

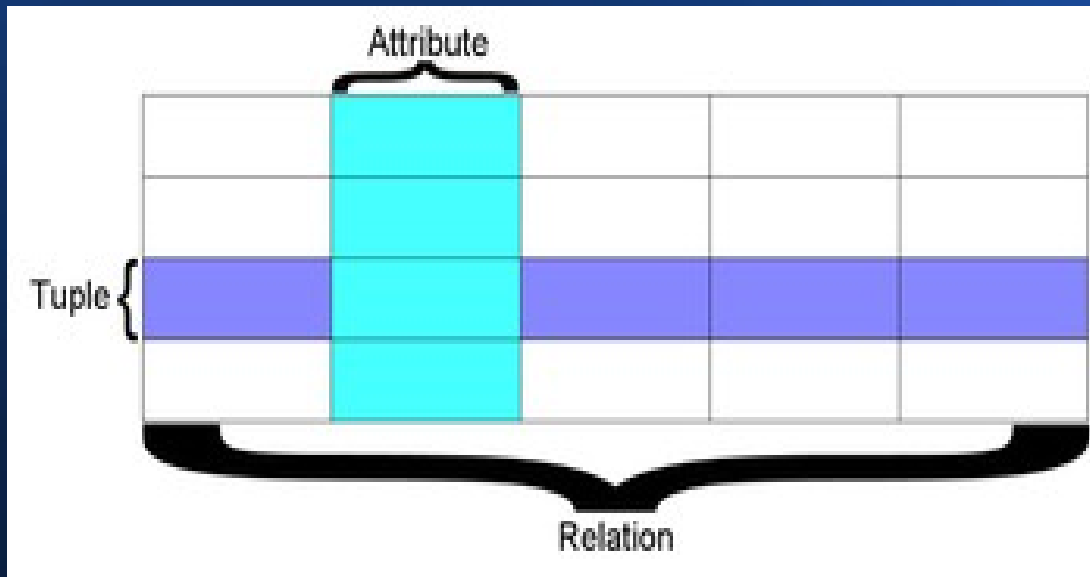
Grasp Acquisition Database



Presentation Outline

- Introduction/Refresher on SQL and Relational Databases
 - Fundamental principles
 - Basic queries and additional database features
- The goals of the GAD
 - Learning from Columbia's Grasp Database
- Proposed GAD Architecture and High-Level Schema
- Future Directions

Relational Databases



- At its core, a relational database is a set of tables (relations).
- A table has columns, which represent a data type (attribute), and rows, which represent the data (tuples).

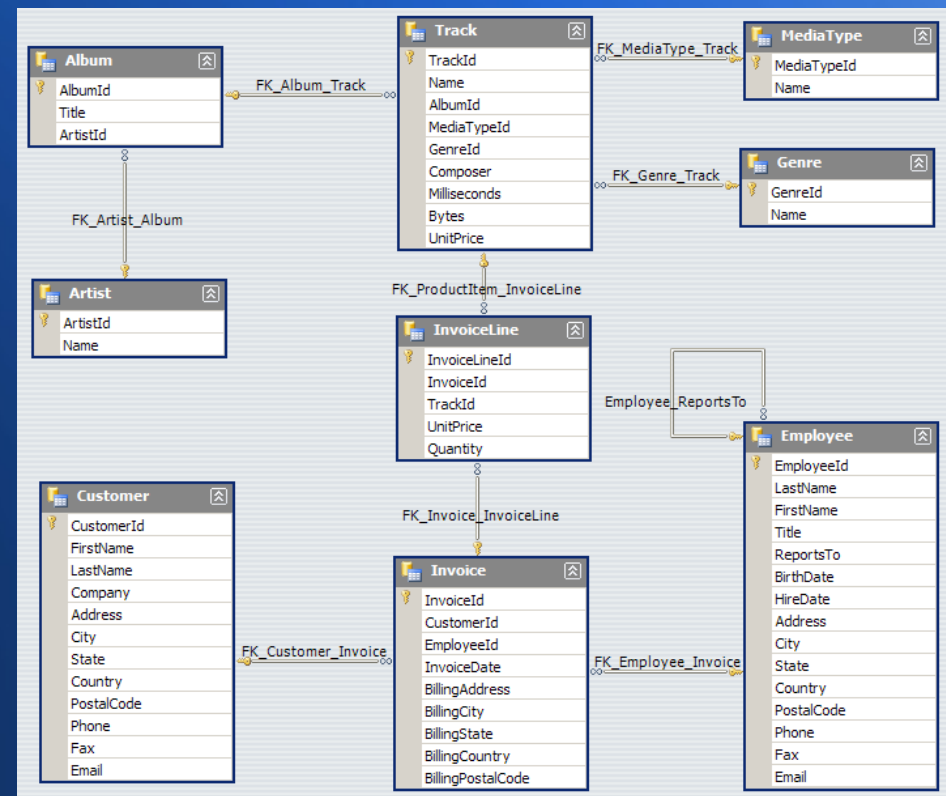
An Example of a DB Table

	Name	Color	Quantity
1	Road Frames	Black	NULL
2	Road Frames	Red	NULL
3	Helmets	Red	288
4	Helmets	Black	324
5	Socks	White	180
6	Socks	White	216
7	Helmets	Blue	216
8	Caps	Multi	288
9	Jerseys	Multi	144
10	Jerseys	Multi	180
11	Jerseys	Multi	216
12	Jerseys	Multi	252

- The example at the left features 3 attributes: Name, Color and Quantity.
- Every table must have a Primary Key, and every record must have a unique PK.

The Advantage of Relational Databases

- Relational databases allow relationships between tables to show logical relations.
- A “Contact” table may have a “Contact_Id” attribute, that allows a connection to a “Grasp” table



Standard Query Language (SQL)

- SQL is the standard programming language for relational databases
- Is very different from imperative languages like Java, C++, Python
 - SQL is a declarative language
 - This means that SQL queries allows you to specify *what* you would like to do, but not *how* to do it.

Manipulating a Database with SQL

- Typically, you write and test queries in a Database Management System (DBMS)
- Most programming languages provide libraries for connecting to a database and making queries to allow CRUD operations on the database (Create, Read, Update and Delete)

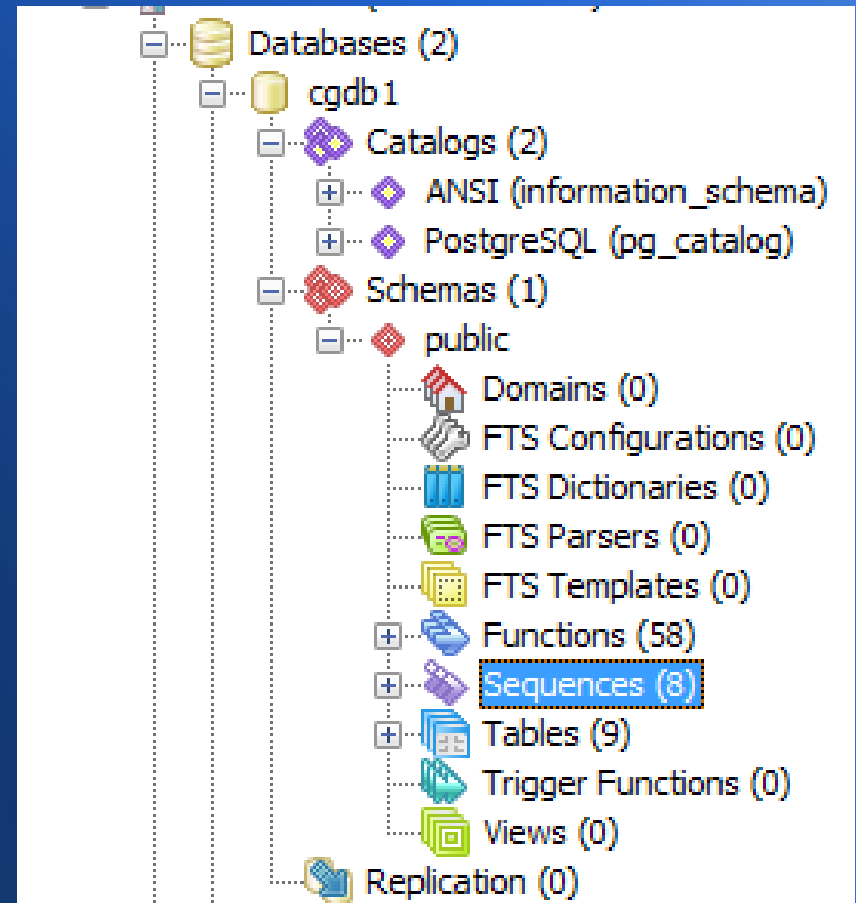
An Introductory Example

```
SELECT *  
  FROM Book  
 WHERE price > 100.00  
 ORDER BY title;
```

- The example to the left represents a simple SQL query.
- The user wishes to view all records in the “Book” table over a certain price
- The output will be ordered alphabetically

Auxiliary Database Features

- Most enterprise-level databases consist of more than just tables and SPROCS.
- Foreign Keys, Views, Schemas, Functions and Triggers are a few of the additional features



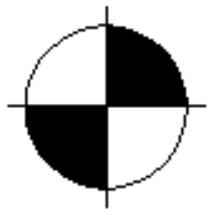
The GAD

- The GAD's main purpose is to store different kinds of grasp data in a logical and useful way.
- In designing the database, we will also want to consider:
 - Data Integrity (Keeping data complete)
 - Normalization (Ensuring data is not redundant and confusing)
 - Extensibility (Keeping our design open to improvements)

Learning from Others: Columbia Grasp Database

- Columbia University's Robotics Group has implemented an open source Grasp Database that is similar in nature to our proposed database
- Their database is a PostgreSQL DB and contains many (~240,000) grasping experiments

Columbia University
Robotics Group



Their Architecture (cont.)

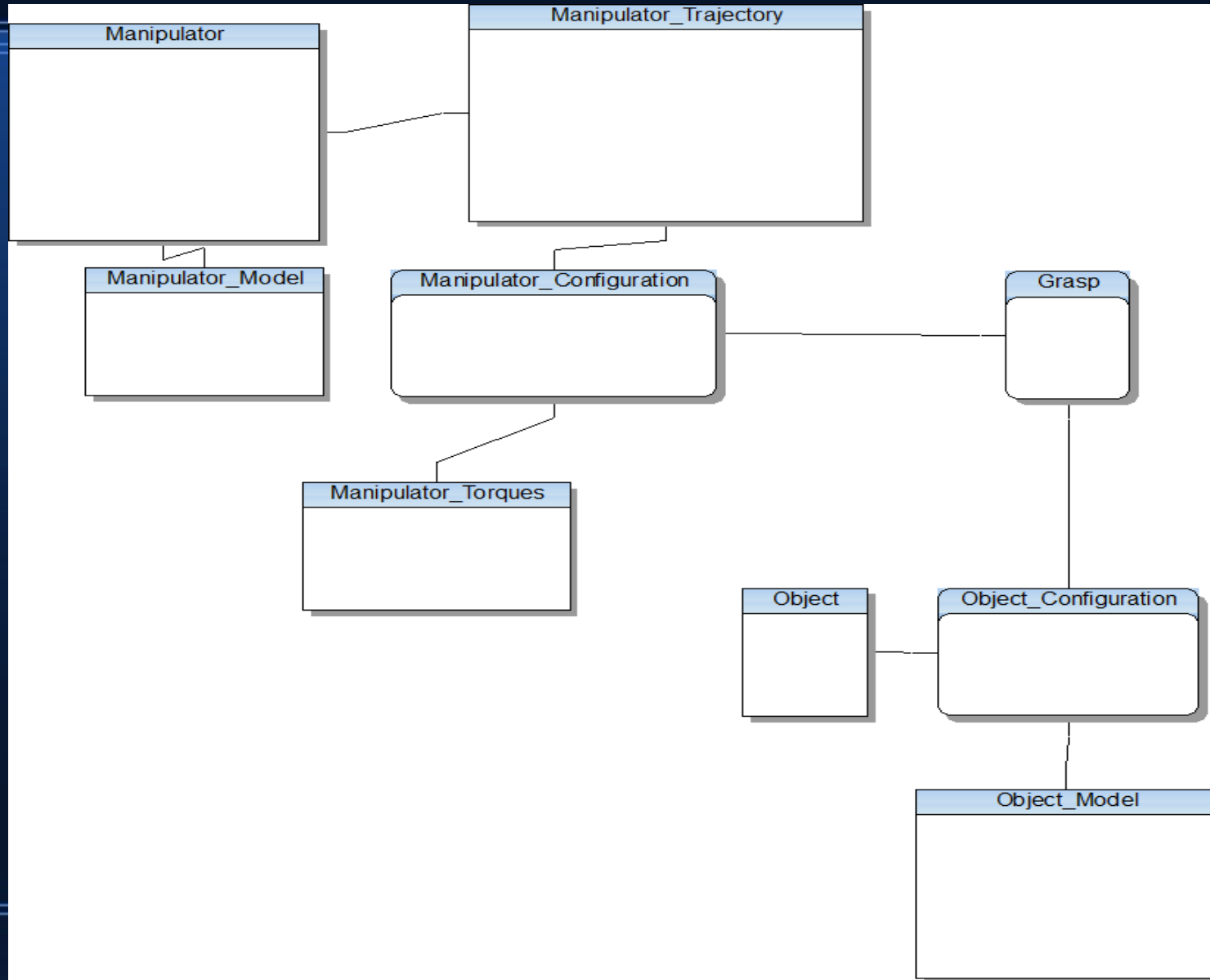
- After downloading their database, I noticed a few good design choices.
- Columbia used a few different hands in their grasp experiments, so they created a “Hand” table to represent the different manipulator end effectors.
- They used a “Grasp” table to represent each grasp.

Proposed DB Server for GAD



- PostgreSQL is considered one of the most reliable and most competitive free and open source database server.
- A DBMS is included with the server.

Proposed DB Schema



Database ERD Explanation

- The database is divided into three main regions
 - Tables corresponding to the manipulator and its properties
 - Tables corresponding to a particular grasp and its properties
 - Tables corresponding to a grasped object and its properties

Database ERD Explanation (cont.)

- For each entry in the “Grasp” table, we can associate properties of the Manipulator and Object that is being grasped in their respective tables.
- This data can be time dependent.
- We should be able to reconstruct every grasp with the data in these 3 table families.
- Video data may present additional difficulties.

Future Enhancements

- A goal of the GAD should be flexibility to accommodate new grasping scenarios
- New tables should be introduced as we implement post-processing and machine learning routines



Thanks

- Thanks to Columbia Robotics Group and you all for your time!