

Lecture 2: Data Models

Database and SQL: Course Notes | Brendan Shea, Ph.D. (Brendan.Shea@rctc.edu)

1 CONTENTS

2	Data models as logical structures	1
2.1	Data Modeling at Hogwarts.....	2
2.2	The Relational Model.....	3
2.3	E-R Diagrams as a “First Step”	4
2.4	Review Questions and Activities.....	4
3	Beyond the Relational Model: Objects, XML, and NoSQL.....	5
3.1	A Very Brief History of modeling for databases	5
3.2	The Object-Oriented Model.....	6
3.3	"Big" data problems and the rise of NoSQL databases	7
3.4	"Abstracting" away the details and Levels of Modeling.....	9
3.5	Review Questions	10
4	Optional Reading: Big Data is People (by Rebecca Lemov)	10

2 DATA MODELS AS LOGICAL STRUCTURES

In this lesson, we'll be answering the following questions:

1. What is a data model, and why is data modeling so important?
2. What are the basic concepts of data modeling?
3. What types of data models are there? Which are the most popular?
4. What are the basic ideas beyond the **relational** model, the **entity-relationship** data model, and the **object-oriented data model**?

Generally, every database has two components: a collection of *data* and some logical *structures* used to organize this data. Flaws in either part will compromise the database's ability to help users convert facts into information and knowledge. So, for example, the best database in the world won't yield good results if populated with false or poorly selected data. Unfortunately, this is often beyond the ability of database designers and administrators to entirely correct on their own, as data collection is often dependent on people such as business or scientific researchers. By contrast, database designers can significantly influence how the data is *structured*. A good structure can help users make the best of incomplete information and help determine how to collect new data. By contrast, a poorly designed structure can make it challenging to discover the "meaning" of our data.

The process of creating such structures is called **data modeling**. Here, a **model** is simply a simplified representation of some aspect of the world, intended to show how different "parts" fit together. Humans design models to help us understand how things "hang together." Models cannot (and should not) catalog every fact we know and are thus always incomplete. As the famous statistician George Box said, "All models

are wrong, but some are useful." It's important to note that data models are NOT pieces of computer code. Instead, a good data model should come *before* any code is written. Data modeling is also a process where early, highly abstract models evolve into more specific and **implementation-ready** models ready to be entered into a database.

2.1 DATA MODELING AT HOGWARTS

Let's suppose that Hogwarts (the fictional school attended by the characters of the *Harry Potter* novels) wishes to create a database. This database would, among other things, allow for better tracking of which students are in which classes, which instructor is teaching them, the membership of different "houses", and so on. Before deciding which RDBMS to use and entering the data into this software, they would need to construct a data model.

Basic Concepts: Entity, Attribute, Relationship. Every data model (and every database built on top of it) can be characterized by entities, attributes, and relationships.

1. An **entity** is a person, place, object, event, or other "thing.". Entities are "what exists" in the world of our model.
 - a. At Hogwarts, entities are "nouns" such as Student, Instructor, Course, Room, or House.
2. An **attribute** (also called a **field** or **column**) is a characteristic property of an entity. Attributes tell us "What sort of things" the entities in our model are.
 - a. Hogwarts's students might have attributes such as `first_name`, `last_name`, `id`, `email`, etc.
 - b. A course might have attributes such as `course_title`, `room`, `credits`, etc.
3. A **relationship** is an association between entities. We can express relationships using verbs such as "Teach," "Enroll," etc.
 - a. Hogwarts students are *enrolled* in classes, while instructors *teach* them. The relations are `Enrolled(Student, Class)` and `Teach(Instructor, Class)`.
 - b. Entity A has a **one-to-many** relationship with entity B if and only entity A can have a relationship with MANY items in B. However, each item in B can have this relationship with only ONE item in A. For example, `Teaches` is a one-to-many relationship between instructors and courses. Each instructor can teach MULTIPLE courses, but each course can only have ONE instructor (at least for this example).
 - c. Relationships can also be **many-to-one** (the same as one-to-many, with B and A reversed), **one-to-one** (just like it sounds), or **many-to-many** (with entities in A being related to an arbitrary number of items in B and vice versa). So, for example, `Enrolled` is a many-to-many relationship between Student and Courses: each student can be enrolled in multiple courses, and each course might have numerous students.
 - d. In the relational model, most/all relationships should be one-to-many. If there is a one-to-one relationship between A and B, then *either* A or B should get a table, and the other should become an attribute of that table (otherwise, we'd be wasting space!). The relational model can't "naturally" handle many-to-many relationships between A and B. Instead, you'll need to create a **link table (or "join table")** C that is "between" A and B, and serves to relate entities in A to entities in B. In the Hogwarts example below, this is the "Enrolled" table.

Business Rules as Constraints on Data Modeling. To decide precisely *which* entities, attributes, and relations to include in the Hogwarts data model, the database designers need to know the **business rules**. These are precise and unambiguous statements of policies and procedures that determine the different parts of the business (or scientific research project, video game, etc.) relate to one another. The rules can sometimes be found by reading company handbooks or talking to administrators (in this case, Dumbledore). Still, they will often require looking at "how things are done" and formulating this as a business rule. Some simple business rules for Hogwarts might be "Students take classes, while Teachers receive salaries" or "A House can have many members, but a student can only belong to one House." Constructing data models

from business rules involves *descriptive* elements (about the organization's current way of organizing data). However, it also requires "*prescriptive*" decisions about systematizing unorganized data.

Question: Think of a purpose for which you might want to make a database. What are some "business rules" you'd like to consider? What are some entities that you might want to include in a database? What are the attributes of these entities? The relationships between them?

2.2 THE RELATIONAL MODEL

Most modern DBMSs are based on the relational model, and we'll focus most of our attention on this model. Here are the basic ideas of this model:

1. Each entity type is represented by a **relation** (or **table**). These "relations" can be visualized as a 2-dimensional collection of "columns" and "rows." The **rows** of this table correspond to specific entities (people, bank accounts, whatever) about which we have stored data. Meanwhile, the columns represent the values of the attributes. A database **record** (one row) describes one entity and all of its features.
2. The tables are related to one another via their attributes. So, for example, the Student and Teacher tables may both contain a course_id field, which in turn references the id attribute in the Course table. These shared attributes allow us to answer questions like "Which students are enrolled in Professor Snape's class?"
3. One-to-one relationships and many-to-one relationships can be captured using attributes. Many-to-many relations will need their own table.

Again, don't worry about the details at this point. The point is just to understand how the relational model "looks."

SELECT * FROM Student;						
id	last_name	first_name	dob	year_in_school	Email	house_id
1	Diggory	Cedric	5/25/1978	5	diggory.dog@gmail.com	2
2	Chang	Cho	3/11/1979	4	Chang23@hogwarts.edu	3
3	Potter	Harry	7/31/1980	3	harry.potter@gmail.com	1
4	Granger	Hermione	9/19/1979	3	hgrangerr@hogwarts.edu	1
5	Malfoy	Draco	10/31/1979	3	DracoM79@hogwarts.edu	4

Here, we have stored information about the students' id numbers, names, email addresses, and houses. "id" is the **primary key** of this table, which means (among other things) that each row in the table must have a *unique* id, since we must be able to "pick this row out" using only the primary key. The attribute house_id on the student table, is a **foreign key**, that serves to *relate* this table to the House table, by referring to *its* primary key (also called "id"):

SELECT * FROM House;		
id	name	History
1	Gryffindor	Gryffindor was one of the four Houses of Hogwarts School of Witchcraft and Wizardry and was founded by Godric Gryffindor.
2	Hufflepuff	Hufflepuff was founded by Helga Hufflepuff, and is the most inclusive among the four houses, valuing hard work, dedication, patience, loyalty, and fair play rather than a particular aptitude in its members.
3	Ravenclaw	Ravenclaw was founded by Rowena Ravenclaw. Members of this house were characterized by their wit, learning, and wisdom.
4	Slytherin	Slytherin was founded by Salazar Slytherin. House characteristics included cunning, resourcefulness, leadership, and ambition.

Finally, we have tables for Course and Enrolled (I've left the Instructor table out). Enrolled is a **linking table** (or **bridge entity**) between Course and Student.

SELECT * FROM Course;			SELECT * FROM Enrolled;	
id	Credits	Name	course_id	student_id
1	3	Potions	1	1

2	3	History of Magic		3	1
3	4	Transmutation		1	2
4	4	Defense Against Dark Arts		2	2

Question: Which students are enrolled in which classes?

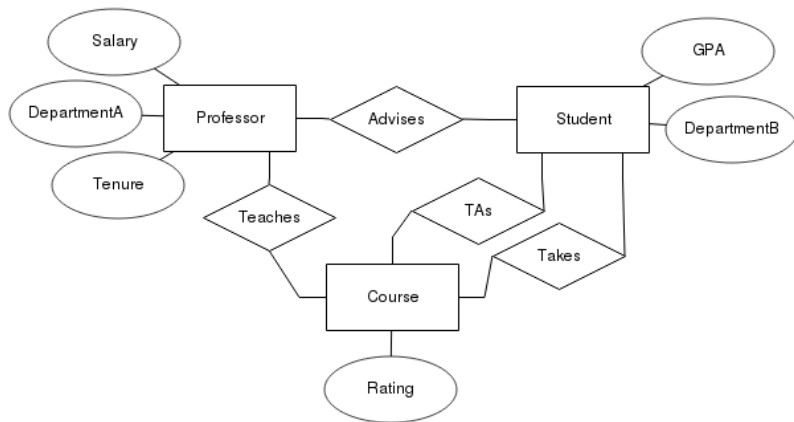
2.3 E-R DIAGRAMS AS A “FIRST STEP”

The **Entity-Relationship Model (ERM)** is a standard model used for high-level design and is often used as a *precursor* to translating into other models (such as the relational model). It generally involves the use of visual diagrams (called **Entity-Relation Diagrams, or ERDs**), with the symbols as follows:

1. Entities (actually, types of entities) are expressed by rectangles/squares, while circles represent attributes of these entities.
2. Diamonds represent relationships (which connect two or more entities).
3. Lines of varying styles **connect** entities, depending on their relationship (one-to-one, many-to-one, or many-to-many).

This is called a **Chen-style** ERD. Later, we'll learn about some other common styles of ERDs.

Here is a simple Chen E-R diagram from a paper by Hayes et. al¹:



Here the entities are Professor, Course, and Student. Each entity has associated attributes. For example, a Professor's attributes include tenure (do they have it or not?), the department they are part of, and their salary. Finally, the entities bear different relationships to one another. Students, for example, might *take* a class or *ta* (serve as "teaching assistant") for that class. Professors, by contrast, can *teach* classes. This diagram does NOT contain the lines necessary to express different kinds of relationships.

Think of one attribute, entity, or relationship you might add to the above diagram. How could you add it?

2.4 REVIEW QUESTIONS AND ACTIVITIES

1. Define the following concepts in your own words: **data model**, **business rule**, **entity**, and **relationship**. Give an example of each.
2. What are the essential components of the **relational database model**?
3. Business rules are clear, short, and unambiguous sentences describing data structure. What are THREE possible business rules for a school data model (such as the Hogwarts example).

¹Hayes, Alexander & Das, Mayukh & Odom, Phillip & Natarajan, Sriraam. (2017). User Friendly Automatic Construction of Background Knowledge: Mode Construction from ER Diagrams. 1-8. 10.1145/3148011.3148027.

4. Suppose you were designing a database containing music information (for example, about different songs, artists, albums, etc.). Do the following::
 - a. Identify at least THREE different entities about which you'd like to store data.
 - b. Identify at least TWO attributes for each of your entities.
 - c. Describe at least TWO relationships between these entities (bonus: what type of relationships are these?).
 - d. Draw an E-R diagram based on your data model
5. Describe the difference between the following types of relationships. Then, give an original example of entities X and Y that fulfill them.
 - a. A one-to-one relationship between X and Y.
 - b. A many-to-one relationship between X and Y.
 - c. A many-to-many relationship between X and Y.

3 BEYOND THE RELATIONAL MODEL: OBJECTS, XML, AND NoSQL

In this lesson, we'll be answering the following questions

1. Which sorts of data models are most widely used in database development? How has this changed over time?
2. What are the basic concepts of the **Object-Oriented** data model?
3. What is **XML**? What is an **XML database**?
4. What is **"Big Data"**? How do **NoSQL** databases relate to it?

3.1 A VERY BRIEF HISTORY OF MODELING FOR DATABASES


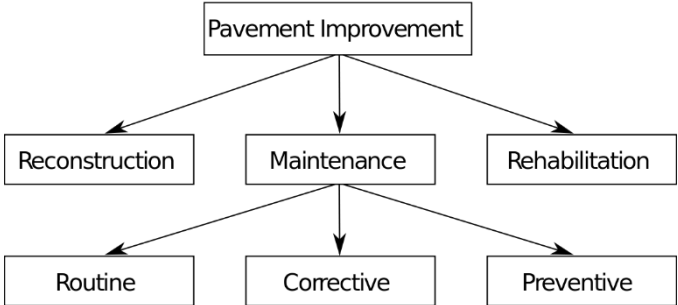
Ideas about how best to model data for databases have changed over the years. These changes were partly driven by changes to computer hardware, since faster hardware served to make different sorts of models realistic, and in part by changes in computer use (modeling web pages is different from modeling banking data). They also reflect continued research on the underlying algorithms and data structures. The main models were as follows:

1. **As you'll recall from previous lessons, a "file" is a collection of records chosen for some purpose.** File system organization of records was common well into the 1960s, even as companies increasingly began to computerize their records. This method of organization is highly inefficient for many purposes.
2. In the 1960s, **hierarchal** (tree-like) and **network** (graph-like) data modeling focused on *linking* different data types to other types. It required that users have relatively extensive knowledge of how each data item related to all the others. (So, even small additions, deletions, or changes could cause large disruptions for users and programmers).
3. The 1970s saw the rise of **the "Relational" and "Entity-Relationship" data models**. The database software (and the SQL language) designed for this model are the basis for most modern databases.
4. The **object-oriented** data model allowed users to define complex "objects" that bundled data with methods that acted on this data. The **"Object-Relation" data model** (implemented by many modern DBMSs) combines the Object model with the Relational Model. He
5. The explosion of web content in the 1990s led to the widespread use of the text-based markup language **eXtensible Markup Language (XML)** in storing, accessing, and exchanging this sort of

semi-structured data. **XML** databases and **XML Hybrid Databases** (Postgres, MS SQL Server, Oracle) incorporate XML with the relational model.

6. The increasing importance of “big data” in the 21st century (and its analysis by corporations such as Google, Facebook, and Amazon) has led to **NoSQL** databases that are NOT based on the relational model. Many (though not all) of these databases deal with cloud-scale operations dealing with semi-structured data, such as that stored in JSON format. As with XML, relational databases have adapted to this by providing increased support for JSON.

For several reasons, the relational model became something like the "standard" data model starting in the 1970s. Among other things, it (1) made users' access to data **structurally independent** from database design, (2) provided easy support for transactions, (3) made resolving "ad hoc" queries about different tables relatively straightforward, and (4) minimized data redundancy and inconstancy. However, other data models work better for specific specialized uses.'

<pre><?xml version="1.0"?> <quiz> <qanda seq="1"> <question> Who was the forty-second president of the U.S.A.? </question> <answer> William Jefferson Clinton </answer> </qanda> <!-- Note: We need to add more questions later.--> </quiz></pre> 	<h3>Hierarchical Model</h3>  <pre> graph TD A[Pavement Improvement] --> B[Reconstruction] A --> C[Maintenance] A --> D[Rehabilitation] C --> E[Routine] C --> F[Corrective] C --> G[Preventive] </pre>
<p>Organizations can use XML to store things like "a question and answer" data structure. (Image source: Wikipedia)</p>	<p>The hierarchal database model requires we navigate to data items by knowing their path. For example, “Pavement Improvements: Maintenance: Preventative.” (Image source: Wikipedia)</p>

3.2 THE OBJECT-ORIENTED MODEL

Object-oriented databases rely on the same basic model as object-oriented programming languages, such as C++ or Java. Where the relational model represents the world in terms of internally SIMPLE entities ("tables" or "relations") that have attributes, the object-oriented sees the world as fundamentally made up of **objects** with "complex" internal structures that combine data with methods acting on this data. In comparison to the relational model, the object-oriented model attempts to capture more of the **semantics** (or “meaning”) of what exactly these objects are. The main concepts of the model are as follows:

1. A **class** is an **abstract** object (that is, classes don't exist without instances!). So, for example, we might have a class for "Student." The concrete, "instantiated objects" are the individual students (Hermione, Ginny, Harry, etc.) who are the **instances** of this class. The individual student-objects **inherit** all the properties associated with the class Student.
2. Each object **encapsulates** (or "contains") both the *data* associated with it (student names, home address, email address, etc.) as well as the "methods" that operate on this data. For example, `HarryPotter.enroll(course_id)` would enroll Harry in a particular class.
3. One powerful feature of the OO approach is its support for **Polymorphism**, which allows us to access the data and methods contained in an object even if we don't know the object's most precise "class." So, for example, if both the Student and Professor classes are sub-classes of the more general Person class, we might be able to call methods like `getBirthDate()` on both Students and Professors.

Object-Oriented Model

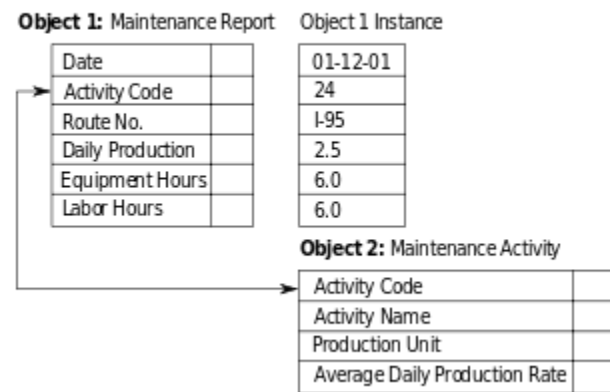


Figure 1 From wikipedia.com. Here "Object 1" and "Object 2" are classes, while "Object 1 instance" is an object!

Don't worry if this seems a bit confusing! Getting the hang of the object-oriented model usually requires spending a fair amount of time programming in a language like C++ or Java. In the context of databases, the most critical aspect of this model is its robust support for *defining* new types of data, and supporting the creation of methods that *act* on it. The main application of pure **object databases** thus tends to come in domains where we need to interact repeatedly with "complex" sorts of objects. Examples include some areas of science (physics, astronomy, engineering) or specialized **real-time** systems (such as air traffic control systems).

3.3 "BIG" DATA PROBLEMS AND THE RISE OF NOSQL DATABASES

Recent technological developments—including digital cameras, smart speakers, mobile phones, and countless Internet of Things (IoT) devices—have made it much easier to record lots of data. Simultaneously, the cost of *storing* such data (on hard drives or solid-state drives) has decreased. In fact, we currently produce and store more data *every day* than all of human history up to the computer age. However, human intellectual capacity (and even our computer algorithms) have NOT advanced at the same rate. This has led to the study of "**Big Data**," as companies, governments, and researchers struggle to make sense of the data.

The Three 'Vs'. While there is no hard-and-fast distinction between "Big" data and ordinary (small?) data, many scholars define Big Data in terms of velocity, volume, and variety:

- **Velocity** is the speed with which data collects and changes. For example, social media companies such as Facebook must deal with billions of simultaneous users interacting with each other and generating new data almost continuously. For better or worse, this is a much *faster* way of developing social interaction data than was possible when people communicated using speech or sent letters through the mail.
- **Volume** is the sheer amount of data that must be analyzed. Companies such as Google, for example, take in enormous amounts of data. They must then process this to deliver search results, target advertisements, etc.
- **Variety** refers to modern devices producing many different *types* of data, which must somehow be "reconciled" to produce usable results. For example, Artificial Intelligence agents such as IBM's Watson (which beat humans on "Jeopardy") needed to learn to "understand" how the many, many

textual sources it processed "related" to one another, even when the system's creators did not explicitly encode this. This problem of reconciling different data sources becomes even more challenging when we attempt to incorporate things such as images, sensor data, etc.

Big Data presents several significant challenges for traditional relational databases. Among the most important are:

1. **The data is often unstructured and “sparse.”** Each record on a table in a relational database contains the *same* attributes. So, for example, one record of a bank account table looks much like the other (though with different values for different customer name, balance, etc.). By contrast, "Big" data gathered from social media sites, sensors, etc., are often wildly diverse. One social media user, for example, might have many photos but few posts, while another might have the opposite. In technical terms, the data is **sparse**, with most items having a value of "null" for most attributes. Relational databases do not deal well with this.
2. **Expanding relational databases is expensive.** Adding millions of new unstructured data items to a relational database is costly (in terms of storage, processing power, etc.). Depending on a user's needs, it is also wasteful, if they don't *need* the full suite of capabilities (such as “ad hoc” queries using SQL) that a relational database offers.
3. **Data analysis requires new tools.** The sorts of analytic tools built into SQL were designed for structured data. Large amounts of unstructured data can be better analyzed using other means.

NoSQL ("Not only SQL" or “noSQL”) databases represent *various* ideas about how to respond to the challenges of Big Data. Many take advantage of **Hadoop** or similar frameworks for “distributing” the storage and processing of data across many machines. They are, in general, designed to have high scalability (easily add more capacity to meet demand), high availability (users don't have to wait), and fault tolerance (the failure of a few machines won't bring the database down). Examples include JSON-based **key-value** (Amazon DynamoDB) and **document stores** (MongoDB) and **wide-column** databases (Google BigTable).

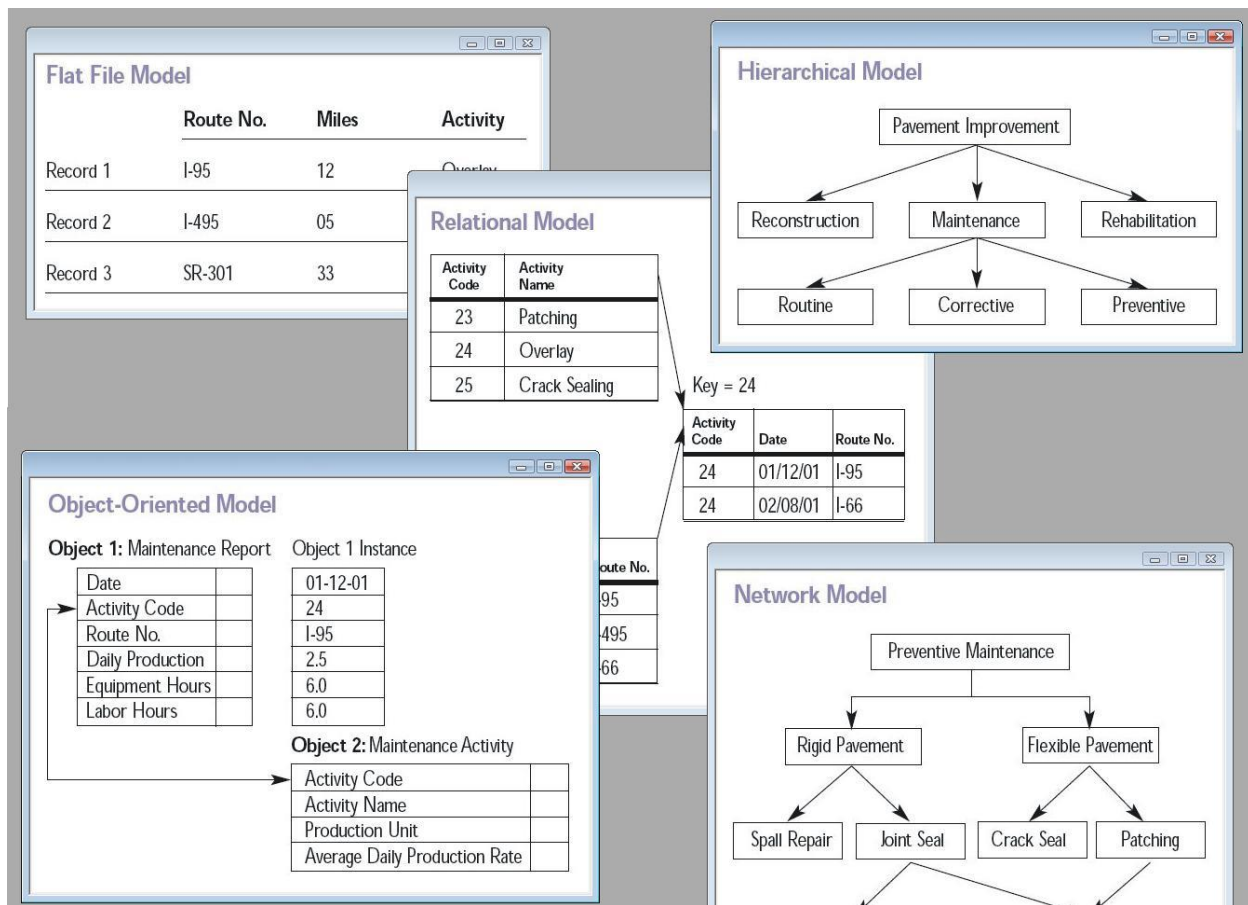


Figure 2 Illustration of Database Models (from Wikimedia commons).

3.4 "ABSTRACTING" AWAY THE DETAILS AND LEVELS OF MODELING

It's important to remember that "data modeling" is a process (as opposed to a "result"), and that it is expected one will use different models as one works through this process. In general, there are the least three (and maybe four) levels of data modeling:

1. **End-user models** are often presented as a series of simplified E-R or Object diagrams. Each of these present data as different end users see it. For example, in the Hogwarts case, an instructor like Snape might care about student grades, while a groundskeeper like Hagrid might care about efficient access to inventory lists. Database designers might create a series of models to capture these different "views" of the data.
2. **Conceptual models** present a "big picture" view of the data. For example, the school head Dumbledore would want a model that contains both Snape's and Hagrid's data. Again, this might involve ERDs or Object diagrams, though with more depth than the previous stage.
3. **Internal (or logical) models** encode the formal logical "rules" that the chosen DBMS will eventually enforce. We need to consider things like the data types for different attributes, the structures of objects/entities, and so on. This is where we must choose our "type" of model: Relational, Object, NoSQL, etc. The logical model for a (pure) object database will look very different from modeling for a (pure) relational database.
 - a. We say that data has **logical independence** if and only if changes to the internal model do NOT affect the conceptual level. This independence is a big deal for end users and programmers, and was a significant reason many organizations adopted the relational model.

4. **Physical models** concern how the data will be stored (on local drives? Shared across a network? or On the "cloud"?). While these aren't the details database developers and administrators spend most of their time worrying about (thanks to the development of modern data models!), this can sometimes impose constraints on how we do step 3.
 - a. We say that data has **physical independence** if changes to the physical substructure (e.g., changing from a single large drive to a collection of networked drives) don't affect the logical level.

3.5 REVIEW QUESTIONS

1. Suppose you were designing a database for a library (which had data on books, patrons, and loans). What might the end-user model look like (either for a staff member or patron)? A conceptual model?
2. Compare and contrast the object-oriented and relational models. Give an example of how each might model something like a Library database. (You don't need to include all of the details!).
3. The successful analysis of Big Data offers many potential advantages, such as producing new medical treatments, creating improved AI systems (such as self-driving cars), and even predicting events such as earthquakes or volcanoes. However, there are also some threats, particularly personal privacy (it's increasingly easy for companies to gather *a lot* of data about you, even if you would rather they wouldn't!).
 - a. Do you think there is a general "right to privacy"? If so, what kind of data should be protected?
 - b. How can data professionals (developers, software engineers, etc.) help ensure their work will *help* rather than *harm* people?

4 OPTIONAL READING: BIG DATA IS PEOPLE (BY REBECCA LEMOV)²

[Brendan's Note: Technology has advanced since this article was written (in 2016). To what extent do you think the problems Lemov identifies have gotten better? Worse?]

We live in what is sometimes called the 'petabyte era', and this pronouncement has provoked much discussion of the sheer size of data stores being created, as well as their rapid growth. Claims circulate along the lines of: 'Every day, we create 2.5 quintillion bytes of data – so much that 90 per cent of the data in the world today has been created in the last two years alone.' This particular statistic comes from IBM's website under the topic: 'What is Big Data?' but similar ones appear regularly in the popular media. The idea has an impact. Among other things, it is used to initiate a conversation in which an IBM representative, via a pop-up entreaty, offers big-data services. Merely defining big data, it seems, generates more opportunities for big data.

And the process continues. Ever more urgently in the press, in business and in scholarly journals, the question arises of what is unique about *big data*. Often the definitions are strangely circular. In 2013, a writer for the *Columbia Journalism Review* described big data as a catchall label that describes the new way of understanding the world through the analysis of vast amounts of data' – a statement that amounts to: big data is big... and it's made of data. Others talk about its transformational properties. In *Wired* magazine, the tech evangelist Chris Anderson claimed the 'end of theory' had been reached. So much data now exists that it is unnecessary to build a hypothesis to test scientifically. The data can, if properly handled and analysed, 'speak for themselves. Many resort to definitions that stress the 'three Vs': a data set is 'big data' if it qualifies as huge in *volume*, high in *velocity*, and diverse in *variety*. The three Vs occasionally pick up a fourth, *veracity*, which can

² Rebecca Lemov, "Why Big Data Is Actually Small, Personal and Very Human," Aeon, 2016, <https://aeon.co/essays/why-big-data-is-actually-small-personal-and-very-human>.

be interpreted in a number of ways. At the least, it evokes the striving to capture entire populations, which opens up new frontiers of possibility.

What is often forgotten, or temporarily put aside, in such excited discussions is how much of this newly created stuff is *made of and out of* personal data, the almost literal mining of subjectivity. In fact, the now common ‘three Vs’ were coined in 2001 by the industry analyst Doug Laney to describe key *problems* in data management, but they’ve become reinterpreted as the very definition of big data’s nearly infinite sense of applicability and precision.

When introducing the topic of big data in a class I teach at Harvard, I often mention the Charlton Heston movie *Soylent Green*, set in a sci-fi dystopian future of 2022, in which pollution, overpopulation and assisted suicide are the norm. Rations take the form of the eponymous soylent-green tablets, purportedly made of high-energy plankton, spewed from an assembly line and destined to feed the have-nots. Heston’s investigation inevitably reveals the foodstuff’s true ingredients, and such is the ubiquity of the film’s famous tagline marking his discovery that I don’t think spoiler alert applies: *Soylent green is people!*

Likewise, I like to argue, if in a different register: ‘Big data is people.’

Most definitions of big data don’t take account of its inherent humanness, nor do they grapple meaningfully with its implications for the relationship between technology and changing ways of defining ourselves. What makes new collections of data different, and therefore significant, is their quality of being *generated continuously* from people’s mundane, scarcely thought-through, seemingly tiny actions such as Tweets, Facebook likes, Twitches, Google searches, online comments, one-click purchases, even viewing-but-skipping-over a photograph in your feed – along with the intimacy of these actions. They are ‘faint images of me’ (to borrow a phrase from William Gibson’s description of massed data traces), lending ghostly new life to the fruits of algorithmic processing.

Examples of the production sites of such data, as the geographer Rob Kitchin recently cataloged them, include the recording of retail purchases; digital devices that save and communicate the history of their own use (such as mobile phones); the logging of transactions and interactions across digital networks (eg email or online banking); clickstream data that record navigation through a website or app; measurements from sensors embedded into objects or environments; the scanning of machine-readable objects such as travel passes or barcodes; ‘automotive telematics’ produced by drivers; and social-media postings. These sources are producing massive, dynamic flows of diverse, fine-grained, relational data.

In 2012, Wal-Mart was generating 2.5 petabytes of data relating to more than 1 million customer transactions every hour. The same year, Facebook reported that it was processing 2.5 billion pieces of content (links, comments), 2.7 billion likes, and 300 million photo uploads per day. Meanwhile, opportunities for granular data-gathering keep evolving. This February, Facebook rolled out a diversified array of six emoji-like buttons to add range and affective specificity to the responsive clicks possible on the site. Another new feature adds more than 50 additional customised gender descriptors to choose from on Facebook, rather than the binary ‘male’ or ‘female’.

Continuously assembled trails of data derived from all those inputs are quickly being put to use. Data streams can feed maps that tell you not just where you are but also where you want to go; they can, as well, fuel preemptive police work – that is, programs that focus investigations based on patterns discerned in data before a subject has committed a crime. *Big data is people*, then, in two senses. It is made up of our clickstreams and navigational choices; and it in turn makes up many socially significant policies and even self-definitions, allegiances, relationships, choices, and categories.

Some cultural critics call what is emerging a ‘[new mind control](#)’ capable of flipping major elections. Others describe a form of rapacious human engineering. Shoshana Zuboff of Harvard Business School argues that the harnessing of behavioural data is having massively disruptive results on freedom, privacy, moral

reasoning, and autonomy – results that will be playing out for decades to come. In her view, it is nothing less than a virulent new form of capitalism.

The momentum of big-data definitions tends to reinforce the impression that big data is devoid of subjectivity, or of any human point of view at all. A set of social-science scholars working in the field of technology studies recently urged researchers to turn from ‘data-centred’ to ‘people-centred’ methods, arguing that too much focus on a data-driven approach neglects the human being who is at the core of sociological studies. This reminder, however useful, neglects the central fact that data traces are *made up of* people.

Contrary to the novelty with which big data is frequently presented, important parts of this information-gathering process are not quite new – not at all new, in fact. Platforms such as social media are of recent design, but the goal of automated access, the concept of human-as-data, and the fantasy of total information long pre-exist the recent developments. This realisation punctures claims that we are grossly transformed as human beings by big data. The circulation of pervasive inaccuracies about big data is a problem because it has a quelling effect. Misconceptions about big data, tautological repetition, and confusion about its very meaning stifle needed conversations about data privacy and data use.

Even as we pay lip service to diminishing domains of privacy and increasing incursions into this beleaguered space – legal incursions, illegal ones, and the varieties in between – and even as we are reminded by whistleblowers that there is abundant cause for concern, we resist connecting the public-sphere discourse with our own circulating intimacies. Likewise, a feeling that big data is inhuman reinforces the sense that it cannot be modified or regulated; it is too often regarded as a raw force of nature that simply must be harnessed. These beliefs foster intrusions of government and private capital forces that people would probably resist much more strenuously if they clearly understood what is happening. The situation boils down, really, to this: to unwittingly accept big data’s hype is to be passive in the face of big data’s mantle of inevitability. Awareness is the only hope.

For all their futuristic trappings, big data and data-driven science resonate strongly with the history of social-scientific techniques, which during the course of the 20th century reached ever more exactly into the realm of the subjective, the self, the intimate and the personal. As the social sciences differentiated themselves – sociology from anthropology from social psychology from economics, each in its own department, each with its own areas of interest and special tools – experts battened down authority and built firewalls against enthusiast amateurs, quasi-professionals, and interloping women. Mainstream, professionalising social science abounded in techniques for data extraction, setting scenes in which subjects would be inclined and accustomed to share their memories, their lives, the seemingly banal details of their first steps or marital first nights.

In Muncie, Indiana, the vast ‘Middletown’ study conducted by the sociologists Robert and Helen Lynd between 1924 and 1926 employed a new grab-bag method (adapted in part from anthropology, in part from sociology) that combined information from interviews, participant-observation, newspaper research, questionnaires and other sources. As the historian Sarah E Igo wrote in *The Averaged American* (2007): ‘No fact or observation seemed too trivial to include in their purview, from the contents of seventh-grade school curricula to that of popular movies, from the number of hours spent on household washing to the size of Middletowners’ backyards.’

The mining of intimacy has a largely untold history. From early 20th-century observational networks to social surveys and polling efforts, to later-century focus groups, techniques evolved to become ever more targeted. The once out of bounds came in bounds in a seemingly relentless process. The ephemeral was materialised, the fleeting anchored. No subject, no state of subjectivity, was to be ignored. As the psychologist James Sully wrote in 1881: ‘The tiny occupant of the cradle has had to bear the piercing glance of the scientific eye.’ Likewise, by mid-century, everything from hallucinations to idle memories of the most pedestrian variety were targeted as data – with, in some cases, experimental data banks built to hold them.

In 1947, the psychologist Roger Barker created the 'Midwest Psychological Field Station, a social-science laboratory stationed in the small town of Oskaloosa, Kansas, in the process of which the town emerged as a kind of *de facto* laboratory. Revolutionising observation opportunities, Barker and his colleagues pioneered the regular capture of data concerning 'everyday life' – the unremarkable yet vexingly hard-to-capture details of boy scouts at play in sandlots, schoolyards and other spaces throughout the town. What appears as trivial detail – seven-year-old Raymond at 7:01 am on Tuesday, 26 April 1949, picks up a sock and begins pulling it on his left foot, slow to wake up and groggy, while his mother jokes: 'Can't you get your peepers open?' – pooled with more such data, mounted and massed together, makes a unique resource for sociological study to access the 'ordinary' cadence of life during a now-bygone time in a much-changed place. The unremarkable, researchers sensed, would inexorably cease to be so.

Meanwhile, other techniques emerged in research environments designed to further intimate revelations. As the researchers Terry Bristol and Edward Fern have shown, focus-group participants – beginning in the late 1950s – entered a situation in which they experienced a unique mix of 'anonymity and arousal' that facilitated 'expression of shared experiences. These developments formed part of an American science for objectifying the realm of the subjective in the modern social sciences. A Midwestern flair ran through several of these projects, winding their way from polling in Indiana to child study in Kansas to the beauty parlours and kitchens of Middletown.

Another area of growing focus during the golden age of behavioural techniques was the use of anthropological subjects to pursue experiments in total access. Scientists looked at these relationships as opportunities to publish and penetrate new domains; research subjects from groups around the globe such as the Cree, the Navajo and Bikini Islanders pursued a range of goals including payment, self-knowledge, participation, feedback and the chance to make one's voice heard in a not-yet-entirely-imagined scientific record.

By many calculations, a Hopi Indian man named Don Talayesva counts as the most-intensively documented such subject in history, in a life stretching from 1890 to 1976. Malaysia participated in 350 hours of formal interviews with the anthropologist Leo Simmons alone between 1938 and 1940, during which he used his life experiences as a Hopi to fill the taxonomic pigeonholes for 'Hopi' within an encyclopedic knowledge bank, the Human Relations Area File, hosted at Yale. There were also 8,000 diary pages Talayesva contributed to ethnographers; 341 dreams written down in wire-bound notebooks; a set of wide-ranging interviews; a full Rorschach protocol and other projective tests; and, as the result of all this, a thriving correspondence with the French surrealist André Breton.

Talayesva's usual rate of pay was seven cents per page of diary-writing and 35 cents per hour of interviewing, adding some expense for the Rorschach test, all of which made him a relatively wealthy man by Hopi standards. Whether or not he remains today the most-documented native person in history, he was the fount of an 'enormous body of data, wrote the author of a psychosexual re-study of the Talayesva corpus. Likewise, for another eminent anthropologist, he provided a storehouse of substantive data'. The man himself became a kind of data pipeline.

The pioneering sociological studies targeted not only individuals but also large groups. The anthropologist Melford Spiro psychologically tested all inhabitants of an entire island in the Western Pacific (Ifaluk) during the same post-Second World War years as neighbouring atolls in the area (Bikini, among others) were sites of intensive nuclear tests. For his academic research, Spiro data-mined whole populations. For American Indians, this ongoing process constituted what the historian Thomas Biolsi calls 'internal pacification'. In a study of Sioux history between the 1880s and 1940s, Biolsi shows how investigations of Sioux life delved increasingly into psychological domains. Such research probed further and further into the remote psyche, which was treated as a kind of territory to be mapped. But the mapping also helped change the territory. Not evenly or regularly, but painstakingly, a transformation of the Sioux 'self' was underway, as Biolsi describes it, and the process of being measured, counted, quantified and (eventually) tested served to aid and abet the subjective changes taking place. In effect, such research subjects were canaries in coalmines.

Experiments in the range of ways to get at what could be considered internal data – or what specialists called ‘subjective materials’ – extended from Indian reservations to occupied areas to reformatories, factories and armies. Large punch-card-driven statistical enquiries opened up new possibilities, as in the US Army’s landmark *The American Soldier* project. Starting on 8 December 1941, the day after Pearl Harbor, and continuing until the last days of the war, the Army’s Research Branch administered more than 200 questionnaires, each usually given to a sample of around 2,500 soldiers, some in overseas theatres of battle and remote outposts.

The result was a mine of data, perhaps unparalleled in magnitude in the history of any single research enterprise in social psychology or sociology,’ in the words of the project’s director, Samuel Stouffer. *The American Soldier* project provided unique access to the inner states of soldiers – as a resulting publication put it, an unbiased look at ‘what the soldier thinks.

Early audiences for the Lumière brothers’ films, especially crowds viewing footage of the oncoming train that seemed about to penetrate their cinema screen, ran out of the theatre in panic because they had not yet become trained in the illusionary calculus involved in making the experience of watching films enjoyable – at least according to the myths surrounding *Arrival of a Train at La Ciotat Station*. Made in 1895, with its first public showing in 1896, the 50-second film comprised one continuous shot of an everyday occurrence – a train steaming into station – yet the camera was positioned on the platform to produce the feeling of a locomotive bearing down on the seated viewer. These ‘naïve’ audiences confused signals that indicated one scenario (danger) with another (watching a film about a dangerous situation).

One sees replications of this process of getting acclimated to new techniques in the arena of penetrating social-science instruments, and to its modern incarnation in big data. Early on, citizens seemed to have little resistance to being asked questions by phone pollsters, whereas today, as few as 3 per cent will answer questions by phone – if they even have a landline. Technology and resistance arose hand in hand.

As the ‘man on the street interview’ debuted in the 1950s and ‘60s, members of the public initially watched in bemusement or alarm as strangers posed random questions accompanied by recording devices. A wonderful depiction of this process appears in the classic 1961 cinema-vérité documentary *Chronique d’un Été* by the anthropologist Jean Rouch and the sociologist Edgar Morin, in which work-worn Parisians exiting the Métro encounter two snappily dressed young women pointing microphones and pressing into their personal space, asking: ‘Monsieur, are you happy?’ The query occasions a range of responses from blank to flirtatious to heart-rending. There is as yet, however, no sense of this as a normal activity, as one can see from the standardised ease with which college students or city commuters answer pointed questions today.

By the second half of the 20th century, citizens (particularly urban dwellers) became increasingly accustomed to the possibility that intrusive questions might be asked at any time and answers expected; evasions also became normalised. The famous Kinsey Report research, built on thousands of interviews, stimulated a wave of prank social surveyors asking women intimate questions about their sex lives. Pretending to be working for the Kinsey report, these caddish pretenders often received fulsome answers until the surprisingly trusting public was warned of predatory practices. At other times, prospective participants queued in eagerness to take part in Kinsey interviews on sexual behaviour, many reporting exhilarating effects that came from feeling oneself to be ‘an infinitesimal cog in one of the greatest fact-finding projects ever undertaken... the first great mass of facts, and figures drawn from a cross-section of all social and educational groups, from which charts, curves and finally conclusions may be drawn,’ as one interviewee reported.

Also around the mid-20th century, a Harvard Business School team under the industrial psychologist Elton Mayo pioneered the gathering of intimate interviews with employees of the Hawthorne Works factory in Cicero, Illinois, carrying out some 20,000 interviews. They aimed to capture what another eminent social scientist famously called the ‘elusive phenomena’. Their answers remain on file at Harvard’s Baker Library, a curious archive of the mundane details of the lives of factory girls circa 1938 or 1941. Jennie, for example, provided the interviewer with details of her evolving hairstyle, hoped-for Christmas gifts, and proclivity to

wear her stockings rolled when working on hot days. Assembly-line girls joked about going out drinking the night before and slowing production during the day. As with anthropologists' American Indian subjects who often spoke back via purportedly neutral measuring instruments (in one case in 1885, Sioux respondents answering a census survey supplied names such as 'Shit Head' and other obscenities), the Hawthorne subjects turned researchers' techniques to their own purposes, at times asking snarky questions, fomenting rebellion, or teasing visiting sociologists.

One day, perhaps not long from now, people will look back at our current decade amazed at the ease and naiveté with which we, enchanted users of new tech, failed to see the value of our own behavioural data resources, and therefore gave them away for little more than ease of use, entertainment value and dubious accretions of status. That is one possibility. On the other hand, the more we can see the process at work, the less the average user falls sway to the hype of 'never before. It becomes possible to disintegrate what is actually new about data-gathering capabilities – arguably: scale and granularity – from those tendencies that existed before, sometimes long in the past.

A recent White House report on 'big data' concluded: 'The technological trajectory, however, is clear: more and more data will be generated about individuals and will persist under the control of others.' When trying to understand the ramifications of this big-data trajectory, I argue, it is necessary again to bear in mind that the data is not only generated about individuals but also *made out of* individuals. It is human data.

In parallel with researchers' increasingly aggressive collection of personal data, modern research subjects became trained how to participate, how to answer, how to obligingly offer up the precincts of the self to scrutiny – our own and others. This training has prepared us for a new level of intrusiveness. We are all now primed to give ourselves up to big data.

To look at the history of the quest to scoop up the totality of human behaviour in a scientific net is to illuminate the present obsessions. In the end, we see that the attempt to capture all the parts of human experience – mostly boiling it down to its everyday-ness – reveals many elements that are familiar, but also some that are distinctly and wildly different. Big data is not a project suddenly spawned by our just-now-invented digital technologies, although it is transformed by them. Instead, we can see that it is a project at the driving core of all of modern life. In many ways, it crowns long-held ambitions to build a transparent machinic self, one capable of optimisation as well as automation.

The behavioural sciences in the 20th century, particularly as practiced by Americans spanning the globe, engaged in an ambitious push to capture ever-more-intimate parts of human experience and to turn them into materials amenable to manipulation by clever machines. This was a prelude to the Rubicon now known as big data. These historical projects, sometimes more and sometimes less closely aligned with government and military sources of support, ran on complex hodgepodge combinations of old and new technology, and paved the way for our own moment in which corporate-cum-research entities feed government data mills rather than the other way around.

This is why the erstwhile goal to gather large amounts of what specialists called 'human materials' resonates so strongly today. It speaks to the tension between *humans* and *materials*, and the desire to turn one into the other. What the Swiss biological historian Bruno Strasser calls the 'supposedly unprecedented data-driven sciences' are not so unprecedented. For that reason, it is necessary to understand what came before in order to grasp what is actually new.

Preceding examples of innovative data collection already targeted inner provinces, and already engaged in subjective data-mining. They were unable to do so on anything resembling the scale today possible by use of digitally derived data streams. Nonetheless, the old imperative to mine inner worlds finds a place at the heart of today's practices. By being arrayed in new tech, and by being incorporated in new ways into our human experiences, it is transformed. As are we. But if we really want to understand that transformation and to speak

up about it – if we want to see what is truly new rather than what is bumpiously paraded as new – we will need to be anchored in the historical particulars. We need to see the human in the data machine.