



---

## Why a Data Model is Important to the Business

*This white paper is an adaptation of Chapter Two of the book [Data Modeling for the Business: A Handbook for Aligning the Business with IT Using High Level Data Models](#), by Steve Hoberman, Donna Burbank, and Chris Bradley.*

## Why Does a High-Level Data Model Matter?

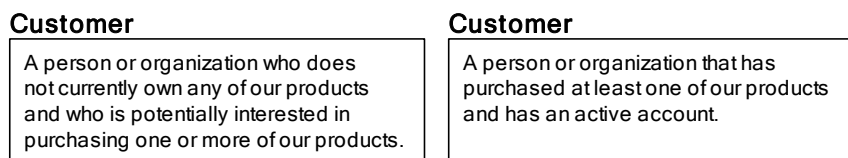
---

A high-level data model conveys the core concepts and/or principles of an organization in a simple way, using concise descriptions. The advantage of developing the high-level model is that it facilitates arriving at common terminology and definitions of the concepts and principles.

Take a concept as simple as ‘customer’. Everyone knows what a customer is, right? Or do they? The definition may change based on a person’s perspective. To the billing department, a customer may be someone who owns a product or service sold by the company and to whom an invoice is sent. To a salesperson, a customer is someone who has not yet bought a product but to whom they hope to make a sale. And there are more things that we need to clarify... Does a customer have to be a person or can another company be considered a customer? Is someone who has purchased a product from us in the past, but does not have an active account, service agreement, or support contract still a customer? Is there a difference between an active and an inactive customer or an existing customer and a prospective customer? Let’s take this last example and build a simple high-level data model to help explain these differences, shown in Figure 1.

A high-level data model uses simple graphical images to describe core concepts and principles of an organization and what they mean.

Figure 1 – High-Level Data Model Showing Two Different Definitions of Customer



“That’s a high-level data model?” you may ask. “It’s just a couple of boxes with text in them!” Yup! The key to a high-level data model is simplicity. An important goal of this model is to align on common terminology, business definitions and rules. The diagram can be as simple as a set of boxes with text in them, as in Figure 1. In future examples, we’ll expand upon this simple model, adding more boxes and a few lines. But remember, a high-level data model should always be simple and clear enough that a non-technical, ‘everyday’ person can understand it.

***“Isn’t this too high-level to be useful?”***

No, the purpose of the high-level data model is describing complex information in a simple way, using a concise description, not a vague one.  $E=mc^2$  is a simple equation that communicates a complex idea to its intended audience using concise notation.

A high-level data model should focus on a particular area that’s important to the audience or user, so that the content and scope is limited to the matter at hand.

*Everything should be made as simple as possible, but not simpler.*

- Albert Einstein

In fact, a high-level data model doesn’t even need to look like a traditional data model or be shown as a ‘picture’ at all. We could have placed the same information in a table or spreadsheet and come up with something like Figure 2.

Figure 2 – High-Level Data Model Shown in Table Format

Customer	A person or organization who does not currently own any of our products and who is potentially interested in purchasing one or more of our products.
Customer	A person or organization that has purchased at least one of our products and has an active account.

Both the diagram and the table format presentation highlight the issue of two different definitions of customer. A data modeling purist would correctly point out that you can’t have two different definitions of customer on a ‘valid’ model.

While strict rules most definitely need to be followed on more detailed data models, the purpose of a high-level data model is communication and gaining consensus on core concepts so that the detailed diagrams created later are based on correct assumptions. It’s okay to bend data modeling rules as long as the focus remains on aligning terminology, definitions and business rules. Looking back at the architecture example from the previous chapter, if my architect started developing detailed floor plans and wiring diagrams for a three-story mansion when what I was looking for was a simple cabin, the end result wouldn’t meet my needs no matter how good the design was.

High-level data models are the result of an iterative process. It is rare that the participants involved in creating the high-level data model will all agree on definitions when they are initially documented, but a first draft can be used to highlight differences of opinion so that discussions for achieving consensus can begin. Organizations and individuals may not even realize that there are different definitions until they are documented in this way. One technique for reaching consensus among the participants is to identify

## Why a Data Model is Important to the Business

the various audiences, business areas, projects, applications, etc. that use each particular term. The meaning of a term can change based on its context, so it is critical to understand the context in which each term appears before consensus can be reached.

Let's go back to the credit card example in the introduction where you keep receiving offers for a new credit card even though you already have one. We'll perform our own mini data architecture project for the bank. Let's start by coming up with a list of all of the departments that access customer information. Our 'model' will be a simple list, shown in Figure 3.

Figure 3 – A Very High-Level Data Model Showing Stakeholders

Marketing
Consumer Credit
Banking

Identifying what data is used and by whom takes a lot of effort. It's much easier to focus on your own project or department, or at least it seems that way on the surface. But this siloed approach may lead to systems that don't work well together. By involving other groups, it's possible to leverage work that has already been done without reinventing the wheel.

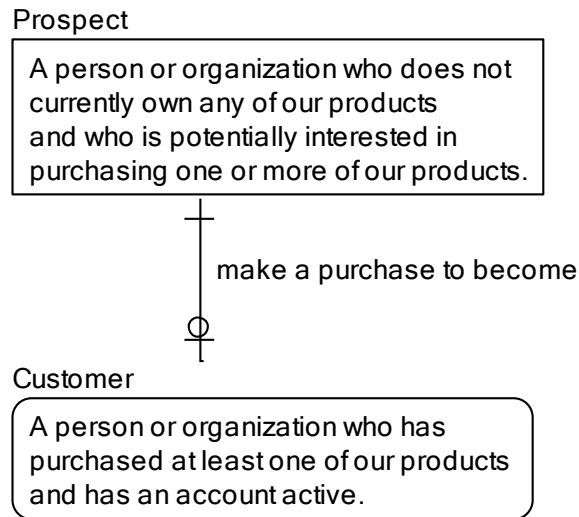
Once the stakeholders are identified (people, groups or organizations that can affect or be affected by an action or policy), it's time to get them to talk to each other. In this case, after identifying and meeting with each of the organizations, the bank came to the conclusion that while the banking and consumer credit departments considered a 'customer' to be a person who had an existing account with the bank, marketing also used the term 'customer' to describe people who did not have an account with the bank.

These two definitions don't really conflict with each other; they represent different stages of the organization's relationship with the party. There are really two types of 'customers' – prospective customers, who have not yet purchased a good or service, and customers who have purchased a good or service and have an active account. We can change the model to reflect the two different meanings using the new terms 'Prospect' and 'Customer'.

If you think back to the blueprint analogy from Chapter 1, the floor plan of the house not only helped me define what I was looking for, but it showed me how the rooms fit together—the relationships between them. High-level data models show how business concepts relate to each other. In fact, showing relationships, which represent business rules between concepts, is equally as important as creating good definitions. For the bank, we'll want to add a relationship that shows that a prospect is related to a customer in that once a prospect purchases a product, they become a customer. We use lines to show these relationships in a data model.

Figure 4 shows a simple high-level data model displaying both the definitions of the concepts and their relationships to each other.

Figure 4 – Data Model Clarifying the Definition of Customer



While this may be intellectually interesting, you may still be wondering what it has to with you getting ads from the bank for something you already have. As you just saw, when the bank developed their high-level data model, 'Customer' split into 'Prospect' and 'Customer', but this took place after their systems were already in operation. So when an employee in the marketing department for consumer credit logged into the bank's sales force web application and asked it to mail advertising about a credit card with a special interest rate to 'customers', the sales force web application then queried a database for a list of 'customers'. When building the database, the IT department was unaware that there were different meanings of the term customer, so both prospects and customers were included and were indistinguishable from each other. The sales force web application extracted a list from the customer database containing both customers and prospects. So you, the innocent customer, received a mailing that really should have gone to prospects.

## Integration

---

Now that you understand why you're receiving erroneous marketing ads and have contacted the bank offering your data architecture services to solve their problem at a discounted rate, you're feeling confident and happy. But the next day you go to your mailbox and shake your head in frustration. Since you have both a checking account and a credit card account with this bank, you'd assume that you would get all of your records on a single statement. But no! You receive a bank statement addressed to 'Jane W. Doe', and a separate credit card statement addressed to 'Jane Doe'. How can they not know you're the same person?

Let's go back to the list of business areas we put together in the last section. We used that list to contact each organization to understand how each defined customer. From that we determined that the marketing department's information should really be stored and/or accessed differently from that of consumer credit and banking. Creating this list is a fairly simple task for our small example, but for a large organization, it can be a daunting one. It may take months or even years to get a true assessment of what information a company has and which departments or applications are using it. If it is difficult to

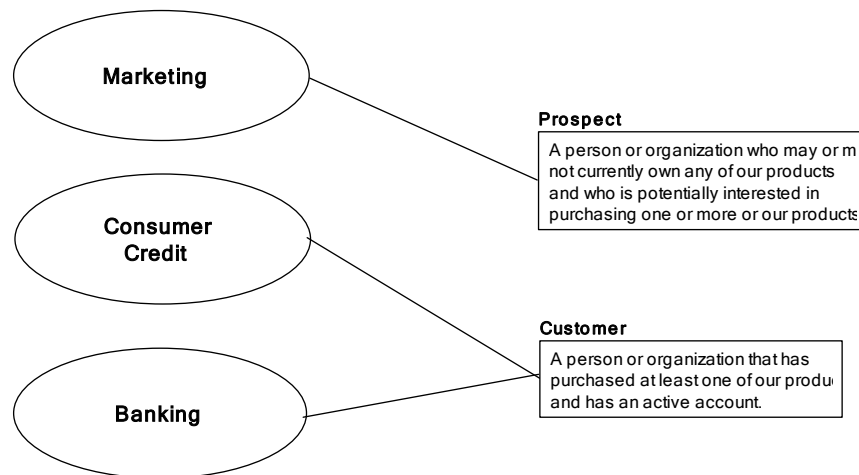
## Why a Data Model is Important to the Business

merely document what information exists, imagine how much harder it is to make sure the information is accurate, consistent, and stored the same way.

In many companies, each department stores its information in its own database, cut off from other groups in the organization. Common terminology, definitions, and rules across departments and projects can break through these silos, enabling the organization to operate as a single, powerful unit. You may have heard the terms Master Data Management, Customer Data Integration, Enterprise Architecture, and Data Warehousing. All of these are initiatives that attack the lack of integration from different angles, and all of them require an accurate data model to be successful. See Chapter 7 for more information on how a high-level data model can aid these initiatives.

For our simple example, let's take the list of departments and map which departments use which customer information, shown in Figure 5.

Figure 5 – Mapping Business Areas to Common Terms



This is the first step towards developing a single statement with both your banking and credit card information. Now, at least, the departments can see that they are using the same data. They have an opportunity to share it and take advantage of new data created by the other department. Since the Banking and Consumer Credit departments know that they are both using the same information, they can just combine their data in a single place to get you an integrated statement, right? Ah, if only it were that simple!

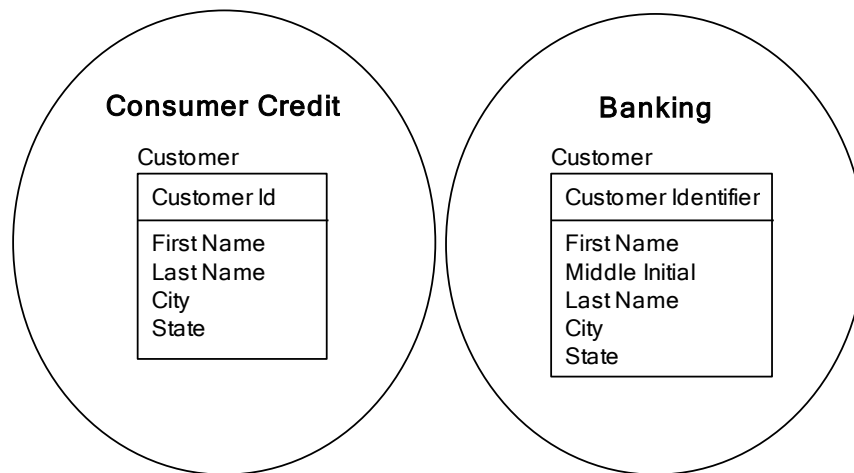
You'll recall from our earlier discussion that the purpose of a data model is to not only document the definitions for and context around information, but also to document the actual physical structure of the databases in which the information is stored. The same information can be stored with many different names, formats, and software platforms, so after we have documented how the data *should* be, we still need to map how it is *today*. This leads us to a discussion of the importance of standards and reuse.

## Standards and Reuse

---

Now that we know that two departments are using the same definition of customer, can we ensure that the information they use is truly the same? In Figure 6, we see the different types of information about customers that are stored by each department. A good high-level data model should be basically understandable even if you don't recognize the notation used. Just like the floor plan of your house, it should provide a clear overview that's easily understood. So for now, ignore the notation and just look at the information that we're tracking about customers in the two business areas: Consumer Credit and Banking.

Figure 6 – Differences in the Customer Information Stored in Different Departments



You'll see that the information is basically the same—the customer's name, location, and an identifier for the customer. But you'll also notice there are subtle differences. Consumer Credit uses the term 'Customer Id' while Banking spells it out as 'Customer Identifier'. In our first example, we found that differences in the meaning of the word customer could have negative effects. In this example, the meaning of the customer identifier field is exactly the same, but different terms are used to describe it. A human could easily determine that 'Customer Id' and 'Customer Identifier' are the same thing (although we should always verify this to be sure), but a computer cannot. Thus, to truly integrate these systems, we would have to use common naming standards to make sure that we can match information correctly. Again, this is an easy enough task in the simple example shown in Figure 6, but can take months or years in a large organization with thousands or even millions of different pieces of information to keep track of.

You'll also see that Banking stores a middle initial for a name, but Consumer Credit does not. This might cause a problem when matching or reporting data. To truly integrate these two departments, we need to identify:

- Common meaning of terms
- Which groups, applications or organizations are using the information and for what purpose
- Standard naming conventions for terms with the same meaning
- Standard design structures for the information

Once we've tackled all of the above, the truly difficult part begins. To this point, we've only aligned on the common meaning, purpose, and design of our information. We haven't begun to address the actual data content or the way it is physically stored. That challenge, which is beyond the scope of this book, is akin to changing the wings on an airplane as it's flying across the Atlantic. The database systems that need to be changed are in use by the mission-critical applications that are running your business; and as we've seen, they are used by many areas across the organization. Before any data is changed, the effect that this change would have on existing systems must be carefully assessed.

So you might be waiting a while before you are able to get your single bank and credit card statement, but at least we were successful in stopping the marketing department from continuing to send you unwanted mail!

### Data Modeling for All

---

As you've seen, arriving at common goals, scope, and context and creating a common understanding of core concepts of the business across functional areas is critical to a successful data management program. They are the reasons we build a high-level data model. It is important that the IT systems that manage the data in your organization are based upon *your* understanding of information, and the business rules that define *your* organization. While business people do not need to be involved in the physical design of database systems, they do need to be involved in the high-level data models that define these database systems. It is the goal of this book to make the process of building a high-level data model an achievable goal for the average business user.