SQL Design

# Parser

## Grammar

### Highlight key grammar components

### Binding variables

### Names

## Building the parser

### Which unit tests

## Visualizing

## Parsing

## Errors

# AST

## Building (visitor pattern)

## NamedExpressions

## Column

## Utility Methods

### Find\_node, find\_nodes, etc.

### Children (for recursion)

## Evaluation

## Object Model

### AST

### Logical

### Boolean

### Numeric, etc.

## Walk relations

## Symbols overview (shadow AST)

# Symbol Attachment

## Shadow AST (purpose is to attach metadata)

## (what validation is done here?)

## Relations

## TableColumns (pulls in metadata)

## How name resolution is done

## What if wrong NameCompare?

# NameCompare

# SqlReader

## Execute

## Execute\_df

## Connection properties

### Host

### Password

### Port

### User

### Driver-dependent properties

## Returning cursors

# Validator

## Overview

## List of Rules

### Inner aggregates

### Presence or absence of keys

### Counts for korolova, etc.

### Supported aggregates

## Does this need NameCompare?

## What about rules that involve propagation of sensitivity? Others not obvious from AST?

# Rewriter

## Valid options (clamp\_dims)

## NameScope

### Anonymous Names

## Pushing expressions like VAR

### Should we swap DP\_SUM for diff priv SUM, etc.?

## Pushing expressions that use the core library

## Pushing expression that needs histogram or other computed columns

# PrivateReader

## Valid options (clamp\_dims)

## Row-based aggregates

## Functional (map/filter/sort)

## This duplicates some validation rules. Is there a reason why?

## Are there some validation rules here that aren’t in validator? If so, why?