Introduction to Relational Databases

Toni Espinosa

Departament of Computer Architecture and Operating Systems, UAB

Outline

- What is a database?
- Tables and relations
- The entity-relationship concept
- Database design from entities and relationships
- Tables and primary keys

What is a database?

- A collection of data
- A set of rules to manipulate data
- A method to mold information into knowledge
  - Is a yellow pages book a database?
  - Is a yellow pages phone service a database?

Why are databases relevant?

- Provide means of consistently extracting knowledge from data
- Solution to manipulate large data sets efficiently
- Can integrate multiple data sources

Why scientific relational databases?

- Large collections of annotated data
- Public databases provide cross-links to other databases
- Individual research lab databases need to integrate public data of interest
- Human activity is constantly being measured and stored

How can a database be useful?

- Provide data analysis language and tools
- What if we used folders and Excel files for our data?
- “Data Analysts” phone number in yellow pages
  - Manually: Look for D pages, then A, then T, ...
  - Linux: grep “data analysts” /tmp/yellow_pages/*.*
  - DB: SELECT * FROM yellow_pages
    WHERE **profession**=“data analyst”

Searches are usually complex

Find all data analysts with experience in Linux:

- Manually: read all descriptions of all data analysts
- Linux: program that reads all yellow_pages files to extract lines with data analysts then find features
- Database
  - SELECT last name
  - FROM yellow pages
  - WHERE **skills** LIKE “%linux%”

Objectives of learning DB systems

- Conceptualize data in terms of relations
- Design/understand relational databases
- use SQL to build and manage databases
- use SQL language to extract data from databases

Flat files vs relational DB

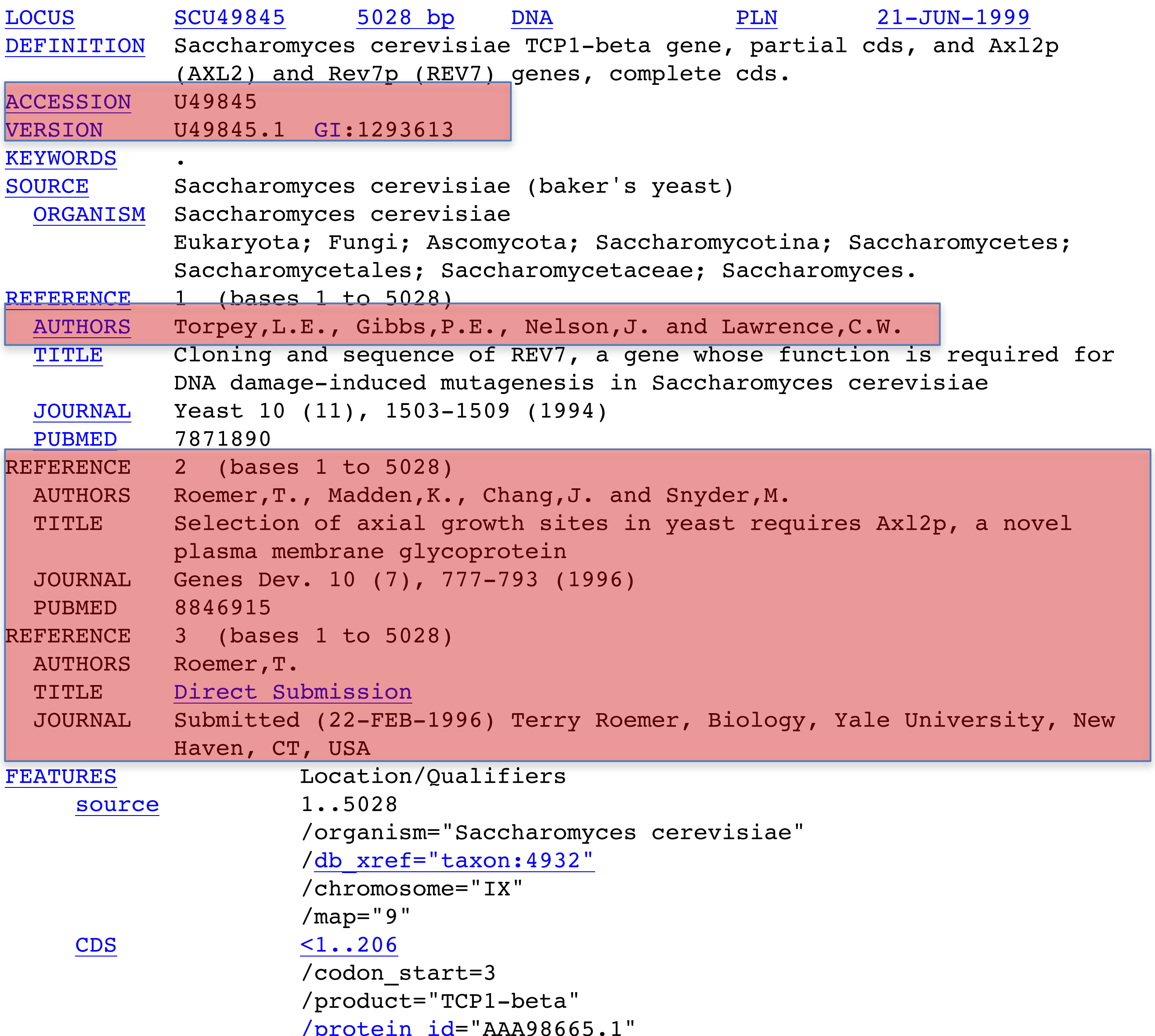
- Flat files use delimited ad-hoc formats to describe data and categories item by item
  - Flat files or custom formats require specific parsers and filters (usually done in Python ->JSON)
- Relational databases store data in terms of their relationship to each other
  - A simplified data manipulation language (DML) can extract information from any database with any design

https://en.wikipedia.org/wiki/Data_manipulation_language

Typical format: JSON (source: google maps)

```
{    "markers":  [
      {
        "name":  "Rixos The Palm  Dubai",
        "location":  [25.1212, 55.1535],
      },
      {
        "name":  "Shangri-La  Hotel",
        "location":  [25.2084, 55.2719]
      }
    ]
}
```

GenBank format



```
LOCUS       SCU49845     5028 bp    DNA             PLN       21-JUN-1999
DEFINITION  Saccharomyces cerevisiae TCP1-beta gene, partial cds, and Axl2p
            (AXL2) and Rev7p (REV7) genes, complete cds.
ACCESSION   U49845
VERSION     U49845.1  GI:1293613
KEYWORDS    .
SOURCE      Saccharomyces cerevisiae (baker's yeast)
  ORGANISM  Saccharomyces cerevisiae
            Eukaryota; Fungi; Ascomycota; Saccharomycotina; Saccharomycetes;
            Saccharomycetales; Saccharomycetaceae; Saccharomyces.
REFERENCE   1  (bases 1 to 5028)
  AUTHORS   Torpey,L.E., Gibbs,P.E., Nelson,J. and Lawrence,C.W.
  TITLE     Cloning and sequence of REV7, a gene whose function is required for
            DNA damage-induced mutagenesis in Saccharomyces cerevisiae
  JOURNAL   Yeast 10 (11), 1503-1509 (1994)
  PUBMED    7871890
REFERENCE   2  (bases 1 to 5028)
  AUTHORS   Roemer,T., Madden,K., Chang,J. and Snyder,M.
  TITLE     Selection of axial growth sites in yeast requires Axl2p, a novel
            plasma membrane glycoprotein
  JOURNAL   Genes Dev. 10 (7), 777-793 (1996)
  PUBMED    8846915
REFERENCE   3  (bases 1 to 5028)
  AUTHORS   Roemer,T.
  TITLE     Direct Submission
  JOURNAL   Submitted (22-FEB-1996) Terry Roemer, Biology, Yale University, New
            Haven, CT, USA
FEATURES             Location/Qualifiers
     source          1..5028
                     /organism="Saccharomyces cerevisiae"
                     /db_xref="taxon:4932"
                     /chromosome="IX"
                     /map="9"
     CDS             <1..206
                     /codon_start=3
                     /product="TCP1-beta"
                     /protein_id="AAA98665.1"
```

But flat files are not relational

Mix of content and structure:

- Data type is part of the data
- Record order is important
- Records contain duplicated data items:
  - source/organism info in genbank
- Some records are hierarchical
  - Records contain multiple subrecords (authors)
- There is an implicit use of a key only clear to experts

Relational databases

- Build data management system on top of data entities relationships: ready for new data adquisition
- Databases are made of tables and links between them
- A data language is used for querying the database (SQL - /sequel/)
- The system that manages the tables and links is called a Data Base Management System (DBMS)

DBMS ACID

ACID model of databases

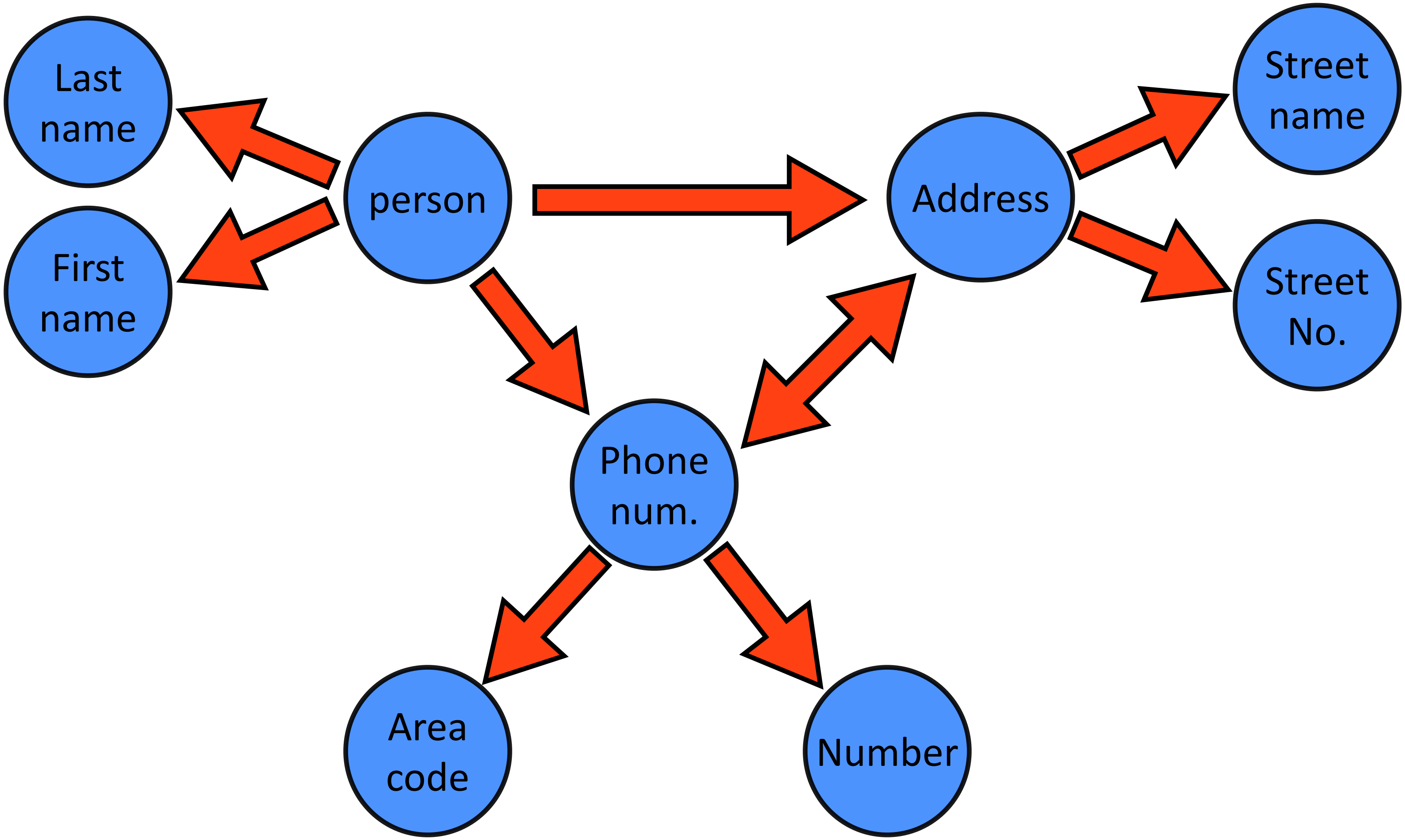
- **Atomicity**: All transactions proceed or fail. “All or nothing”
- **Consistency**: Only valid data can be part of the database
- **Isolation**: Any concurrent execution of transactions will produce the same result as generating them one after the other
- **Durability**: Once a transaction is committed, it will remain so even after any error or problem

Well known DBMS

- MySQL/MariaDB
  - World most popular DBMS
  - Popular in open source LAMP software stack: now mariadb.org
  - Property of Oracle since 2009
- PostgresSQL
  - Open Source DMBS
  - Large Linux Support, MacOS since Lion
  - Object Oriented
- Oracle
  - High end DBMS for complex data models
  - Huge amount of available functionality
  - License is around $40K per CPU
  - Evaluation purposes is free

Data conceptualization: from data to DB

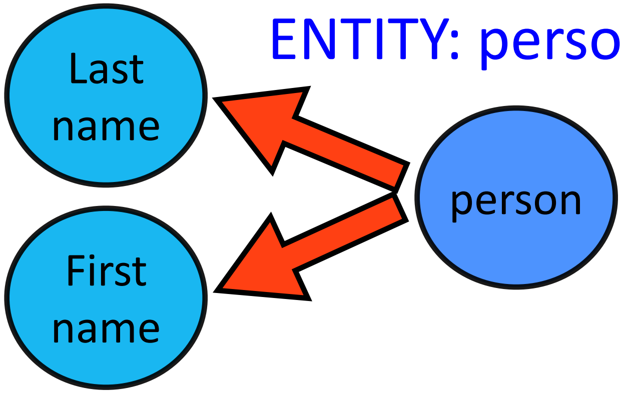
- Phone book application data model



Structuring data into tables

- Data is stored in tables with multiple columns (attributes)
- Each record is a row of our table (tuple)

ENTITY: person



Attributes

First name	Last name
John	Smith
David	Waterson
Andrew	Locke

Tuples

What’s in a table?

- Tables are relations where operations are applied to
- All rows should be different
- Each attribute for a tuple has only one value
- Tuples within a table are not sorted
- Each tuple is identified by a unique number named ***Primary Key***

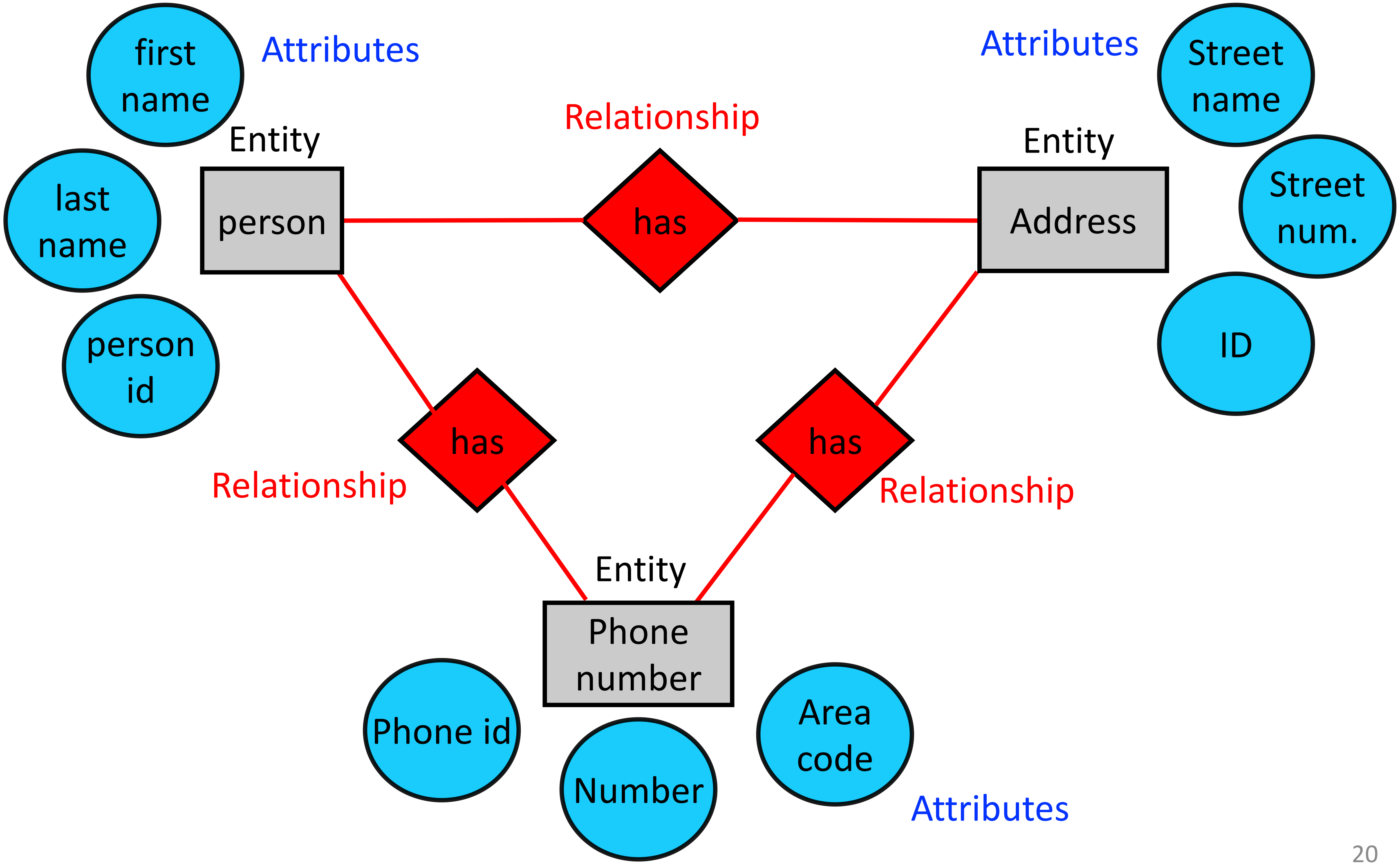
ID	First name	Last name
1	John	Smith
2	David	Waterson
3	Andrew	Locke

Database basic design principles

How do we create a database from a given data source?

1. Find out the data elements: the entities
2. Draw relationships between entities
3. Make schema simple
4. Avoid redundancy
5. Make sure the design describes the data accurately

Database table design example



Entities become our first tables

ENTITY

ID	fname	lname

PERSON

ENTITY

ID	str_no	str_name

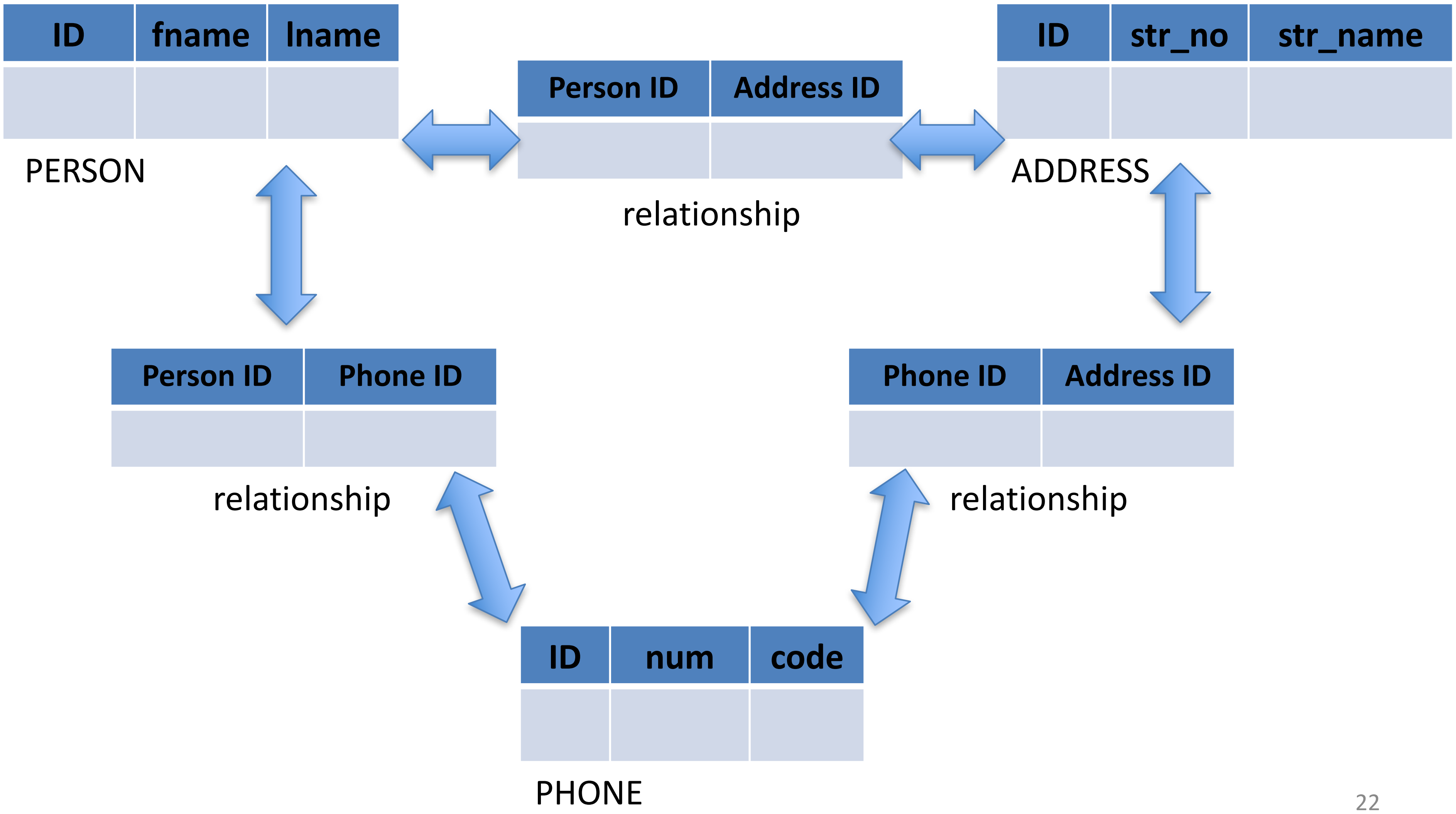
ADDRESS

ENTITY

ID	num	code

PHONE

Entity-relationships to DB tables



ID	fname	lname

PERSON

Person ID	Address ID

relationship

ID	str_no	str_name

ADDRESS

Person ID	Phone ID

relationship

Phone ID	Address ID

relationship

ID	num	code

PHONE

What have we done:
Entity-Relationship Diagram

1. Identify data attributes
2. Conceptualize entities by grouping related attributes
3. Identify relationships/links
4. Draw preliminary Entity-Relationship diagram
5. Add cardinalities and references

WORK!:
Normalizing our author data

Article data set review

Post date: 24 Jan 2017

Content type: Article

Author: Stefano Maffulli

Title: Maffulli,Brotli: A new compression algorithm for faster Internet

Comment count: 12

Path:/article/17/1/brotli-compression-algorithm

Tags: Internet

Word count: 590

Objective: extract a database design from a flat data file

Which tables, attributes, relationships?

Which of these features are related?

Post date: 24 Jan 2017
Content type: Article
Author: Stefano Maffulli
Title: Maffulli,Brotli: A new compression algorithm for faster Internet
Comment count: 12
Path:/article/17/1/brotli-compression-algorithm
Tags: Internet
Word count: 590

Which features are related?

- Post date: 24 Jan 2017
- Content type: Article, Poll
- Author: Stefano Maffulli
- Title: Brotli: A new compression algorithm for faster Internet
- Comment count: 12
- Path:/article/17/1/brotli-compression-algorithm
- Tags: Internet, Business, Programming
- Word count: 590

First step: identify entities and attributes

- Date, title, comment count, word count, path
  - describe characteristics of the post
- Content type
  - Defines a category of different content
- Authors
  - Name of authors
- Tags
  - Defines a category of tags for an article

Second step:
can you name entities from the list?
A <entity> is defined by: <attributes>

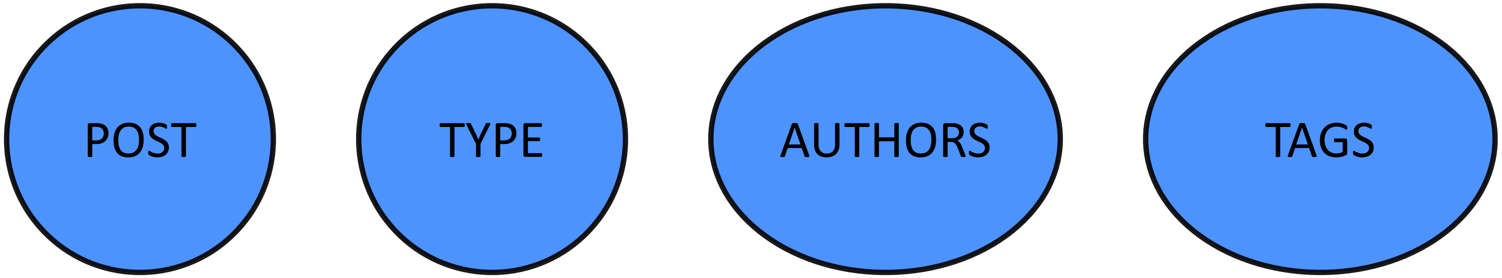
- Date, title, comment count, word count, path
  - describe characteristics of the post
- Content type
  - Defines a category of different content
- Authors
  - Name of authors
- Tags
  - Defines a category of tags for an article

2nd step: identify entities by grouping attributes

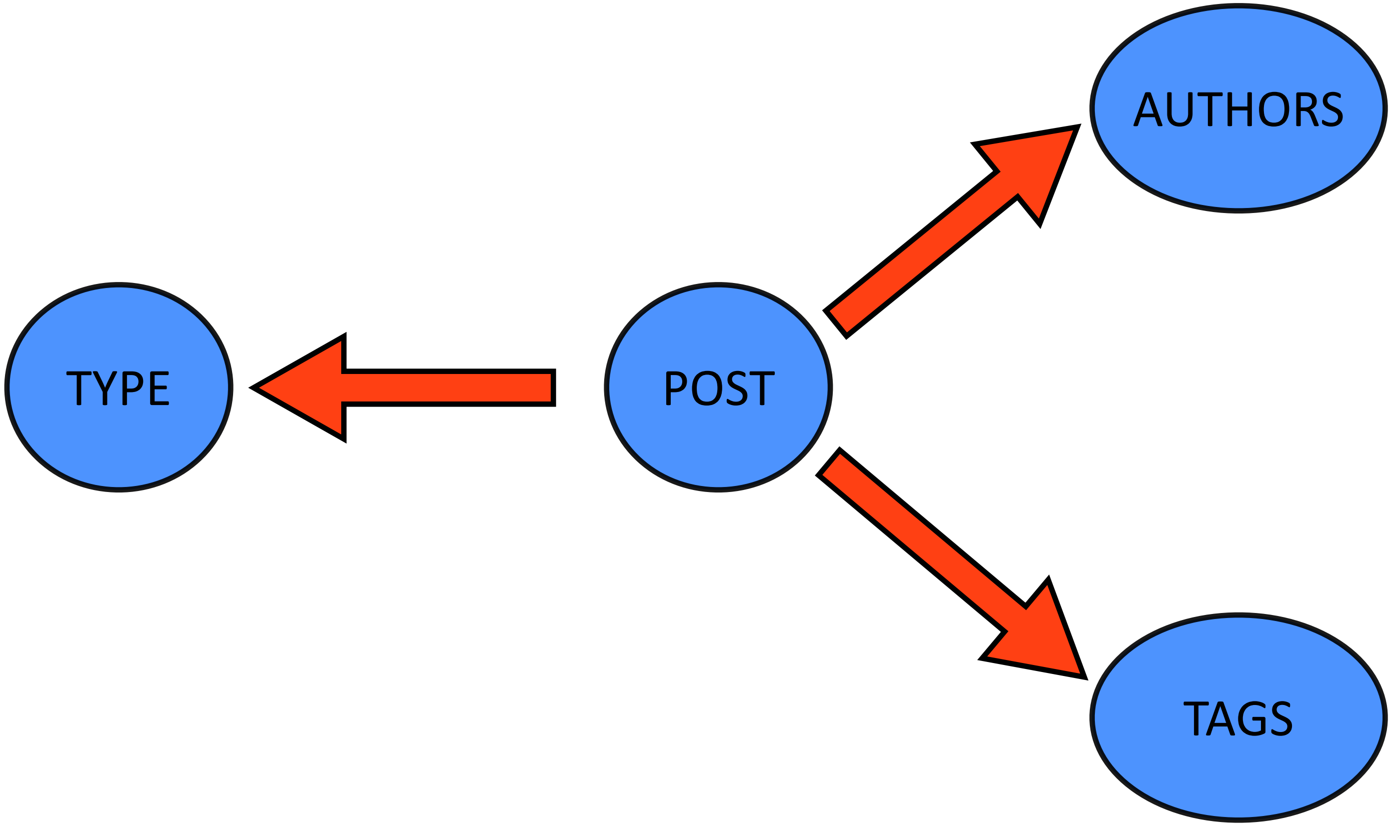
- A post is described by:
  - A title, counts, a date of creation and a path
- Content type is described by:
  - A list of content categories
- Authors are described by:
  - Name and surname of authors
- Tags are described by
  - A list of text labels

Can you draw individual entities and their links?

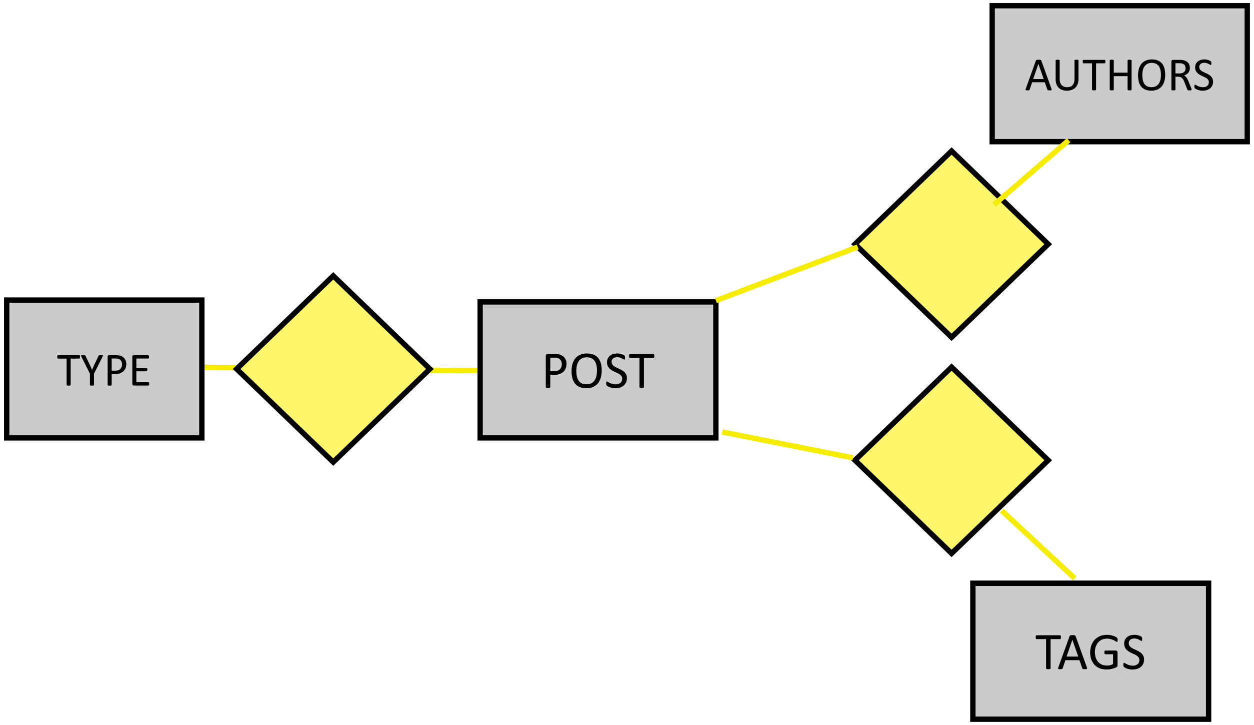
3rd step: draw individual entities



Which are the relationships between our entities?

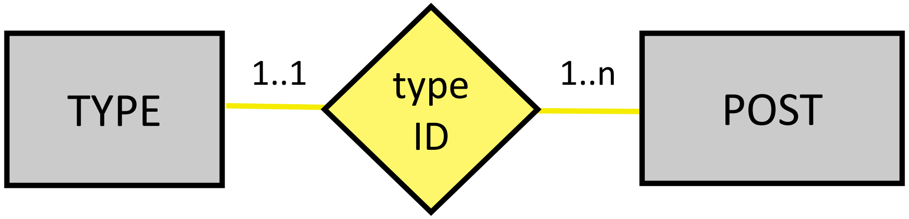


Draw entity relationships



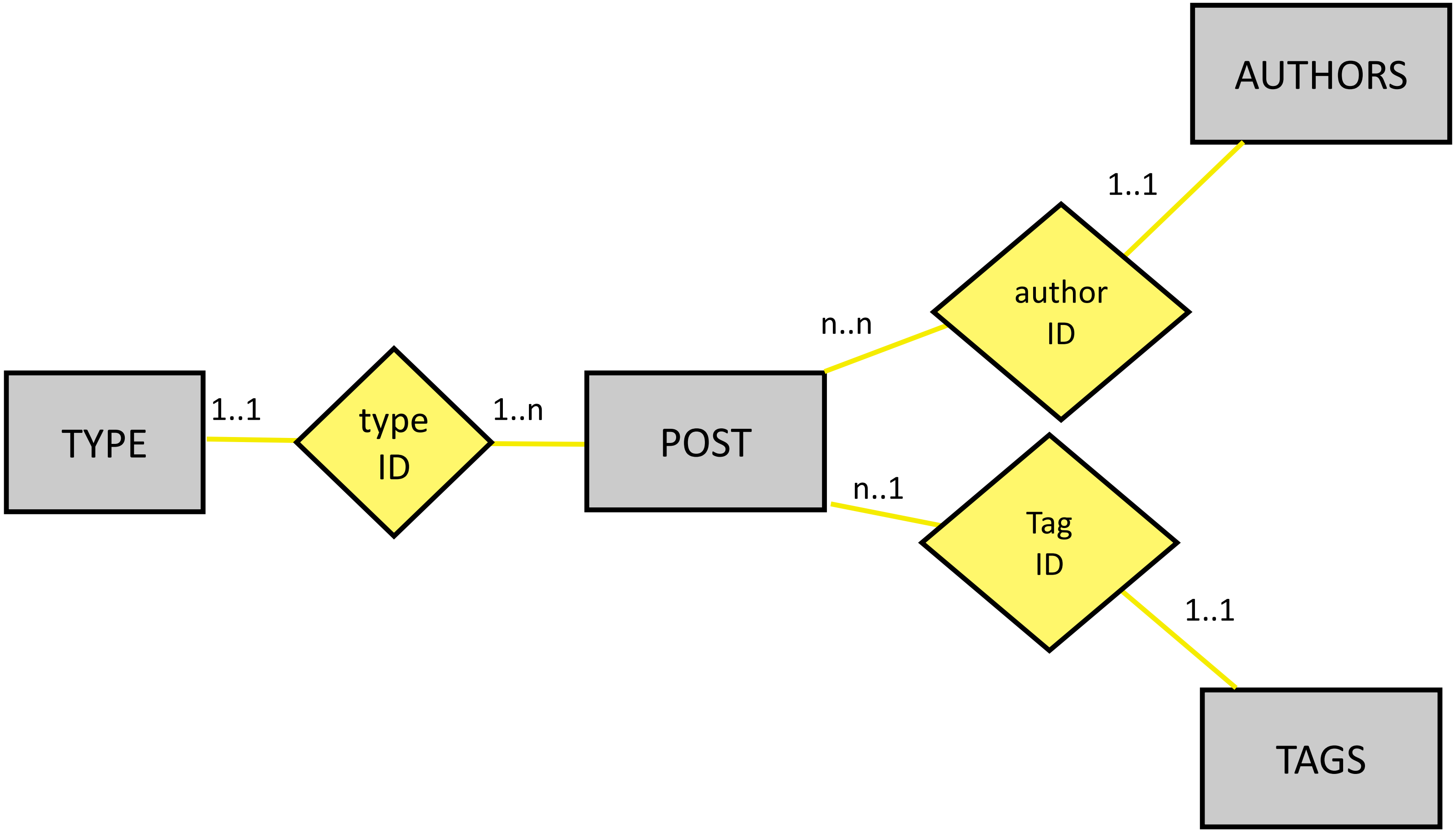
Add cardinalities and references

- One type category is associated to one type Id: 1->1
- One type category can be found in many posts: 1->n
- Each individual post contains just one type: 1->1

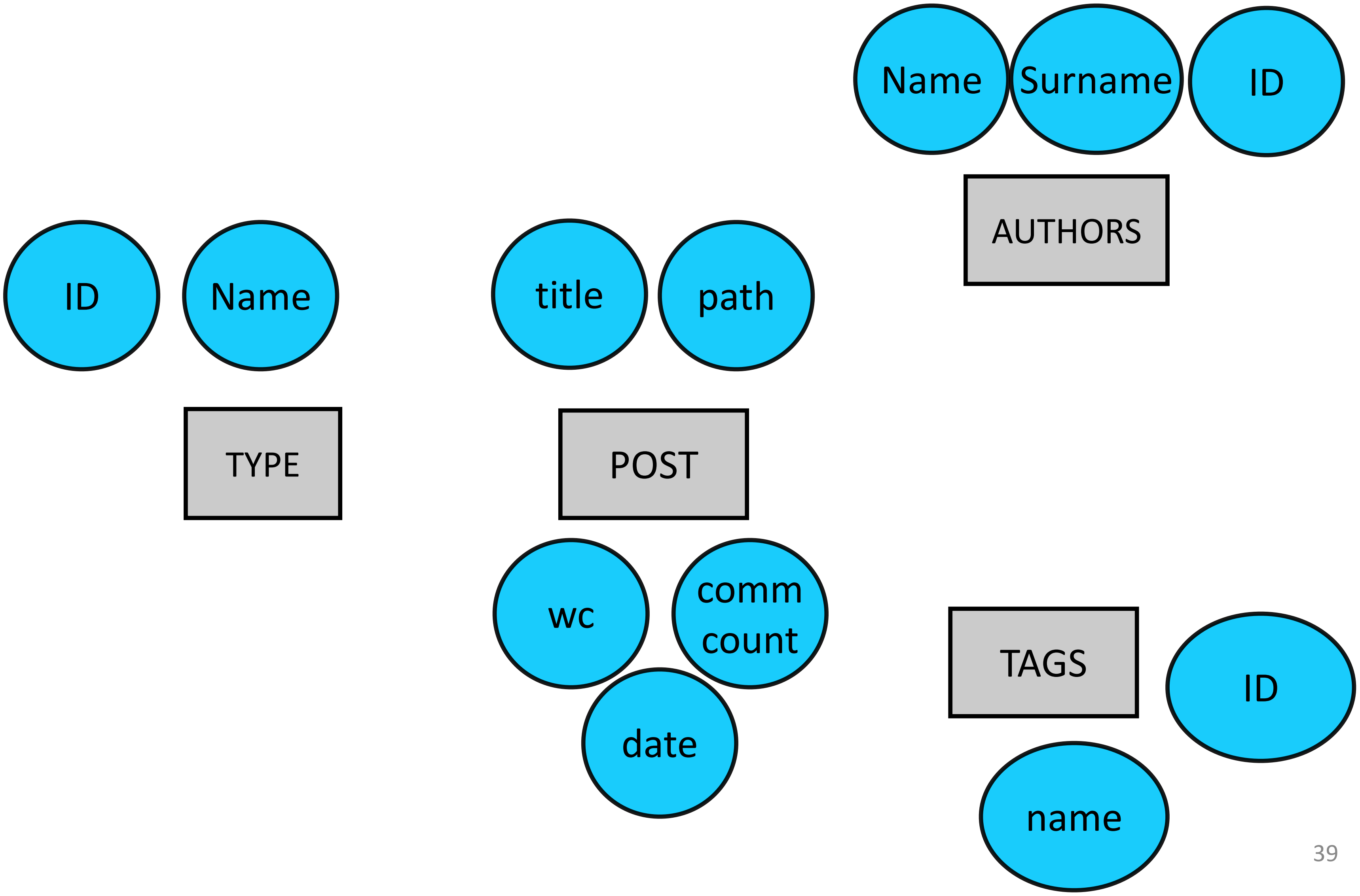


Can you draw relationship cardinalities?

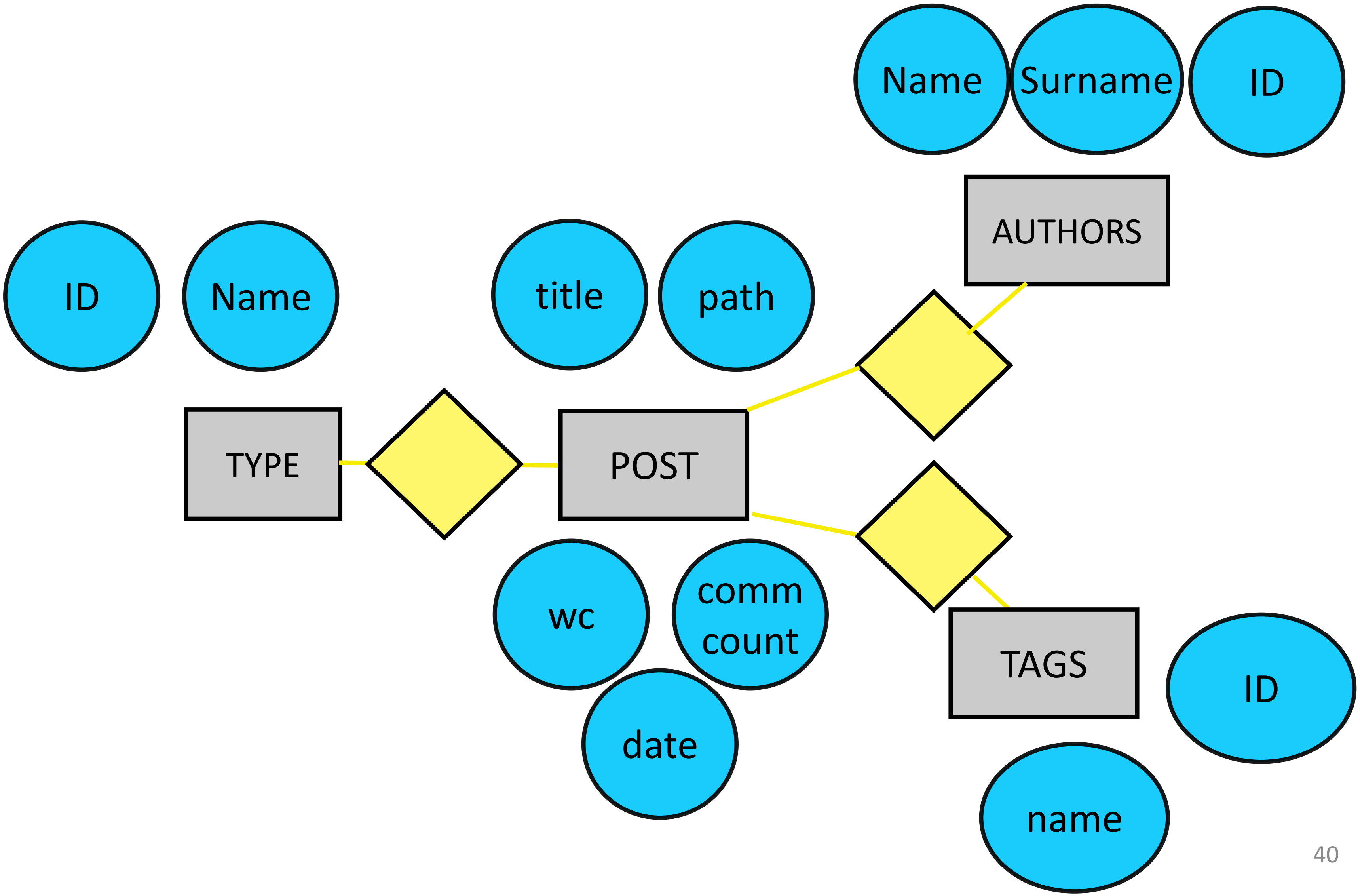
Add cardinalities and references



Draw entity attributes



First Entity-Relationship diagram



Summary

- Databases follow ACID design principles
- Databases are made of tables that describe relations
- Relations are entities that have attributes and tuples
- Databases can be designed from Entity-Relationship diagrams that are easily converted to tables
- Primary keys define unique individual tuples and represent links between tables

Exercise: design your own database

GeneName	GeneDescript	GeneBankId		
BRCA1 GBE1	Collagen Collagen	L02870 S75295		
LocusId	**LocusDescr**	**GeneBankId**		
1294 2632	Glucan Glucan	L02870 S75295		
Tissue	**Experiment**	**Value**	**SampleId**	
Liver Liver	1 1	12 67	sample1 sample256	
GO ID	**GO Descr**	**GeneBankId**		
0005202 0003844	Serine Proteine Glucan Enzyme	L02870 S75295		
Experiment	**GeneBankId**	**SampleId**	**Species**	
1 1	L02870 L02870	Sample1 sample256	Human Human	

Some nomenclature

- Gene Ontology: described function database
  - *Reference: Donna Magglot: "Gene: a directory of genes". The NCBI Handbook. http://www.ncbi.nlm.nih.gov/books/NBK21085*

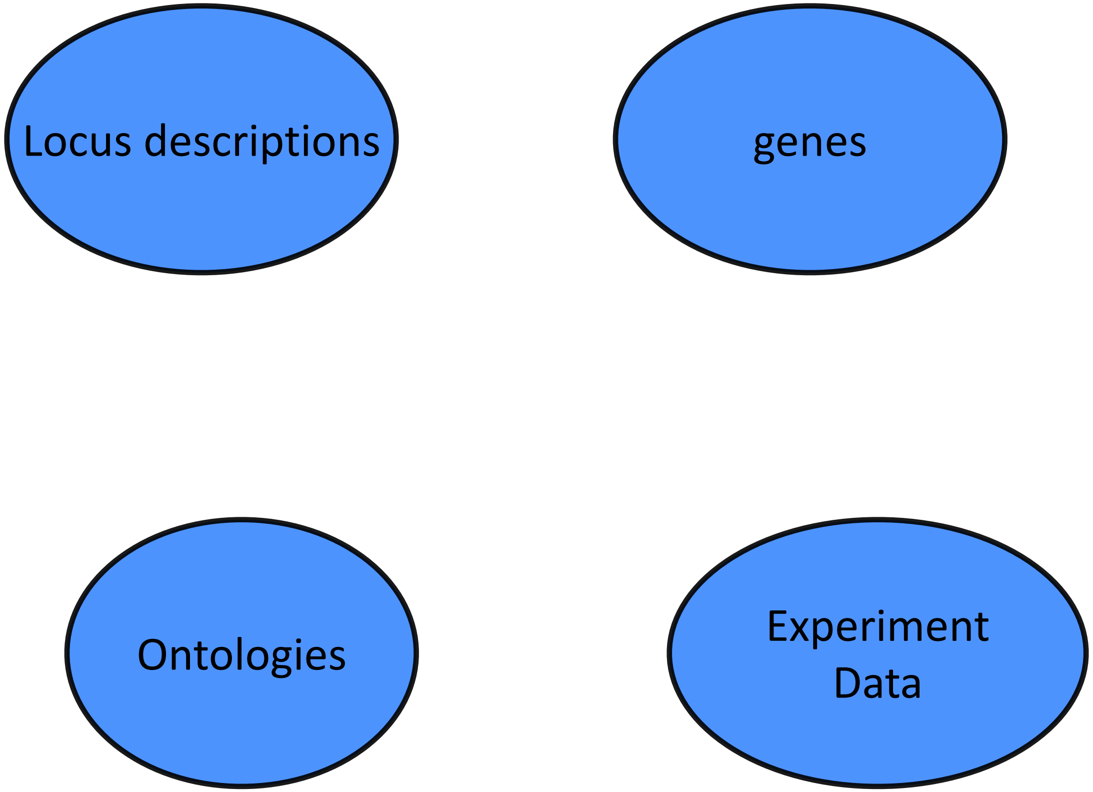
List of attributes

- GeneName, GeneDescription, GeneBankId
- LocusId, LocusDescription,GenBankId
- GO ID, GO Description, GeneBankId
- Tissue, Experiment, Species, Sample Id,GeneBankId

Group attributes to find entities

- Gene descriptions
  - Name, description, GenBankId
- Ontologies
  - GO Id, GO description
- Locus
  - Locus id, locus description
- Experiment data
  - Sample species, Experiment number, expression value, tissue

Entities of our diagram



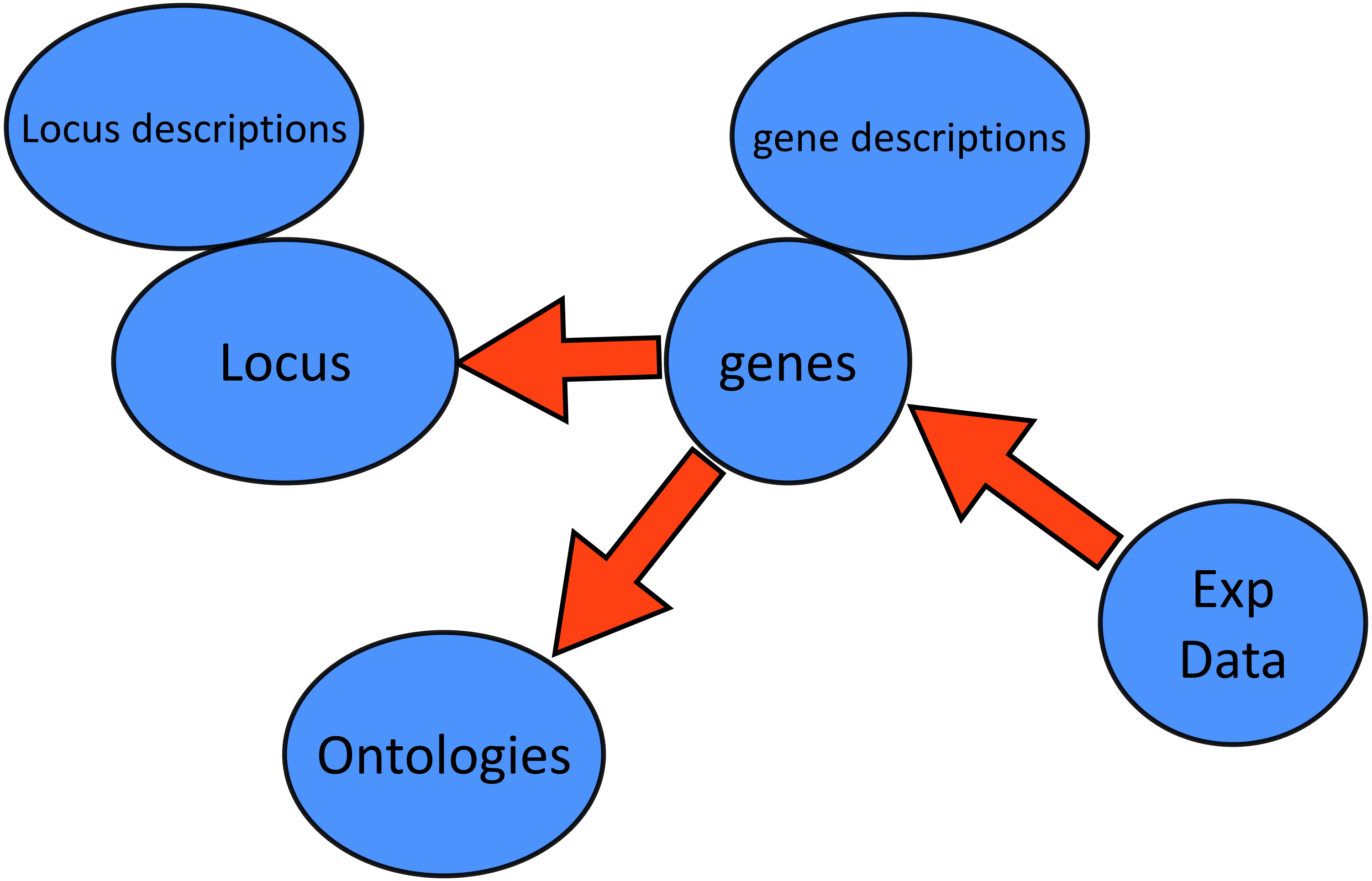
Locus descriptions

genes

Ontologies

Experiment Data

Draw entity relationships in diagram



Locus descriptions

gene descriptions

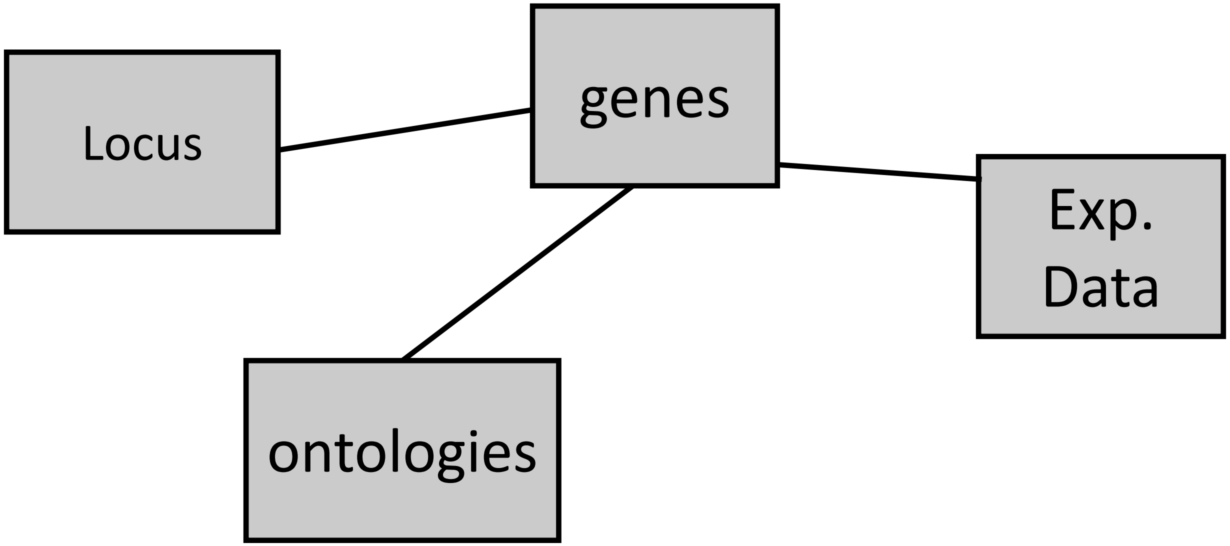
Locus

genes

Ontologies

Exp Data

Entity-Relationship diagram



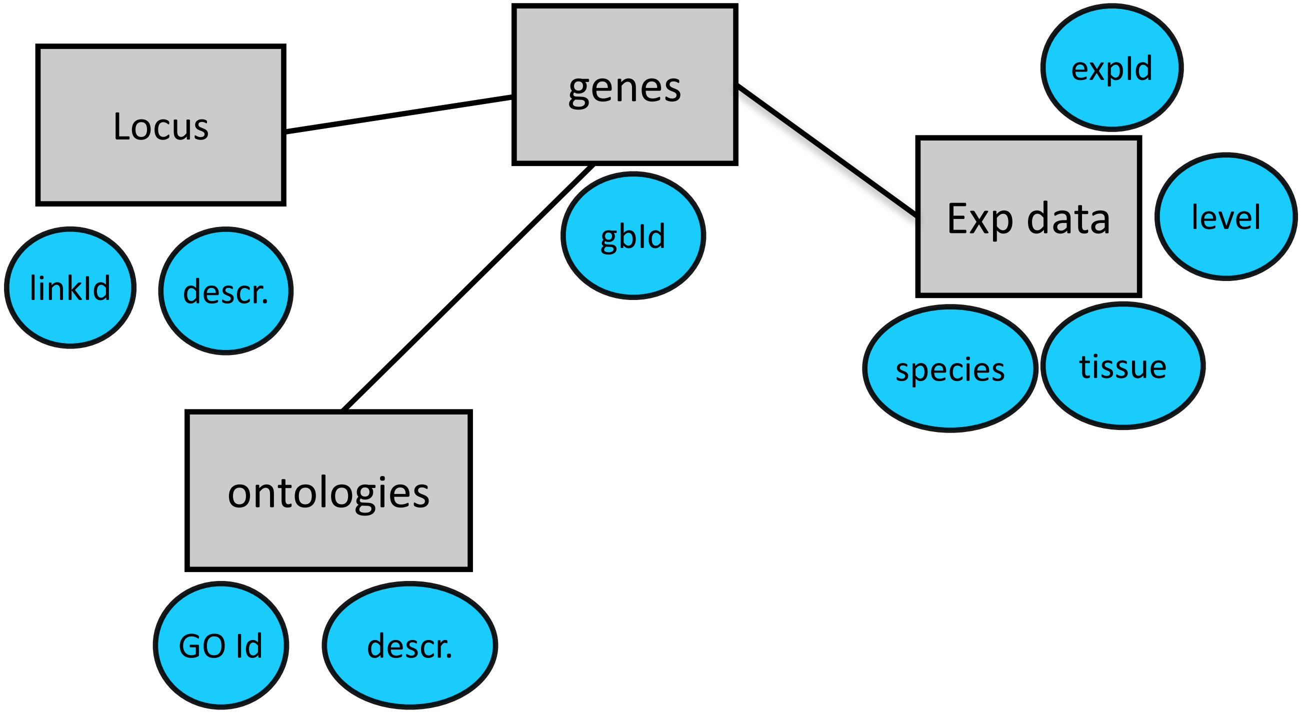
Locus

genes

Exp.
Data

ontologies

E-R diagram

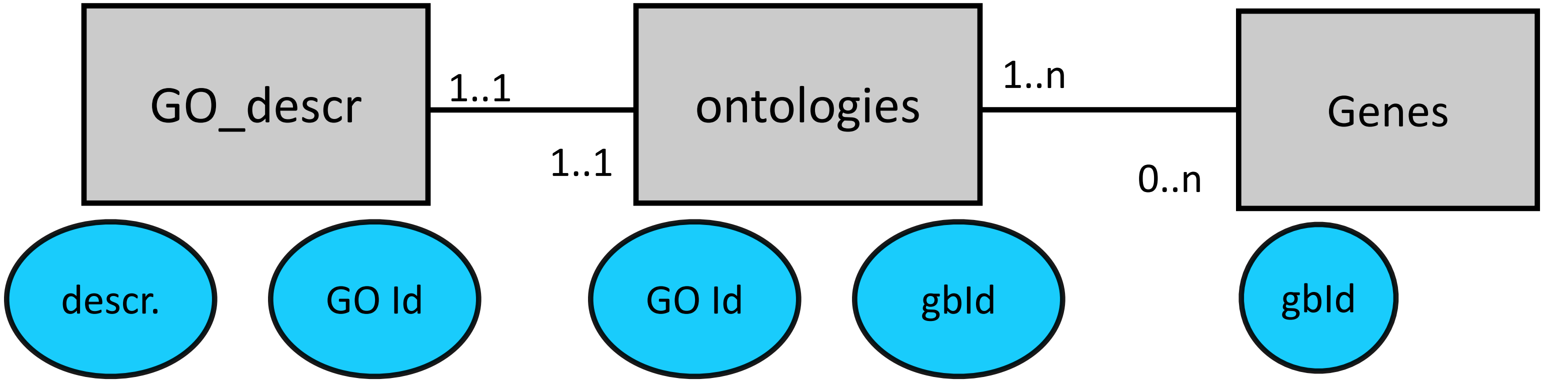

Locus
genes
Exp data
expId
level
linkId
descr.
gbId
species
tissue
ontologies
GO Id
descr.

Entities and relationships

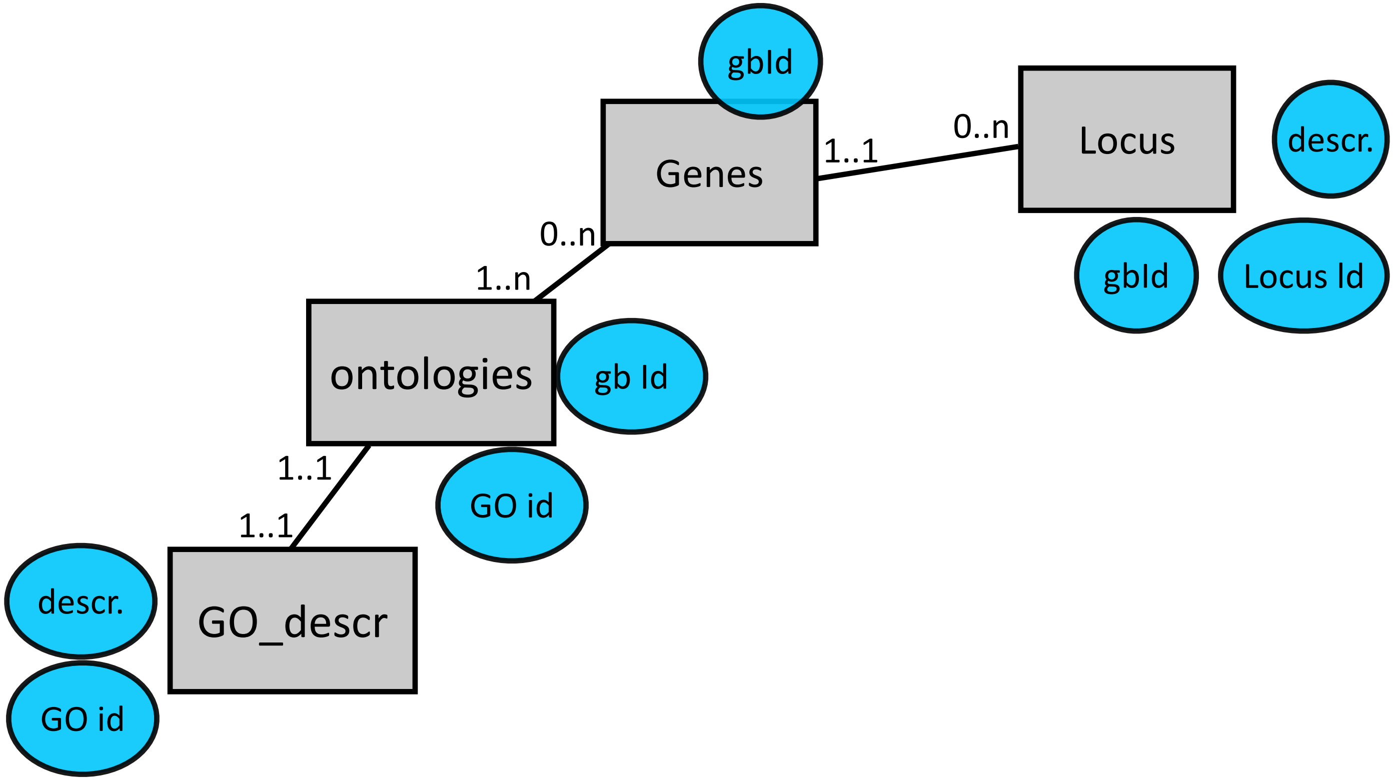
- Table **Ontologies** is not well normalized: descriptions are repeated for some ontology terms
- How do we decide what to split out?
  - Try to identify entities already existing within the data
  - Imagine all possible relationships between them

E-R analysis of ontologies table

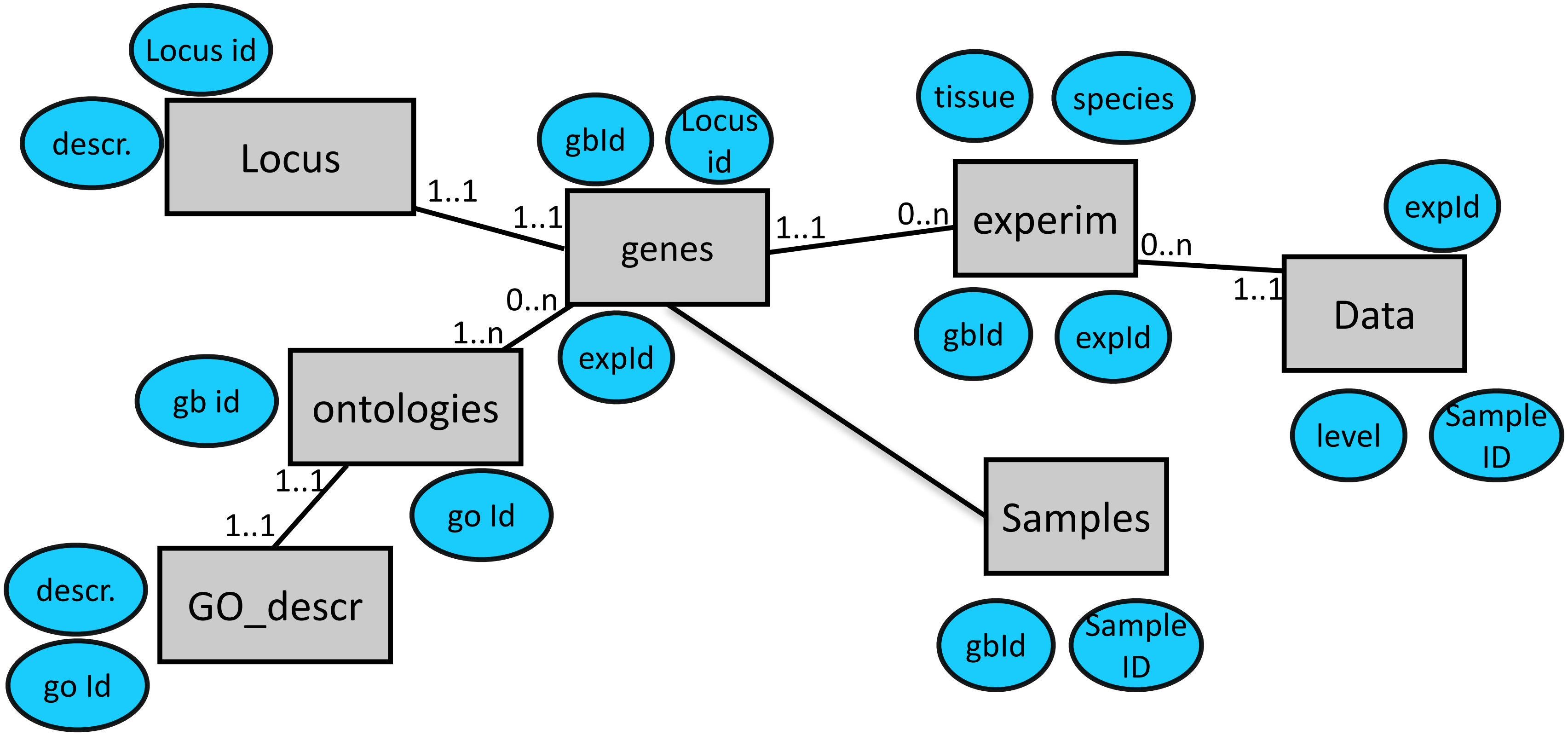
- Entities: Genes and GO Id identification numbers
- Relationships
  - one gene can have many ontology annotations
  - one ontology annotation has to one annotation GO id
- Create two tables: ontologies and GO descriptions



Redesign E-R graph for *GO_descr*



E-R new design


Locus id
descr.
Locus
1..1
1..1
gbId
Locus id
genes
1..1
0..n
tissue
species
experim
0..n
expId
Data
1..1
0..n
1..n
expId
gbId
expId
gb id
ontologies
level
Sample ID
1..1
go Id
1..1
Samples
descr.
GO_descr
go Id
gbId
Sample ID