One planner fits all

(How Apache Calcite makes it easier to write a DBMS)

Lightning talk at XLDB 2015
Stanford, California
Julian Hyde (Hortonworks)
“One size fits all” is an idea whose time has come and gone

–Mike Stonebraker (2005)
• Hadoop and other open source technologies have deconstructed the DBMS

• Query parser/API + catalog + authorization + algorithms + scheduler + engine + data format + storage
Conventional DB

Catalog

Parser

Algebra

Algorithms

Data
Apache Calcite
DB framework

Schema SPI
Parser
Algebra
Engine
Operators, Rules, Statistics, Cost model

Engine
Data
Engine
Data
Engine
Data
SELECT products.name, COUNT(*)
FROM sales
JOIN products USING (productId)
WHERE sales.discount IS NOT NULL
GROUP BY products.name
ORDER BY COUNT(*) DESC

translate SQL to relational algebra

FilterIntoJoinRule

scan [products]
join
filter
aggregate
sort

scan [sales]
scan [products]
filter'

scan [sales]
join'
aggregate
sort
Relational algebra

- Robust
- Allows re-use
- Complex cost-based optimization
- Multiple front-ends & back-ends
- Not just for “flat” relations
Thank you!

Download: http://calcite.incubator.apache.org

Use Calcite to build your next database!

Calcite powers Apache Hive, Drill, Phoenix, Kylin

An Apache Incubator project since May 2014

@julianhyde

What’s in the box?

• SQL parser & AST
• JDBC/ODBC framework
• Built-in operators (project, filter, …)
• In-memory engine
• 100+ rules
• Planning engines
• Adapters (CSV, JDBC, Mongo, …)
• Streaming SQL
• Materialized views