WILEY

DATA
SHARING

Using a Common Data Architecture

Michael H. Brackett
Foreword by John Zachman

▼ DATA SHARING
▼ USING A
▼ COMMON DATA
▼ ARCHITECTURE

Michael H. Brackett

This text is printed on acid-free paper.

This publication is designed to provide accurate and authoritative information in regard to the subject matter covered. It is sold with the understanding that the publisher is not engaged in rendering professional services. If legal, accounting, medical, psychological, or any other expert assistance is required, the services of a competent professional person should be sought. FROM A DECLARATION OF PRINCIPLES JOINTLY ADOPTED BY A COMMITTEE OF THE AMERICAN BAR ASSOCIATION AND A COMMITTEE OF PUBLISHERS.

▼ ABOUT THE AUTHOR
▼
▼

Mr. Brackett is the Data Resource Coordinator for the State of Washington with the Department of Information Services. He has been in the data processing field over 30 years and has worked for the State for 27 years in a variety of agencies, including the Department of Natural Resources, Department of Fisheries, Employment Security Department, and Washington State University.

Mr. Brackett has developed many innovative concepts and techniques for designing applications and data. He is currently developing a common data architecture for the State that includes State agencies, cities, counties, Indian tribes, public utilities, and Federal agencies. That common data architecture covers large multi-jurisdictional areas, such as water resource, growth management, criminal justice, and health care.

Mr. Brackett has a BS in Forestry and an MS in Forestry (Botany) from the University of Washington, and an MS in Soils (Geology) from Washington State University. He has written and published three previous books on application and data design. He has also written many articles and given numerous presentations at local, national, and international conferences on the topic of application and data design.

Mr. Brackett is a past board member for the Seattle Chapter of the Data Resource Management Association. He teaches Data Modeling and Design in the Data Resource Management Certificate Program at the University of Washington. He also teaches Data Architecture classes for the State. He is listed in *Who's Who in the West* and *Who's Who in Education*.

▼ FOREWORD

▼ BY JOHN A. ZACHMAN

When Michael Brackett initially suggested that I write a foreword to his latest book, I asked him to send me a copy of the manuscript so I could decide a.) whether I was competent to make a meaningful contribution, and b.) whether I actually liked the book. I am simply unable to write a foreword to a book unless I really like it. Needless to say, I really liked Michael's book.

I would call *Data Sharing Using a Common Data Architecture* a Vince Lombardi kind of a book. For those readers who are not football fans and don't know who Vince Lombardi was, what I mean by this metaphor is, this is a "block and tackle" book, a "back to the basics" approach to information systems and data architecture that is very refreshing.

From the very beginning of data processing, we have been inundated with technological panaceas, some kind of magic that would somehow obviate hard work...thinking-kind of hard work. My opinion is that not only are the users, in Michael's terminology, "clients," looking for "silver bullets," but those of us who are information professionals are equally culpable! Every time a new technology or a new tool comes on the scene, it always seems to be perceived to be the final word, the elixir that will forever make all pain go away. I think this "silver bullet" mentality has set very unrealistic expectations, and presently is causing severe problems for the information community.

As this book is published, it is my personal perception that the credibility of the information profession is at an all-time low. As the world moves relentlessly, at break-neck speed, into the Age of Information, public and private enterprises alike are finding themselves under enormous stress—restructuring, reengineering, downsizing, transforming, redefining their roles in a new and dynamic global marketplace. Unfortunately, I/S is being perceived as an inhibitor of the enterprise's

efforts to survive the dramatic changes taking place all around and within it. It seems that it always takes I/S years and hundreds of thousands of dollars to respond to the enterprise's demands, or else "we can't even do it at all!"

I have had the opportunity to work with a very prestigious and very competent I/S organization that supports a major division of a very respected aerospace giant. They had one of the "Big Six Consulting Firms" analyze the quality of their information systems support of this division, and the final report basically said that I/S "takes too long, costs too much and doesn't meet expectations." Unfortunately, I think this is the rule rather than the exception for the information profession as a whole, and the root of the problem appears to be twofold: first, in the dynamic environment of the Information Age, *anything you do* is likely to seem to take too long, cost too much and not meet expectations; but second and more insidious, if your perception is that there is some kind of magic in the technology, *it is a guarantee* that it is going to take too long, cost too much and not meet expectations!

Clearly, my bias is that there is no magic. There is only creativity, ingenuity and a lot of hard work, both on the part of the client as well as on the part of I/S. We are in this Information Age game together and it is imperative that we work together if we expect the enterprise to emerge safely into the mainstream of the Information Age environment.

That is why *Data Sharing Using a Common Data Architecture* is such an important book. Michael Brackett is "telling it like it is." There is a lot of data out there in virtually every enterprise that is in sorry shape. It is not accurate, it is not consistent, it is not timely, it is not integrated, it is not accessible, it is expensive, and it is misleading to management. If it is ever going to be of any use to anyone outside of its immediate users, somebody is going to have to sit down and examine it element by element. Each element is going to have to be understood in terms of what it means and then be uniquely identified, named, defined, and related appropriately, Hard decisions and choices will have to be made and consensus reached between potential users. Incidentally, I particularly appreciated Michael's wisdom that "resolution of issues should not be forced." It takes time to reach consensus, and forcing resolution does not result in quality models. Also, "an integrated architecture cannot be build with automatic documentation and reengineering tools because there is a discovery process involved." That is, no tool is going to be able to do the work of analyzing all of the existing data elements

and figuring out what the original programmers intended them to mean when they created them in the first place.

One might argue that it is not worth the time and money to invest the effort in the enterprise's existing data, but to focus only on new systems and new data. I would suggest that I have not yet encountered any enterprise that is ready to categorically dismiss its current investment in its existing systems. Even if the existing systems could be completely ignored, data in any new systems would ultimately have to receive the same element by element specification and consensus if its value is to be realized beyond merely a single implementation.

There is a good precedence in the world of linguistic studies for matters of this nature. Every culture that has been able to advance its body of knowledge and succeed in sophisticated undertakings, especially in the highly technical environment we associate with modern civilization, has done so on the basis of a formalized, well-documented language. For languages there are dictionaries, directories, thesauruses, indexes, glossaries, lexicons, concordances, and cross-references. The same concepts are relevant within the language (data) of an enterprise if it is to employ information technology and enterprise knowledge to its own advantage in the dynamics and vagaries of the Information Age. This may sound tedious, but nonetheless, it is a prerequisite. The book illustrates how to prepare each of these reference works (that is, the dictionaries, directories, thesauruses, indexes, glossaries, lexicons, concordances, cross-references, and so on) in support of the management and preservation of value for the data of the enterprise.

Data Sharing Using a Common Data Architecture is an exhaustive reference, defining every data concept I have ever encountered plus concepts I have never even thought of that are relevant to the management of data. For example, Michael points out that "data accuracy is a new category of data quality in the architectural infrastructure," and includes things like precision, scale, resolution, level of detail, frequency of collection, volatility, method of capture, degree of reproducibility, confidence, and so on. The book is not only a comprehensive source for subjects related to data, but it also contains concise definitions for those subjects.

It is clear that Michael Brackett has had an enormous amount of experience in doing the actual work of architecting data, not only because of the exhaustive definitions or the explicit instructions on how to handle every anomaly, but also because of the richness and

diversity of his illustrations and his pragmatic methodological suggestions.

For anyone who is merely involved in a reverse engineering project or a data warehouse project (that is, even something far short of an enterprise-wide, common data architecture strategy), *Data Sharing Using a Common Data Architecture* will be an invaluable aid.

I believe this book will stand as a foundation work. It is very detailed and explicit, yet it is very readable. It is very technical, yet it is intelligible to the non-technical reader. It suggests approaches that require nothing more than a word-processor and a data base management system to implement. No special tools are required to employ the concepts and address/solve the enterprise data problem. It would be nice if some better tools were available and I am sure that it is only a matter of time, but in fact, there is no magic in the tools in themselves. The tools merely support the people who have to do the actual work of defining, naming, identifying, and relating.

You might not agree with every single idea or technique or definition or even the methodological approach Michael takes. Still, *Data Sharing Using a Common Data Architecture* is a valuable reference where it comes to managing data.

I think we tend to become overwhelmed by the magnitude and complexity of the "data problem," particularly in enterprises of any size and any sizable inventory of existing systems. However, it only takes time and costs money. If you don't start, you will never finish. Clever ideas and tools will come and go. There still will be a lot of data out there that is not clearly understood and therefore whose value is not at all leveragable. But any enterprise that wants to exist over the next few years and needs to retain any of its current knowledge-base is going to have to buckle down and do this data work sooner or later. Once again, I am sure it will be argued that you need only to "build new" and ignore what is already out there—and it might work. In this case, one would only have to define the "common data architecture" and go from there. However, the practicality of the matter is it will be hard to get around the substantial investment already made in the current systems—plus, the enterprise will have to continue to operate in some fashion for some period of time during the transition to the new environment. If this is true, then it only stands to reason that someone is going to have to do the work of mapping the existing data elements to the "common data architecture."

It appears to me that this is a universal issue and sooner or later, everyone is going to have to get back to the basics and "(block and) tackle" this formidable data problem. But Michael says, "just when it appears that the job is insurmountable, the critical mass of information is achieved and disparate data fall into place within the common data architecture." That sounds like some pretty good news to me.

I think you will find this book valuable, informative, readable, and an excellent reference.

LaCanada, California
May 20, 1993

▼ ▼ ▼ ▼ ACKNOWLEDGMENTS

The author thanks Bruce Jorgensen and Tim Feetham for their ideas and comments about how the current environment can be improved. The author thanks Joe Oates and Tom Haughey for their input about the implementation of new technologies to solve our current problems. The author thanks Gary Guinotte for his review of the book and the publishers for their reviews and comments about the content and the method of presentation. Their input helped make this book a polished product.

The author thanks the students in his classes, the conference attendees, and data refining project team members who provided input about the current problems they face and how the new techniques described in this book help solve their problems. Their input helped make the techniques people-oriented and client-friendly. The author also thanks Denise Prowse for her comments about the techniques and features of the Data Resource Guide, and Barb Malen for her help in developing the Data Resource Guide and making it a useful product.

Special thanks go to Carey Clark and Tom Crawford for the time spent discussing different approaches for controlling disparate data and managing an organization's data resource, and for their reviews of the book. Without their thoughts and comments, many of the concepts and techniques presented in this book would not be available.

Finally, with deep respect, the author thanks Mr. John Zachman for his intuitive insight into understanding and resolving many of the problems we face today. Through several long discussions with Mr. Zachman, the author has gained some of that insight and applied it to understanding and resolving the disparate data situation.

▼ PREFACE
▼
▼

Wouldn't it be a pleasure to know and understand all the data in your organization? Wouldn't it be great to easily identify and readily share those data to develop information that supports business strategies? Wouldn't it be wonderful to have a formal data resource that provides just-in-time data for developing just-in-time information to support just-in-time decision making?

That's what this book is about. It defines a common data architecture, its contents, and its uses. It describes how a common data architecture is developed for defining disparate data in a common context and sharing those data across projects within an organization and across organizations. It explains how a common data architecture is used to achieve short-term, incrementally cost-effective benefits with minimum impact on business operations, while achieving long-term goals.

Let's face reality! The data in most organizations are disparate data. They are poorly named, defined, structured, and documented. They are highly redundant, and redundant versions are often inconsistent. They are highly variable in their format, content, and meaning. They cannot be readily identified, easily accessed, or properly used. Large quantities of useful data already exist, and people are not aware of these hidden data or their value. The reality for most organizations is a large quantity of disparate data.

Organizations are basing their existence on these disparate data. They have a rapidly increasing demand for information, often on short notice. This information is often critical to their success in a dynamic business world. However, the information cannot be prepared quickly from the existing disparate data. The conflict between existing disparate data and the high demand for information are creating a crisis in many organizations.

The crisis can be resolved by developing a common data architecture for understanding and sharing disparate data. The vision is a formal data resource where all data are defined in a common context. The formal data resource evolves to a mature data resource that has less variability and all the data necessary to support rapid development of critical information for a dynamic organization. The common data architecture provides a base for achieving the vision. It helps people build incrementally toward the vision. It gives people the courage to face the reality of today's situation and stop the trend of continuing to develop disparate data.

Developing a common data architecture is relatively easy for new data at the project level. However, it is more difficult to develop a data architecture for new data that is consistent across several large projects, across an organization, or across several organizations. It is extremely difficult to develop a data architecture that integrates disparate data within an organization or across organizations. Many of the techniques and tools that work well for new data at the project level do not work for disparate data.

A common data architecture helps people understand and share disparate data within an organization, and across organizations where there is no point of common authority. It helps people manage data in purchased applications. It helps people manage multiple, conflicting mandates from other organizations. It increases productivity, because real productivity comes from understanding and reusing existing data, not creating new data. It improves data quality so that data directly support business activities. It provides visibility and credibility for both clients and executives.

If this sounds like the approach you need—read on!

There are no global, one-shot, overnight cure-alls for the current disparate data situation. Many people believe that tools can build an architecture, align existing data to that architecture, and clean up the current situation. This is not true! Tools don't build architectures, and tools will not solve the problems with existing data. People build architectures, and people align existing data with architectures. Only people can clean up disparate data. Tools only support the efforts of people, just like a carpenter's toolbox supports the carpenter. The knowledge and skills are in the people, not in the tools.

Building a common data architecture does not require *changing* existing applications or databases. It does not require changing existing data names or formats. It does not prevent the acquisition or contin-

ued use of purchased applications. It does provide a way to define all existing and new data in a common context so they can be easily understood and readily shared.

This book presents a practical, real-world approach for developing and using a common data architecture to define all data in a common context. It paints the vision of a formal data resource library and provides techniques for building incrementally toward that vision. It provides simple concepts, terms that are easy to understand, and techniques that are easy to follow. It explains pitfalls that may be met and how they can be avoided. It emphasizes people and technology equally, because it is the people who make or break any technology. It unfolds an new way of thinking about data that breaks the continuing tradition of developing disparate data.

This book is not about designing and modeling data or constructing databases. It is not about traditional data resource management or data administration. It is not about the use of CASE tools or information engineering techniques. It is about understanding existing data within the context of a common data architecture. It is about using data engineering techniques to understand and share data, and to develop a data resource that supports information engineering. This book builds on a previous book by the author, titled *Practical Data Design*. The concepts and techniques for modeling and designing data are explained in detail in that book

Chapter 1 describes the current disparate data situation, an engineering perspective to solving that situation, and a vision for sharing data. Chapter 2 explains the concepts of a common data architecture and how it fits into the information technology infrastructure. Chapters 3, 4, 5, 6, and 7 explain the individual components of the common data architecture: data naming, data definition, data structure, data quality, and data documentation.

Chapter 8 describes the concepts and techniques for refining data to a common data architecture. Chapters 9, 10, and 11 describe the description, structure, and quality of disparate data. Chapter 12 describes techniques for cross referencing disparate data to the common data architecture. Chapter 13 describes techniques for identifying official data variations and developing data translation schemes. Chapter 14 describes the entire process for identifying disparate data, developing a formal data resource, and evolving to a mature data resource.

Chapter 15 explains the concepts and techniques for ensuring that the formal data resource is complete and contains all the data necessary to support information engineering. Chapter 16 explains how the data sharing reality is achieved. It emphasizes the importance of people and the creation of a win–win situation. An extensive Glossary is provided. Appendices provide a data lexicon that supports the data naming taxonomy, data names and descriptions, data resource survey questions, data cross reference criteria, and data structure criteria.

This is not a book about the future. It's about today and getting to the future. It's about people surviving their disparate data situations. It's about people sharing the problems they face with data and sharing the resolution of those problems. Most people want to change the current situation; they just don't know how to change without adversely affecting the business. They are wary of past cure-alls and vendor hype. They are unsure about how to approach the uncertainty and variability of disparate data to gain control of their data

There is a window of opportunity for organizations to gain control of their data. The rate at which an organization gains control of its data is its own choice. However, control must be gained before open systems, decentralization, and further fragmentation close that window. Control can be gained by developing a common data architecture that provides a central architecture to support decentralized use and deployment information systems. An organization must step ahead to gain control of its data. There is a risk and cost for stepping ahead, but the risk and cost are far greater if nothing is done and disparate data proliferate.

Is it possible to gain control of your disparate data? The answer is a resounding *Yes!* The common data architecture provides an approach that ensures a win–win situation for clients, executives, and the organization.

Olympia, Washington
August, 1993

CONTENTS