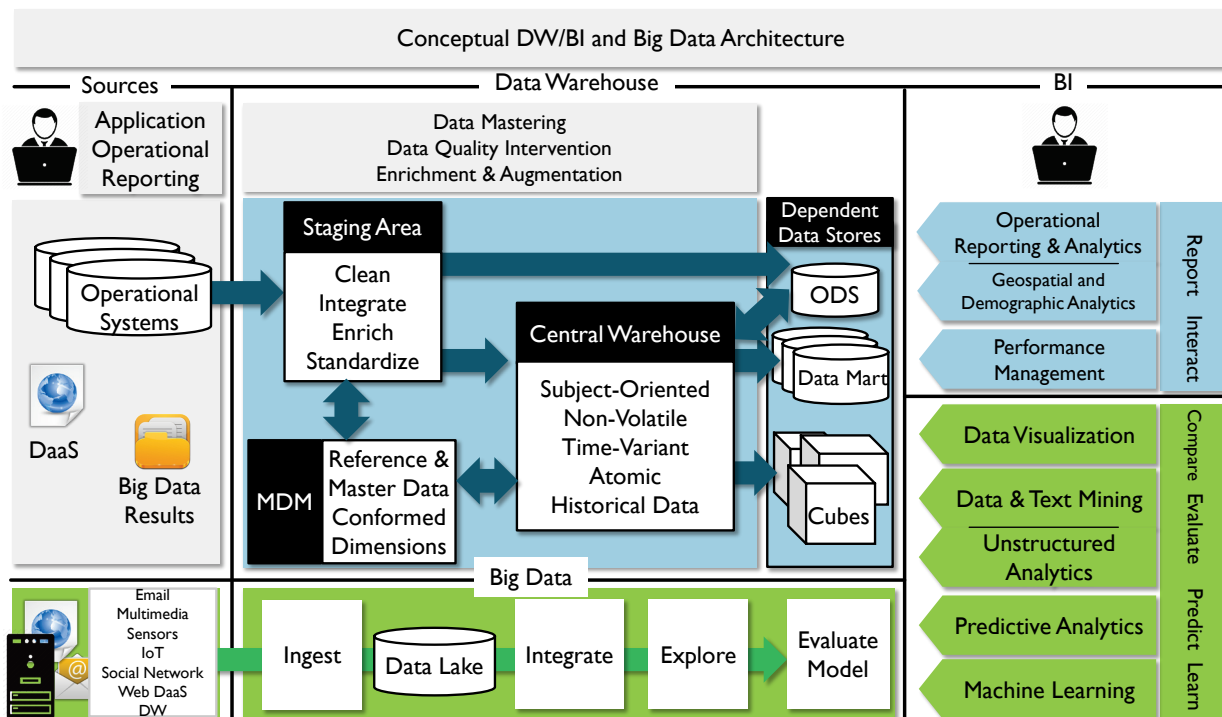


Figure 82 Conceptual DW/BI and Big Data Architecture

1.3.7.3 Data Storage Areas

The warehouse has a set of storage areas:

- **Staging area:** A staging area is an intermediate data store between an original data source and the centralized data repository. Data is staged so that it can be transformed, integrated, and prepped for loading to the warehouse.
- **Reference and Master Data conformed dimensions:** Reference and Master Data may be stored in separate repositories. The data warehouse feeds new Master Data and is fed by conformed dimension contents from the separate repositories.
- **Central Warehouse:** Once transformed and prepped, the DW data usually persists in the central or atomic layer. This layer maintains all historical atomic data as well as the latest instance of the batch run. The data structure of this area is developed and influenced based on performance needs and use patterns. Several design elements are brought to bear:
 - The relationship between the business key and surrogate keys for performance
 - Creation of indices and foreign keys to support dimensions
 - Change Data Capture (CDC) techniques that are used to detect, maintain, and store history
- **Operational Data Store (ODS):** The ODS is a version of a central persisted store that supports lower latencies and, therefore, operational use. Since the ODS contains a time window of data and not the history, it can be refreshed much more quickly than a warehouse. Sometimes real-time streams are snapshotted at predefined intervals into the ODS to enable integrated reporting and analysis. Over time, with the increasing frequency of updates driven by business needs, and growing technology and techniques to integrate real-time data into the DW, many installations have merged their ODS into the existing DW or Data Mart architecture.
- **Data marts:** A data mart is a type of data store often used to support presentation layers of the data warehouse environment. It is also used for presenting a departmental or functional sub-set of the DW for integrated reporting, query, and analysis of historical information. The data mart is oriented to a specific subject area, a single department, or a single business process. It can also form the basis of a virtualized warehouse where the combined marts comprise the resulting warehouse entity. Data integration processes will refresh, update, or expand the contents of the various marts from the persistence layer.
- **Cubes:** Three classic implementation approaches support Online Analytical Processing (OLAP). Their names relate to underlying database types, such as Relational, Multi-dimensional, and Hybrid.

1.3.8 Types of Load Processing

Data warehousing involves two main types of data integration processes: historical loads and ongoing updates. Historical data is usually loaded only once, or a few times while working out data issues, and then never again. Ongoing updates are consistently scheduled and executed to keep the data in the warehouse up-to-date.

1.3.8.1 Historical Data

One advantage of a data warehouse is that it can capture detailed history of the data it stores. There are different methods to capture this detail. An organization that wants to capture history should design based on requirements. Being able to reproduce point-in-time snapshots requires a different approach than simply presenting current state.

The Inmon data warehouse suggests that all data is stored in a single data warehouse layer. This layer will store cleansed, standardized, and governed atomic level data. A common integration and transformation layer facilitates reuse across the delivery implementations. An enterprise data model is required for success. Once validated, this single store is available to different data consumers via a star structured data mart.

The Kimball data warehouse suggests that the data warehouse is composed of a combination of departmental data marts containing cleansed, standardized, and governed data. The data marts will store the history at the atomic level. Conformed dimensions and conformed facts will deliver enterprise level information.

Another approach, the Data Vault, also cleanses and standardizes as part of the staging process. History is stored in a normalized atomic structure, dimensional surrogate, primary and alternate keys are defined. Ensuring that the business and surrogate key relationship remains intact becomes the secondary role of the vault – this is the data mart history. Facts are persisted here as atomic structures. The vault is then available to a variety of data consumers via data marts. By retaining the history inside the vault, reloading facts is possible when later increments introduce grain changes. It is possible to virtualize the presentation layer, facilitating agile incremental delivery and collaborative development with the business community. A final materialization process can implement a more traditional star data mart for production end user consumption.

1.3.8.2 Batch Change Data Capture

Data Warehouses are often loaded daily and serviced by a nightly batch window. The load process can accommodate a variety of change detection, as each source system may require differing change capture techniques.

Database log techniques are likely candidates for in-house developed applications as vendor purchased applications are unlikely to tolerate modification with triggers or additional overhead. Time stamped or log table loads are the most common. Full loads occur when dealing with legacy systems built without native time stamping capabilities (yes, there are applications without databases) or when certain batch recovery conditions apply. Table 28 summarizes the difference between change data capture techniques, including their relative complexity and speed. The overlap column identifies whether there may be data duplication between source system changes and the target environment. When Overlap is 'Yes', this change data may already be present. When the Delete indicator is set to 'Yes', the Change Data Method will track any deletes that have occurred in the source system – useful for expiring dimensions no longer in use. When Deletes are not tracked by the source system, additional efforts are required to determine when they occur. (See Chapter 8.)

Table 28 CDC Technique Comparison

Method	Source System Requirement	Complexity	Fact Load	Dimension Load	Overlap	Deletes
Time stamped Delta Load	Changes in the source system are stamped with the system date and time.	Low	Fast	Fast	Yes	No
Log Table Delta Load	Source system changes are captured and stored in log tables	Medium	Nominal	Nominal	Yes	Yes
Database Transaction Log	Database captures changes in the transaction log	High	Nominal	Nominal	No	Yes
Message Delta	Source system changes are published as [near] real-time messages	Extreme	Slow	Slow	No	Yes
Full Load	No change indicator, tables extracted in full and compared to identify change	Simple	Slow	Nominal	Yes	Yes

1.3.8.3 Near-real-time and Real-time

With the onset of Operational BI (or Operational Analytics) pushing for lower latency and more integration of real-time or near-real-time data into the data warehouse, new architectural approaches emerged to deal with the inclusion of volatile data. For example, a common application of operational BI is the automated banking machine data provisioning. When making a banking transaction, historical balances and new balances resulting from immediate banking actions need to be presented to the banking customer real-time. Two key design concepts that are required for provisioning data in near-real-time are isolation of change and alternatives to batch processing.

The impact of the changes from new volatile data must be isolated from the bulk of the historical, non-volatile DW data. Typical architectural approaches for isolation include a combination of building partitions and using union queries for the different partitions. Alternatives to batch processing handle the increasingly shorter latency requirements for data availability in the DW. There are three main types: trickle feeds, messaging, and streaming, which differ by where data is accumulated while waiting to be processed. (See Chapter 8.)

- **Trickle feeds (Source accumulation):** Rather than run on a nightly schedule, trickle feeds execute batch loads on a more frequent schedule (e.g., hourly, every 5 minutes) or when a threshold is reached (e.g., 300 transactions, 1G of data). This allows some processing to happen during the day, but not as intensely as with a dedicated nightly batch process. Care is needed to ensure that if a trickle feed batch takes longer to complete than the time between feeds, the next feed is delayed so that the data is still loaded in proper order.
- **Messaging (Bus accumulation):** Message interaction in real-time or near-real-time is useful when extremely small packets of data (messages, events, or transactions) are published to a bus as they occur. Target systems subscribe to the bus, and incrementally process the packets into the warehouse as needed. Source systems and target systems are independent of each other. Data-as-a-Service (DaaS) frequently uses this method.

- **Streaming (Target accumulation):** Rather than wait on a source-based schedule or threshold, a target system collects data as it is received into a buffer area or queue, and processes it in order. The result interaction or some aggregate may later appear as an additional feed to the warehouse.

2. Activities

2.1 Understand Requirements

Developing a data warehouse is different from developing an operational system. Operational systems depend on precise, specific requirements. Data warehouses bring together data that will be used in a range of different ways. Moreover, usage will evolve over time as users analyze and explore data. Take time in the initial phases to ask questions related to capabilities and sources of data to support these capabilities. This time to design pays off in reduced rework costs later when the data processing is being tested using the actual data sources.

In gathering requirements for DW/BI projects, begin with business goals and strategy. Identify and scope the business areas, then identify and interview the appropriate business people. Ask what they do and why. Capture specific questions they are asking now, and those they want to ask of the data. Document how they distinguish between and categorize important aspects of the information. Where possible, define and capture key performance metrics and calculations. These can uncover business rules that provide the foundation for automation of Data Quality expectations.

Catalog requirements and prioritize them into those necessary for production go-live and adoption of the warehouse and those that can wait. Look for items that are simple and valuable to jump-start the productivity of the initial project release. A DW/BI project requirements write-up should frame the whole context of the business areas and / or processes that are in scope.

2.2 Define and Maintain the DW/BI Architecture

The DW/BI architecture should describe where data comes from, where it goes, when it goes, why and how it goes into a warehouse. The 'how' includes the hardware and software detail and the organizing framework to bring all the activities together. Technical requirements should include performance, availability, and timing needs. (See Chapters 4 and 8.)

2.2.1 Define DW/BI Technical Architecture

The best DW/BI architectures will design a mechanism to connect back to transactional level and operational level reports in an atomic DW. This mechanism will protect the DW from having to carry every transactional detail. An example is providing a viewing mechanism for key operational reports or forms based on a transactional key, such as Invoice Number. Customers will always want all the detail available, but some of the

operational data, such as long description fields, has value only in the context of the original report, and does not provide analytic value.

A conceptual architecture is a starting point. Many activities are necessary to correctly align the non-functional requirements to the business needs. Prototyping can quickly prove or disprove key points before making expensive commitments to technologies or architectures. In addition, empowering the business community with knowledge and adoption programs championed through a sanctioned change management team will assist in transition and ongoing operational success.

A natural extension to this transformation process is the maintenance, or at least validation, with the enterprise data model. Since the focus is on which data structures are in use by which organizational areas, check the physical deployment against the logical model. Make any updates if omissions or errors arise.

2.2.2 Define DW/BI Management Processes

Address production management with a coordinated and integrated maintenance process, delivering regular releases to the business community.

It is crucial to establish a standard release plan (see Section 2.6). Ideally, the warehouse project team should manage each update to the deployed data product as a software release that provisions additional functionality. Establishing a schedule for releases allows for an annual demand and resource plan and standard delivery schedule. Use the internal release to tweak this standardized schedule, the resource expectations and estimate sheets derived for it.

Establishing a functioning release process ensures that management understands this to be a data product-centric proactive process and not an installed product addressed through reactive issue resolution. It is critical to work pro-actively and collaboratively in a cross-functional team to continuously grow and enhancement features – reactive support systems reduce adoption.

2.3 Develop the Data Warehouse and Data Marts

Typically, DW/BI projects have three concurrent development tracks:

- **Data:** The data necessary to support the analysis the business wants to do. This track involves identifying the best sources for the data and designing rules for how the data is remediated, transformed, integrated, stored, and made available for use by the applications. This step also includes deciding how to handle data that doesn't fit expectations.
- **Technology:** The back-end systems and processes supporting the data storage and movement. Integration with the existing enterprise is fundamental, as the warehouse is not an island unto itself. Enterprise Architectures, specifically Technology and Application specialties, usually manage this track.
- **Business Intelligence tools:** The suite of applications necessary for data consumers to gain meaningful insight from deployed data products.

2.3.1 Map Sources to Targets

Source-to-target mapping establishes transformation rules for entities and data elements from individual sources to a target system. Such mapping also documents lineage for each data element available in the BI environment back to its respective source(s).

The most difficult part of any mapping effort is determining valid links or equivalencies between data elements in multiple systems. Consider the effort to consolidate data into a DW from multiple billing or order management systems. Chances are that tables and fields that contain equivalent data do not have the same names or structures.

A solid taxonomy is necessary to map data elements in different systems to a consistent structure in the DW. Most often, this taxonomy is the logical data model. The mapping process must also address whether data in different structures is to be appended, changed in place, or inserted.

2.3.2 Remediate and Transform Data

Data remediation or cleansing activities enforce standards and correct and enhance the domain values of individual data elements. Remediation is particularly necessary for initial loads where significant history is involved. To reduce the complexity of the target system, source systems should be made responsible for data remediation and correction.

Develop strategies for rows of data that are loaded but found to be incorrect. A policy for deleting old records may cause some havoc with related tables and surrogate keys, expiring a row and loading the new data as a completely new row may be a better option.

An optimistic load strategy may include creating dimension entries to accommodate fact data. Such a process must account for how to update and expire such entries. Pessimistic load strategies should include a recycle area for fact data that cannot be associated with corresponding dimension keys. These entries require appropriate notification, alerting and reporting to ensure they are tracked, and reloaded later. Fact jobs should consider first loading recycled entries, then processing newly arrived content.

Data transformation focuses on activities that implement business rules within a technical system. Data transformation is essential to data integration. Defining the correct rules by which to integrate data often requires direct involvement from Data Stewards and other SMEs. Rules should be documented so that they can be governed. Data integration tools perform these tasks. (See Chapter 8.)

2.4 Populate the Data Warehouse

The largest part of the work in any DW/BI effort is the preparation and processing of the data. The design decisions and principles for what data detail the DW contains are a key design priority for DW/BI architecture. Publishing clear rules for what data will be available via only operational reporting (such as in non-DW) is critical to the success of DW/BI efforts.

The key factors to consider when defining a population approach are required latency, availability of sources, batch windows or upload intervals, target databases, dimensional aspects, and timeframe consistency of the data warehouse and data mart. The approach must also address Data Quality processing, time to perform transformations, and late-arriving dimensions and data rejects.

Another aspect to defining a population approach centers around change data capture process – detecting changes in the source system, integrating those changes together, and aligning changes across time. Several databases now provision log capture functionality that data integration tools can operate on directly, so the database tells the user what has changed. Scripting processes can be written or generated where this function is not available. Several techniques are available to the design and build teams for integration and latency alignment across heterogeneous feeds.

The first increment paves the way for additional capability development and onboarding new business units. Many new technologies, processes, and skills are necessary, as well as careful planning and attention to detail. Downstream increments are to build on top of this foundational element, so more investments are recommended to sustain high quality data, technical architecture, and transitioning to production. Create processes to facilitate and automate timely identification of data errors with end user workflow integration.

2.5 Implement the Business Intelligence Portfolio

Implementing the BI Portfolio is about identifying the right tools for the right user communities within or across business units. Find similarities through alignment of common business processes, performance analysis, management styles, and requirements.

2.5.1 Group Users According to Needs

In defining the target user groups, there is a spectrum of BI needs. First, know the user groups and then match the tool to the user groups in the company. On one end of the spectrum are IT developers concerned with extracting data, who focus on advanced functionality. On the other end, information consumers may want fast access to previously developed and executed reports. These consumers may want some degree of interactivity, such as drill, filter, sort, or may only want to see a static report.

Users may move from one class to another as their skills increase or as they perform different functions. A supply chain manager, for example, may want to view a static report on financials but a highly interactive report for analyzing inventory. A financial analyst and a line manager responsible for expenses may be power users when analyzing total expenses, but are satisfied with a static report of one phone bill. Executives and managers will use a combination of fixed reports, dashboards, and scorecards. Managers and power users tend to want to drill into these reports slice and dice the data to identify the root causes of problems. External customers may use any of these tools as part of their experience.

2.5.2 Match Tools to User Requirements

The marketplace offers an impressive range of reporting and analytics tools. Major BI vendors offer classic pixel-perfect report capabilities that were once the domain of application reports. Many application vendors offer embedded analytics with standard content fetched from pre-populated cubes or aggregate tables. Virtualization has blurred the lines between on-premises data sources and external purchased or open data, and in some cases provisions user-controlled report-centric integration on demand. In other words, it is prudent for companies to use common infrastructure and delivery mechanisms. These include the web, email, and applications for the delivery of all kinds of information and reports, of which DW/BI is a subset.

Many vendors are now combining related BI Tools, through mergers and acquisitions or net new development, and offering BI Suites. Suites are the primary option at the Enterprise Architecture level but given that most organizations have already purchased individual tools, or embraced open source tools, questions around replacement versus co-existence are likely to surface. Remember that every BI tool comes with a price, requiring system resources, support, training, and architectural integration.

2.6 Maintain Data Products

An implemented warehouse and its customer-facing BI tools is a data product. Enhancements (extensions, augmentations, or modifications) to an existing DW platform should be implemented incrementally.

Maintaining the scope for an increment, and executing a critical path for key work items, can be a challenge in a dynamic work environment. Set priorities with business partners and focus work on mandatory enhancements.

2.6.1 Release Management

Release Management is critical to an incremental development processes that grows new capabilities, enhances the production deployment, and ensures provision of regular maintenance across the deployed assets. This process will keep the warehouse up-to-date, clean, and operating at its best. However, this process requires the same alignment between IT and Business as between the Data Warehouse model and the BI capabilities. It is a continual improvement effort.

Figure 83 illustrates an example release process, based on a quarterly schedule. Over the year, there are three business-driven releases and one technology-based release (to address requirements internal to the warehouse). The process should enable incremental development of the warehouse and management of the backlog of requirements.

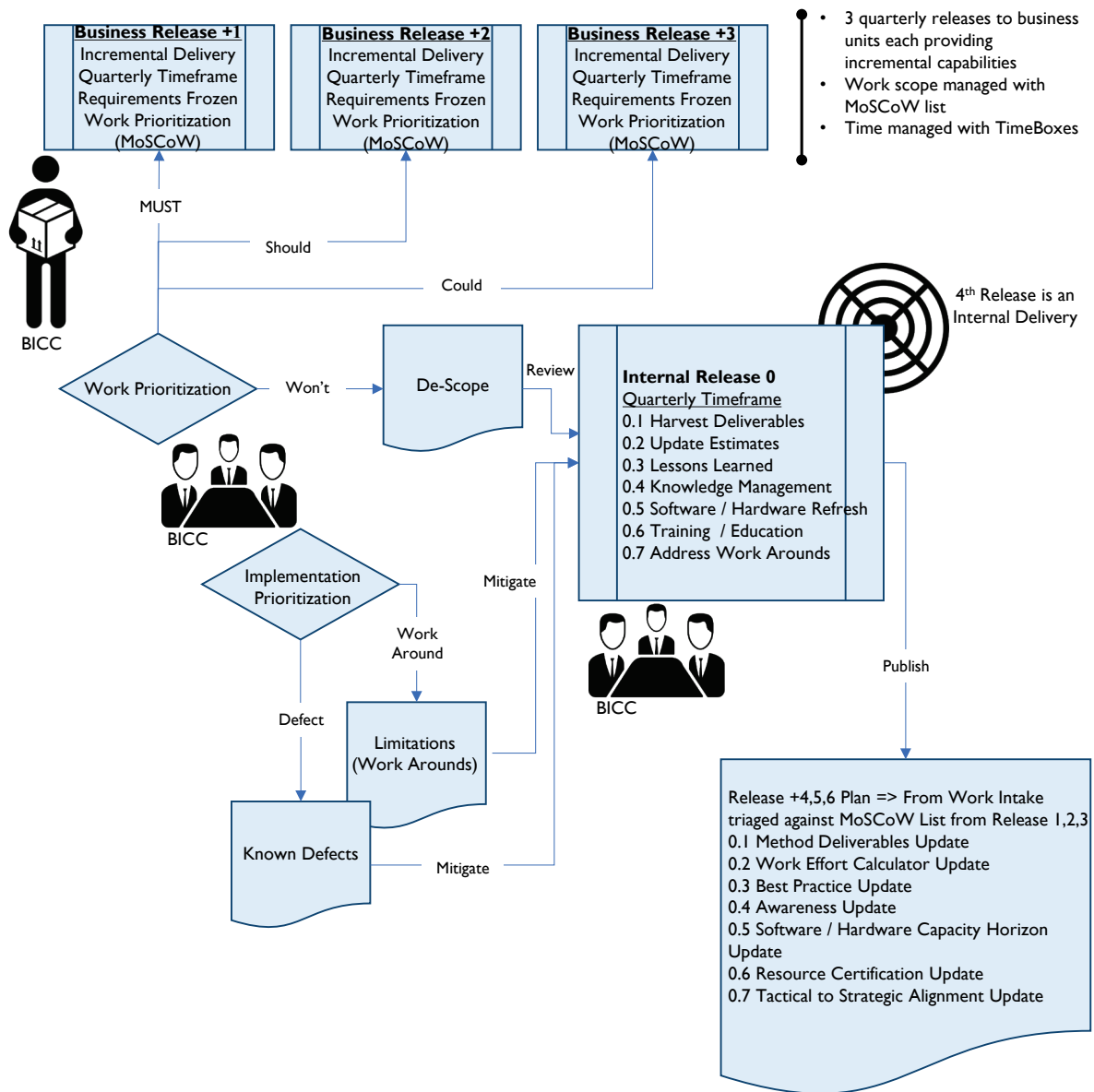


Figure 83 Release Process Example

2.6.2 Manage Data Product Development Lifecycle

While data consumers are using the existing DW, the DW team is preparing for the next iteration, with the understanding that not all items will go to production. Align the iterations to releases with a backorder work list prioritized by the business units. Each iteration will extend an existing increment or add new functionality by onboarding a business unit. Releases will align functionality to the business unit, whereas the iteration will align the functionality to the configuration itself managed by the product manager.

Those items that business believes ready and feasible for further investigation can be reviewed, adjusted if necessary, and then promoted to a pilot or sandbox environment, where business users investigate new approaches, experiment with new techniques, or develop new models or learning algorithms. This area may

see less governance and supervision than other business-facing areas but some form of sandbox prioritizing is necessary.

Akin to the traditional quality assurance or testing environment, scrutinize items in the pilot area for fit into the production world. How well pilot items perform determines their next steps. Take care not to promote blindly and without regard to downstream Data Quality or governance issues. The lifespan in production is just an existential measure: it must be of the highest practical quality to be in production.

Items passing the pilot and deemed production-ready by both business and IT representatives can be promoted to production as new data products. This completes one iteration.

Items not passing pilot can be rejected entirely or returned to development for fine-tuning. Perhaps additional support from the DW team is needed at this time to advance the item in the next promotion iteration.

2.6.3 Monitor and Tune Load Processes

Monitor load processing across the system for bottlenecks and dependencies. Employ database tuning techniques where and when needed, including partitioning, tuned backup, and recovery strategies. Archiving is a difficult subject in data warehousing.

Users often consider the data warehouse as an active archive due to the long histories that are built and are unwilling, particularly if the On Line Analytical Processing (OLAP) sources have dropped records, to see the data warehouse engage in archiving. (See Chapter 6.)

2.6.4 Monitor and Tune BI Activity and Performance

A best practice for BI monitoring and tuning is to define and display a set of customer-facing satisfaction metrics. Average query response time and the number of users per day, week, or month are examples of useful metrics. In addition to the statistical measures available from the systems, it is useful to survey DW/BI customers regularly.

Regular review of usage statistics and patterns is essential. Reports providing frequency and resource usage of data, queries, and reports allow prudent enhancement. Tuning BI activity is analogous to the principle of profiling applications in order to know where the bottlenecks are and where to apply optimization efforts. The creation of indexes and aggregations is most effective when done according to usage patterns and statistics. Tremendous performance gains can come from simple solutions such as posting the completed daily results to a report that runs hundreds or thousands of times a day.

Transparency and visibility are the key principles that should drive DW/BI monitoring. The more one can expose the details of the DW/BI activities, the more data consumers can see and understand what is going on (and have confidence in the BI), and less direct end-customer support will be required. Providing a dashboard that exposes the high-level status of data delivery activities, with drill-down capability, is a best practice that allows an on-demand-pull of information by both support personnel and customers.

The addition of Data Quality measures will enhance the value of this dashboard where performance is more than just speed and timing. Use heat maps to visualize workload on infrastructure, data throughput, and compliance to operating agreement levels.

3. Tools

Choosing the initial set of tools can be a long process. It includes attempting to satisfy near-term requirements, non-functional specifications, and the as-yet to be created next generation requirements. Decision criteria tool sets, process implementation tools, and professional services offerings can facilitate and expedite this activity. It is critical to evaluate not only the conventional build or buy positions, but also the rent option provisioned as Software-as-a-Service. Renting SaaS tools and the associated expertise is weighed against the cost of building from scratch or deploying purchased products from vendors. Consider ongoing upgrade and potential replacement costs as well. Alignment to a set OLA (Operational Level Agreement) can bridge forecasted costs, and provide input into setting compelling fees and penalties for term violations.

3.1 Metadata Repositories

Large organizations often find themselves with many tools from different vendors, each deployed potentially at differing versions. Key to this effort is the ability to stitch Metadata together from a variety of sources. Automating and integrating population of this repository can be achieved with a variety of techniques. (See Chapter 13.)

3.1.1 Data Dictionary / Glossary

A data dictionary is necessary to support the use of a DW. The dictionary describes data in business terms and includes other information needed to use the data (e.g., data types, details of structure, and security restrictions). Often, the content for the data dictionary comes directly from the logical data model. Plan for high quality Metadata by ensuring modelers take a disciplined approach to managing definitions as part of the modeling process.

In some organizations, business users actively participate in the development of the data dictionary by supplying, defining, and then stewarding corrections to definitions of subject area data elements. Embrace this activity through a collaboration tool, monitor activities through a Center of Excellence, and ensure that content created through this activity is retained in the logical model. Ensuring agreement between the business-facing content and the technical-facing physical data model will reduce the risk of downstream errors and rework. (See Chapter 13.)

3.1.2 Data and Data Model Lineage

Many data integration tools offer lineage analysis that considers both the developed population code and the physical data model and database. Some offer web interfaces to monitor and update definitions and other Metadata. Documented data lineage serves many purposes:

- Investigation of the root causes of data issues
- Impact analysis for system changes or data issues
- Ability to determine the reliability of data, based on its origin

Look to implement an integrated impact and lineage tool that can understand all the moving parts involved in the load process, as well as end user reporting and analytics. Impact analysis reports will outline which components are affected by a potential change, expediting and streamlining estimating and maintenance tasks.

Many key business processes, relationships, and terminologies are captured and explained during development of the data model. The logical data model holds much of this information, which is often lost or ignored during development or production deployment. It is critical to ensure that this information is not discarded and that the logical and physical models are updated after deployment and are in sync.

3.2 Data Integration Tools

Data integration tools are used to populate a data warehouse. In addition to doing the work of integrating data, they enable scheduling of jobs in ways that account for complex data delivery from multiple sources. In selecting a tool, also account for these features that enable management of the system:

- Process audit, control, restart, and scheduling
- The ability to selectively extract data elements at execution time and pass that extract to a downstream system for audit purposes
- Controlling which operations can or cannot execute and restarting a failed or aborted run (see Chapter 8)

A variety of data integration tools also offer integration capabilities with the BI portfolio, supporting import and export of workflow messages, email, or even semantic layers. Workflow integration can drive Data Quality defect identification, resolution, and escalation processes. Messaging through email or alert processing driven from email is a common practice especially for mobile devices. In addition, the ability to provision a data target as a semantic layer can be a data virtualization candidate for agile implementations.

3.3 Business Intelligence Tools Types

The maturity of the BI market, and a wide range of available BI tools, makes it rare for companies to build their own BI tools.⁶⁸ The purpose of this section is to introduce the types of tools available in the BI marketplace, and provide an overview of their chief characteristics with information to help match the tools to the

⁶⁸ The material in this section is primarily from “The Business Intelligence Market” by Cindi Howson, BIScorecard®, <http://bit.ly/2tNirv5>; used by permission, with minor changes and additions.

appropriate customer-level capabilities. BI tools are evolving quickly, enabling a transition from IT-led, standardized reporting to self-service, business-driven data exploration.⁶⁹

- **Operational reporting** is the application of BI tools to analyze business trends, both short-term (month-over-month) and longer-term (year-over-year). Operational reporting can also help discover trends and patterns. Use Tactical BI to support short-term business decisions.
- **Business performance management (BPM)** includes the formal assessment of metrics aligned with organizational goals. This assessment usually happens at the executive level. Use Strategic BI to support long-term corporate goals and objectives.
- **Descriptive, self-service analytics** provides BI to the front lines of the business, where analytical capabilities guide operational decisions. Operational analytics couples BI applications with operational functions and processes, to guide decisions in near-real-time. The requirement for low latency (near real-time data capture and data delivery) will drive the architectural approach to operational analytics solutions. Service-oriented Architecture (SOA) and Big Data become necessary to support operational analytics fully (see Chapters 8 and 15).

3.3.1 Operational Reporting

Operational Reporting involves business users generating reports directly from transactional systems, operational applications, or a data warehouse. This is typically an application functionality. Often business areas will start to use a DW for operational reporting, especially if DW/BI governance is poor, or the DW contains additional data that enhances the operational, transaction data. Often the reports will appear as ad-hoc queries, when in fact they are simple reports or are used to initiate workflow. From a data management perspective, the key is to understand if the data necessary for this reporting exists within the application itself, or if it requires data enhancements from the DW or operational data store.

Data exploration and reporting tools, sometimes called ad-hoc query tools, enable users to author their own reports or create outputs for use by others. They are less concerned with the precise layout because they are not trying to generate an invoice or the like. However, they do want to include charts and tables quickly and intuitively. Often the reports created by business users become standard reports, not exclusively used for ad hoc business questions.

The needs within business operations reporting are often different from the needs within business query and reporting. With business query and reporting, the data source is usually a data warehouse or data mart (though not always). While IT develops production reports, power users and ad hoc business users develop their own reports with business query tools. Use reports generated with business query tools individually, departmentally, or enterprise-wide.

⁶⁹ Dataversity refers to this trend as the “democratization of data technologies.” See Ghosh, Paramita. “A Comparative Study of Business Intelligence and Analytics Market Trends.” Dataversity. January 17, 2017. <http://bit.ly/2sTgXTJ> (accessed 2017-01-22).

Production reporting crosses the DW/BI boundary and often queries transactional systems to produce operational items such as invoices or bank statements. The developers of production reports tend to be IT personnel.

Traditional BI tools cover some data visualization methods such as tables, pie charts, line charts, area charts, bar charts, histograms, turnkey box (candlestick) as examples fairly well. Data visualizations can be delivered in a static format, such as a published report, or a more interactive online format; and some support end-user interaction where drilling or filtering capabilities facilitate analysis of data within the visualization. Others allow the visualization to be changed by the user on demand. (See Chapter 14.)

3.3.2 Business Performance Management

Performance management is a set of integrated organizational processes and applications designed to optimize execution of business strategy; applications include budgeting, planning, and financial consolidation. There have been a number of major acquisitions in this segment, as ERP vendors and BI vendors see great growth opportunities here and believe BI and Performance Management are converging. How frequently customers buy BI and performance management from the same vendor depends on product capabilities.

Broadly speaking, Performance Management technology enables processes to help meet organizational goals. Measurement and a feedback loop with positive reinforcement are key elements. Within the BI space, this has taken the form of many strategic enterprise applications, such as budgeting, forecasting, or resource planning. Another specialization has formed in this area: creating scorecards driven by dashboards for user interaction. Dashboards, like those found in automobiles, provide the necessary summary or aggregate information to the end user with most recent updates (Eckerson, 2005).

3.3.3 Operational Analytic Applications

Henry Morris of IDC coined the term *Analytic Applications* in the 1990s, clarifying how they are different from general OLAP and BI tools (Morris, 1999). Analytic applications include the logic and processes to extract data from well-known source systems, such as vendor ERP systems, a data model for the data mart, and pre-built reports and dashboards. They provide businesses with a pre-built solution to optimize a functional area (people management, for example) or industry vertical (retail analytics, for example). Different types of analytic applications include customer, financial, supply chain, manufacturing, and human resource applications.

3.3.3.1 Multi-dimensional Analysis – OLAP

Online Analytical Processing (OLAP) refers to an approach to providing fast performance for multi-dimensional analytic queries. The term OLAP originated, in part, to make a clear distinction from OLTP, Online Transactional Processing. The typical output of OLAP queries is in a matrix format. The dimensions form the rows and columns of the matrix, and the factors, or measures, are the values inside the matrix. Conceptually, this illustrates as a cube. Multi-dimensional analysis with cubes is particularly useful where there are well-known ways analysts want to look at summaries of data.

A traditional application is financial analysis, where analysts want to repeatedly traverse known hierarchies to analyze data; for example, date (such as Year, Quarter, Month, Week, Day), organization (such as Region, Country, Business Unit, Department), and product hierarchy (such as Product Category, Product Line, Product). Many tools today embed OLAP cubes into their software footprint and some even seamlessly automate and integrate the definition and population process. This means that any user in any business process can slice and dice their data. Align this capability with the power users in the subject area communities and deliver it along a self-service channel empowering these selected users to analyze their data their way.

Typically, OLAP tools have both a server component and an end user client-facing component installed on the desktop, or available on the web. Some desktop components are accessible from within a spreadsheet appearing as an embedded menu or function item. The architecture selected (ROLAP, MOLAP, HOLAP) will guide the development efforts but common to all will be definition of cube structure, aggregate needs, Metadata augmentation and analysis of data sparsity.

Structuring the cube to provision desired functional requirements may require splitting larger dimensions into separate cubes to accommodate storage, population, or calculation requirements. Use levels of aggregation to ensure calculation and retrieval of desired formulas occurs within agreed upon response times. End user augmentation of hierarchies enable fulfillment the aggregation, calculation, or population requirements. In addition, sparsity of cube data may require addition or removal of aggregate structures or refine materialization needs in the warehouse data layer provisioning it.

Provisioning role-based security or multi-language text within the cube may require extra dimensions, additional functions, calculations, or sometimes creating separate cube structures. Striking a balance between end user flexibility, performance, and server workloads means some negotiating is to be expected. The negotiation typically occurs during the loading processes and may require hierarchy changes, aggregate structure changes or additional warehouse materialized data objects. Strike the right balance among cube count, server workload, and delivered flexibility, so that the refresh occurs in a timely manner, and cubes provide reliable and consistent queries without high storage or server utilization costs.

The value of On Line Analytical Processing (OLAP) Tools and cubes is reduction of the chance of confusion and erroneous interpretation, by aligning the data content with the analyst's mental model. The analyst can navigate through the database and screen for a particular subset of the data, changing the data's orientation and defining analytical calculations. Slice-and-dice is the user-initiated process of navigation by calling for page displays interactively, through the specification of slices via rotations and drill down / up. Common OLAP operations include slice and dice, drill down, drill up, roll up, and pivot.

- **Slice:** A slice is a subset of a multi-dimensional array corresponding to a single value for one or more members of the dimensions not in the subset.
- **Dice:** The dice operation is a slice on more than two dimensions of a data cube, or more than two consecutive slices.
- **Drill down / up:** Drilling down or up is a specific analytical technique whereby the user navigates among levels of data, ranging from the most summarized (up) to the most detailed (down).
- **Roll-up:** A roll-up involves computing all of the data relationships for one or more dimensions. To do this, define a computational relationship or formula.

- **Pivot:** A pivot changes the dimensional orientation of a report or page display.

Three classic implementation approaches support Online Analytical Processing.

- **Relational Online Analytical Processing (ROLAP):** ROLAP supports OLAP by using techniques that implement multi-dimensionality in the two-dimensional tables of relational database management systems (RDBMS). Star schema joins are a common database design technique used in ROLAP environments.
- **Multi-dimensional Online Analytical Processing (MOLAP):** MOLAP supports OLAP by using proprietary and specialized multi-dimensional database technology.
- **Hybrid Online Analytical Processing (HOLAP):** This is simply a combination of ROLAP and MOLAP. HOLAP implementations allow part of the data to be stored in MOLAP form and another part of the data to be stored in ROLAP. Implementations vary on the control a designer has to vary the mix of partitioning.

4. Techniques

4.1 Prototypes to Drive Requirements

Quickly prioritize requirements before the implementation activities begin by creating a demonstration set of data and applying discovery steps in a joint prototype effort. Advances in data virtualization technologies can alleviate some of the traditional implementation pains through collaborative prototyping techniques.

Profiling the data contributes to prototyping and helps reduce risk associated with unexpected data. The DW is often the first place where the pain of poor quality data in source systems or data entry functions becomes apparent. Profiling also discloses differences between sources that may present obstacles to data integration. Data may be of high quality within its sources, but because sources differ, the data integration process becomes more complicated.

Evaluation of the state of the source data leads to more accurate up-front estimates for feasibility and scope of effort. The evaluation is also important for setting appropriate expectations. Plan to collaborate with the Data Quality and Data Governance team(s) and to draw on the expertise of other SMEs to understand data discrepancies and risks. (See Chapters 11 and 13.)

4.2 Self-Service BI

Self-service is a fundamental delivery channel within the BI portfolio. This typically funnels user activity within a governed portal where, depending on the privileges of the user, a variety of functionality is provided ranging from messaging, alerts, viewing scheduled production reports, interacting with analytic reports, developing ad hoc reporting and of course dash boarding and score carding. Reports can be pushed to the portal on standard

schedules, to be retrieved by the users at their leisure. Users can also pull data by executing reports from within the portal. These portals share content across organizational boundaries. Extending the collaboration tool outwards toward the user community can also provide self-service tips and tricks, an integrated communicate on load status, overall performance, and release progress as well as dialog forums. Mediate forum content through the support channel and then facilitate with user group sessions through the maintenance channel.

Visualization and statistical analysis tooling allows for rapid data exploration and discovery. Some tools allow for business-centric construction of dashboard like objects that can be rapidly shared, reviewed, and revitalized. Once the domain of IT and developers only, many data shaping, calculation, and visualization techniques can now be employed by the business community. This offers a degree of workload distribution and integration efforts can be feasibly prototyped through business channels and then materialized and optimized by IT.

4.3 Queryable Data

In order to maintain lineage, all structures and processes should have the capability to create and store audit information at a grain useful for tracking and reporting. Allowing users to query this audit data enables the users to verify for themselves the condition and arrival of the data, which improves user confidence. Audit information also allows for more detailed trouble-shooting when data issues arise.

5. Implementation Guidelines

A stable architecture that can scale to meet future requirements is paramount to the success of a data warehouse. A production support team capable of dealing with the daily loading, analysis, and end user feedback is mandatory. In addition, to sustain success, ensure that the warehouse and the business unit teams are aligned.

5.1 Readiness Assessment / Risk Assessment

There may be a gap between when an organization embraces a new venture, and when it has the ability to sustain that venture. Successful projects start with a Prerequisite Checklist. All IT projects should have business support, be aligned with strategy, and have a defined architectural approach. In addition, a DW should:

- Define data sensitivity and security constraints
- Perform tool selection
- Secure resources
- Create an ingestion process to evaluate and receive source data

Identify and inventory sensitive or restricted data elements in the warehouse. This data will need to be masked or obfuscated to prevent access by unauthorized personnel. Additional constraints may apply when considering outsourcing for implementation or maintenance activities.

Account for security constraints before selecting tools and assigning resources. Ensure Data Governance processes for review and approval have been followed. DW/BI projects risk refocus or total cancellation due to these overarching factors.

5.2 Release Roadmap

Because they require a large development effort, warehouses are built incrementally. Whatever method chosen to implement, be it waterfall, iterative or agile, it should account for the desired end state. That is why a roadmap is a valuable planning tool. The method combined with the maintenance processes can be both flexible and adaptive to balance the pressures of individual project delivery with overall goals of re-usable data and infrastructure.

An incremental approach leveraging the DW bus matrix as a communication and marketing tool is suggested. Use business-determined priorities tethered by exposure metrics to determine how much rigor and overhead to apply to each increment; a small single-sourced delivery may afford rule relaxation especially when limited exposure is felt should those issues be realized by the organization. Each increment will modify existing capabilities or add brand new capabilities typically aligned with a newly onboarded business unit. Apply a consistent needs and abilities process to determine the next business unit to onboard. Maintain a back-order or work item list to identify outstanding capabilities and the business-facing priorities. Determine any technical dependencies that require delivery in a different order. Then package this work into a software release. Each release can be delivered at an agreed-upon pace: quarterly, monthly, weekly, or even faster when appropriate. Manage the releases with the business partners by assembling a roadmap: a listing of releases by date by capabilities.

5.3 Configuration Management

Configuration management aligns with the release roadmap and provides the necessary back office stitching and scripts to automate development, testing, and transportation to production. It also brands the model by the release at the database level, and ties the codebase to that brand in an automated manner so that manually coded, generated programs and semantic layer content is harmonized across the environment and is versioned controlled.

5.4 Organization and Cultural Change

Starting with and keeping a consistent business focus throughout the DW/BI lifecycle is essential to success. Looking at the value chain of the enterprise is a good way to understand the business context. The specific business processes in a company's value chain provide a natural business-oriented context in which to frame areas of analysis.

Most importantly, align projects behind real business needs and assess the necessary business support, considering these critical success factors:

- **Business sponsorship:** Is there appropriate executive sponsorship, i.e., an identified and engaged steering committee and commensurate funding? DW/BI projects require strong executive sponsorship.
- **Business goals and scope:** Is there a clearly identified business need, purpose, and scope for the effort?
- **Business resources:** Is there a commitment by business management to the availability and engagement of the appropriate business subject matter experts? The lack of commitment is a common point of failure and a good enough reason to halt a DW/BI project until commitment is confirmed.
- **Business readiness:** Is the business partner prepared for a long-term incremental delivery? Have they committed themselves to establishing centers of excellence to sustain the product in future releases? How broad is the average knowledge or skill gap within the target community and can that be crossed within a single increment?
- **Vision alignment:** How well does the IT Strategy support the Business Vision? It is vital to ensure that desired functional requirements correspond to business capabilities that are or can be sustained in the immediate IT roadmap. Any significant departures or material gaps in capability alignment can stall or stop a DW/BI program.

5.4.1 Dedicated Team

Many organizations have a dedicated team to manage the ongoing operations of the production environment. (See Chapter 6). A separate set of hands operating the delivered data product is beneficial to workload optimization as this group has repeating tasks on a calendar cycle and may be further used for any escalation items whereas the maintenance channel will see workload spikes aligned to specific deliverables.

A front office support group interacts with the maintenance team to foster inter-department relationships and ensure critical activities are addressed in upcoming releases. It notifies the team of any deficiencies to be addressed. A back office support team in operations will ensure that production configuration has executed as required. They will escalate alerts and report on throughput status.

6. DW/BI Governance

Industries that are highly regulated and need compliance-centric reporting will benefit greatly from a well-governed data warehouse. Critical to ongoing support and vital to release planning is ensuring that governance activities are completed and addressed during the implementation. More and more organizations are extending their Software Development Lifecycle with specific deliverables aimed at addressing governance needs. Warehouse governance processes should be aligned with risk management. They should be business-driven,

since different kinds of businesses have different needs (e.g., marketing and advertising companies will use their data differently from financial institutions). Governance processes should mitigate risk, not curtail execution.

The most critical functions are those that govern the business-operated discovery or refinement area, and those that ensure pristine quality within the warehouse itself. As the refinement area leads all initiative boundaries, handshaking and well running procedures are necessary to instantiate, operate, transfer, and discard the data in these areas. Data archival and time horizons are key elements in boundary agreements as they help avoid sprawl. Monitoring of these environments and schedules to determine longevity terms are included in user group sessions as well as management meetings. Loading data into the warehouse means assigning time, resources, and programming efforts to see remediated, credible, high quality data arrive to the end user community, in a timely manner of course.

Consider one-off or limited-use events as part of the lifecycle, and perhaps curtail them within the pilot area itself, or within a user-controlled 'sandbox' area. Real-time analysis processes can feed time-aligned aggregate results back into the data warehouse through an automated process. Policy is defined for the procedures enacted upon the real-time environment, and governance applies to the brokerage of results into the warehouse for organizational consumption.

Apply data discrimination to known or cataloged items managed through a risk exposure mitigation matrix. Those items with a deemed high exposure, and low mitigation or difficult early detection, warrant governance functions to curtail the associated risk. Depending on the sensitivity of the data being examined, a separate workspace for selected local personnel may also be required. A thorough review with corporate security and legal personnel during policy formation creates a final safety net.

6.1 Enabling Business Acceptance

A key success factor is business acceptance of data, including the data being understandable, having verifiable quality, and having a demonstrable lineage. Sign-off by the Business on the data should be part of the User Acceptance Testing. Perform structured random testing of the data in the BI tool against data in the source systems over the initial load, and after a few update load cycles, to meet sign-off criteria. Meeting these requirements is paramount for every DW/BI implementation. Consider, up-front, a few critically important architectural sub-components, along with their supporting activities:

- **Conceptual Data Model:** What information is core to the organization? What are the key business concepts and how are they related to each other?
- **Data Quality feedback loop:** How are data issues identified and remediated? How are owners of the systems in which issues originate informed about problems and held accountable for fixing them? What is the remediation process for issues that are caused by the DW data integration processes?
- **End-to-end Metadata:** How does the architecture support the integrated end-to-end flow of Metadata? In particular, is access to meaning and context designed into the architecture? How do data consumers answer basic questions like "What does this report mean?" or "What does this metric mean?"

- **End-to-end verifiable data lineage:** Are the items exposed to business users traceable to the source systems in an automated, maintained manner? Is a system of record identified for all data?

6.2 Customer / User Satisfaction

Perceptions of the quality of data will drive customer satisfaction but satisfaction is dependent on other factors as well, such as data consumers' understanding of the data and the operations team's responsiveness to identified issues. Collecting, understanding, and acting on customer feedback can be facilitated through regularly scheduled meetings with user representatives. Such interaction can also help the warehouse team share information about the release roadmap and understand how data consumers are using the warehouse.

6.3 Service Level Agreements

Business and technical expectations for the environments should be specified in Service Level Agreements (SLAs). Often, the response time, data retention, and availability requirements differ greatly between classes of business needs and their respective supporting systems (e.g., ODS versus DW versus data mart).

6.4 Reporting Strategy

Ensure that a reporting strategy exists within and across the BI Portfolio. A reporting strategy includes standards, processes, guidelines, best practices, and procedures. It will ensure users have clear, accurate, and timely information. The reporting strategy must address

- Security access to ensure that only entitled users will gain access to sensitive data elements
- Access mechanisms to describe how users want to interact, report, examine, or view their data
- User community type and appropriate tool to consume it with
- Nature of the reports summary, detailed, exception as well as frequency, timing, distribution, and storage formats
- Potential use of visualization capabilities to provision graphical output
- Trade-offs between timeliness and performance

Standard reports should be evaluated periodically to ensure they are still providing value, as just executing reports incurs cost in storage and processing. Implementation and maintenance processes and management activities are critical. Aligning the appropriate reporting tools to the business community is a critical success factor. Depending on the size and nature of the organization, there are probably many different reporting tools used in a variety of processes. Ensure that the audience is capable of making the best use of the reporting tools; users that are more sophisticated will have increasingly complex demands. Maintain a decision matrix based on these demands to determine upgrades or future tool selection.

Data source governance monitoring and control are also vital. Ensure that appropriate levels of data are provisioned securely for authorized personnel, and that subscription data is accessible according to agreed-upon levels.

A Center of Excellence can provide training, start-up sets, design best practices, data source tips and tricks and other point solutions or artifacts to help empower business users towards a self-service model. In addition to knowledge management, this center can provide timely communications across the developer, designer, analyst and subscribing user communities.

6.5 Metrics

6.5.1 Usage Metrics

DW usage metrics typically include the number of registered users, as well as connected users or concurrent connected users. These metrics show how many people within the organization are using the data warehouse. How many user accounts are licensed for each tool is a great start, especially for the auditors. However, how many actually connect with that tool is a better measurement, and how many queries (or query equivalents) are dispatched by a user community per timeframe is an even better technical measurement, especially for capacity planning. Allow for multiple analysis metrics such as audit users, generated user query capacity, and consuming users.

6.5.2 Subject Area Coverage Percentages

Subject area coverage percentages measure how much of the warehouse (from a data topology perspective) is being accessed by each department. They also highlight which data is shared across departments, and which is not, but could be.

Mapping operational source(s) to targets is another natural extension, which enforces and validates the lineage and Metadata already collected, and can provide penetration analysis for which source systems are in analytical use by which departments. This can help focus tuning efforts on those high impact analytic queries by mitigating any changes to heavily used sourced objects.

6.5.3 Response and Performance Metrics

Most query tools measure response time. Retrieve response or performance metrics from tools. This data will inform metrics about the number and type of users.

Harvest load times for each data product in raw format from the population processes. These should also be expressed as a percentage of expected support: so a mart that is expected to be refreshed daily and loaded in a four-hour window is 100% supported when it loads in four hours. Apply this process to any extracts generated for downstream processing too.

Most tools will retain, in a log or repository, query records, data refresh, and data extract times for the objects provided to the users. Divide this data into scheduled and executed objects, and express as raw counts both attempted and succeeded. Highly popular objects or queries performing poorly are likely in-need of attention

before satisfaction metrics suffer. This can guide defect analysis, maintenance planning, as well as capacity planning if a group of objects is failing regularly. Remediation may vary depending on the tool, but sometimes creating or dropping one index can result in great improvements. (See Chapter 6.)

A natural follow on for this is the validation and adjustment of service levels. Adjust items that have consistently failed in the next release, or in the absence of necessary funding, the support level must be reduced.

7. Works Cited / Recommended

- Adamson, Christopher. *Mastering Data Warehouse Aggregates: Solutions for Star Schema Performance*. John Wiley and Sons, 2006. Print.
- Adelman, Sid and Larissa T. Moss. *Data Warehouse Project Management*. Addison-Wesley Professional, 2000. Print.
- Adelman, Sid, Larissa Moss and Majid Abai. *Data Strategy*. Addison-Wesley Professional, 2005. Print.
- Adelman, Sid, et al. *Impossible Data Warehouse Situations: Solutions from the Experts*. Addison-Wesley, 2002. Print.
- Aggarwal, Charu. *Data Mining: The Textbook*. Springer, 2015. Print.
- Biere, Mike. *Business Intelligence for the Enterprise*. IBM Press, 2003. Print.
- Biere, Mike. *The New Era of Enterprise Business Intelligence: Using Analytics to Achieve a Global Competitive Advantage*. IBM Press, 2010. Print. IBM Press.
- Brown, Meta S. *Data Mining for Dummies*. For Dummies, 2014. Print. For Dummies.
- Chorianopoulos, Antonios. *Effective CRM using Predictive Analytics*. Wiley, 2016. Print.
- Delmater, Rhonda and Monte Hancock Jr. *Data Mining Explained; A Manager's Guide to Customer-Centric Business Intelligence*. Digital Press, 2001. Print.
- Dyche, Jill. *E-Data: Turning Data Into Information With Data Warehousing*. Addison- Wesley, 2000. Print.
- Eckerson, Wayne W. *Performance Dashboards: Measuring, Monitoring, and Managing Your Business*. Wiley, 2005. Print.
- Han, Jiawei, Micheline Kamber and Jian Pei. *Data Mining: Concepts and Techniques*. 3rd ed. Morgan Kaufmann, 2011. Print. The Morgan Kaufmann Ser in Data Management Systems.
- Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2nd ed. Springer, 2011. Print. Springer Series in Statistics.
- Hill, Thomas, and Paul Lewicki. *Statistics: Methods and Applications*. Statsoft, Inc., 2005. Print.
- Howson, Cindi. *Successful Business Intelligence: Unlock the Value of BI and Big Data*. 2nd ed. McGraw-Hill Osborne Media, 2013. Print.
- Imhoff, Claudia, Lisa Loftis, and Jonathan G. Geiger. *Building the Customer-Centric Enterprise: Data Warehousing Techniques for Supporting Customer Relationship Management*. John Wiley and Sons, 2001. Print.
- Imhoff, Claudia, Nicholas Gallemmo, and Jonathan G. Geiger. *Mastering Data Warehouse Design: Relational and Dimensional Techniques*. John Wiley and Sons, 2003. Print.
- Inmon, W. H., Claudia Imhoff, and Ryan Sousa. *The Corporate Information Factory*. 2nd ed. John Wiley and Sons, 2000. Print.
- Inmon, W.H., and Krish Krishnan. *Building the Unstructured Data Warehouse*. Technics Publications, LLC., 2011. Print.
- Josey, Andrew. *TOGAF Version 9.1 Enterprise Edition: An Introduction*. The Open Group, 2011. Kindle. Open Group White Paper.
- Kaplan, Robert S and David P. Norton. *The Balanced Scorecard: Translating Strategy into Action*. Harvard Business Review Press, 1996. Kindle.
- Kimball, Ralph, and Margy Ross. *The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling*. 3d ed. Wiley, 2013. Print.
- Kimball, Ralph, et al. *The Data Warehouse Lifecycle Toolkit*. 2nd ed. Wiley, 2008. Print.
- Kimball, Ralph. *The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleaning, Conforming, and Delivering Data*. Amazon Digital Services, Inc., 2007. Kindle.
- Linoff, Gordon S. and Michael J. A. Berry. *Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management*. 3rd ed. Wiley, 2011. Print.
- Linstedt, Dan. *The Official Data Vault Standards Document (Version 1.0) (Data Warehouse Architecture)*. Amazon Digital Services, Inc., 2012. Kindle.
- Loukides, Mike. *What Is Data Science?* O'Reilly Media, 2012. Kindle.

- Lublinsky, Boris, Kevin T. Smith, and Alexey Yakubovich. *Professional Hadoop Solutions*. Wrox, 2013. Print.
- Malik, Shadan. *Enterprise Dashboards: Design and Best Practices for IT*. Wiley, 2005. Print.
- Morris, Henry. "Analytic Applications and Business Performance Management." *DM Review Magazine*, March, 1999. <http://bit.ly/2rRrP4x>.
- Moss, Larissa T., and Shaku Atre. *Business Intelligence Roadmap: The Complete Project Lifecycle for Decision-Support Applications*. Addison-Wesley Professional, 2003. Print.
- Ponniah, Paulraj. *Data Warehousing Fundamentals: A Comprehensive Guide for IT Professionals*. Wiley-Interscience, 2001. Print.
- Provost, Foster and Tom Fawcett. *Data Science for Business: What you need to know about data mining and data-analytic thinking*. O'Reilly Media, 2013. Print.
- Reeves, Laura L. *A Manager's Guide to Data Warehousing*. Wiley, 2009. Print.
- Russell, Matthew A. *Mining the Social Web: Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More*. 2nd ed. O'Reilly Media, 2013. Print.
- Silverston, Len, and Paul Agnew. *The Data Model Resource Book Volume 3: Universal Patterns for Data Modeling*. Wiley, 2008. Print.
- Simon, Alan. *Modern Enterprise Business Intelligence and Data Management: A Roadmap for IT Directors, Managers, and Architects*. Morgan Kaufmann, 2014. Print.
- Thomsen, Erik. *OLAP Solutions: Building Multidimensional Information Systems*. 2nd ed. Wiley, 2002. Print.
- Vitt, Elizabeth, Michael Luckevich and Stacia Misner. *Business Intelligence*. Microsoft Press, 2008. Print. Developer Reference.
- WAGmob. *Big Data and Hadoop*. WAGmob, 2013. Kindle.
- Wremble, Robert and Christian Koncilia. *Data Warehouses and Olap: Concepts, Architectures and Solutions*. IGI Global, 2006. Print.