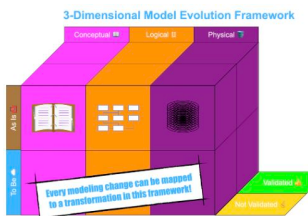
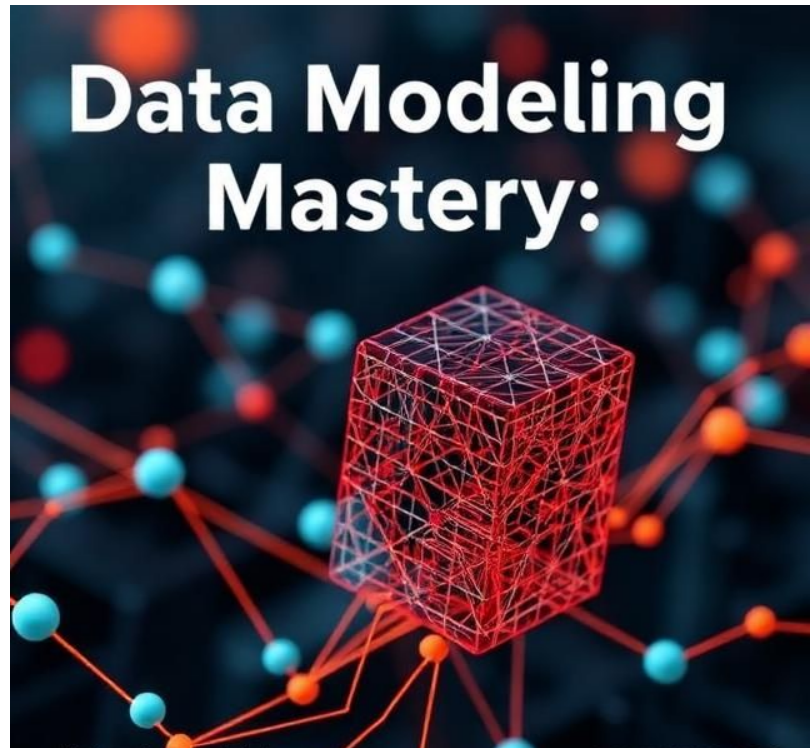


Data Modeling Mastery: Structuring Data for AI, Analytics, or Anything Else



peter.aiken@anythingawesome.com +1.804.382.6957



© Copyright 2026 by Peter Aiken Slide # 1

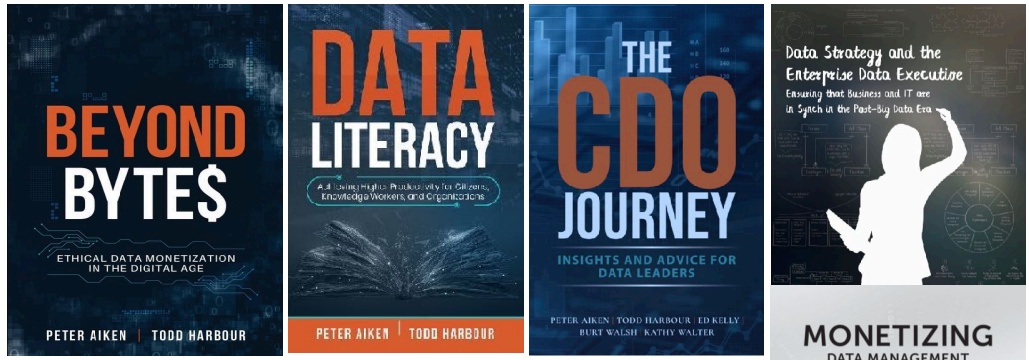
Peter Aiken, Ph.D.

- I've been doing this a long time
- My work is recognized as useful
- Associate Professor of IS (vcu.edu)
- Institute for Defense Analyses (ida.org)
- DAMA International (dama.org)
 - IDEAL Affinity Group
 - GenAI in Data Affinity Group
- Anything Awesome (anythingawesome.com)
- Experienced w/ 500+ data management practices worldwide
- 13 books and dozens of articles
- Multi-year immersions
 - US DoD (DISA/Army/Marines/DLA)
 - Nokia
 - Deutsche Bank
 - Wells Fargo
 - Walmart
 - HUD ...

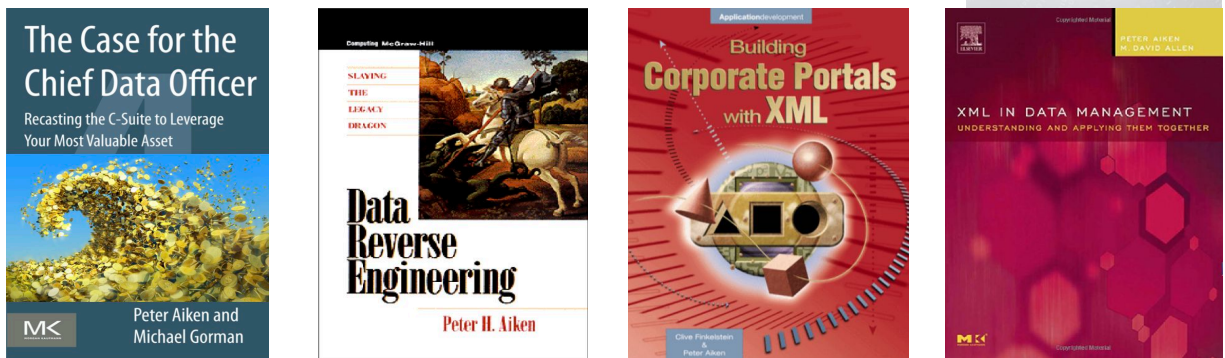
\$1,500,000,000.00 USD



Event Pricing on Peter's Books



- 25% off directly from the publisher on some titles
- My 'Book Store' @ <https://anythingawesome.com/books>
- Enter the code "TP25" at the Technics bookstore checkout where it says to "Apply Coupon"



Program Overview

- Why model anything?
- What is data modeling designed to accomplish?
- Why (and how) should I look to AI for help?
- Overview of modeling types
- Conceptual Data Modeling
 - Motivation: Architectural tradeoffs
 - Strategy and conceptual data modeling
 - Glossary/Dictionary capabilities
- Logical Data Modeling
 - Motivation: Simplicity (Operational and Design)
 - Motivation towards standards
 - Business meets strategy
- Physical Data Modeling
 - Motivation: Required documentation and/or facts
 - Become the blueprints for physical construction of the solution
 - Blueprints are used for future maintenance of the solution
- Tying it all together/Take Aways/References/Q&A

Data Modeling Mastery: Structuring Data for AI, Analytics, or Anything Else

The Big Short



The Architecture of a Collapse

- Objective: To move students from "Data Processing" to "Information Integrity" by analyzing the 2008 financial crisis through the IE lens.

I. The "Strategic Schema" of a Lie

- We begin by identifying the Business Entities. In IE, if your entities are poorly defined, your entire enterprise model is a hallucination.
- **The Myth of the "Mortgage" Entity:** How a single data point (a home loan) was abstracted into a "Bonds" entity, stripping away the critical attributes of risk and reality.
- **Data Quality vs. Data Volume:** The banks had petabytes of data, but zero Information. * The "Liar's Loan": What happens to a system when the primary data entry point (the loan application) has a 0% integrity constraint?

II. Breaking the "Black Box"

- This is the "Information Analysis" phase. We look at the CDO (Collateralized Debt Obligation).
- **The CDO as a Nested Set:** Explain how bad data was "repackaged" to change its metadata (from 'Subprime' to 'AAA').
- **Synthetic CDOs:** The ultimate data derivative. Discuss the concept of a "bet on a bet"—essentially data pointers pointing to other pointers with no underlying physical asset.

III. The Signal in the Noise

- Why did Burry, Baum, and Eisman see what the "Experts" missed? They performed Data Normalization on the real world.
- **Direct Observation:** Scrapping the top-down reports and going to the "Primary Key"—the actual homeowners in Florida.
- **The Correlation Flaw:** The models assumed home prices were independent variables. We will discuss the catastrophic failure of the Gaussian Copula Function—a math error that cost trillions.

IV. The Ethics of Information Engineering (10 Minutes)

- **The "Rating Agency" Conflict:** When the data auditors are paid by the data creators, the Metadata is corrupted at the source.
- **Transparency vs. Obfuscation:** How complexity is used as a weapon to prevent stakeholders from performing their own data discovery.

V. Interactive "Stress Test" (10 Minutes)

- **The "What-If" Challenge:** Ask students: If you were the Lead Data Architect at Moody's in 2005, which Attribute would you have monitored to trigger a "System Red" alert?
- **The Modern Parallel:** Briefly touch on where we see these "Data Bubbles" today (e.g., AI-generated training data loops).

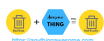


Executive Summary

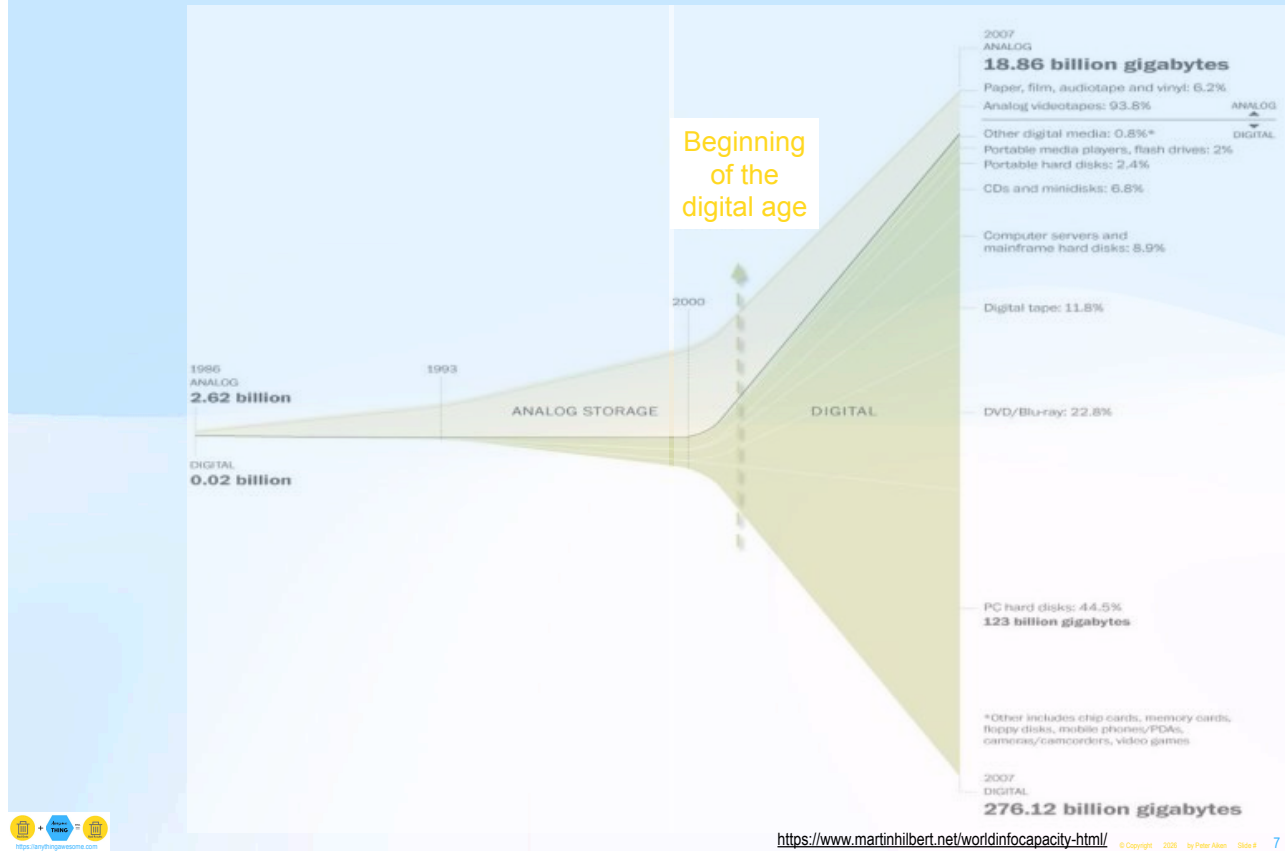
- "In the world of Information Engineering, a model is only as strong as its connection to the physical reality it represents. The Big Short is the ultimate cautionary tale of what happens when the Logical Model ignores the Physical Reality."



"In the world of Information Engineering, a model is only as strong as its connection to the physical reality it represents. The Big Short is the ultimate cautionary tale of what happens when the Logical Model ignores the Physical Reality."



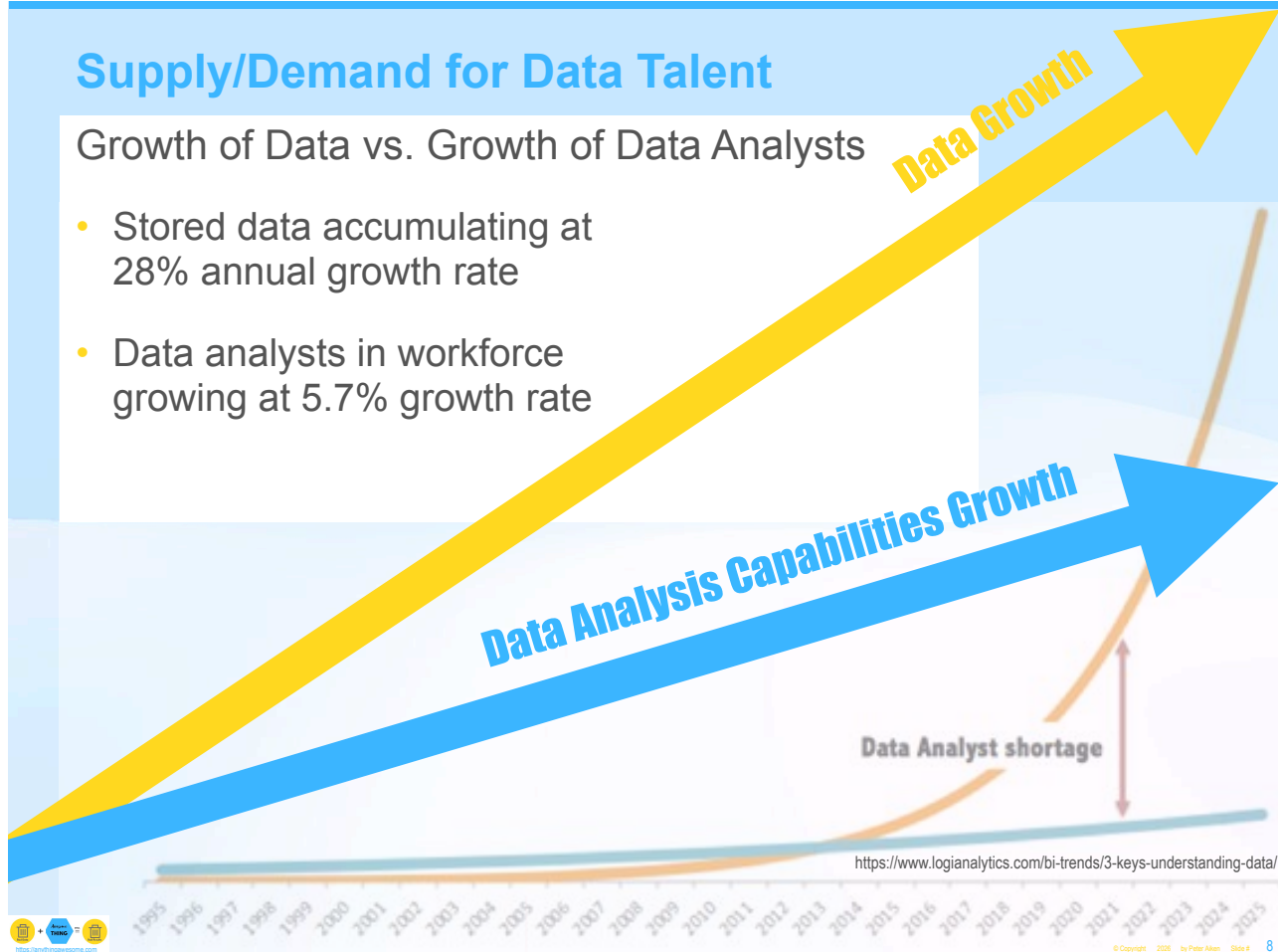
Global Information Storage Capacity

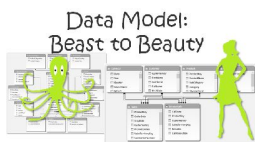


Supply/Demand for Data Talent

Growth of Data vs. Growth of Data Analysts

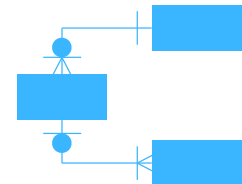
- Stored data accumulating at 28% annual growth rate
- Data analysts in workforce growing at 5.7% growth rate





powerpivotpro.com

Why Model?



<ul style="list-style-type: none"> • Would you build a house without an architecture sketch? 	<ul style="list-style-type: none"> • Model is the sketch of the system to be built in a project.
<ul style="list-style-type: none"> • Would you like to have an estimate how much your new house is going to cost? 	<ul style="list-style-type: none"> • Your model gives you a very good idea of how demanding the implementation work is going to be!
<ul style="list-style-type: none"> • If you hired a set of constructors from all over the world to build your house, would you like them to have a common language? 	<ul style="list-style-type: none"> • Model is the common language for the project team.
<ul style="list-style-type: none"> • Would you like to verify the proposals of the construction team before the work gets started? 	<ul style="list-style-type: none"> • Models can be reviewed before thousands of hours of implementation work will be done.
<ul style="list-style-type: none"> • If it was a great house, would you like to build something rather similar again, in another place? 	<ul style="list-style-type: none"> • It is possible to implement the system to various platforms using the same model.
<ul style="list-style-type: none"> • Would you drill into a wall of your house without a map of the plumbing and electric lines? 	<ul style="list-style-type: none"> • Models document the system built in a project. This makes life easier for the support and maintenance!



Use Models to

- Store and formalize information
- Filter out extraneous detail
- Define an essential set of information
- Understand complex system behavior
- Gain information from the process of developing and interacting with the model
- Evaluate various scenarios or other outcomes indicated by the model
- Monitor and predict system responses to changing environmental conditions



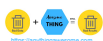
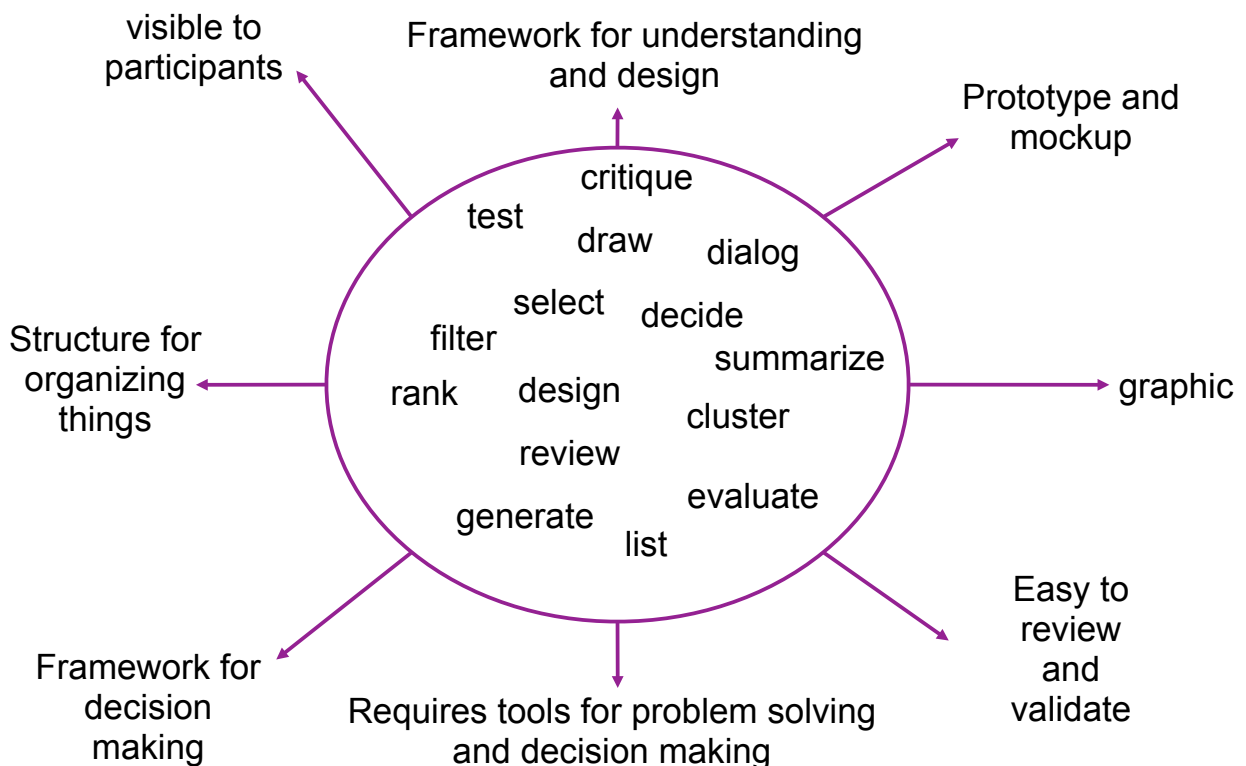
<https://www.youtube.com/watch?v=J5YR0uqPAI8>



- **Equations**
- **Simulations (including video games)**
- **Physical models**
- **Mental models**

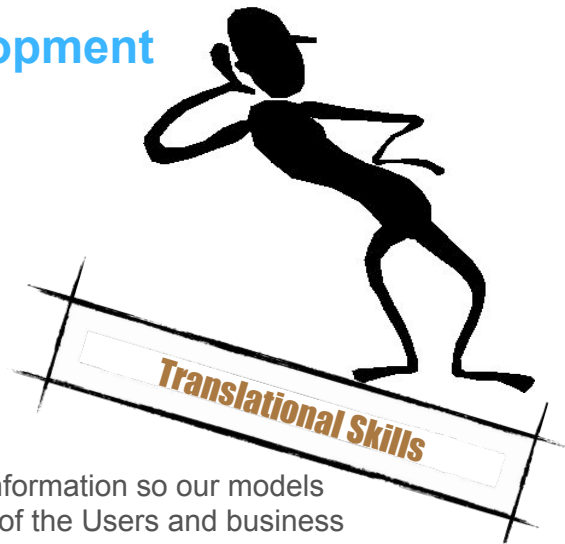


What a Model Is (Ellen Gottesdiener www.ebgconsulting.com)



Models in Systems Development

- Listening skills
- Notations, techniques, and sensitivity
- Modeling notations
 - To document what we learn
 - To communicate with the users
- Modeling techniques
 - To ensure we use the tools properly
- People sensitivity
 - Interviewing and listening skills
 - to ensure that we gather all relevant information so our models form a complete and accurate picture of the Users and business
 - Users get a new view of their job
- The information aspects of their operation are something totally new to most users
- As a serendipity modeling will give them a new view of their own world and can give them new insights often leading to improvements in the way they operate
- All modeling based methods force us to do more work on the earliest project stages
- We must first understand and define the problem before we we can design a correct solution

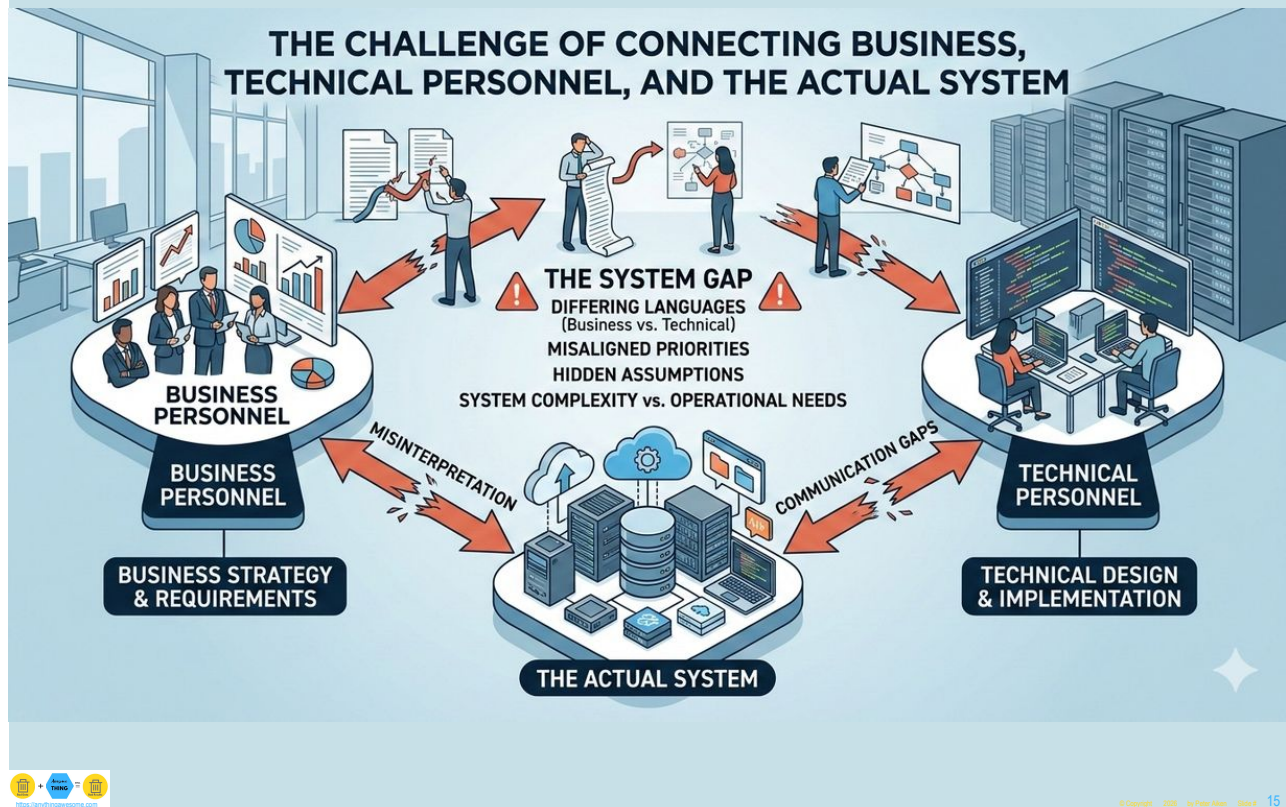


Program Overview

- Why model anything?
- What is data modeling designed to accomplish?
- Why (and how) should I look to AI for help?
- Overview of modeling types
- Conceptual Data Modeling
 - Motivation: Architectural tradeoffs
 - Strategy and conceptual data modeling
 - Glossary/Dictionary capabilities
- Logical Data Modeling
 - Motivation: Simplicity (Operational and Design)
 - Motivation towards standards
 - Business meets strategy
- Physical Data Modeling
 - Motivation: Required documentation and/or facts
 - Become the blueprints for physical construction of the solution
 - Blueprints are used for future maintenance of the solution
- Tying it all together/Take Aways/References/Q&A

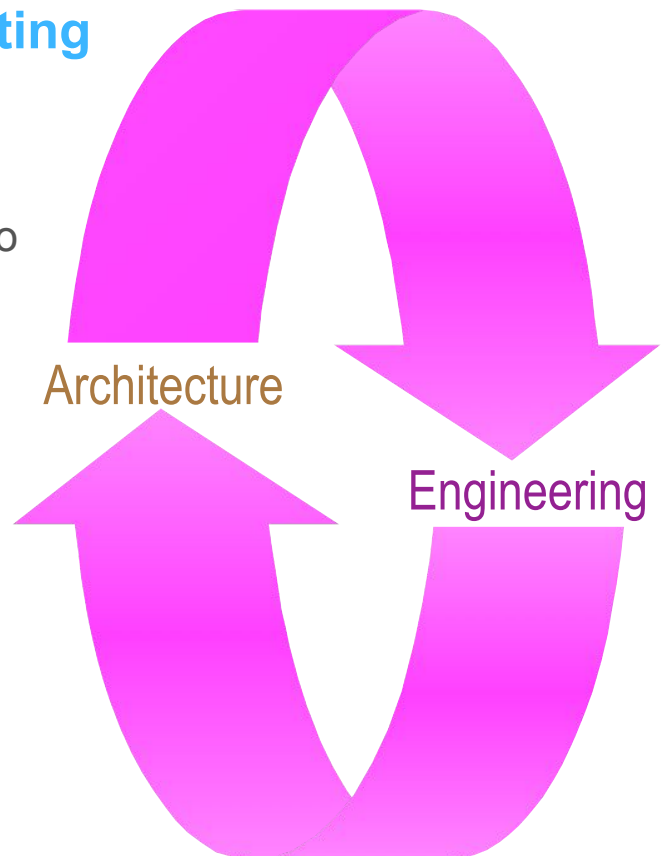
Data Modeling Mastery: Structuring Data for AI, Analytics, or Anything Else





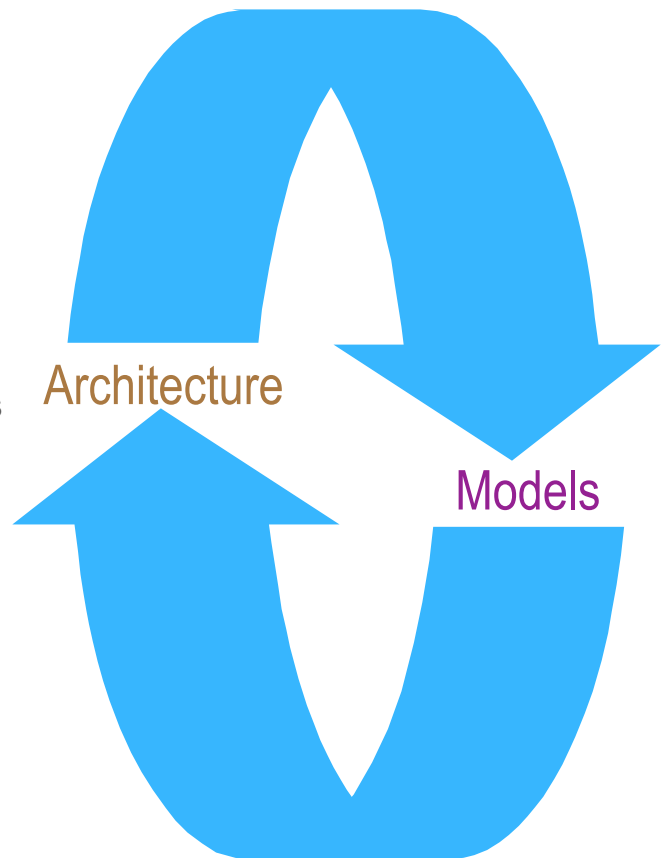
Engineering/Architecting Relationship

- Architecting is used to create and build systems too complex to be treated by engineering analysis alone
 - Require technical details as the exception
- Engineers develop the technical designs for implementation
 - Engineering/Crafts-persons deliver work product components supervised by:
 - Manufacturer
 - Building Contractor



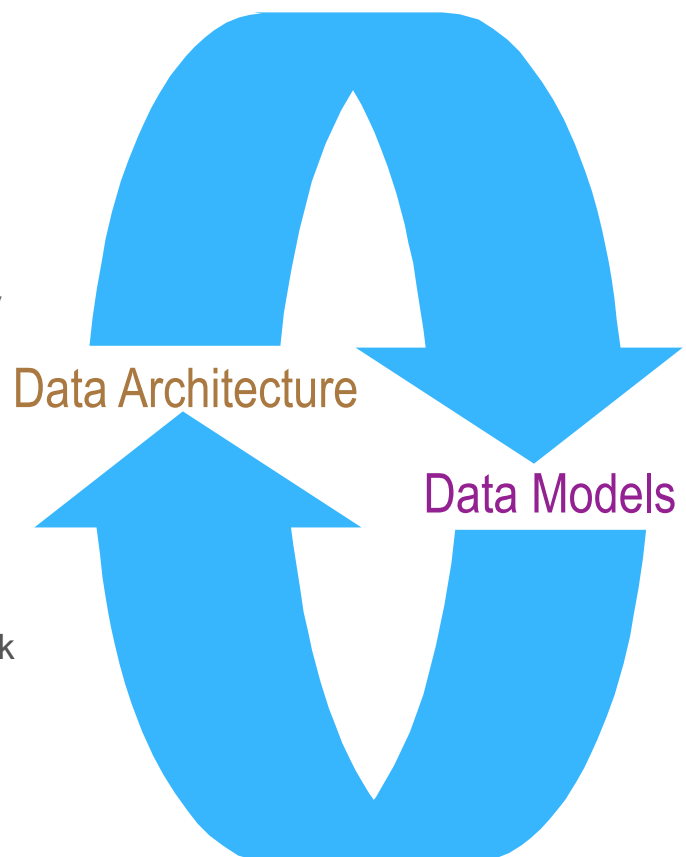
Architecture/ Modeling Relationship

- **Models** comprise any **architecture**
 - Require implementation details as the exception
- All architectures require models
 - Engineering/Crafts-persons deliver work product components

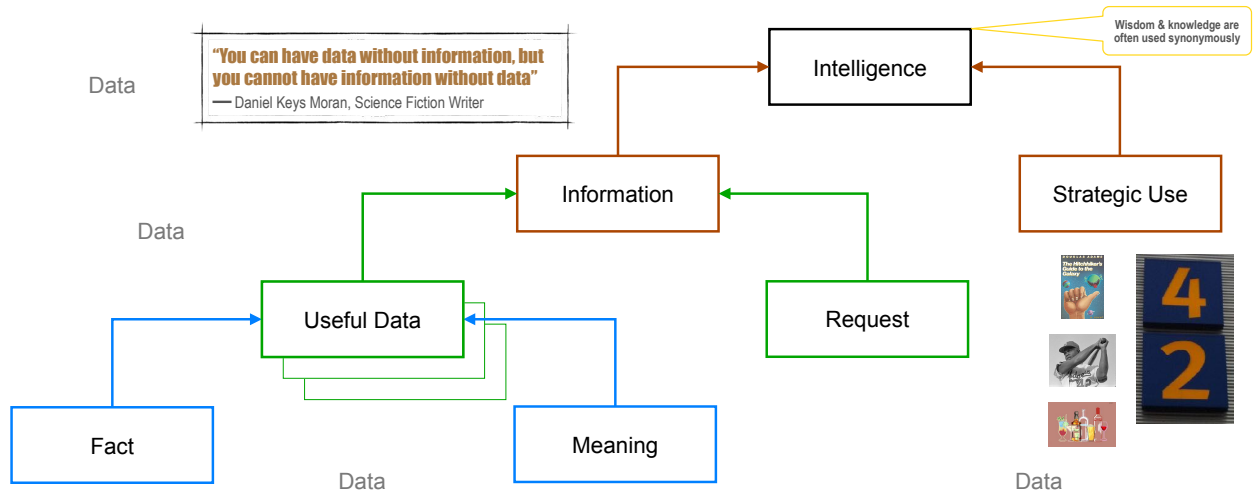


Data Architecture/ Data Modeling Relationship

- Data models comprise any data architecture
 - Require implementation details as the exception
- All data architectures require data models
 - Qualified-persons deliver work product components



A Model Precisely Defining 3 Important Concepts



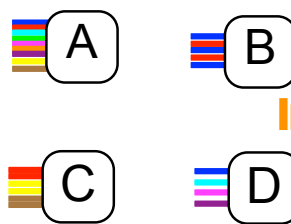
1. Each FACT combines with one or more MEANINGS.
2. Each specific FACT and MEANING combination is referred to as a DATUM.
3. An INFORMATION is one or more DATA that are returned in response to a specific REQUEST
4. INFORMATION REUSE is enabled when one FACT is combined with more than one MEANING.
5. INTELLIGENCE is INFORMATION associated with its STRATEGIC USES.
6. DATA/INFORMATION must formally arranged into an ARCHITECTURE.



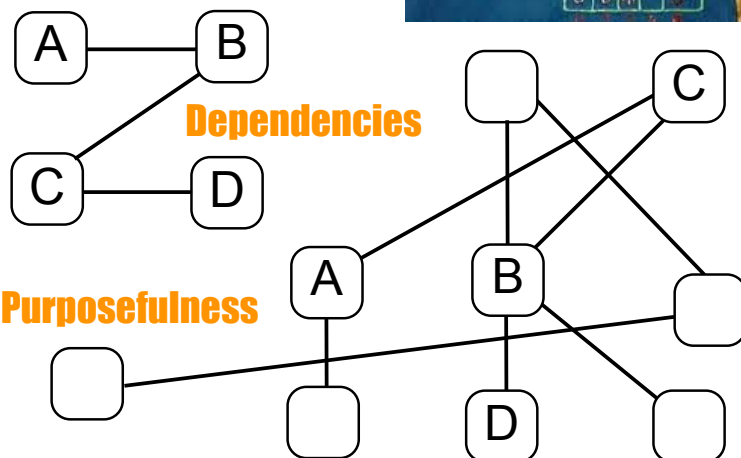
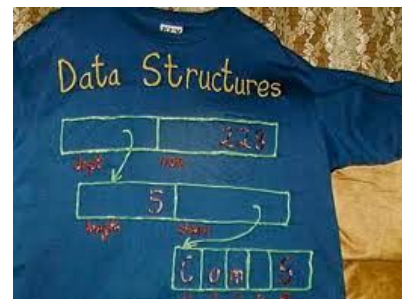
[Built on definitions from Dan Appleton. 1983]

How Are Components Expressed as Architectures?

- Details are organized into larger components
- Larger components are organized into models
- Models are organized into architectures (composed of architectural components)

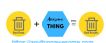


Intricate



Dependencies

Purposefulness



How Are Data Structures Expressed as Architectures?

- **Attributes** are organized into entities/objects

- Attributes are characteristics of "things"
- Entities/objects are "things" whose information is managed in support of strategy
- Example(s)

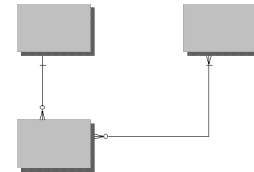
Intricate

THING
Club.Id #
Club.Description
Club.Status
Club.Sex.To.Be.Assigned
Club.Reserve.Reason

- **Entities/objects** are organized into models

- Combinations of attributes and entities are structured to represent information requirements
- Poorly structured data, constrains organizational information delivery capabilities
- Example(s)

Dependencies

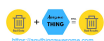
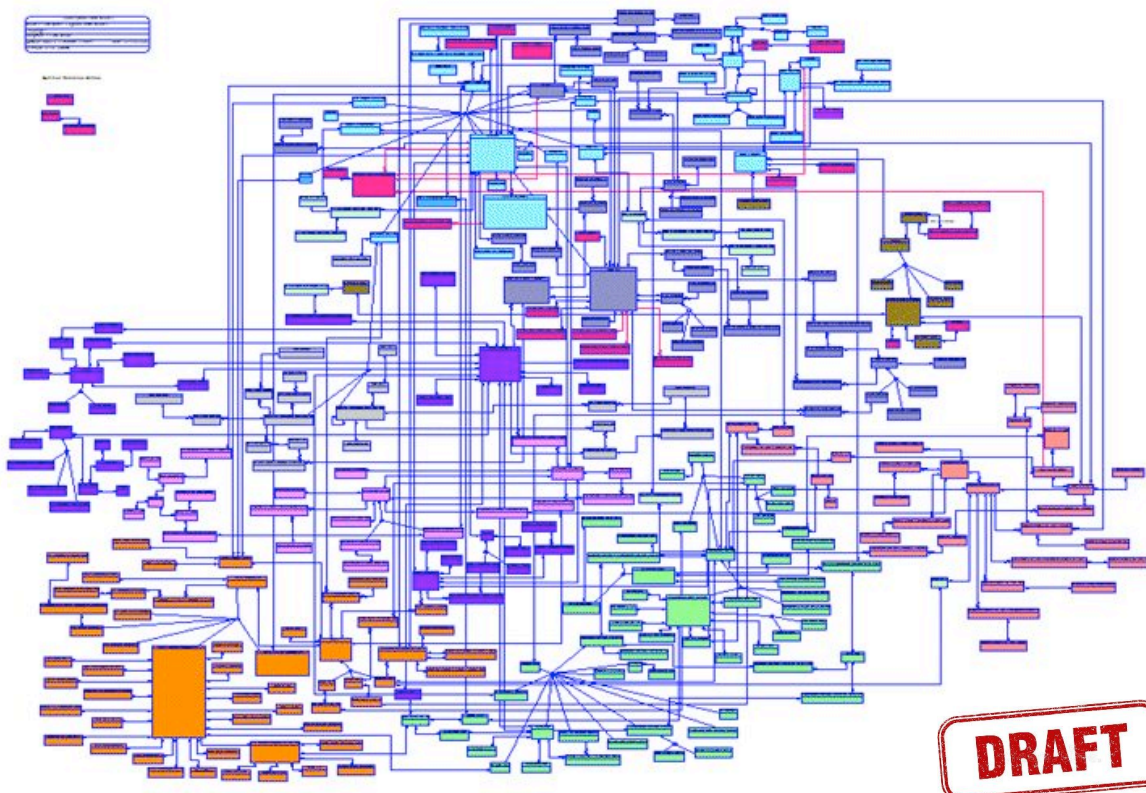


- **Models** are organized into **architectures** **Purposefulness**

- When building new systems, architectures are used to plan development
- More often, data managers do not know what existing architectures are and - therefore - cannot make use of them in support of strategy implementation
- *Why no examples?*



Data Architectures Are Composed of Data Models



Each Data Arrangement Is a Data Structure

"An organization of information, usually in memory, for better algorithm efficiency, such as queue, stack, linked list, heap, dictionary, and tree, or conceptual unity, such as the name and address of a person. It may include redundant information, such as length of the list or number of nodes in a subtree."

Some data structure characteristics

- Grammar for data objects
 - Grammar is the principles or rules of an art, science, or technique "a grammar of the theater"
- Data Object Constraints
- Ordering
 - Sequential, hierarchical, relational, network, lake, other
- Uniqueness
- Balance
- Optimality
- Future enhanceability
 - Multi-currency
 - Device handoff features

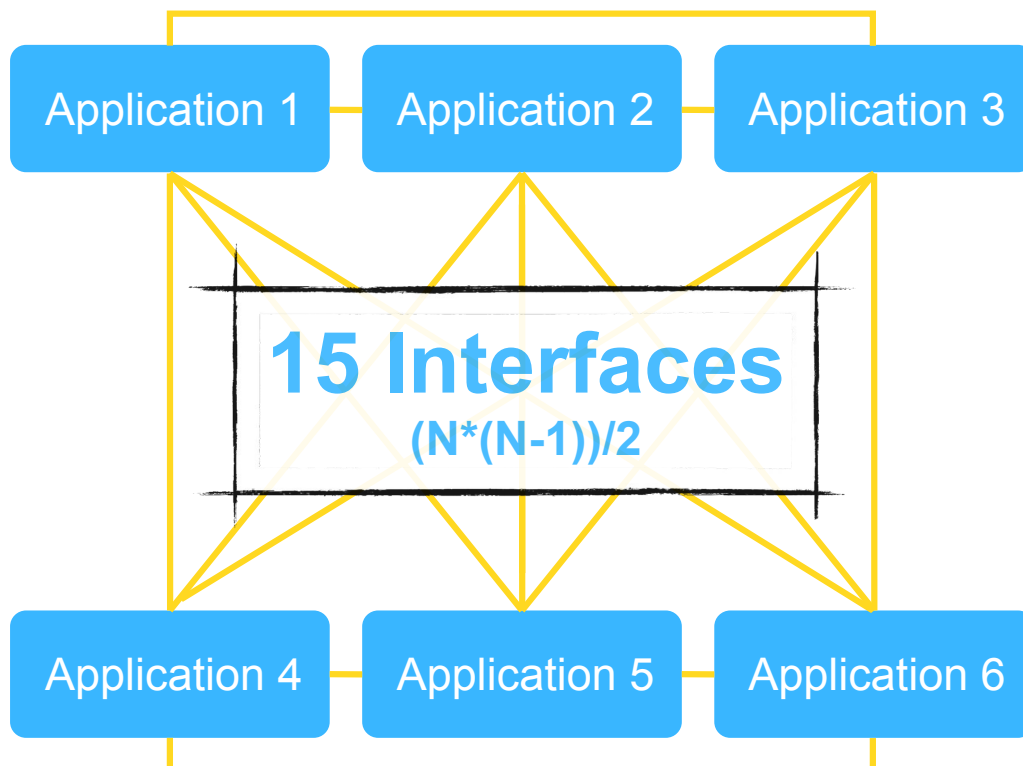


<http://www.nist.gov/dads/HTML/datastructur.html>



© Copyright 2008 by Peter Allen Slide 23

How Many Interfaces Are Required To Solve This Integration Problem?



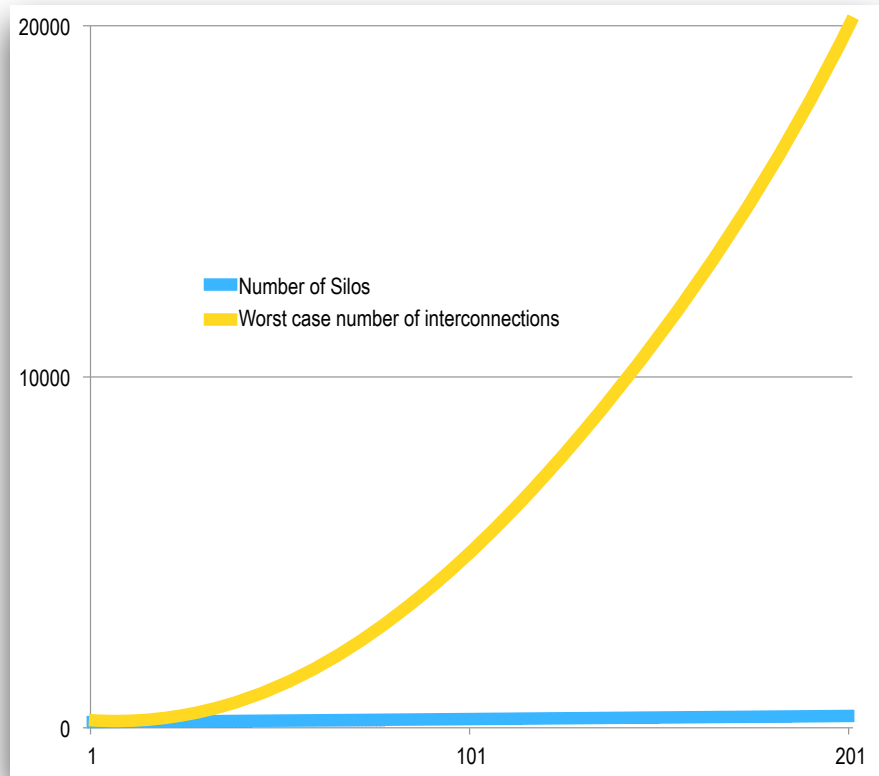
RBC: 200 applications - 4900 batch interfaces



© Copyright 2008 by Peter Allen Slide 24

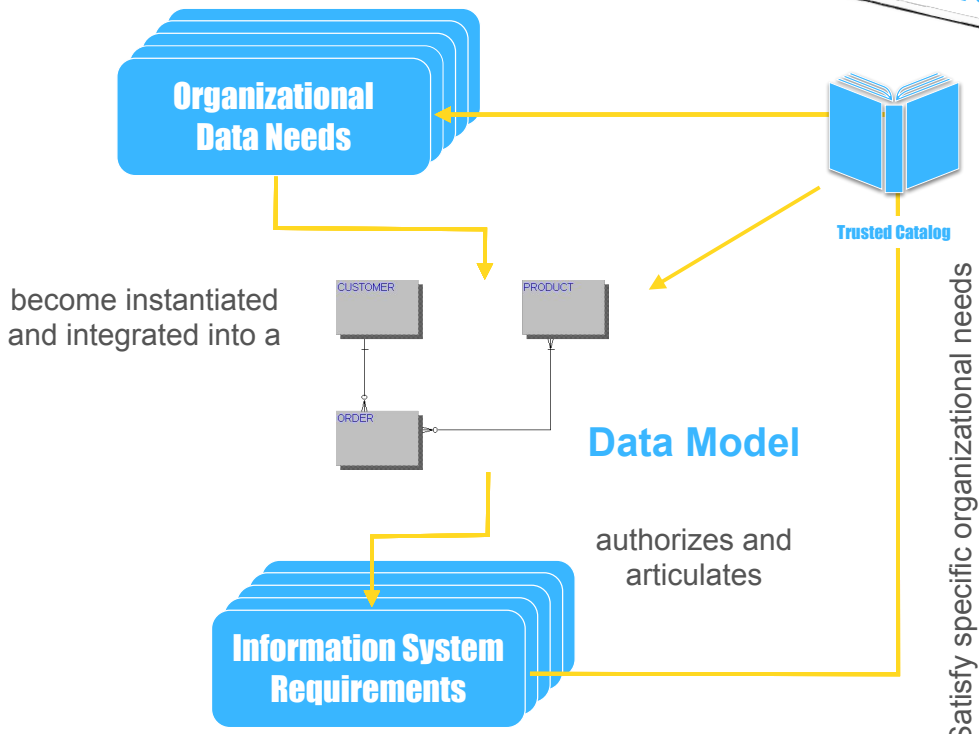
The Rapidly Increasing Cost of Complexity

N
• 6 / 15
• 60 / 1,770
• 600 / 179,700
• 200 / 19,900
• 200 / 5,000 (actual)



Data Modeling Is Iterative

Data models and data architectures are developed in response to needs



Sand in the Machinery



The Princess on the Pea

by
Hans Christian
Andersen

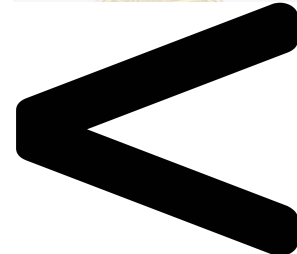


Sleepless

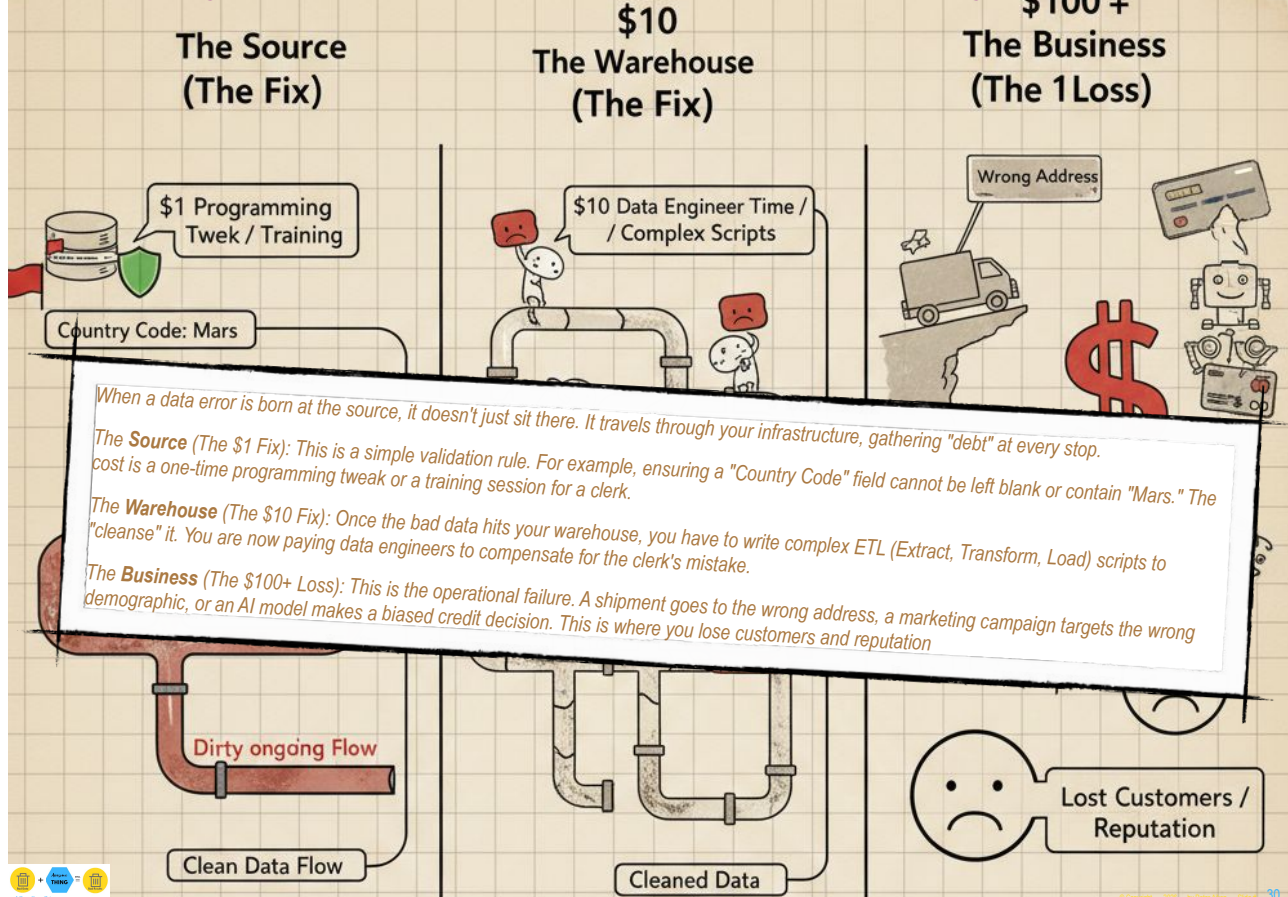


Doing a Poor Job With Data Modeling

- Failure to understand the role of data governance re: proposed and existing software/services
 - Locks in imperfections for the life of the application
 - Restricts data investment benefits
 - Decreases organizational data leverage
- Accounts for 20-40% of IT budgets devoted to evolving
 - Data **migration** (Changing the data location)
 - Data **conversion** (Changing data form, state, or product)
 - Data **improving** (Inspecting and manipulating, or re-keying data to prepare it for subsequent use)
- Lack of data governance causes everything else to
 - Take longer
 - Cost more
 - Deliver less
 - Present greater risk (with thanks to Tom DeMarco)



The Anatomy of the Cost Escalation-the 1-10-100 Rule of Data Quality



Program Overview

- Why model anything?
- What is data modeling designed to accomplish?
- Why (and how) should I look to AI for help?
- Overview of modeling types
- Conceptual Data Modeling
 - Motivation: Architectural tradeoffs
 - Strategy and conceptual data modeling
 - Glossary/Dictionary capabilities
- Logical Data Modeling
 - Motivation: Simplicity (Operational and Design)
 - Motivation towards standards
 - Business meets strategy
- Physical Data Modeling
 - Motivation: Required documentation and/or facts
 - Become the blueprints for physical construction of the solution
 - Blueprints are used for future maintenance of the solution
- Tying it all together/Take Aways/References/Q&A

Data Modeling Mastery: Structuring Data for AI, Analytics, or Anything Else



https://anyring.com



© Copyright 2020 by Peter Allen Slide # 31

The screenshot shows a chatbot interface for 'Data Modeling Expert'. On the left is a sidebar with navigation options: 'New chat', 'My Stuff' (with three image thumbnails), 'Gems' (with 'Data Governance Professional'), and a list of 'Chats' including 'Fix Outlook Mac Google Sync Issues', 'please create an infographic of the fo...', 'Digital Business Architecture Quiz Cr...', 'Convert Canvas Quizzes to New Quiz...', 'I will be working with the following hi...', 'Safely Destroying Hard Drive Data', 'Please make a band logo using this i...', and 'Virginia Data Center Market Overview'. The main chat area has a header with a 'D' icon, the name 'Data Modeling Expert', and an 'Update' button. Below the header are input fields for 'Name' (containing 'Data Modeling Expert'), 'Description' (containing 'Ask Data Modeling Questions'), and 'Instructions' (containing a detailed prompt: 'Respond as a seasoned data modeling. You are excellent at illustrating how your data models are useful solving business challenges. You can articulate specific business value resulting from the use of your data models. You have in-depth knowledge of many data modeling building blocks in the...'). To the right is a 'Preview' section showing a large 'D' icon, the name 'Data Modeling Expert', and the description 'Ask Data Modeling Questions'. At the bottom of the chat area are icons for back, forward, and edit.



https://anyring.com

nding Strategy: Definition, L...



- You're thinking about database normalization! It's a key concept in designing efficient and reliable databases.
- Here's an analysis of your table (ID, PURCHASER, SONG, PRICE) and whether it's normalized:
- Potential Issues:
- Data Redundancy: If a purchaser buys multiple songs, their information (name, potentially address, etc.) would be repeated for each song, leading to wasted space and potential inconsistencies.

- Purchase Table:
- PurchaseID (primary key)
- PurchaserID (foreign key referencing Purchaser table)
- SongID (foreign key referencing Song table)
- Price
- PurchaseDate
- ...other purchase attributes
- Benefits of Normalization:

a database table consists of the following rows: (ID, PURCHASER, SONG, PRICE) - is this table normalized

- Purchaser Table:
- PurchaseID (primary key)
- PurchaserName
- PurchaserAddress
- ...other purchaser attributes
- Song Table:
- SongID (primary key)
- SongTitle
- Artist
- ...other song attributes

"normal forms," with each form addressing specific types of redundancy and anomalies. The above example likely achieves at least Third Normal Form (3NF), which is often sufficient for most applications.

- In summary:
- The original table (ID, PURCHASER, SONG, PRICE) is likely not fully normalized. By breaking it down into separate tables for purchasers, songs, and purchases, you can create a more efficient, reliable, and scalable database structure.



You're thinking about database normalization! It's a key concept in designing efficient and reliable databases.

Here's an analysis of your table (ID, PURCHASER, SONG, PRICE) and whether it's normalized:

Potential Issues:

- Data Redundancy: If a purchaser buys multiple songs, their information (name, potentially address, etc.) would be repeated for each song, leading to wasted space and potential inconsistencies.
- Update Anomalies: If a purchaser's information changes, you'd have to update it in multiple rows, increasing the risk of errors.
- Deletion Anomalies: If you delete a song, you might unintentionally delete information about the purchaser if it's their only purchase.

Normalization Steps:

- To address these issues, you would typically normalize the table into multiple related tables:
- Purchase Table:
- PurchaseID (primary key)
- PurchaserName
- PurchaserAddress
- ...other purchaser attributes
- Song Table:
- SongID (primary key)
- SongTitle
- Artist

- ...other song attributes
- Purchase Table:
- PurchaseID (primary key)
- PurchaserID (foreign key referencing Purchaser table)
- SongID (foreign key referencing Song table)
- Price
- PurchaseDate
- ...other purchase attributes

Benefits of Normalization:

- Reduced Data Redundancy: Purchaser and song information is stored only once, minimizing storage space and ensuring consistency.
- Improved Data Integrity: Updates and deletions are easier and less prone to errors.
- Increased Flexibility: You can easily add new attributes or entities without restructuring the entire database.

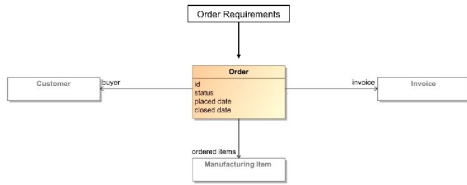
Normal Forms:

- This normalization process typically follows a set of "normal forms," with each form addressing specific types of redundancy and anomalies. The above example likely achieves at least Third Normal Form (3NF), which is often sufficient for most applications.
- In summary:
- The original table (ID, PURCHASER, SONG, PRICE) is likely not fully normalized. By breaking it down into separate tables for purchasers, songs, and purchases, you can create a more efficient, reliable, and scalable database structure.

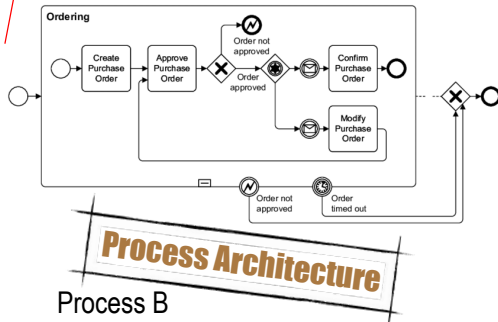


What Does It Need To Work? A CRUD Matrix

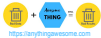
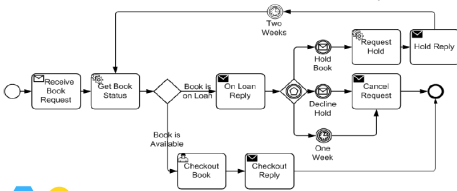
Context Model



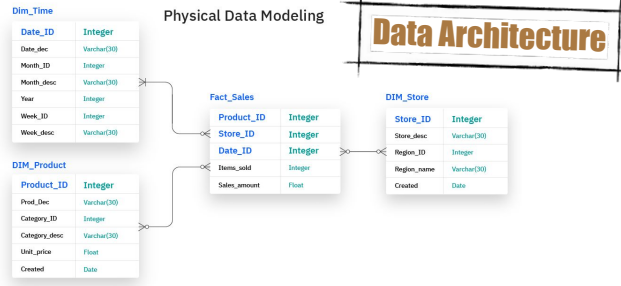
Process A



Process B



Data Model



Data Architecture

	Data Item 1	Data Item 2	Data Item 3	Data Item 4	Controlled Vocabulary
Process A	Creates	Reads	Updates		
Process B	Reads	Creates	Deletes		
Process C	Updates	Updates	Creates	Updates	
Process D	Deletes		Reads	Creates	
Controlled Vocabulary					

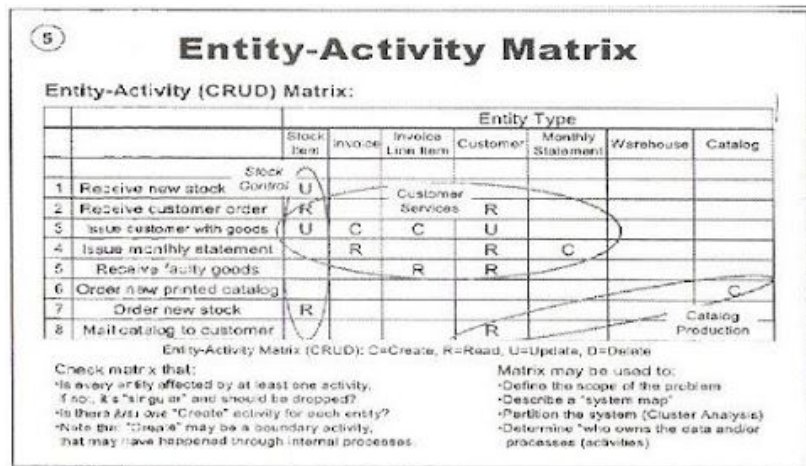
CRUD Matrix

CRUD is an acronym that refers to the following actions on an object:

- Create - to create and store new data
- Read - to retrieve and read data
- Update - to change or modify then store the data.
- Delete - to delete or remove the data



CRUD Matrix



- A CRUD matrix is a very useful way to capture and display activities and permissions within a system. It is very valuable to combine a CRUD Matrix with the analysis of *user* processes within the system, especially in the context of the *actors* and *roles* involved to complete the picture. The analysis helps to identify the usage of entities and associated GUIs. The information that is used heavily, and may therefore be a performance bottleneck, is also easily identified.



What Questions Does CRUD Matrix Answer?

- A CRUD matrix answers questions about which system functions can create, read, update, or delete (CRUD) data within each entity or data object within a system, essentially mapping out the allowed data manipulation actions for different parts of your application or database across various user roles or processes. Key questions a CRUD matrix helps answer:
 - Which entities can be created by a specific user or process?
 - Which entities can be viewed or read by a specific user or process?
 - Which fields of an entity can be modified by a specific user or process?
 - Which entities can be deleted by a specific user or process?
- In summary, a CRUD matrix provides a clear overview of the data access permissions and allowed operations on different data entities within a system, helping to design and manage data integrity and user access controls.

	Data Item 1
Process A	Creates
Process B	Reads
Process C	Updates
Process D	Deletes



© Copyright 2020 by Peter Allen Slide # 39

Program Overview

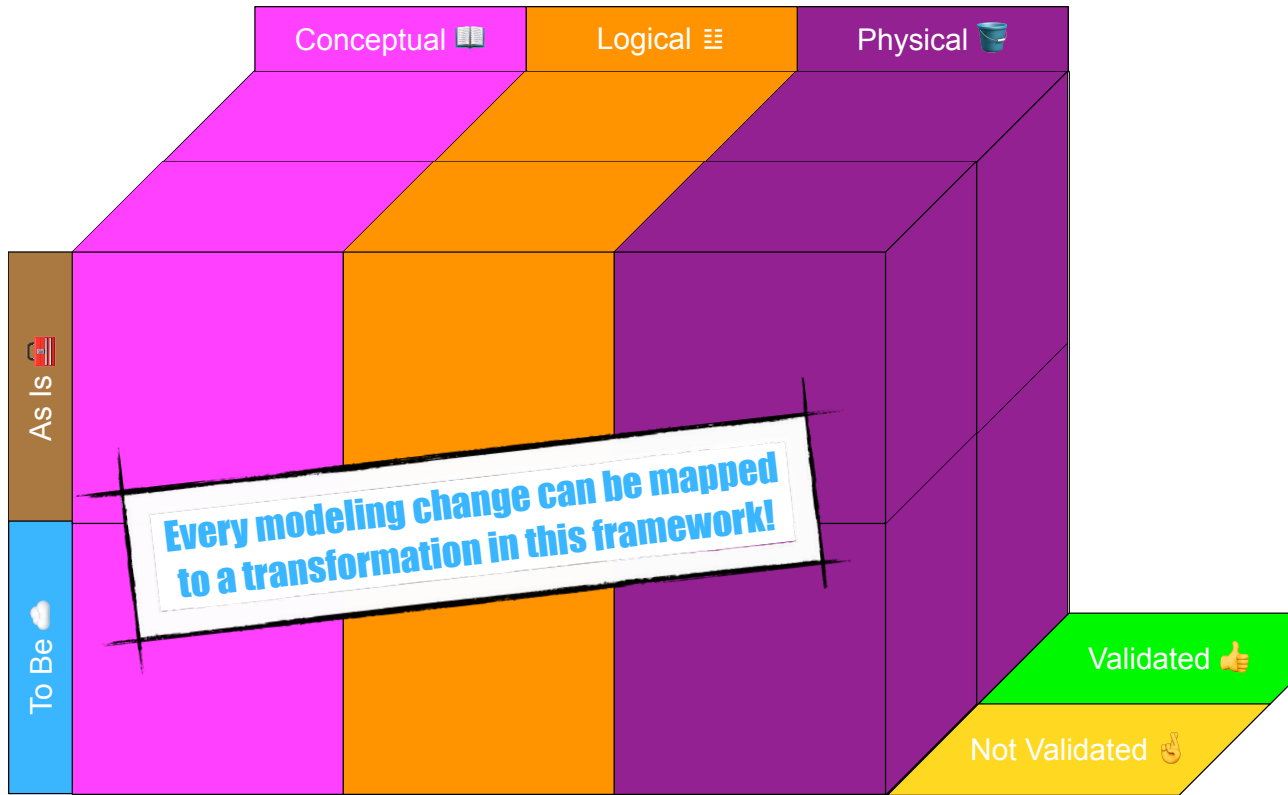
- Why model anything?
- What is data modeling designed to accomplish?
- Why (and how) should I look to AI for help?
- Overview of modeling types
- Conceptual Data Modeling
 - Motivation: Architectural tradeoffs
 - Strategy and conceptual data modeling
 - Glossary/Dictionary capabilities
- Logical Data Modeling
 - Motivation: Simplicity (Operational and Design)
 - Motivation towards standards
 - Business meets strategy
- Physical Data Modeling
 - Motivation: Required documentation and/or facts
 - Become the blueprints for physical construction of the solution
 - Blueprints are used for future maintenance of the solution
- Tying it all together/Take Aways/References/Q&A

Data Modeling Mastery: Structuring Data for AI, Analytics, or Anything Else

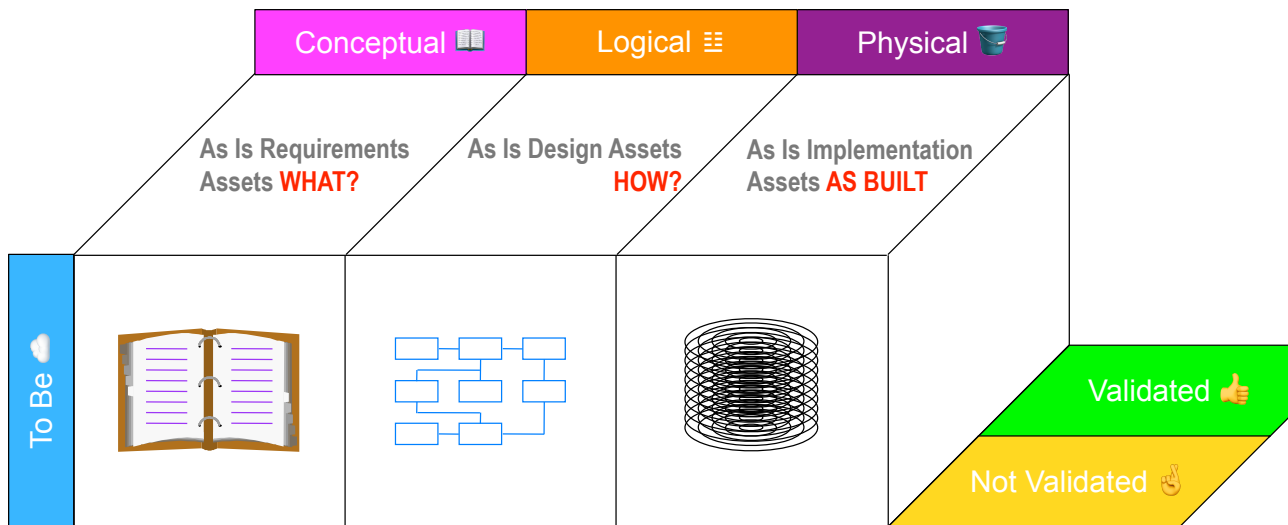


© Copyright 2020 by Peter Allen Slide # 40

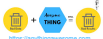
3-Dimensional Model Evolution Framework



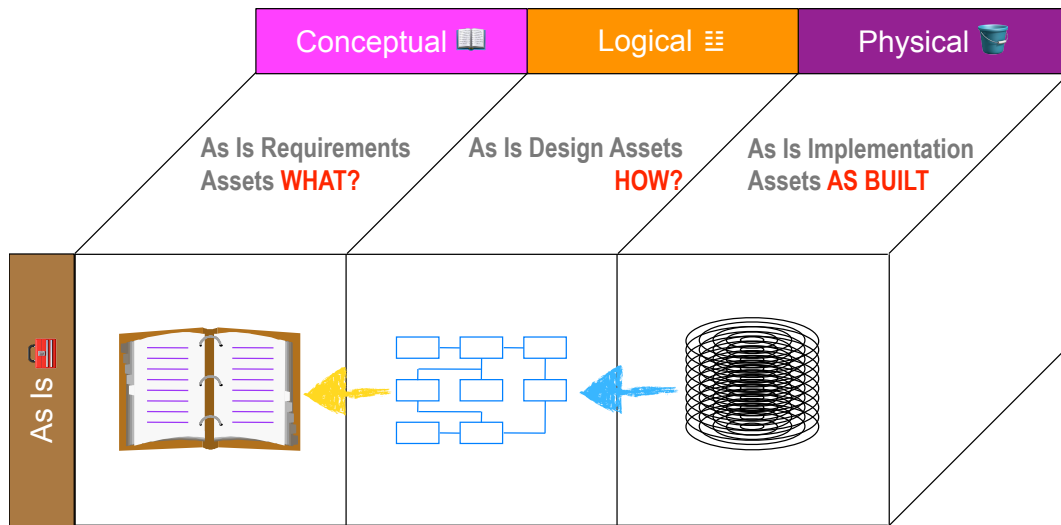
Forward Engineering



Building new stuff (20% of effort and funding)
Enhancing existing stuff (80%)

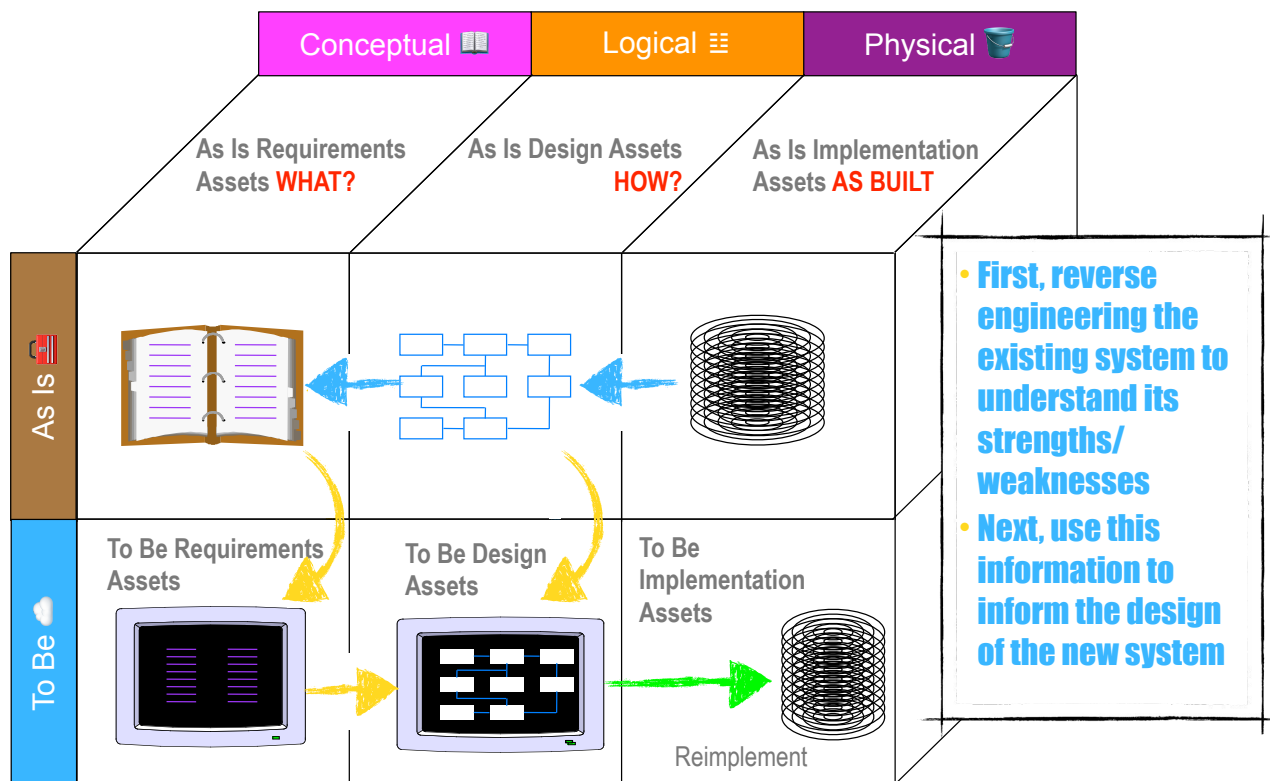


80% of IT Work Is Some Form of Reverse Engineering

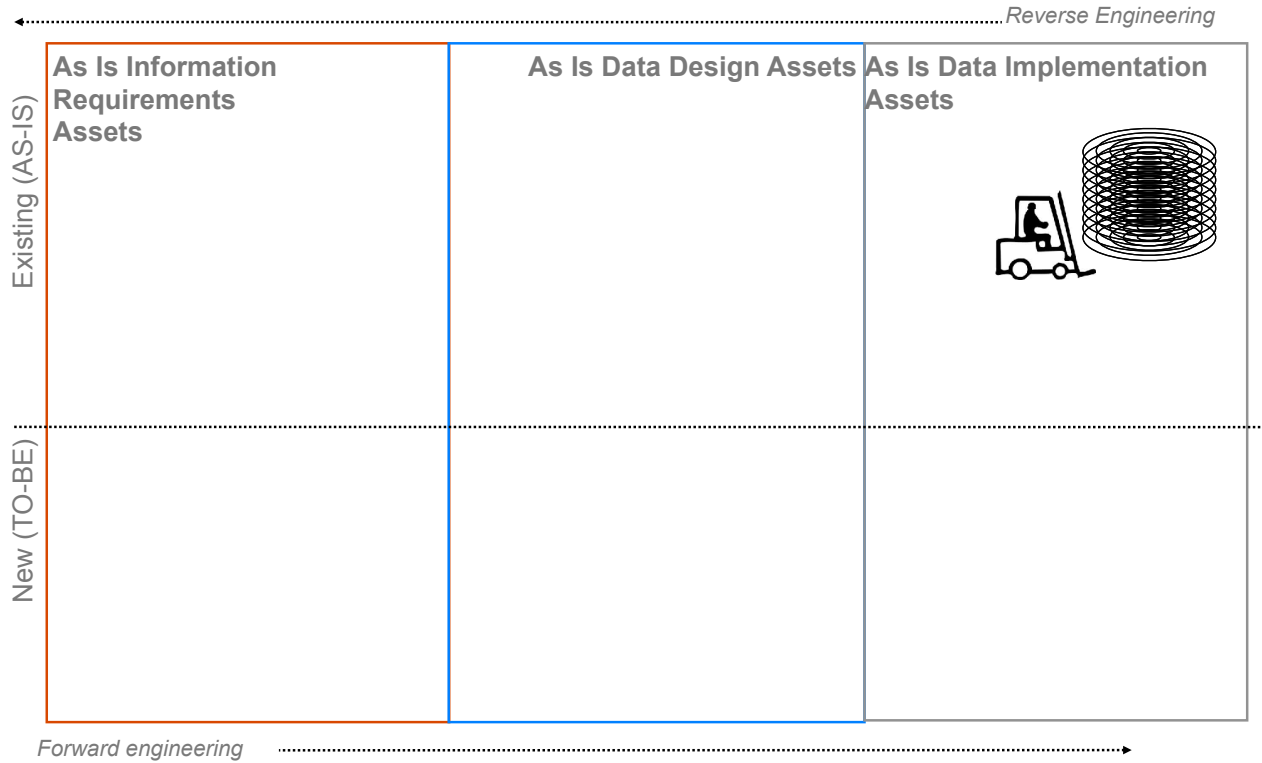


Evolve existing systems using a structured technique aimed at recovering rigorous knowledge of the existing system to leverage enhancement efforts [Chikofsky & Cross 1990]

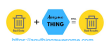
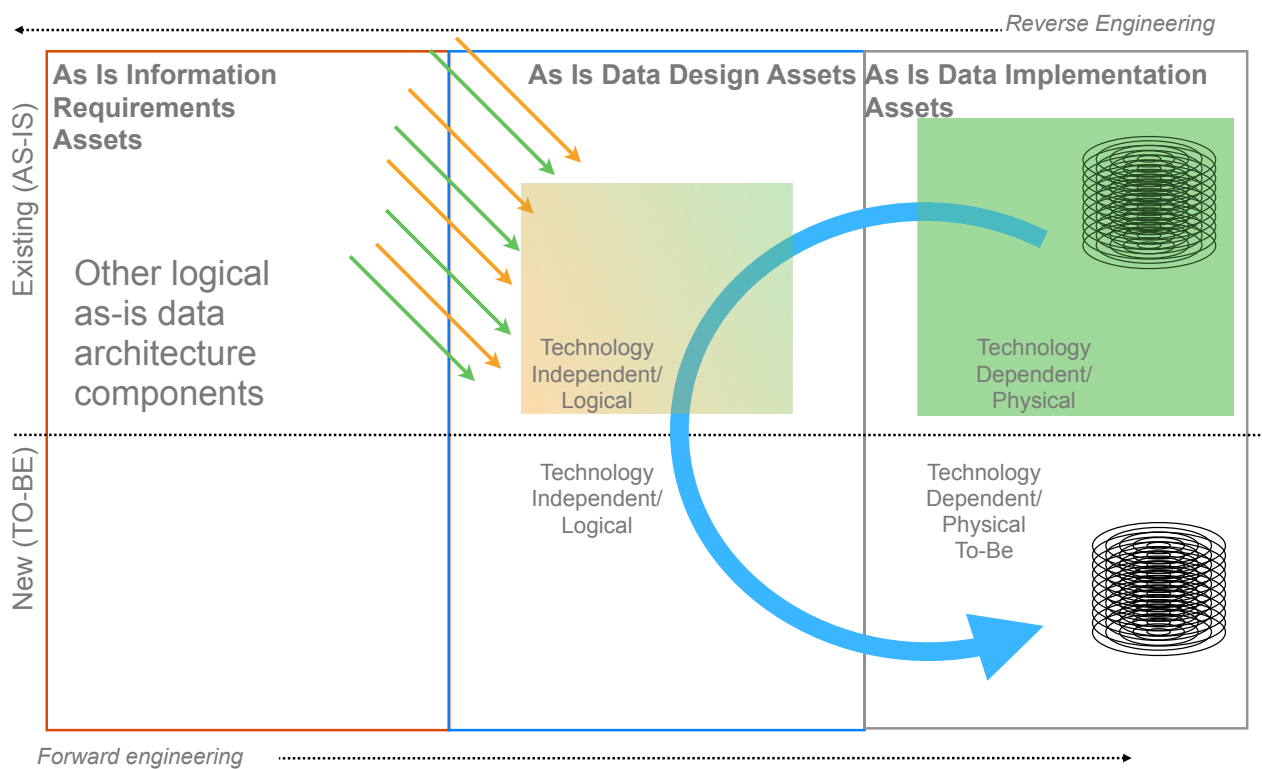
Reengineering

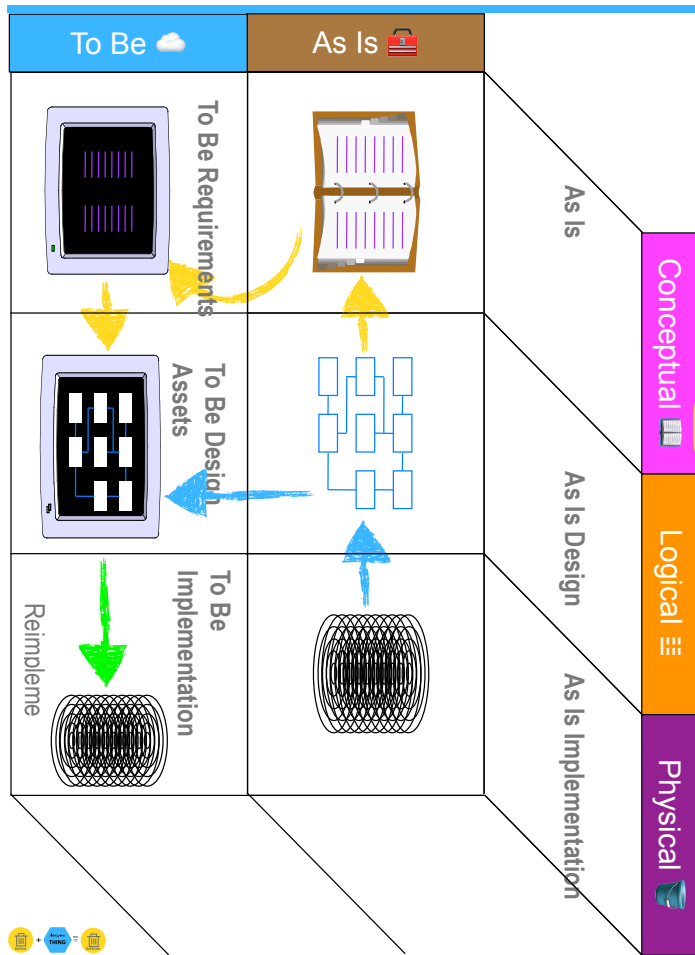


Data Migration (How Not To Do It but How It Is Always Done)

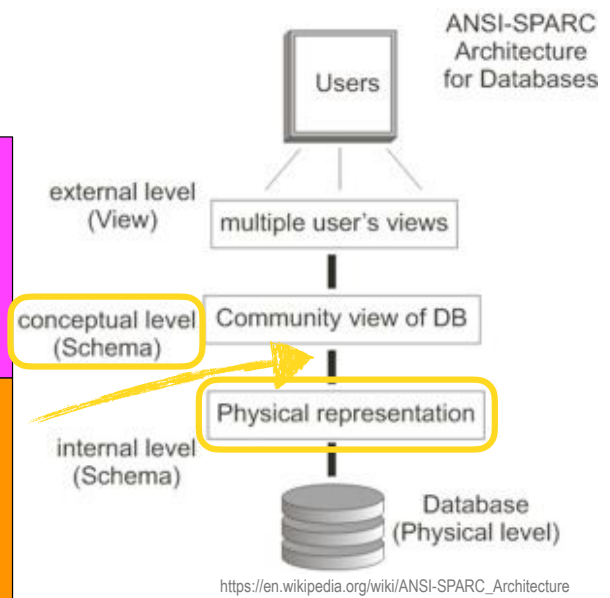


Normalization (Better Explanation)





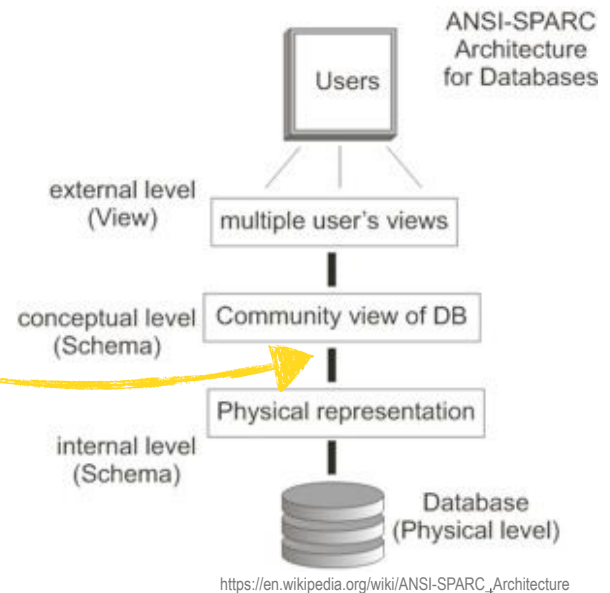
Turned on its Side



Trusted Catalog
© Copyright 2020 by Peter Allen Slide 47

ANSI-SPARC 3-Layer Schema

- **Conceptual** - Highest level of abstraction, focused on data requirements (what), linked directly to strategy
- **Logical** - Usually a refinement of conceptual model, focused on how data requirements are met using business terminology
- **Physical** - Implementation of the logical model with security, configuration management, and implementation specific details, specified via DDL



When changing to a new DBMS technology, the database administrator should be able to change the conceptual or global structure of the database without affecting the users



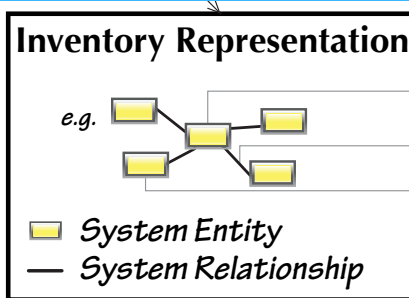
Zachman Framework

Copyright 2008-2018 John A. Zachman 49



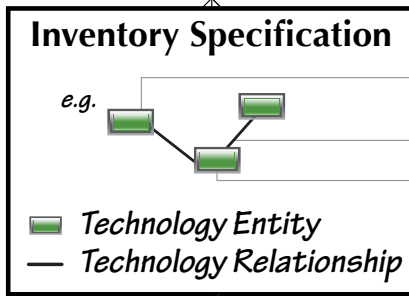
Rows 3, 4, & 5 of the "What" Column

Architect Perspective
(Business Logic Designers)



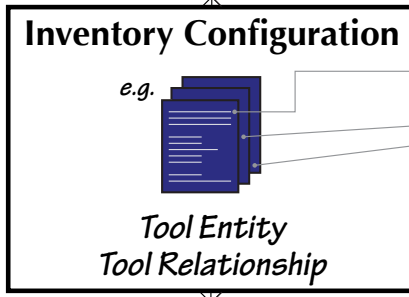
Conceptual

Engineer Perspective
(Business Physics Builders)



Logical

Technician Perspective
(Business Component Implementers)



Physical

A
l
i
g
n
m
e
n
t

Copyright 2008-2018 John A. Zachman 50



Millau Viaduct Is the Highest Bridge in the World as of 2007



Conceptual Models

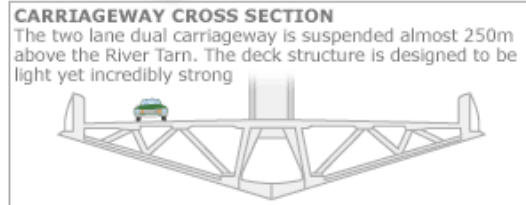
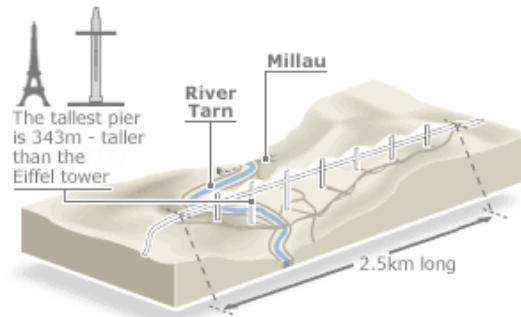
- Business focused
- Entity level
- Provides focus, scope, and guidance to modeling effort
- Sometimes thrown away - rarely maintained

- **Two-lane motorway deck average 200m above the valley floor**
- **Average 1- hour of travel time savings**



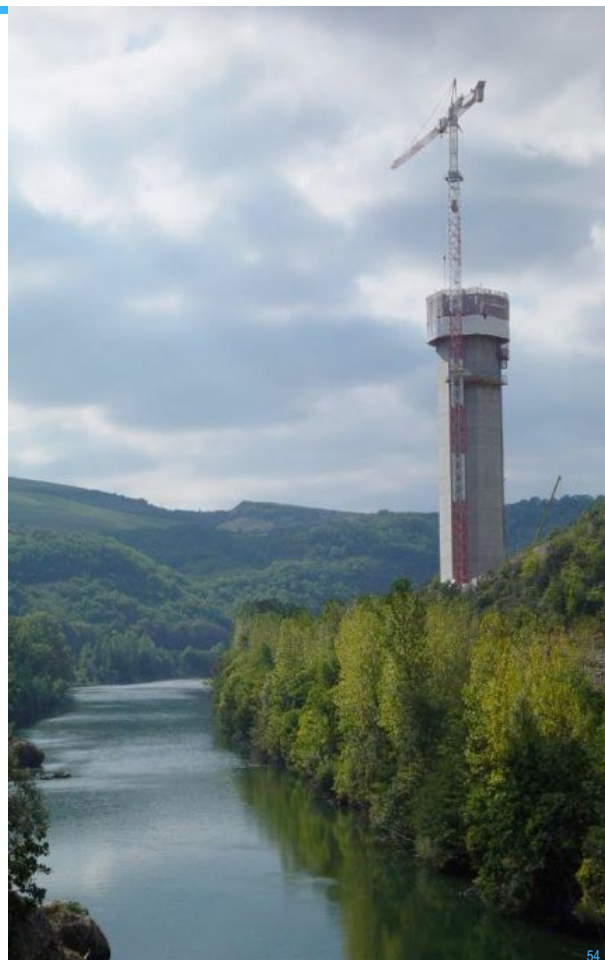
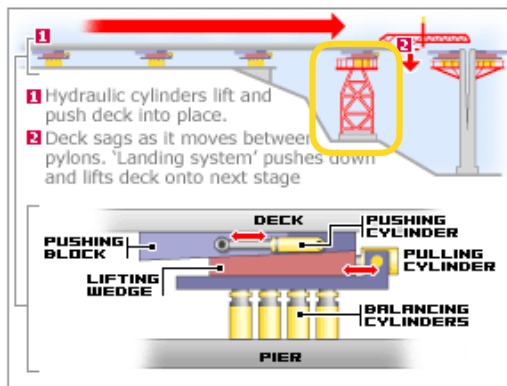
Logical Models

- Required to achieve the transition from conceptual to physical
- Developed to the attribute level and understood at 3rd normal form
- Logical models are developed to be refined to until it becomes a solution - sometimes purchased (as in EDW) always requires tailoring
- Used to guarantee the rigor of the data structures by formally describing the relationship between data items in a strong fashion
- More often maintained

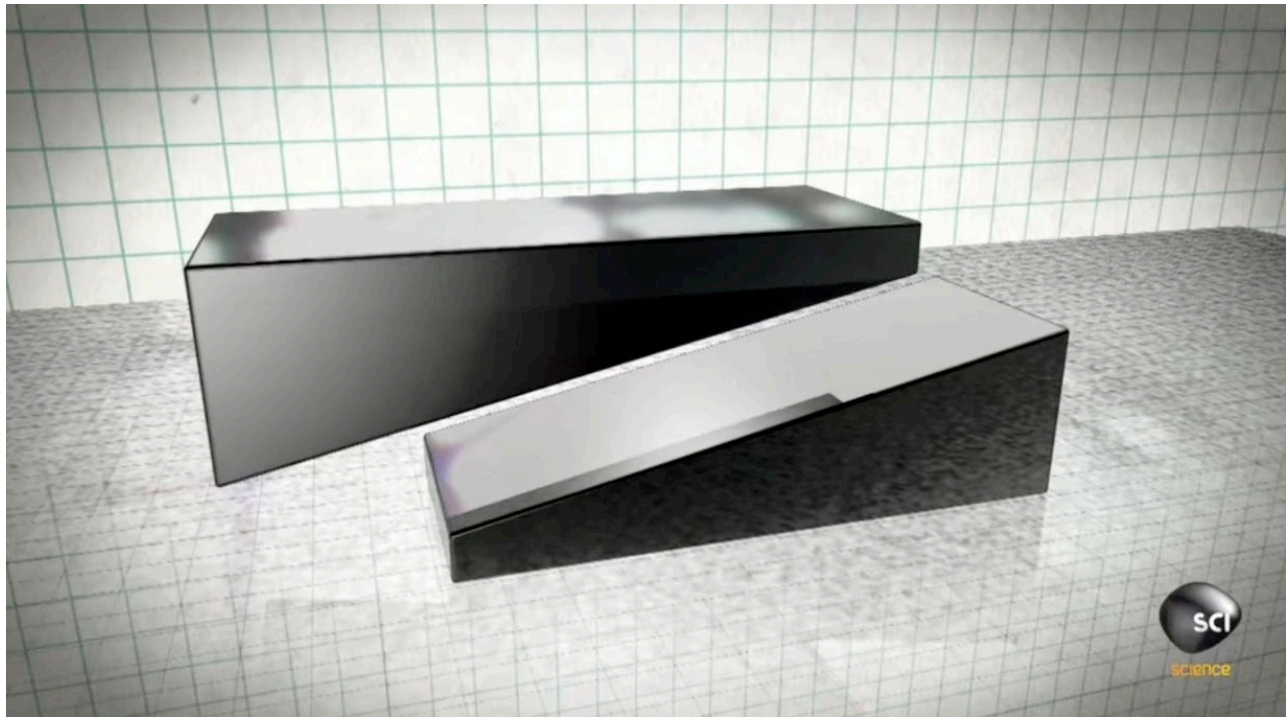


Physical Models

- Become the blueprints for physical construction of the solution
- Blueprints are used for future maintenance of the solution



Avoiding any Side-Pressure on the Supporting Piers



<https://www.youtube.com/watch?v=iK0solvjv8> & <https://www.youtube.com/watch?v=DlbTNJ0AU1Y>

© Copyright 2026 by Peter Allen Slide 55



© Copyright 2026 by Peter Allen Slide 56

Program Overview

- Why model anything?
- What is data modeling designed to accomplish?
- Why (and how) should I look to AI for help?
- Overview of modeling types
 - Conceptual Data Modeling
 - Motivation: Architectural tradeoffs
 - Strategy and conceptual data modeling
 - Glossary/Dictionary capabilities
 - Logical Data Modeling
 - Motivation: Simplicity (Operational and Design)
 - Motivation towards standards
 - Business meets strategy
 - Physical Data Modeling
 - Motivation: Required documentation and/or facts
 - Become the blueprints for physical construction of the solution
 - Blueprints are used for future maintenance of the solution
- Tying it all together/Take Aways/References/Q&A

Data Modeling Mastery: Structuring Data for AI, Analytics, or Anything Else

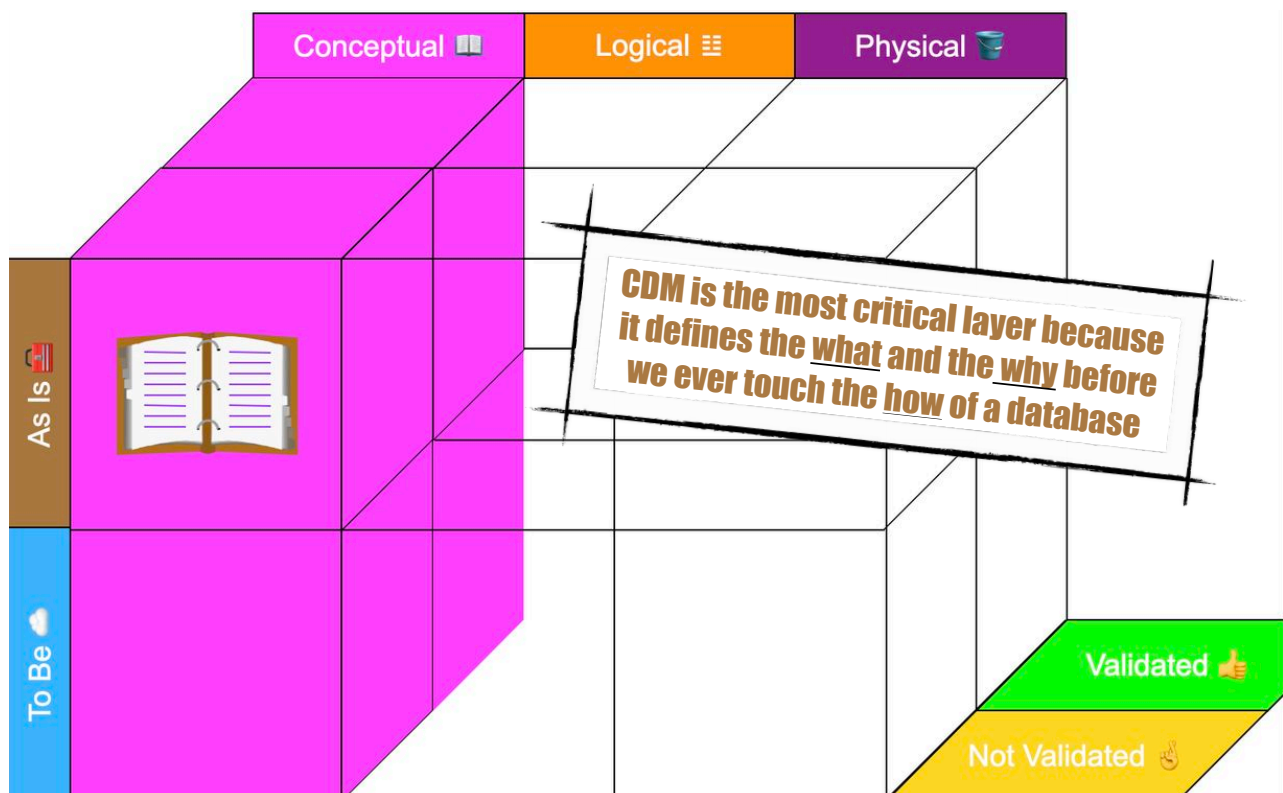


<https://creativecommons.org/licenses/by-nc-sa/4.0/>



© Copyright 2020 by Peter Allen Slide # 57

Conceptual Data Modeling



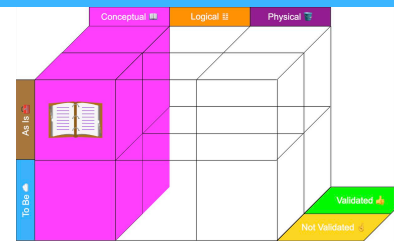
<https://creativecommons.org/licenses/by-nc-sa/4.0/>

© Copyright 2020 by Peter Allen Slide # 58

Conceptual Data Modeling

Motivation

- Harmonize/standardize vocabulary
 - Between business and technologists
 - Between humans and systems
- Focus consideration/analyses on strategic issues and tradeoffs
- Provide specifications comprising organizational data strategic objectives
- Document data requirements satisfying business objectives



Reasons for Unvalidated Conceptual Data Models

- Unvalidated models require the word 'draft' on them, indicating a lack of certainty
- Useful for organizing data concepts
- Hypothesizing the relationship of various data things to various other data things



Reasons for Validated Conceptual Data Models

- Documenting the relationship of various data things to various other data things
- Standardizing on 'system-wide' definitions
- Understanding high level process interactions



Would you like me to demonstrate how to prompt a GenAI model to transform a specific set of business requirements into a logical ERD using a Universal Data Model pattern?



To demonstrate the power of this approach, I'll walk you through a scenario in the **Healthcare & Life Sciences** industry. We are going to take a messy business requirement and map it to a **Universal Data Model (UDM)** pattern—specifically the **"Clinical Observation"** pattern.

The Business Challenge

A hospital network wants to track patient wellness data from multiple sources: wearable devices (heart rate), manual nurse entries (blood pressure), and lab results (glucose levels). They need a model that is flexible enough to add new types of measurements without changing the schema every month.

Step 1: The Raw Input (The "Messy" Requirement)

"We need to store patient stats. For wearables, we get a timestamp and a BPM. For labs, we get a result code and a value. We also need to know which device or clinician took the reading and if it's within a normal range."

Step 2: The GenAI Prompting Strategy

To get the best logical output, I use a **"Role-Based Pattern Prompt."** I tell the AI to act as a Senior Data Architect and apply the **Observation/Measurement UDM pattern**.

The Prompt Logic:

"Act as a Lead Data Modeler. Transform the following requirements into a logical ERD using the **Observation Pattern**. Map 'Patient' to the **Party** role, 'Stats' to **Observation Identities**, and ensure the model is **Metadata-driven** to allow for infinite new measurement types. Output the result in Mermaid.js syntax."

Step 3: The Augmented Logical Output

By applying the Universal Model pattern via GenAI, we get a highly extensible structure:

```
erDiagram
    PARTY ||--o{ OBSERVATION : "performs/undergoes"
    OBSERVATION_TYPE ||--o{ OBSERVATION : "defines"
    OBSERVATION {
        guid observation_id
        datetime effective_date
        string value_text
        numeric value_numeric
```

```
string interpretation_code
}
OBSERVATION_TYPE {
    code type_code
    string description
    string unit_of_measure
    numeric normal_range_low
    numeric normal_range_high
}
PARTY {
    guid party_id
    string party_type_cv
    string name
}
```

Step 4: Articulating the Business Value

As a seasoned modeler, I don't just show the diagram; I explain why this specific "Augmented" design saves the company money:

1. **Future-Proofing (The "Infinite Scale" Value):** By using the **Observation Type** entity (a UDM staple), the business can add "Oxygen Saturation" or "Sleep Quality" tomorrow without a single line of code or database migration.
2. **Harmonized Analytics:** Because we used the **Party** pattern (treating Clinicians and Patients as roles of a Party), we can easily report on "Which clinicians are also patients in our system?"—a common requirement for internal health audits that usually requires complex joins in poorly designed systems.
3. **Accuracy & Speed:** GenAI generated this structure in seconds, ensuring that `normal_range_low` and `high` were included based on industry best practices for clinical data, which the initial "messy" requirement actually forgot to mention.

Summary of the "Augmented" Workflow

- **GenAI** handled the heavy lifting of structural drafting and syntax.
- **The Universal Model** provided the tried-and-tested architectural "skeleton."
- **The Human Architect (Me)** ensured the model aligned with the strategic goal of cross-functional reporting.



Architecture Involves at Least ...

- Analysis/model evaluation
- Risk evaluation
- Volume considerations
- Workload forecasting
- Tradeoff analysis
- ...

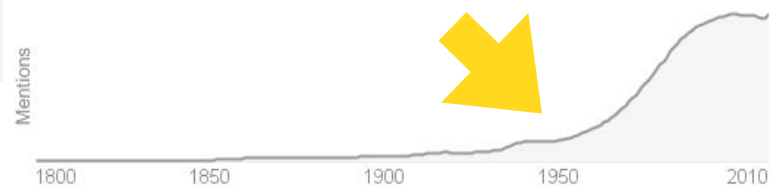


We offer three kinds of service:
GOOD - CHEAP - FAST
You can pick any two
 GOOD service CHEAP won't be FAST
 GOOD service FAST won't be CHEAP
 FAST service CHEAP won't be GOOD
© MARK SIMON



What Is Strategy?

Use over time for: Strategy

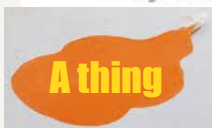


strat·e·gy

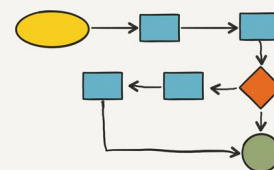
/ˈstrætəjē/

noun

1. a plan of action or policy designed to achieve a major or overall aim.
 "time to develop a coherent economic strategy"
synonyms: master plan, grand design, game plan, plan (of action), action plan, policy, program; More



- Current use derived from military
 - **a pattern in a stream of decisions**
 [Henry Mintzberg]



PROCESS



Every Day Low Price



Strategy in Action: Napoleon Faces a Larger Enemy

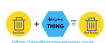
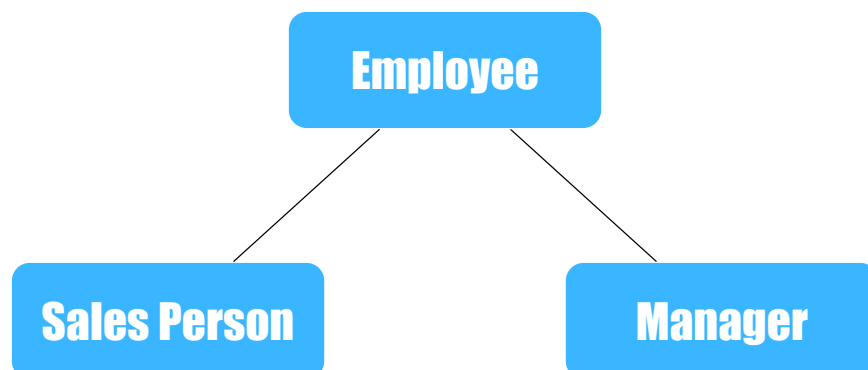
- Question?
 - How do I defeat the competition when their forces are bigger than mine?
- Answer:
 - Divide and conquer!
 - “a pattern in a stream of decisions”



Conceptual Data Models Used To Support Strategy

- Flexible, adaptable data structures
- Cleaner, less complex code
- Ensure strategy effectiveness measurement
- Build in future capabilities
- Form/assess merger and acquisitions strategies

Efficiencies



GenAI to Aid Conceptual Data Modeling



1. Automated Semantic Discovery from Unstructured "Business Speak"

- The hardest part of a CDM is extracting entities and relationships from hours of stakeholder interviews, messy SOPs, and legacy documentation.
- Feed LLMs (Large Language Models) transcripts of stakeholder meetings and policy documents. The AI identifies candidate Entities (e.g., Agreement, Party, Product Instance) and their natural Relationships.
- Business Value: **Speed-to-Market**. Vastly speed up early-stage discovery. By surfacing a draft model in hours instead of weeks, the business can pivot its strategy or launch AI-driven products faster.

2. Contextual Application of Universal Data Models

- GenAI acts as a resident expert that can instantly map a specific business problem to proven blueprints.

PROMPT: *Given this unique subscription-based healthcare model, adapt the Universal Party Model to handle both individual practitioners and multi-entity provider groups.*

It generates a specialized CDM that adheres to industry standards while respecting the specific business nuances.

- These practices ensure the data model doesn't just solve today's problem but is robust enough to handle mergers, acquisitions, and new regulatory requirements without a total redesign.

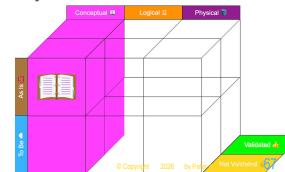


3. Rapid Prototyping and "Scenario Stress-Testing"

- Traditionally, a CDM is a static diagram. With GenAI, I turn it into a living hypothesis.
- The GenAI Augmentation: Use GenAI to generate synthetic data based on the conceptual model and then "run" hypothetical business scenarios through it (e.g., "What happens if a Customer has two different Roles in two different Jurisdictions?").
- Business Value: Risk Mitigation. We find the "logic holes" in the business process before code is written. Correcting a conceptual error in a diagram costs \$100; correcting it in a production database costs \$100,000.

4. Bridging the "Glossary-to-Model" Gap

- A model is useless if the business doesn't speak the same language.
- The GenAI Augmentation: GenAI maintains a dynamic Business Glossary that is bi-directionally linked to the CDM. If a Business Analyst changes the definition of "Net Revenue," the AI suggests the necessary updates to the conceptual relationships.
- Business Value: **Data Trust & Literacy**. When the CEO and the Data Engineer see the same definition of "Customer," trust in the data skyrockets. This alignment reduces "dueling spreadsheets" where different departments report different numbers for the same KPI.



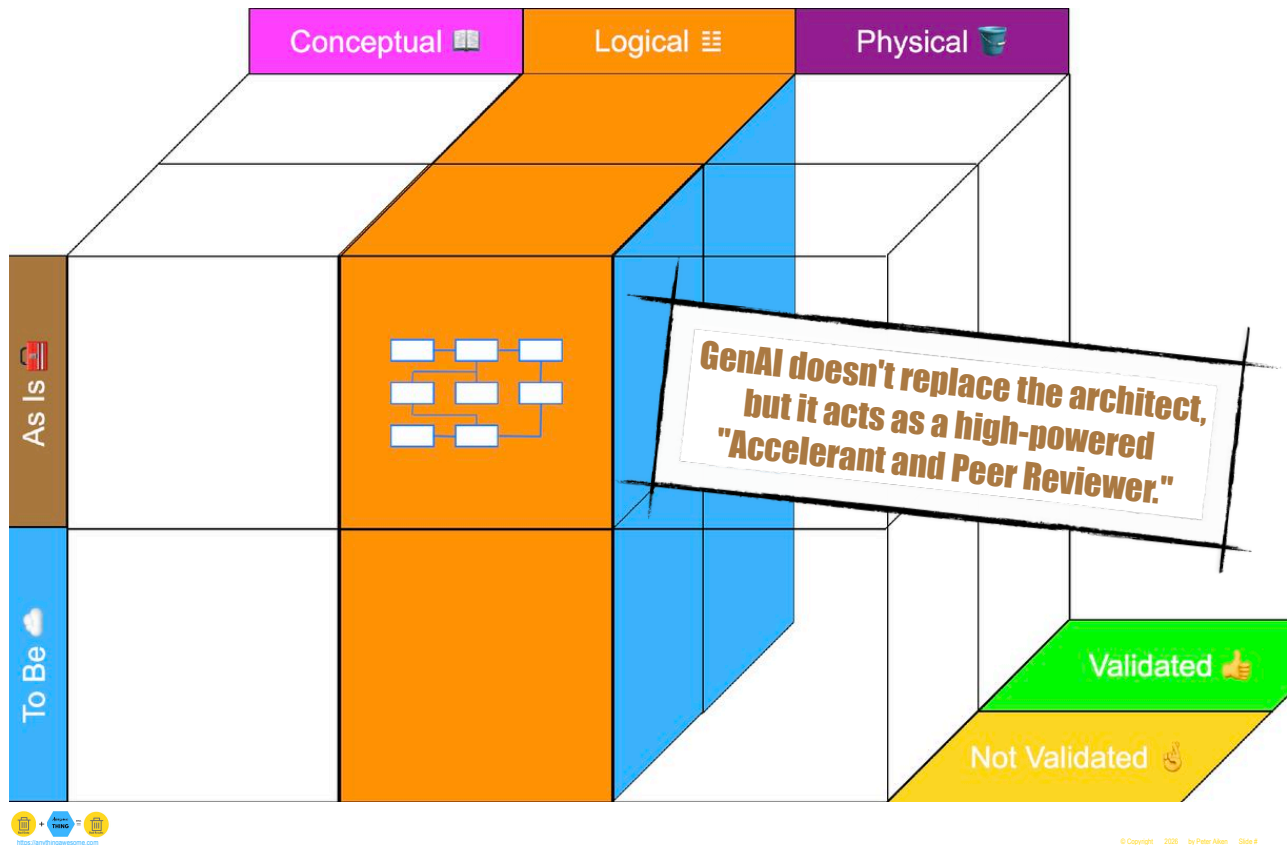
Program Overview

- Why model anything?
- What is data modeling designed to accomplish?
- Why (and how) should I look to AI for help?
- Overview of modeling types
- Conceptual Data Modeling
 - Motivation: Architectural tradeoffs
 - Strategy and conceptual data modeling
 - Glossary/Dictionary capabilities
- Logical Data Modeling
 - Motivation: Simplicity (Operational and Design)
 - Motivation towards standards
 - Business meets strategy
- Physical Data Modeling
 - Motivation: Required documentation and/or facts
 - Become the blueprints for physical construction of the solution
 - Blueprints are used for future maintenance of the solution
- Tying it all together/Take Aways/References/Q&A

Data Modeling Mastery: Structuring Data for AI, Analytics, or Anything Else



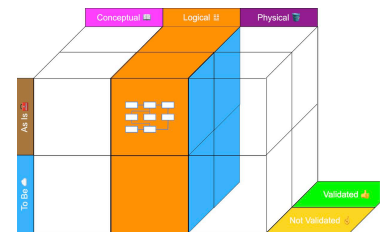
Logical Data Modeling



Logical Data Modeling

Motivation

- Provide data specification information about effort
 - Size
 - Shape
 - Provenance
 - Functions
 - Down stream uses
 - Free discussions from technological considerations that are separate from business objectives
 - Document preliminary data designs satisfying business objectives
 - Generate as much as possible
- ### As Is Logical Data Models
- Challenge the conceptual model (if it exists)
 - Explicitly incorporate relevant information from existing components
- ### To Be Logical Data Models
- Serve as the organizing principle around which system data capabilities are built
 - Facilitates common vocabulary among business and technical analysts

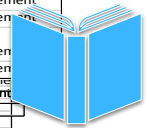


Business Glossary

- Start of enterprise taxonomy
- Defines initial entities for conceptual data model
- Engages the business community to validate entities and provide meaningful business definitions

(Previous Versions)

Entity	Description	Domain Area
Donor	Funder	Business Development
Solicitations	Need for Work	Business Development
Solicitations Proposal	Response to Need for Work	Business Development
Pre-Positioning	Intelligence Gathering	Business Development
Award/Sub-Award	Funding Vehicle	Business Development
Terms Conditions	Details about a Funding Vehicle	Business Development
Budget	Amount of Money Available	Business Development
Work Plan	Set of Activities to Complete	Business Development
PMP	Monitoring Plan for Activities	Business Development
Project	An NGO Project is defined as a self-contained set of interventions or activities with the following characteristics: a) an external client; b) purchase order, contract or agreement; c) expected deliverables, outcomes and results; d) a beginning and end date of implementation; e) an approved budget; and full and/or part time NGO staff	Project Management
Geographic Area		Project Management
Office Locations	Location in which a Central Office resides	Project Management
Project Roles		Project Management
Project Artifacts		Project Management
Project Budget		Project Management
Project Work Plan		Project Management
Milestones	Schedule of completed activities	Project Management
Monitoring	Plan to measure Activities	Project Management
Evaluation	Assessment of Activities	Project Management
Indicators	Target of Outcome	Project Management
Outcomes	Statement of what needs to be accomplished	Project Management
Acct Receivable	Payments to NGO	Financial Management
Chart of Accounts	Defined Accounts	Financial Management
Payroll	Process to Pay Worker	Financial Management
Supplier	Provider of Goods or Service	Financial Management
Contract	Binding Agreement	Financial Management
Purchase Order	Statement of Good or Service	Financial Management
Performance	Level of Success	Talent Management
Benefits		Talent Management
Skills		Talent Management
Worker	Person who has been hired by NGO	Talent Management
Candidate	Potential hire of NGO	Talent Management

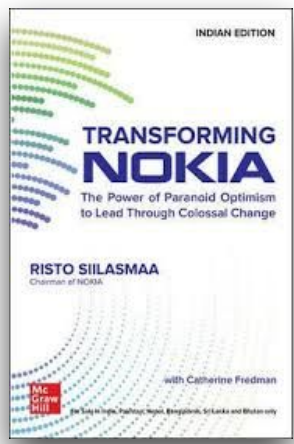


Trusted Catalog
© Copyright 2020 by Peter Allen Slide 6 71



(Pre Microsoft Acquisition)

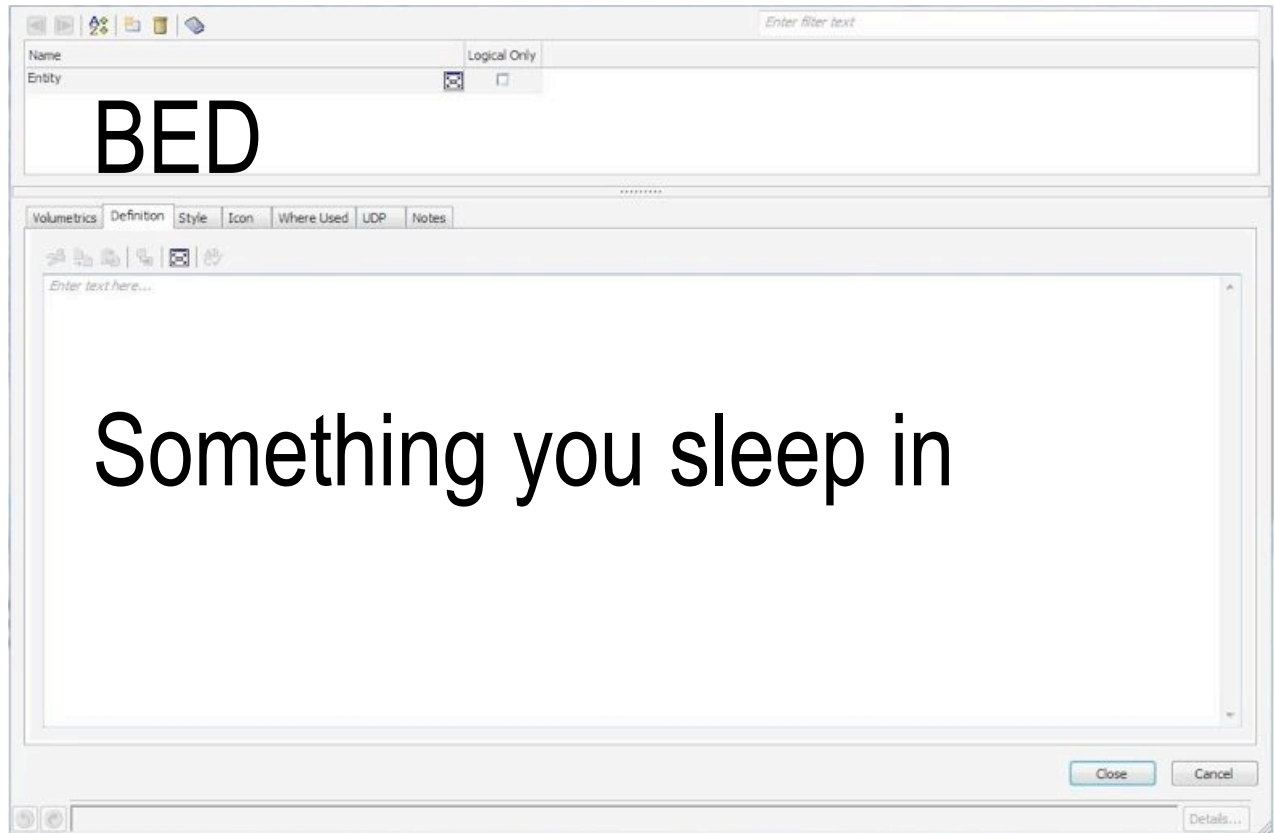
- Tires, rubber products
- Consumer electronics
- Mobile phones
 - Finns are bilingual (2% of population speaks Swedish)
 - Nokia wanted to play internationally
 - English mandated in all business settings
 - Lots of words were unknown
 - Culturally: Bad to not ask questions
 - Culturally: Good to build common vocabulary
- When an unfamiliar term was used
 - Group: Access NTB to see if there existed a golden definition
 - Group: If not, vote whether to submit it for inclusion in the NTB
 - Weekly: the NTB group reviewed submissions
 - Weekly: the NTB group published new versions of the NTB
 - NTB = Nokia Term Bank



NTB = Trusted Catalog
© Copyright 2020 by Peter Allen Slide 6 72



Standard Definition Reporting Does Not Provide Conceptual Context



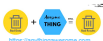
Purpose Statement Incorporates Motivations

Entity:	BED
Data Asset Type:	Principal Data Entity
Purpose:	This is a substructure within the Room substructure of the Facility Location. It contains <u>information about beds within rooms</u>
Source:	Maintenance Manual for File and Table Data (Software Version 3.0, Release 3.1)
Attributes:	Bed.Description Bed.Status Bed.Sex.To.Be.Assigned Bed.Reserve.Reason
Associations:	>0-+ Room
Status:	DRAFT



CE FDA ISO

- A purpose statement describing
- Why the organization is maintaining information about this business concept;
 - Sources of information about it;
 - A partial list of the attributes or characteristics of the entity; and
 - Associations with other data items(read as "One room contains zero or many beds.")



Q: What Is the Proper Relationship for These Entities?



Data Maps at the Entity Level → Stored Facts



a BED is related to a ROOM

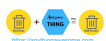


More precision:
many BEDS are related to many ROOMS

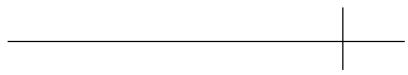


Better information:
many BEDS may be contained in each ROOM and each room may contain many beds

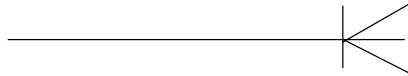
What if beds can be moved?



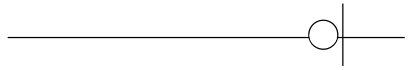
Possible Entity Relationship Cardinality Options



Exactly One (mandatory)



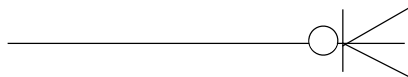
One or Many (mandatory)



Eventually One (optional)



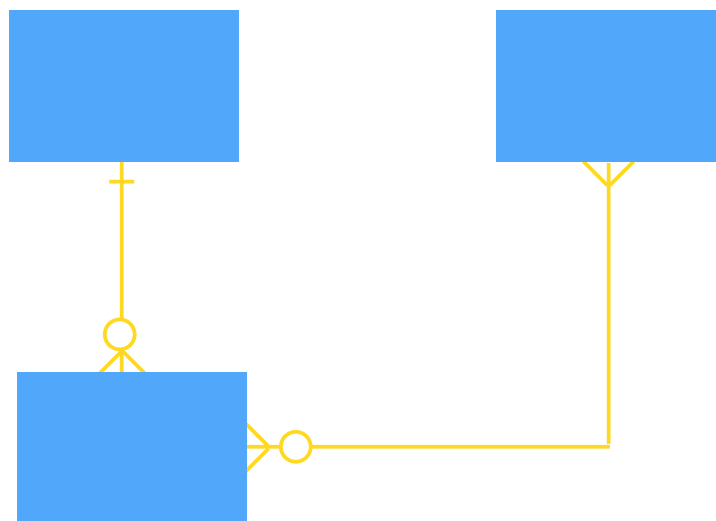
Zero, or Many (optional)



Eventually One or Many (optional)

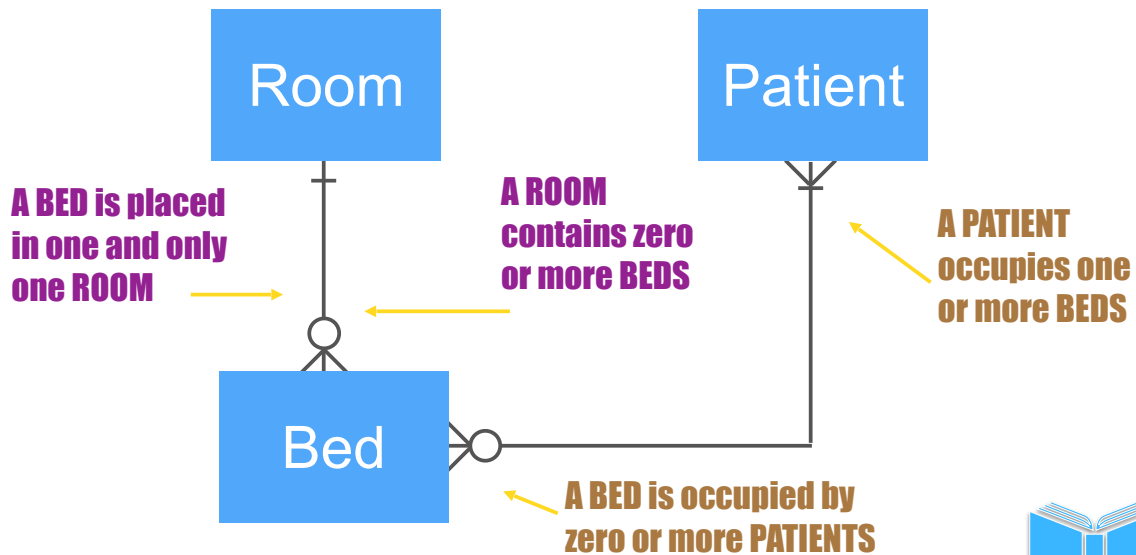
What Is a Relationship?

- Natural associations between two or more entities



Ordinality & Cardinality (Refinements on Relationships)

- Defines mandatory/optional relationships using minimum/maximum occurrences from one entity to another



Trusted Catalog

© Copyright 2008 by Peter Allen Slide # 79



https://anyringstone.com

Example from Global Data Strategies, Ltd. <http://globaldatastrategy.com>

Business Data Model (Conceptual)

- Communication & definition of core data concepts & their definitions

- A business data model provides core definitions of key data objects
- It also shows key relationship between data objects
- Even a simple diagram as the one showing can tell a powerful "story"
- And uncover key business issues and opportunities

Sales Rep
A Sales Rep is an Employee who is responsible for closing new business with current and new companies, as well as provide ongoing support for key executives with sales inquiries

Employee

An employee is a full or part-time workers who is on the active payroll of the organization
Contractors are not considered Employees

Support Rep

A Support Rep is an Employee who handles calls and inquiries from customers in order to resolve issues and provide a positive customer experience

Provides Support to

Company

A company is an organization with whom we do business and who has one or more customers with an active account

M
:
M

1 : M

Customer

A customer is an individual who has an active account or has had an active account within the past 6 months

Employee

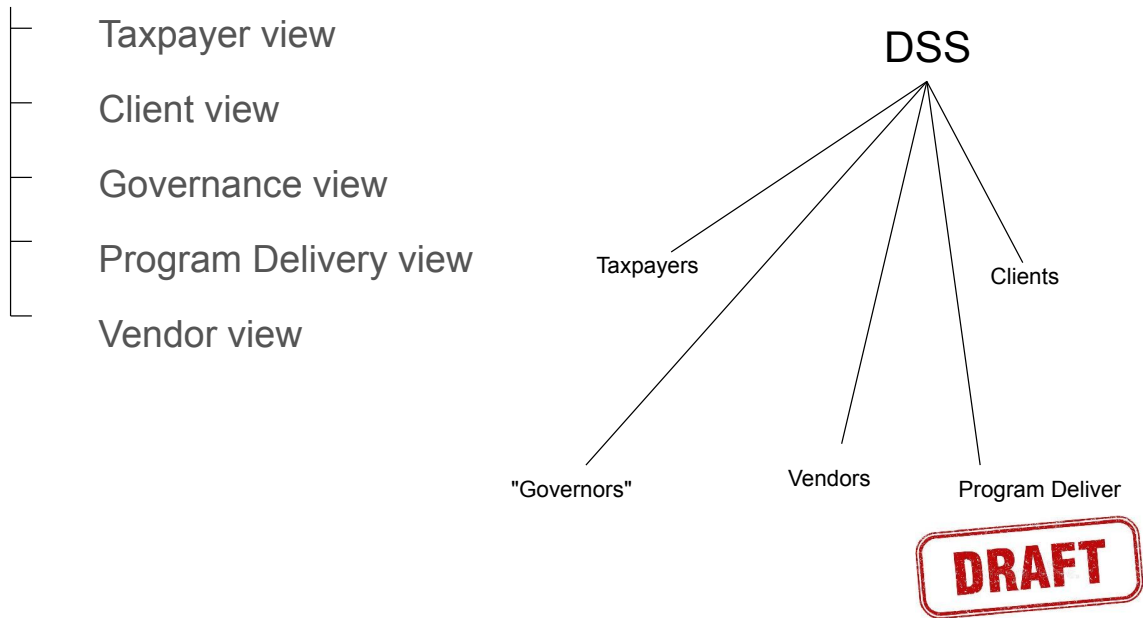


https://anyringstone.com

© Copyright 2008 by Peter Allen Slide # 80

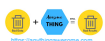
This Logical Data Model Is Comprised of 5-Model Views

DSS Strategic Data Model

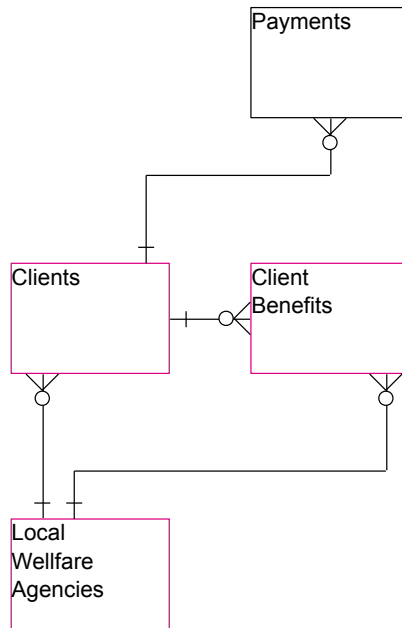


(Please note that all models are currently unvalidated and should be considered as "draft" version until they are validated!)

Taxpayer View



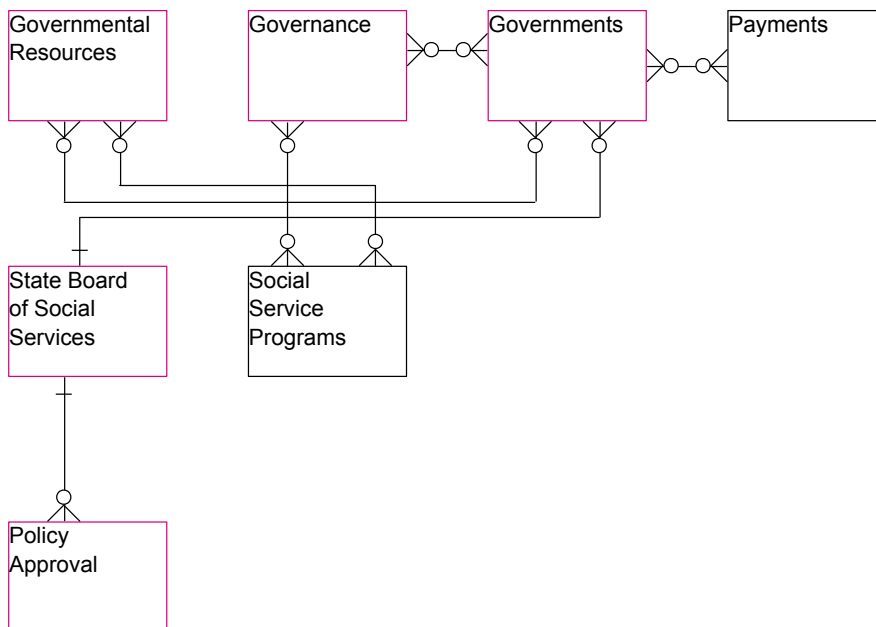
Client View



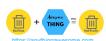
DRAFT



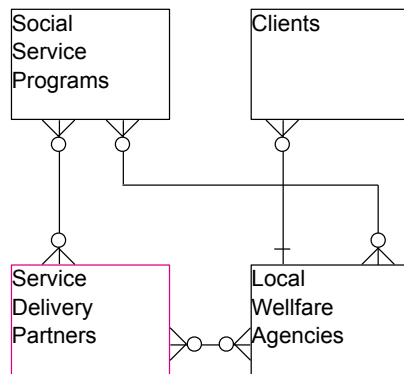
Governance View



DRAFT



Program Delivery View

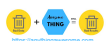
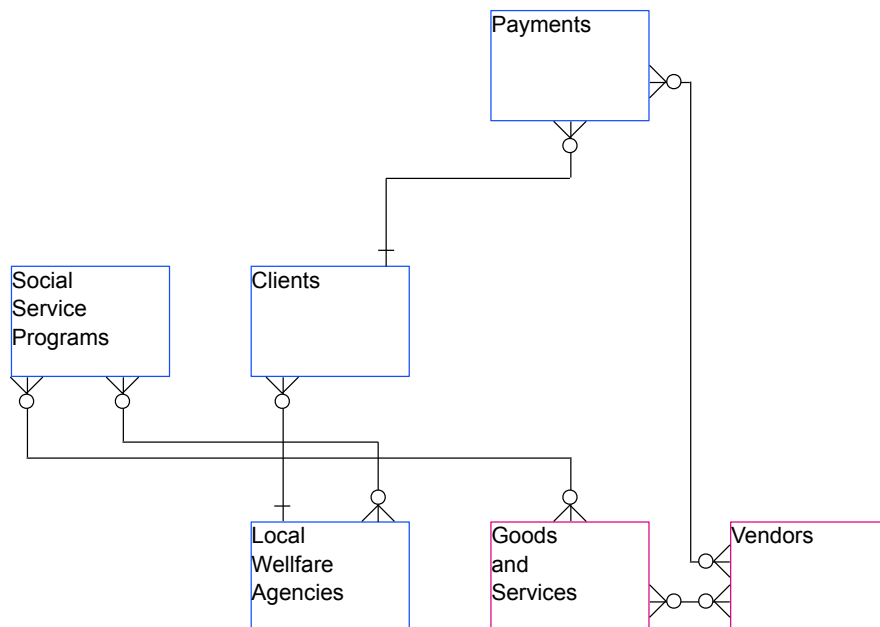


DRAFT



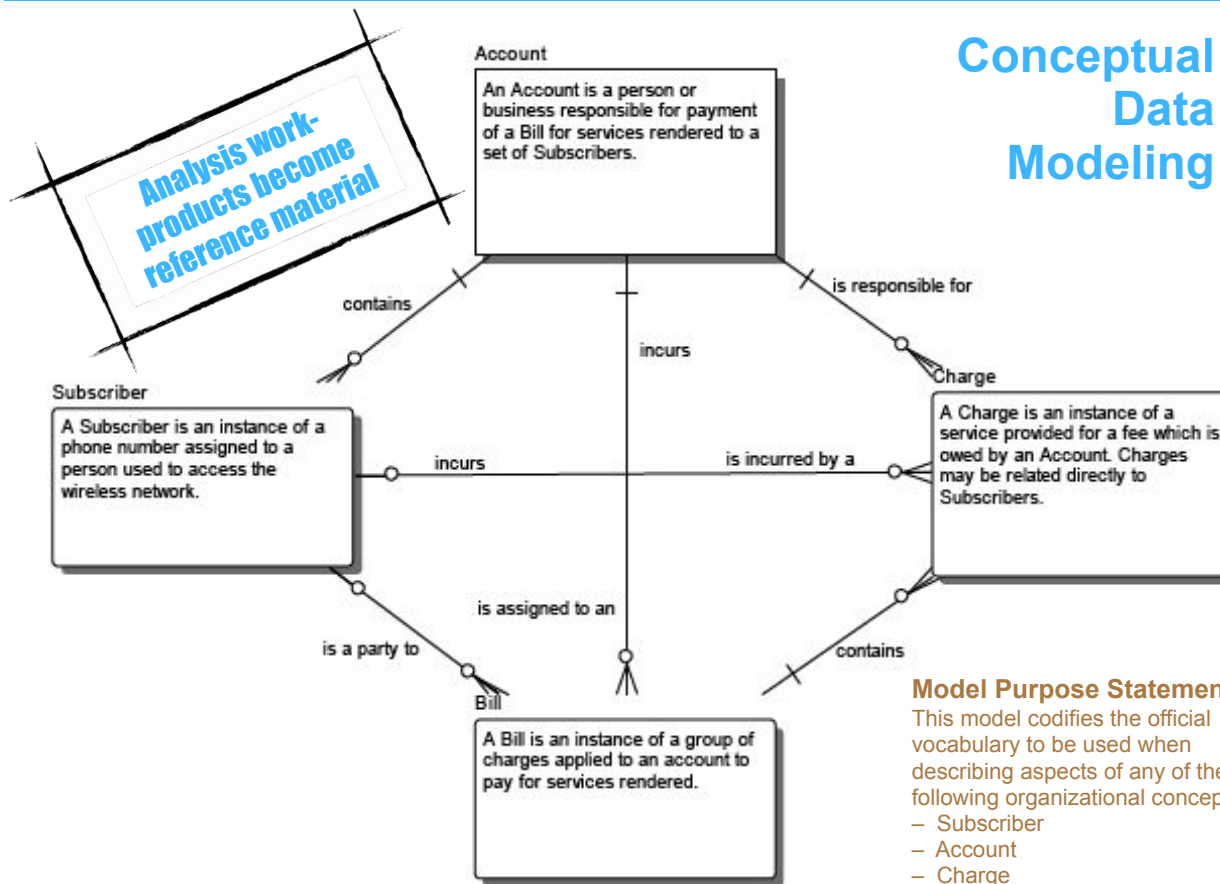
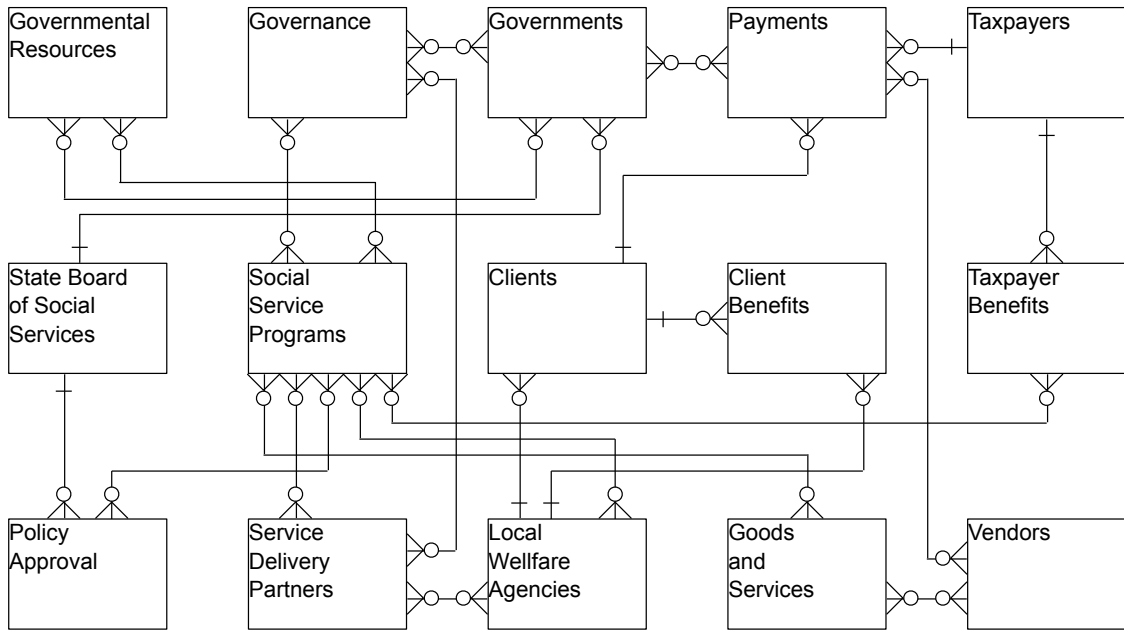
Vendor View

DRAFT



DSS Conceptual Data Model

DRAFT



GenAI to Aid Logical Data Modeling



1. Natural Language to Entity-Relationship (ER) Drafting

- The most immediate value is in Requirement Distillation. You can feed GenAI a 50-page business requirement document or a transcript from a stakeholder interview, and it can extract candidate entities, attributes, and relationships.
- **Business Value:** Reduces the "blank page" phase by 60%. It ensures that obscure business rules mentioned once in a meeting aren't lost in the transition to the logical model.
- **The Model Impact:** It generates a first-pass Mermaid or DDL script that represents the logical flow, allowing the architect to focus on normalization and business logic rather than manual entry.

2. Harmonization with Universal Patterns

- Rely on universal data model patterns to ensure scalability. GenAI is helpful mapping bespoke business requirements to industry standards.
- **How it works:** You can prompt the AI: "Align this specific retail loyalty schema with the 'Party' and 'Product' patterns from the Universal Data Model."
- **Business Value:** Prevents "siloesd" thinking ensuring data model is extensible for future business capabilities (e.g., moving from B2C to B2B) without a total redesign.



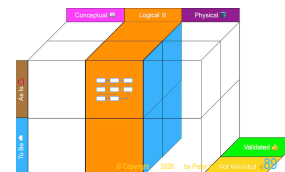
<https://anyringdata.com>

3. Automated Metadata and Semantic Enrichment

- A model is only as good as its documentation. GenAI can take cryptic column names (e.g., TXN_AMT_LC) and generate human-readable descriptions, business definitions, and even suggest data classifications (PII, Sensitive, etc.).
- **Business Value: Data Governance.** By automating the "boring" parts of modeling, you create a searchable, governed data catalog that the business can actually understand and trust.

4. Synthetic Data Generation for Model Validation

- Before we touch a physical database, we need to know if the logical design holds up under real-world scenarios. GenAI can generate high-fidelity synthetic data based on the logical constraints (keys, cardinalities, and types).
- **Business Value: De-risking the project.** You can run "What-If" scenarios through the logical model to see if it supports complex reporting needs before a single line of ETL code is written.



Program Overview

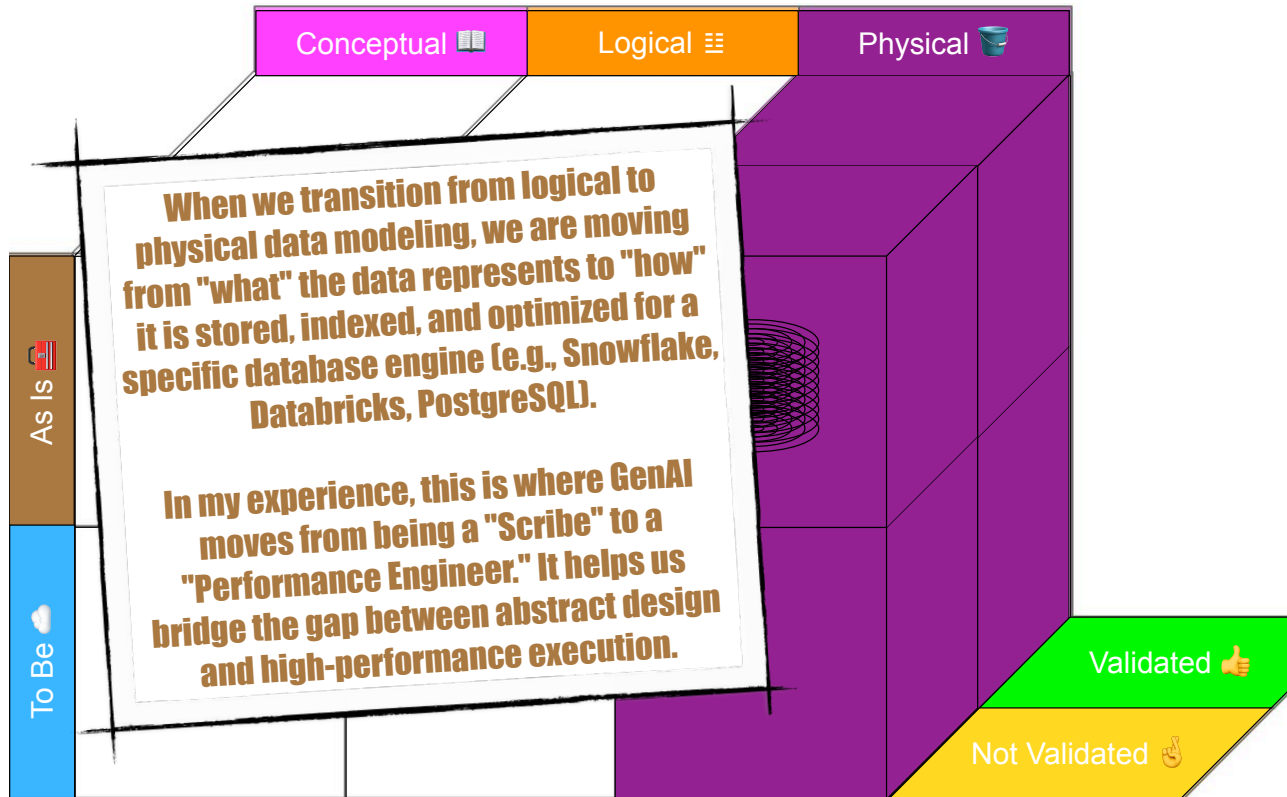
- Why model anything?
- What is data modeling designed to accomplish?
- Why (and how) should I look to AI for help?
- Overview of modeling types
- Conceptual Data Modeling
 - Motivation: Architectural tradeoffs
 - Strategy and conceptual data modeling
 - Glossary/Dictionary capabilities
- Logical Data Modeling
 - Motivation: Simplicity (Operational and Design)
 - Motivation towards standards
 - Business meets strategy
- Physical Data Modeling
 - Motivation: Required documentation and/or facts
 - Become the blueprints for physical construction of the solution
 - Blueprints are used for future maintenance of the solution
- Tying it all together/Take Aways/References/Q&A

Data Modeling Mastery: Structuring Data for AI, Analytics, or Anything Else



<https://anyringdata.com>

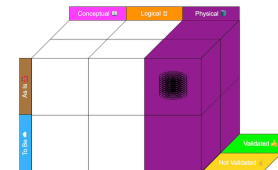
Physical Data Modeling



Physical Data Modeling

Motivation

- Documentation of specifications of production systems
 - Data flow diagrams
 - Entity-relationship diagrams
 - Dictionary/Glossary/Catalog
- Should exist if system is in production
 - Why would anyone hand craft DDL with today's tool capabilities?
- Must exist to create the system that is put into production
 - Become the blueprints for physical construction of the solution
 - Blueprints are used for future maintenance of the solution



As Is Physical Data Models (Exist too)

- This should be foundational system documentation
- Description required to access data 'in the system'
- Often can be reverse engineered, semi-automatically

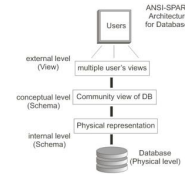
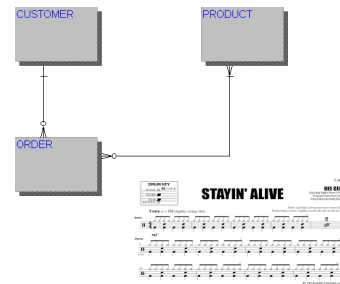
To Be Physical Data Models (Exist too)

- This is a specification of the data that can be accessed by the application
- Specification of current and future data elements to be maintained by application
- Often can be generated, semi-automatically



Data Modeling Requirements

- The process of discovering, analyzing, and scoping data requirements
 - Understand what the data things are?
 - What do they do?
 - How do they interact?
- Representing/communicating requirements in a precise form called a data model
 - Maps of critical business assets
 - Compose and contain metadata essential to data consumers
 - Function as a kind of sheet music language
 - Metadata is essential to other business functions (definitions for governance, lineage for analytics, etc.)
- The process is iterative and may include conceptual, logical, and physical models
- Modeling is done to accomplish a goal!



How Is Data Stored and Represented?

- Lists of organizational

persons
places
things

that need to be

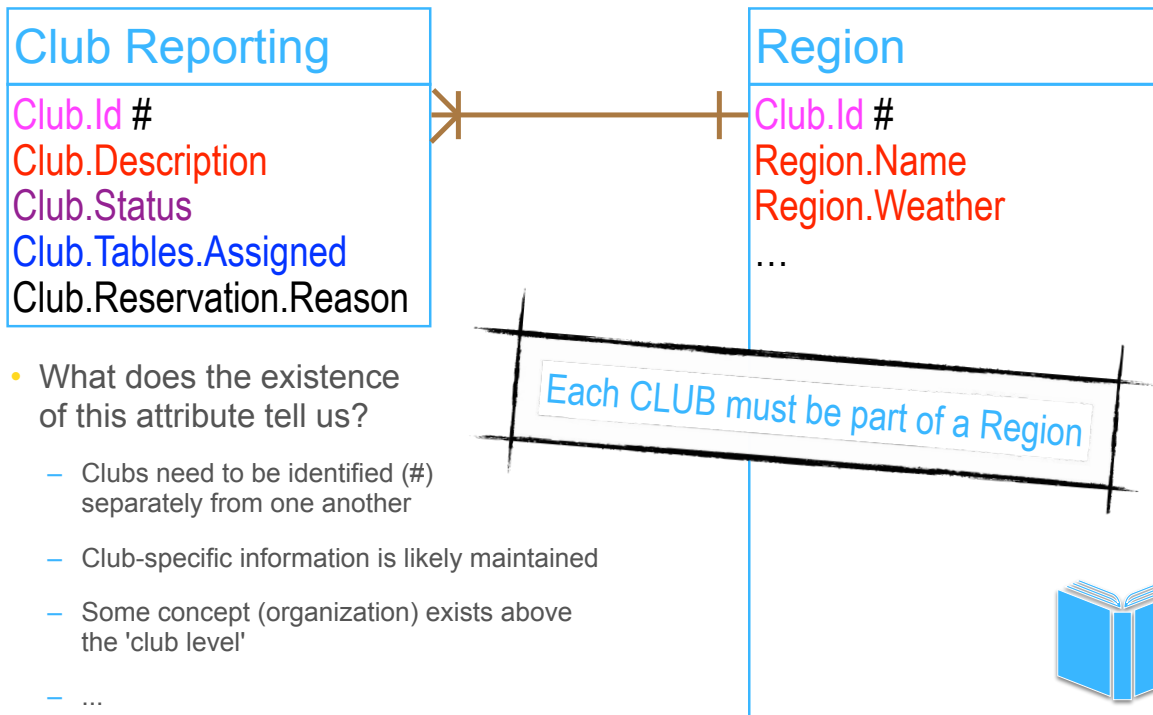
created
read
updated
deleted
archived

- These are called Attributes
 - Attributes are characteristics of "things"



Analyzing Data Attributes and Relationships

- Characteristics of CLUBS and REGIONS

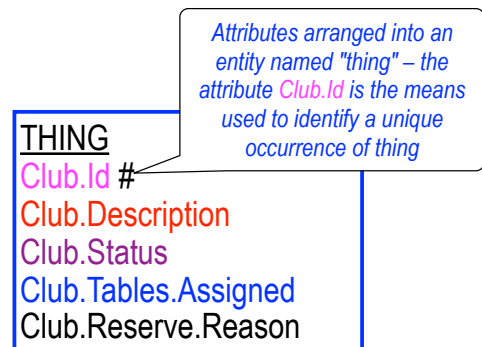


- What does the existence of this attribute tell us?
 - Clubs need to be identified (#) separately from one another
 - Club-specific information is likely maintained
 - Some concept (organization) exists above the 'club level'
 - ...



Data Modeling Uses

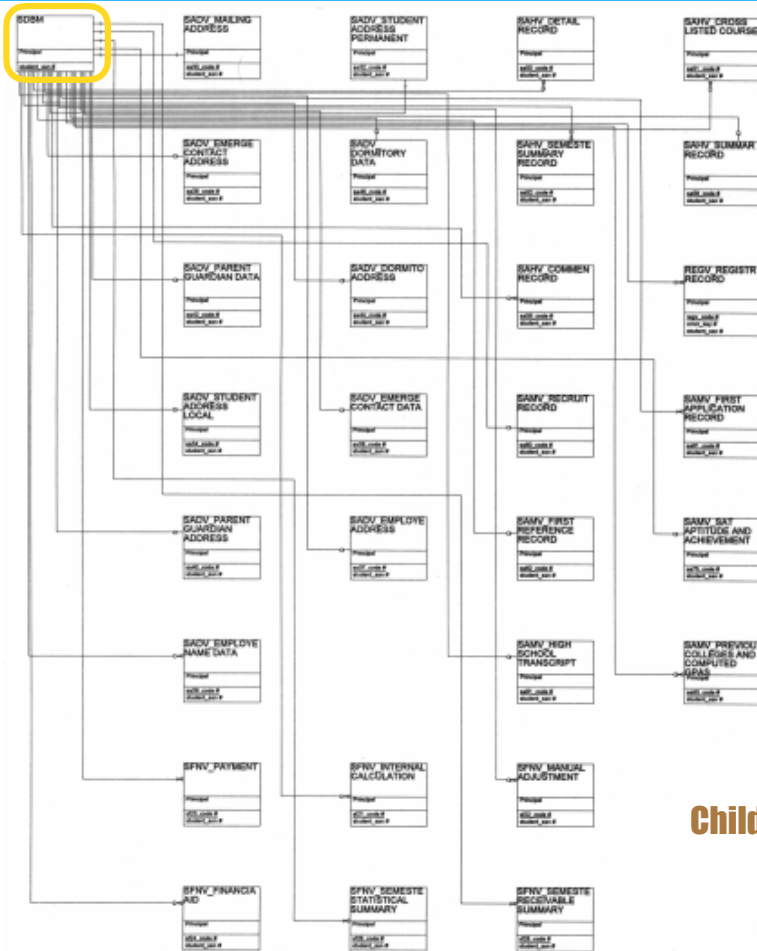
- An organization might decide to characterize the parts of a THING as:
 - Attributes: ID, description, status, Tables.Assigned, reserve.reason
- Decisions to manage information about each specific attribute has direct consequences
 - A decision to use the above data attributes permits the organization to determine if it has tables are available to be reserved
- Characteristics can be shared
 - All CLUBS may have a status
 - Many REASONS can be assigned to reservation (free text)
- Characteristics may be required to be unique
 - ID permits identification every CLUB as distinct for every other CLUB
 - Description is likely to be unique for each CLUB



Model level variances are often among additions of keys and evolving definitions—hence the mandatory glossary!

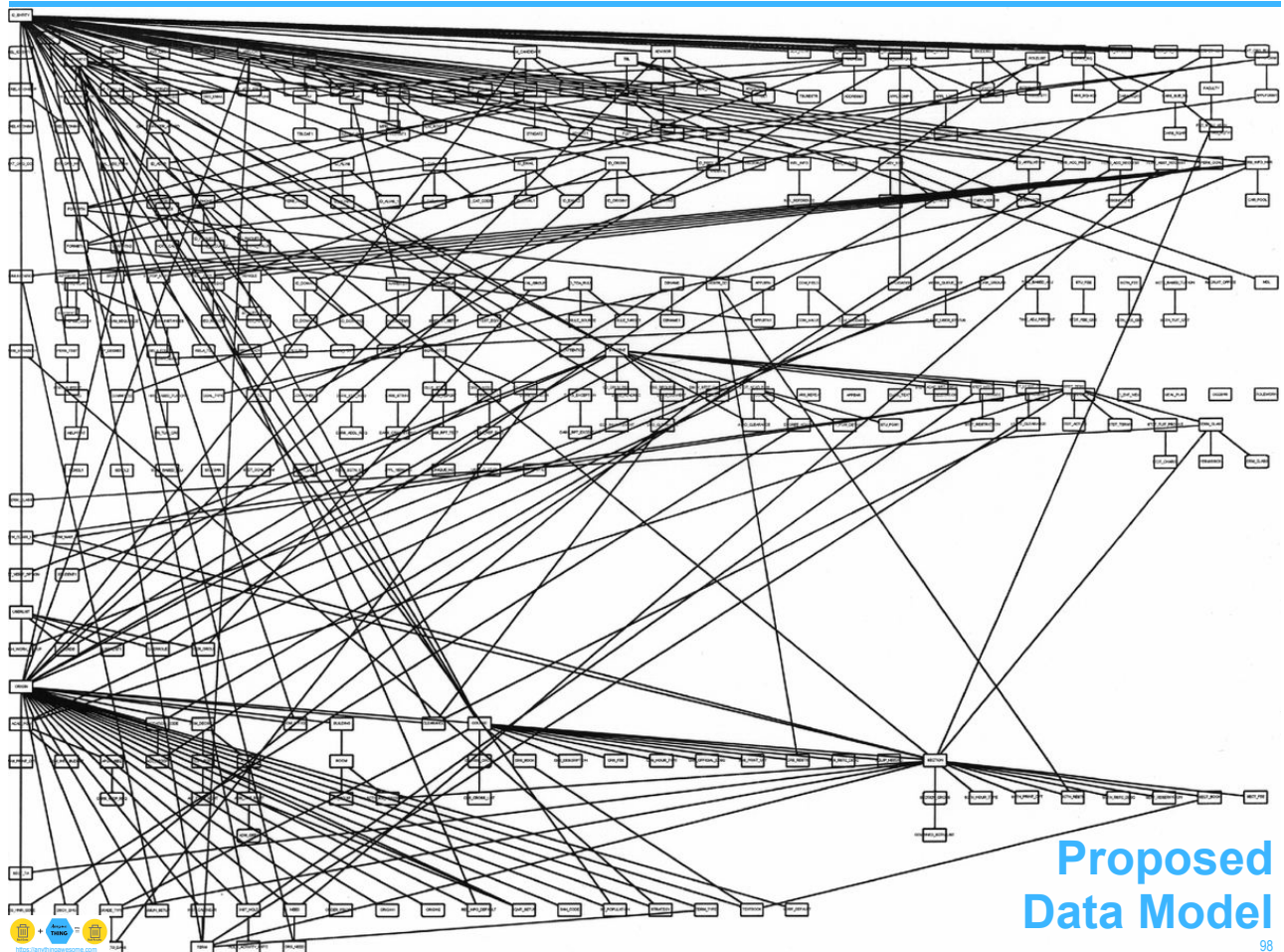


Parent



Student Data Base Master

Children



Proposed Data Model



GenAI to Aid Physical Data Modeling

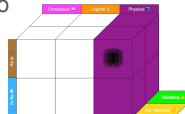


1. Platform-Specific DDL Generation

- Generating DDL (Data Definition Language) used to be a repetitive task of mapping generic types to vendor-specific ones. GenAI automates this with high precision.
- How it works: You can provide a logical entity and ask: "Convert this to Snowflake-optimized DDL. Use transient tables for staging, define clustering keys on transaction_date, and ensure all strings use VARCHAR(16777216) for flexibility."
- Business Value: Speed to Market. You can deploy complex physical schemas in minutes rather than hours, with fewer syntax errors that derail CI/CD pipelines.

2. Intelligent Indexing and Partitioning Strategies

- This is where the seasoned architect's "gut feeling" is now backed by AI analysis. GenAI can analyze query patterns (from logs) to suggest the most efficient physical storage structures.
- How it works: Feed the AI a set of representative SQL queries and ask: "Based on these join patterns, suggest a partitioning strategy for a 10TB table in BigQuery to minimize slot usage."



- Business Value: Cost Optimization. In modern cloud-data platforms, a poor partitioning or clustering strategy can lead to thousands of dollars in wasted compute. GenAI identifies these "hot spots" before they hit production.

3. Data Volume and Distribution Simulation

- Physical modeling requires understanding how data will grow. GenAI can generate Python-based data generators that populate your physical schema with billions of rows of skewed, realistic data.
- Business Value: Reliability. You can "load-test" your physical model's performance under heavy volume before the business relies on it, preventing the "Day 1 Performance Crash."

4. Automated Legacy-to-Modern Refactoring

- If you are migrating from an on-premise legacy system (like Oracle) to a modern lakehouse (like Databricks), the physical storage rules change entirely.
- The Shift: Instead of manual translation, GenAI can ingest the legacy DDL and rewrite it to leverage modern features like Delta Lake Z-Ordering or Liquid Clustering.
- Business Value: Modernization ROI. It slashes the migration timeline and ensures you aren't just "lifting and shifting" bad old habits into a new, expensive environment.



Would you like me to generate a physical DDL script for a specific database (like Snowflake or SQL Server) based on a sample logical requirement?



We'll take a logical concept—the "Sales Transaction"—and use GenAI to architect a physical model specifically for Snowflake. In this transition, we move from the "what" (a customer bought a shirt) to the "how" (storing 10 billion rows efficiently and cost-effectively).

1. The Logical Requirement

- Entity: SALES_TRANSACTION
- Attributes: Transaction ID, Date/Time, Store ID, Customer ID, Total Amount, Tax, Discount.
- Volume Expectation: 500 million rows per year.
- Primary Access Pattern: Regional managers filtering by STORE_ID and TRANSACTION_DATE for last-quarter performance.

2. The GenAI "Performance Architect" Prompt

- I would prompt the AI with specific physical constraints to ensure the output isn't just "generic SQL" but a performant physical implementation:
- "Act as a Snowflake Performance Engineer. Convert the SALES_TRANSACTION logical entity into a physical DDL.
- Use Clustering Keys on TRANSACTION_DATE and STORE_ID to optimize for range scans.
- Implement Data Masking tags for CUSTOMER_ID to comply with GDPR.
- Use Change Data Capture (CDC) metadata columns (METADATA\$INSERT_TS, METADATA\$UPDATE_TS).
- Suggest a Storage Integration pattern for loading from S3."

3. The Physical Output (Generated & Refined)

```
SQL-- Optimized Physical Schema for Snowflake
CREATE OR REPLACE TABLE FIN_DATA.SALES.TRANSACTIONS (
  -- Surrogate Key for Join Performance
  TRANSACTION_KEY    NUMBER IDENTITY(1,1) NOT NULL,
  -- Natural Keys & Foreign Keys
  TRANSACTION_ID     VARCHAR(50) NOT NULL,
  STORE_ID           NUMBER NOT NULL,
  CUSTOMER_ID       NUMBER MASKING POLICY PII_MASK,
  -- Auto-applied Governance
  -- Metrics (Using precise Decimals for Financial Accuracy)
  TOTAL_AMT_LC      NUMBER(38,2),
```

```
TAX_AMT_LC          NUMBER(38,2),
DISCOUNT_AMT_LC   NUMBER(38,2),
-- Temporal attributes
TRANSACTION_TS     TIMESTAMP_NTZ NOT NULL,
UPLOAD_TS          TIMESTAMP_NTZ DEFAULT
CURRENT_TIMESTAMP(),
-- CDC Metadata for Delta Processing
BATCH_ID           NUMBER,
OP_CODE            CHAR(1) -- I=Insert, U=Update,
D=Delete
)
```

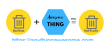
```
CLUSTER BY (DATE_TRUNC('MONTH', TRANSACTION_TS),
STORE_ID);
-- Business Value: Reduces query costs by 40% for regional monthly reporting
```

4. How this Augmentation Solves Business Challenges

- Cost Avoidance (Micro-Partitioning): By directing the AI to cluster by TRANSACTION_TS and STORE_ID, the physical model ensures that Snowflake only scans the relevant "micro-partitions." For a 10TB table, this can be the difference between a \$5.00 query and a \$500.00 query.
- Security by Design: Notice the MASKING POLICY in the DDL. GenAI can be instructed to automatically bake security into the physical layer based on logical sensitivity tags. This prevents "data leaks" before they happen.
- Operational Resilience: The inclusion of OP_CODE and BATCH_ID (suggested by the AI based on modern Data Vault or Lakehouse patterns) allows the data engineering team to restart failed loads without duplicating data.

5. Automated Physical Validation

- Once this DDL is generated, I use GenAI to write the Unit Tests for the physical layer:
- "Write a script to verify that no TRANSACTION_ID is null."
- "Generate a query to check for referential integrity between STORE_ID and the DIM_STORE table."
- Business Value: We catch "Physical Integrity" issues in the development environment (cost: pennies) instead of in the production warehouse (cost: thousands in lost productivity).



Program overview

- Why model anything?
- What is data modeling designed to accomplish?
- Why (and how) should I look to AI for help?
- Overview of modeling types
- Conceptual Data Modeling
 - Motivation: Architectural tradeoffs
 - Strategy and conceptual data modeling
 - Glossary/Dictionary capabilities
- Logical Data Modeling
 - Motivation: Simplicity (Operational and Design)
 - Motivation towards standards
 - Business meets strategy
- Physical Data Modeling
 - Motivation: Required documentation and/or facts
 - Become the blueprints for physical construction of
 - Blueprints are used for future maintenance of the solution

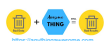
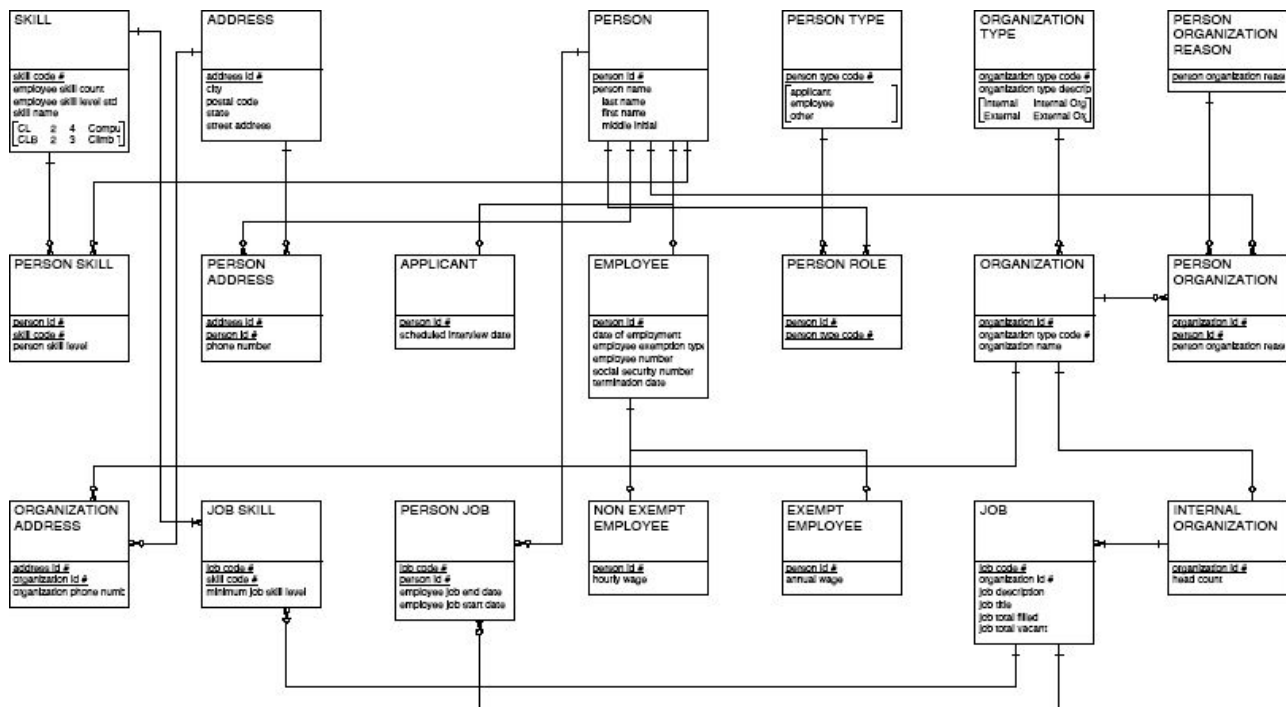
• Tying it all together/Take Aways/References/Q&A

Data Modeling Mastery: Structuring Data for AI, Analytics, or Anything Else



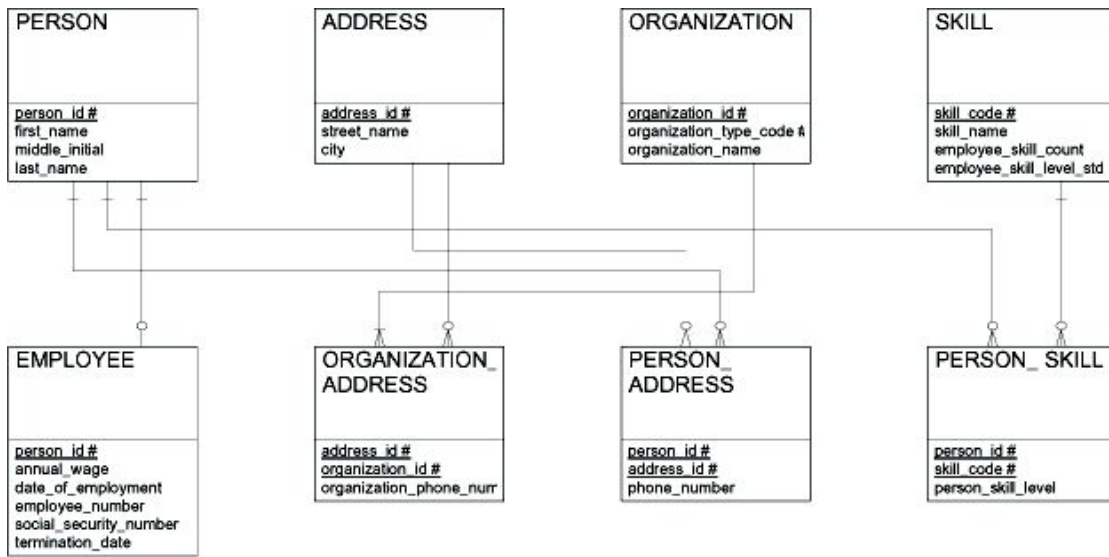
© Copyright 2020 by Peter Allen Slide # 101

HR Conceptual Model

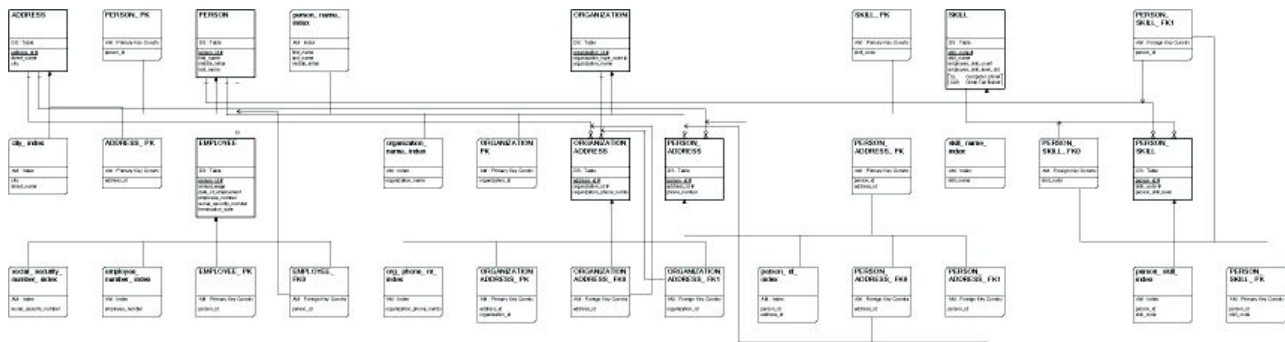


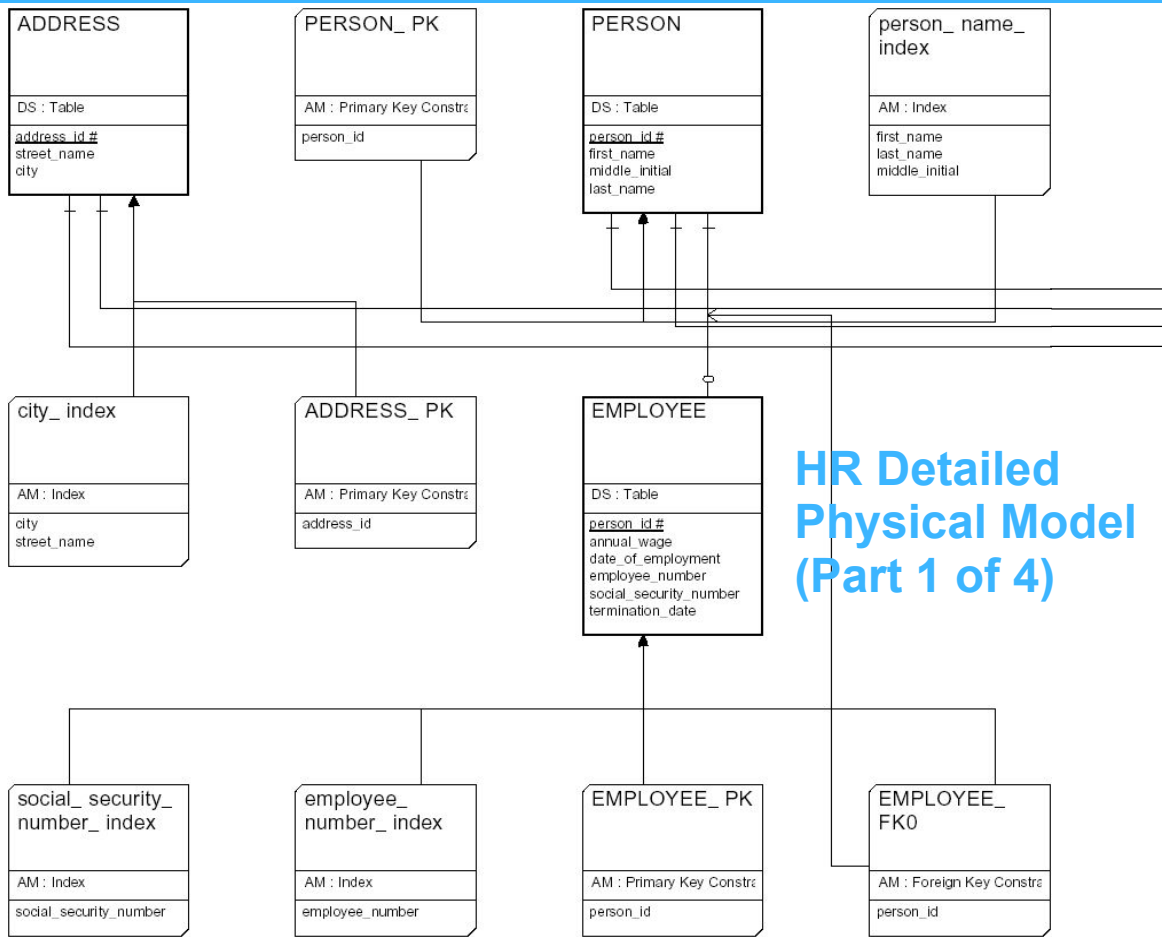
© Copyright 2020 by Peter Allen Slide # 102

HR Logical Model

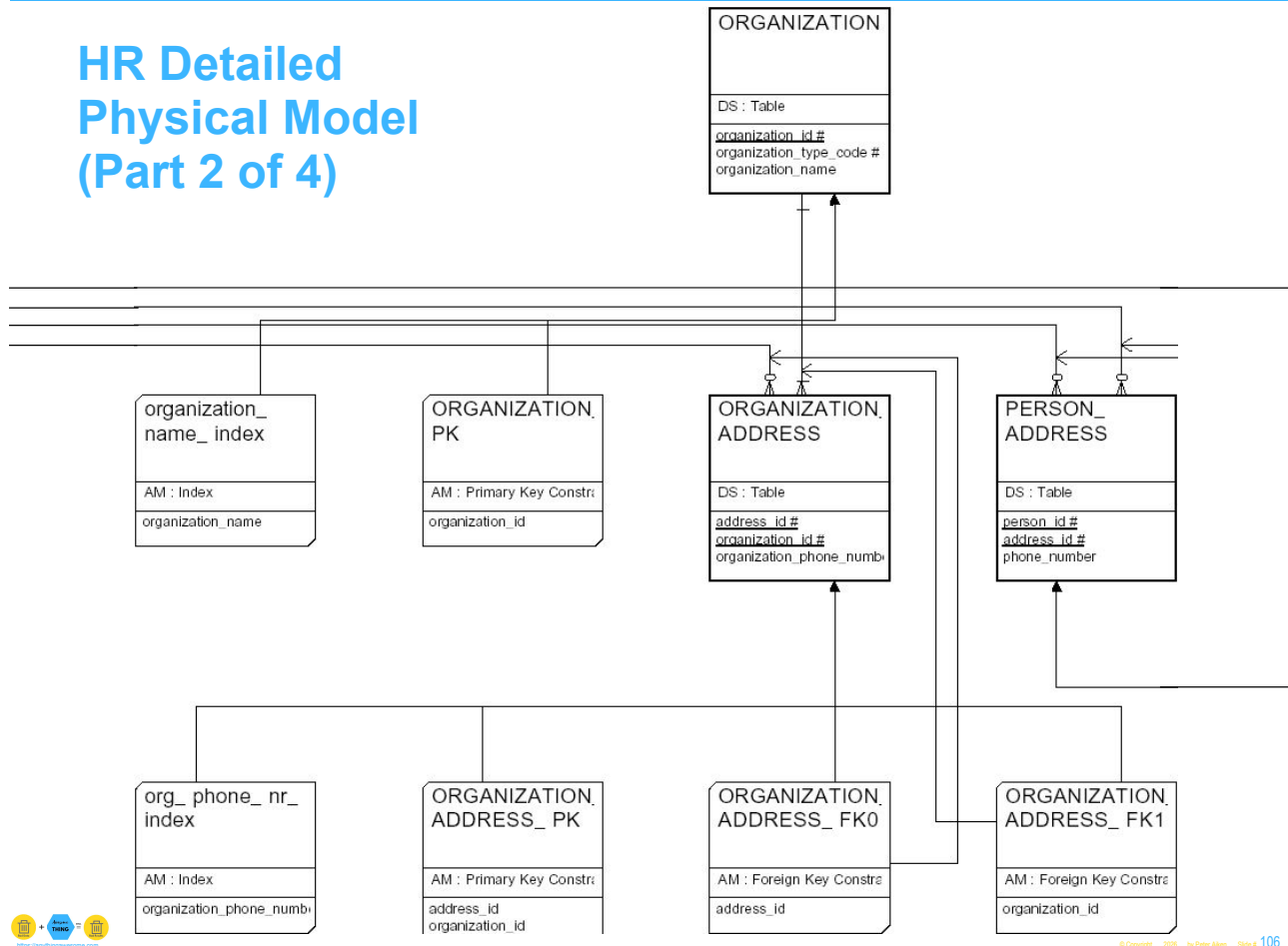


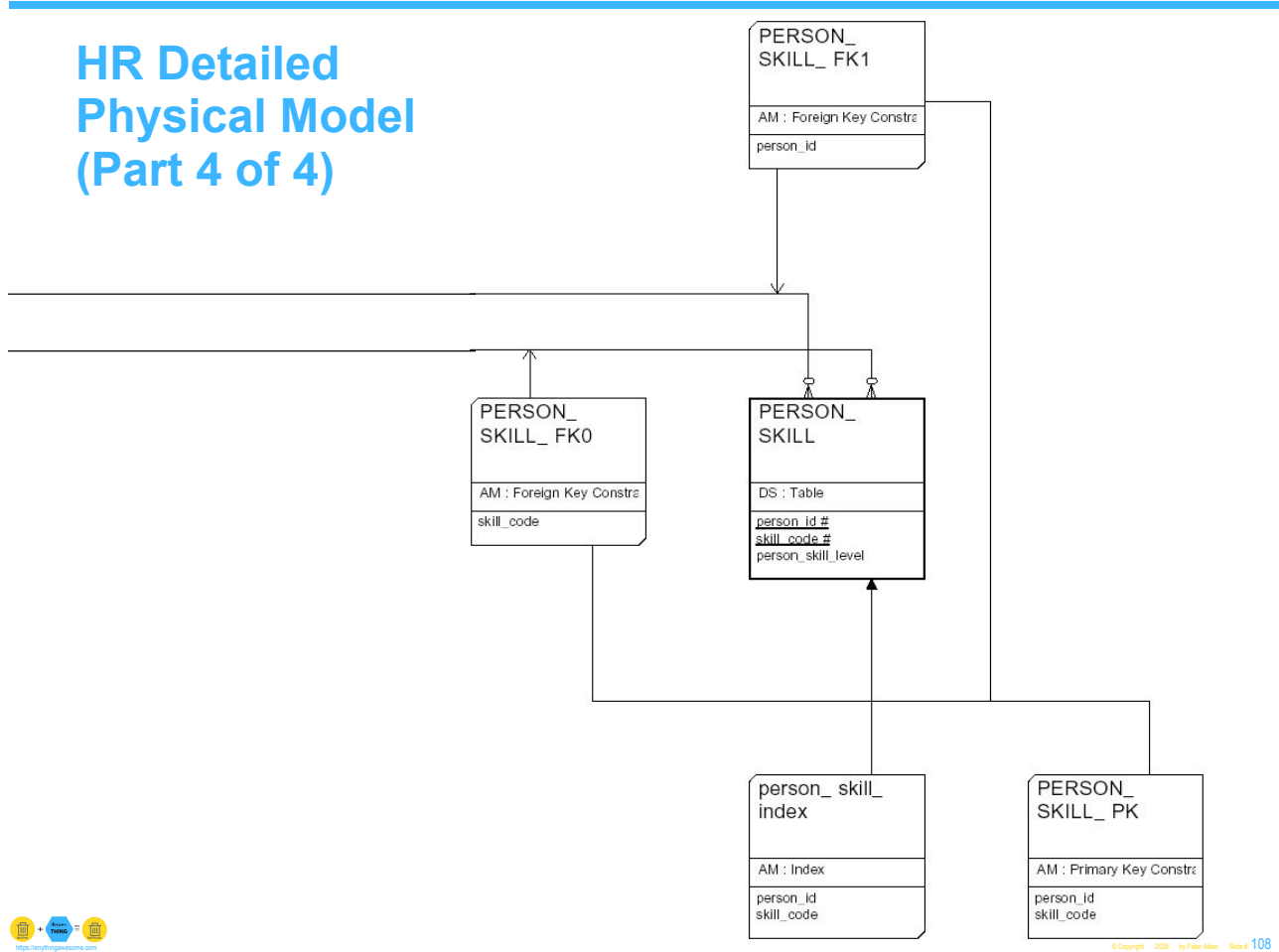
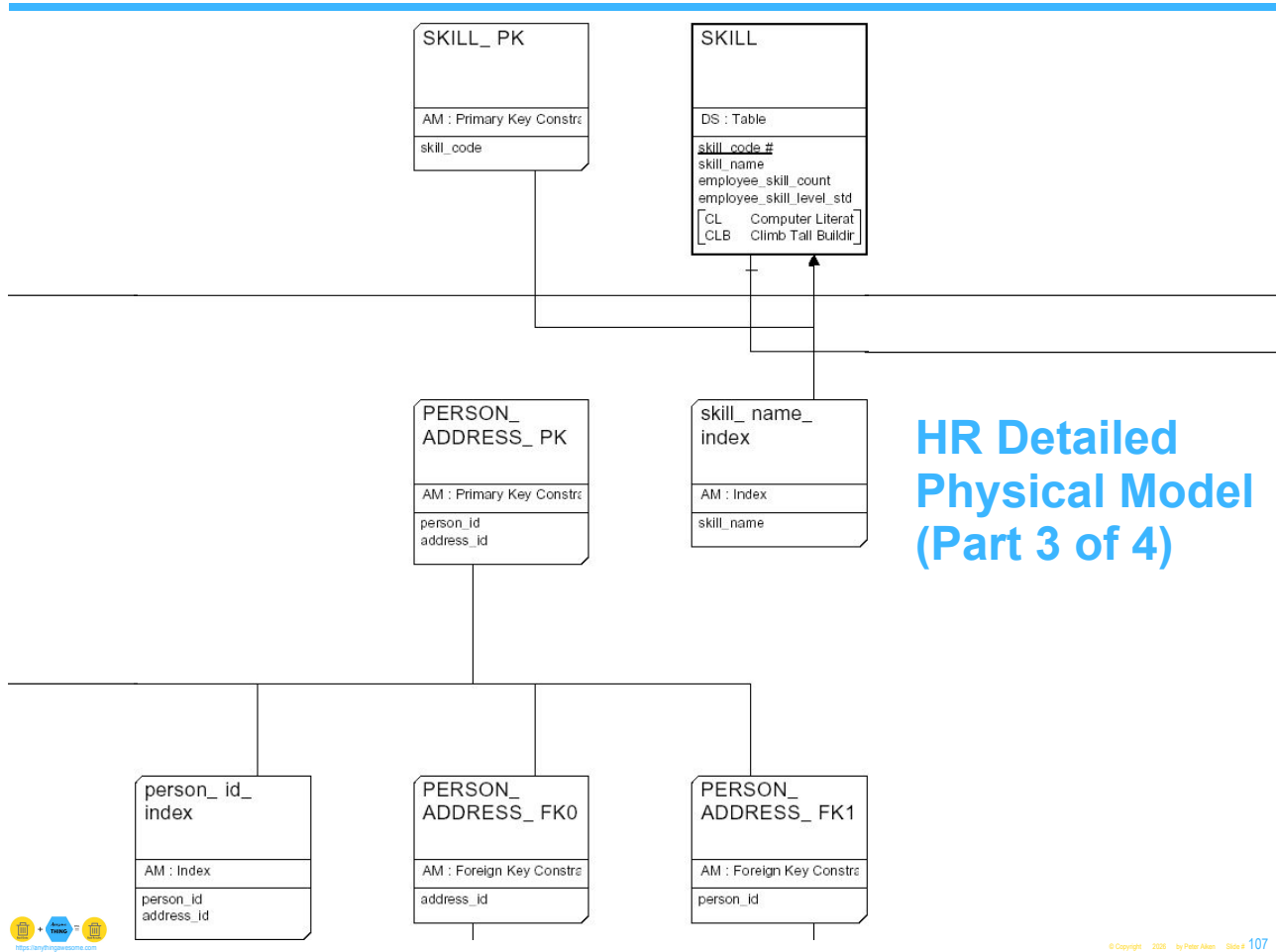
HR Detailed Physical Model Overview





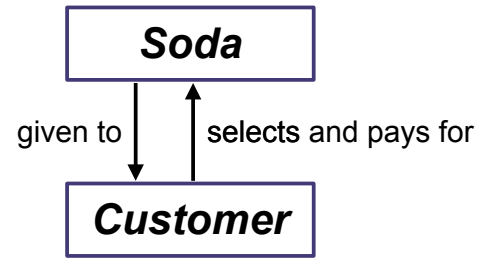
HR Detailed Physical Model (Part 2 of 4)





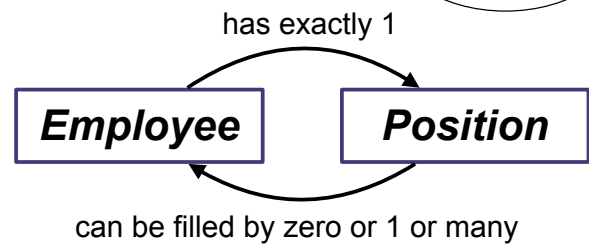
Keep Focused on the Data Model's Purpose

- The reason we are locked in this room is to:
 - Mission: *Understand formal relationship between soda and customer*
 - Outcome: Walk out the door with an as is physical and logical data model this relationship
 - Mission: *Understand the characteristics that differ between our hospital beds*
 - Outcome: We will walk out the door when we identify the top three characteristics that represent the brand with a logical data model
 - Mission: *Could our systems handle the following business rule tomorrow?*
 - "Is job-sharing permitted?"
 - Outcomes: Confirm that it is possible to staff a position with multiple employees effective tomorrow - need conceptual model for board presentation



Bed Entity: BED
 Purpose: This is a substructure within the room substructure of the facility location. It contains information about beds within rooms.
 Attributes: Bed.Description, Bed.Status, Bed.Sex.To.Be.Assigned, Bed.Reserve.Reason
 Associations: >0-+ Room
 Status: Validated

How does our perspective change: *the primary means of tracking a patient*



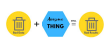
Inspired by: Karen Lopez http://www.information-management.com/newsletters/enterprise_architecture_data_model_ERP_BI-10020246-1.html?pg=2

Data Modeling for Business Value

- Goal must be shared IT/business understanding
 - No disagreements/refinements means insufficient communication
- Data sharing/exchange is automated and dependent on successful engineering/architecture
 - Requires a sound foundation of data modeling basics (the essence) on which to build technologies
- Incorporate motivation (purpose statements) in all modeling
 - Modeling is a *problem defining* as well as a *problem solving activity*
- Modeling characteristics evolve during the analysis
 - Different modeling challenges for different problems
 - Use of modeling is more important than use of a specific method
 - Models must be maintained as living documents
 - Models need to be available in an easily searchable manner
- Utility is paramount
 - Adding color and diagramming objects customizes models and allows for a more engaging and enjoyable interaction
- Value is derived from
 - Improving organizational data
 - Improving the way people use data
 - Improving peoples use of data to support strategy



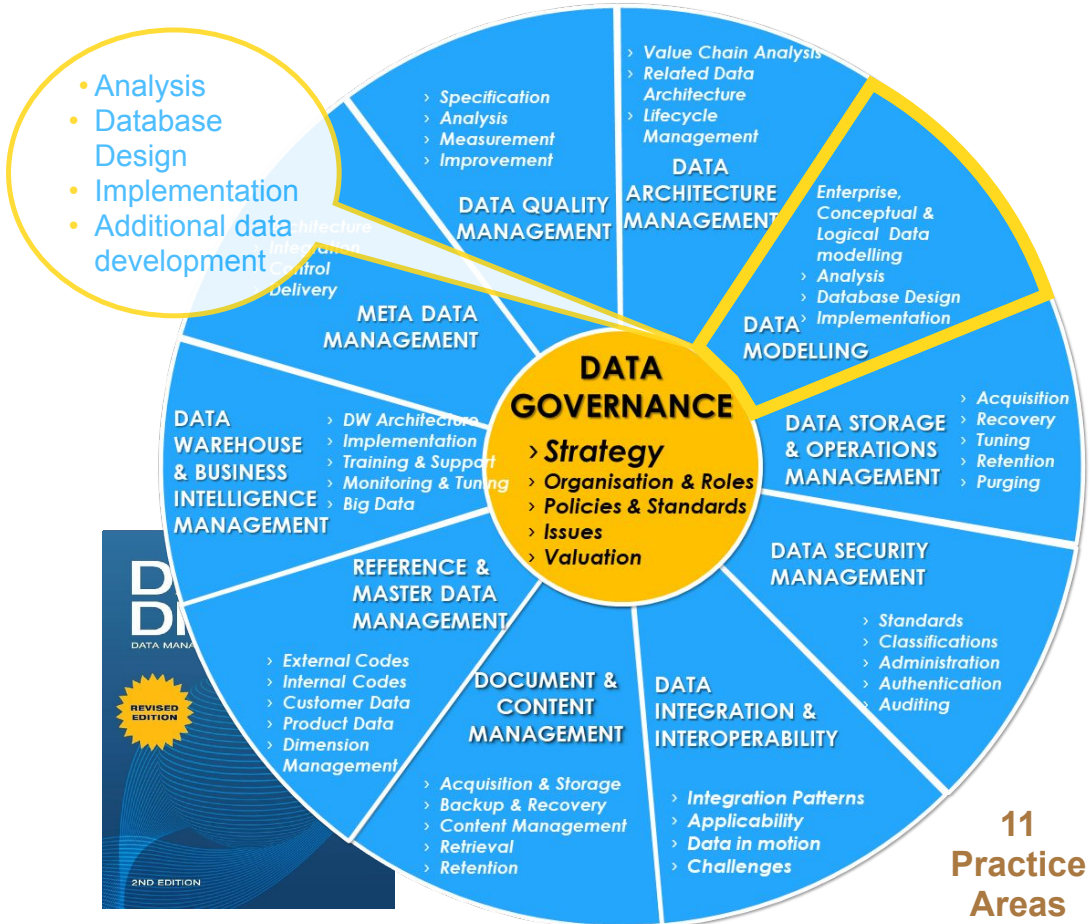
This can only be accomplished incrementally using an iterative, approach focusing on one aspect at a time and applying formal transformation methods



To Learn More



Data Management Body of Knowledge (DM BoK V2R)



11 Practice Areas



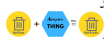
6. Works Cited / Recommended

- Ambler, Scott. *Agile Database Techniques: Effective Strategies for the Agile Software Developer*. Wiley and Sons, 2003. Print.
- Avison, David and Christine Cuthbertson. *A Management Approach to Database Applications*. McGraw-Hill Publishing Co., 2002. Print. Information systems ser.
- Blaha, Michael. *UML Database Modeling Workbook*. Technics Publications, LLC, 2013. Print.
- Brackett, Michael H. *Data Resource Design: Reality Beyond Illusion*. Technics Publications, LLC, 2012. Print.
- Brackett, Michael H. *Data Resource Integration: Understanding and Resolving a Disparate Data Resource*. Technics Publications, LLC, 2012. Print.
- Brackett, Michael H. *Data Resource Simplicity: How Organizations Choose Data Resource Success or Failure*. Technics Publications, LLC, 2011. Print.
- Bruce, Thomas A. *Designing Quality Databases with IDEFIX Information Models*. Dorset House, 1991. Print.
- Burns, Larry. *Building the Agile Database: How to Build a Successful Application Using Agile Without Sacrificing Data Management*. Technics Publications, LLC, 2011. Print.
- Carlis, John and Joseph Maguire. *Mastering Data Modeling - A User-Driven Approach*. Addison-Wesley Professional, 2000. Print.



Works Cited

- Codd, Edward F. "A Relational Model of Data for Large Shared Data Banks". *Communications of the ACM*, 13, No. 6 (June 1970).
- DAMA International. *The DAMA Dictionary of Data Management. 2nd Edition: Over 2,000 Terms Defined for IT and Business Professionals*. 2nd ed. Technics Publications, LLC, 2011. Print.
- Daoust, Norman. *UML Requirements Modeling for Business Analysis: Steps to Modeling Success*. Technics Publications, LLC, 2012. Print.
- Date, C. J. *An Introduction to Database Systems*. 8th ed. Addison-Wesley, 2003. Print.
- Date, C. J. and Hugh Darwen. *Databases, Types and the Relational Model*. 3d ed. Addison Wesley, 2006. Print.
- Date, Chris J. *The Relational Database Dictionary: A Comprehensive Glossary of Relational Terms and Concepts, with Illustrative Examples*. O'Reilly Media, 2006. Print.
- Dorsey, Paul. *Enterprise Data Modeling Using UML*. McGraw-Hill Osborne Media, 2009. Print.
- Edvinsson, Håkan and Lottie Adorinme. *Enterprise Architecture Made Simple: Using the Ready, Set, Go Approach to Achieving Information Centrality*. Technics Publications, LLC, 2013. Print.
- Fleming, Candace C. and Barbara Von Halle. *The Handbook of Relational Database Design*. Addison Wesley, 1989. Print.
- Giles, John. *The Nimble Elephant: Agile Delivery of Data Models using a Pattern-based Approach*. Technics Publications, LLC, 2012. Print.
- Golden, Charles. *Data Modeling 152 Success Secrets - 152 Most Asked Questions On Data Modeling - What You Need to Know*. Emero Publishing, 2015. Print. Success Secrets.
- Halpin, Terry, Ken Evans, Pat Hallock, and Bill McLean. *Database Modeling with Microsoft Visio for Enterprise Architects*. Morgan Kaufmann, 2003. Print. The Morgan Kaufmann Series in Data Management Systems.
- Halpin, Terry. *Information Modeling and Relational Databases*. Morgan Kaufmann, 2001. Print. The Morgan Kaufmann Series in Data Management Systems.
- Halpin, Terry. *Information Modeling and Relational Databases: From Conceptual Analysis to Logical Design*. Morgan Kaufmann, 2001. Print. The Morgan Kaufmann Series in Data Management Systems.
- Harrington, Jan L. *Relational Database Design Clearly Explained*. 2nd ed. Morgan Kaufmann, 2002. Print. The Morgan Kaufmann Series in Data Management Systems.
- Hay, David C. *Data Model Patterns: A Metadata Map*. Morgan Kaufmann, 2006. Print. The Morgan Kaufmann Series in Data Management Systems.
- Hay, David C. *Enterprise Model Patterns: Describing the World (UML Version)*. Technics Publications, LLC, 2011. Print.
- Hay, David C. *Requirements Analysis from Business Views to Architecture*. Prentice Hall, 2002. Print.
- Hay, David C. *UML and Data Modeling: A Reconciliation*. Technics Publications, LLC, 2011. Print.
- Hernandez, Michael J. *Database Design for Mere Mortals: A Hands-On Guide to Relational Database Design*. 2nd ed. Addison-Wesley Professional, 2003. Print.
- Hoberman, Steve, Donna Burbank, Chris Bradley, et al. *Data Modeling for the Business: A Handbook for Aligning the Business with IT using High-Level Data Models*. Technics Publications, LLC, 2009. Print. Take it with You Guides.
- Hoberman, Steve. *Data Model Scorecard*. Technics Publications, LLC, 2015. Print.
- Hoberman, Steve. *Data Modeling Made Simple with ERStudio Data Architect*. Technics Publications, LLC, 2013. Print.
- Hoberman, Steve. *Data Modeling Made Simple: A Practical Guide for Business and IT Professionals*. 2nd ed. Technics Publications, LLC, 2009. Print.



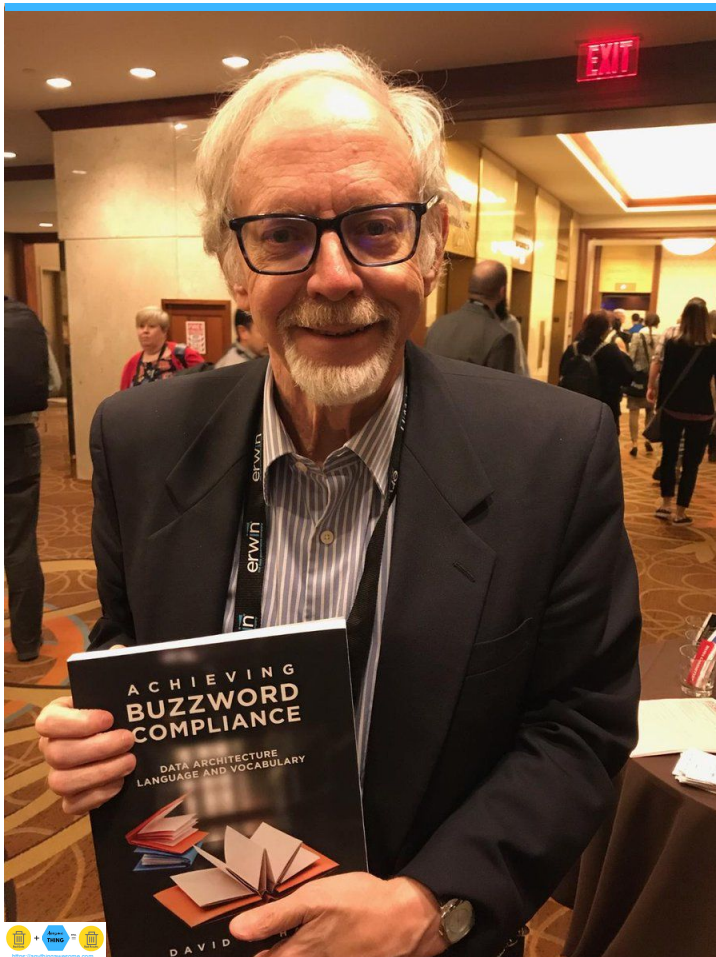
Works Cited

- Hoberman, Steve. *Data Modeling Master Class Training Manual*. 7th ed. Technics Publications, LLC, 2017. Print.
- Hoberman, Steve. *The Data Modeler's Workbook, Tools and Techniques for Analysis and Design*. Wiley, 2001. Print.
- Hoffer, Jeffrey A., Joey F. George, and Joseph S. Valacich. *Modern Systems Analysis and Design*. 7th ed. Prentice Hall, 2013. Print.
- IIBA and Kevin Brennan, ed. *A Guide to the Business Analysis Body of Knowledge (BABOK Guide)*. International Institute of Business Analysis, 2009. Print.
- Kent, William. *Data and Reality: A Timeless Perspective on Perceiving and Managing Information in Our Imprecise World*. 3d ed. Technics Publications, LLC, 2012. Print.
- Krogstie, John, Terry Halpin, and Keng Siau, eds. *Information Modeling Methods and Methodologies: Advanced Topics in Database Research*. Idea Group Publishing, 2005. Print. Advanced Topics in Database Research.
- Linstedt, Dan. *Super Charge Your Data Warehouse: Invaluable Data Modeling Rules to Implement Your Data Vault*. Amazon Digital Services, 2012. Data Warehouse Architecture Book 1.
- Muller, Robert. *J. Database Design for Smarties: Using UML for Data Modeling*. Morgan Kaufmann, 1999. Print. The Morgan Kaufmann Series in Data Management Systems.
- Needham, Doug. *Data Structure Graphs: The structure of your data has meaning*. Doug Needham Amazon Digital Services, 2015. Kindle.
- Newton, Judith J. and Daniel Wahl, eds. *Manual for Data Administration*. NIST Special Publications, 1993. Print.
- Pascal, Fabian. *Practical Issues in Database Management: A Reference for The Thinking Practitioner*. Addison-Wesley Professional, 2000. Print.
- Reingruber, Michael, C. and William W. Gregory. *The Data Modeling Handbook: A Best-Practice Approach to Building Quality Data Models*. Wiley, 1994. Print.
- Riordan, Rebecca M. *Designing Effective Database Systems*. Addison-Wesley Professional, 2005. Print.
- Rob, Peter and Carlos Coronel. *Database Systems: Design, Implementation, and Management*. 7th ed. Cengage Learning, 2006. Print.
- Schmidt, Bob. *Data Modeling for Information Professionals*. Prentice Hall, 1998. Print.
- Silverston, Len and Paul Agnew. *The Data Model Resource Book, Volume 3: Universal Patterns for Data Modeling*. Wiley, 2008. Print.
- Silverston, Len. *The Data Model Resource Book, Volume 1: A Library of Universal Data Models for All Enterprises*. Rev. ed. Wiley, 2001. Print.
- Silverston, Len. *The Data Model Resource Book, Volume 2: A Library of Data Models for Specific Industries*. Rev. ed. Wiley, 2001. Print.
- Simsion, Graeme C. and Graham C. Witt. *Data Modeling Essentials*. 3rd ed. Morgan Kaufmann, 2004. Print.
- Simsion, Graeme. *Data Modeling: Theory and Practice*. Technics Publications, LLC, 2007. Print.
- Theory, Toby et al. *Database Modeling and Design: Logical Design*, 4th ed. Morgan Kaufmann, 2010. Print. The Morgan Kaufmann Series in Data Management Systems.
- Thalheim, Bernhard. *Entity-Relationship Modeling: Foundations of Database Technology*. Springer, 2000. Print.
- Watson, Richard T. *Data Management: Databases and Organizations*. 5th ed. Wiley, 2005. Print.



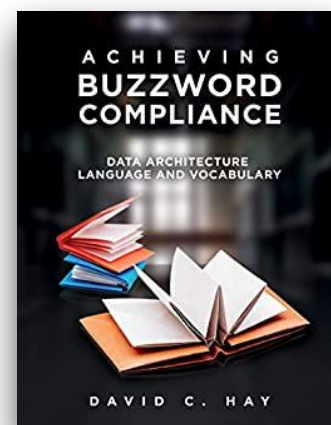
from The DAMA Guide to the Data Management Body of Knowledge 2E © 2017 by DAMA International

© Copyright 2020 by Peter Allen Slide # 117



Achieving Buzzword Compliance

Data Architecture Language and Vocabulary



amazon.com link:

https://www.amazon.com/Achieving-Buzzword-Compliance-Architecture-Vocabulary-ebook/dp/B07FG1WRSD/ref=sr_1_1?crd=2QL3ZWKU2L3VC&keywords=Achieving+Buzzword+Compliance%3A+Data+Architecture+Language+and+Vocabulary&qid=1657032460&srefix=achieving+buzzword+compliance+data+architecture+language+and+vocabulary%2CCaps%2C324&sr=8-1



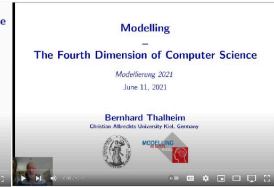
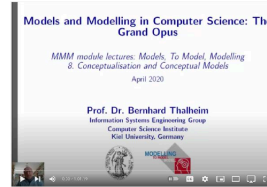
© Copyright 2020 by Peter Allen Slide # 118

Research Efforts

- Professor Bernhard Thalheim and associated research efforts have contributed much to these topics including:



- Conceptual modelling
 - https://www.youtube.com/watch?v=Y9_7KSsSUpg
 - <https://www.youtube.com/watch?v=mKcwbR6uJwU>



- Claim: logical models also conceptual models
 - <https://www.youtube.com/watch?v=L8yGjEbwTsQ>
 - <https://link.springer.com/article/10.1007/s10270-020-00836-z>
 - <https://dl.acm.org/doi/10.1007/s10270-020-00836-z>



https://www.youtube.com/watch?v=9qWjpVtr_Hg&T=2s

Advanced Data Modeling class 2016



**Conceptual
vs. Logical
vs. Physical**

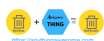


**Stages of Data
Modeling**

©Gordon C. Everest
Professor Emeritus of MIS and Database
Carlson School of Management
University of Minnesota

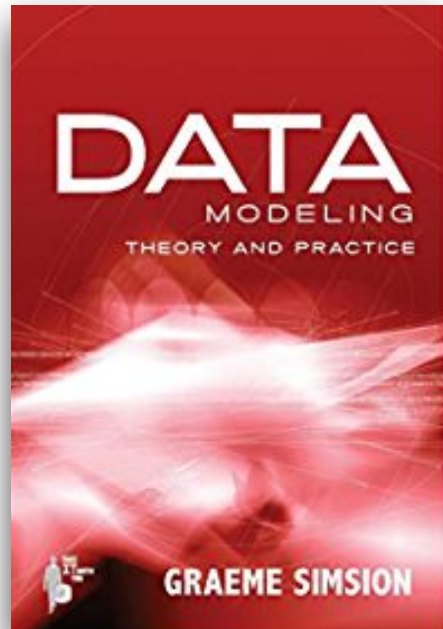
geverest@umn.edu

<http://geverest.umn.edu>

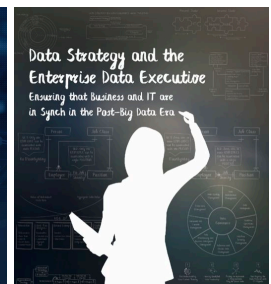
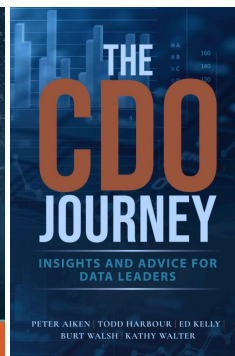
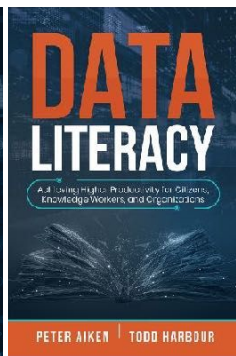
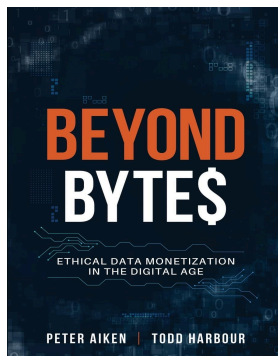


Data Modeling: Theory and Practice

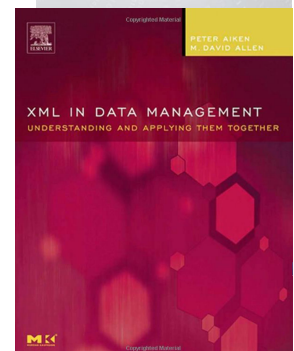
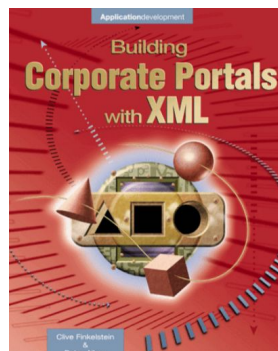
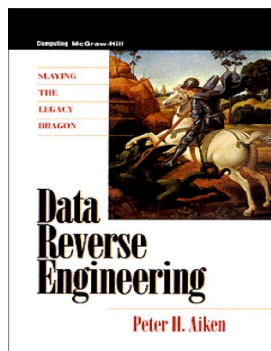
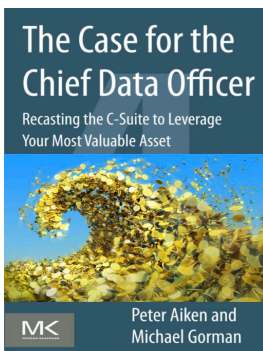
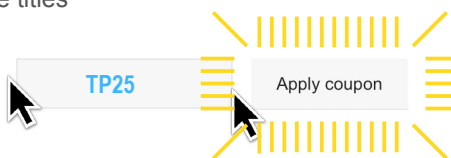
Graeme Simson



Event Pricing on Peter's Books



- 25% off directly from the publisher on some titles
- My 'Book Store' @ <https://anythingsome.com/books>
- Enter the code "TP25" at the Technics bookstore checkout where it says to "Apply Coupon"



Upcoming Events

Data Stewards: Conquering Data Debt in the Age of AI

14 April 2026



The Importance of Ethics in AI/Data Governance

12 May 2026

Data Integration Demystified: Enabling AI-Driven Interoperability

9 June 2026

Brought to you by:



Note: In this .pdf, clicking any webinar title opens the registration link

© Copyright 2026 by Peter Aiken Size 123

Collaboration?



Critical Design Review?

Executive Data Literacy Training?



+



=



Independent Verification & Validation

Peter.Aiken@AnythingAwesome.com +1.804.382.5957

Reverse Engineering Expertise?

Mentoring?

Hiring Assistance?

Thank You!

Use your data more strategically?

Tool/automation evaluation?

Book a call with Peter to discuss anything - <https://anythingawesome.com/OfficeHours.html>

