Thomas Sugimoto
8/2/18
CS 340 Intro to Databases

# Pokémon Database

## URL to HTML Pages

http://access.engr.oregonstate.edu:4266/ (you will need to use a VPN with the OSU login to access it)

The database is not complete but for review purposes you can look up Pokemon number 11, Metapod. It will have the complete amount of information for moves, evolutions, and locations.

http://access.engr.oregonstate.edu:4266/pokemon/Metapod

## Peer Review Feedback

Your website is very impressive and well thought-out. I am not familiar with Pokemon but I was able to easily understand what you were trying to accomplish.
CREATE functionalities:
All of your INSERTING functionalities seem to be working great and it worked flawlessly in PHPMyAdmin. The only issue that may occur is not having INSERTs for locations and moves on the web page. I am aware that you have an UPDATE option for locations but I am not sure if that is enough. Other than that everything looks and worked great.
READ functionalities:
All of the entities are shown in their corresponding rows and the relationships appear to be listed correctly. Additionally, the information is displayed in a very clear and user friendly way. The only suggestion I have is the search functionality. I may have been using it incorrectly, but it did not work.
UPDATE functionalities:
Your UPDATE interface works very well. However, when I attempt to refresh the page it seems to not have implemented my changes.
DELETE functionalities:
The DELETE functionality is not working at all. Fixing this issue may prove beneficial for the entities and the relationships they interact with.

- Jillian Emard

Data Manipulation Queries: - "every table should be used in at least one select query" - It looks like Types doesn't have an accompanying select query. - Only Pokemon has an Insert Into query, but the specs say that every table should have the ability to add new entries individually. - There's no update queries (required to have at least one). - Appropriate number of delete queries. - The file looks syntactically correct. HTML Pages: - The website is very detailed and thought-out - good job! Data is clearly displayed and well-organized in tables. - Add form for Pokemon looks good. - Are there going to be Update and other Add forms? Good luck with the project!

- Becky Chao

Awesome work! I'm excited to see where this project goes as we near the end of the term. It is clear from the diagrams and sql files that a lot of thought went into each concept. You're making a functional pokedex, and for that should be commended. I unfortunately don't have many recommendations to improve your project. As it stands it works great and I am confident that the completed work will be just as impressive.

- Liam Beckman

First off sorry, my original submission was intended for the other person I was reviewing but here is yours: Overall your draft seems very well done and complete. The relationships explanation at the end if

very helpful and makes sense. All the data types seem to be reasonable and the entities make sense as well. The Pokemon to Pokemon relationship seems like it could be a one to one relationship since each pokemon only evolves into one other pokemon.

- Timothy Ip

## Actions and Updates Based on Feedback

It was a hard decision but I chose to change some of the design of the Pokemon table and make the types of the pokemon a 1 to many relationship instead of many to many. Since there is only a max of 2 types for each pokemon it can be divided into type1 and type2. The rest of the documents have been updated.

As given by the feedback, I have added many new queries that were missing.

I am adamant on my decision and I believe the reviewer's correction to be incorrect. I clarified that some unique Pokémon can evolve into many different ones requiring the Pokémon to Pokémon relationship to be one-to-many.

Some of the names for the Entities, and Attributes were alter because they were reserved words for the MySQL syntax.

Added a move_type since it is a simple relationship between moves and their types since a move must have only 1 type.

# Project Outline

My database project will be based on the Pokémon series. The Pokémon series is a animated show and a video game serious with a setting of a world with so called "Pocket Monsters" where players and characters catch, train and battle their Pokémon. It is a good choice because it is an expanded universe and has many relations between Pokémon, moves, types and locations. Pokémon are unique species of monsters that each have up to two different types, as well can have up to 4 unique moves. Each Pokémon also can come from a specific region from the original red and blue Pokémon games. I will be only doing the original 151 Pokémon in the region called "Kanto" but the information about the Pokémon and moves will be updated to the current generation.

# Database Outline

Here are the entities of the Pokémon database and each's unique attributes.

- **Pokemon** – This is the main list of all the Pokémon in the database. To simplify, I will not use the accented e (é) as it is originally. It has the following attributes:
    - **id** – The pokedex number is the id number that is assigned to help the database easily identify the Pokémon. It is an increasing number which is the primary key. The pokedex from the game has already given each Pokémon a number so it is not automatic. The value cannot be null.

    - **name** – The name of the character with a string of 100 max characters. This cannot be null.
    - **type1** – The first type of the pokemon. This must not be null and must point to a type
    - **type2** – The second type of the pokemon. This can be null and the default is 0..
    - **description** – The given description of the Pokémon from the original games.
- **Move** – The move of the Pokémon is the action that a character can do during a Pokémon battle.
    - **move_id** – This is an assigned auto-incrementing number used as a key for all the moves. This cannot be null.
    - **move_name** – This is the name of the move with a string of 100 max characters and it cannot be blank. This cannot be null.
    - **move_category** – This is a classification of the move. There are only 3 options: Physical, Special, and Status. Represented as a string and the default is Physical.
    - **move_type** – This is the type of the move that points the type table. Each move can have only one type.
    - **move_power** – The power of the given move. It is an int and must be greater than or equal to 0.
    - **move_pp** – PP stands for power points, it's the number of times a Pokémon can use the specific move. This will be represented as an int and must also be greater than 0.
- **Types** – There are 18 different elemental Pokémon types ranging from water, and fire to ghost and dragon. Each type is more or less effective against each other and some types have unique interactions.
    - **type_id** – The auto-incrementing id for each type, represented as an int. This cannot be null.
    - **type_name** – The name of the type as a string.
- **Location** – The various locations of the Pokémon game world. To make it simpler this will only be the locations from the original Pokémon games red and blue.
    - **location_id** – The locations auto-increasing int id number.
    - **location_name** – the location's name as a string.
    - **location_description** – The description of the location given by the game.

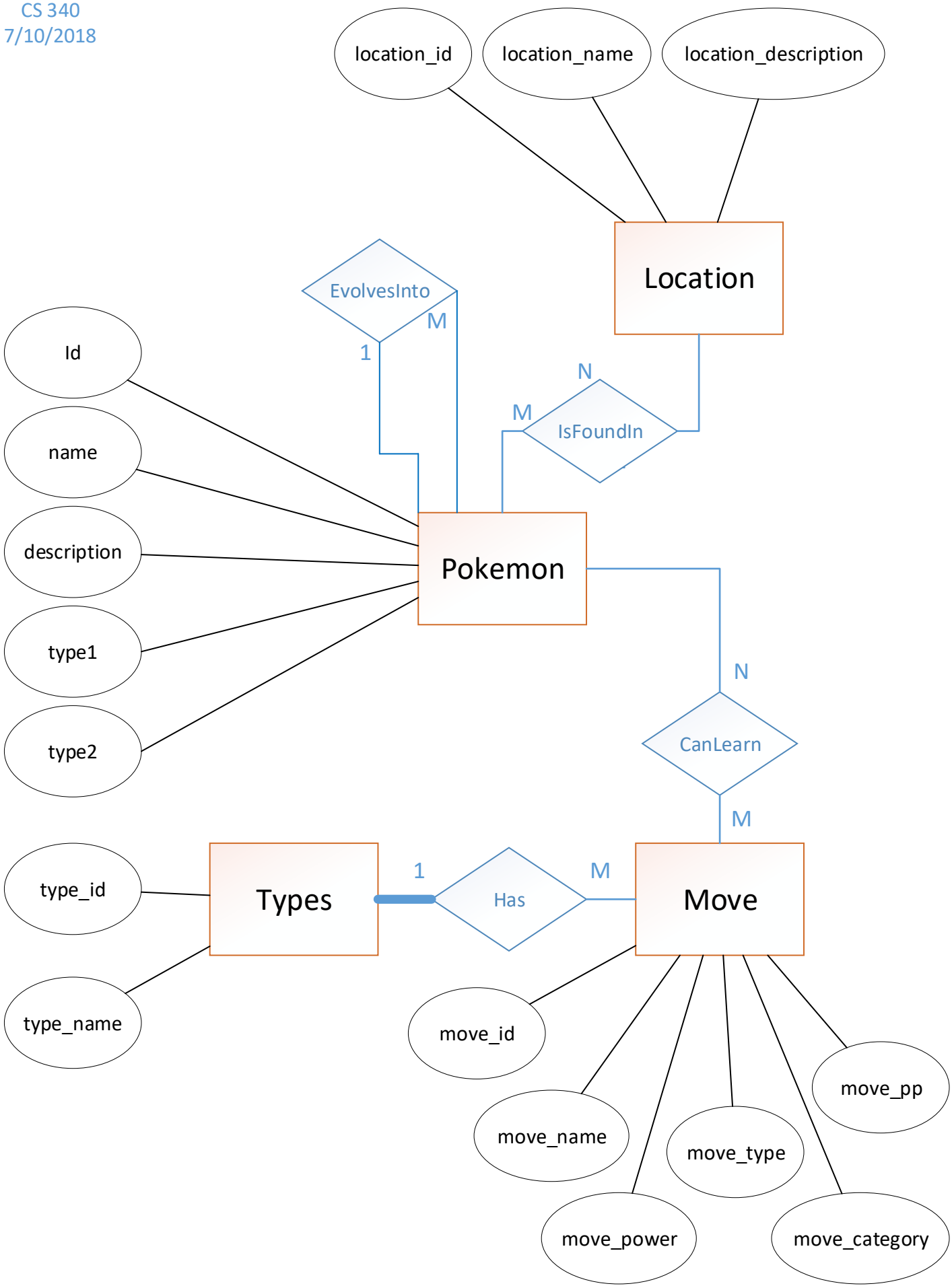
## Relationships Outline

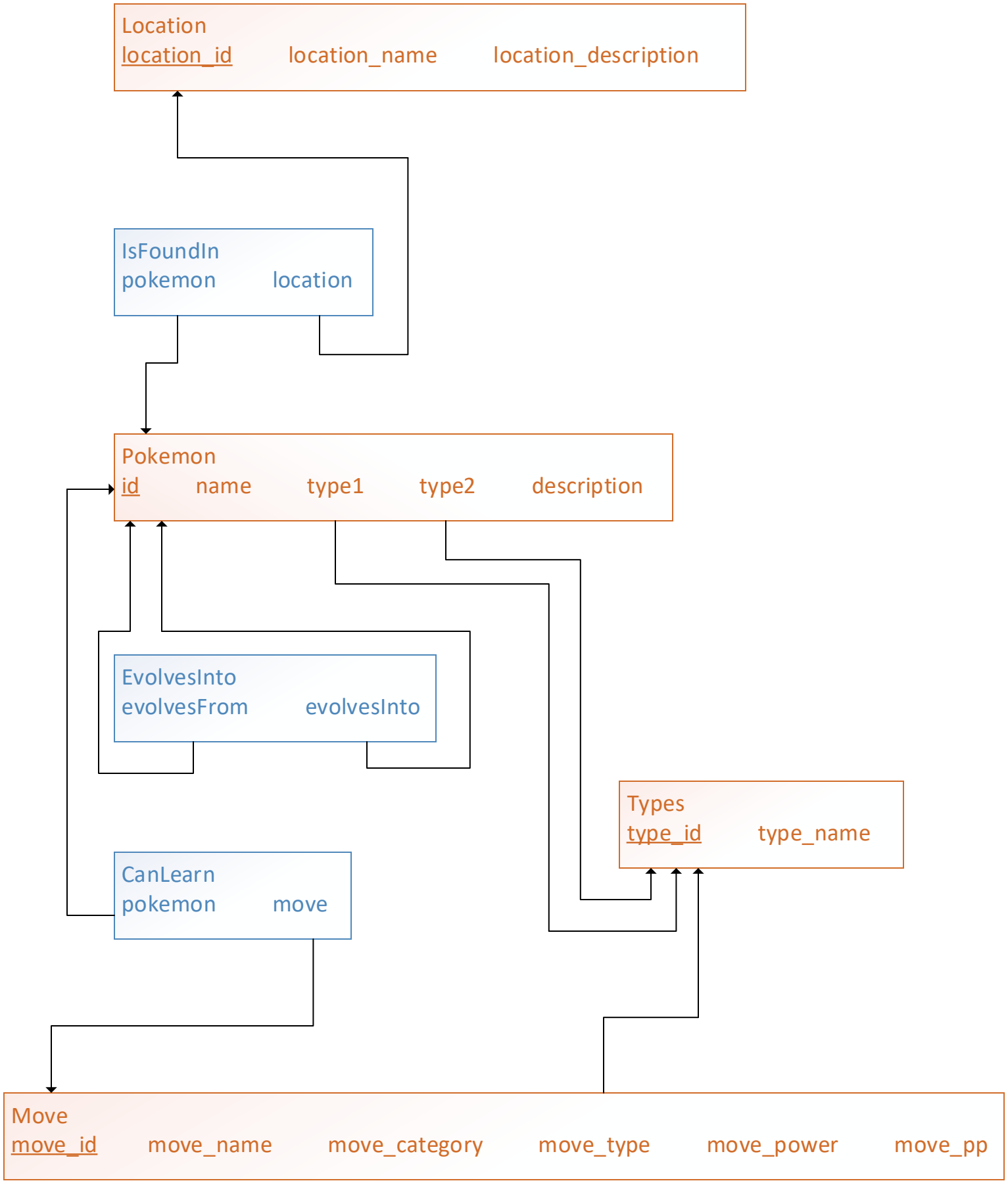
- **Has** - Moves and types (many-to-one) – This is the relationship between moves and types, each move must only have one type, but each type and have many moves with that type.

Thomas Sugimoto
8/2/18
CS 340 Intro to Databases

- **IsType** - Pokémon and types (one-to-many) – This relationship states that each Pokémon has at least one type with a maximum of two types (type1 and type2) and the types cannot be the same. Each type can be assigned to many different Pokémon.
- **EvolvesInto** - Pokémon and Pokémon (one-to-many) – Pokémon can evolve into different Pokémon and this shows the predecessors and successor of each Pokémon. Also there are some Pokémon that can evolve into many other Pokémon such as Eevee so that is why it's a one-to-many relationship.
- **CanLearn** - Pokémon and Moves (many-to-many) – Pokémon can learn up to 4 different moves at the same time but this is a list of potential moves that the Pokémon is able to learn (which are many). Moves can be learned by many Pokémon.
- **IsFoundIn** - Pokémon and Locations (many-to-many) – Pokémon can come from many locations and there are many locations have Pokémon. Pokémon can be found at multiple locations and some locations can have no Pokémon. Also some Pokémon might not have a specific location.

Thomas Sugimoto
CS 340
7/10/2018




Location
location_id location_name location_description
IsFoundIn
pokemon location
Pokemon
id name type1 type2 description
EvolvesInto
evolvesFrom evolvesInto
Types
type_id type_name
CanLearn
pokemon move
Move
move_id move_name move_category move_type move_power move_pp