TDWI Data Virtualization

Solving Complex Data Integration Challenges

Previews of TDWI course books offer an opportunity to see the quality of our material and help you to select the courses that best fit your needs. The previews cannot be printed.

TDWI strives to provide course books that are content-rich and that serve as useful reference documents after a class has ended.

This preview shows selected pages that are representative of the entire course book; pages are not consecutive. The page numbers shown at the bottom of each page indicate their actual position in the course book. All table-of-contents pages are included to illustrate all of the topics covered by the course.

COURSE OBJECTIVES

You Will Learn:

- ***How models are used to define and frame analytic needs***
- ***Data virtualization definitions and terminology***
- ***Business case and technical rationale for data virtualization***
- ***Key concepts and foundational principles of virtualization—views, services, etc.***
- ***Data virtualization life cycle, capabilities, and processes***
- ***How to extend the data warehouse with virtualization***
- ***How virtualization enables federation and enterprise data integration***
- ***How virtualization is applied to big data and cloud data challenges***
- ***How companies use virtualization to solve business problems and drive business agility***

TDWI takes pride in the educational soundness and technical accuracy of all our courses. Please send us your comments—we would like to hear from you. Address your feedback to:

info@tdwi.org

Publication Date: November 2017

TABLE OF CONTENTS

Module 1	*Data Virtualization Concepts and Principles ….*	*1-1*
Module 2	*Data Integration Architecture ……...*	*2-1*
Module 3	*Data Virtualization in Integration Architecture ..*	*3-1*
Module 4	*Data Virtualization Platforms ……..... *	*4-1*
Module 5	*Implementing Data Virtualization ……..... *	*5-1*
Module 6	*Getting Started with Data Virtualization ……...*	*6-1*
Appendix	*Bibliography and References ……*	*A-1*

Module 1

Data Virtualization Concepts and Principles

Topic	Page
Data Virtualization Basics	1-2
Why Data Virtualization?	1-14
The Data Virtualization Foundation	1-20
Virtualize or Materialize?	1-28

Data Virtualization Basics

Data Virtualization Defined

Virtual: not physically existing as such but made by software to appear to do so

oxford dictionaries online
www.oxforddictionaries.com

Data Virtualization: the presentation of data as an abstract layer, independent of underlying database systems, structures and storage

wikipedia

Data virtualization is the process of offering data consumers a data access interface that hides the technical aspects of stored data, such as location, storage structure, API, access language, and storage technology.

Rick van der Lans
www.b-eye-network.com/view/14815.

Data virtualization is a data integration technique that provides complete, high-quality and actionable information through virtual integration of data across multiple, disparate internal and external data sources.

Judith Davis & Robert Eve
Data Virtualization: Going Beyond Traditional Data Integration to Achieve Business Agility

Data Virtualization Basics

Data Virtualization Defined

WHAT IT MEANS TO BE VIRTUAL

The Oxford Dictionary defines *virtual* as "not physically existing as such but made by software to appear to do so." Virtual data, then, is a data structure that appears to exist but does not exist as a physically stored set of data. Data virtualization (DV) includes the processes and technologies that are used to create virtual data.

Wikipedia describes *data virtualization* as "the presentation of data as an abstract layer, independent of underlying database systems, structures, and storage." This definition captures two key elements of DV:

- abstraction
- decoupling (removal of dependencies)

FROM THE EXPERTS

The facing page shows two definitions from recognized experts in the subject of data virtualization. Key concepts in Rick van der Lans's definition include:

- virtualization as a process
- data consumers
- hidden technology

Judith Davis and Robert Eve define virtualization from a purposeful perspective, with the purpose encompassing:

- integration of disparate data
- reach across internal and external data sources
- complete information
- high-quality information
- actionable information

Virtualize or Materialize?

Decision Factors

Virtualize or Materialize?

Decision Factors

COMPLEX DECISIONS

Deciding to integrate data materially, virtually, or as a hybrid is a complex process that involves many variables:

- **Time to Solution—**the speed at which a data integration solution is needed. Greater urgency indicates virtualization.
- **Cost Sensitivity—**extremely limited budget indicates virtualization.
- **Requirements Stability—**concerned with clarity of data integration requirements. Clear and stable requirements are suited to materialization uncertain and volatile requirements fit virtualization.
- **Replication Constraints—**considers privacy and policy limits to creating multiple copies of data. Use virtualization when constraints are strong.
- **Organizational Personality—**describes a cultural continuum that ranges from cautious and risk-averse to adventurous. Highly cautious organizations are better suited to tried-and-true methods such as ETL.
- **Source System Availability—**essential to virtualization. Limited availability makes on-demand integration difficult to achieve.
- **Source System Load—**considers the processing capacity of source systems to take on additional query demand. For source systems with little headroom, demands of virtualization may exceed capacity.
- **Data Cleansing Needs—**messy data that requires complex data cleansing is a poor fit for virtualization.
- **Transformation Complexity—**considers the structures, dependencies, and quantities of business and data rules that must be applied to integrate data. Highly complex transformations are better suited to materialization than to virtualization.
- **Application Focus—**ranges from operational and real-time decision support to time-series analysis and data mining. The real consideration here is the amount of history that is needed in integrated data. When history needs exceed that which is available in source systems at any point in time, then materialization is necessary.
- **Data Format—**multidimensional and other nonSQL target data structures are better suited to materialization than to virtualization.
- **Target Data Freshness—**real-time and very low latency data are virtualization friendly and difficult to achieve with ETL and materialization.
- **Data Volume Per Query—**processing large amounts of data with each query is not ideal for data virtualization.

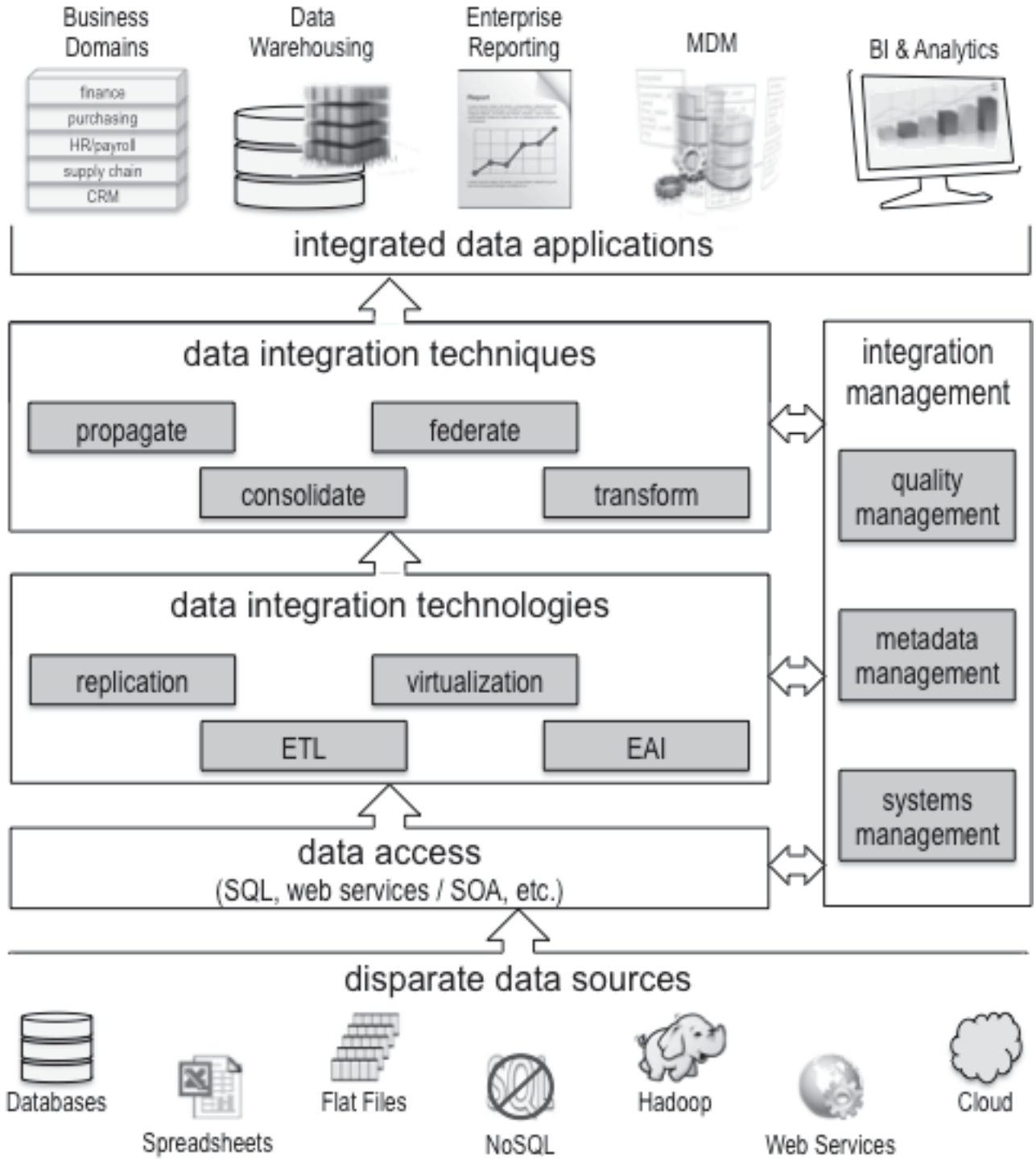
Module 2

Data Integration Architecture

Topic	Page
Integration Architecture Concepts	2-2
Data Virtualization Architecture Examples	2-8
Virtualize or Materialize?	2-16

Integration Architecture Concepts

Integration Architecture Defined



Integration Architecture Concepts

Integration Architecture Defined

ARCHITECTURE

Architecture defines the roles, structure, relationships, and rules by which a collection of components constitute a cohesive whole—the glue that bonds individual parts into a system. Architecture is an early-stage design activity that precedes detailed design, specification, and construction. Effective architecture ensures that the things we build:

- Are suited to the purposes for which they are intended
- Comply with regulations and standards
- Fit gracefully into their environment
- Are sustainable through their expected lifespan
- Are aesthetically pleasing

These principles hold true for architecture of many things—buildings, bridges, information systems, and more.

DATA INTEGRATION ARCHITECTURE

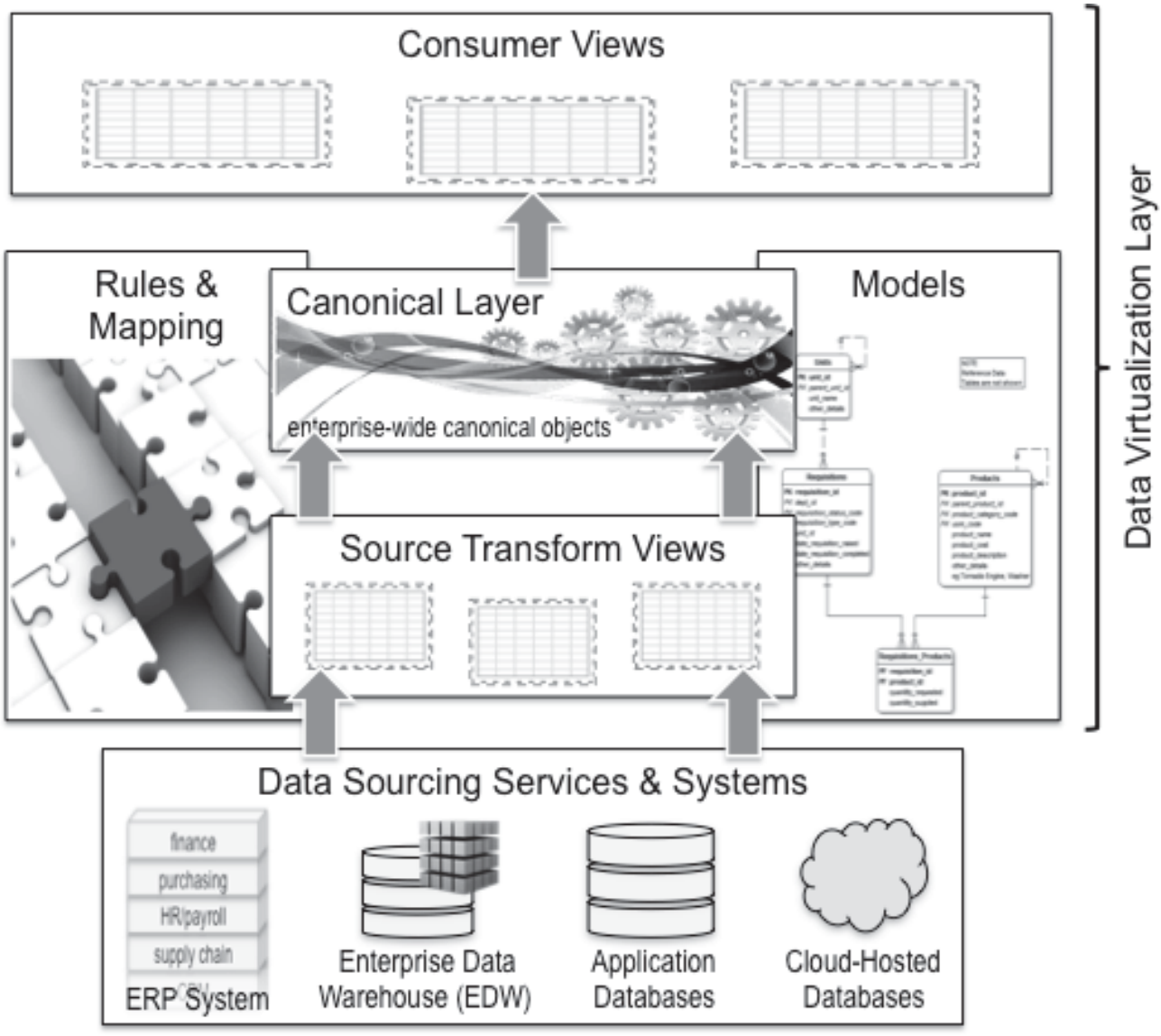
Data integration architecture defines the roles, structure, relationships, and rules to aggregate a collection of data integration components into a data integration system.

The facing page illustrates generic data integration architecture comprising these components:

- **Disparate data sources**—The non-integrated data that is the target of data integration activity. The scope of data types ranges from highly structured relational data to unstructured, web, cloud, and "big data" sources.
- **Data access methods**—The means by which integration technologies connect to data sources. These methods encompass all of the common data access protocols.
- **Data integration technologies**—The classes of tools that are available to automate and execute data integration tasks: data replication, data virtualization, extract-transform-load (ETL), and enterprise application integration (EAI).
- **Data integration techniques**—The methods, processes, and products that are used to combine, connect, and rationalize disparate data as a unified data resource: propagation, transformation, consolidation, and federation.
- **Integrated data applications**—The business and information systems that access and use integrated data
- **Integration management**—The essential components to for integration system internals: quality, metadata, and systems management.

Data Virtualization Architecture Examples

Ministry Social Services Logical Architecture



Data Virtualization Architecture Examples

Ministry Social Services Logical Architecture

THREE-LAYER ARCHITECTURE

The example on the facing page is drawn from a case study of Compassion International described in the book *Data Virtualization.*[1]

The data virtualization system is designed to integrate data from multiple, complex sources including ERP, EDW, application databases, and cloud-hosted databases. Progression from source views of data to consumer views depends on

- multilayer architecture,
- models to describe the data in source and in business contexts,
- mapping and rules to drive data transformations.

The data virtualization layer encompasses three sub-layers: source transform views, canonical objects, and consumer views—each considered to be a collection of "building blocks." The characteristics of the building blocks as described by Davis and Eve are:

- They are actual views where query is possible—not just logical objects.
- They look more like source system views at the bottom of the diagram and become increasingly business-oriented as you move upward.
- They encapsulate standard and reusable business logic.
- They can be cached for performance optimization.
- Each is documented in a wiki form accessible to end-users and to developers.

[1] *Data Virtualization*, pp. 81-91, Davis and Eve, 2011 (www.datavirtualizationbook.com)

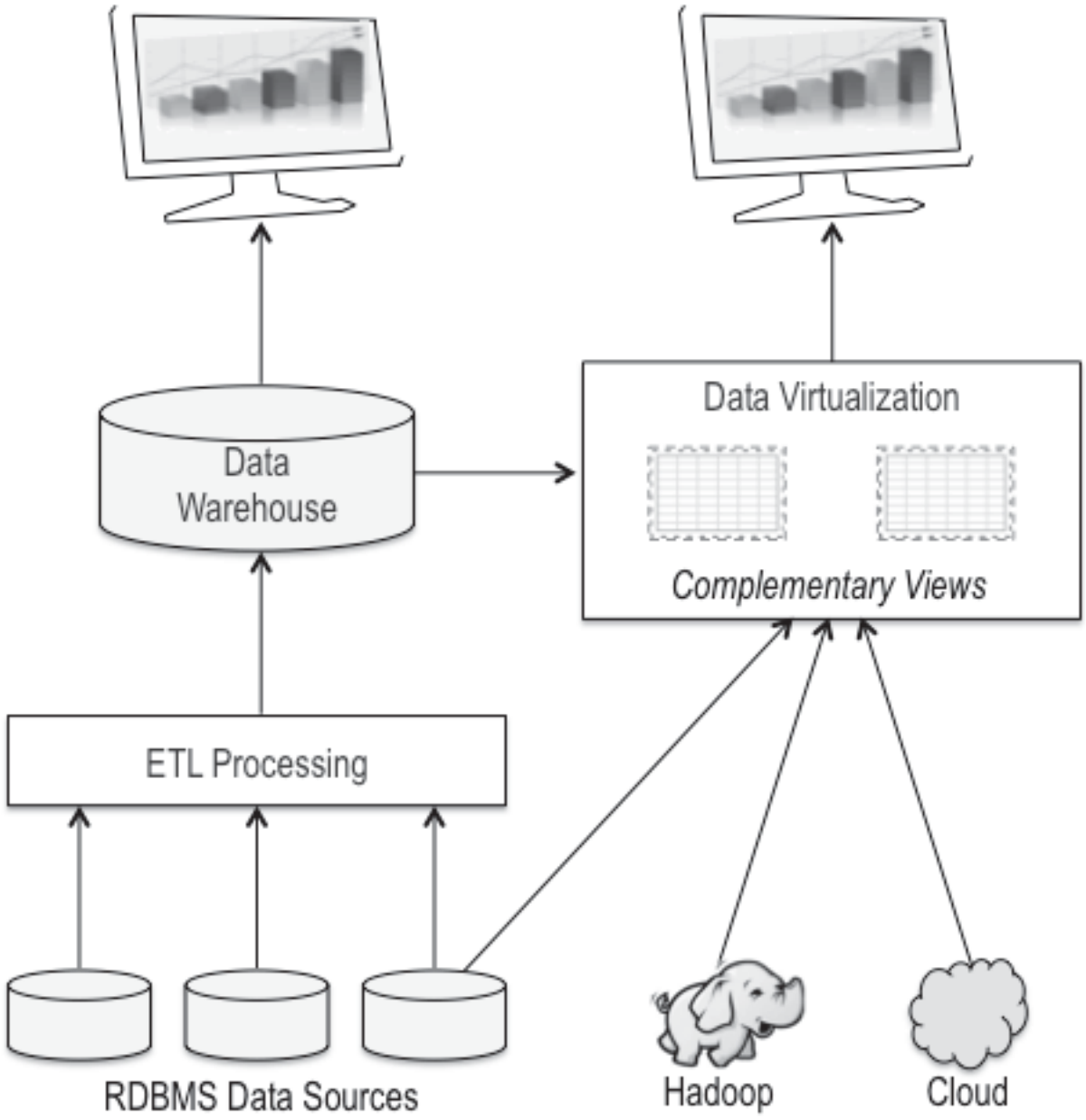
Module 3

Data Virtualization in Integration Architecture

Topic	Page
Virtualization in Data Integration Projects	3-2
Data Warehousing Use Cases	3-4
Data Federation Use Cases	3-16
MDM and EIM Use Cases	3-22
More Data Virtualization Applications	3-30
Virtualize or Materialize?	3-34

Data Warehousing Use Cases

Data Warehouse Augmentation



Data Warehousing Use Cases

Data Warehouse Augmentation

EXTENDING THE EXISTING DATA WAREHOUSE

Traditional data warehousing systems are designed to provide integration of structured data through extract, transform, and load (ETL) processes. The output of ETL processing is integrated data that is physically stored in a relational database and made available for downstream reporting and access. Depending on the data architecture, additional data stores such as data marts may exist to optimize the information delivery functions. The batch nature of ETL processing necessitates some latency of warehouse data.

CHALLENGES

Long-term success and sustainability of a data warehouse is based on ability to adapt and evolve to the meet continuously changing information needs. The time and effort required to bring in additional source data is a significant challenge for existing data warehouses. The challenge and the complexities increase when the new requirements include unstructured data. Real-time data requirements bring additional challenges in ETL-based data warehousing processes.

Abundance of unstructured data and the impact of big data technologies bring both opportunities and challenges. The emergence of big data in a modern business context—especially social media data—creates opportunity to analyze and better understand customer perceptions and behaviors. The opportunity comes with complexity, though—unstructured social data is not a quick and easy fit into a traditional data warehouse.

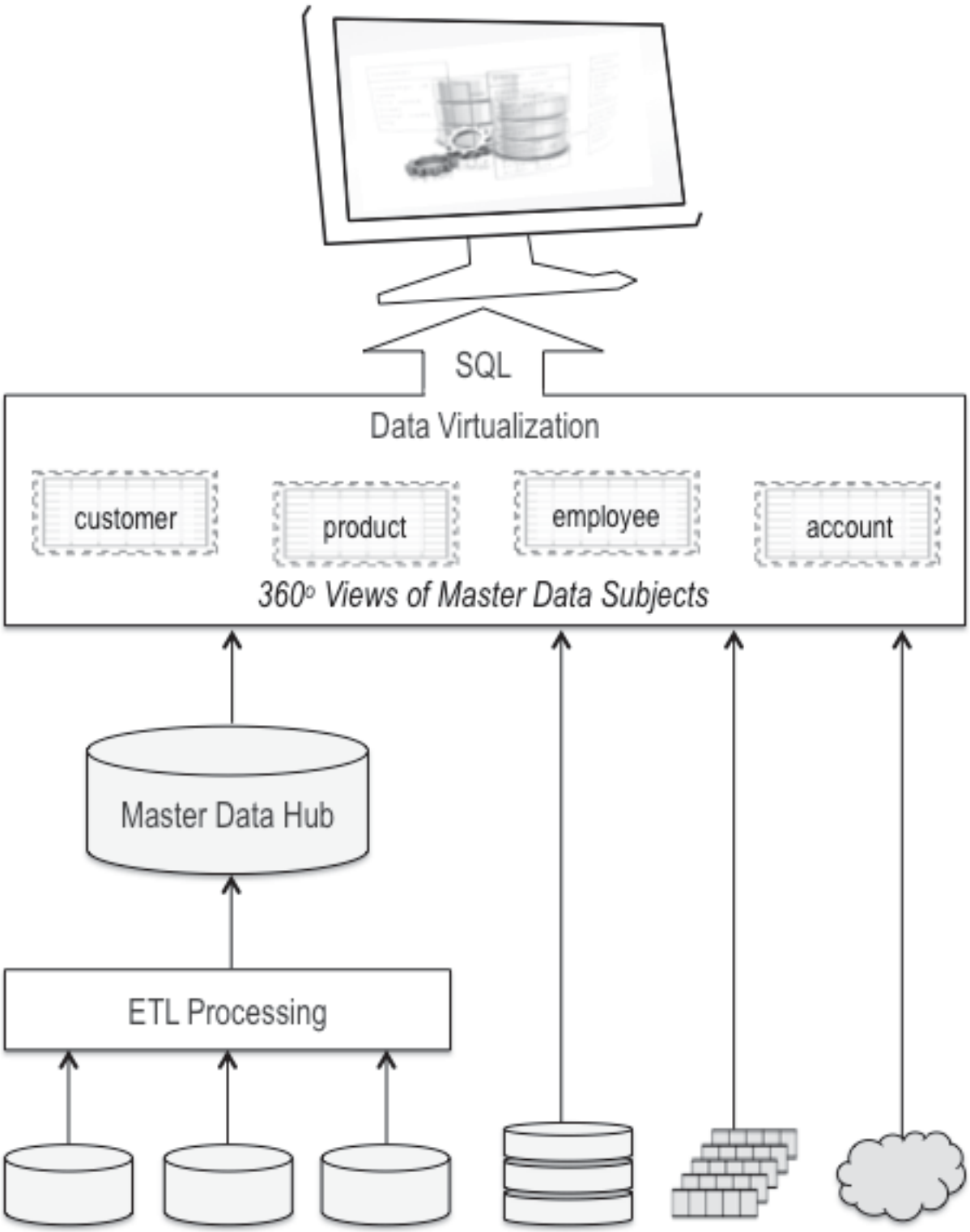
OPPORTUNITY ENABLED BY VIRTUALIZATION

Data Virtualization can be applied to complement and augment an existing data warehouse with virtual views to meet new information requirements. Unstructured data, cloud data, and real-time data integration can be implemented without extensive and disruptive changes to the core data model and ETL processing.

Speed of delivery and speed of data are accelerated with virtualization. Leveraging new and existing data sources more rapidly advances business agility. Unstructured data is integrated with structured data and new reporting applications are quickly implemented.

MDM and EIM Use Cases

Master Data Hub Extension



MDM and EIM Use Cases

Master Data Hub Extension

A 360° VIEW OF MASTER ENTITIES

Master data is the reference data that is shared across many business functions—data about customers, products, employees, accounts, etc. The Master Data Management (MDM) vision is often described as providing a 360° view of these entities—a consistent and complete view from all perspectives. The 360° view includes past, present, and future information about identity, relationships, activity, value, and expectations.

MASTER DATA HUB

A common architecture for MDM is integration of master data into a shared database called a hub. This is similar to the hub of hub-and-spoke data warehouse architecture—a single point of integration for the data in scope.

CHALLENGES

Choosing which data elements to include in a hub is always difficult. Too much data makes synchronization and consolidation exceptionally difficult. Too little data has very limited impact.

A master data hub typically contains current identity and shared descriptive information about master data entities. The core function of MDM is, in fact, identity management. The hub may also contain some relationship information—again limited to the current state. A customer, for example, may be represented in the hub with customer number, customer name, mailing address, email address, and relationships with customer loyalty programs.

Compare this example with the description of 360° view above. The hub contains current identity and some relationships. It lacks past and future information and it is missing data about activity, value, and expectations. It is impractical, however, to include transaction detail (activity and value), transaction history (past), lifetime value calculations (value and future), retention forecasts (future) and other details in the hub. The vision of a 360° view is not possible with an MDM hub alone.

FEDERATING MASTER DATA

Virtualization makes the 360° view possible by federation of master hub data and detailed data from various data source. The hub serves consolidation and integration needs for current identity and relationship data. Virtualized views extend the hub to complete the view with transaction detail, past and future perspectives, etc.

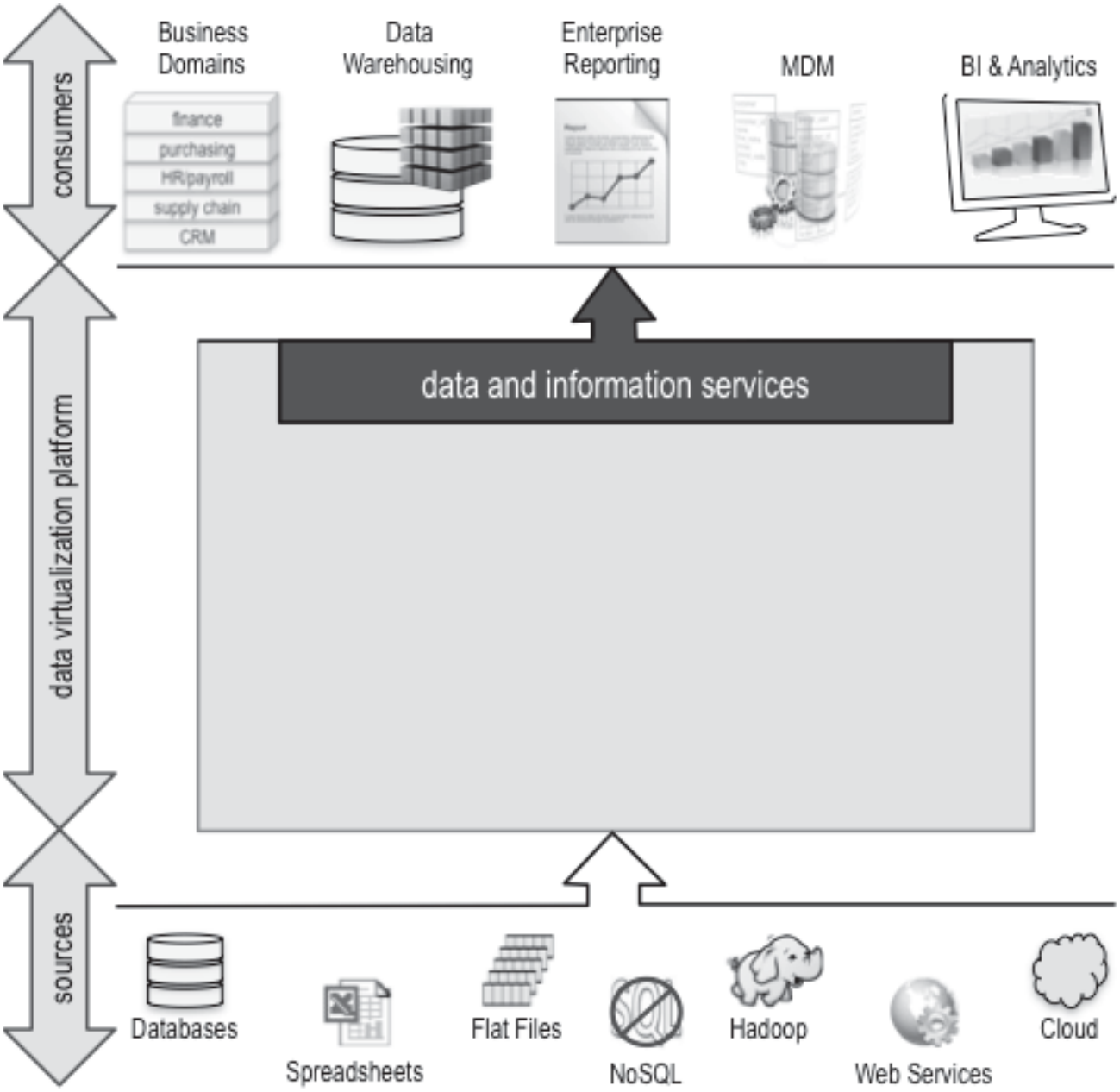
Module 4

Data Virtualization Platforms

Topic	Page
Platform Requirements	4-2
Platform Capabilities	4-8
Platform Variations	4-28

Platform Requirements

Data and Information Services



Platform Requirements

Data and Information Services

VIRTUALIZATION PLATFORMS

A data virtualization platform encompasses all of the tools, technology, and practices that are needed to connect data consumers with disparate data sources using methods where:

- Data is accessed in business context and using business language.
- Data is integrated and source disparity minimized or eliminated.
- Data of different types—structured, semistructured, multistructured, and unstructured—can be combined in a single view.
- The data does not need to be replicated or redundantly stored.

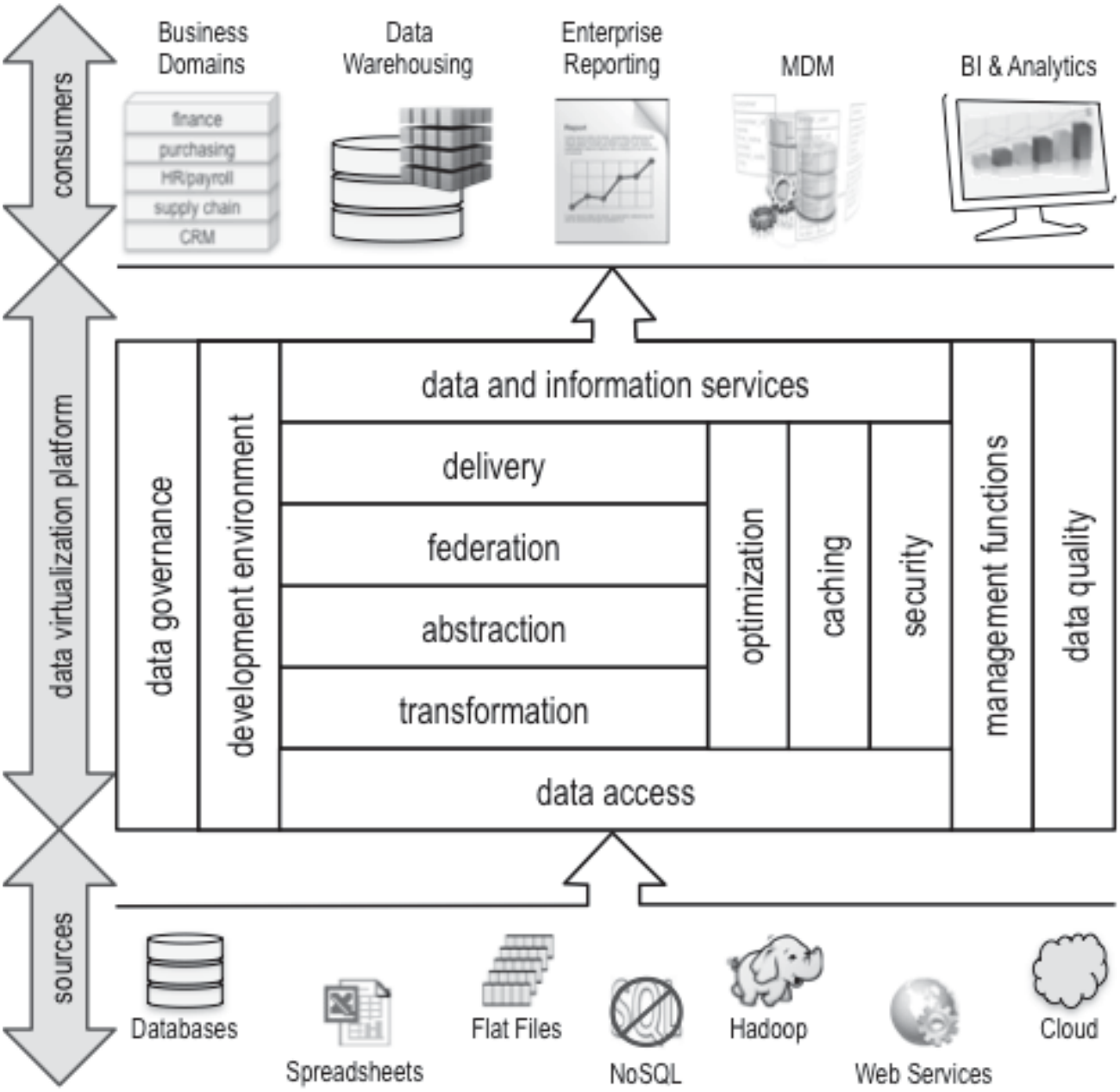
A data virtualization platform can be described as a collection of requirements and capabilities. A data virtualization platform must provide capabilities for delivery, development, and management of data and information services.

DELIVERY: THE TOP-LEVEL REQUIREMENT

The first and most fundamental requirement of a data virtualization platform is to provide data and information services. These services provide the data access methods that are essential for data consumers to gain access to data. Typical data access methods include SQL access to relational views and web services for both structured and unstructured data.

Platform Variations

Stand-Alone Data Virtualization



Platform Variations

Stand-Alone Data Virtualization

VIRTUALIZATION ONLY

A stand-alone platform meets the requirements and provides the capabilities needed to implement and operate data virtualization. The standalone platform is compatible with, but operates independently of data warehousing and business analytics platforms.

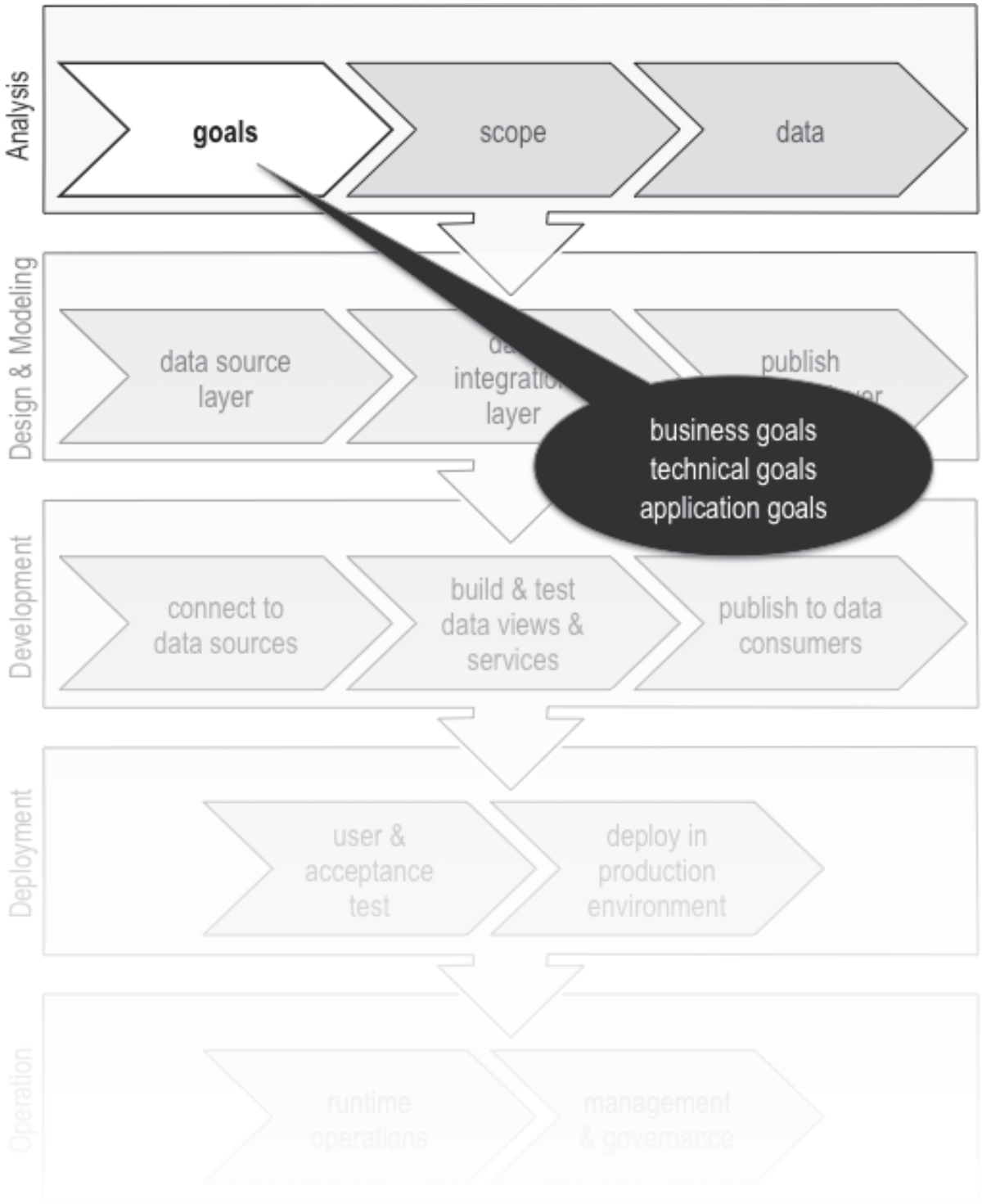
Module 5

Implementing Data Virtualization

Topic	Page
Analysis	5-2
Design and Modeling	5-10
Development	5-16
Deployment	5-24
Operation	5-26
Virtualize or Materialize?	5-30

Analysis

Goals and Purpose



Analysis

Goals and Purpose

OVERVIEW

Implementing data virtualization is a process that progresses through activities of analysis, design and modeling, development, deployment, and operation. Execute this sequence within a plan of phased adoption—i.e., incremental implementation and repetition of steps in the sequence for managed growth of goals, scope, and maturity of data virtualization.

Begin analysis with goal setting. It is important to know *why* data virtualization before defining *what* and *how.*

BUSINESS GOALS

Business goals for data virtualization certainly include business agility, but more specific goals provide the foundation for scoping, data analysis, design, and modeling. Think about the business processes where you want to have impact, and the kind of impact for each process. Consider impacts such as decision speed, completeness, and quality of decision-making information, opportunity recognition and realization, reduction of uncertainty, mitigation of risk, process effectiveness, and process efficiency.

TECHNICAL GOALS

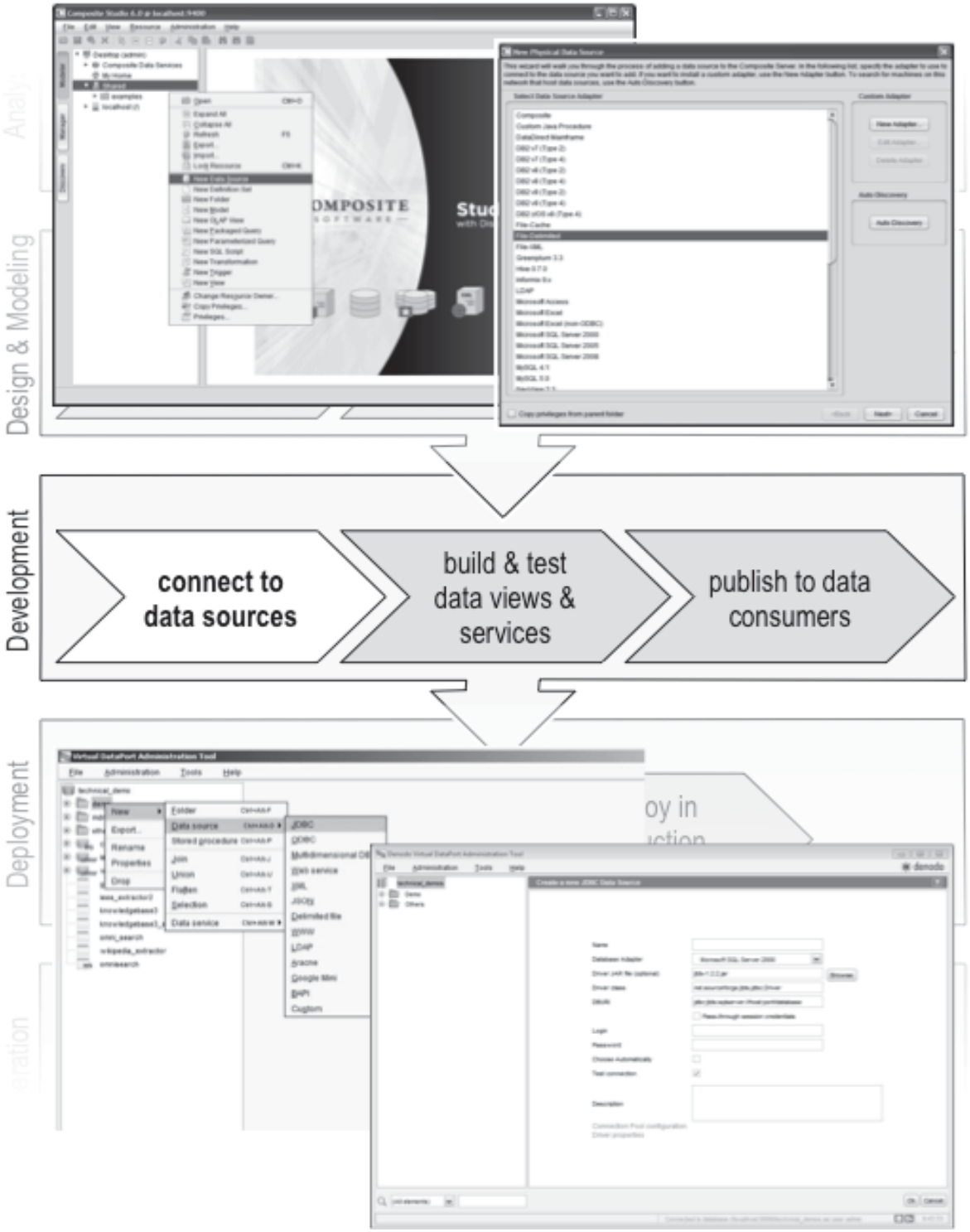
Technical goals for data virtualization usually focus on increasing speed of information, integrating unstructured data with structured data, reducing data latency, accelerating development cycles, incorporating big data into the information resource. Consider each and be specific about the goals: Speed of what information, integration of which unstructured data, latency of which data, and so on.

APPLICATION GOALS

Application goals look at the purpose of data virtualization from a data consumer perspective. Which systems—data warehousing, ERP, MDM, enterprise reporting, business analytics, etc.—must data virtualization serve to achieve the stated business and technical goals? Which technical goals apply to each system? How will meeting technical goals for an application help to meet the business goals?

Development

Connect to Data Sources



Development

Connect to Data Sources

FROM DESIGN TO DEVELOPMENT

With analysis and design complete (or in the case of prototyping or agile development, complete enough) proceed to development activities—the work of creating data virtualization functions. Development encompasses three main categories of activity that correspond with the three layers of architecture—connecting to data sources, building views and services, and publishing to consumers.

REACHING DATA SOURCES

Begin development by connecting to data sources. The specific steps of data connection vary depending on data virtualization platform and tools. The development environment of each platform will include screens and processes to build data connections. The screenshot examples shown here illustrate data connection with two widely used data virtualization platforms: TIBCO Data Virtualization Platform, which primarily connects using drivers, and Denodo Platform whose primary connection method is wrappers.

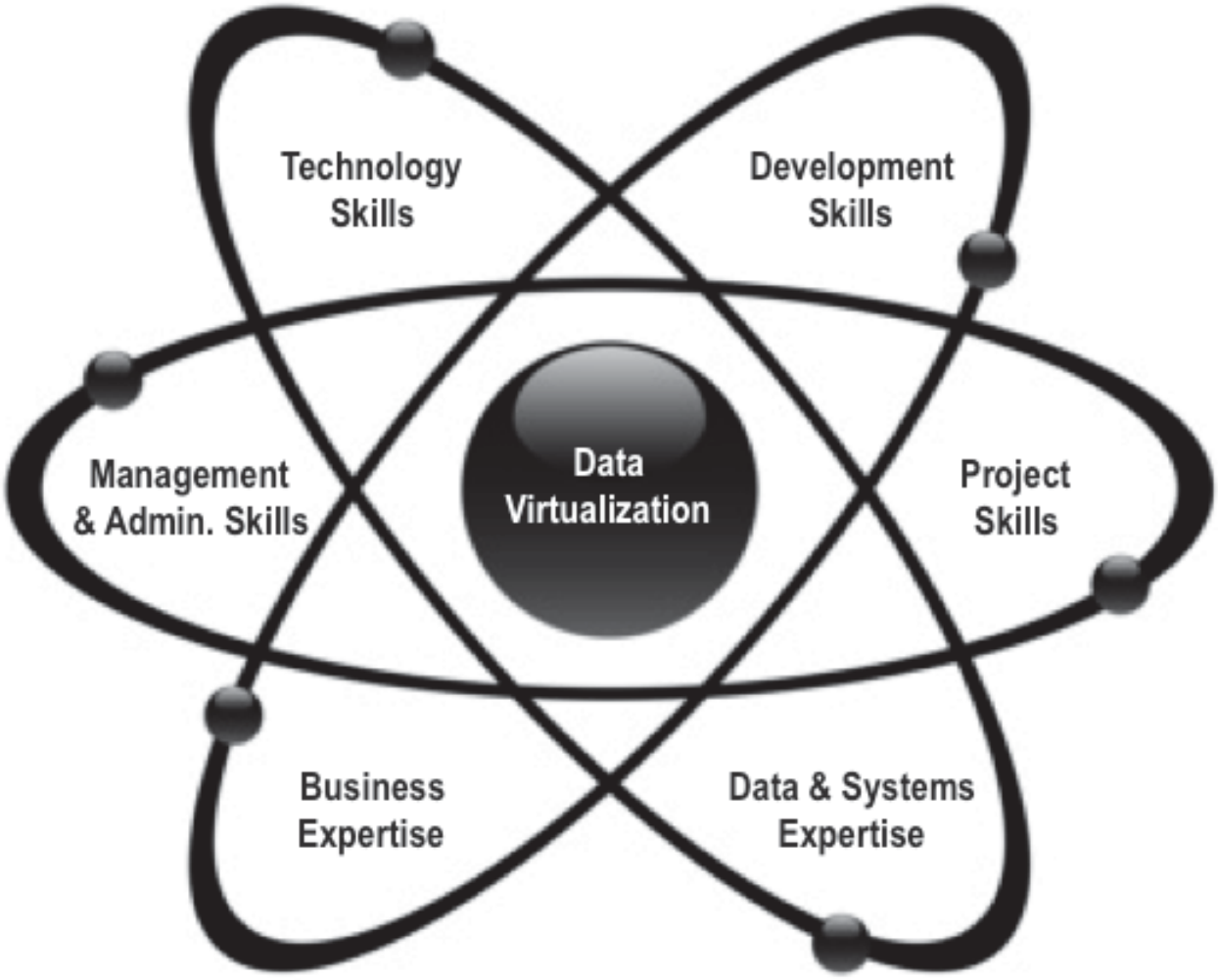
Module 6

Getting Started with Data Virtualization

Topic	Page
Skills and Competencies	6-2
Human Factors	6-4
Goals and Expectations	6-6
Best Practices	6-12

Skills and Competencies

Capabilities and Expertise



Skills and Competencies

Capabilities and Expertise

RANGE OF SKILLS

As you've seen throughout the course, there are many different aspects of data virtualization ranging from business to technology, from architecture to operations, and from analysis to deployment. Building and operating data virtualization systems demands a similarly broad range of skills and competencies including:

- Technical skills and data virtualization platform knowledge
- Business subject expertise
- Data and systems subject expertise
- Project skills from planning to execution
- Development skills and capabilities
- Management and administrative skills applied to data, security, projects, and technology

Goals and Expectations

Data Virtualization Readiness

GETTING STARTED If you're just getting started with data virtualization, begin with realistic and achievable expectations. List your data virtualization goals, and then assess your readiness to achieve those goals. The facing page lists 15 common kinds of goals. Achieving all of them simultaneously is a tall order even for seasoned data virtualization teams. Create early successes, then build upon them to grow your capabilities, advance your readiness, and evolve your data virtualization organization.

