

# CSE 344: Section 7

# SQL++, Cost Estimation

February 15th, 2018



# SQL++ Crash Course

# What is SQL++?

Just like SQL but parsed for processing JSON data

SQL++ has keywords to handle collections of data (i.e. non-flat data)

# Motivation for SQL++?

Why SQL++? Why not some other query language?

People are used to/like specifying data through SQL syntax (like C++/Java)

SQL-like language enforces idea of physical data independence

# Useful Keywords/Syntax

`is_array( ... )` ----> checks if value is an array

`split(s, d)` ----> splits string `s` on delimiter `d`

`[ ... ]` ----> explicitly construct array

`(CASE WHEN ... THEN ... ELSE ... END)` ----> combine with `“is_array(…)”`

`MISSING` ----> reserved keyword like `“NULL”`

`` ... `` ----> backtick needed for accessing keys with names containing `“-”`

# Cost Estimation

# Looking Inside the Black Box

Query optimization is a complicated task

What good is it for us to know these “implementation details?”

Indexes! Design your DB instance to speed up critical queries.

For each relation we have  $> (\text{number of attributes})!$  possible indexes.

Real-world tables can have 50+ attributes = quindecillion indexes