



A Quick Introduction to a RELATIONAL DATABASE MANAGEMENT SYSTEM (RDBMS)

John R. Merrill
and
Cyndi E. Merrill



Agenda

- Agenda Revisions
- Basic Reasons for an RDBMS
- What is an RDBMS?
- Some Examples of RDBMSs
- Some Common Uses of RDBMSs
- Components of an RDBMS
- What is Data Modeling?
- Data Models
- An SIRS Area 2 CUG, MS Access Example
 - A contacts spreadsheet with relationships
 - An MS Access database for the contacts
- Creating an MS Access Database



Basic Reasons for an RDBMS 1

- Easily handle more complex data than flat files / spreadsheets
 - Multiple (0, 1, 2,...) business addresses
 - Complex relationships among records, such as supervisor, supervisee, CEO,... relationships
- Perform simple queries more efficiently (rather than complex spreadsheet selects or sorts)
- Perform more complex information needs (questions asked / queries) than selects / sorts in a flat file / spreadsheet
 - Everybody living in city1 but working (i.e. having offices) in city2 and (either city3 or city4)



Basic Reasons for an RDBMS 2

- **Work only with the subset of data you need**
 - An RDBMS would store states for addresses in their own “table”.
 - If you’re working with the state, you deal only with a table of states and don’t carry along all the other data (names, etc.).
- **Allow considerably higher scalability**
 - If go from 100 records to 1 million records, you don’t have to add 999,900 states (with repetitions). Each new record simply points to the state already in the state table.
- **Simplifying storing & using complex data**
 - Instead of having the same state (say California in all the ways you’d like to display it) in every record, there is one entry (CA, California, Cal,...) in a state table, only referenced in the main table of records.
 - “District of Columbia” is 20 characters long.



What is an RDBMS? – History 1

- Started with flat files - pretty much free-form character data
- Realized there were different sorts of data
 - Went to files much like current spreadsheets
- Realized there were relationships between files
 - Put field in “primary” file which is a record # in “secondary” file



What is an RDBMS? – History 2

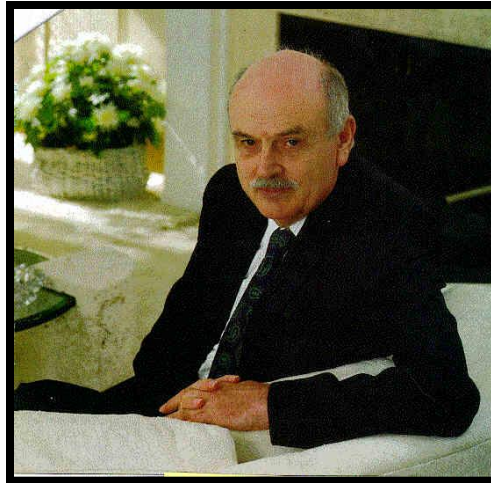
- Realized there were faster ways to move between those files
 - Created Indexed Files with “keys” between file
 - IBM’s were called ISAM & VSAM
- Realized there were more complicated relationships needed
 - Created databases, families of tables/files



What is an RDBMS? – History 3

- Realized there should be “rules” for creating “relational” databases
 - Created rules for structuring the data (normalization, E.F. Codd)
 - Created a standard visualization for the relational data structures (ER modeling, Peter Chen)
 - Created a language for handling the relational data (standardized language called SQL = “Sequel” = Structured Query Language)

What is an RDBMS? – Pioneers



■ Dr. E. F. Codd

- Recognized as the inventor of the relational data model which was the foundation for relational database management systems - 1970
- Developed the set of 12 rules governing the field of Online Analytical Processing (OLAP)
- Created "Normal Forms"
 - 1st Normal Form, 2nd Normal Form, 3rd Normal Form

What is an RDBMS? – Pioneers



■ Dr. Peter Chen

- Originator of Entity-Relationship Modeling – 1976
- Made the relationships & structure of data straightforward to handle
- Made the relationships & structure of data easy to visualize



What's an RDBMS? – Definitions 1

- In a manner of speaking, a Relational Data Base Management System (RDBMS) is a set of tables (each somewhat like a spreadsheet) with relationships (sometimes quite complex) among those tables.
 - Tables (in logical models tables are called Entities)
 - Columns in tables (in logical models called Attributes)
 - Relationships are just called Relationships



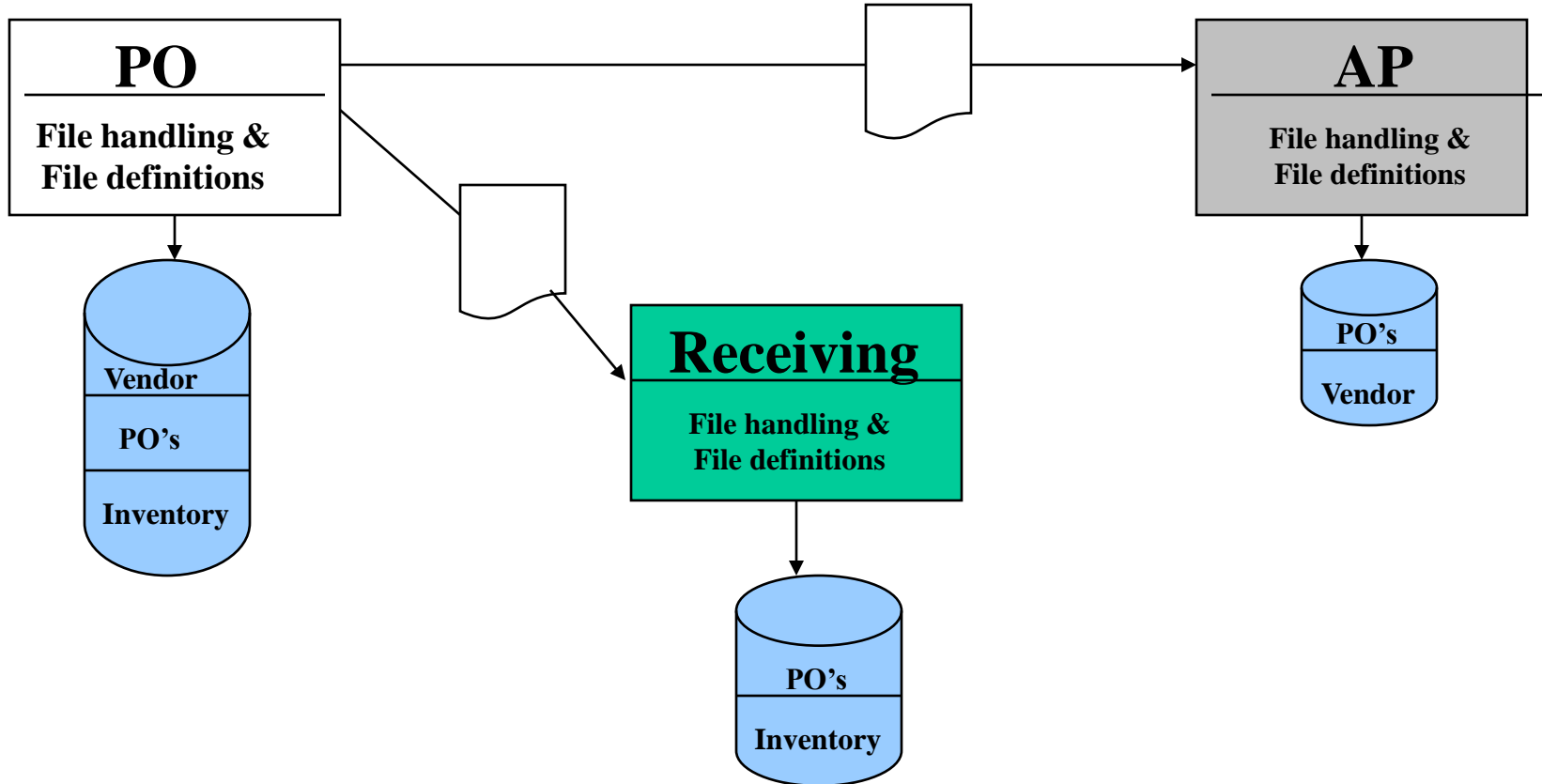
What's an RDBMS? – Definitions 2

- WordNet, Princeton University
 - relational database
 - n : a database in which relations between information items are explicitly specified as accessible attributes; "in a relational database the data are organized as a number of differently sized tables"

Also see the handout (FOLDOC definition)

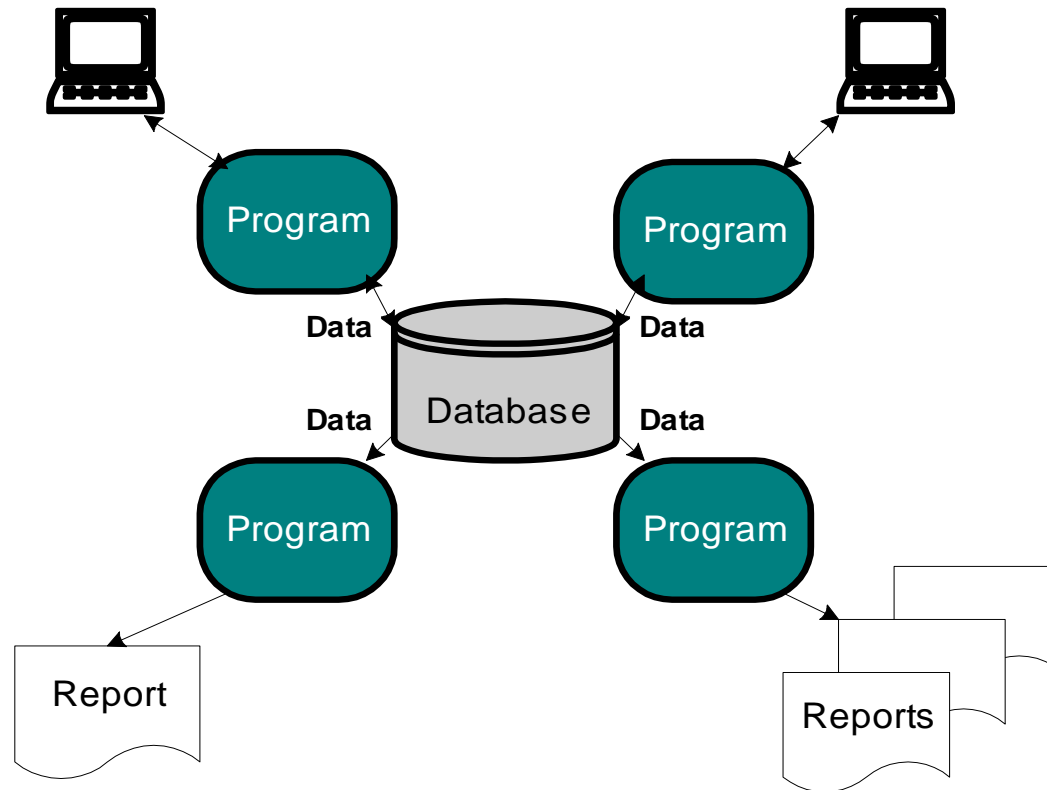
What's an RDBMS? – Basic Reason

Traditional File-Based & Spreadsheet-Based Systems Move Data Among 'Silo-ed' Applications



What's an RDBMS? – Basic Reason

Databases allow a
Data-Centered Perspective





Some Examples of RDBMSs

- MS Access
- MS SQL Server
- DB2 (/UDB)
- Oracle
- Teradata
- MySQL & SQLnet



Some Common Uses of an RDBMS

- Enterprise Resource Planning (ERP) Systems
 - SAP
 - PeopleSoft / Oracle Financials
 - Lawson
 - Baan
 - ...
- Customer Relationship Management Systems
 - Siebel
 - Salesforce.com
 - ...
- Custom Data Collection & Financial Systems
 - Commercially available Sales Order & Financial Systems
 - Peachtree, Quicken, ...
 - Home-grown financial (GL, AR, AP) systems
 - ...



Components of an RDBMS 1

- Data structures / data models of relational data base management systems contain:
 - Entities (logical) / Tables (physical)
 - Attributes (logical) / Columns (physical)
 - Relationships (implemented by Joins)



Components of an RDBMS 3

Definition of Entity

A physical database structure containing data attributes. It is also referred to as a table.

- PERSON
- PLACE
- THING
- CONCEPT
- EVENT

EMPLOYEE
EMPLOYEE_ID INT
FIRST_NAME TEXT(50)
LAST_NAME TEXT(50)
EMP_NUMBER INT
DOH DATE
DOB DATE
ADDRESS TEXT(50)
CITY TEXT(50)
STATE TEXT(50)
ZIP_CODE TEXT(50)
AREA_CODE TEXT(50)
PHONE TEXT(50)
DEPT_NO INT



Components of an RDBMS 4

Sample Data – Employee Entity

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMP_NUMBER	...	DEPT_NO
1	Edgar	Codd	100		342
2	Peter	Chen	101		456
3	William	Inmon	102		223
...
10000	Ralph	Kimball	10100		876



Components of an RDBMS 5

- Attributes/Columns
 - Like columns in a spreadsheet
 - Each column holds exactly one kind of data
 - Examples:
 - Last_Name,
 - State_Name,
 - 5Digit_Zipcode



Components of an RDBMS 6

Definition of Attribute

An attribute is a data element of an entity. It is also referred to as a data field or column within a table.

EMPLOYEE
EMPLOYEE_ID INT
FIRST_NAME TEXT(50)
LAST_NAME TEXT(50)
EMP_NUMBER INT
DOH DATE
DOB DATE
ADDRESS TEXT(50)
CITY TEXT(50)
STATE TEXT(50)
ZIP_CODE TEXT(50)
AREA_CODE TEXT(50)
PHONE TEXT(50)
DEPT_NO INT



Components of an RDBMS 7

■ Relationships

■ One-to-Many

- Each Employee Record has exactly one State
- Each State can have more than 1 Employee
 - So. Employee-to-State is One-to-Many (and Mandatory)

■ One-to-One

- Each record is related to exactly 1 record in the 2nd table

■ Many-to-Many

- Several records in the 1st table can be related to several records in the 2nd table
 - An employee might have several supervisors and a supervisor might have any number of employees
 - RDBMS have to do special things to handle many-to-many

■ To draw information from both tables, you 'Join' the tables

- Typically joins are slow

Components of an RDBMS 8

■ Definition of Relationship

- A means by which two or more tables are connected together by comparing values in specific attributes.
 - For example, Employee to Department
 - Relationship is also referred to as a join.



Components of an RDBMS 9

- Sample Data – Employee related to Department

EMPLOYEE

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMP_NUMBER	...	DEPT_NO
1	Edgar	Codd	100	...	342
2	Peter	Chen	101	...	456
3	William	Inmon	102	...	223
...
10000	Ralph	Kimball	10100	...	876

DEPARTMENT

DEPT_NO	DEPT_NAME	...	BUS_UNIT
100	Admin	...	1
223	Sales	...	5
342	R&D	...	3
456	Engineering	...	8
...
876	Education	...	11



What is Data Modeling?

- Data modeling is the task of specifying what data is to be stored, and how it is to be organized within a data base
- A “data model” is a visualization of the data structure of a relational data base

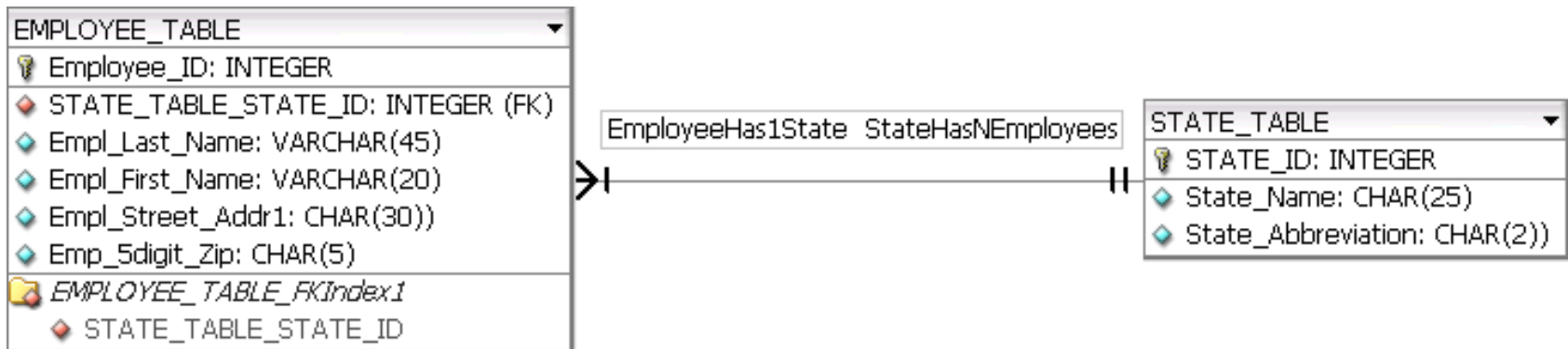
Data Model 1 - Logical



- A “Logical Model” of the Employee-State Pair
 - For simplicity, assume each employee can only have one state but a state can have many employees

Data Model 2 - Physical

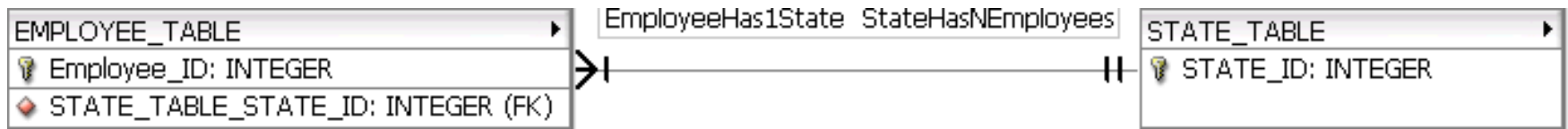
- A “Physical Model” of the Employee-State Pair
 - Add specifics on data type & size



Data Model 3 - Conceptual



- A “Conceptual Model” of the Employee-State Pair





Adding Cities to the Data 1

- Suppose you want to keep track of addresses for each record?
 - In particular, 1 home address & several business addresses
 - Would add columns to a spreadsheet to hold these relationships
- Then could answer questions / queries such as:
 - "Who lives in Walnut Creek?"
 - "Who works in San Francisco?"
- Could even do (often manually with a 2-tier sort / select)
 - Who lives in Walnut Creek and works in Oakland?



Adding Cities to the Data 2

BUT

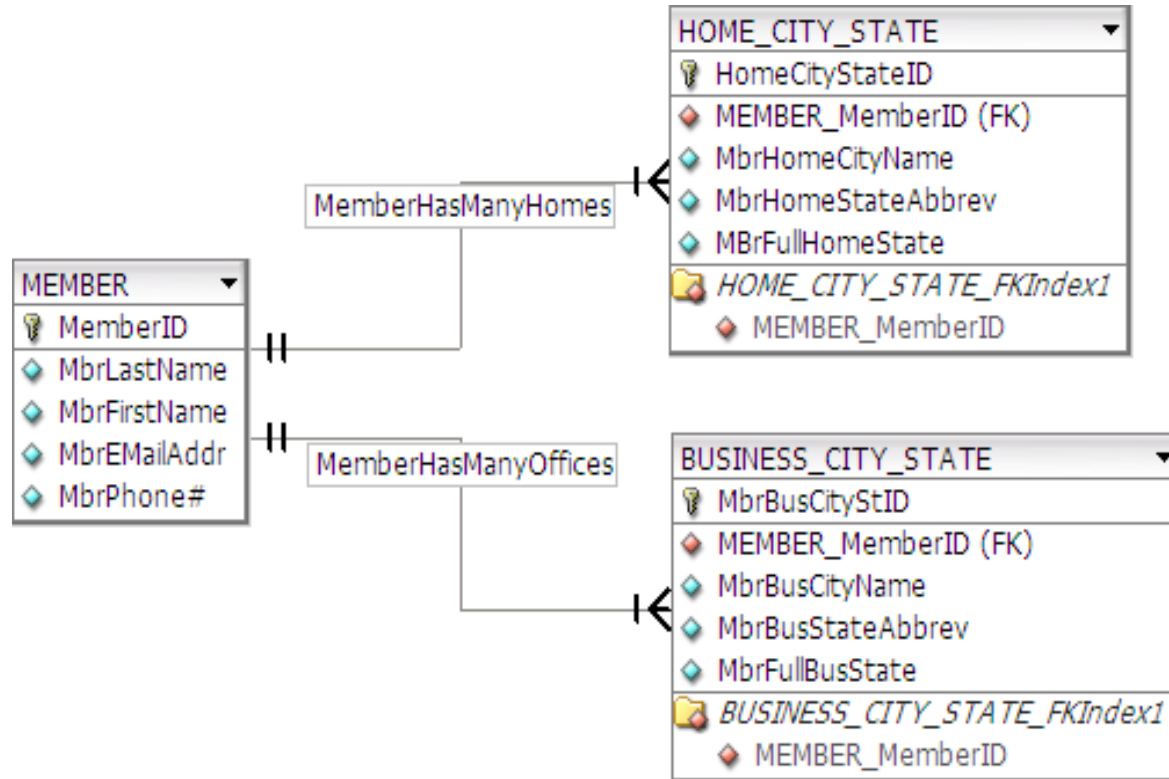
- How would you do:
 - Who lives in Walnut Creek and works in San Francisco or [in (either Oakland or Pleasanton)]?
- How would you handle more than one business &/or home addresses without limiting the #?
- The need for doing more complex queries & more than 1 set of related data drove the development of relational databases

The Contact Spreadsheet

<u>Last Name</u>	<u>First Name</u>	<u>e-mail address</u>	<u>Phone</u>	<u>home city</u>	<u>home state</u>	<u>business city</u>	<u>business state</u>
<u>Aalders</u>	<u>Doug</u>	<u>duaalders@astound.net</u>	<u>934-3684</u>	<u>Walnut Creek</u>	<u>CA</u>	<u>Sacramento</u>	<u>CA</u>
<u>Aalders</u>	<u>Doug</u>	<u>duaalders@astound.net</u>	<u>934-3684</u>	<u>Walnut Creek</u>	<u>CA</u>	<u>San Francisco</u>	<u>CA</u>
<u>Aalders</u>	<u>Doug</u>	<u>duaalders@astound.net</u>	<u>934-3684</u>	<u>Walnut Creek</u>	<u>CA</u>	<u>Oakland</u>	<u>CA</u>
<u>Andrews</u>	<u>Bill</u>	<u>bbandrews@astound.net</u>	<u>934-0375</u>	<u>Oakland</u>	<u>CA</u>	<u>Sacramento</u>	<u>CA</u>
<u>Arena</u>	<u>Matt</u>	<u>marena@astound.net</u>	<u>946-0404</u>	<u>Pleasanton</u>	<u>CA</u>	<u>San Francisco</u>	<u>CA</u>
<u>Atwater</u>	<u>Don</u>	<u>datwater@pacbell.net</u>	<u>673-1669</u>	<u>Walnut Creek</u>	<u>CA</u>	<u>Oakland</u>	<u>CA</u>
<u>Bailey</u>	<u>Jeff</u>	<u>jeffstuff@astound.net</u>	<u>673-1669</u>	<u>Oakland</u>	<u>CA</u>	<u>Sacramento</u>	<u>CA</u>
<u>Bailey</u>	<u>Jeff</u>	<u>jeffstuff@astound.net</u>	<u>673-1669</u>	<u>Oakland</u>	<u>CA</u>	<u>Oakland</u>	<u>CA</u>
<u>Bailey</u>	<u>Jeff</u>	<u>jeffstuff@astound.net</u>	<u>673-1669</u>	<u>Oakland</u>	<u>CA</u>	<u>San Francisco</u>	<u>CA</u>
<u>Baker</u>	<u>Tom</u>	<u>tbaker@astound.net</u>	<u>689-4577</u>	<u>Pleasanton</u>	<u>CA</u>	<u>Sacramento</u>	<u>CA</u>

- Comes from the CUG Attendance Spreadsheet
- Arbitrarily added City & State data
- See the full contact spreadsheet in the handout

The (Logical) RDBMS Data Model





Area 2 CUG, MS Access Example - 1

- The data in the example (the contact .xls)
- Starting MS Access
- Explaining the screens
- The model of the data in the example
- A simple query for data from the example
- A more complex query
- Entering data electronically & manually
- Queries suggested from the audience



Area 2 CUG, MS Access Example - 2

- Setting up an MS Access database
- Deciding on the data structure
- Using the data structure to
 - Establish the tables
 - Establish the relationships
- Using import to gather data
- Creating forms to gather data



Extensions beyond this Discussion

- Other useful databases
 - Tracking volunteers versus location versus schedule
 - Voter registrations
 - at Safeway 2pm-6pm with others at Albertson's from 1pm-5pm
 - Householding (storing by address with people attached)
- Uses you might not do but that might be of interest
 - Buying patterns (customer classifications)
 - Promotions by customer group
 - Running an organization by its data instead of by intuition
 - Business Intelligence / data mining (data in own structure)
- What's beyond the Relational Data Base Management System?
- Discussion questions

Conclusion + Q&A

- Looked at:
 - Definitions of RDBMS
 - Common Uses of an RDBMS
 - Components of an RDBMS
 - Creating Data Models
 - An MS Access Example
- Questions, Thoughts, Comments?

