Previews of TDWI course books are provided as an opportunity to see the quality of our material and help you to select the courses that best fit your needs. The previews can not 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. The pages shown are not consecutive. The page numbers as they appear in the actual course material are shown at the bottom of each page. All table-of-contents pages are included to illustrate all of the topics covered by a course.

TDWI Data Warehousing Architectures:

Choosing the Right Data Warehousing Approach

The Data Warehousing Institute takes pride in the educational soundness and technical accuracy of all of our courses. Please give us your comments – we'd like to hear from you. Address your feedback to:

email: info@tdwi.org

Publication Date: February 2005

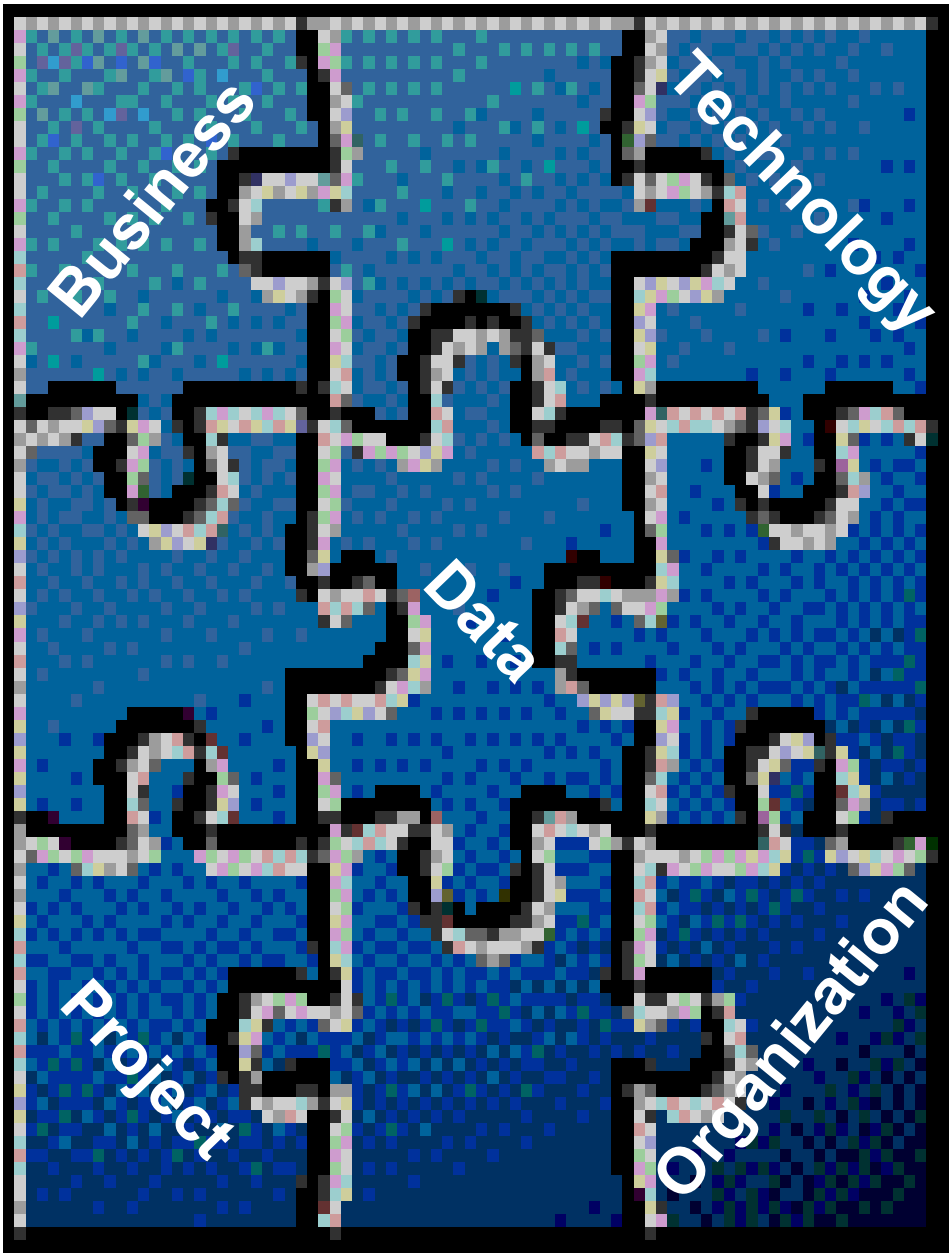
TABLE OF CONTENTS

Module 1

Architecture Concepts

Architecture Defined

What is Data Warehousing Architecture?



Architecture Defined

What is Data Warehousing Architecture?

DEFINITIONS

Architecture describes, at an abstract level, the major components of something to be constructed, the roles and functions of those components, and the relationships among them. Information systems architecture describes the components of information systems (data, function, network, etc.) and the roles and relationships of those components. Data warehousing architecture is a subset of information systems architecture specifically oriented toward systems whose purpose is to receive disparate data and deliver integrated information.

Many data warehousing architectures exist today, with differences in the components, structure, associations, and language. Architectural needs across multiple organizations are as varied as their information needs. The ideal architecture for one organization may be especially ill-suited to another. The purpose of this course is to examine several representative architectures and provide guidelines to determine which is best suited (or which parts of several may be combined) to meet your organization's unique needs.

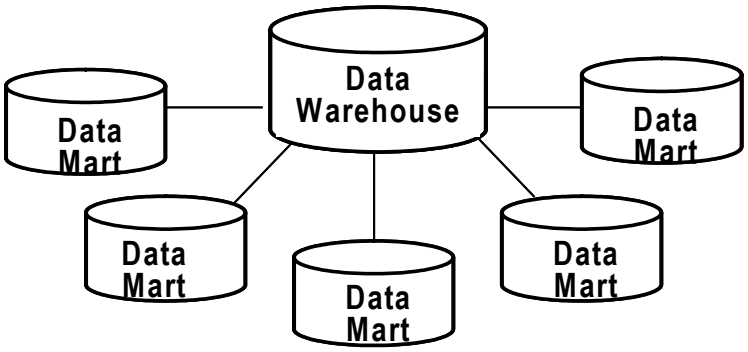
KINDS OF DATA WAREHOUSING ARCHITECTURE

Data warehousing architecture addresses business alignment of the data warehouse, data structures and flows, technology, project sequences and lifecycles, and organizational integration of the warehouse. A complete data warehousing program is broad in scope and focused on analysis and design the following areas:

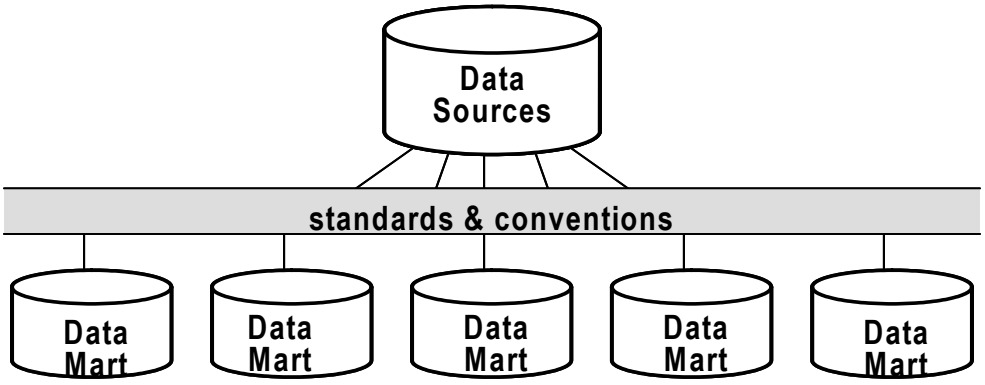
- Business Architecture – Understanding of business goals, drivers, and information needs.
- Data Architecture – Understanding of source data. Requirements and standards for warehousing data and warehouse metadata.
- Technology Architecture – Identification of standards for hardware, software, and communications technology. Specification of the data warehousing toolset.
- Project Architecture – Incremental development plan for the data warehouse. Defined scope of each increment. Sequence and dependencies among increments.
- Organizational Architecture – Identification of training, support, and communications responsibilities.

Data Architecture

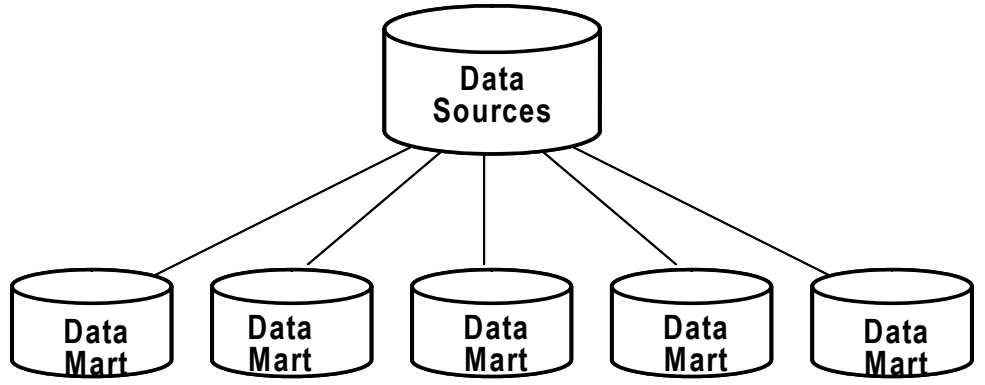
Data Integration



Dependency through hub & spoke structure



Conformity through bus structure



Independence

Data Architecture

Data Integration

INTEGRATION DEBATES

Integration is one of the fundamental concepts of data warehousing. Inmon's early definition of a data warehouse identifies "integrated" as one of the four defining characteristics of a data warehouse. The discussions surrounding integration center around how to achieve that integration, with particular focus on data marts:

- Should data marts **depend** on a single integrated source?
- Should data marts **conform** to integration standards?
- Should data marts be **independent** in process and structure?

DEPENDENCY

Dependent data marts are architected to be populated from a single integrated data source – typically a data warehouse. When designing dependent data marts, integration is the responsibility of the data warehouse. The warehouse serves as an "integration hub" in a hub-and-spoke relationship between warehouse and data marts. Common arguments for and against dependency are shown below:

PROS	CONS
Integration work is done only once. Integration is consistent across all marts. Warehouse data is reusable by many marts.	Too much up front analysis & modeling. It takes too long to design. Warehouse grows large & hard to maintain.

CONFORMITY

Conformed data marts are populated directly from original sources, and achieve the goals of integration by complying with established data definition and data structure standards. Conforming facts and dimensions are the interface standards to support mart-to-source and mart-to-mart consistency in a bus architecture. The common arguments are:

PROS	CONS
Integration is done where business needs it. Doesn't require lots of up front modeling. You can achieve it incrementally.	Too easy to get around standards. Standards don't have enterprise view. May have to re-work existing marts.

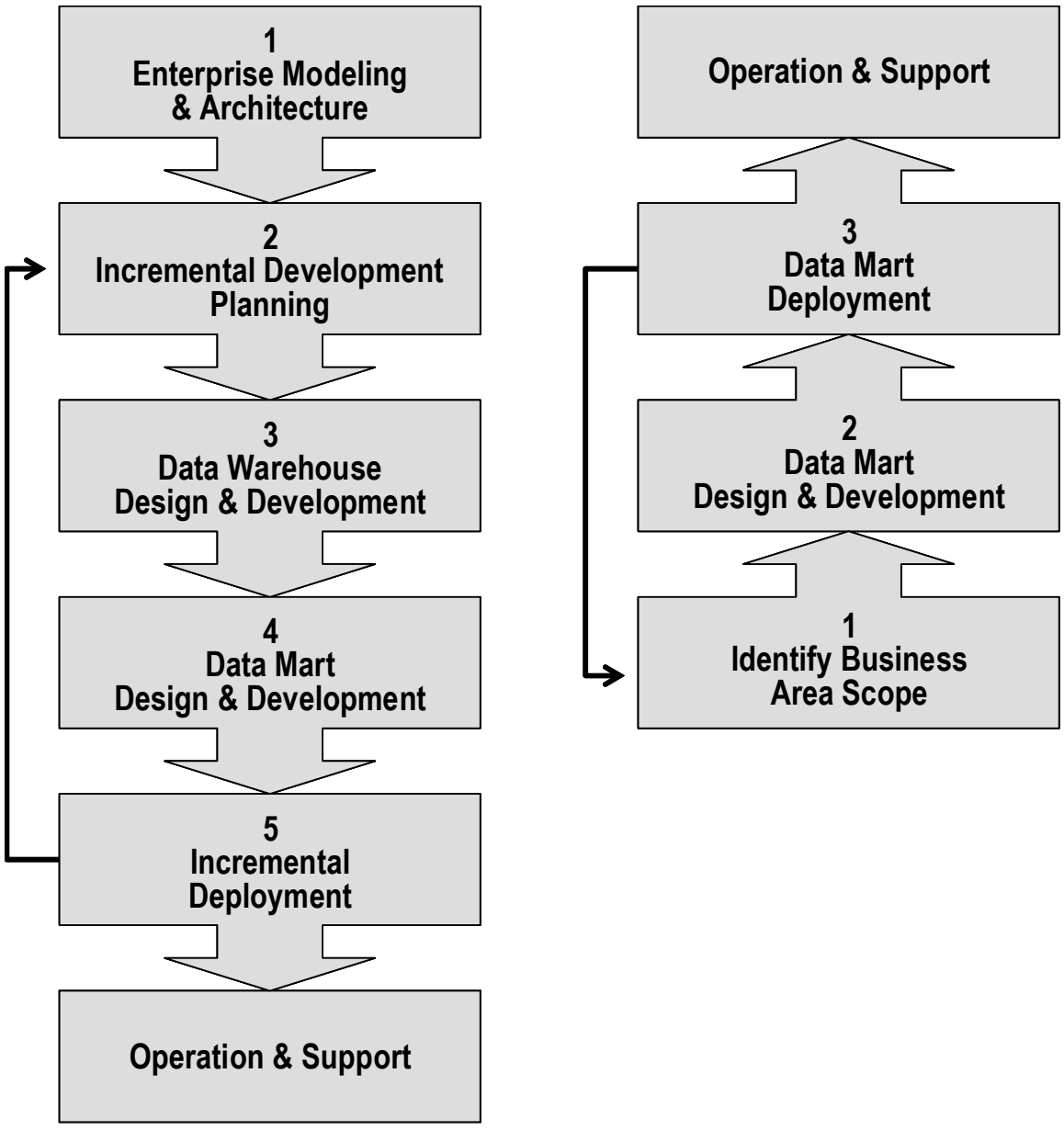
INDEPENDENCE

Independent data marts are populated directly from original sources without applying data definition and data structure standards. No mart-to-mart consistency is intended with independent marts. The common arguments are:

PROS	CONS
It's the fastest way to build a mart. It meets individual user needs.	There is no integration. It doesn't fit the definition of data warehouse.

Project Architecture

Top-Down vs. Bottom-Up



Project Architecture

Top-Down vs. Bottom-Up

TOP-DOWN AND BOTTOM-UP DEBATES

Project architecture is really methodology. It describes the process by which warehousing results will be produced. Debates about the warehouse development process are as widespread as those about data structure and data flow. While some developers argue fervently for a user-driven approach, others strongly advocate an enterprise approach. The user-driven approach, commonly called *bottom-up,* is often described as "the Kimball approach." The enterprise-driven process is commonly called *top-down,* and is thought of as "the Inmon approach." While this "Inmon vs. Kimball" debate continues, with each faction apparently convinced that they have discovered the one right way to build a data warehouse, the two approaches are not as different as popular discussion would have us believe.

TOP-DOWN APPROACHES

A top-down approach to warehouse development begins at the enterprise, and emphasizes the data warehouse as a primary integrated information resource. Data warehouse structure is determined through enterprise modeling. Content is determined by a combination of business information needs and available source data. Typically, the data warehouse is the sole source of data to populate data marts. Common arguments for and against a top-down approach are

PROS	CONS
Produces a flexible enterprise architecture. Avoids overhead of independent data marts. Compatible with ODS & data mining needs. Keeps detailed data in a relational form. Eliminates redundant data extracts.	It takes too long to deliver something. It costs too much to get started. Drill-through (mart-to-warehouse) is complex. Marts may need detailed data anyway.

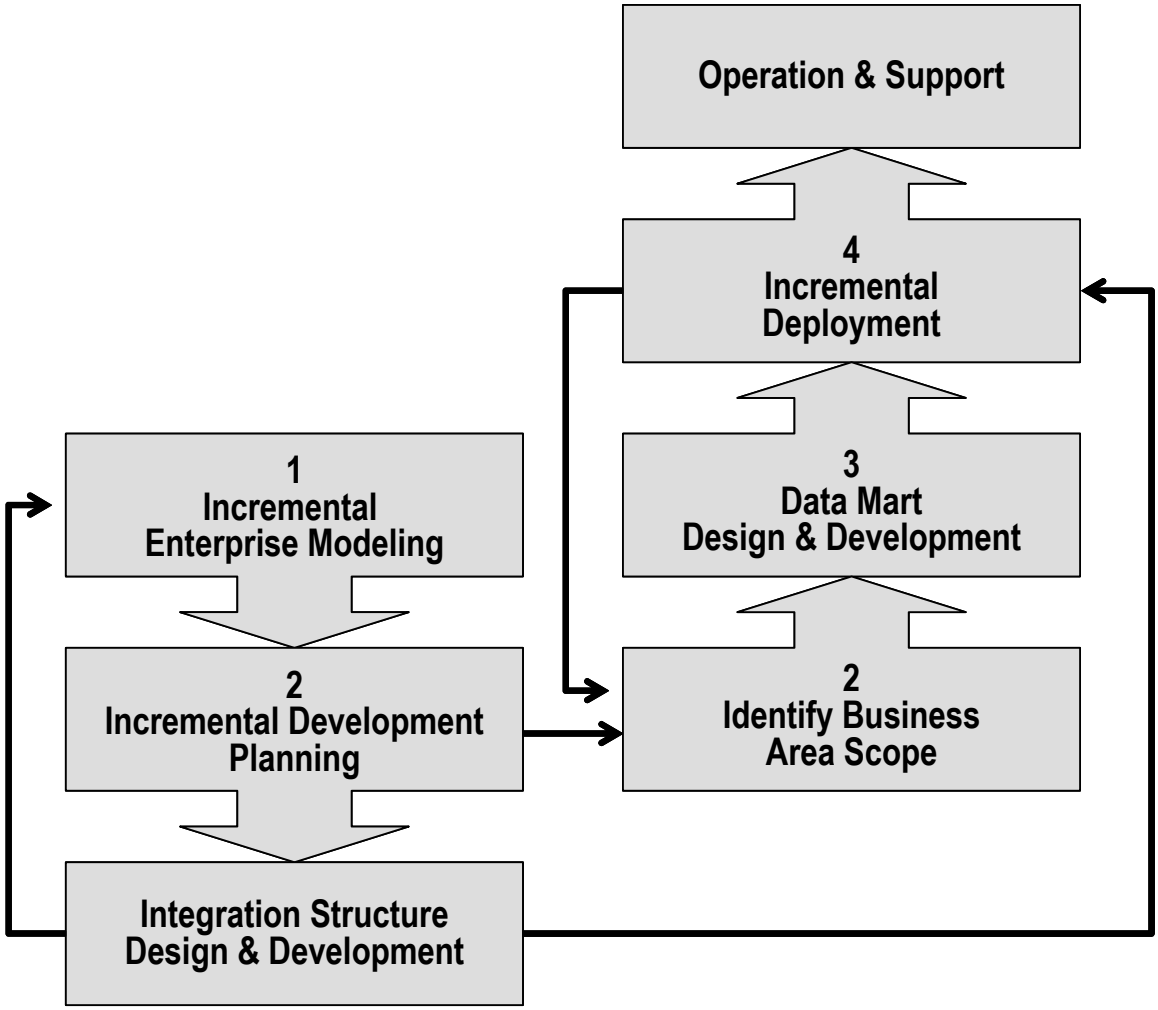
BOTTOM-UP APPROACHES

A bottom-up approach to data warehousing begins with business information needs, and emphasizes data marts as the primary resource for delivering information to support business analytics. Data marts are populated by moving source data through a transient staging area. Atomic data is kept in the data marts. Conformed dimensions are the principal integration mechanism. Common arguments for and against bottom-up approaches are

PROS	CONS
Fast, low-cost development & deployment. Minimizes "backroom" operations. Keeps atomic data in the data marts. Star schema are extensible. Staging can eliminate redundant extracts.	Difficult to join data across multiple stars. Conformity standards are easy to violate. May have redundant data transformations. May have inconsistent data transformations. Can't support ODS or operational reporting. Difficult to support data mining.

Project Architecture

Hybrid Approaches



Project Architecture

Hybrid Approaches

COMBINING TOP-DOWN & BOTTOM-UP

Hybrid approaches combine some elements of bottom-up development with some from top-down methods. The objective of a hybrid approach is rapid development within an enterprise context. A typical hybrid approach quickly develops a skeletal enterprise model before beginning iterative development of data marts. The data warehouse is populated only as data is needed by data marts, and is sometimes constructed in a retrofit mode after data marts have been deployed. Metadata consistency and conformed dimensions are the initial integration tools, with the data warehouse being a secondary means of integration. Common arguments for and against hybrid methods include:

PROS	CONS
Rapid development with enterprise context. Avoids independent data marts. Limits warehouse to data that provides value. Enforces consistency of metadata. Avoids redundant extracts through warehouse. Prevents inconsistent transformations.	Incomplete enterprise modeling. Warehouse retrofit only delays time & cost. Metadata as cornerstone is difficult to manage. May depend on specific technology.

Architecture & Methodology Selection Factors

Overview

✓	**strategic vs. tactical decision support**
✓	**integration of data**
✓	**structure of data (dimensional, relational, normalized)**
✓	**consistency (of metadata and business rules)**
✓	**scalability (growth in scope)**
✓	**persistence (retention of data)**
✓	**sustainability (maintainable and adaptable to change)**
✓	**technology requirements**
✓	**staffing and skills requirements**
✓	**time to delivery**
✓	**cost to deploy**
✓	**cost to operate**

Architecture & Methodology Selection Factors

Overview

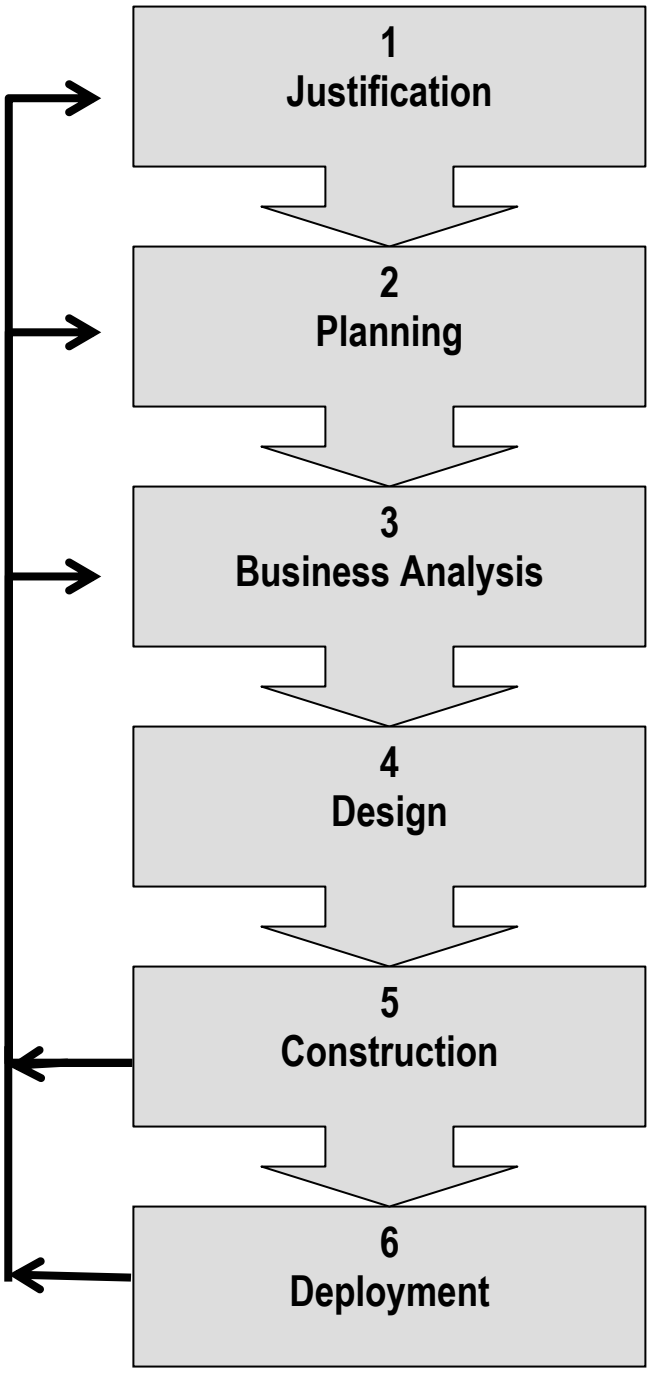
CRITERIA TO MAKE A CHOICE

No two organizations have precisely the same goals, expectations, and constraints for data warehousing. Thus, no single architecture or methodology is an ideal fit for everyone. Understanding the needs and limits of your organization is the first step to determining a best-fit architecture and methodology.

This course uses the twelve items illustrated on the facing page as assessment criteria useful in making architecture and methodology decisions. These are decision factors common to nearly every organization. Your organization may have other important criteria to include in such an assessment.

Common Architectures and Approaches

Hybrid Example - Business Intelligence Roadmap



Business Intelligence Roadmap
Moss

Common Architectures and Approaches

Hybrid Example - Business Intelligence Roadmap

BUSINESS INTELLIGENCE ROADMAP OVERVIEW

The Business Intelligence Roadmap is a flexible methodology developed and promoted by Larissa Moss. It is an example of a Hybrid Approach because it can be customized to meet the needs of the given architecture that is being developed.

It is based on a series of standard project stages that have their base in solving a business problem and pursuing a business opportunity. Each stage has a series of steps with specific deliverables defined. There are a total of sixteen steps defined in the methodology. The key to flexibility is the fact that not all sixteen steps necessarily apply to allow projects. Guidelines are provide to understand which steps should be carried out based on the components that are defined by the architecture. The published guidelines provide recommendations about which steps are appropriate. Risks of not carrying out specific steps are also provided to assist the project team.

The methodology starts with understanding the business drivers that require the BI solution. The data and technology infrastructures needed are then defined at an enterprise level bounded by the scope of the business problem. This a typical balance found between bottom up and top down found in hybrid methodologies.

Common Architectures and Approaches

Twelve Factors Assessment

STRATEGIC VS. TACTICAL DECISION SUPPORT

architecture properties	methodology properties
• has structures & components to deliver the right kinds of information, in friendliest form, to executives responsible for business strategies • has structures & components to deliver the right kinds of information, in the most useful forms, to managers and knowledge workers responsible for business processes	• has specific activities & deliverables for enterprise focus in the business architecture • has specific activities & deliverables for business process alignment in the business architecture • has specific activities & deliverables to identify & express information needs of executives, managers & knowledge workers

INTEGRATION OF DATA

architecture properties	methodology properties
• explicitly assigns data stores as having responsibility to contain integrated data • recognizes data transformation rules as architectural components	• expresses scope of integration as a deliverable or part of a deliverable • includes activities to define, implement, & enforce consistent data integration rules

STRUCTURE OF DATA

architecture properties	methodology properties
• supports use of both relational & dimensional data structures • uses data structures appropriate to explicitly stated roles of data stores • avoids data stores with multiple, conflicting roles	• includes both dimensional and relational modeling activities • includes modeling at multiple levels of abstraction • offers guidance and heuristics to determine the data structure best suited for a data store

CONSISTENCY OF METADATA & BUSINESS RULES

architecture properties	methodology properties
• includes metadata components • associates metadata with other architecture components • recognizes business rules & data transformation rules as metadata	• includes metadata collection activities • includes activities & deliverables to enforce consistency of rules & metadata

SCALABILITY (GROWTH IN SCOPE)

architecture properties	methodology properties
• business architecture supports incremental expansion of scope • data architecture uses readily extensible data structures • data architecture supports incremental growth in number of subjects, number of users, and ways to use	• includes activities & deliverables that express the business scope, subject area scope, and scope of user community • iteratively reviews and adjusts scope as needed

PERSISTENCE OF DATA

architecture properties	methodology properties
• includes data store(s) with explicit responsibility to retain atomic historical data • data and technical architectures support continuously growing large volumes of data • supports ability to prospectively capture historical data not yet targeted to a data mart	• includes activities to identify data retention requirements • includes activities to define data purge/archive requirements & methods, and to build data purge/archive procedures • provides techniques and guidance for prospective data capture

SUSTAINABILITY

architecture properties	methodology properties
• manages component redundancy to limit impact of change • has a relatively simple, readily understood set of components, roles, and relationships • able to limit impact of change with ETL technology	• includes activities & deliverables to recognize and respond to change

Common Architectures and Approaches

Twelve Factors Assessment

ASSESSMENT CRITERIA

Each of the approaches is evaluated against the twelve using the norms shown on the facing page and below:

TECHNOLOGY REQUIREMENTS

architecture properties	methodology properties
• not dependent on one or a very few specific products • adaptable to a broad range of technologies and products	• includes activities & deliverables to identify technology needs, and to select and deploy products & tools

STAFFING & SKILLS REQUIREMENTS

architecture properties	methodology properties
• relatively few kinds of components & relationships among them • readily understood structure of architecture components	• relatively few kinds of activities and deliverables • fully described and well-defined methodology

TIME TO DELIVERY

architecture properties	methodology properties
• business architecture that defines the scope of the warehousing program • projects with small scope	• activities to assess readiness & verify that the warehousing program has an achievable scope • incremental & evolutionary deployment through a series of small projects

COST TO DEPLOY

architecture properties	methodology properties
• business architecture that defines the scope of the warehousing program • limited start-up time followed by projects with small scope • deployable with limited, moderate cost technology	• incremental & evolutionary deployment through a series of small projects • not dependent on one or a few costly products

COST TO OPERATE

architecture properties	methodology properties
• relatively few kinds of components & relationships among them • business architecture that limits the scope of the warehousing program • organizational architecture with defined responsibilities to monitor, verify, & support • triggering, scheduling, & error recovery requirements that may be automated with ETL tools • limited data redundancy to contain storage costs • limited process redundancy to minimize redundant extracts & transformations	• includes activities & deliverables to design triggering, scheduling, verification, & error-recovery procedures • includes activities & deliverables to monitor, manage, and support the data warehouse

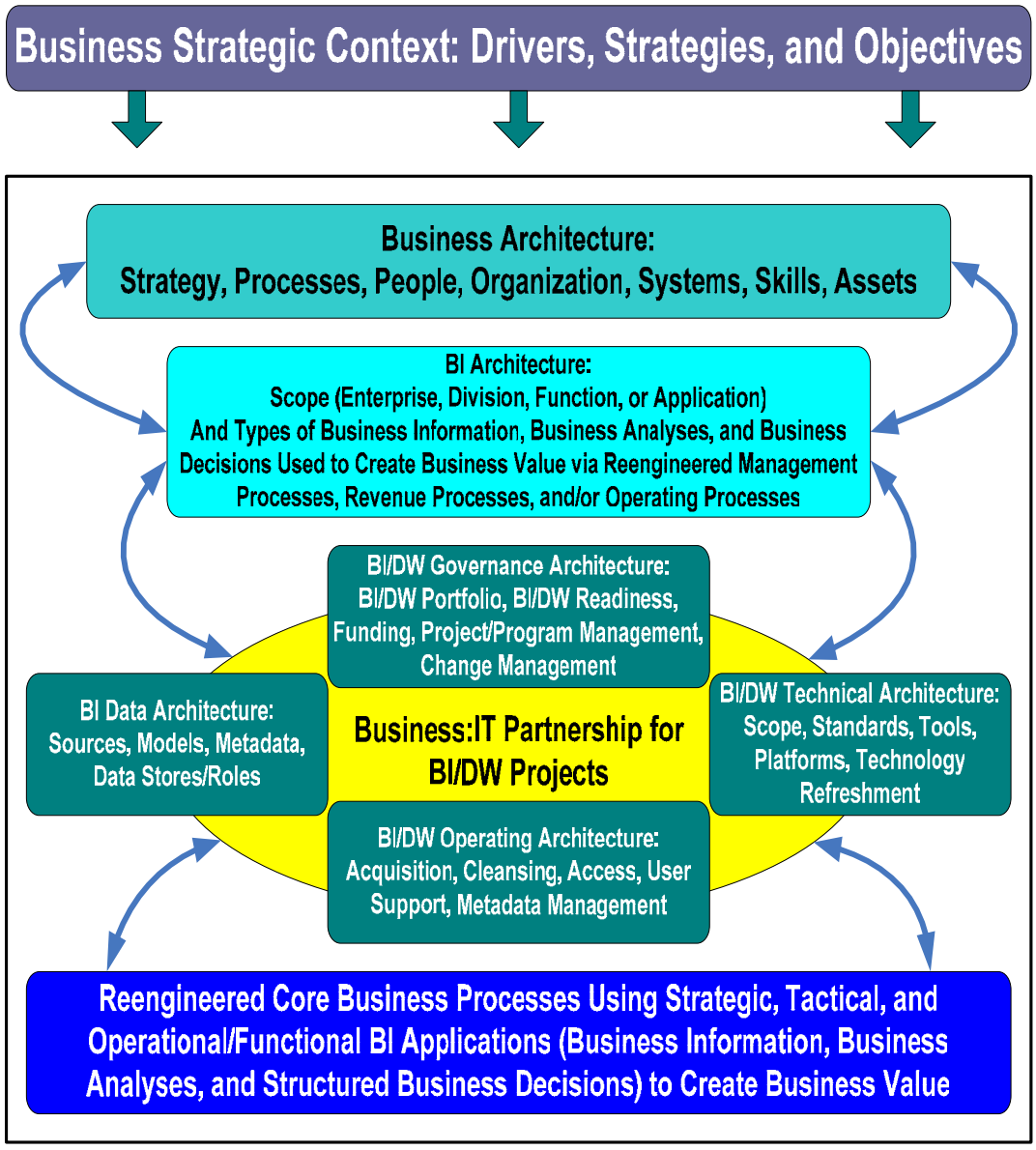
Module 2

Enterprise-Oriented Approaches

This page intentionally left blank.

BI Pathway

Business Intelligence Architecture Overview



BI Pathway

Business Intelligence Architecture Overview

BUSINESS ARCHITECTURE

This includes the strategies (hopefully aligned), core business processes, organizational structures, systems, assets, skills, and people that an organization uses to serve its customers and create stakeholder value.

BI ARCHITECTURE

This includes: (1) formal scope of BI initiatives within an organization, line-of-business, division or application; (2) portfolio of BI initiatives to be undertaken, e.g. strategic, tactical or operational; (3) core management processes, revenue-generating processes, and operational processes whose performance will be improved; and (4) specific value capture mechanisms associated with each initiative, including performance measures.

BUSINESS AND IT PARTNERSHIP

BI initiatives deliver business value through an effective Business and IT Partnership and high-performing BI/DW project teams, enabled by four separate architectures.

- **BI/DW GOVERNANCE ARCHITECTURE**

  This includes: (1) the BI/DW Portfolio; (2) risk identification and management facilitated by readiness assessments; (3) consistent funding at appropriate levels; (4) effective BI/DW program and project management methodology; and (5) change management for technical and business process changes.

- **BI/DW DATA ARCHITECTURE**

  This includes best practices, including: (1) analyzing and cataloging data sources; (2) identifying requirements for business information, business analyses, and structured business decisions to define the data and metadata requirements; (3) definitions of data stores, their roles, and associated data structures; and (4) metadata requirements

- **BI/DW TECHNICAL ARCHITECTURE**

  This includes activities to deliver a robust technical environment, including: (1) capacity planning for the scope of BI initiatives; (2) technology platforms; (3) data and technical standards; (4) tools for the BI/DW information creation process (source to data to BI applications to user access); (5) procedures for technology refreshment and scalability.

- **BI/DW OPERATING ARCHITECTURE**

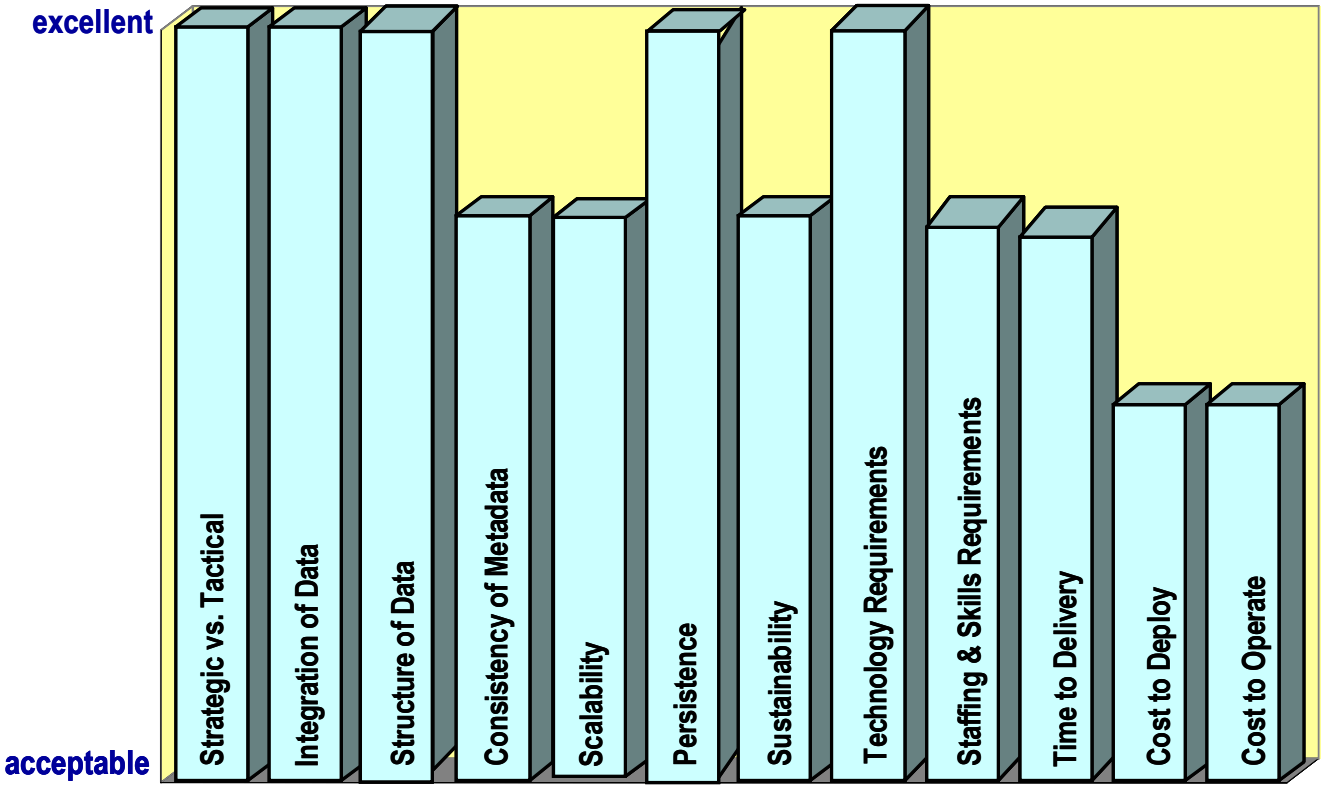
  This includes the high-quality processes for BI/DW, including; (1) source data acquisition and intake; (2) data profiling and cleansing; (3) refreshing data; (3) managing metadata; and (4) supporting users.

REENGINEERED CORE BUSINESS PROCESSES

BI initiatives create value by changing management processes, revenue-generating processes, and/or operating processes to become more effective and/or cost efficient. Accordingly, the reengineered business processes are the essential component in the architecture and the means by which business value is created.

BI Pathway

Twelve Factors Assessment



BI Pathway

Twelve Factors Assessment

BI PATHWAY ASSESSMENT

The graph shows a subjective evaluation of the BI Pathway approach against the twelve factors described in Module One. All factors are shown on a qualitative scale ranging from "acceptable" to "excellent". This is intended to show how the evaluation approach can be used to assist individual project teams in their selection of an approach. All of the architecture and methodology norms are listed for each of the twelve factors in Appendix C. In the details shown in the Appendix, items considered to be strengths are show in **bold** text. Items considered to be particular weaknesses are shown in *italic* text.

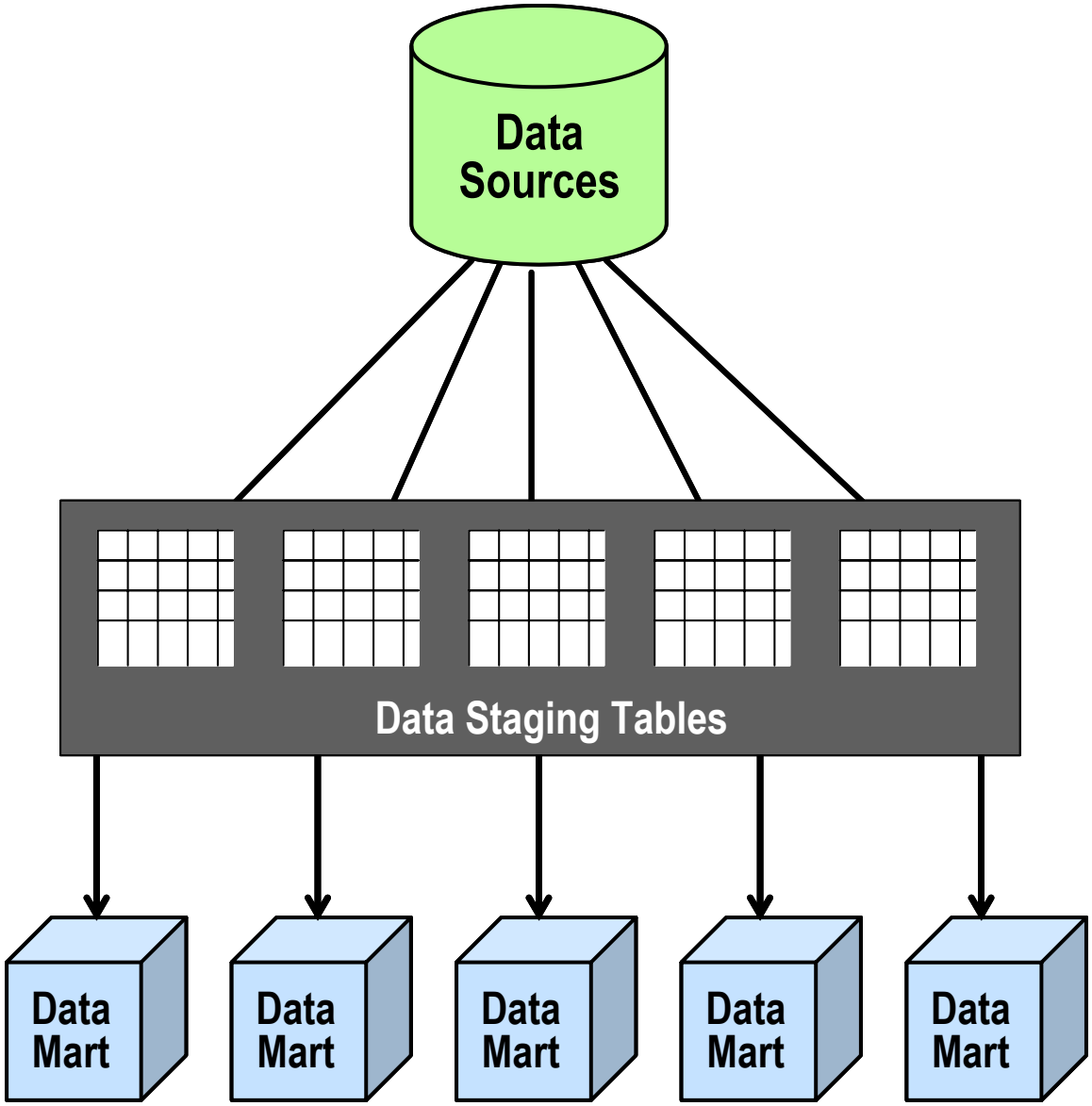
Module 3

Data Mart-Oriented Approaches

This page intentionally left blank.

Data Warehouse Lifecycle

Architecture Overview



Data Warehouse Lifecycle

Architecture Overview

KIMBALL LIFECYCLE COMPONENTS

At an abstract level, this architecture is the least complex among all popular data warehousing approaches. It has the fewest number of components and is specifically aligned with methodology whose goal is rapid production of data marts. The lifecycle components are:

Data Mart – Kimball's *The Data Warehouse Toolkit* provides two definitions of data mart: (1) "A logical and physical subset of the data warehouse's presentation area." and (2) "a flexible set of data ideally based on the most atomic (granular) data possible to extract from an operational source, and presented in a symmetric (dimensional) model that is most resilient when faced with unexpected user queries."

Data Source – The *Data Warehouse Toolkit* defines a source system as "an operational system of record whose function it is to capture the transactions or other performance metrics from a business's processes. Alternatively, the source system may be external to the organization but is still capturing information that is needed in the data warehouse."

Data Staging Tables – Data staging is described in this life cycle to include all ETL processing necessary to populate data marts. Data staging tables are all of the intermediate data stores needed as data moves through the ETL processes. Staging data typically persists only as long as needed to populate a data mart.

Note that in this architecture all data marts are dimensional, and the data warehouse exists only as a composite of all data marts.

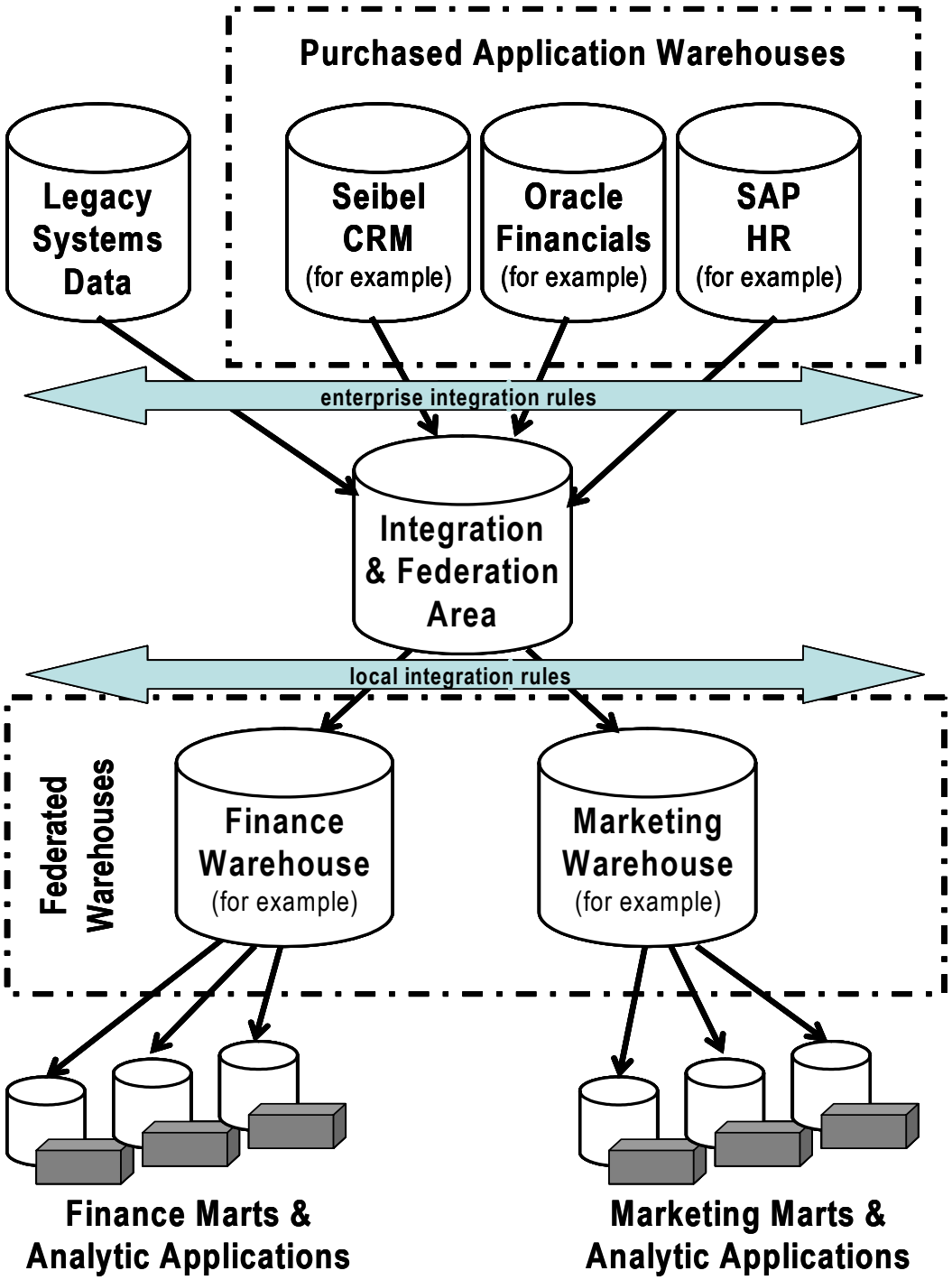
Module 4

Federation & Hybrid Approaches

This page intentionally left blank.

Federation

Architecture Overview



Federation

Architecture Overview

FEDERATED ARCHITECTURE OBJECTIVES

Doug Hackney's federated approach is unique in the field of warehousing architectures. Federation recognizes the reality in some organizations that non-architected components already exist and must be integrated. In an environment that includes any of:

- packaged applications that include some BI components,
- ERP modules that deliver some proprietary data warehousing pieces,
- existing non-architected data marts (Hackney terms these "legamarts")

Data integration is an entirely different challenge than when starting with a clean slate. In such an environment, federated enterprise architecture establishes standards to provide a "single version of the truth" by sharing key metrics, measures, and dimensions across a collection of business intelligence systems.

ENTERPRISE DATA MART ARCHITECTURE

Federation targets data mart consistency with an enterprise data mart architecture (EDMA) comprised of:

- Common sources – Identification of a single "master" source for each dimension key and each metric.
- Common dimensions – Standards of definition, structure, and membership for customer, product, time, geography, organization, and other hierarchies used across the enterprise.
- Common business rules – Standard algorithms to calculate metrics, and to derive classification and structure of business data.
- Common semantics – Conventions of terminology for the language that the business uses to label itself, its elements, its structure, its metrics, and its activities.
- Common metrics – Standard business performance measures.

ENTERPRISE DATA WAREHOUSE ARCHITECTURE

In addition to EDMA, an enterprise data warehouse architecture (EDWA) is needed to identify:

- The inventory of business intelligence (BI) solutions that exist.
- The collection of data contained in those systems (as a data model).
- The flow of data through BI systems.
- An assessment of the value and viability of each BI system.
- An integration and federation plan.

ABOUT THE DIAGRAM

Federation architecture is unique for each organization, and is best illustrated by example. This diagram illustrates federation by example (Seibel, SAP, etc.) rather than by abstraction.

Module 5

Architecture and Methodology Decisions

This page intentionally left blank.

Methodology Selection

Evaluation Criteria

criterion	description
process completeness criteria	
results oriented	has a well-defined set of deliverables and a distinct product
fully described components	each component is fully described and has a defined role
cohesion of results	Every result produced throughout the process has a reason to exist
rigor in the process	is detailed enough to ensure no gaps in flow of results and activities
appropriate level of detail	sufficiently detailed to achieve desired rigor, but not excessively detailed
familiarity of techniques	uses familiar, common, and proven techniques to produce results
process flexibility	adaptable to unique needs of organizations, projects, and teams
project planning usefulness	is useful as a planning template for projects
role/responsibility identification	identifies roles and responsibilities to perform activities & produce results
process usability criteria	
adaptable	quickly and readily adjusts to unanticipated project circumstances
model based	supports and employs modeling at multiple levels of abstraction
goal driven	facilitates results-based definition and measurement of project goals
traceable	results are fully traceable through a network of deliverable dependencies
teachable	can readily be learned by anyone with the requisite skills, & experience
documented	each component (activities, deliverables, etc.) is documented
team enabling	process & deliverable dependencies help to identify team dependencies
referenceable	has a community of users who can attest to its usability
measurable	provides ability to track process and project metrics
data warehouse enabling criteria	
scaleable	not size and scope dependent; works for warehouse, mart, and ODS
comprehensive	includes a robust set of results for all parts of the warehousing product
evolutionary	uses concept of multiple, small projects to accomplish large objectives
business information focused	business information needs are significant as both inputs and results
data structure independent	without data structure bias; works for both relational & dimensional data
acquisition method independent	without acquisition technique bias; works for both “push” and “pull”
vendor & tool independent	not specifically dependent on a single vendor’s tool set

Methodology Selection

Evaluation Criteria

METHODOLOGY CONSIDERATIONS

As already discussed, architectural decisions and methodology choices are tightly coupled. The right architecture with an incompatible methodology has little chance of success. And a methodology that is poorly suited to the organization and people who must employ it also risks failure. It is important to have:

- a data warehousing architecture fitted to the expectations, needs, and constraints of your data warehousing program,
- a data warehousing methodology appropriate to the organization and people who must put it in practice, and
- a good match between architecture and methodology.

METHODOLOGY GUIDELINES

The facing page itemizes several criteria that are useful when evaluating methodologies. These criteria and their application are fully described in Appendix A, which contains reprints of a three-part series of articles previously published in *The Journal of Data Warehousing:*

Evaluating Data Warehousing Methodologies: Objectives & Criteria

Evaluating Data Warehousing Methodologies: An Evaluation Process

Implementing Data Warehousing Methodology: Guidelines for Success

Architecture Selection

Organizational & Environmental Factors

weight	criteria	CIF		step by step		BDW		Lifecycle		Rapid Marts		Collective		Federation	
		raw	wgt	raw	wgt	raw	wgt	raw	wgt	raw	wgt	raw	wgt	raw	wgt
	strategic vs. tactical														
	integration of data														
	structure of data														
	consistency of metadata														
	scalability														
	persistence of data														
	sustainability														
	technology														
	staffing & skills														
	time to delivery														
	cost to deploy														
	cost to operate														
	process completeness														
	process usability														
	data warehouse enabling														
	overall weighted average														

Architecture Selection

Organizational & Environmental Factors

EVALUATION & DECISION TOOL

A spreadsheet similar to the table shown on the facing page is a useful tool to evaluate your needs and help to make architecture and methodology decisions. Use the table or spreadsheet with these steps:

1. Examine each of the twelve architectural assessment factors and the three categories of methodology criteria to identify expectations and constraints specific to your data warehousing program. Assign each a weighting factor using the scale below, or one of your own design:

   2.0 = critical success factor for the warehousing program
   1.0 = tactical success factor for warehousing projects
   0.5 = not a concern for the warehousing program or projects
   0.0 = contrary to the goals of the warehousing program

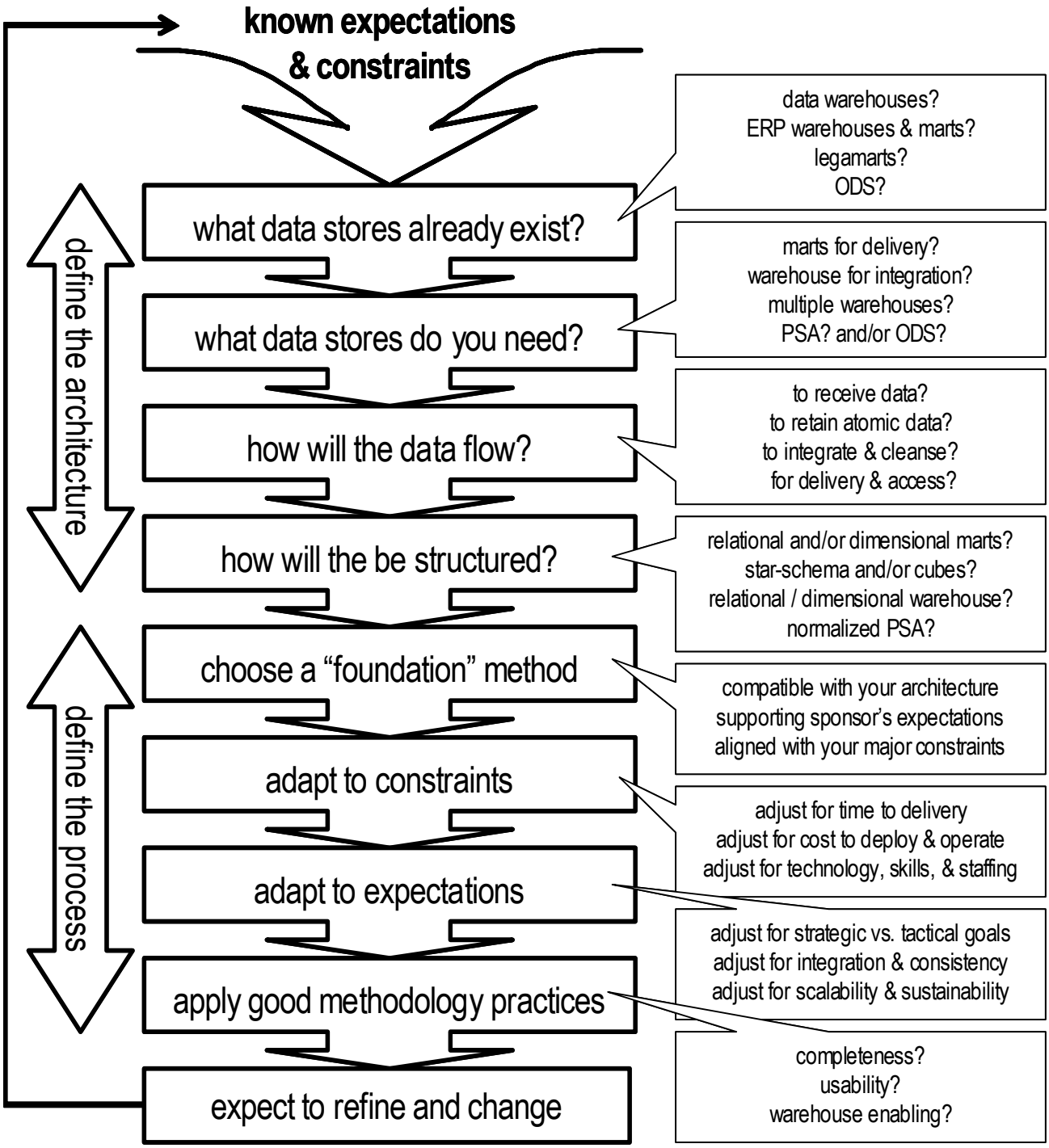
2. For each combination of approach and factor, multiply the raw score from the "all factors equal" assessment by the weighting factor to calculate a factor weighted average.

3. Sum the factor weighted scores for each approach, and divide the sum by 15 to calculate an overall weighted average for each approach.

This evaluation prepares you to make *informed* architecture and methodology decisions.

Conclusions

Designing & Implementing a Unique Hybrid Approach



Conclusions

Designing and Implementing a Unique Hybrid Approach

SOME GUIDELINES The diagram on the facing page depicts a process to guide development of a hybrid approach tailored to the unique circumstances of your data warehousing program. The table below provides further detail.

Define the Architecture	
what data stores already exist?	• If you have multiple data warehouses, disconnected ERP warehouses, or "legamarts" you may want to include some of the elements of federated architecture
what data stores do you need?	• If you only need data marts for delivery of information (business unit focus instead of enterprise focus) the Kimball architecture is a good fit • If you need a data warehouse in the near term to provide enterprise focus consider CIF, BDW or the three-tier architecture • If you eventually need enterprise integration but not in the near term consider Mimno's architecture • If you need multiple but related data warehouses look at federation • If you need an ODS consider CIF • If you need to retain historical data at an atomic level consider three-tier warehousing's PSA
how will the data flow?	• If you need to capture and retain atomic level data include the PSA concept from three-tier warehousing • If you need to separate integration and cleansing from delivery and access look at a two- or three-tier architecture • If you need to isolate delivery (data marts) from source system changes consider BIW, CIF, or three-tier warehousing
how will the data be structured?	• If dimensional data structures meet all of your data warehousing needs consider Lifecycle • If you need the extensibility of relational data stores, use a CIF, BDW, or three-tier warehouse • If you need to have data in the third normal form, look at the PSA component of three-tier
Define the Process	
choose foundation method ***then*** **adapt to constraints & expectations**	• If the architecture includes a warehouse distinct from data marts exclude the Lifecycle approach • If the architecture has only data marts start with the Lifecycle approach • If your sponsor expects rapid delivery, consider bottom-up methods • If your sponsor expects an enterprise-wide solution, consider top-down methods • If time or cost are major constraints consider bottom-up methods • If volume, complexity, and quality of source data are constraints consider top-down methods • If volatility of source systems is a constraint consider top-down methods • If elusive business requirements are a constraint consider CIF • If staffing and skills are a constraint consider bottom-up methods
apply good methodology practices	• Check & adjust for completeness (does it produce everything you need?) • Check & adjust for usability (is it documented, understandable, and tool supported?)
Expect to Refine & Change	