### Relational Data Model

- Relations
- Attributes
- Tuples
- Relations
- Primary Keys
- Objectives
- Comparison to other models
- Components
- Relation Properties
- Kinds of Relations

### Relations

Introduced by Dr. Codd in 1970 based on set theory

![Relations Diagram]

### Attributes

- Attribute name
  - All attribute names must be unique within a relation.
- Attribute domain
  - A set of all possible values that can be taken by an attribute.
    - Data
    - Domain
    - Range
- Attribute values
  - Values contained currently in an attribute
- Relation degree
  - The number of attributes in a relation
Tuples

- Aggregation of attribute values
  - tuple1 = (s1, 'Jose', 21, 'Stone Mountain', 3.7)
  - tuple2 = (s4, 'Sue', 20, 'Atlanta', 2.9)
- Cardinality: The number of tuples in a relation (Not to be confused with cardinality in the E-R model)

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Age</th>
<th>Address</th>
<th>GPA</th>
</tr>
</thead>
<tbody>
<tr>
<td>s1</td>
<td>Jose</td>
<td>21</td>
<td>Stone Mountain</td>
<td>3.7</td>
</tr>
<tr>
<td>s2</td>
<td>Alice</td>
<td>18</td>
<td>Buck Head</td>
<td>4.0</td>
</tr>
<tr>
<td>s3</td>
<td>Tom</td>
<td>32</td>
<td>Dunwoody</td>
<td>3.0</td>
</tr>
<tr>
<td>s4</td>
<td>Sue</td>
<td>20</td>
<td>Atlanta</td>
<td>2.9</td>
</tr>
<tr>
<td>s5</td>
<td>Steve</td>
<td>27</td>
<td>Stone Mountain</td>
<td>3.5</td>
</tr>
</tbody>
</table>

The Objectives of the Relational Model

- Data Independence - Logical/Physical
- Communicability - The use of an easy to convey, simple structure
- Set Processing - Operations performed on “chunks” of data

Relational vs. Other Models

- Terminology - we use the terms tuple, attribute, table etc. to signify we are using a totally different model (mathematical)
- Implementation Independence - a separate logical model from physical
- Logical Key Pointers - implicit in the values of the attributes
- High-level Programming Language - e.g. query languages
- Normalization - a tool for improving the integrity of the database
Introduction to Database Management Systems

Components of the Relational Model

To be truly relational all three must be present:

- **Structural**
  - All data represented as a two-dimensional table

- **Integrity**
  - Referential - non-null foreign keys must refer to a tuple that exists
  - Entity - no part of the primary key may be null
  - User-defined - "roll your own"

- **Manipulative**
  - Operators used to transform relations to other relations
    - E.g. - Relational Algebra, Relational Calculus, QBE

Relation Properties (Structural)

- Each tuple has the same number of attributes as all other tuples in the same relation (even if a value is "null")
- Each column has a distinct name (attribute)
- Each column contains values about the same attribute
- Each attribute is "single-valued" or atomic. I.e. it can’t be further decomposed
- Each tuple is distinct or unique
- Each tuple has a unique identifier
- The order of the attributes is immaterial
- The order of the tuples is immaterial
- All attributes must contain atomic values only

Relational DBMS Rules (Codd)

0. The DBMS may be managed entirely through relational capabilities
1. The Information Rule - all data and metadata (data about data) are represented in one way, as values in a two-dimensional table
2. The Guaranteed Access Rule - in a relational database you can retrieve any value in any table (because all the tuples are unique)
3. Systematic Treatment of Null Values - e.g. a payrate of 0 is not the same as someone without a payrate (null value)
4. An active on-line catalog based on the relational model. (Users can query metadata just like data)
5. The Comprehensive Data Sub-language Rule - there is a minimum, comprehensive character-string syntax language to define the database
6. The View Updating Rule - data may be change through a new view
7. High-level insert, update, and delete - set processing also applies not only to looking at data, but also to changing data
Relational DBMS Rules (Cont’d)

8. Physical Data Independence - the relational model is a logical model, i.e., it is not constrained by implementation.
9. Logical Data Independence - the separation of views of data from schema (definition of data).
10. Integrity Independence - relational model doesn’t have built in constraints, they are to be maintained by the DBMS.
11. Distribution Independence - the users don’t need to know where the data resides.

Primary Keys

- Candidate key -- One or a group of attributes $K$ that satisfy
  - Uniqueness: No two tuples have the same value for $K$.
  - Minimality: If $K$ is composite, no component of $K$ can be eliminated without destroying the Uniqueness property.
- Primary key -- One of the candidate keys.
- Alternate keys -- Other candidate keys.

Foreign Key

- Is an attribute that references a key attribute of other relation.
- FK may or may not be an key attribute.

Employee (SSN, DriverLicenseNo, Name, Salary)
Student (SID, Name, Age, Address, GPA)
Section (sect#, instructorID, course#, time, date)
Schedule (room#, building, time, date, course#)
### Foreign Key Properties

- FK and the referenced PK attribute may have different names.
- The values of FK must draw from the value set of PK.

#### Domain

- Value set

#### Foreign Key

- Value set

### A Relation Example

<table>
<thead>
<tr>
<th>Student ID</th>
<th>Name</th>
<th>Age</th>
<th>Address</th>
<th>GPA</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Alice</td>
<td>18</td>
<td>Stone Mountain</td>
<td>3.3</td>
</tr>
<tr>
<td>2</td>
<td>Tom</td>
<td>22</td>
<td>Buck Head</td>
<td>2.8</td>
</tr>
<tr>
<td>3</td>
<td>Sue</td>
<td>20</td>
<td>Dunwoody</td>
<td>3.0</td>
</tr>
<tr>
<td>4</td>
<td>Steve</td>
<td>27</td>
<td>Atlanta</td>
<td>2.9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Instructor ID</th>
<th>Name</th>
<th>Rank</th>
<th>Department</th>
</tr>
</thead>
<tbody>
<tr>
<td>e1</td>
<td>Lisa</td>
<td>Prof</td>
<td>CIS</td>
</tr>
<tr>
<td>e2</td>
<td>Fred</td>
<td>Asst Prof</td>
<td>MATH</td>
</tr>
<tr>
<td>e3</td>
<td>John</td>
<td>Asstc Prof</td>
<td>CSc</td>
</tr>
<tr>
<td>e4</td>
<td>Abe</td>
<td>Asst Prof</td>
<td>EE</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course #</th>
<th>Credits</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIS2010</td>
<td>3.0</td>
<td>Intro to Info Systems</td>
</tr>
<tr>
<td>CIS3210</td>
<td>3.0</td>
<td>Intro to Program w. VB</td>
</tr>
<tr>
<td>CIS3215</td>
<td>3.0</td>
<td>Intermediate Visual Program</td>
</tr>
<tr>
<td>CIS3730</td>
<td>3.0</td>
<td>Intro to DBMS</td>
</tr>
<tr>
<td>CIS3300</td>
<td>3.0</td>
<td>Analysis/Design Info Systems</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class</th>
<th>Time</th>
<th>Room</th>
</tr>
</thead>
<tbody>
<tr>
<td>100-CS</td>
<td>12:10 pm</td>
<td>100-CS</td>
</tr>
<tr>
<td>521-K</td>
<td>8:00 am</td>
<td>521-K</td>
</tr>
<tr>
<td>327-ALS</td>
<td>4:30 pm</td>
<td>327-ALS</td>
</tr>
<tr>
<td>327-ALS</td>
<td>7:15 pm</td>
<td>327-ALS</td>
</tr>
<tr>
<td>521-K</td>
<td>8:00 am</td>
<td>521-K</td>
</tr>
<tr>
<td>621-G</td>
<td>4:30 pm</td>
<td>621-G</td>
</tr>
<tr>
<td>327-S</td>
<td>7:15 pm</td>
<td>327-S</td>
</tr>
<tr>
<td>100-CS</td>
<td>12:10 pm</td>
<td>100-CS</td>
</tr>
<tr>
<td>100-CS</td>
<td>12:10 pm</td>
<td>100-CS</td>
</tr>
</tbody>
</table>