August 5th 2022

# Bases de datos

## Introducción y motivación

# Las cuatro "V" del Big Data: volumen

"Research has determined that **a single well can produce about one terabyte of production data – every day.** That must be processed, organized and stored by someone, and can take hours of company time to do so."

Fuente: [Production and operational data management made easier | Oil & Gas Product News (oilandgasproductnews.com)]

# Las cuatro "V" del Big Data: volumen

## London Stock Exchange completes acquisition of Refinitiv

29 January 2021 1 3 1

Source: London Stock Exchange

**Further to the announcement on 28 January 2021, London Stock Exchange Group plc ("LSEG" or the "Company") confirms that its all-share acquisition of Refinitiv (the "Transaction") has now completed.**

"It collects, aggregates, analyses, curates, stores and distributes huge volumes of data from a large number of sources. **It collects more than one petabyte of data every day** and processes more than a million unstructured documents. Its data platform provides 40 billion market updates every day."

Fuente: https://www.computerweekly.com/news/252467734/London-Stock-Exchange-agrees-to-acquire-Refinitiv-for-22bn

https://lawtomated.com/structured-data-vs-unstructured-data-what-are-they-and-why-care/

# Las cuatro "V" del Big Data: variedad

- Base tecnológica de una ciudad inteligente, compuesta por sensores, comunicaciones y portales de datos abiertos.
- Procesamiento geográfico combinado con información de tráfico.
- Videos procesados en tiempo real.
- Sensores lumínicos, de energía, de concentración de NO2.

Fuente: http://www.oas.org/es/sms/cicte/ciberciudades.pdf

# Las cuatro "V" del Big Data: variedad (tipos de datos nuevos)

IDC predicts that by 2015 over 90 percent of that data will be unstructured (e.g., images, videos, MP3 music files, and other files based on social media and Web enabled workloads). While this data is full of rich information it is hard to understand and analyze[2]. For many organizations, they haven't even begun to understand how to do it with these new data types.

Fuente:
https://www.hp.com/hpinfo/newsroom/press_kits/2012/HPDiscoverFrankfurt2012/IO_Whitepaper_Harness_the_Power_of_Big_Data.pdf

Fuente:
Data Never Sleeps 8.0

domo.com

# Las cuatro "V" del Big Data: velocidad

THE WEEK

OPINION BRIEFINGS SPEED READS TALKING POINTS CARTOONS MORE

IN-DEPTH BRIEFING

## Wall Street's secret advantage: High-speed trading

*They're unknown and invisible to most of us, but electronic trading programs now rule the stock markets*

THE WEEK STAFF
JANUARY 11, 2015

**What is high-speed trading?** It's Wall Street's winning edge. By harnessing massive computer power to buy and sell stocks in the blink of an eye, high speed traders leverage tiny changes in value to make huge profits. The technique was pioneered in the

**"Does co-location increase speed?** Yes. **The physical proximity to the exchange server** reduces the time from when a firm's buy or sell order is entered and when it's executed. "By co-locating," says Adam Afshar of Hyde Park Global, a high-speed trading firm, "we are able to take 21 milliseconds off our trades. In the past, 21 milliseconds was a trivial matter. Now it's a pivotal matter." Several academic studies have found that shaving even one millisecond off every trade can be worth $100 million a year to a large, high-speed trading firm."

Fuente: https://theweek.com/articles/493238/wall-streets-secret-advantage-highspeed-trading

# Las cuatro "V" del Big Data: veracidad

Semantic Web - Interoperability, Usability, Applicability *an IOS Press Journal*

## Linked Data, Big Data, and the 4th Paradigm

*Editorial*

Pascal Hitzler,[a] Krzysztof Janowicz,[b]
[a] *Kno.e.sis Center, Wright State University, USA*
[b] *University of California, Santa Barbara, USA*

"Big Data, an increasing number of V's has been used to characterize different dimensions and challenges of Big Data: volume, velocity, variety, value, and veracity. Interestingly, different (scientific) disciplines highlight certain dimensions and neglect others. For instance, super computing seems to be mostly interested in the volume dimension while researchers working on sensor webs and the internet of things seem to push on the velocity front. **The social sciences and humanities, in contrast, are more interested in value and veracity**."

Fuente: http://www.semantic-web-journal.net/system/files/swj488.pdf

# Las cuatro "V" del Big Data: veracidad

The Scientific World Journal

ScientificWorldJournal. 2015; 2015: 453597. PMCID: I
Published online 2015 Oct 1. PMI
doi: 10.1155/2015/453597

**Distilling Big Data: Refining Quality Information in the Era of Yottabytes**

The data quality can generally be grouped under several categories like **accuracy, believability, reputation, objectivity, factuality, consistency, freedom from bias, correctness, and unambiguousness**.

Fuente: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4606078/#

BLOG@CACM

# The End of a DBMS Era (Might Be Upon Us)

By Michael Stonebraker
June 30, 2009
Comments (7)

VIEW AS: SHARE:

Relational database management systems (DBMSs) have been remarkably successful in capturing the DBMS marketplace. To a first approximation they are "the only game in town," and the major vendors (IBM, Oracle, and Microsoft) enjoy an overwhelming market share. They are selling "one size fits all"; i.e., a single relational engine appropriate for all DBMS needs. Moreover, the code line from all of the major vendors is quite elderly, in all cases dating from the 1980s. Hence, the major vendors sell software that is a quarter century old, and has been extended and morphed to meet today's needs. In my opinion, these legacy systems are at the end of their useful life. They deserve to be sent to the "home for tired software."

Fuente: https://cacm.acm.org/blogs/blog-cacm/32212-the-end-of-a-dbms-era-might-be-upon-us/fulltext

# A Relational Model of Data for Large Shared Data Banks

E. F. Codd
*IBM Research Laboratory, San Jose, California*

**Future users of large data banks must be protected from having to know how the data is organized in the machine (the internal representation). A prompting service which supplies such information is not a satisfactory solution. Activities of users at terminals and most application programs should remain unaffected when the internal representation of data is changed and even when some aspects of the external representation are changed. Changes in data representation will often be needed as a result of changes in query, update, and report traffic and natural growth in the types of stored information.**

**Existing noninferential, formatted data systems provide users with tree-structured files or slightly more general network models of the data. In Section 1, inadequacies of these models are discussed. A model based on *n*-ary relations, a normal form for data base relations, and the concept of a universal data sublanguage are introduced. In Section 2, certain operations on relations (other than logical inference) are discussed and applied to the problems of redundancy and consistency in the user's model.**

KEY WORDS AND PHRASES: data bank, data base, data structure, data organization, hierarchies of data, networks of data, relations, derivability, redundancy, consistency, composition, join, retrieval language, predicate calculus, security, data integrity
CR CATEGORIES: 3.70, 3.73, 3.75, 4.20, 4.22, 4.29

## 1. Relational Model and Normal Form

### 1.1. Introduction

This paper is concerned with the application of elementary relation theory to systems which provide shared access to large banks of formatted data. Except for a paper by Childs [1], the principal application of relations to data systems has been to deductive question-answering systems. Levein and Maron [2] provide numerous references to work in this area.

In contrast, the problems treated here are those of *data independence*—the independence of application programs and terminal activities from growth in data types and changes in data representation—and certain kinds of *data inconsistency* which are expected to become troublesome even in nondeductive systems.

The relational view (or model) of data described in Section 1 appears to be superior in several respects to the graph or network model [3, 4] presently in vogue for noninferential systems. It provides a means of describing data with its natural structure only—that is, without superimposing any additional structure for machine representation purposes. Accordingly, it provides a basis for a high level data language which will yield maximal independence between programs on the one hand and machine representation and organization of data on the other.

A further advantage of the relational view is that it forms a sound basis for treating derivability, redundancy, and consistency of relations—these are discussed in Section 2. The network model, on the other hand, has spawned a number of confusions, not the least of which is mistaking the derivation of connections for the derivation of relations (see remarks in Section 2 on the "connection trap").

Finally, the relational view permits a clearer evaluation of the scope and logical limitations of present formatted data systems, and also the relative merits (from a logical standpoint) of competing representations of data within a single system. Examples of this clearer perspective are cited in various parts of this paper. Implementations of systems to support the relational model are not discussed.

### 1.2. Data Dependencies in Present Systems

The provision of data description tables in recently developed information systems represents a major advance toward the goal of data independence [5, 6, 7]. Such tables facilitate changing certain characteristics of the data representation stored in a data bank. However, the variety of data representation characteristics which can be changed *without logically impairing some application programs* is still quite limited. Further, the model of data with which users interact is still cluttered with representational properties, particularly in regard to the representation of collections of data (as opposed to individual items). Three of the principal kinds of data dependencies which still need to be removed are: ordering dependence, indexing dependence, and access path dependence. In some systems these dependencies are not clearly separable from one another.

1.2.1. *Ordering Dependence.* Elements of data in a data bank may be stored in a variety of ways, some involving no concern for ordering, some permitting each element to participate in one ordering only, others permitting each element to participate in several orderings. Let us consider those existing systems which either require or permit data elements to be stored in at least one total ordering which is closely associated with the hardware-determined ordering of addresses. For example, the records of a file concerning parts might be stored in ascending order by part serial number. Such systems normally permit application programs to assume that the order of presentation of records from such a file is identical to (or is a subordering of) the

# What is a Database?

- **Relational Database Management System**
- **ANSI Standard SQL**

# What is a Database?

A database is a collection of permanently stored data that is:

- Logically related - the data relates to other data (tables to tables).
- Shared - many users may access the data.
- Protected - access to data is controlled.
- Managed - the data has integrity and value.

# Logical/Relational Modeling

## The Logical Model

- **Should be the same regardless of data volume**
- **Data is organized according to what it represents - real world business data in table (relational) form**
- **Is generic – the logical model is the template for physical implementation on any RDBMS platform**

## Normalization

- **Process of reducing a complex data structure into a simple, stable one**
- **Involves removing redundant attributes, keys, and relationships from the conceptual data model**

# Transaction processing

**is a unit of work that must be executed atomically and in apparent isolation**

***1. Logging*** ***in order to assure durability***

***2. Concurrency control*** ***transactions must appear to execute in isolation***

***3. Deadlock resolution*** ***transactions compete for resources***

# The ACID properties of transactions

***"A" stands for "atomicity"*** ***the all-or-nothing execution of transactions.***

***"I" stands for "isolation"*** ***the fact that each transaction must appear to be executed as if no other transaction is executing at the same time.***

***"D" stands for durability*** ***the condition that the effect on the database of a transaction must be never lost, once the transaction has completed.***

***The remaining letter "C" stands for consistency.*** ***Transactions are expected to preserve the consistency of the database.***

# Relational Databases

- Relational Databases are founded on Set Theory and based on the Relational Model.
- A Relational Database consists of a collection of logically related tables.
- A table is a two dimensional representation of data consisting of rows and columns.

Column (Attribute)

EMPLOYEE

| EMPLOYEE NUMBER | MANAGER EMPLOYEE NUMBER | DEPARTMENT NUMBER | JOB CODE | LAST NAME | FIRST NAME | HIRE DATE | BIRTH DATE | SALARY AMOUNT |
|---|---|---|---|---|---|---|---|---|
| 1006 | 1019 | 301 | 312101 | Stein | John | 861015 | 631015 | 3945000 |
| 1008 | 1019 | 301 | 312102 | Kanieski | Carol | 870201 | 680517 | 3925000 |
| 1005 | 0801 | 403 | 431100 | Ryan | Loretta | 861015 | 650910 | 4120000 |
| 1004 | 1003 | 401 | 412101 | Johnson | Darlene | 861015 | 560423 | 4630000 |
| 1007 | | | | Villegas | Arnando | 870102 | 470131 | 5970000 |
| 1003 | 0801 | 401 | 411100 | Trader | James | 860731 | 570619 | 4785000 |

Row (Single Entity)

Relation

**The employee table has:**

- **Nine columns of data**
- **Six rows of data - one per employee**
- **Missing data values represented by nulls**
- **Column and row order are arbitrary**

# Primary Keys

Primary Key (PK) values uniquely identify each row in a table.

EMPLOYEE

| EMPLOYEE NUMBER | MANAGER EMPLOYEE NUMBER | DEPARTMENT NUMBER | JOB CODE | LAST NAME | FIRST NAME | HIRE DATE | BIRTH DATE | SALARY AMOUNT |
|---|---|---|---|---|---|---|---|---|
| PK | | | | | | | | |
| 1006 | 1019 | 301 | 312101 | Stein | John | 861015 | 631015 | 3945000 |
| 1008 | 1019 | 301 | 312102 | Kanieski | Carol | 870201 | 680517 | 3925000 |
| 1005 | 0801 | 403 | 431100 | Ryan | Loretta | 861015 | 650910 | 4120000 |
| 1004 | 1003 | 401 | 412101 | Johnson | Darlene | 861015 | 560423 | 4630000 |
| 1007 | | | | Villegas | Arnando | 870102 | 470131 | 5970000 |
| 1003 | 0801 | 401 | 411100 | Trader | James | 860731 | 570619 | 4785000 |

**Primary Key Rules**

- **A Primary Key is required for every table.**
- **Only one Primary Key is allowed in a table.**
- **Primary Keys may consist of one or more columns.**
- **Primary Keys cannot have duplicate values.**
- **Primary Keys cannot be null.**
- **Primary Keys are considered "non-changing" values.**

# Foreign Keys

EMPLOYEE (partial listing)

Job Code Table

| EMPLOYEE NUMBER | MANAGER EMPLOYEE NUMBER | DEPARTMENT NUMBER | JOB CODE | LAST NAME | FIRST NAME | HIRE DATE | BIRTH DATE | SALARY AMOUNT |
|---|---|---|---|---|---|---|---|---|
| PK | FK | FK | FK | | | | | |
| 1006 | 1019 | 301 | 312101 | Stein | John | 861015 | 631015 | 3945000 |
| 1008 | 1019 | 301 | 312102 | Kanieski | Carol | 870201 | 680517 | 3925000 |
| 1005 | 0801 | 403 | 431100 | Ryan | Loretta | 861015 | 650910 | 4120000 |
| 1004 | 1003 | 401 | 412101 | Johnson | Darlene | 861015 | 560423 | 4630000 |
| 1007 | | | | Villegas | Arnando | 870102 | 470131 | 5970000 |
| 1003 | 0801 | 401 | 411100 | Trader | James | 860731 | 570619 | 4785000 |

Foreign Key (FK) values model relationships.

DEPARTMENT

| DEPARTMENT NUMBER | DEPARTMENT NAME | BUDGET AMOUNT | MANAGER EMPLOYEE NUMBER |
|---|---|---|---|
| PK | | | FK |
| 501 | marketing sales | 80050000 | 1017 |
| 301 | research and development | 46560000 | 1019 |
| 302 | product planning | 22600000 | 1016 |
| 403 | education | 93200000 | 1005 |
| 402 | software support | 30800000 | 1011 |
| 401 | customer support | 98230000 | 1003 |
| 201 | technical operations | 29380000 | 1025 |

- **Foreign Keys (FK) are optional.**
- **A table may have more than one FK.**
- **A FK may consist of more than one column.**
- **FK values may be duplicated.**
- **FK values may be null.**
- **FK values may be changed.**
- **FK values must exist elsewhere as a PK (i.e. have referential integrity).**

# Answering Questions with a Relational Database

EMPLOYEE

| EMPLOYEE NUMBER | MANAGER EMPLOYEE NUMBER | DEPARTMENT NUMBER | JOB CODE | LAST NAME | FIRST NAME | HIRE DATE | BIRTH DATE | SALARY AMOUNT |
|---|---|---|---|---|---|---|---|---|
| PK | FK | FK | FK | | | | | |
| 1006 | 1019 | 301 | 312101 | Stein | John | 861015 | 631015 | 3945000 |
| 1008 | 1019 | 301 | 312102 | Kanieski | Carol | 870201 | 680517 | 3925000 |
| 1005 | 0801 | 403 | 431100 | Ryan | Loretta | 861015 | 650910 | 4120000 |
| 1004 | 1003 | 401 | 412101 | Johnson | Darlene | 861015 | 560423 | 4630000 |
| 1007 | | | 432101 | Villegas | Arnando | 870102 | 470131 | 5970000 |
| 1003 | 0801 | 401 | 411100 | Trader | James | 860731 | 570619 | 4785000 |

DEPARTMENT

| DEPARTMENT NUMBER | DEPARTMENT NAME | BUDGET AMOUNT | MANAGER EMPLOYEE NUMBER |
|---|---|---|---|
| PK | | | FK |
| 501 | marketing sales | 80050000 | 1017 |
| 301 | research and development | 46560000 | 1019 |
| 302 | product planning | 22600000 | 1016 |
| 403 | education | 93200000 | 1005 |
| 402 | software support | 30800000 | 1011 |
| 401 | customer support | 98230000 | 1003 |
| 201 | technical operations | 29380000 | 1025 |

1. Name the department in which James Trader works.
2. Who manages the Education Department?
3. Identify by name an employee who works for James Trader.
4. James Trader manages which department?

# Relational Advantages

Advantages of a Relational Database compared to other database methodologies include:

- More flexible than other types
- Allows businesses to quickly respond to changing conditions
- Being data-driven vs. application driven
- Modeling the business, not the processes
- Makes applications easier to build because the data does more of the work
- A single copy of the data may serve multiple purposes
- Supporting trend toward end-user computing
- Being easy to understand
- No need to know the access path
- Solidly founded in set theory

# The Advantage of Using Detail Data

QUESTION: How effective was the national advertisement for jeans that ran June 6 through June 8?

STORE ITEM DAY

| STORE NUMBER | ITEM NUMBER | DATE | QUANTITY SOLD |
|---|---|---|---|
| PK | | | |
| FK | | | |
| 1 | | June01 | 110 |
| 1 | | June02 | 126 |
| 1 | | June03 | 127 |
| 1 | | June04 | 144 |
| 1 | | June05 | 102 |
| 1 | 2 | June06 | 344 |
| 1 | 2 | June07 | 410 |
| 1 | 2 | June08 | 426 |
| 1 | 2 | June09 | 297 |
| 1 | 2 | June10 | 164 |
| 1 | 2 | June11 | 167 |
| 1 | 2 | June12 | 115 |
| 1 | 2 | June13 | 108 |
| 1 | 2 | June14 | 99 |
| ... | ... 2 | ... | ... |
| | 2 | | |
| 2 | 2 | June01 | 50 |
| 2 | 2 | June02 | 47 |
| 2 | 2 | June03 | 32 |
| 2 | 2 | June04 | 20 |
| 2 | | June05 | 37 |
| 2 | | June06 | 144 |
| 2 | 2 | June07 | 126 |
| 2 | 2 | June08 | 164 |
| 2 | 2 | June09 | 96 |
| 2 | 2 | June10 | 48 |
| 2 | 2 | June11 | 34 |
| 2 | 2 | June12 | 24 |
| 2 | 2 | June13 | 30 |
| 2 | 2 | June14 | 45 |
| ... | ... 2 | ... | ... |
| 5 | 2 | June01 | 13 |
| 5 | 2 | June02 | 14 |
| 5 | 2 | June03 | 10 |
| 5 | 2 | June04 | 12 |
| 5 | | June05 | 15 |
| 5 | 2 | June06 | 32 |
| 5 | 2 | June07 | 65 |
| 5 | | June08 | 87 |
| 5 | 2 | June09 | 18 |
| 5 | 2 | June10 | 26 |
| 5 | 2 | June11 | 8 |
| 5 | 2 | June12 | 14 |
| 5 | 2 | June13 | 22 |
| 5 | 2 | June14 | 11 |

STORE ITEM DAY

| STORE NUMBER | ITEM NUMBER | WEEK ENDING | QUANTITY SOLD |
|---|---|---|---|
| PK | | | |
| FK | | | |
| ... | ... | ... | ... |
| 1 | 2 | June07 | 1363 |
| 1 | 2 | June14 | 1376 |
| ... | ... | ... | ... |
| 2 | 2 | June 07 | 456 |
| 2 | 2 | June14 | 441 |
| ... | ... | ... | ... |
| 5 | 2 | June07 | 161 |
| 5 | 2 | June14 | 186 |
| ... | ... | ... | ... |

## DETAIL DATA vs. SUMMARY DATA

- Detail data gives a more accurate picture.
- Correct business decisions result.

# Data Warehouse Usage Evolution

## STAGE 1
REPORTING
WHAT
happened?

Primarily Batch Pre-Defined Reports

Batch

## STAGE 2
ANALYZING
WHY
did it happen?

Increase in Ad Hoc Queries

Ad Hoc

## STAGE 3
PREDICTING
WHAT
will happen?

Analytical Modeling Grows

Analytics

## STAGE 4
OPERATIONALIZING
WHAT
is happening?

Continuous Update & Time Sensitive Queries Become Important

Continuous Update Short Queries

## STAGE 5
ACTIVE WAREHOUSING
MAKING
it happen!

Event Based Triggering Takes Hold

Event-Based Triggering

# Data Analytics

Toad™ for Oracle™

RStudio

python™

# Data Analytics / Data Governance

# The Data Warehouse

A central, enterprise-wide database that contains information extracted from
operational systems.

- Based on enterprise-wide model
- Populated by extraction/loading of data from operational systems
- Provides a single view of the business

# The Parsing Engine

The Parsing Engine is responsible for:

- Managing individual sessions
- Parsing and optimizing your SQL requests
- Building query plans with the Optimizer
- Dispatching the optimized plan
- Sending the answer set response back to the requesting client

# Database Objects

Databases are repositories for Database Objects:

- Tables – rows and columns of data
- Views – predefined subsets of existing tables
- Triggers – SQL statements associated with a table
- Stored Procedures – program stored
- User Defined Function –program that provides additional functionality

# Database Space

Database objects require space in a Database or User as follows:

- Tables - require Perm Space
- Views - do not require Perm Space
- Triggers - do not require Perm Space
- Stored Procedures - require Perm Space
- UDFs – require Perm Space

Space limits are specified for each database and for each user:

- Perm Space - maximum amount of space available for permanent tables and index subtables
- Spool Space - maximum amount of work space available for request processing
  - holds intermediate query result sets.

# RAID Protection

## RAID 1 (Mirroring)

- Each physical disk has an exact.
- Mirroring reduces available disk space by 50%.

## RAID 5 (Parity)

- Data striped across rank of 4 disks.
- If a disk fails, any missing block may be reconstructed using the other three disks.
- Parity reduces available disk space by 25% in a 4-disk rank.
- Reconstruction of failed disks takes longer than RAID 1.

| Summary | RAID-1 - Good performance with disk failures<br>Higher cost in terms of disk space<br><br>RAID-5 - Reduced performance with disk failures<br>Lower cost in terms of disk space |
|---|---|