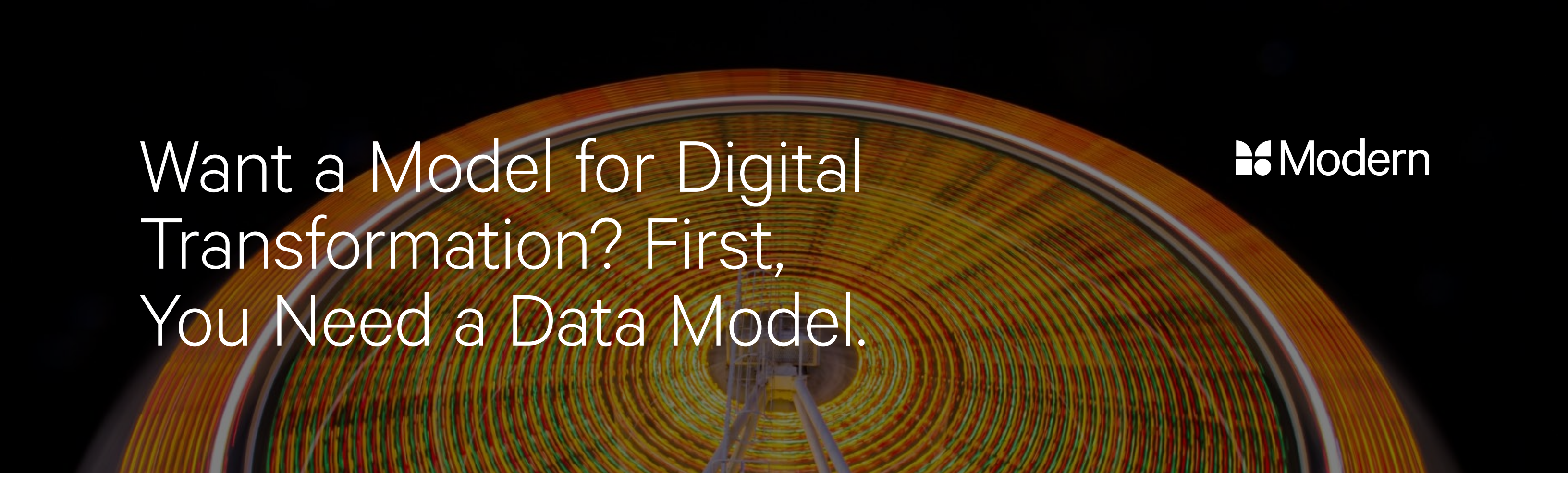

Want a Model for Digital Transformation? First, You Need a Data Model.

Modern

A data model is a visualization of data elements and the relationships between them. Such models should be dynamic, evolving with the needs of the organization. They act as a blueprint where data structure and business needs meet. A well-designed data model will:

- Provide more clarity into your data
- Reduce costs of development and maintenance
- Improve the creation and maintenance of contracts
- Act as documentation for all business roles
- Make data more consistent across the organization
- Give foresight to continually meet business needs

Creating a data model that works for your organization is a collaborative effort. It requires a cross-functional team to get the right data in the right place at the right time. A good data model also ensures governance and compliance, making it easier and safer for data end-users to access and utilize the information they need.

Think you need to rip and replace your data architecture before building a data model? Think again. A good data model will help create the highest functioning system on top of a legacy database. The next step for your organization's digital transformation may be closer than you think — it simply takes a modern approach.

Start with business objectives

Before you start working on constructing a data model, the first step is to involve the data end-users. Business stakeholders and other information users should be involved from the beginning to discuss their needs and plans for using the data. During these conversations, a conceptual data model should be used to fuel discussion and inspire use cases from the data end-users.

Taking this first step will ensure that the model will actually make the needed data accessible. Aside from accuracy, what makes data "good" or "bad" is determined by its quality of usability. That usability starts with knowing who and how the data needs to be used. Knowing who needs which data will also inform decisions about security and compliance, helping inform the next step: the logical model.

The logical data model step is less theoretical and more concrete. Rather than just mapping out the data entities broadly, the logical data model defines the structure of these entities, the relationships between them, and their attributes. This stage leverages data modeling notation systems and is typically used for data warehouse design or reporting system development projects.

The last step is consolidation into a non-abstract physical data model. This describes how data will be physically stored within your database. It includes the relationships among entities, primary keys, foreign keys, partitions, CPUs, tablespaces, and similar detailed information. The finished product is a finalized design that can be translated directly into a relational database.

Common pitfalls to avoid

Working with the data end-users should help inform the correct levels of granularity when it comes to the conceptual data model stage. It's important to strike a balance between gathering enough data without inundating the system with loads of data that won't be used. This is especially important when working with legacy systems that could be slowed down by an overload of data. From a business perspective, ask what level of data is needed to answer the necessary analytical questions, power marketing personalization, customer service enablement, and empower sales teams.

It may be difficult to track down all the data sources needed to answer business questions. Now more than ever, data is stored in a multitude of locations, including in Excel spreadsheets and memos. This raises the potential for many data silos in the company. Technical marketers may create local databases, or a sales manager may keep a list of leads in a personal memo. If your data model doesn't have the flexibility to include these sources, you might be limiting the scope of your analyses or machine learning models.

Dimensional hierarchies are another important and often overlooked piece of building a data model. Improperly built hierarchies result in the inability to delve deeper into data and force analysts to write overly complex queries. Especially when using a legacy system, it's important to optimize dimensional hierarchies according to how data will be used to ensure speed, deep insights, and accuracy.

Calculated fields can also be a common pitfall. If you calculate them on your existing infrastructure, especially if it is a legacy infrastructure, you run the risk of slowing down all of the processes or even potentially breaking pipelines. A well-designed data model will take derived values into consideration, making for more agile processes and better analysis.

To translate the physical model stage within a data source, many organizations still rely on ETL (extract, transform, and load) models to move the data into a cloud data warehouse. This can be a costly and error-prone process, with many "data loader" tools copying huge amounts of data from sources into your cloud data warehouse.

Because ETL relies on data being duplicated directly from the source into your cloud data warehouse, much of the data will likely be superfluous or obscure — which makes it difficult for the analytics team to interpret. This can also make the transformation step challenging as well, since the data should be anonymized and masked at this step. It is an essential step for data privacy, but given that much of the data may be disorganized, duplicated, and/or incorrectly labeled, it creates a potential vulnerability.

A modern approach

Luckily, there is a modern approach that avoids pitfalls and makes it easy to build a strong data model and ultimately transform your data ecosystem. It starts with visibility into the data at the source.

Novel tools like Modern's DataOS give you full transparency into all of your available data and provide metadata about the data—before it's moved. This metadata can be used to form a comprehensive picture of what is available to inform conversations with the data end-users. Then, only data needed for business outcomes can be moved into your cloud data warehouse, without unnecessary duplication.

Metadata about data elements, their attributes, and relationships to one another can also inform dimensional hierarchies. A well-built data ecosystem will have a visualization of this information, these relationships, and important tags. DataOS generates a visual view of this data, as well as functionality to search the data and metadata. Additionally, a data model view for business users makes conversations about data structure and use simpler and more effective.

Since all available data can be discovered and tagged at the source, privacy and compliance become simpler. Having this level of visibility enables the identification of sensitive data before it's moved, allowing comprehensive masking or anonymizing before the data is loaded. DataOS also includes native governance, which makes it easier to expose only the appropriate level of data for the data end-user while still meeting their needs. This can unlock more elegant structuring of data without needing to hide data in silos.

Creating a metrics and entity layer prevents data integrity and consistency issues. Derived values can be computed as part a "model, map, and load" (MML) process as data is loaded and stored in your cloud data warehouse. The metrics and entity layer offer a place for these calculations to live that creates organization-wide consistency and better analyses. Modern tools like DataOS have built-in metrics and entity layers that can ensure the entire business is working off the same definitions in your data.

Think it's time to build a transformative data model? Think Modern.

BY C. BOSTIAN

TheModern DataCompany

Want a Model for Digital Transformation? First, You Need a Data Model.

The Modern Data Company
306 Cambridge Ave
Palo Alto, CA 94306
TheModernDataCompany.com
info@TMDC.io