Web Based Databases & Relational Database Design

Dr. Thomas Hicks
Computer Science Department
Trinity University

Time To Update Your Resume?

"Web based applications are the present and the future. No matter what your background, … your resume is only going to improve with Web applications development experience."

Brad Bulger, Software Engineer

How Many Of You Are Going To Apply For A Summer Internship, Graduate School, or Permanent Computer Science Position In The Next Few Months?

Successful Completion Of CSCI-3343 - Database Design Will Add The Following To Your Resume

Operating Systems: Linux, Windows XP Pro, Vista, Windows 2003 Server, etc.

Database Environments: MSSQL, Visual Studio, Access, MySQL, SQL, etc.

Web Servers: Windows IIS, Apache, PHP, ASP, etc.

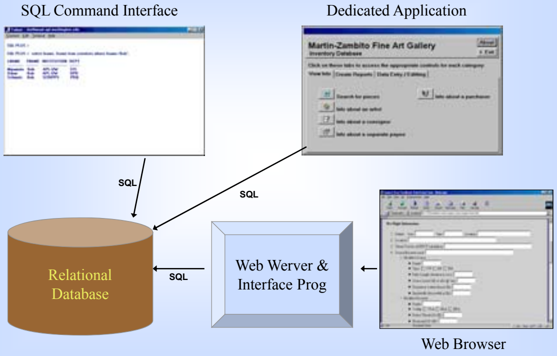
Database Skills: Design Relational Databases, Form/Report Design, Multi-User Apps, Integrate Web & Network Apps

Spreadsheets Are Not Databases!

They have only a small portion of the functionality we expect in a database or database management system

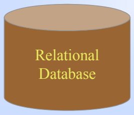
Databases can serve a single user on a single computer, works groups of numbers of people on a local area network, or even hundreds of people and trillions of bytes of data

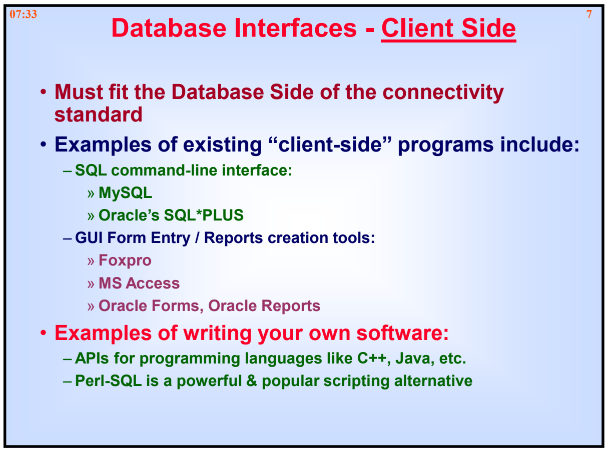
Database Interface Options



Database Interfaces - Database Server Side

- Database Runs as a Separate Process - may be on a separate computer
- One database server might even be serving out data for multiple databases
- Standards for how the SQL commands come into the database and data comes out:
  - –ODBC : Open DataBase Connectivity
  - –JDBC : Java DataBase Connectivity
  - –OLE-DB : Object Linking & Embedding - DataBase
  - –Proprietary such as Oracle's Pro*C

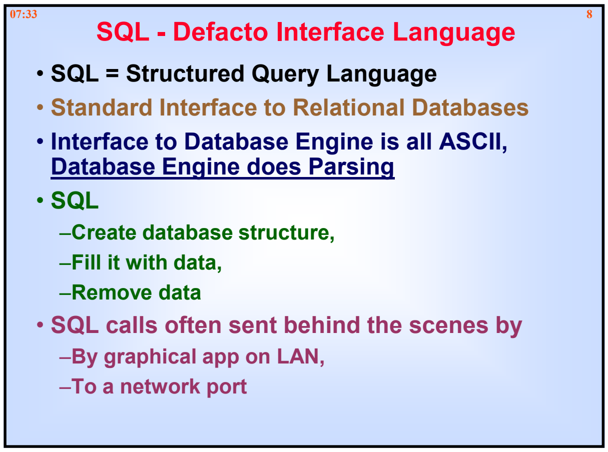




07:33 7

Database Interfaces - Client Side

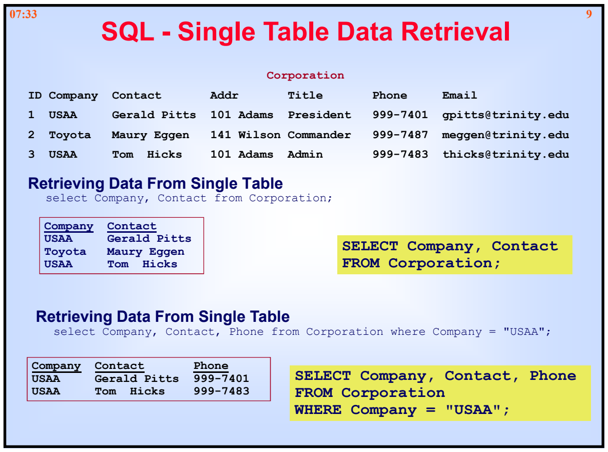
- Must fit the Database Side of the connectivity standard
- Examples of existing “client-side” programs include:
  - SQL command-line interface:
    - » MySQL
    - » Oracle’s SQL*PLUS
  - GUI Form Entry / Reports creation tools:
    - » Foxpro
    - » MS Access
    - » Oracle Forms, Oracle Reports
- Examples of writing your own software:
  - APIs for programming languages like C++, Java, etc.
  - Perl-SQL is a powerful & popular scripting alternative



07:33 8

SQL - Defacto Interface Language

- SQL = Structured Query Language
- Standard Interface to Relational Databases
- Interface to Database Engine is all ASCII, Database Engine does Parsing
- SQL
  - –Create database structure,
  - –Fill it with data,
  - –Remove data
- SQL calls often sent behind the scenes by
  - –By graphical app on LAN,
  - –To a network port



07:33 9

SQL - Single Table Data Retrieval

Corporation

ID	Company	Contact	Addr	Title	Phone	Email
1	USAA	Gerald Pitts	101 Adams	President	999-7401	gpitts@trinity.edu
2	Toyota	Maury Eggen	141 Wilson	Commander	999-7487	meggen@trinity.edu
3	USAA	Tom Hicks	101 Adams	Admin	999-7483	thicks@trinity.edu

Retrieving Data From Single Table

```
select Company, Contact from Corporation;
```

Company	Contact
USAA	Gerald Pitts
Toyota	Maury Eggen
USAA	Tom Hicks

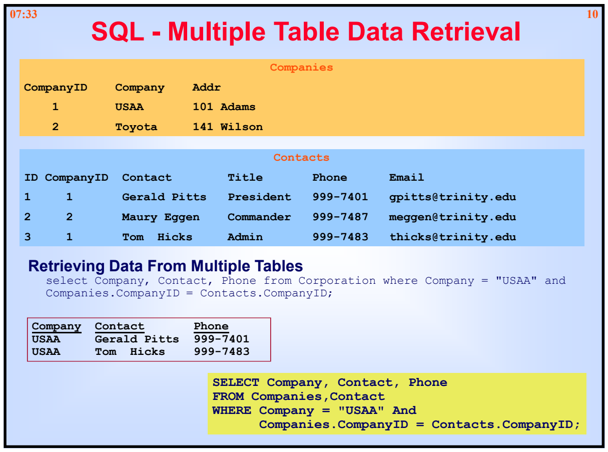
```
SELECT Company, Contact
FROM Corporation;
```

Retrieving Data From Single Table

```
select Company, Contact, Phone from Corporation where Company = "USAA";
```

Company	Contact	Phone
USAA	Gerald Pitts	999-7401
USAA	Tom Hicks	999-7483

```
SELECT Company, Contact, Phone
FROM Corporation
WHERE Company = "USAA";
```



07:33 10

SQL - Multiple Table Data Retrieval

Companies

CompanyID	Company	Addr
1	USAA	101 Adams
2	Toyota	141 Wilson

Contacts

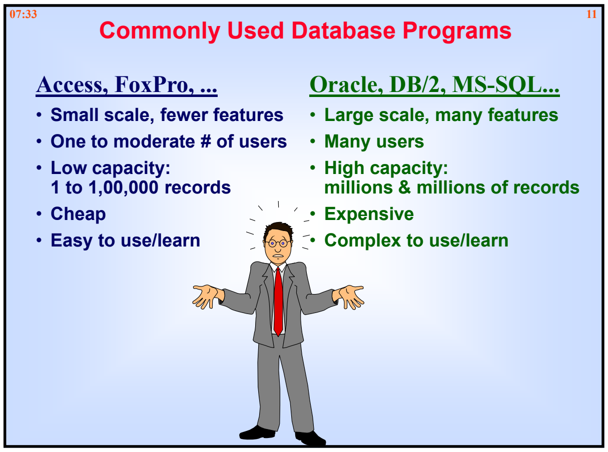
ID	CompanyID	Contact	Title	Phone	Email
1	1	Gerald Pitts	President	999-7401	gpitts@trinity.edu
2	2	Maury Eggen	Commander	999-7487	meggen@trinity.edu
3	1	Tom Hicks	Admin	999-7483	thicks@trinity.edu

Retrieving Data From Multiple Tables

```
select Company, Contact, Phone from Corporation where Company = "USAA" and
Companies.CompanyID = Contacts.CompanyID;
```

Company	Contact	Phone
USAA	Gerald Pitts	999-7401
USAA	Tom Hicks	999-7483

```
SELECT Company, Contact, Phone
FROM Companies,Contact
WHERE Company = "USAA" And
      Companies.CompanyID = Contacts.CompanyID;
```



07:33 11

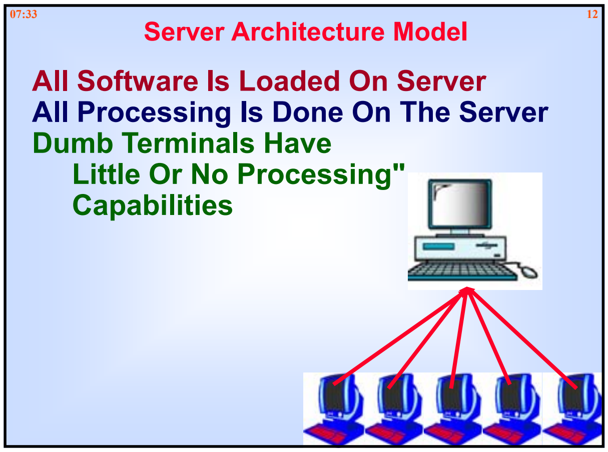
Commonly Used Database Programs

Access, FoxPro, ...

- Small scale, fewer features
- One to moderate # of users
- Low capacity: 1 to 1,00,000 records
- Cheap
- Easy to use/learn

Oracle, DB/2, MS-SQL...

- Large scale, many features
- Many users
- High capacity: millions & millions of records
- Expensive
- Complex to use/learn



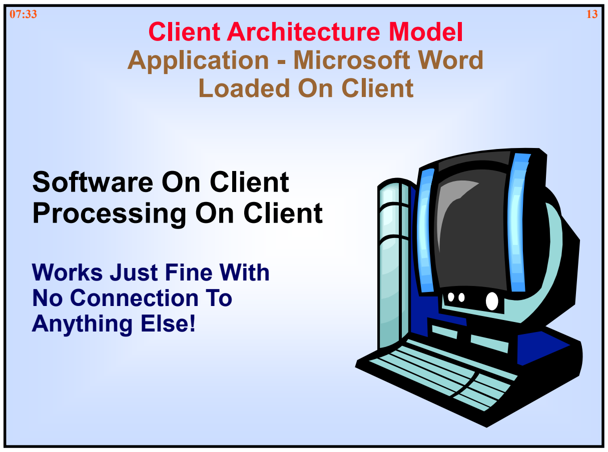
07:33 12

Server Architecture Model

All Software Is Loaded On Server

All Processing Is Done On The Server

Dumb Terminals Have
Little Or No Processing"
Capabilities

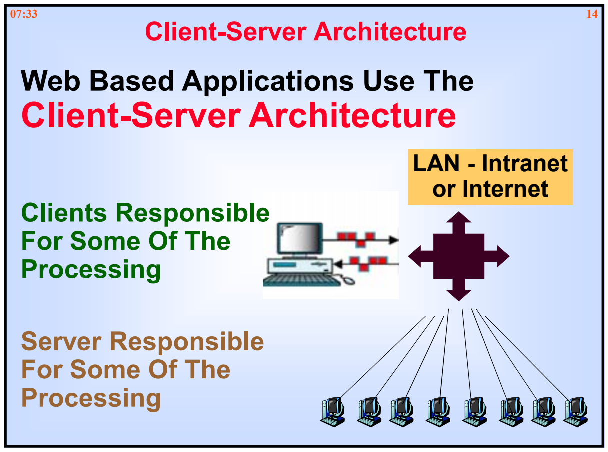


07:33 13

Client Architecture Model
Application - Microsoft Word
Loaded On Client

Software On Client
Processing On Client

Works Just Fine With
No Connection To
Anything Else!



07:33 14

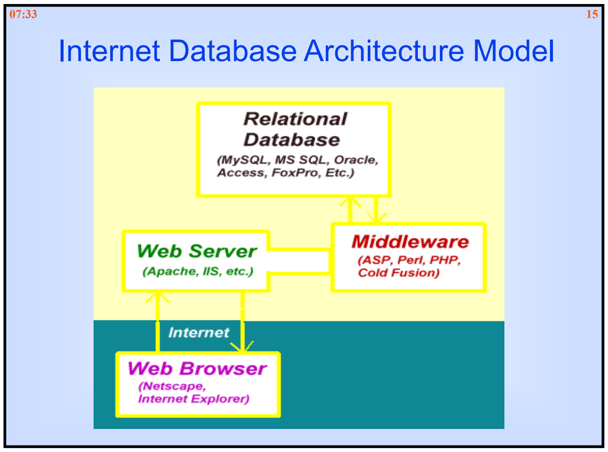
Client-Server Architecture

Web Based Applications Use The
Client-Server Architecture

LAN - Intranet
or Internet

Clients Responsible
For Some Of The
Processing

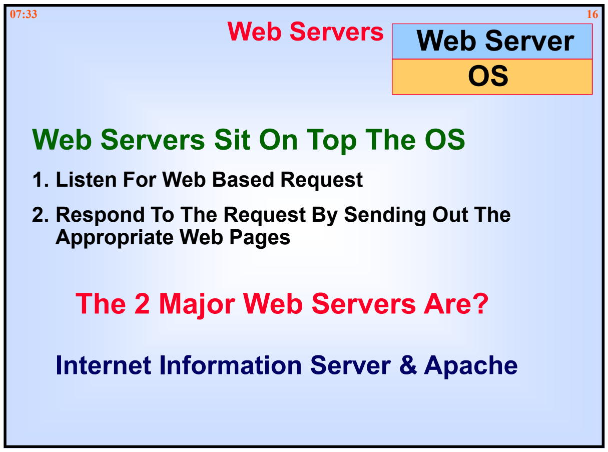
Server Responsible
For Some Of The
Processing



07:33 15

Internet Database Architecture Model

Relational Database
(MySQL, MS SQL, Oracle, Access, FoxPro, Etc.)
Web Server
(Apache, IIS, etc.)
Middleware
(ASP, Perl, PHP, Cold Fusion)
Internet
Web Browser
(Netscape, Internet Explorer)



07:33 16

Web Servers

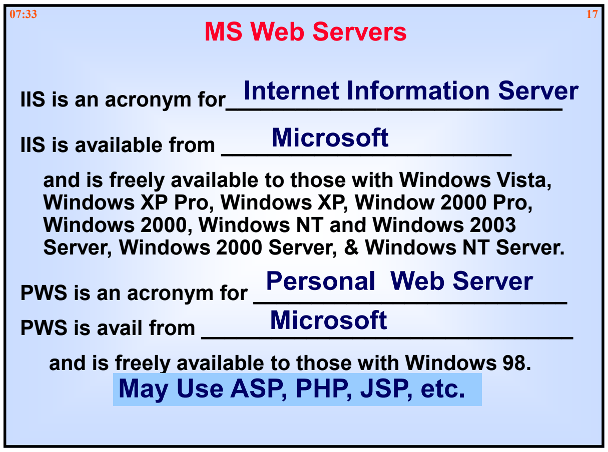
Web Server
OS

Web Servers Sit On Top The OS

1. Listen For Web Based Request
2. Respond To The Request By Sending Out The Appropriate Web Pages

The 2 Major Web Servers Are?

Internet Information Server & Apache



07:33 17

MS Web Servers

IIS is an acronym for Internet Information Server

IIS is available from Microsoft

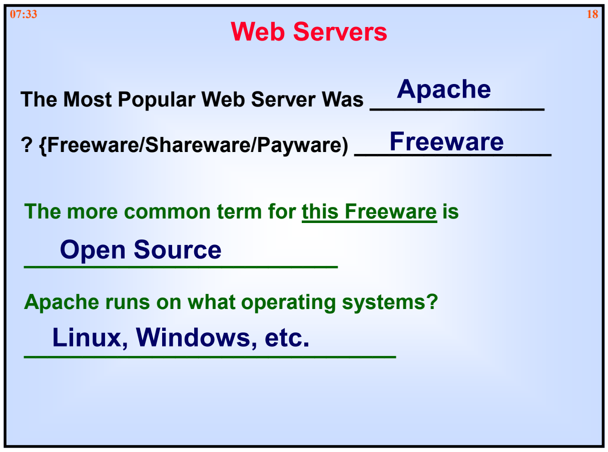
and is freely available to those with Windows Vista, Windows XP Pro, Windows XP, Window 2000 Pro, Windows 2000, Windows NT and Windows 2003 Server, Windows 2000 Server, & Windows NT Server.

PWS is an acronym for Personal Web Server

PWS is avail from Microsoft

and is freely available to those with Windows 98.

May Use ASP, PHP, JSP, etc.



07:33 18

Web Servers

The Most Popular Web Server Was Apache

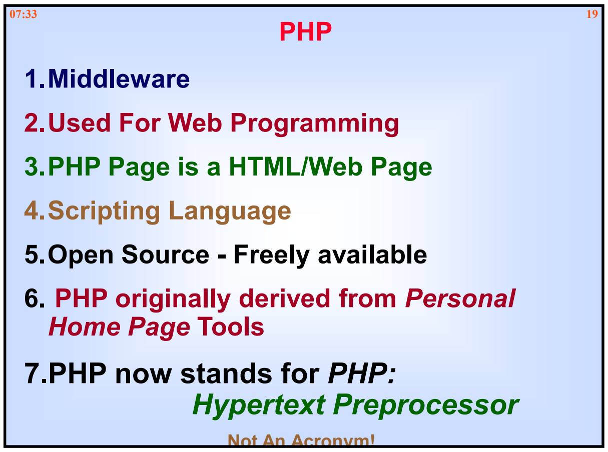
? {Freeware/Shareware/Payware) Freeware

The more common term for this Freeware is

Open Source

Apache runs on what operating systems?

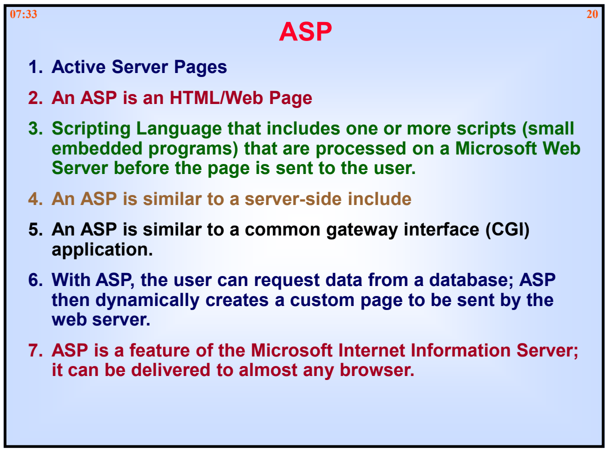
Linux, Windows, etc.



PHP

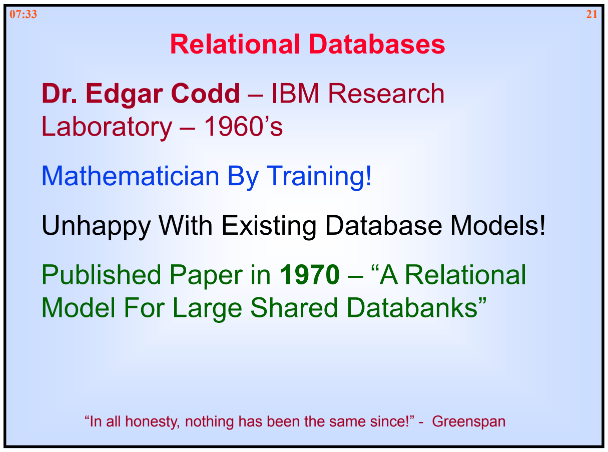
1. Middleware
2. Used For Web Programming
3. PHP Page is a HTML/Web Page
4. Scripting Language
5. Open Source - Freely available
6. PHP originally derived from *Personal Home Page* Tools
7. PHP now stands for *PHP: Hypertext Preprocessor*

Not An Acronym!



ASP

1. Active Server Pages
2. An ASP is an HTML/Web Page
3. Scripting Language that includes one or more scripts (small embedded programs) that are processed on a Microsoft Web Server before the page is sent to the user.
4. An ASP is similar to a server-side include
5. An ASP is similar to a common gateway interface (CGI) application.
6. With ASP, the user can request data from a database; ASP then dynamically creates a custom page to be sent by the web server.
7. ASP is a feature of the Microsoft Internet Information Server; it can be delivered to almost any browser.



Relational Databases

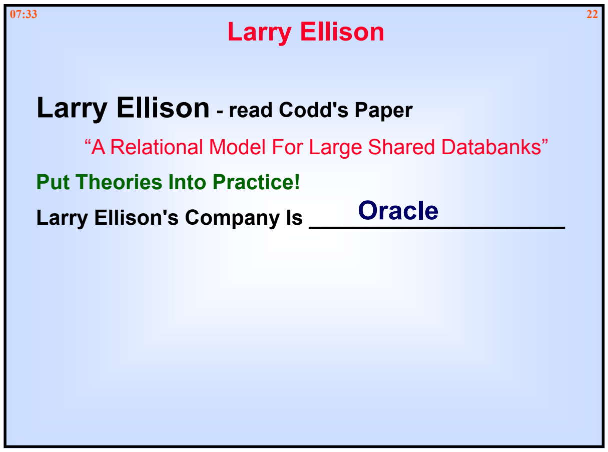
Dr. Edgar Codd – IBM Research Laboratory – 1960’s

Mathematician By Training!

Unhappy With Existing Database Models!

Published Paper in **1970** – “A Relational Model For Large Shared Databanks”

“In all honesty, nothing has been the same since!” - Greenspan



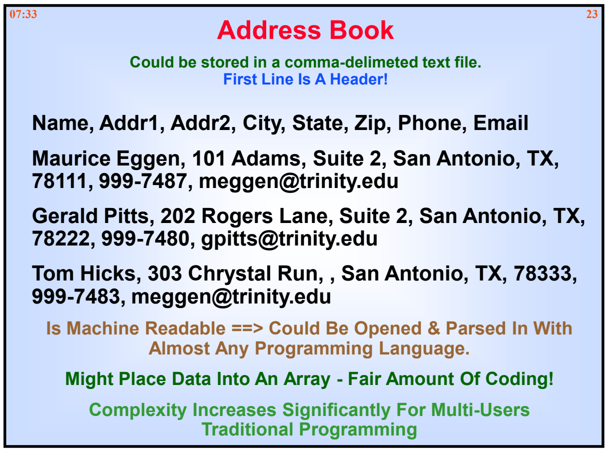
Larry Ellison

Larry Ellison - read Codd's Paper

“A Relational Model For Large Shared Databanks”

Put Theories Into Practice!

Larry Ellison's Company Is ____Oracle____



Address Book

Could be stored in a comma-delimeted text file.
First Line Is A Header!

Name, Addr1, Addr2, City, State, Zip, Phone, Email

Maurice Eggen, 101 Adams, Suite 2, San Antonio, TX, 78111, 999-7487, meggen@trinity.edu

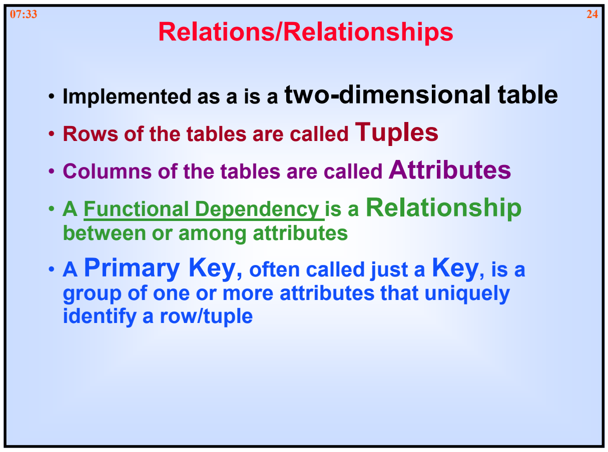
Gerald Pitts, 202 Rogers Lane, Suite 2, San Antonio, TX, 78222, 999-7480, gpitts@trinity.edu

Tom Hicks, 303 Chrystal Run, , San Antonio, TX, 78333, 999-7483, meggen@trinity.edu

Is Machine Readable ==> Could Be Opened & Parsed In With Almost Any Programming Language.

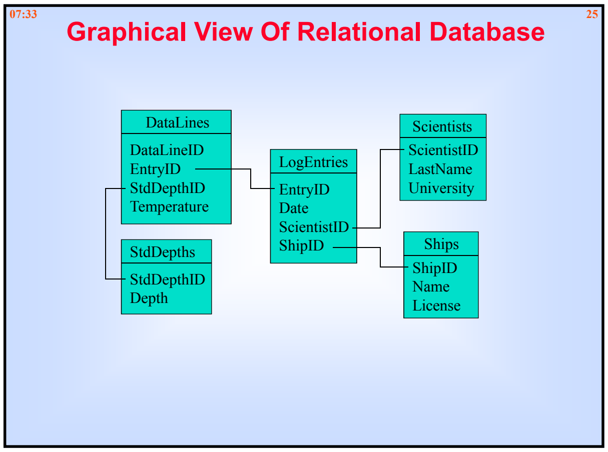
Might Place Data Into An Array - Fair Amount Of Coding!

Complexity Increases Significantly For Multi-Users
Traditional Programming



Relations/Relationships

- Implemented as a is a **two-dimensional table**
- Rows of the tables are called **Tuples**
- Columns of the tables are called **Attributes**
- A Functional Dependency is a **Relationship** between or among attributes
- A **Primary Key**, often called just a **Key**, is a group of one or more attributes that uniquely identify a row/tuple



Graphical View Of Relational Database

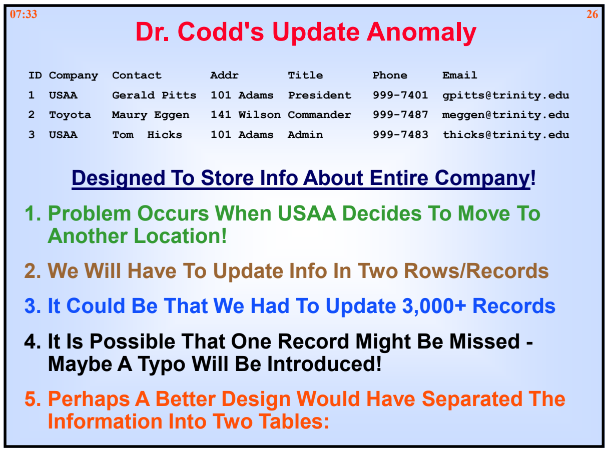
DataLines: DataLineID, EntryID, StdDepthID, Temperature

StdDepths: StdDepthID, Depth

LogEntries: EntryID, Date, ScientistID, ShipID

Scientists: ScientistID, LastName, University

Ships: ShipID, Name, License

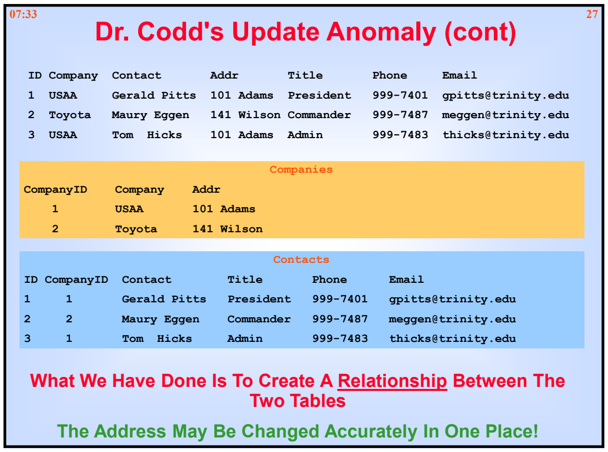


Dr. Codd's Update Anomaly

ID	Company	Contact	Addr	Title	Phone	Email
1	USAA	Gerald Pitts	101 Adams	President	999-7401	gpitts@trinity.edu
2	Toyota	Maury Eggen	141 Wilson	Commander	999-7487	meggen@trinity.edu
3	USAA	Tom Hicks	101 Adams	Admin	999-7483	thicks@trinity.edu

Designed To Store Info About Entire Company!

1. Problem Occurs When USAA Decides To Move To Another Location!
2. We Will Have To Update Info In Two Rows/Records
3. It Could Be That We Had To Update 3,000+ Records
4. It Is Possible That One Record Might Be Missed - Maybe A Typo Will Be Introduced!
5. Perhaps A Better Design Would Have Separated The Information Into Two Tables:



Dr. Codd's Update Anomaly (cont)

ID	Company	Contact	Addr	Title	Phone	Email
1	USAA	Gerald Pitts	101 Adams	President	999-7401	gpitts@trinity.edu
2	Toyota	Maury Eggen	141 Wilson	Commander	999-7487	meggen@trinity.edu
3	USAA	Tom Hicks	101 Adams	Admin	999-7483	thicks@trinity.edu

Companies

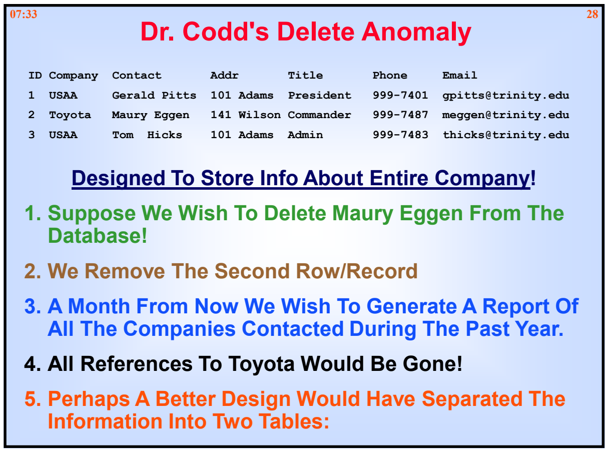
CompanyID	Company	Addr
1	USAA	101 Adams
2	Toyota	141 Wilson

Contacts

ID	CompanyID	Contact	Title	Phone	Email
1	1	Gerald Pitts	President	999-7401	gpitts@trinity.edu
2	2	Maury Eggen	Commander	999-7487	meggen@trinity.edu
3	1	Tom Hicks	Admin	999-7483	thicks@trinity.edu

What We Have Done Is To Create A Relationship Between The Two Tables

The Address May Be Changed Accurately In One Place!

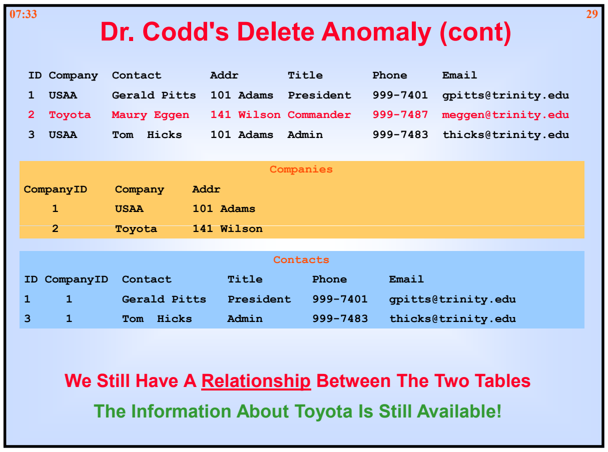


Dr. Codd's Delete Anomaly

ID	Company	Contact	Addr	Title	Phone	Email
1	USAA	Gerald Pitts	101 Adams	President	999-7401	gpitts@trinity.edu
2	Toyota	Maury Eggen	141 Wilson	Commander	999-7487	meggen@trinity.edu
3	USAA	Tom Hicks	101 Adams	Admin	999-7483	thicks@trinity.edu

Designed To Store Info About Entire Company!

1. Suppose We Wish To Delete Maury Eggen From The Database!
2. We Remove The Second Row/Record
3. A Month From Now We Wish To Generate A Report Of All The Companies Contacted During The Past Year.
4. All References To Toyota Would Be Gone!
5. Perhaps A Better Design Would Have Separated The Information Into Two Tables:



Dr. Codd's Delete Anomaly (cont)

ID	Company	Contact	Addr	Title	Phone	Email
1	USAA	Gerald Pitts	101 Adams	President	999-7401	gpitts@trinity.edu
2	Toyota	Maury Eggen	141 Wilson	Commander	999-7487	meggen@trinity.edu
3	USAA	Tom Hicks	101 Adams	Admin	999-7483	thicks@trinity.edu

Companies

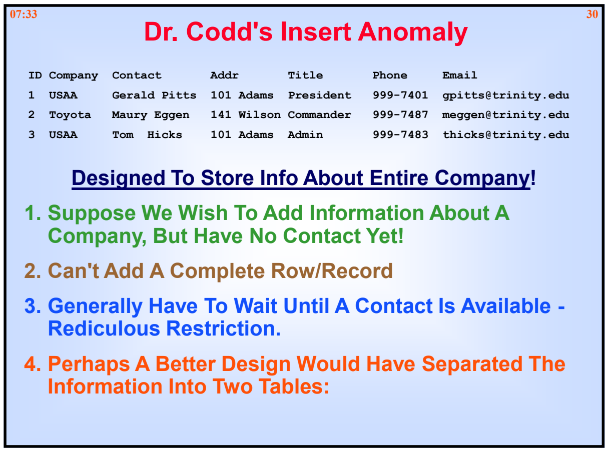
CompanyID	Company	Addr
1	USAA	101 Adams
2	Toyota	141 Wilson

Contacts

ID	CompanyID	Contact	Title	Phone	Email
1	1	Gerald Pitts	President	999-7401	gpitts@trinity.edu
3	1	Tom Hicks	Admin	999-7483	thicks@trinity.edu

We Still Have A Relationship Between The Two Tables

The Information About Toyota Is Still Available!

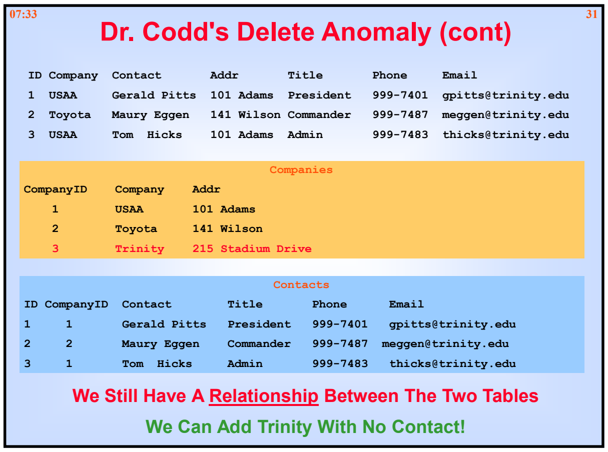


Dr. Codd's Insert Anomaly

ID	Company	Contact	Addr	Title	Phone	Email
1	USAA	Gerald Pitts	101 Adams	President	999-7401	gpitts@trinity.edu
2	Toyota	Maury Eggen	141 Wilson	Commander	999-7487	meggen@trinity.edu
3	USAA	Tom Hicks	101 Adams	Admin	999-7483	thicks@trinity.edu

Designed To Store Info About Entire Company!

1. Suppose We Wish To Add Information About A Company, But Have No Contact Yet!
2. Can't Add A Complete Row/Record
3. Generally Have To Wait Until A Contact Is Available - Rediculous Restriction.
4. Perhaps A Better Design Would Have Separated The Information Into Two Tables:



Dr. Codd's Delete Anomaly (cont)

ID	Company	Contact	Addr	Title	Phone	Email
1	USAA	Gerald Pitts	101 Adams	President	999-7401	gpitts@trinity.edu
2	Toyota	Maury Eggen	141 Wilson	Commander	999-7487	meggen@trinity.edu
3	USAA	Tom Hicks	101 Adams	Admin	999-7483	thicks@trinity.edu

Companies

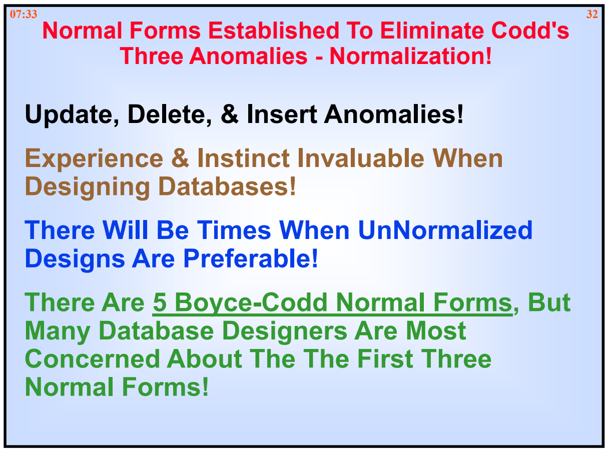
CompanyID	Company	Addr
1	USAA	101 Adams
2	Toyota	141 Wilson
3	Trinity	215 Stadium Drive

Contacts

ID	CompanyID	Contact	Title	Phone	Email
1	1	Gerald Pitts	President	999-7401	gpitts@trinity.edu
2	2	Maury Eggen	Commander	999-7487	meggen@trinity.edu
3	1	Tom Hicks	Admin	999-7483	thicks@trinity.edu

We Still Have A Relationship Between The Two Tables

We Can Add Trinity With No Contact!



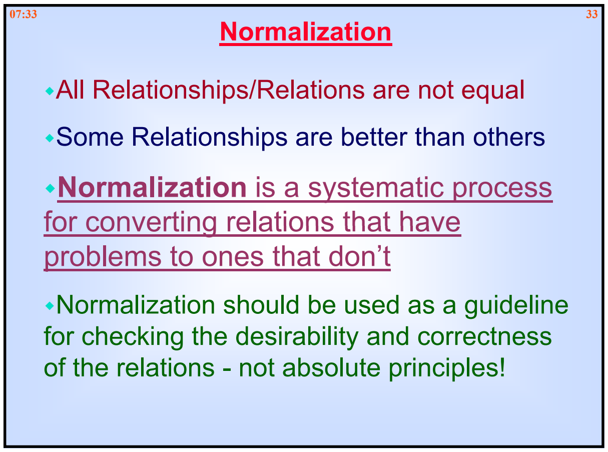
Normal Forms Established To Eliminate Codd's Three Anomalies - Normalization!

Update, Delete, & Insert Anomalies!

Experience & Instinct Invaluable When Designing Databases!

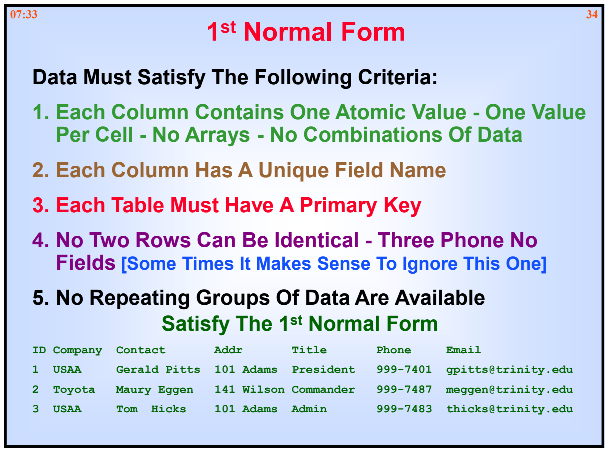
There Will Be Times When UnNormalized Designs Are Preferable!

There Are 5 Boyce-Codd Normal Forms, But Many Database Designers Are Most Concerned About The The First Three Normal Forms!



Normalization

- All Relationships/Relations are not equal
- Some Relationships are better than others
- **Normalization** is a systematic process for converting relations that have problems to ones that don't
- Normalization should be used as a guideline for checking the desirability and correctness of the relations - not absolute principles!



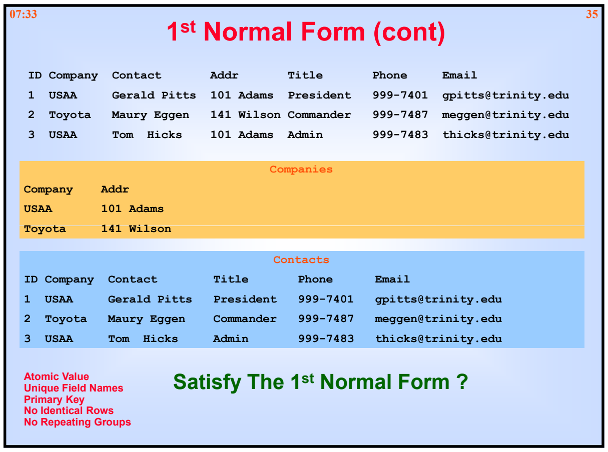
1st Normal Form

Data Must Satisfy The Following Criteria:

1. Each Column Contains One Atomic Value - One Value Per Cell - No Arrays - No Combinations Of Data
2. Each Column Has A Unique Field Name
3. Each Table Must Have A Primary Key
4. No Two Rows Can Be Identical - Three Phone No Fields [Some Times It Makes Sense To Ignore This One]
5. No Repeating Groups Of Data Are Available

Satisfy The 1st Normal Form

ID	Company	Contact	Addr	Title	Phone	Email
1	USAA	Gerald Pitts	101 Adams	President	999-7401	gpitts@trinity.edu
2	Toyota	Maury Eggen	141 Wilson	Commander	999-7487	meggen@trinity.edu
3	USAA	Tom Hicks	101 Adams	Admin	999-7483	thicks@trinity.edu



1st Normal Form (cont)

ID	Company	Contact	Addr	Title	Phone	Email
1	USAA	Gerald Pitts	101 Adams	President	999-7401	gpitts@trinity.edu
2	Toyota	Maury Eggen	141 Wilson	Commander	999-7487	meggen@trinity.edu
3	USAA	Tom Hicks	101 Adams	Admin	999-7483	thicks@trinity.edu

Companies

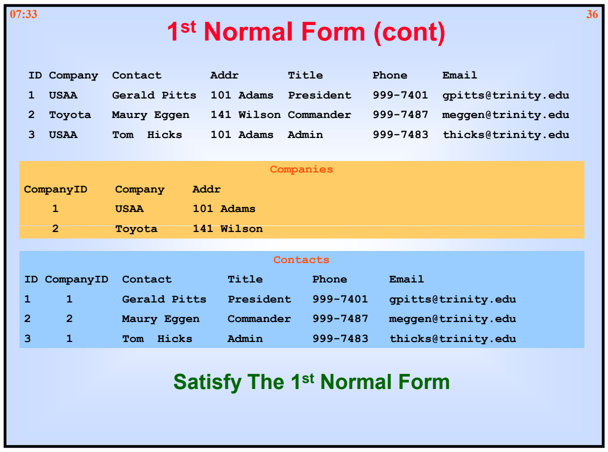
Company	Addr
USAA	101 Adams
Toyota	141 Wilson

Contacts

ID	Company	Contact	Title	Phone	Email
1	USAA	Gerald Pitts	President	999-7401	gpitts@trinity.edu
2	Toyota	Maury Eggen	Commander	999-7487	meggen@trinity.edu
3	USAA	Tom Hicks	Admin	999-7483	thicks@trinity.edu

Atomic Value
Unique Field Names
Primary Key
No Identical Rows
No Repeating Groups

Satisfy The 1st Normal Form ?



1st Normal Form (cont)

ID	Company	Contact	Addr	Title	Phone	Email
1	USAA	Gerald Pitts	101 Adams	President	999-7401	gpitts@trinity.edu
2	Toyota	Maury Eggen	141 Wilson	Commander	999-7487	meggen@trinity.edu
3	USAA	Tom Hicks	101 Adams	Admin	999-7483	thicks@trinity.edu

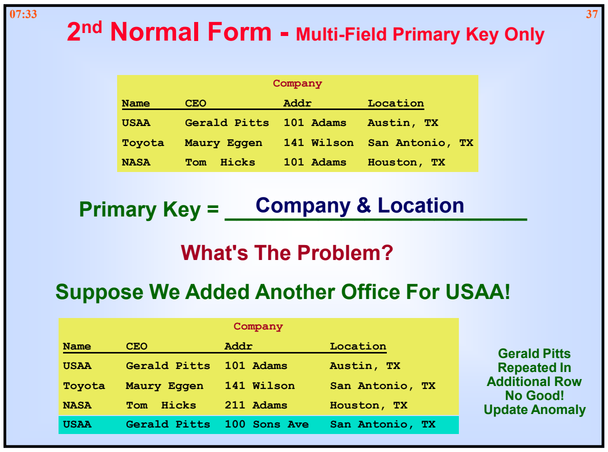
Companies

CompanyID	Company	Addr
1	USAA	101 Adams
2	Toyota	141 Wilson

Contacts

ID	CompanyID	Contact	Title	Phone	Email
1	1	Gerald Pitts	President	999-7401	gpitts@trinity.edu
2	2	Maury Eggen	Commander	999-7487	meggen@trinity.edu
3	1	Tom Hicks	Admin	999-7483	thicks@trinity.edu

Satisfy The 1st Normal Form



2nd Normal Form - Multi-Field Primary Key Only

Company

Name	CEO	Addr	Location
USAA	Gerald Pitts	101 Adams	Austin, TX
Toyota	Maury Eggen	141 Wilson	San Antonio, TX
NASA	Tom Hicks	101 Adams	Houston, TX

Primary Key = Company & Location

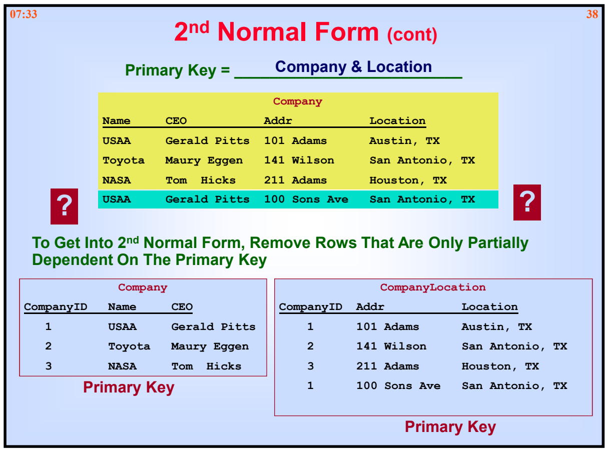
What's The Problem?

Suppose We Added Another Office For USAA!

Company

Name	CEO	Addr	Location
USAA	Gerald Pitts	101 Adams	Austin, TX
Toyota	Maury Eggen	141 Wilson	San Antonio, TX
NASA	Tom Hicks	211 Adams	Houston, TX
USAA	Gerald Pitts	100 Sons Ave	San Antonio, TX

Gerald Pitts Repeated In Additional Row No Good! Update Anomaly



2nd Normal Form (cont)

Primary Key = Company & Location

Company

Name	CEO	Addr	Location
USAA	Gerald Pitts	101 Adams	Austin, TX
Toyota	Maury Eggen	141 Wilson	San Antonio, TX
NASA	Tom Hicks	211 Adams	Houston, TX
USAA	Gerald Pitts	100 Sons Ave	San Antonio, TX

? ?

To Get Into 2nd Normal Form, Remove Rows That Are Only Partially Dependent On The Primary Key

Company

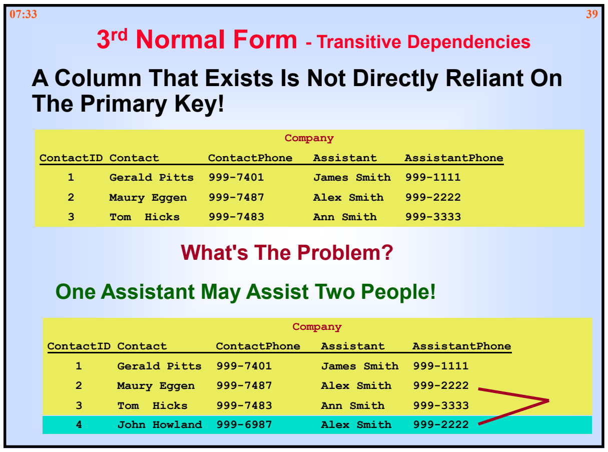
CompanyID	Name	CEO
1	USAA	Gerald Pitts
2	Toyota	Maury Eggen
3	NASA	Tom Hicks

Primary Key

CompanyLocation

CompanyID	Addr	Location
1	101 Adams	Austin, TX
2	141 Wilson	San Antonio, TX
3	211 Adams	Houston, TX
1	100 Sons Ave	San Antonio, TX

Primary Key



3rd Normal Form - Transitive Dependencies

A Column That Exists Is Not Directly Reliant On The Primary Key!

Company

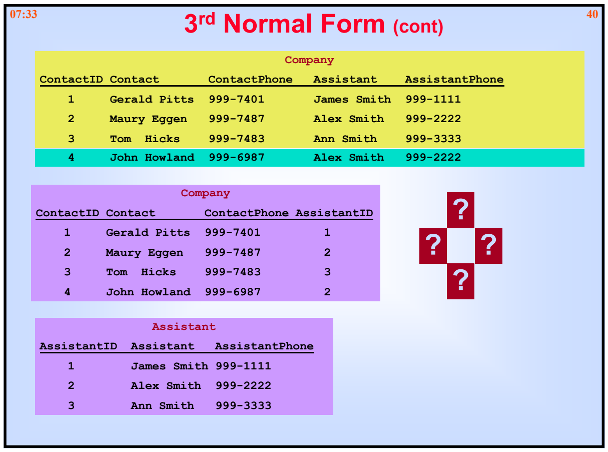
ContactID	Contact	ContactPhone	Assistant	AssistantPhone
1	Gerald Pitts	999-7401	James Smith	999-1111
2	Maury Eggen	999-7487	Alex Smith	999-2222
3	Tom Hicks	999-7483	Ann Smith	999-3333

What's The Problem?

One Assistant May Assist Two People!

Company

ContactID	Contact	ContactPhone	Assistant	AssistantPhone
1	Gerald Pitts	999-7401	James Smith	999-1111
2	Maury Eggen	999-7487	Alex Smith	999-2222
3	Tom Hicks	999-7483	Ann Smith	999-3333
4	John Howland	999-6987	Alex Smith	999-2222



3rd Normal Form (cont)

Company

ContactID	Contact	ContactPhone	Assistant	AssistantPhone
1	Gerald Pitts	999-7401	James Smith	999-1111
2	Maury Eggen	999-7487	Alex Smith	999-2222
3	Tom Hicks	999-7483	Ann Smith	999-3333
4	John Howland	999-6987	Alex Smith	999-2222

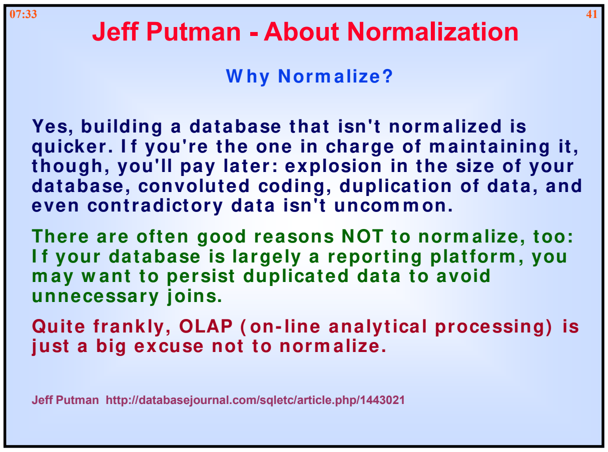
Company

ContactID	Contact	ContactPhone	AssistantID
1	Gerald Pitts	999-7401	1
2	Maury Eggen	999-7487	2
3	Tom Hicks	999-7483	3
4	John Howland	999-6987	2

? ? ? ?

Assistant

AssistantID	Assistant	AssistantPhone
1	James Smith	999-1111
2	Alex Smith	999-2222
3	Ann Smith	999-3333



Jeff Putman - About Normalization

Why Normalize?

Yes, building a database that isn't normalized is quicker. If you're the one in charge of maintaining it, though, you'll pay later: explosion in the size of your database, convoluted coding, duplication of data, and even contradictory data isn't uncommon.

There are often good reasons NOT to normalize, too: If your database is largely a reporting platform, you may want to persist duplicated data to avoid unnecessary joins.

Quite frankly, OLAP (on-line analytical processing) is just a big excuse not to normalize.

Jeff Putman http://databasejournal.com/sqletc/article.php/1443021