BINF 8211/6211
Design and Implementation of Bioinformatics Databases
Lecture #7

Dr. D. Andrew Carr
Dept. Bioinformatics and Genomics UNCC
Spring 2016

File types and meanings

- FastQ files: Output from NGS sequencing instruments
  - Capillary sequencers produced .abi files or .fas files to show how and electropherogram was interpreted as a base string
- Standard Flowgram Files (SFF) files
- SAM/BAM files
  - BAM is the binary form of a SAM file (Sequence Alignment/Map format)
    - A tab-delimited text file with sequence alignment data
    - Indexes the read by genomic position
    - http://samtools.sourceforge.net/

FastQ files

- A text-based format for storing the base call and the quality score for a DNA sequence together. Extension is usually .fq or .fastq

- Generally you will see 4 lines/read
  - @ with an identifier and a description that is optional
  - The base calls as letters
  - A '+' and possibly the identifier repeated (optional)
  - The quality values, encoded as single ASCI characters (note that @ and + are *used in the quality string* so you have to be sure you are on the correct line).

```
@SEQ_ID
GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTTGTTCAACTCACAGTTT
+
!''*((((***+))%%%++)(%%%%).1***-+*''))**55CCF>>>>>>CCCCCCC65
```

```
@SRR001666.1 071112_SLXA-EAS1_s_7:5:1:817:345 length=36
GGGTGATGGCCGCTGCCGATGGCGTCAAATCCCACC
+SRR001666.1 071112_SLXA-EAS1_s_7:5:1:817:345 length=36
IIIIIIIIIIIIIIIIIIIIIIIIIIIIII9IG9IC
```

Examples in these slides are taken from Wikipedia, Sourceforge, SEQanswers and http://wiki.christophchamp.com/index.php/FASTQ_format

Sanger FastQ

- The first line is descriptive and arbitrarily long – SRR is a Sequence Read archive identifier – this will ALWAYS be present in NCBI files.
  - The 071112_SLXA-EAS1 is another identifier indicating the name of the platform – Solexa was bought by Illumina.
  - EAS1 etc provides meta-data about the instrument (next slides)
  - Length tells you how many base calls and quality scores to expect.

- Add 33 to the actual Phred Score, then select the ASCII character in this position. A range of 0-93 is allowed.
  - Characters 33-126 are allowed.
  - Illumina has used a number of encoding schemes and special characters in different versions, so be sure to pay attention to the version and check the manual before combining data.

```
@SRR001666.1 071112_SLXA-EAS1_s_7:5:1:817:345 length=36
GGGTGATGGCCGCTGCCGATGGCGTCAAATCCCACC
+SRR001666.1 071112_SLXA-EAS1_s_7:5:1:817:345 length=36
IIIIIIIIIIIIIIIIIIIIIIIIIIIII9IG9IC
```

```
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
 |                         |    |        |                              |                     |
33                        59   64       73                            104                   126
 0........................26...31.......40
```

Illumina FastQ

- The output from Illumina platforms provides a lot of information in the identifier.

```
@HWUSI-EAS100R:6:73:941:1973#0/1
```

HWUSI-EAS100R	the unique instrument name
6	flowcell lane
73	tile number within the flowcell lane
941	'x'-coordinate of the cluster within the tile
1973	'y'-coordinate of the cluster within the tile
#0	index number for a multiplexed sample (0 for no indexing)
/1	the member of a pair, /1 or /2 *(paired-end or mate-pair reads only)*

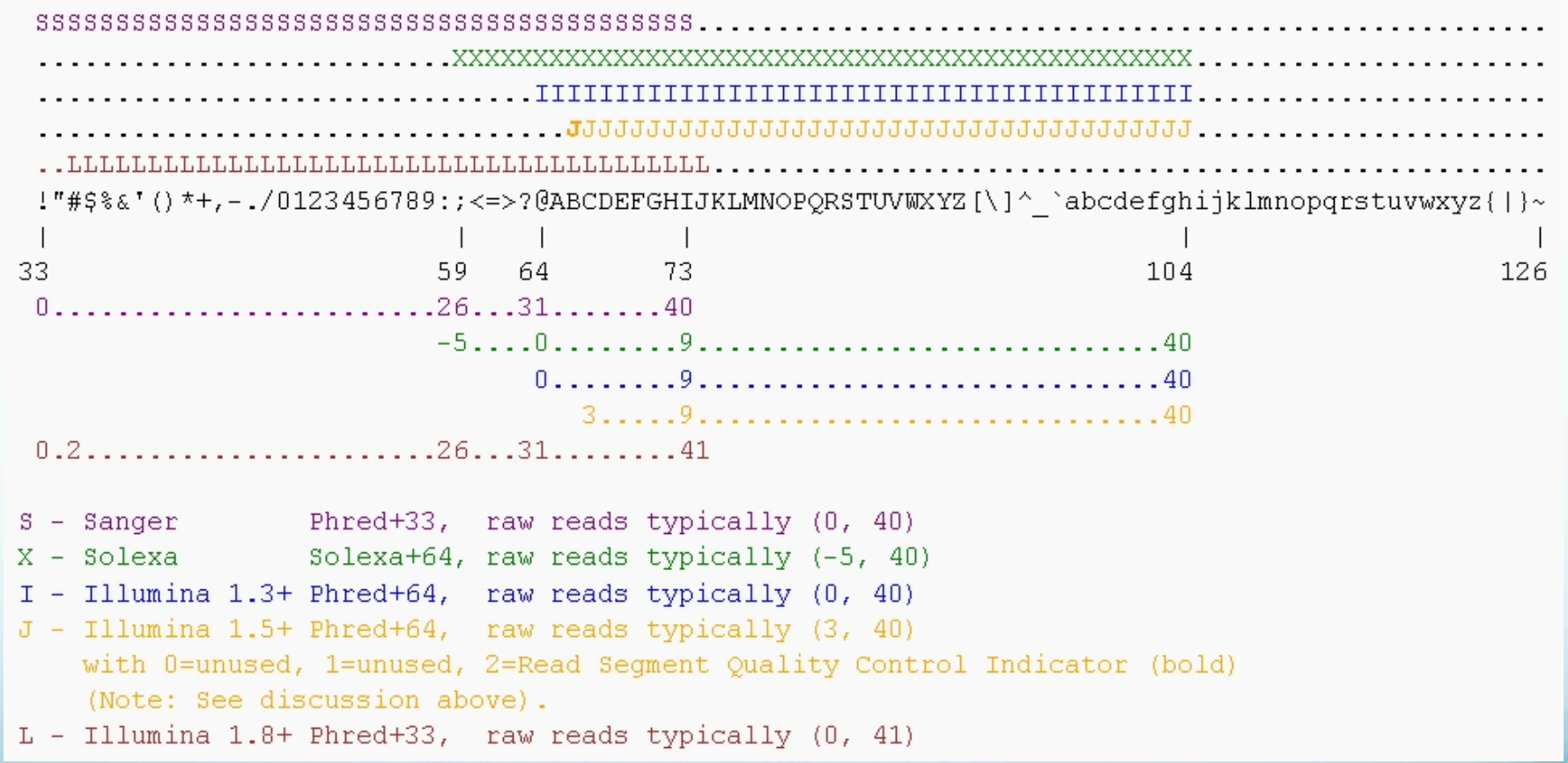
Versions of the Illumina pipeline since 1.4 appear to use **#NNNNNN** instead of **#0** for t

With Casava 1.8 the format of the '@' line has changed:

```
@EAS139:136:FC706VJ:2:2104:15343:197393 1:Y:18:ATCACG
```

EAS139	the unique instrument name
136	the run id
FC706VJ	the flowcell id
2	flowcell lane
2104	tile number within the flowcell lane
15343	'x'-coordinate of the cluster within the tile
197393	'y'-coordinate of the cluster within the tile
1	the member of a pair, 1 or 2 *(paired-end or mate-pair reads only)*
Y	Y if the read is filtered, N otherwise
18	0 when none of the control bits are on, otherwise it is an even number
ATCACG	index sequence

Q-value encodings



Sanger format encodes a Phred quality score from 0-93 using ASCII 33-126, as does SAM

Warm-up question

- When is a linking table added to a model?

Linking Table Review -- Bridge Entities

- When you have a M:N relationship in your conceptual model, implementation will require that you decompose this to two 1:M relationships by adding another entity.
  - The new entity is called a Bridge or Composite Entity.
    - In spreadsheet land AKA the Linking Table.
  - The Bridge Entity PK is a composite of the PK of each parent entity set – when an attribute from another table it is called a Foreign Key. It this is the primary key of that table it will be unique and not null.
    - PK (FK1 NOT NULL, FK2 NOT NULL) where '1' is a named entity and '2' is a named entity.
    - There is an identifying relationship between the bridge entity and each parent (notation uses solid line).
    - The bridge entity is *existence-dependent* on the two parent entity sets.
    - The bridge entity can have additional attributes.

Linking Table example

Table Name: Enzyme

Protein_ID	Protein_Name
DAA20347	Chymotrypsin
AGI80160	Elastase

Table Name: ProteinTarget

Protein_ID	Protein_Name	Enzyme_Name
AAB59562	Keratin	Chymotrypsin
AAA98797	Albumin	Chymotrypsin
NP_000549	Hemoglobin, alpha subunit	Chymotrypsin
DAA20347	Chymotrypsin	Chymotrypsin
NT2NL_HUMAN	Notch Homolog2	Elastase

proteolyzes

Table Name: Enzyme

Protein_ID	Protein_Name
DAA20347	Chymotrypsin
AGI80160	Elastase

Table Name: Enzyme-ProteinTarget

PK, FK on Enzyme	**Protein_ID**
PK, FK on ProteinTarget	**Target_ID**
	CDD_ID

Table Name: ProteinTarget

Target_ID	Protein_Name	Enzyme_Name
AAB59562	Keratin	Chymotrypsin
AAA98797	Albumin	Chymotrypsin
NP_000549	Hemoglobin, alpha subunit	Chymotrypsin
DAA20347	Chymotrypsin	Chymotrypsin
NT2NL_HUMAN	Notch Homolog2	Elastase

Conceptual vs Relational Models

- What is the difference?

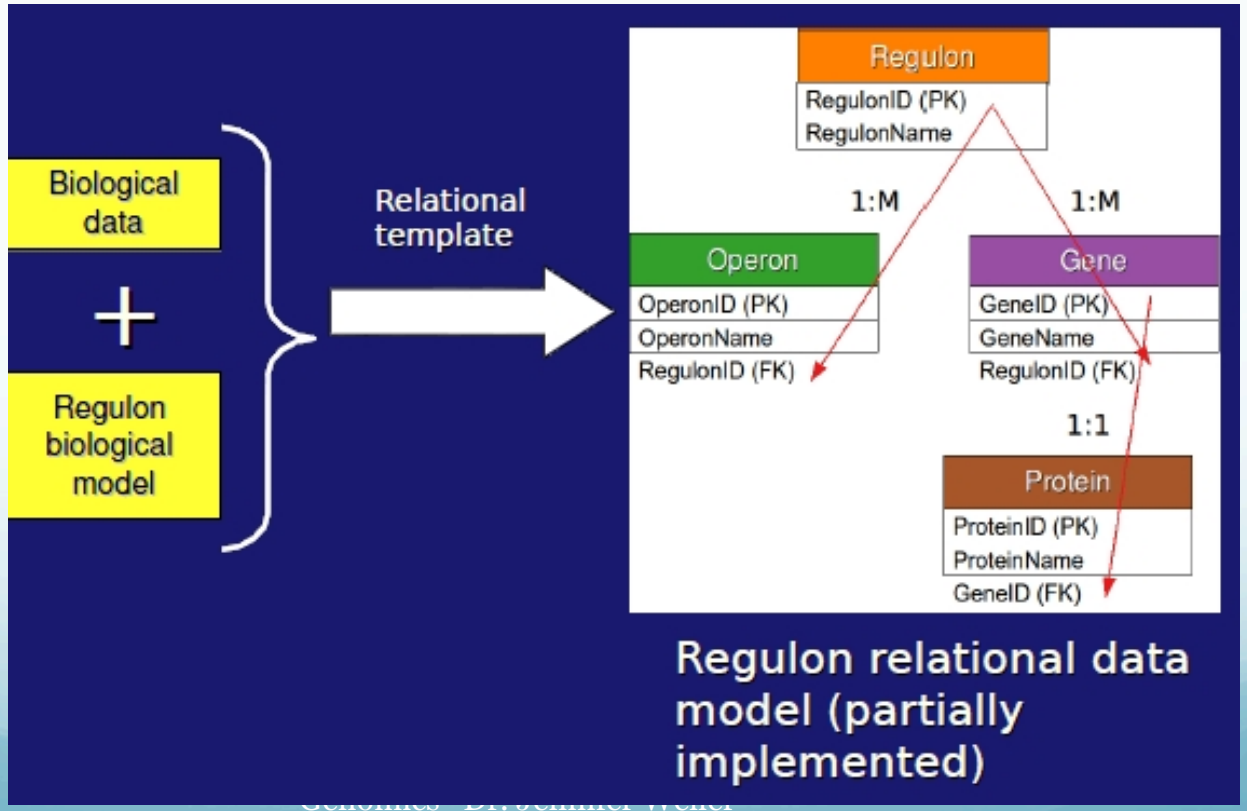
Data model levels

- Conceptual data modeling, AKA the Use Case – highest level relationships and rules. It includes
  - The scope of the model – WHAT is stored
  - Used for communicating – standard symbols and texts are employed (ER modeling, UML) to developers and users.
  - The processes, such as inputs (and sources), outputs (applications that produce these), transformation steps (for example a reference location based on creating a SAM file)
- Logical data modeling - defines the elements and their characteristics and the relationships that interconnect them.
- Physical modeling - application of the logical data model using database management software (DBMS) – how data are stored.

- Note: the logical data model explicitly determines the structure of data and limits the type of DBMS that will be effective.

The relational database has its theoretical underpinnings in set theory and predicate logic, and was proposed by EF Codd (1970).

- Practical use increased as both hardware and software systems were developed.



A relation is represented graphically as a table, where the entities and their attributes can be listed.

Each row in a relation (table) is called a tuple

Entity (table) names must be unique

Attribute (column) names must be unique within the table

Each row must have uniquely identifiable by using a combination of its attributes (primary key, composite keys); this means no duplicate rows

Columns are constrained to a specific range of values (domain). For example, RegulonID is defined as an integer.

Regulon

RegulonID	RegulonName
1	Regulon A
2	Regulon B

Operon

OperonID	OperonName
1	Operon A
2	Operon B

Gene

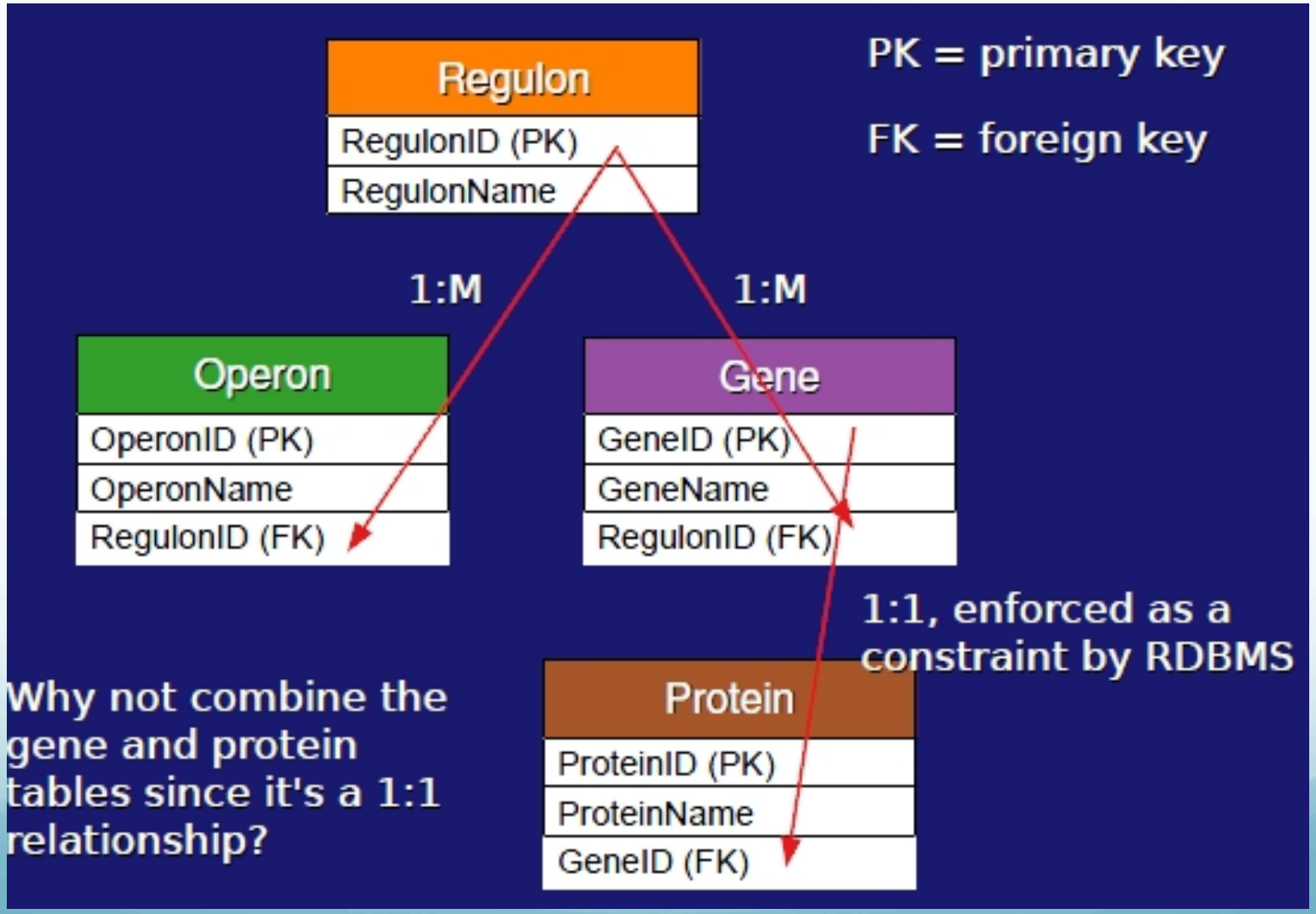
GeneID	GeneName
1	Gene A
2	Gene B
3	Gene C
4	Gene D

Protein

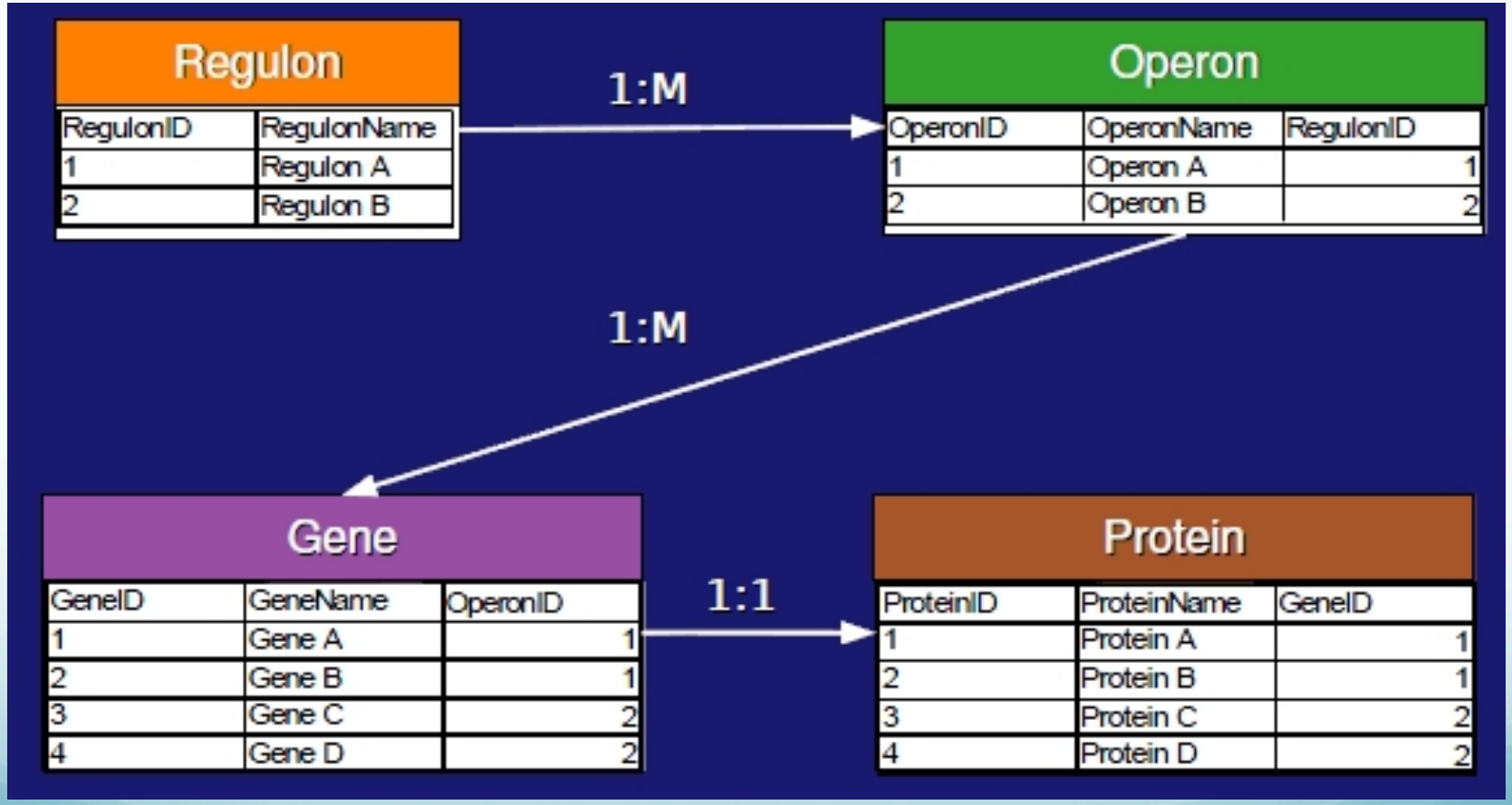
ProteinID	ProteinName
1	Protein A
2	Protein B
3	Protein C
4	Protein D

A relational table stores a collection of related entities

Relationships (between relations) are created using controlled redundancy, as shown.



A many-to-many relationship can be represented through decomposing the tables.



Codd's 12 Relational Database Rules: 1-5

Rule order	Rule Name	Text
1	Information	All information is logically represented by column values in rows in tables
2	Access Guarantee	Every table value must be accessible via {Table name|PK value|Column name}
3	Nulls Handled systematically	Nulls must be handled systematically & independent of data type
4	On-line catalog based on relational model	Meta-data must be stored, handled and available as any other data, in db tables
5	Data sublanguage enabled	The DBMS must support at least one defined, declarative language for DDL and DML tasks and for transaction management

Rule order	Rule Name	Text
6	View updating	If a view is theoretically updatable the system must be able to do it
7	Set-level insert, delete, update	The database must support set-level data functions
8	Physical Data Independence	Application programs and tools will be logically unaffected when the physical access method, or storage structures, is changed
9	Logical Data Independence	Application programs and tools will be logically unaffected when table structures are changed (order of columns or adding new columns) if original values are preserved
10	Integrity Independence	Integrity constraints are stored in the database system catalog

Rule order	Rule Name	Text
11	Distribution Independence	End users and application programs are not affected (and do not need awareness of) data location
12	Cannot subvert integrity rules	It is not possible to bypass integrity rules of the database, at any access level
	Rule Zero – no mis-naming	To be termed a relational database, only relational facilities can be used in its management.

Note – at this point there are many more rules (300?) that have been stated, although I have not found a compact listing.

Rule 1: The relation (table) is the fundamental construct in the relational model

A table is perceived as a 2D structure, having rows and defined columns (logical structure)

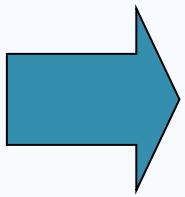
A relational table is a collection of entities of the same type (a set) of attributes

Table = Entity set = tuples

Related entities have the same attributes, with different values; think class vs. object in object-oriented programming

Rule 2: a key consists of one or more attributes that uniquely identifies a table row.

The key's role is based on a concept known as determination



A → B

Given the value of A, we can determine the value of B

Composite key

(A, B) → C

Given the values of A and B, we can determine the value of C

Any attribute that is part of a key is called a key attribute

Rule 3: Nulls must be handled systematically & independent of *data type*.

Data types

Data types are system assignments that tell the system how much space a value will require and what operations are valid.

Static data type assignments will produce an error if unrecognized elements are entered.

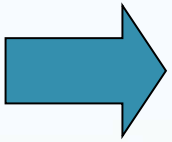
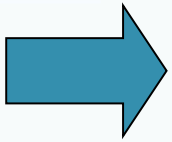
Dynamic data type assignments let you enter any value, although your business rules may not allow you to do anything sensible with them (SQLite uses dynamic typing).

The NULL data type

A NULL is not the same thing as a zero. Either missing information or Non- applicable information should be specifically represented and handled in a way that is distinct from other information, and treated very systematically.

SQL uses NULL for *both* missing and inapplicable, which is not Codd's intent (there should be distinct rules) but it works in practice.

Rule 10: A relational database maintains its integrity using primary and foreign keys

Entity integrity		A table does not contain duplicate entity instances; enforced by the primary key requirement
Referential integrity		A foreign key value can't exist without a corresponding primary key value in another table

In practice, integrity constraints can be violated by a poor database design.

SQL allows multisets (duplicate rows) in violation of Codd's rule.

Some common data types

Data Type	Format	Description
String	CHAR(n)	Fixed-length character string of user-specified length n, up to 255 characters.
String	VARCHAR(n)	A variable-length character string with user-defined maximum length n
Numeric	INT	An integer (finite set that is machine-dependent)
Numeric	SMALLINT	A small integer
Numeric	NUMERIC(p,d) Or DECIMAL(p,d)	A fixed-point number with user-specified precision. The number has p digits (plus a sign) and d of those p digits are to the right of the decimal point.
Numeric	REAL, DOUBLE-PRECISION	Floating-point and double-precision floating point numbers (machine-specific restrictions)
Numeric	FLOAT(n)	A floating-point number, with precision of at least n digits
Special	DATE	Julian calendar date (allows proper calculations)

Rule 4: Meta-data must be stored, handled and available as any other data, in db tables

The data dictionary or catalog in the RDBMS is used to define this information, and the system tables store the information.

This makes the system self-defining.

Assignment #1 – Collating data Bioinformatics

- Choose an animal:
  - Use the web to acquire the following.
    - Meta data around the animal
    - Two nucleotide sequence file from the animal.
    - Translation of some of the sequence
    - Structure (file) for at least one of the proteins (from translation)
    - Data about a path way the sequence is involved in.
  - You will need to keep track of every website you use.
  - Where you pull the data from.
  - The format of the data.
    - Be able to explain what common data types you will use to store the information.

Assignment #1 – Collating data Bioinformatics

- What you will turn in (round 1)
  - Name of animal
  - Name of sequence files and their types
  - Name of the protiens
  - Name of the pathways
  - All of the bibliographic information surrounding your acquisition.
    - Website used.
    - Date data was uploaded. – Source of the data
    - Name of the format data is stored in File type.
- The due at 8:00 a.m February 18th
- Need to have the files for class February 18th.