Evaluating Data Warehousing Methodologies: 
Objectives and Criteria

by Dr. James Thomann and David L. Wells

With each new technical discipline, Information Technology (IT) practitioners seek guidance for the “what and how” of putting the technology to work. Even with the mixed reviews that methodology received through the Computer Aided Systems Engineering (CASE) movement of the 1980’s, this guidance is usually sought in the form of methodology. With rapidly growing interest in data warehousing, a corresponding desire for data warehousing methodology is not surprising. A number of methodologies, both formal and informal, have emerged since Bill Inmon first popularized data warehousing. Counted among the formal methodologies are those of Inmon, Kelly, and Hadden. The informal methods include those embedded in course materials, “how to” books, and data warehousing tools. The purpose of this article is to put forth a set of criteria by which organizations can evaluate and select among the multiple offerings of data warehousing methodology.

The Data Warehousing Institute is frequently asked to recommend methodology, but is reluctant to do so. The “best” data warehousing methodology may well be different for each organization depending on needs, size, level of process maturity, and other factors. Although the Institute does not want to recommend methodology, we do want to offer some guidance in making the choice. This article starts that guidance effort by looking at methodology purpose and history, offering a quick assessment of the state of data warehousing methodology, and setting forth formal criteria for methodology evaluation and usage.

Methodology: Purpose and Short History

The strict meaning of methodology is “study of methods” but in IT the term has become synonymous with the methods themselves. In the early 1970’s (when the earliest IT methodologies of Yourdon, Constantine, etc. were emerging) a group at the University of Massachusetts undertook a study of methodology. This group distinguished between method and methodology by defining a methodology as the formal, documented and rigorous way to accomplish something. They defined a method as a less formal approach. The definition of methodology that emerged from this group is “a detailed set of steps or procedures to accomplish a defined goal”.

The key words in the above definition are defined goal. The primary purpose of a methodology is to achieve a predictable result. A secondary goal of methodology is to provide a process that is repeatable, trainable and consistent. Perhaps the greatest number of failed methodology implementations arises from too much focus on tasks and too little focus on the results. When practicing methodology becomes an exercise in checking off steps the product will certainly suffer.

The methodology work of the 1970’s came to the fore in the mid-1980’s with the advent of CASE technology. The objective was automation of methodology, and the effort continued to focus on activities more than on results. Many CASE failures can be directly attributed to the expectation that methodology builds systems and people simply execute the methodology. The real lesson of the CASE era is that it takes good people to build good systems.

---

2 Thomann, Meta-methodology: A Methodology to Create Methodologies, University of Massachusetts, Amherst, MA, 1974.
In 1993, Michael Hammer and James Champy brought new thinking to the methodology field with their manifesto, *Reengineering the Corporation*. Hammer and Champy describe a concept of business process — a series of activities that receives inputs and produces an output (product) that has value to a customer. They emphasize that the key to successful business process orientation is focus on products and customers. And they state that most organizations have institutionalized business processes that need to be *reengineered* to regain product and customer focus.

Although Hammer and Champy were not focused on information technology, the relevance of their message is too striking to ignore. Methodologies are, in fact, the business processes of IT. And many of those processes need to be reengineered. The misplaced focus — activities instead of customers and products — is precisely the problem with many methodologies. A look at business process components, then, provides a basis to understand and apply methodology evaluation criteria. The key components of any business process (including IT methodologies) are:

- **Outputs / Products**: A business process may produce two kinds of products — an end-product and by-products. It is important to keep focus on the end-product as the output most highly valued by customers.
- **Inputs**: The raw materials used to produce a product each need to make a direct and known contribution to that product. Inputs may be products of other internal business processes or may be externally supplied. Quality of input directly affects product quality, and source of input may influence quality of input.
- **Activities**: The steps taken to produce a product — combining or transforming inputs to add value — are the process activities. As with inputs, each activity must make a direct and known contribution to the product. There may be multiple paths through the activities, with a single main path handling normal cases and multiple alternative paths designed to handle known exceptions.
- **Events**: Events stimulate action throughout the process. A triggering event starts process execution. Other events may cause branching through the alternative paths of the process.
- **Actors / Roles**: A role represents a set of skills and responsibilities required by the process. Actors are the people who fulfill the roles. One actor may perform in multiple roles, and a role may be filled by more than one actor.
- **Heuristics**: Heuristics are a set of detailed explanations, techniques, examples, and rules of thumb that combine to form a “help system” for the process. When in doubt about some aspect of the process, turn to the heuristics for guidance.

These components, and the focus on product and customer, provide a strong foundation for sound methodology and effective use of that methodology. We use this process-orientation to establish evaluation criteria. And we choose to discuss the object of these criteria as *data warehousing processes*, and not as methodologies.

**Data Warehousing Processes – The Current State**

The quest for data warehousing processes is similar to that described for previous IT methodological efforts. A defined process for building data warehouses is seen as a mixed blessing — necessary, but problematic. There are a number of DW processes available for use, both formal and informal as previously discussed. Among these processes there are many similarities and some significant differences. Let’s first explore the similarities.

At a very high level, most of the current processes are divided into two major segments: **architecting and implementing**. Architecting establishes a vision for the data warehouse and plans the implementation sequence. Architecture addresses business information needs, data warehouse configuration, and standards and conventions at an

---

4 One of the most common ways that a process is institutionalized is by organizational compartmentalization. Each organization unit is focused on its activities, and the enterprise view focuses on the flow of activities. Most of us have experienced these processes as bureaucracy and red tape.

5 Hackney, *Understanding and Implementing Successful Data Marts*, page 135.
enterprise or broad scope level. Implementing is performed as a series of small projects, each of which delivers a subset of the data warehouse with measurable business value.

The other area of high similarity among most data warehousing processes is the components of the warehousing product. Virtually all of the processes provide for analysis, design, construction and/or usage of the following components:

- **Source data** – the operational and external data needed to populate the data warehouse.
- **Extract components** – automated procedures designed to remove (copy) required data from the source environment.
- **Transform components** – automated procedures to change the extracted data into forms that assume data warehouse characteristics and meet business information needs.
- **Load components** – automated procedures that place transformed data into the data warehouse.
- **Data Warehouse (or Data Mart)** – the storage containers of the transformed data available for use by the business.
- **Access components** – the means for business people to access the data warehouse to meet their information needs.
- **Metadata** – data about warehouse contents and warehouse processing that is needed to use, maintain, and administer the data warehouse.
- **Data cleansing components** – automated procedures that detect and repair data quality defects.
- **Data archiving components** – automated procedures and storage facilities for permanent retention of historical data.

The differences among available data warehousing processes are many and varied, but mostly related to the details of the process. Common areas of difference include:

- Organization - the activities are structured, sequenced, or grouped differently.
- Terminology – different terms for the same concept, or the same terms with different meanings.\(^7\)
- Deliverables - different sets of deliverables directed at producing the same end product, a data warehouse or data mart.
- Techniques and heuristics – unique approaches to the activities, especially for information gathering.
- Levels of detail – both in specificity of deliverables and of activities.
- Cohesion and rigor – differing levels of process discipline and completeness.
- Tool independence – some are dependent on specific automated tools and are optimized to work with only those tools.

We consider these differences, within a common process framework, to be good news. Just as there are many and diverse IT organizations, so too must there be diversity in data warehousing processes. Formal and systematic process evaluation is the means to determine the best process fit for your organization.

### A Framework for Process Evaluation

Systematic assessment of IT business processes is based on meaningful, measurable criteria that assess some aspect of the processes’ ability to achieve the goal of producing a customer-valued product. Figure 1 illustrates the structure of evaluation criteria that we have developed for data warehousing processes. These criteria are derived from experiences with many IT processes, including processes for application development and for data warehousing. The assessment structure begins with a foundation of general process criteria that focus on answering two questions:

- How complete is the process?
- How useable is the process?

\(^7\) For this reason the Data Warehousing Institute has undertaken a project to achieve consensus among industry experts and some key practitioners on a standard set of data warehousing terms, and to produce a glossary of those terms.
Beyond the criteria that apply generally to all IT business processes, data warehousing processes have some unique requirements. Specific to data warehousing processes, we include a third set of criteria targeted at the question:

- How effectively does the process enable development, deployment, and sustained operation of data warehouses?

![Figure 1: Data Warehousing Process Evaluation Criteria](image)

**Process Completeness Criteria**

Completeness is the essential first level of process evaluation. Ensuring that the process has all of the components and attributes necessary to achieve its defined goal is fundamental to process evaluation. A process that is readily understood and highly usable, but that does only part of the job serves only to create a false sense of security in failing projects.

Consider the following criteria when evaluating the completeness of any IT business process:

1. **Results Oriented**: The process offers a consistent well-defined set of deliverables. Each deliverable has a described role and known contribution that it makes toward producing the product. Dependencies among deliverables are known. The highest-level process overview is described as a flow of results.
2. **Fully Described Components**: Each process component is fully described and its role in the process is documented. The minimum set of components includes results (deliverables), inputs, activities, events, and roles. Robust descriptions will include examples and heuristics.
3. **Cohesion of Results**: Every result produced throughout the process has a reason to exist. Results produced late in the process are based upon earlier results. No result is produced that does not contribute to production of the end product. At each stage of the process, appropriate attention is given to all dimensions of the product (e.g., data, function, location, organization, and timing).
4. **Rigor in the Process**: The process is detailed enough to ensure no gaps in flow of results or in flow of activities. All inputs are associated with activities that use them, and all results are associated with the activities that produce them. The normal path of product development through the process is clear and unambiguous. Common exceptions are identified and associated with the events or conditions by which they are recognized. Alternative activity flow is identified where required by exceptions.
5. **Appropriate Level of Detail**: The process is sufficiently detailed to achieve necessary rigor. Although common exceptions are addressed, the process does not account for rare exceptions and remotely possible contingencies. Sufficient detail is provided to describe what each result should be, but no attempt is made to prescribe how the
results are produced. (Note: Heuristics may address how to produce results, but they are guidance, not prescription.)

6. **Familiarity of Techniques**: The process uses familiar, common, and proven techniques to produce results where practical. The de facto standard for data modeling, for example, is entity-relationship modeling. Ideally, a process will reuse proven techniques where they serve to produce the desired result, adapt existing techniques where necessary to produce the result, and introduce new techniques only when reuse or adaptation do not meet the needs of the process.

7. **Process Flexibility**: The process can be adapted to meet unique needs of different organizations, projects, teams, and individuals. Adaptation is accomplished by adding to or deleting from the set of results and activities. When results and activities are added to the process, they can be easily embedded into the process flow. When results and activities are removed from the process, the impacts on cohesion and rigor are readily identified.

8. **Project/Planning Usefulness**: The process is useful as a planning template for projects. Activities are layered or structured hierarchically, and are similar in magnitude or relative importance at each level. Activities serve as the basis to develop a work breakdown structure for a project. Results provide the foundation to establish project milestones. Role assignments and heuristics help to estimate project effort.

9. **Role/Responsibility Identification**: The process identifies the roles required to achieve results and associates roles with the activities to produce the results. Roles are described as sets of skills. Where multiple roles are associated with an activity, the responsibilities of each role are identified.

---

**Process Usability Criteria**

Completeness is not sufficient to ensure process success. Remember the earlier assertion that good systems are built by good people. An effective IT business process, then, must be people friendly. We further assert that building good systems requires teams of people working in a project structure. Thus, an effective system development process must be project and team oriented. A useable process has the following characteristics:

1. **Adaptable**: The process is results-based, not rules-based. It is quickly and readily adjusted to adapt to unanticipated project circumstances. Activities and results may be added, removed, or combined as needed to meet the unique and immediate pressures of projects. (Note: Adaptability is similar to flexibility as discussed in completeness criteria. The key distinction is one of timing and intent. The completeness criterion addresses planned adjustments to implement a process in an organization. The usability criterion targets need to make unplanned adjustments in the midst of a project.)

2. **Model Based**: The process recognizes the value of modeling at multiple levels of abstraction. The minimum set of modeling levels includes conceptual, logical, and physical modeling. A robust process also provides support for context modeling. The process also recognizes the importance of modeling multiple product dimensions. The minimum set of dimensions includes data, function, and location. Robust processes include modeling support for modeling of organization, time, and motivation.

3. **Goal Driven**: The key to process execution is clearly defined, measurable, and attainable goals. The process facilitates results-based definition of project goals. It enables project management and project tracking through measures of goal achievement.

4. **Traceable**: All results of the process are fully traceable, both forward and backward, through a network of deliverable dependencies. Forward traceability ensures that when any input or result is changed, the impact on subsequent results is identifiable. Backward traceability enables reverse tracking from a product through the chain of intermediate results and back to the original inputs from which the product was produced. Sources and causes of product defects may be readily identified through reverse traceability.

5. **Teachable**: The process can readily be learned by anyone with the requisite skills, aptitude, and experience. Learning resources are provided for the process. Minimum learning resources are documentation, examples, and a self-study guide. Robust processes support learning with classroom training, hand-on workshops, and mentoring services.

6. **Documented**: Each component of the process – deliverables, activities, heuristics, and process-unique techniques – is documented. References are provided to documentation of reused techniques. The documentation is organized and structured to serve as an easy-to-use guide for practitioners of the process.
7. **Team Enabling:** Process goals serve as the common goals that bond a team. Dependency of results is useful to identify dependencies among team members. The process describes relationships and dependencies among roles.
8. **Referenceable:** The process has a community of users who can attest to its usability. Process user groups or communities of practice exist and are accessible.
9. **Measurable:** The process provides the ability to track metrics that are needed to manage projects and effect process improvements.

**Data Warehouse Enabling Criteria**

The following criteria describe desirable characteristics that are specific to data warehousing processes. A process that fails on these criteria will not deliver a data warehouse or data mart that will actually be used.

1. **Scalable:** The process is not size and scope dependent. It can be employed to build data warehouses, data marts, or operational data stores. It is equally as effective for building enterprise-level data warehouses as for building independent data marts.
2. **Comprehensive:** The process includes a robust set of results for each component of the data warehousing product. Each component has a continuous thread of results that begins with initial inputs and ends with the completed product. The minimum set of product components includes source data, extract procedures, transformation procedures, warehouse loading procedures, archiving procedures, metadata facilities and data access procedures. Extended processes include threads of results for data cleansing procedures, for organizational outcomes (training, support, communication, etc.), for information delivery applications (standard queries and reports, DSS/EIS systems, etc.), and for operational needs (monitoring requirements, refresh procedures, backup and recovery requirements, etc.).
3. **Evolutionary:** The process embraces the principle of multiple, small projects to accomplish large objectives. It includes activities and results to decompose efforts that are too large to be manageable, and to package them as a collection of dependent projects. Heuristics for project size, scope, and dependency are included in the process.
4. **Business InformationFocused:** The earliest and most significant process inputs are related to business needs for information. Information needs are named and described as results of at least one activity. Information needs are forward traceable to the product components which implement their solution. Each product component is backward traceable to the information needs that it supports.
5. **Data Structure Independent:** The process is neither limited to nor biased toward a particular method of structuring data. It implements relational and dimensional data warehouses with equal effectiveness. It recognizes that a single data warehouse may contain both dimensional and relational data. The process provides heuristics to choose among alternative data structures. Techniques, heuristics, and results for aggregated, partitioned, and summary data structures are included in the process.
6. **Acquisition Technique Independent:** The process is neither limited to nor biased toward a particular technique for acquiring data. It implements “pull” and “push” approaches equally well, and recognizes that a single data warehouse environment may have need to use multiple techniques of both the “push” and “pull” types. Techniques and guidelines for choosing among acquisition techniques are provided.
7. **Vendor and Tool Independent:** The process is not specifically dependent on the tool set of a single vendor. It is practical to implement and use the process with no requirement to discard old tools or acquire new ones. Retooling your data warehousing environment will not demand extensive process reengineering.

**Conclusions**

This article describes the term data warehousing process, discusses the need for formal evaluation of data warehousing processes, and puts forth a set of criteria by which they may be evaluated. It takes the first step in addressing the questions surrounding data warehousing methodologies. Here, we have established the foundation for you to evaluate and select among data warehousing processes. It is, however, only a first step.
This is the first in a series of three articles. The second, *Evaluating Data Warehousing Methodologies: An Evaluation Process*, describes a process and techniques for putting these criteria to work. We describe a process that you can use (or adapt) to conduct your own evaluation. We also discuss the work being conducted by The Data Warehousing Institute to assess current data warehousing processes.

The final article in the series, *Implementing Data Warehousing Methodology: Guidelines for Success*, describes the challenges that you’ll face once a data warehousing process has been selected. Issues of adaptation, change, and acceptance are among the foremost of these challenges. We’ll provide discussion of each implementation issue, and provide guidelines and tips to overcome implementation barriers.