Storing Intermediate Results in Space and Time

SQL Graphs in jSQL_e

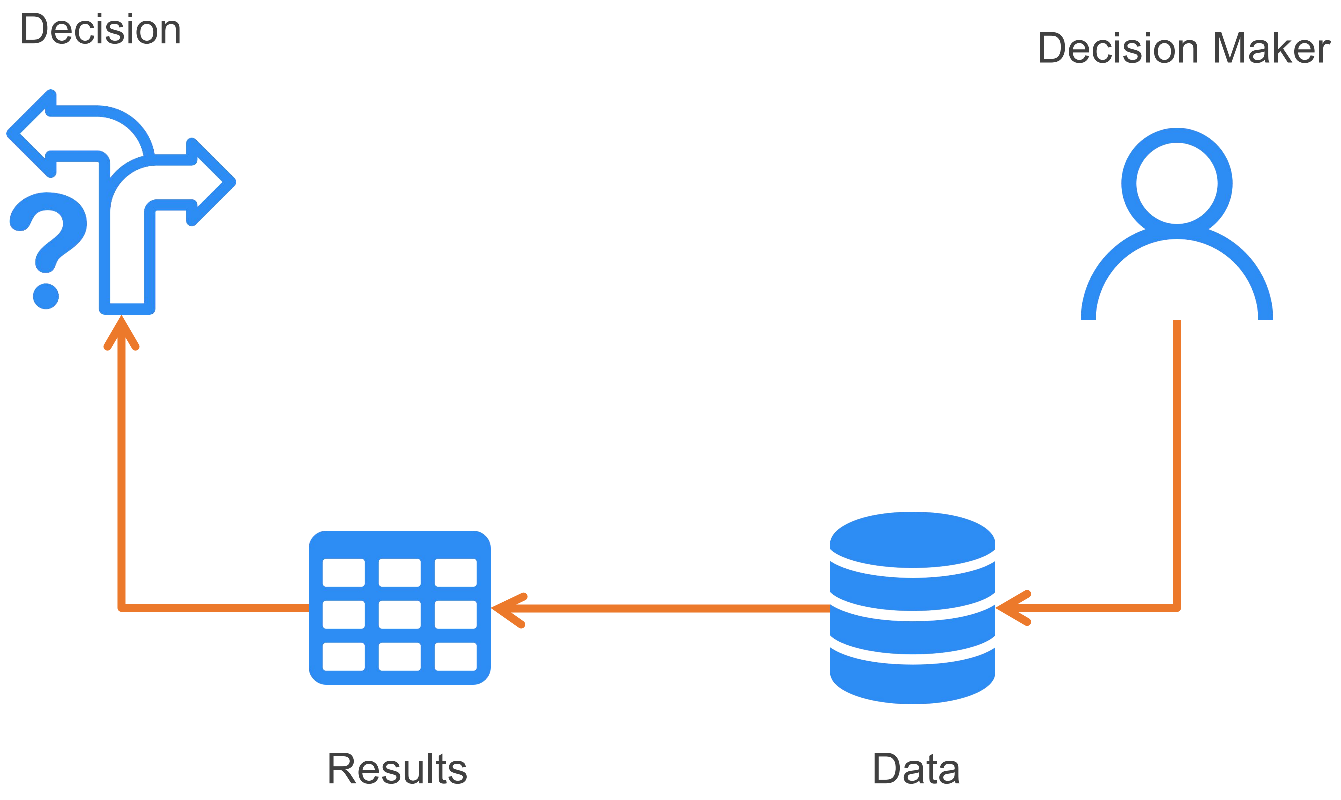
By: *Basem Elazzabi* (elazzabi@pdx.edu)

Advisors: *David Maier & Kristin Tufte*

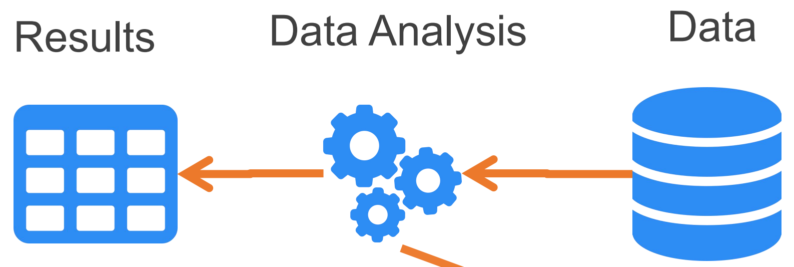
Portland State UNIVERSITY

The Big Picture

Data-Driven Decision Making

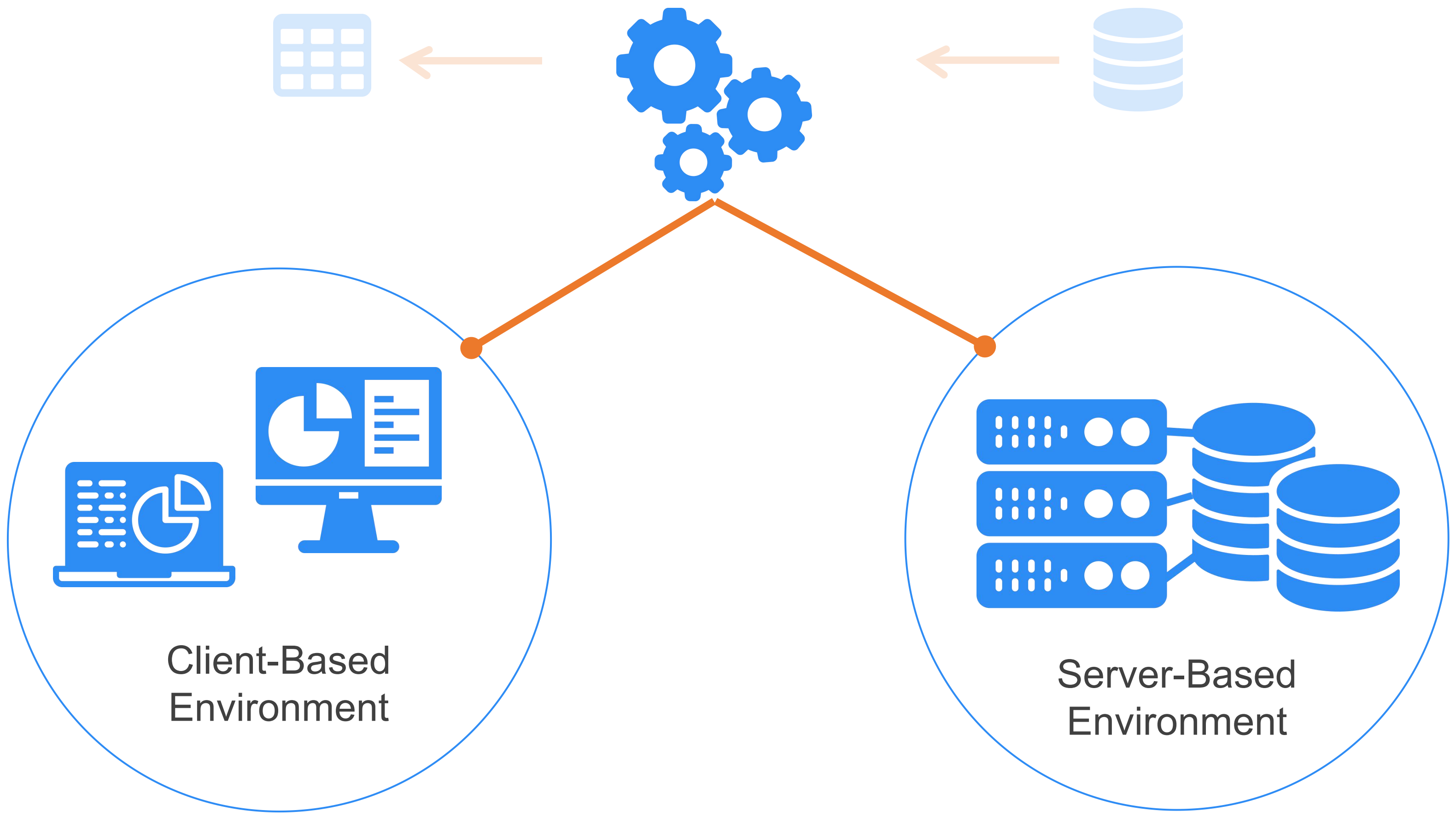


Data Analysis

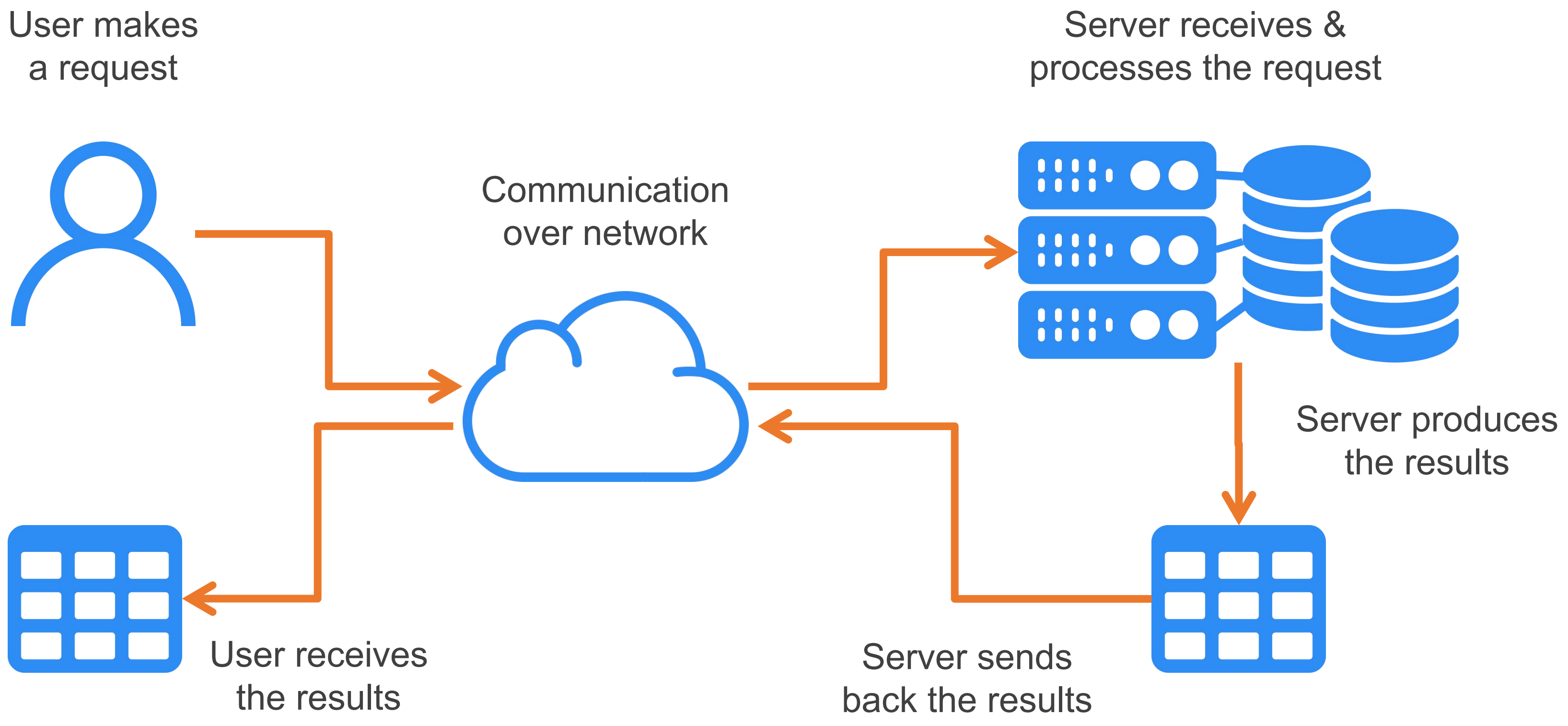


Facilitating the data-analysis process

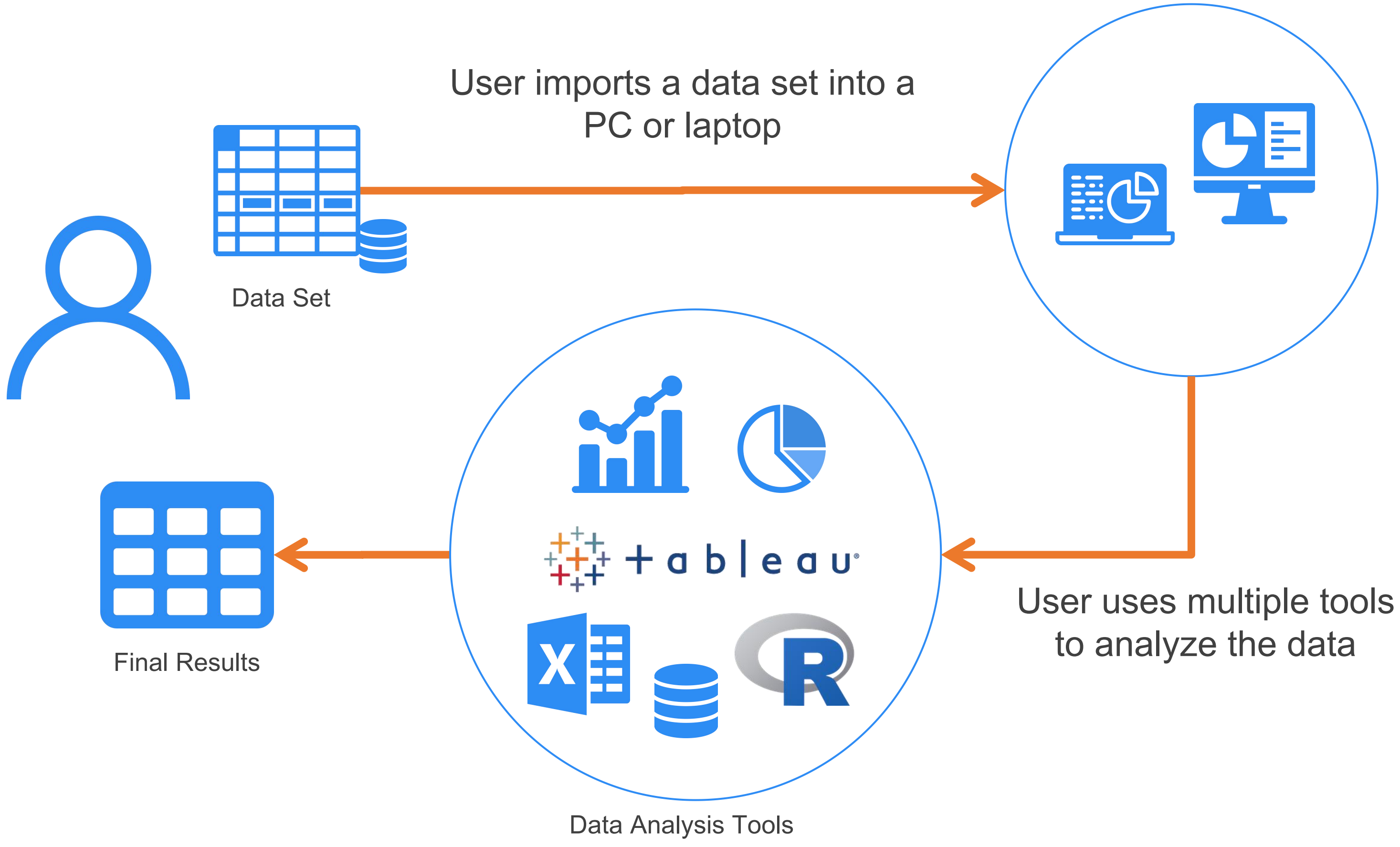
Types of data analysis



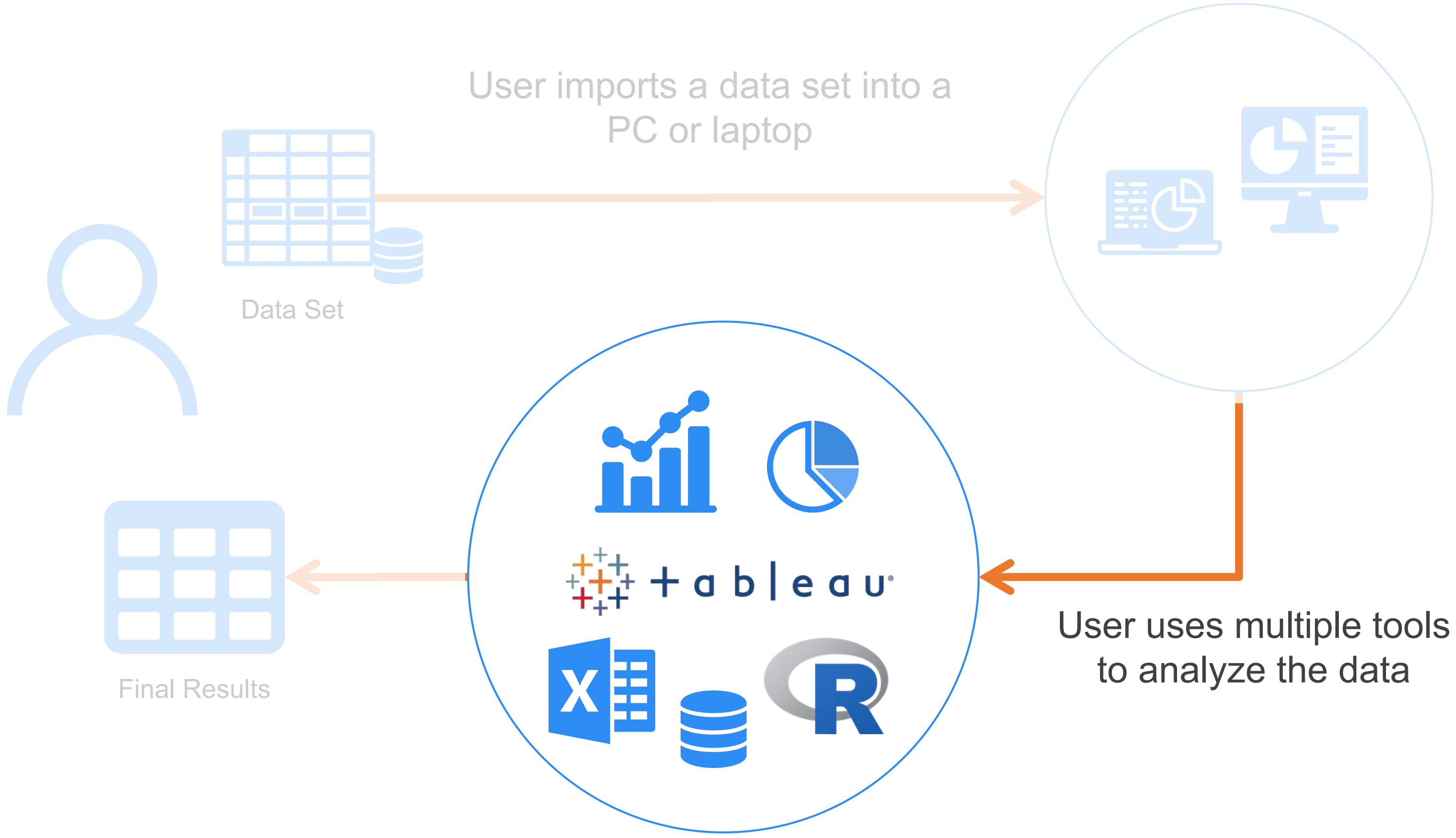
Server-Based Environment


User makes a request
Communication over network
Server receives & processes the request
Server produces the results
Server sends back the results
User receives the results

Client-Based Environment


User imports a data set into a PC or laptop
Data Set
User uses multiple tools to analyze the data
tableau
R
Final Results
Data Analysis Tools

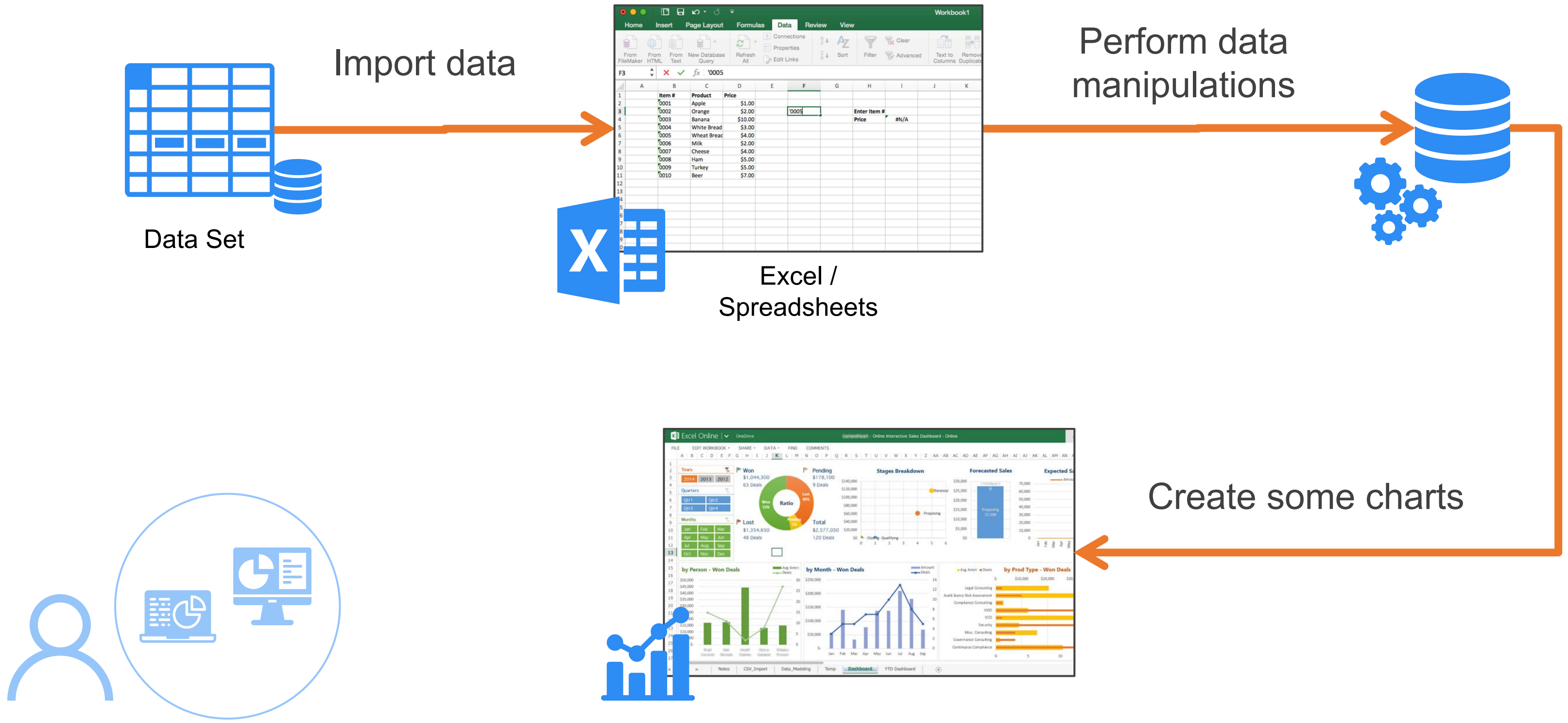
Client-Based Environment


User imports a data set into a PC or laptop
Data Set
Final Results
+ a b l e a u
User uses multiple tools to analyze the data

Data Analysis Tools

The Problem

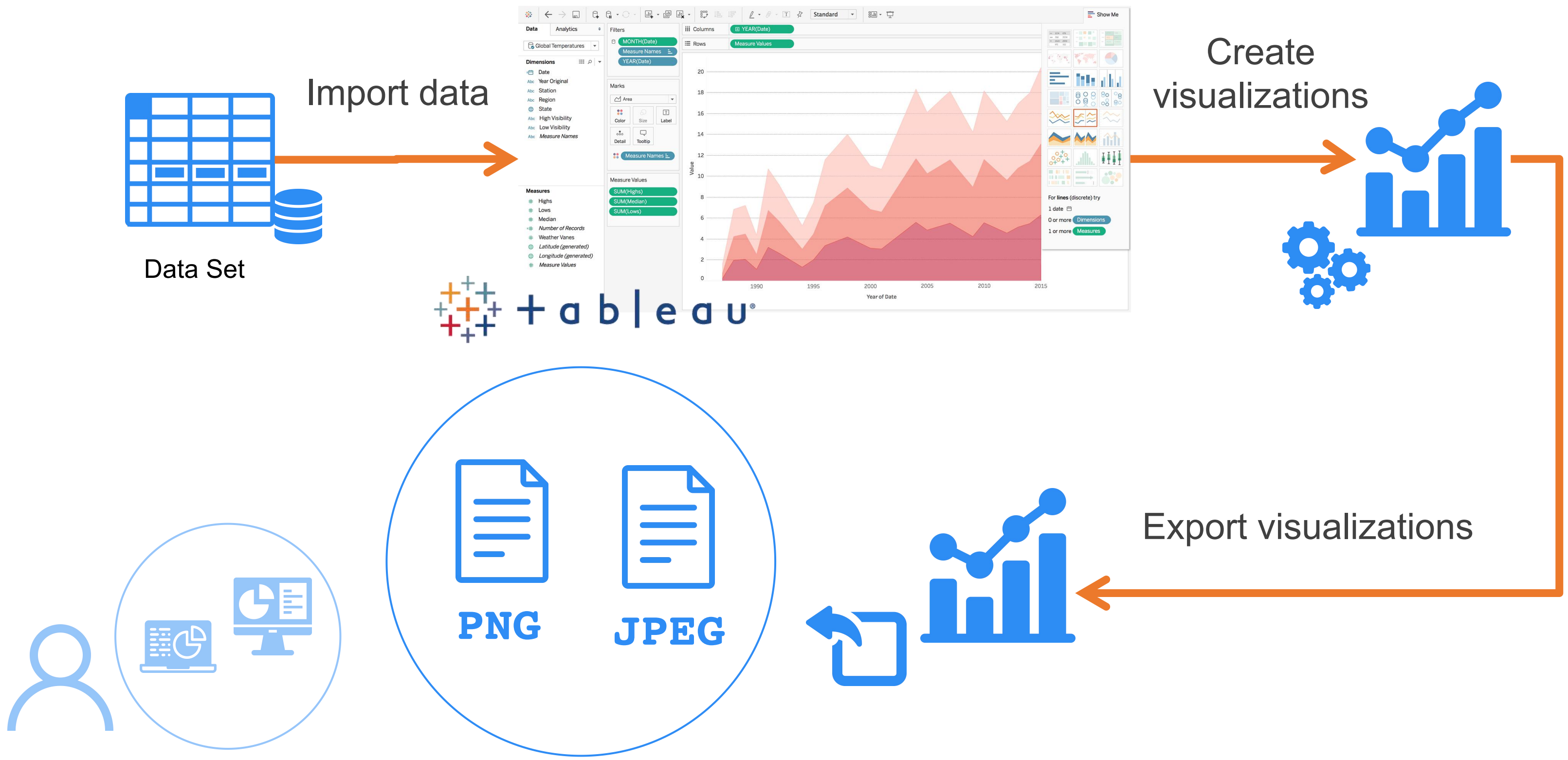
Typical Data-Analysis Process



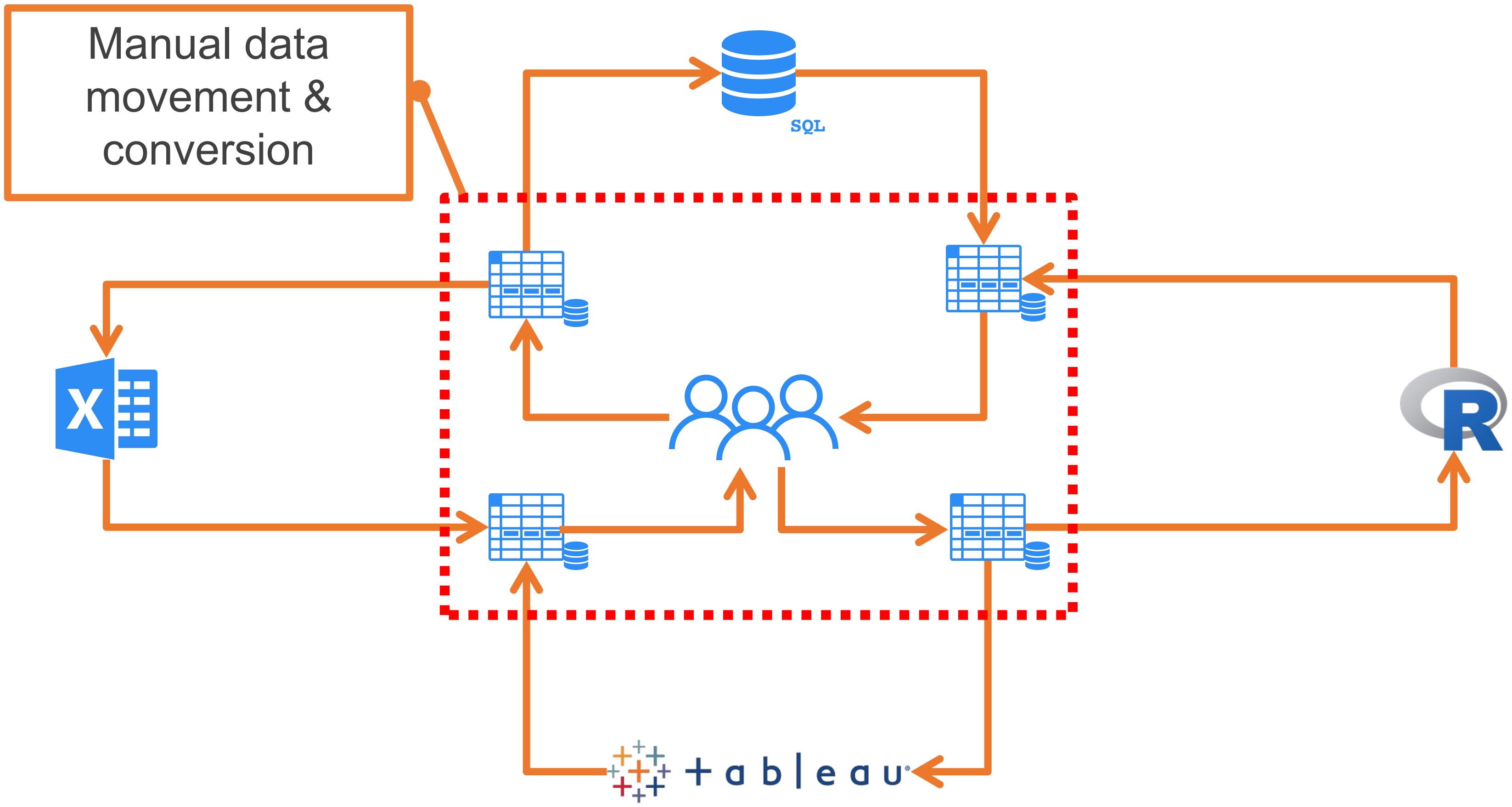
Item #	Product	Price
0001	Apple	$1.00
0002	Orange	$2.00
0003	Banana	$10.00
0004	White Bread	$3.00
0005	Wheat Bread	$4.00
0006	Milk	$2.00
0007	Cheese	$4.00
0008	Ham	$5.00
0009	Turkey	$5.00
0010	Beer	$7.00

Enter Item #	'0005
Price	#N/A

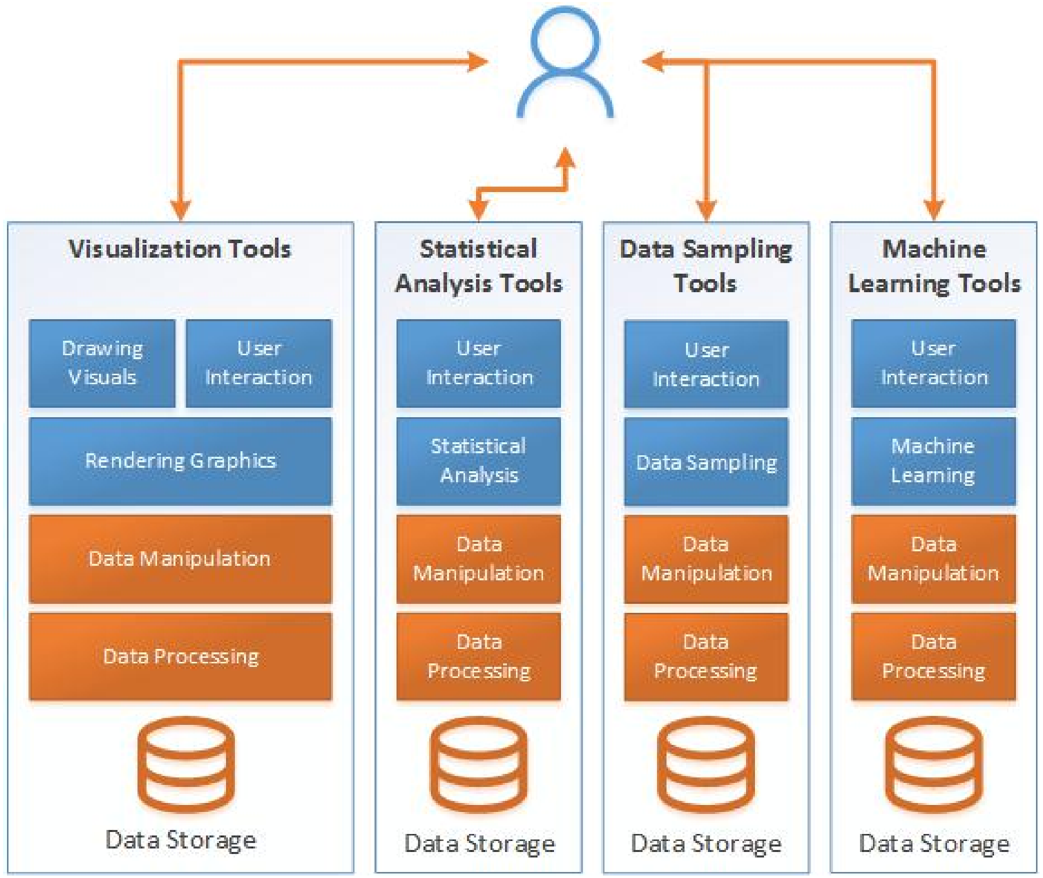
Typical Data-Analysis Process



Typical Data-Analysis Process

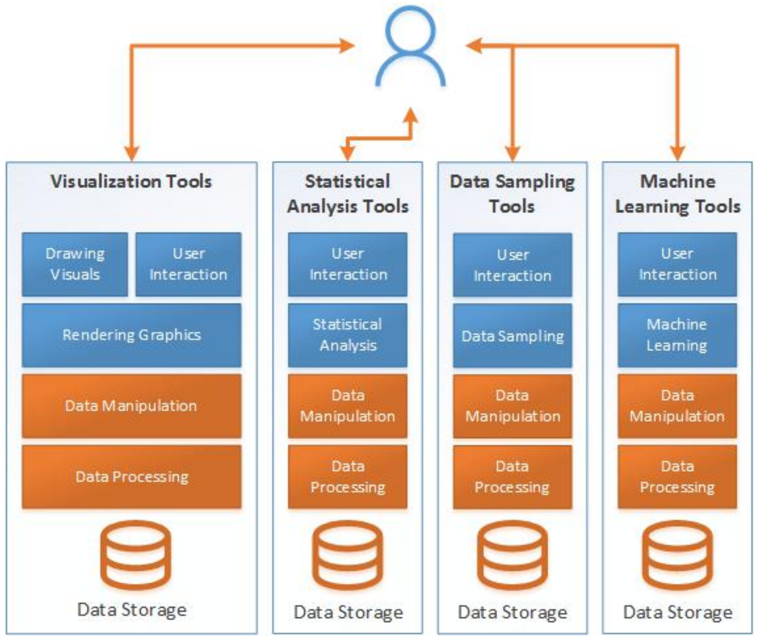


Manual Data Movement



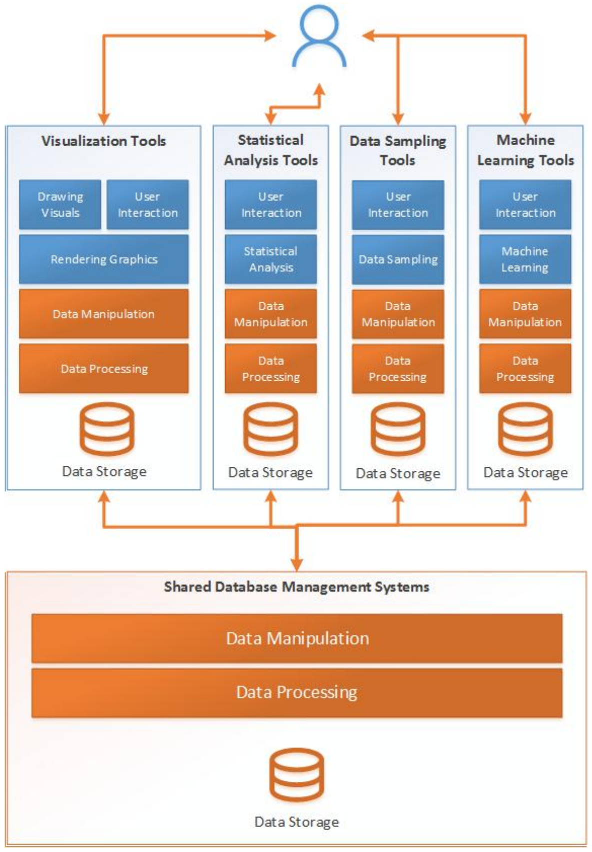
Manual Data Movement

- Wastes a lot of time
- Takes a lot of effort
- Requires technical skills
- Wastes space
- Redoing computations
- Difficult to inspect results

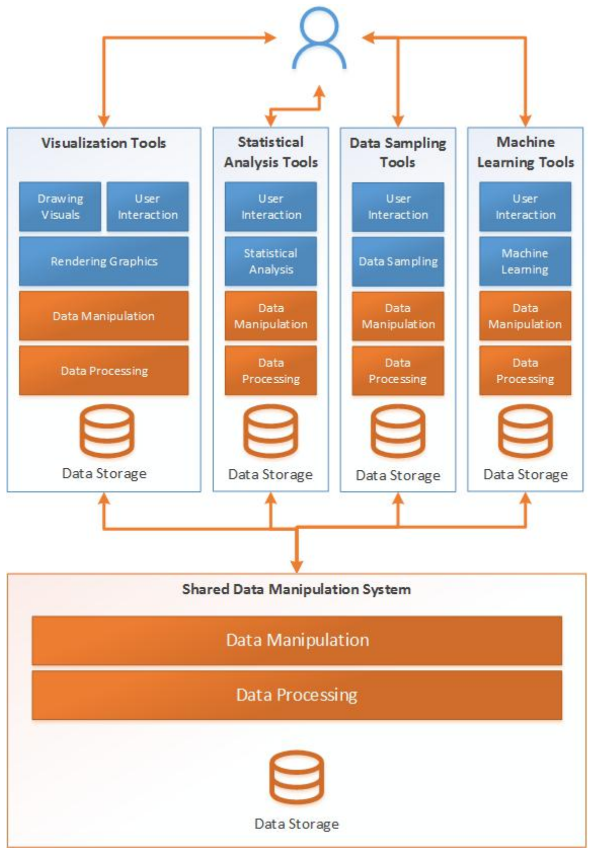


Shared Data-Manipulation Systems

- Still wastes a lot of time
- Still takes a lot of effort
- Still requires technical skills
- Still wastes space
- Still redoing computations
- Still difficult to inspect results

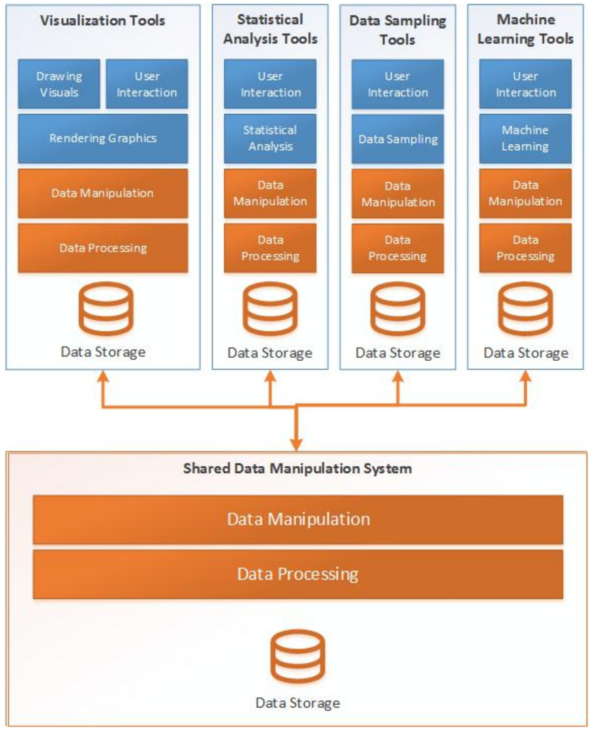


The Ideal Environment



The Ideal Environment

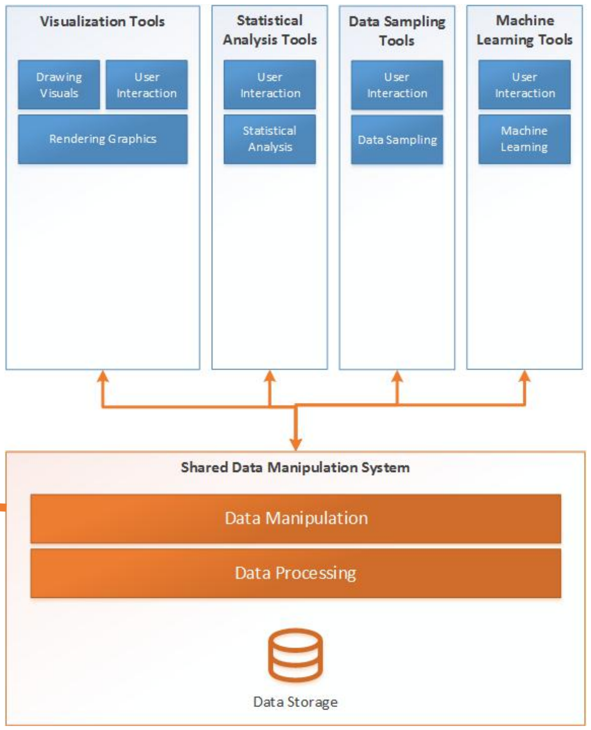
- Eliminate data movement and conversion



The Ideal Environment

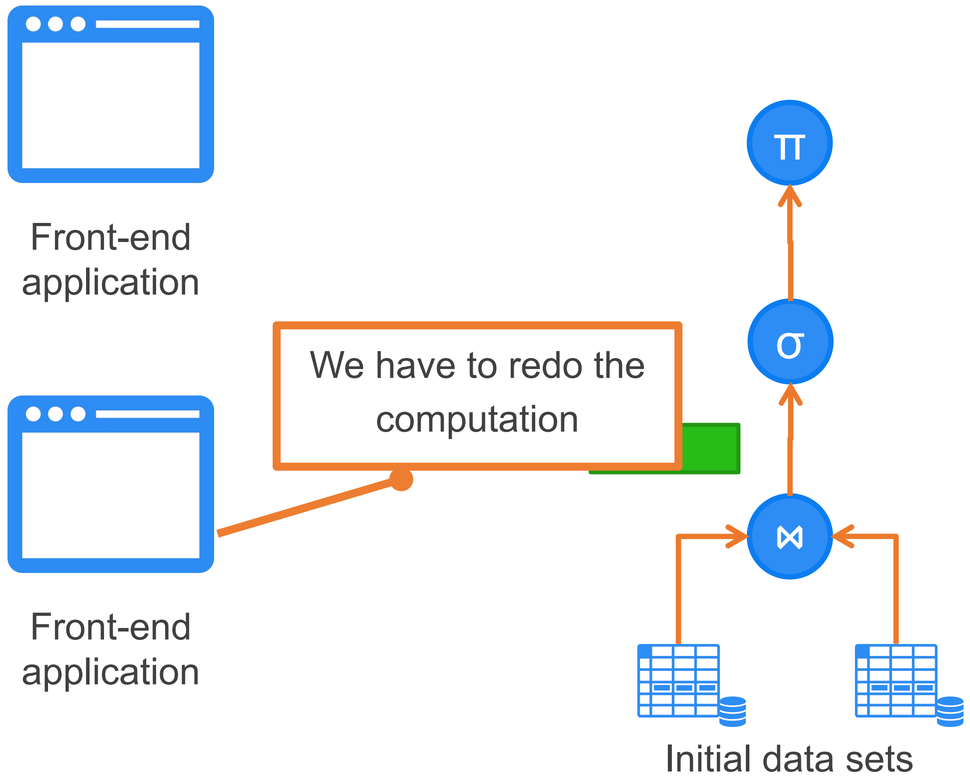
- Eliminate data movement and conversion
- Factor out the data manipulation process and storage into the shared system.

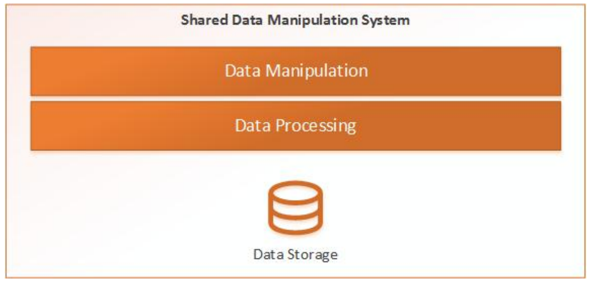
How can we build this system?



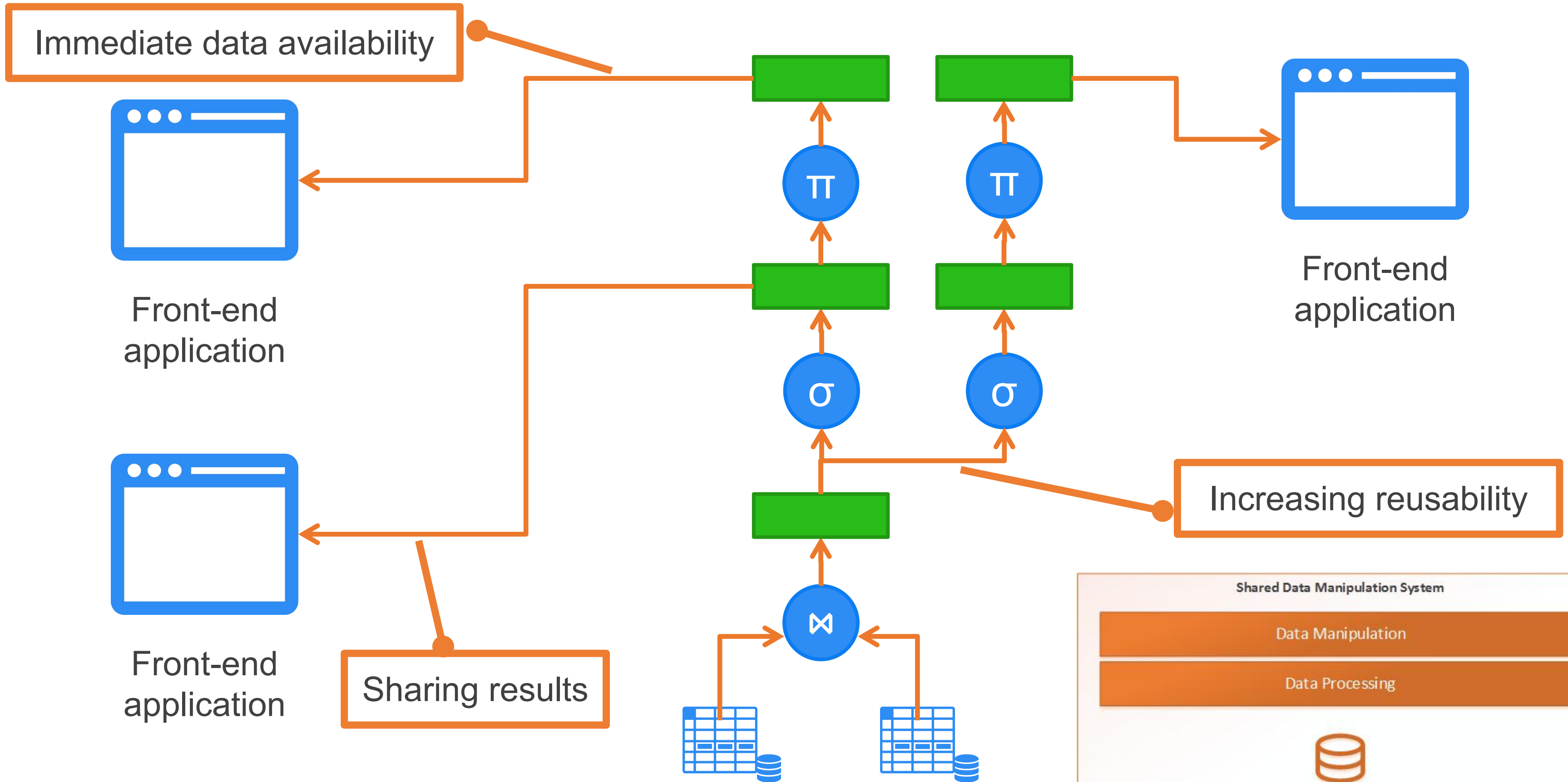
The Data Model

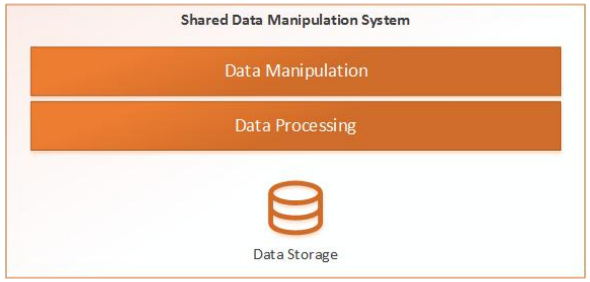
The Current Data Model



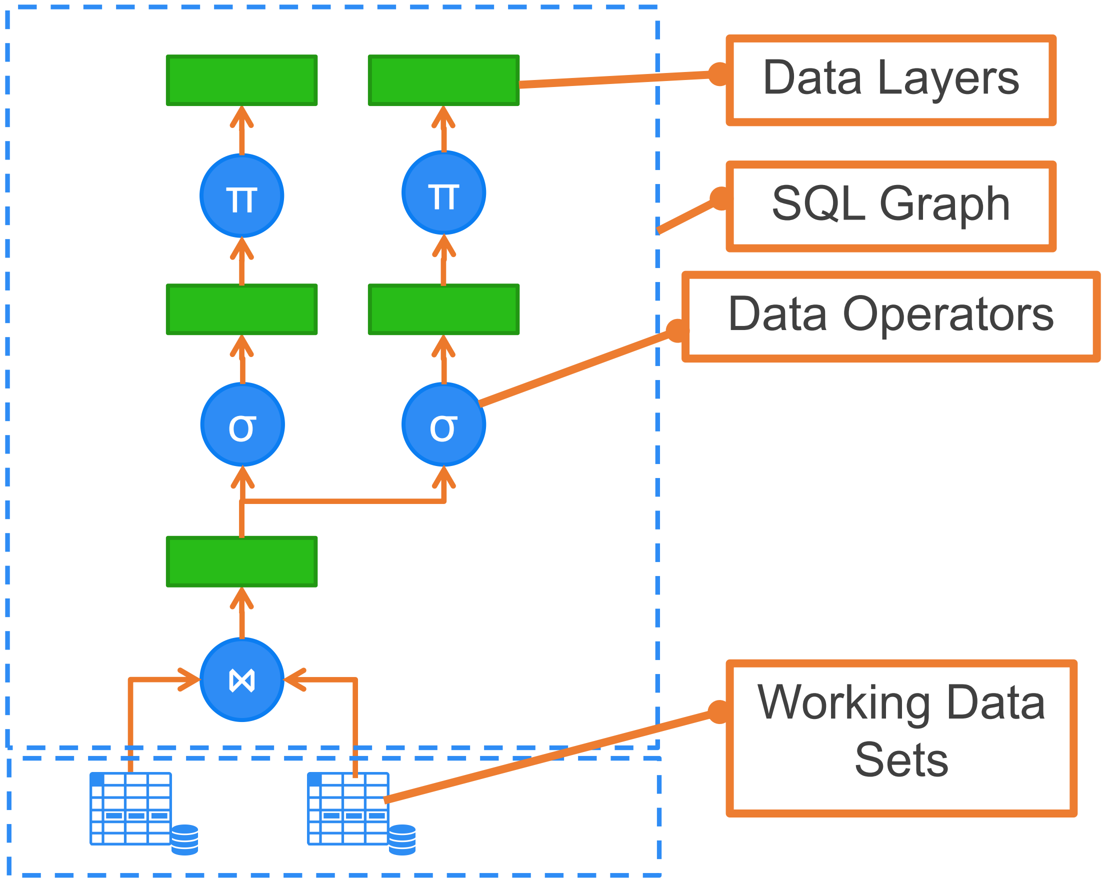


The Proposed Data Model



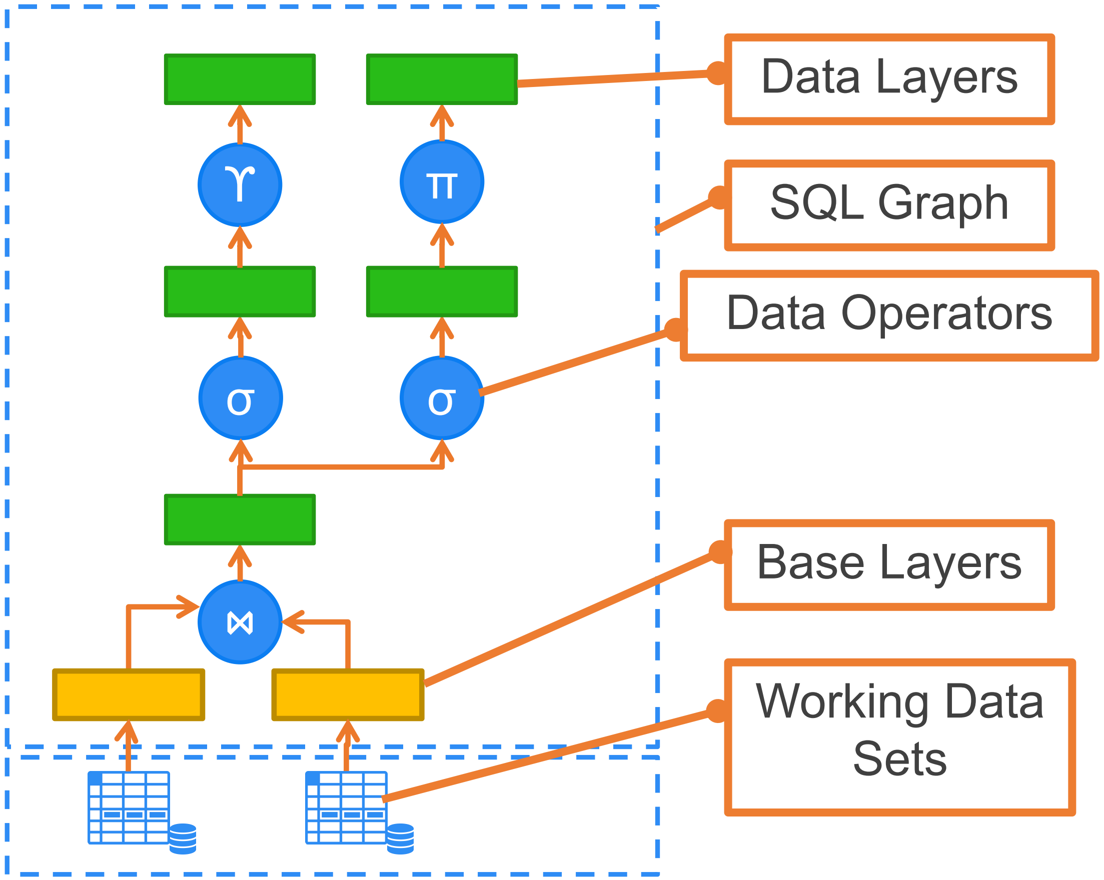


The Proposed Data Model



The Proposed Data Model

- Immediate data availability
- Share results
- Eliminate data movement
- Eliminate data conversion
- Increase reusability

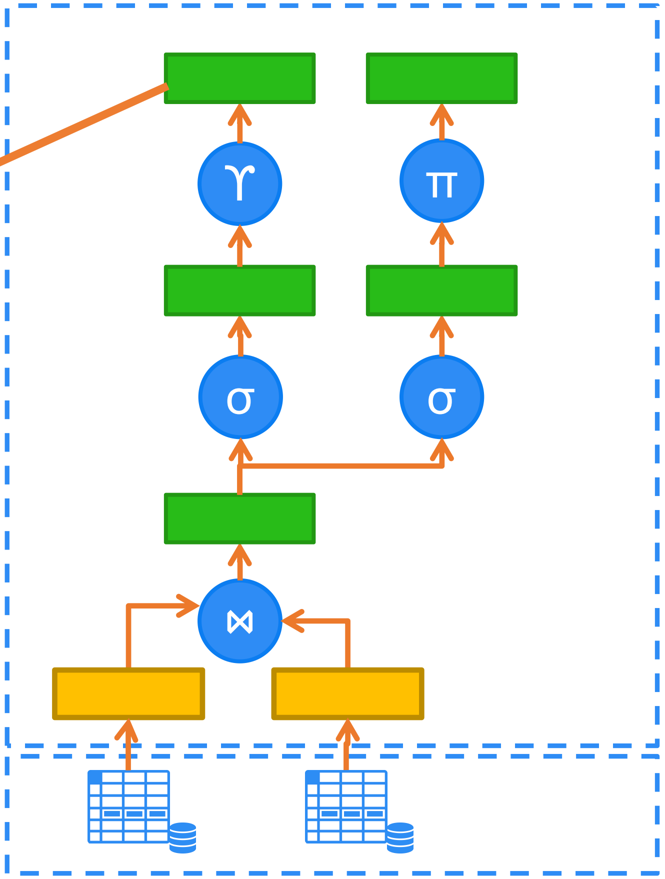


The Challenges

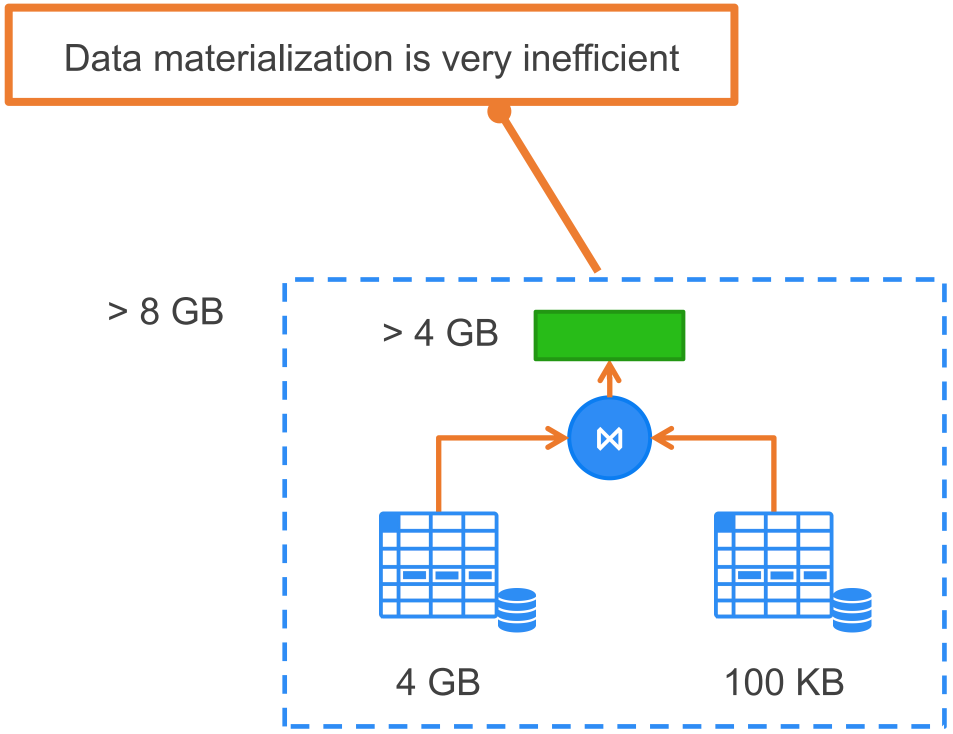
The Challenges

1. How to store intermediate results efficiently.

2. How to provide data accessibility with interactive speed.



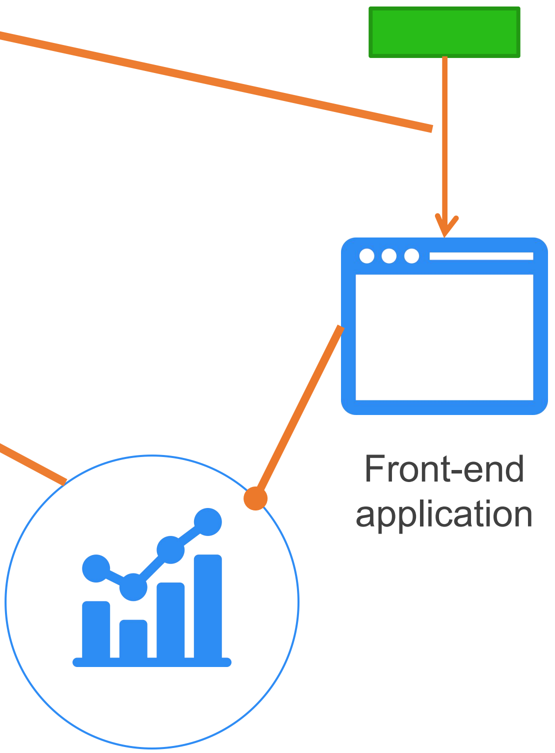
Storage Cost - Materialization



Data-Access Time

Applications have requirements on how fast they need to acquire the data.

Interactive visualization tools need to respond within 500ms



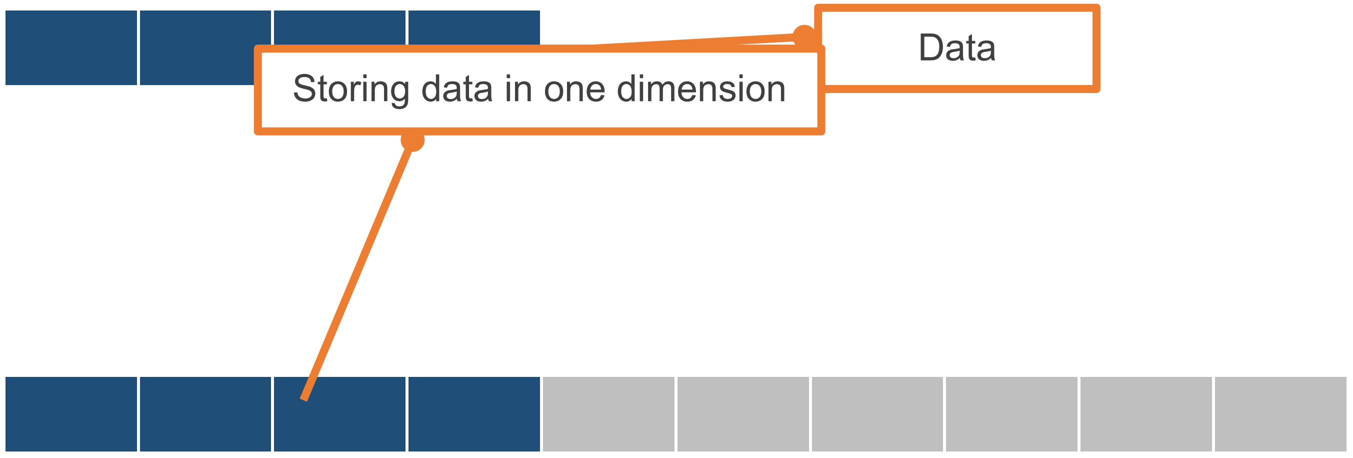
Research Goals

Store the data and all or most intermediate results in main memory

Provide data accessibility with interactive speed

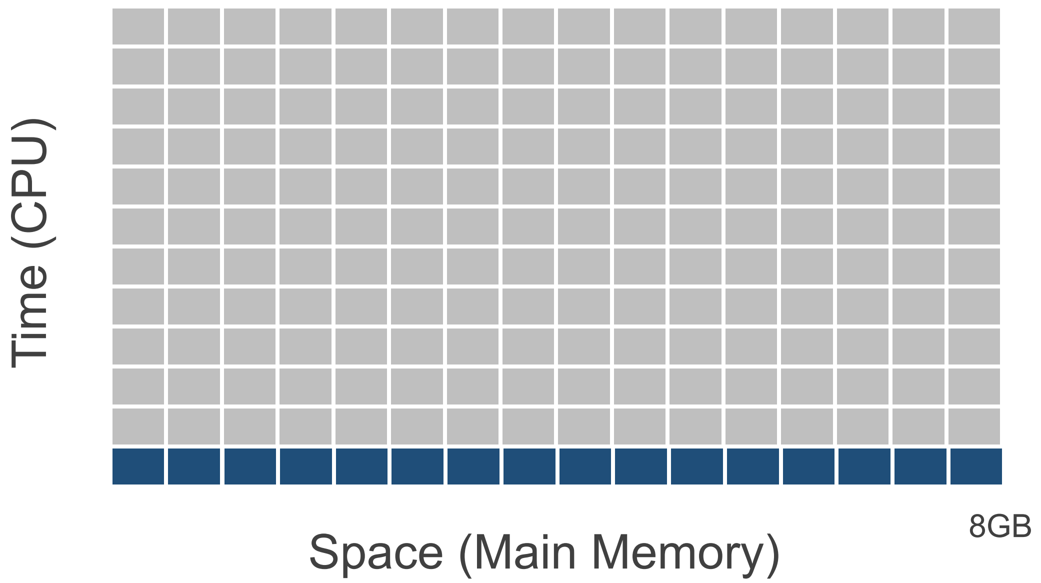
Storing Data in Space and Time

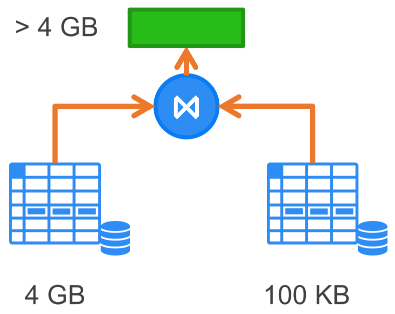
Storing Data in One Dimension



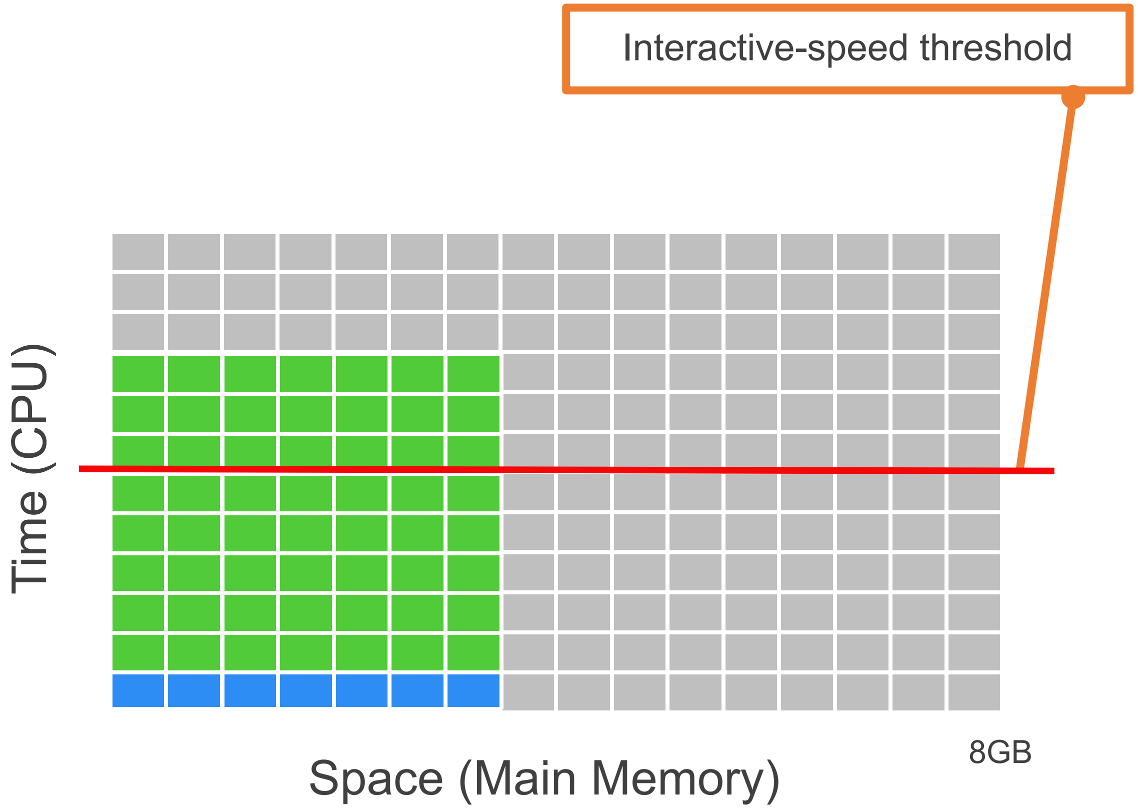
Space (Main Memory)

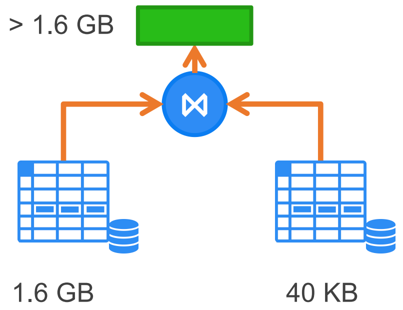
Storing Data in Two Dimensions



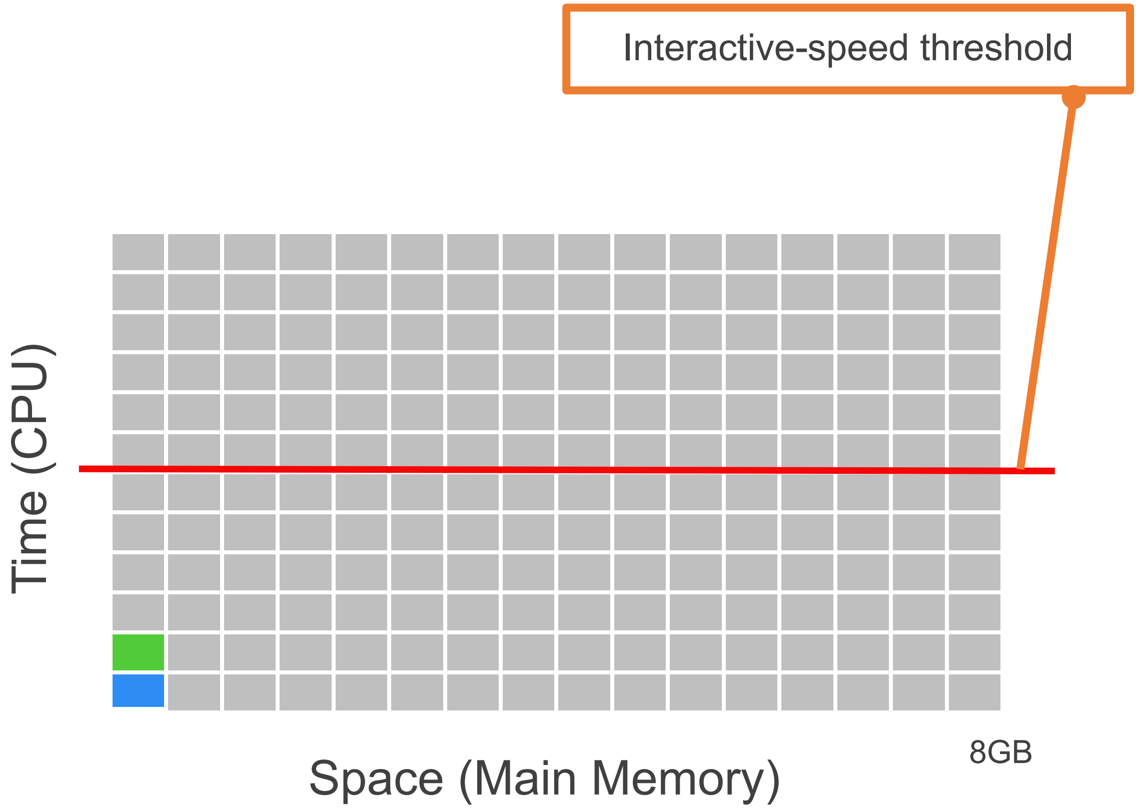


Storing Data in Two Dimensions





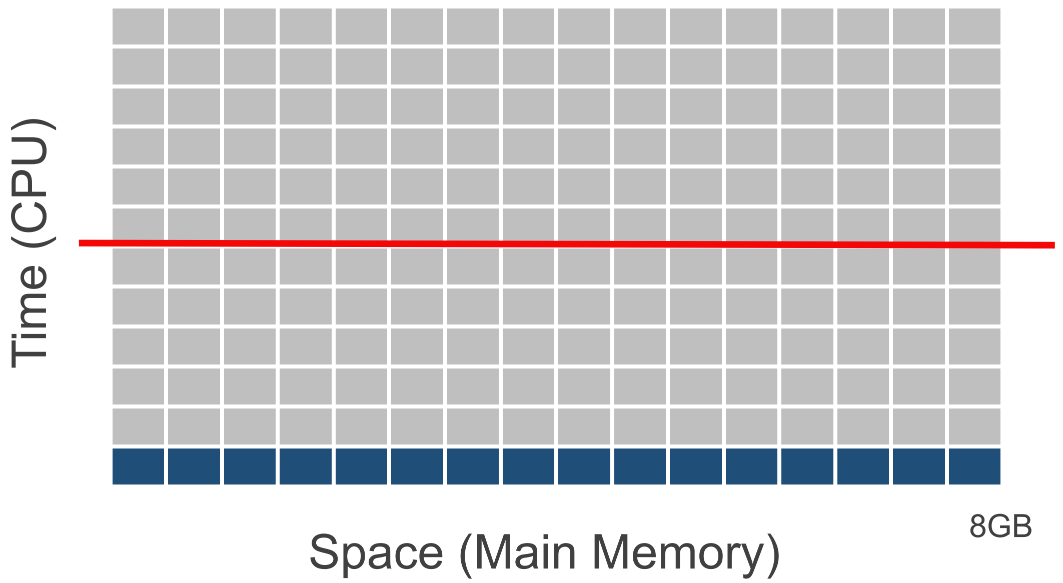
Storing Data in Two Dimensions

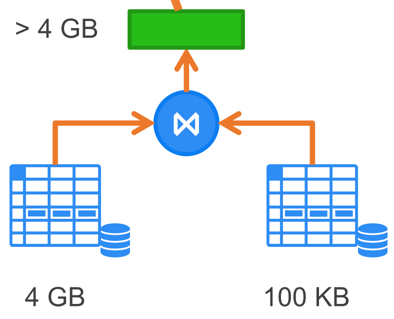


The Solution

Space Cost

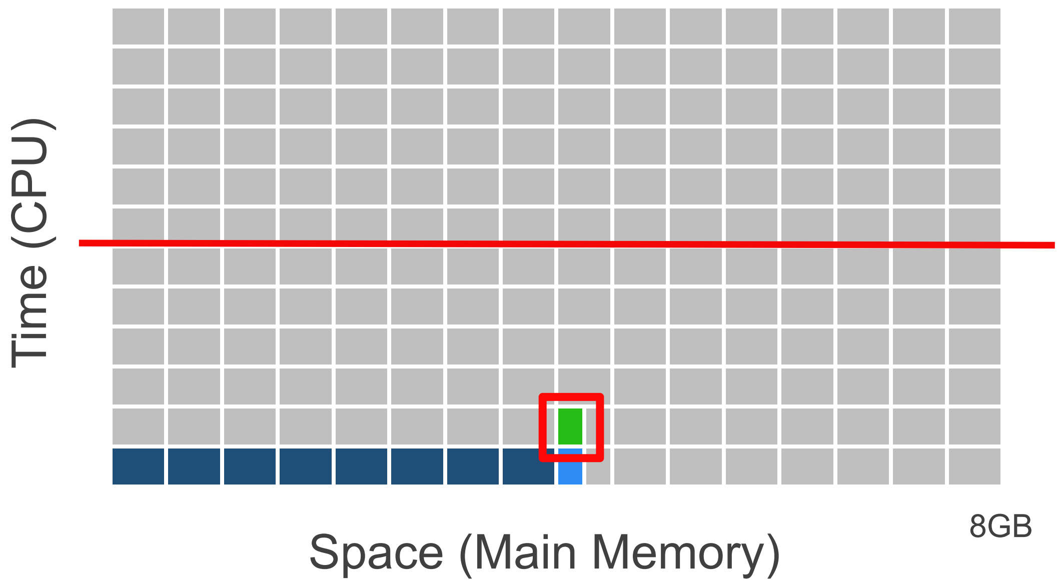
Using materialization, total space cost of intermediate results is > 4GB

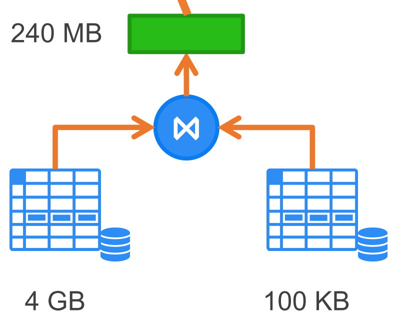




Space Cost

Using block referencing, total space cost of intermediate results is 240MB

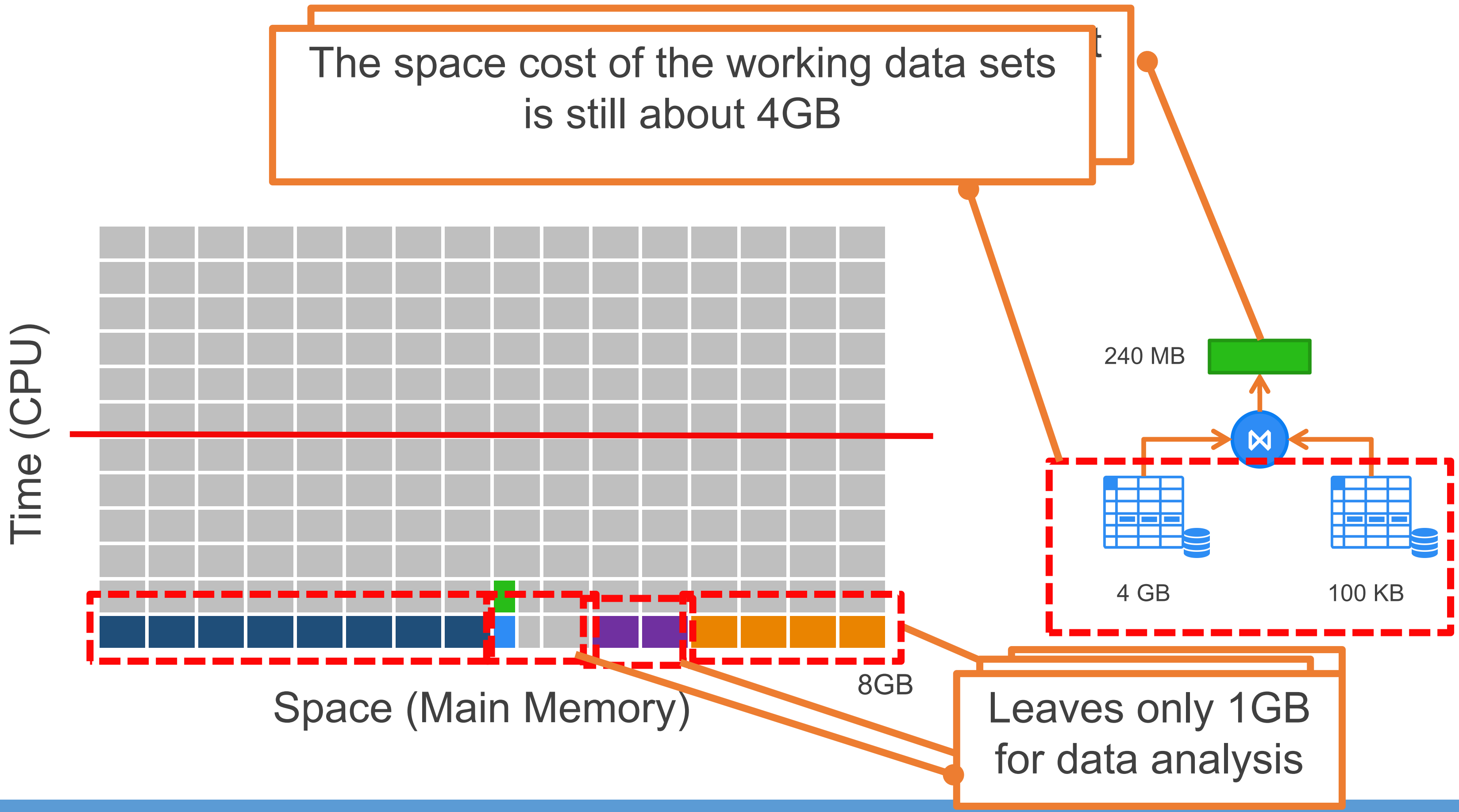




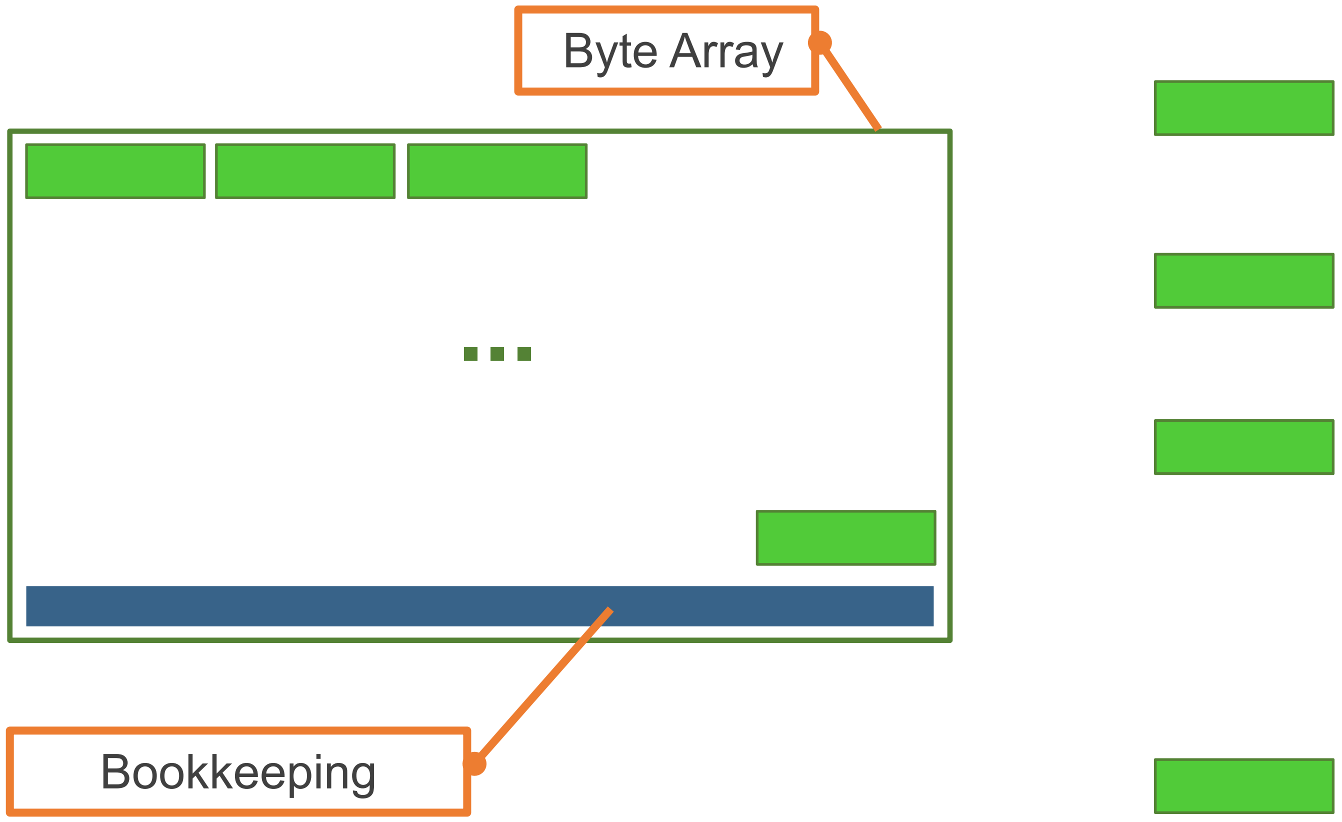
Space Cost of Working Data Sets

The space cost of the working data sets
is still about 4GB

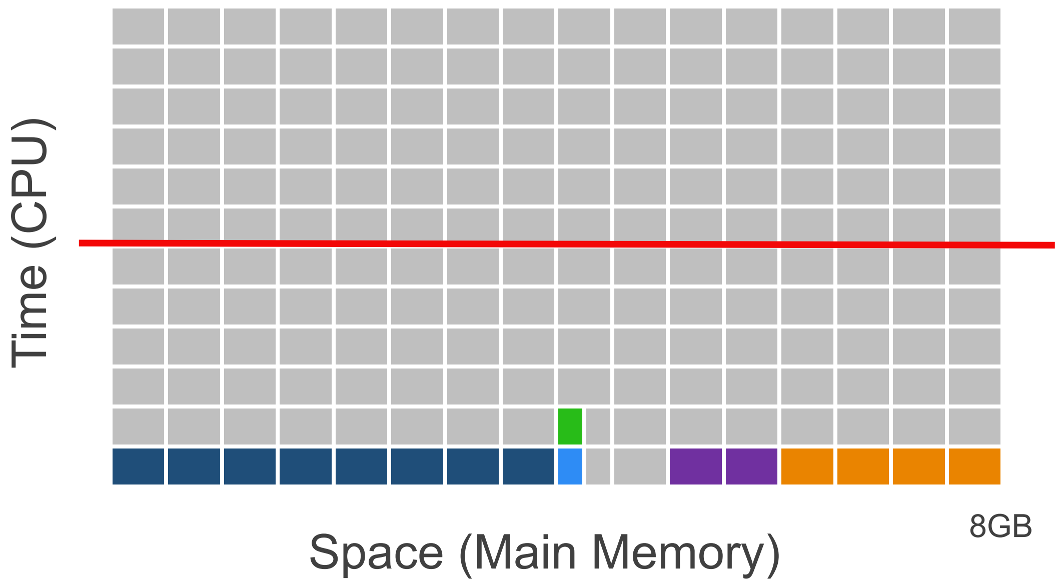
Leaves only 1GB
for data analysis

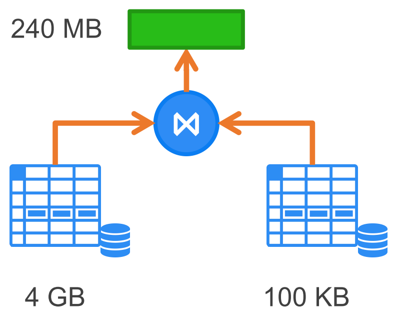


The New Storage Engine



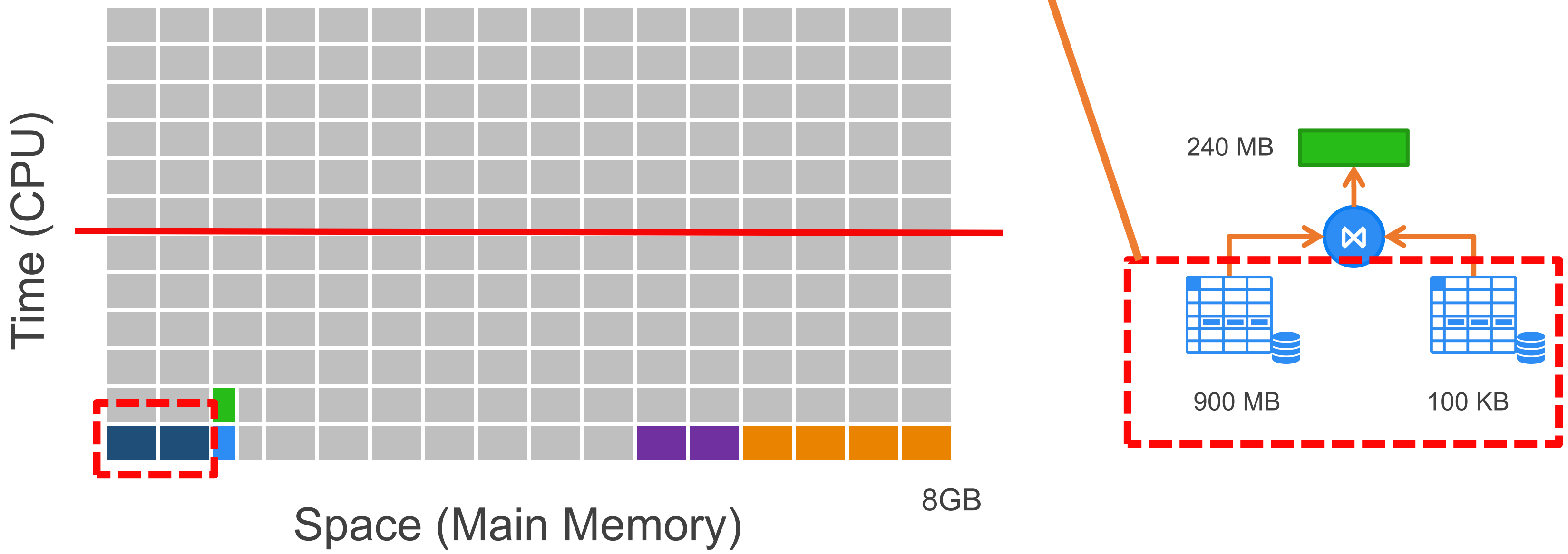
Space Cost of Working Data Sets





Space Cost of Working Data Sets

The new space cost of the working data sets is about **900 MB** instead of **4 GB**



Experiment

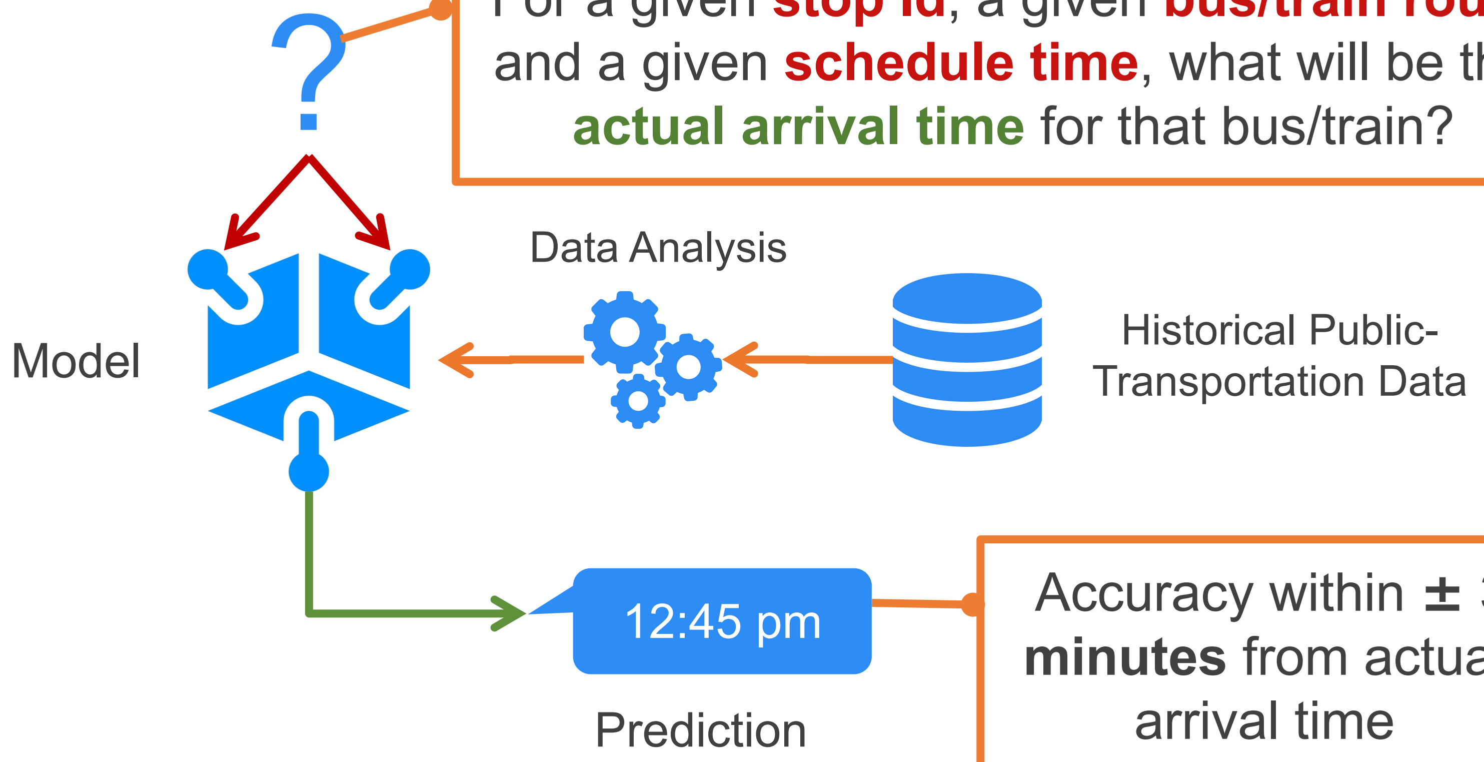
Realistic Use-Case

Experiment Goals

- Perform a realistic data-analysis use-case using **jSQL$_e$**.
- Keep the results of all intermediate results and the original data set in main memory.
- Simulate the same analysis in three other well-known systems and compare the space and time costs.

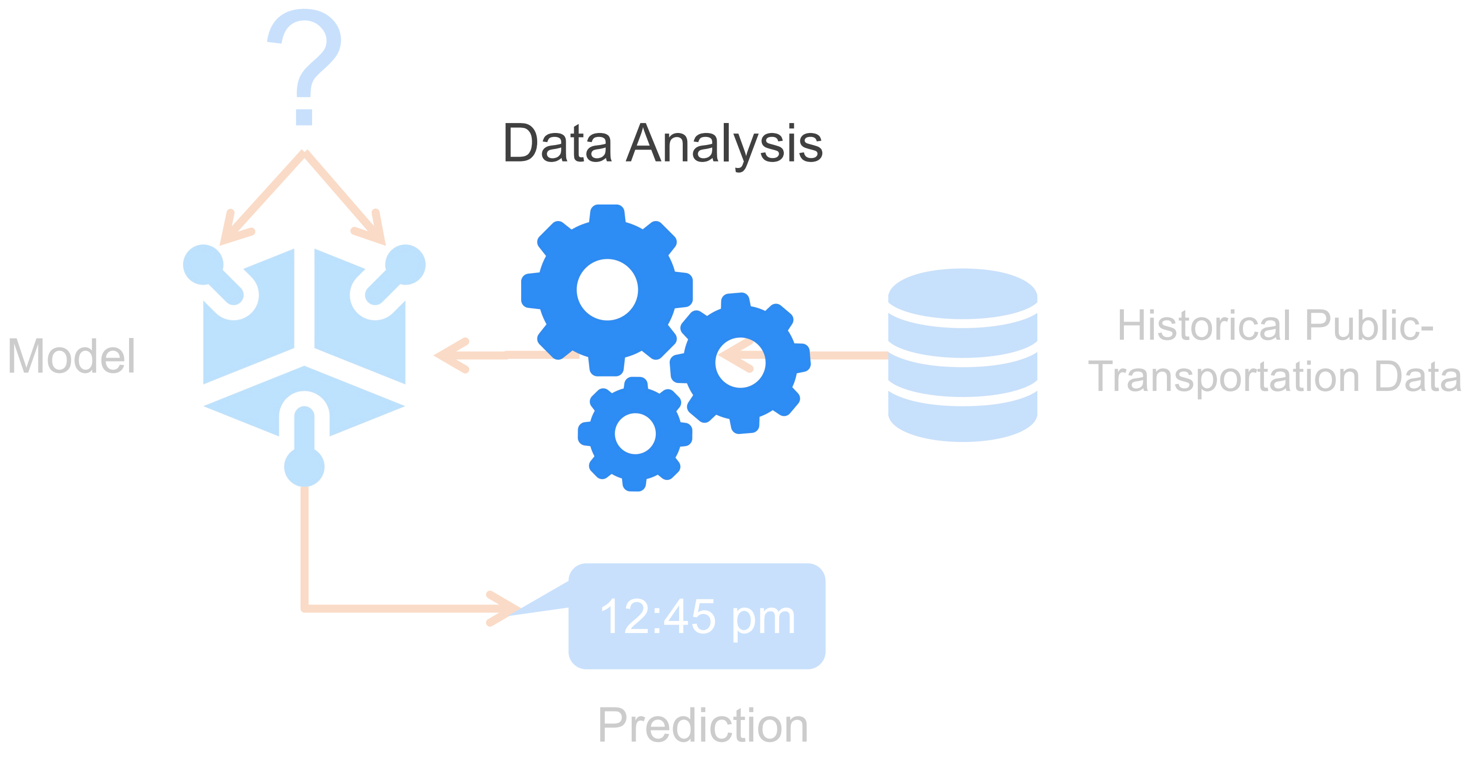
Data-Analysis Objective

For a given **stop Id**, a given **bus/train route**, and a given **schedule time**, what will be the **actual arrival time** for that bus/train?

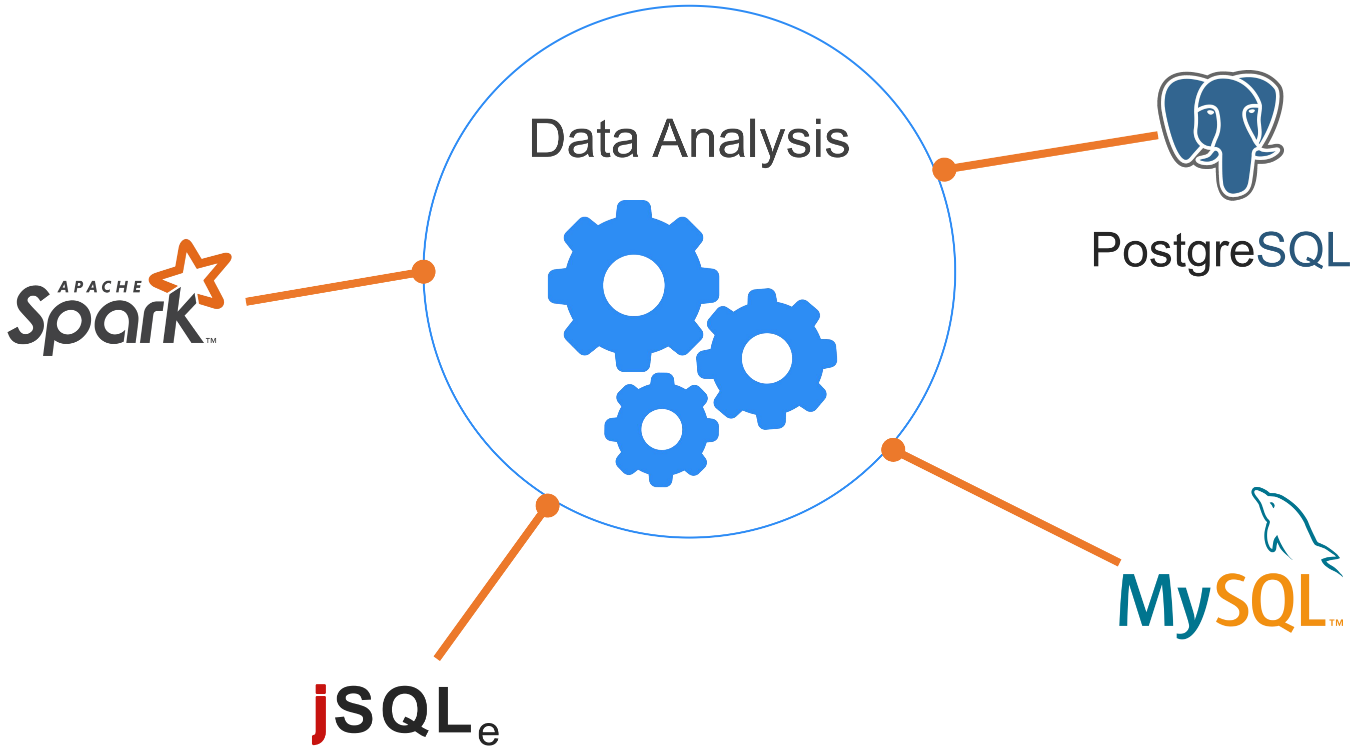


Accuracy within **± 3 minutes** from actual arrival time

Data-Analysis Objective



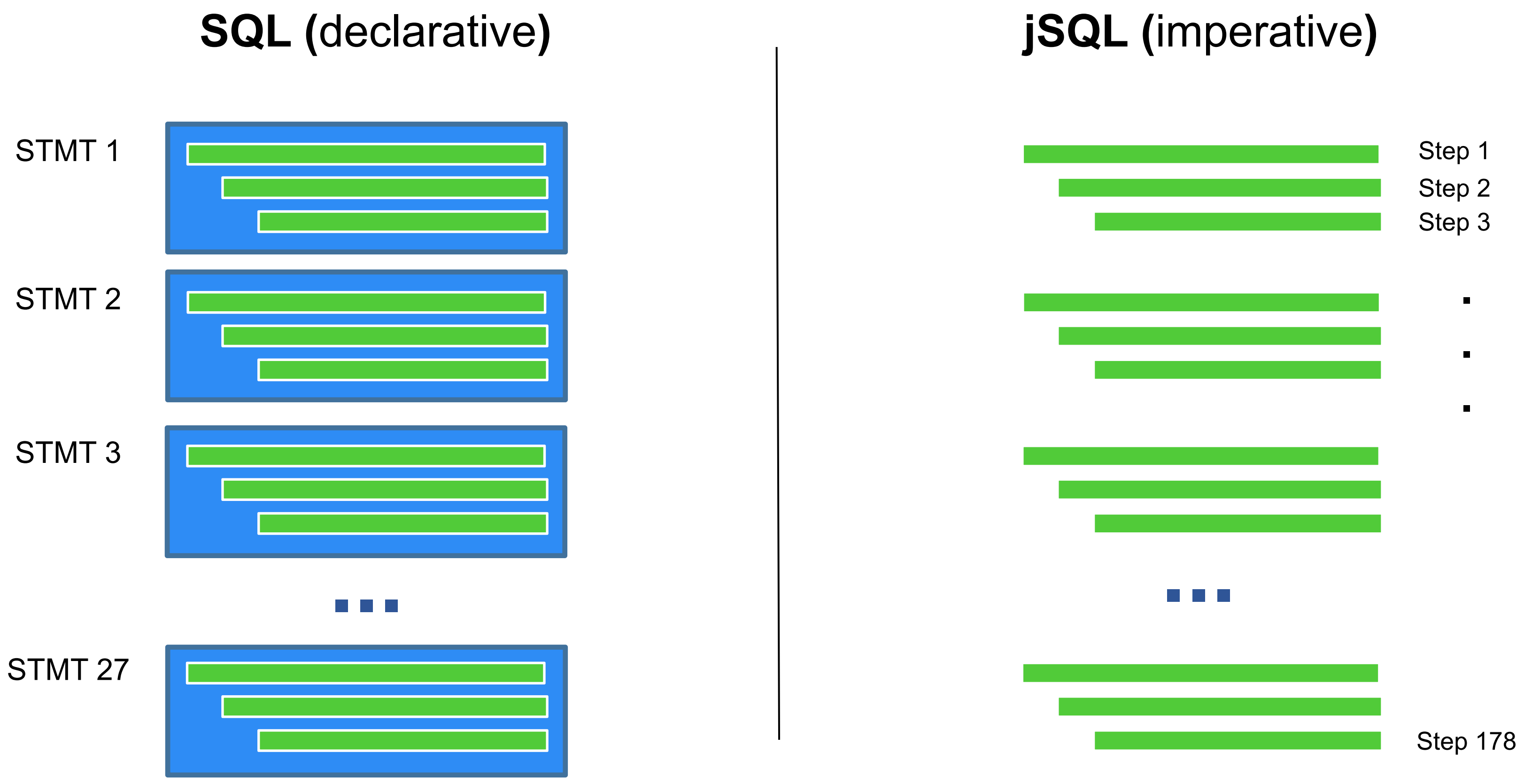
Data-Analysis Objective



Experiment Setup

- We used **6 months** of historical data from **TriMet.**
- We used a desktop computer
  - RAM: **8GB**
  - CPU: **4 cores, i5, 3.5GH**
- Space limit for data analysis is **6GB**
- We ran all four systems on a single core (one thread).

Experiment Setup



Experiment Setup

SQL (declarative)

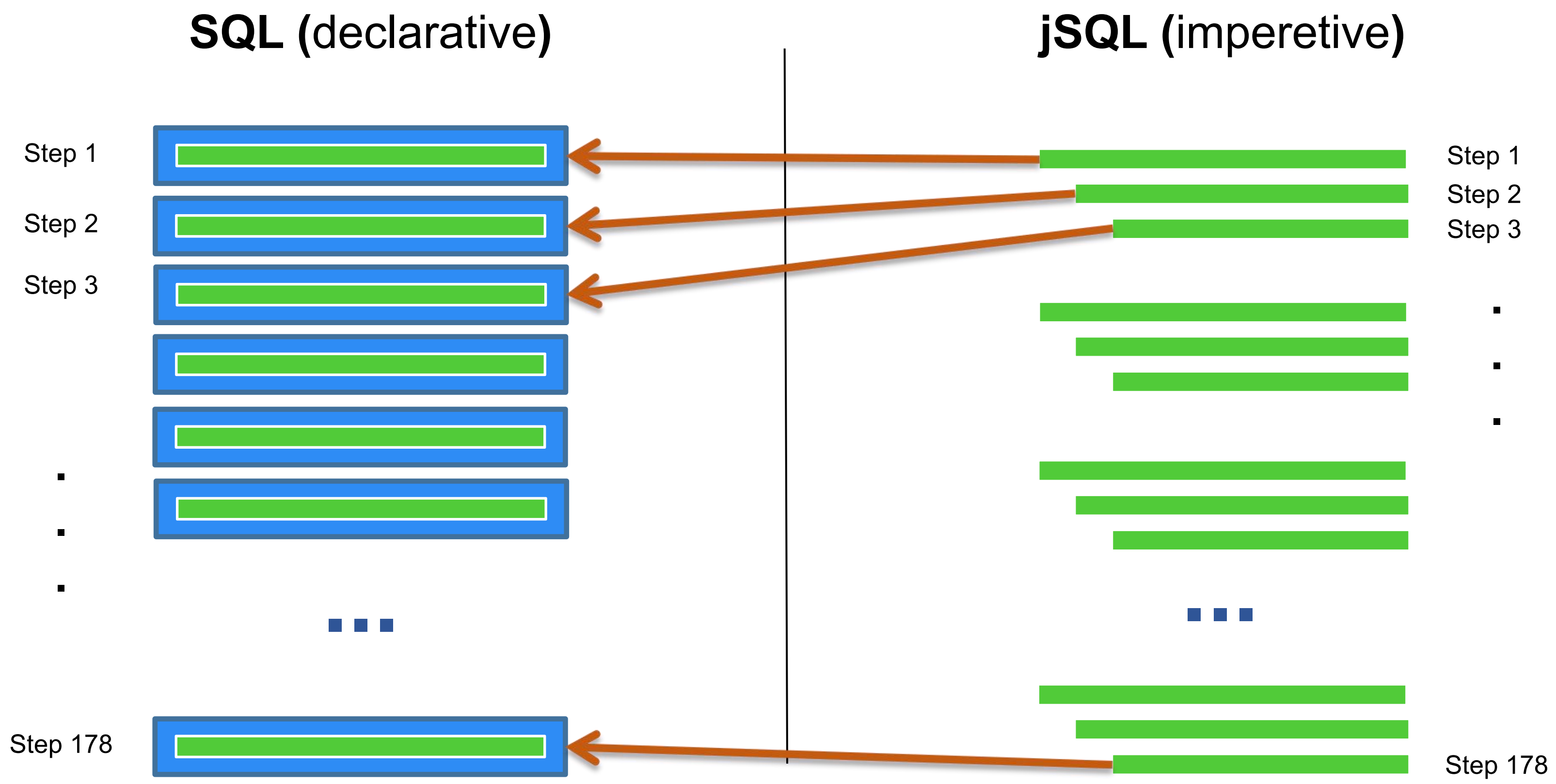
```
SELECT
  service_date, route_number,
  schedule_time, arrive_time
FROM
  historical_data
WHERE
  service_date = ‘2021-01-29’;
```

jSQL (imperative)

```
L1 = SELECT historical_data
  WHERE
    service_date = ‘2021-01-29’;
```

```
L2 = PROJECT L1 WITH
  service_date, route_number,
  schedule_time, arrive_time;
```

Experiment Setup



Experiment Setup

PostgreSQL (declarative)

```
CREATE TABLE L1 AS
SELECT * FROM historical_data
WHERE
  service_date = ‘2021-01-29’;
```

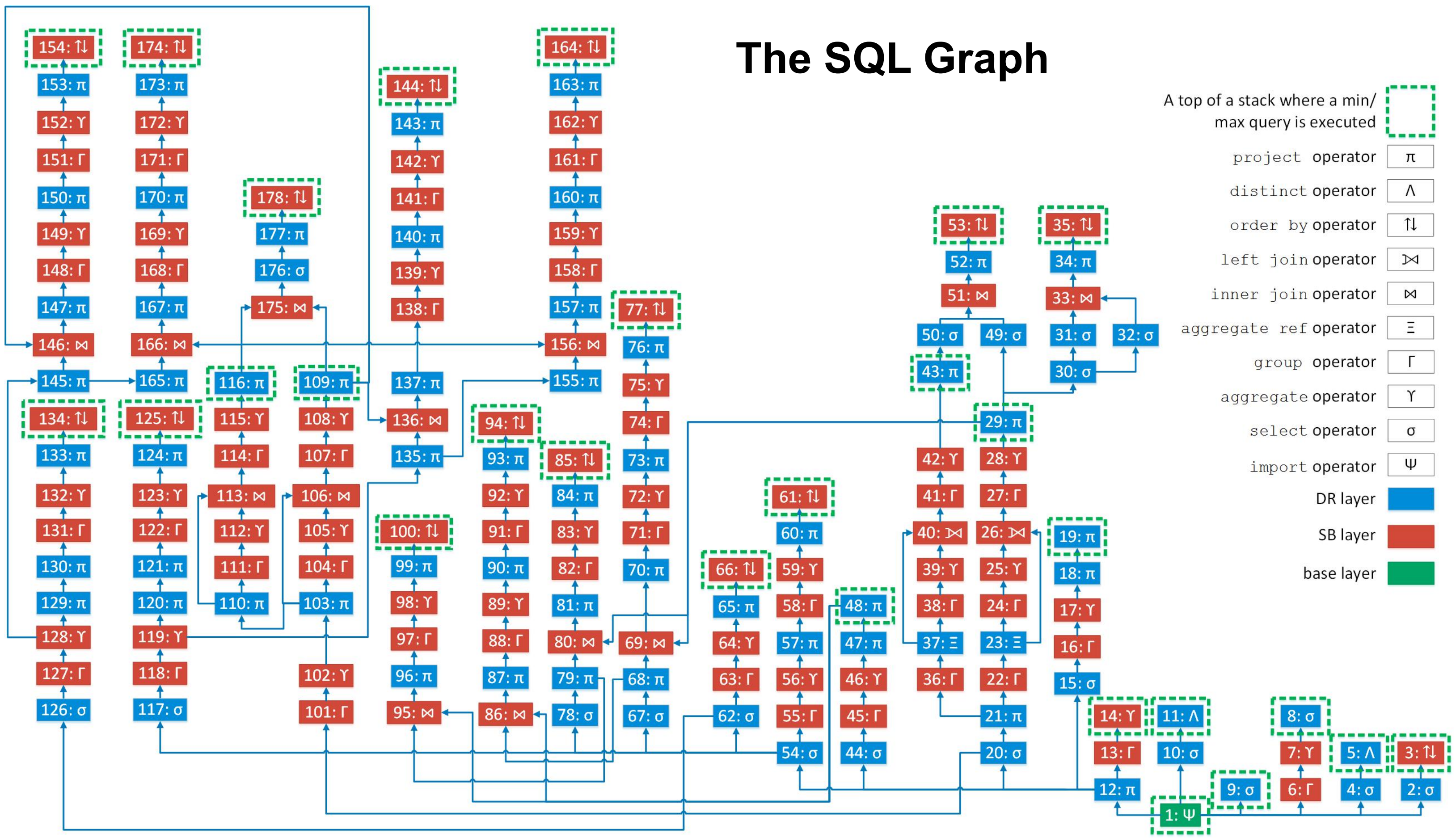
```
CREATE TABLE L2 AS
SELECT
  service_date, route_number,
  schedule_time, arrive_time
FROM L1;
```

jSQL (imperative)

```
L1 = SELECT historical_data
  WHERE
    service_date = ‘2021-01-29’;
```

```
L2 = PROJECT L1 WITH
  service_date, route_number,
  schedule_time, arrive_time;
```

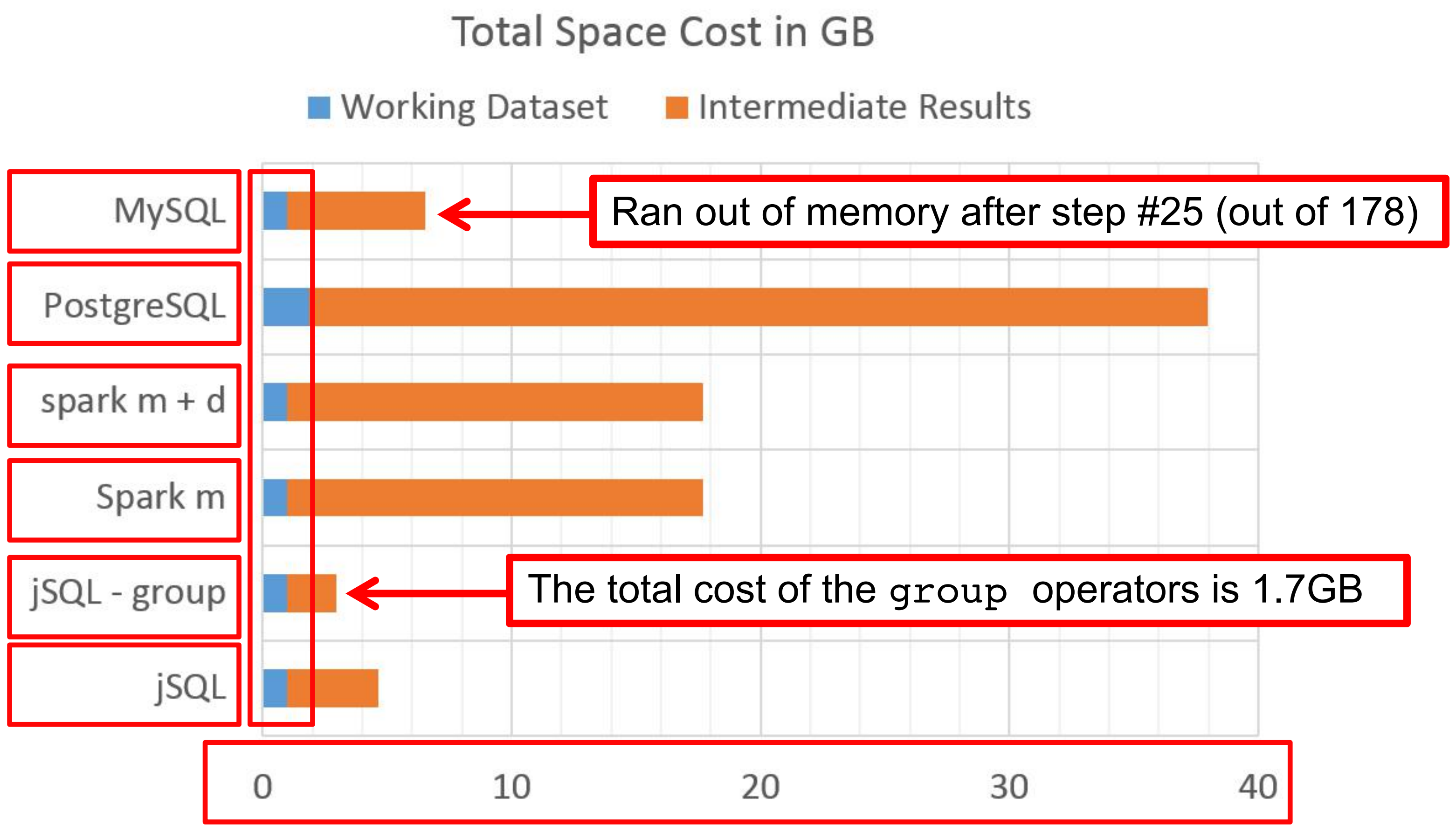
Experiment Setup



Experiment Results

Realistic Use-Case

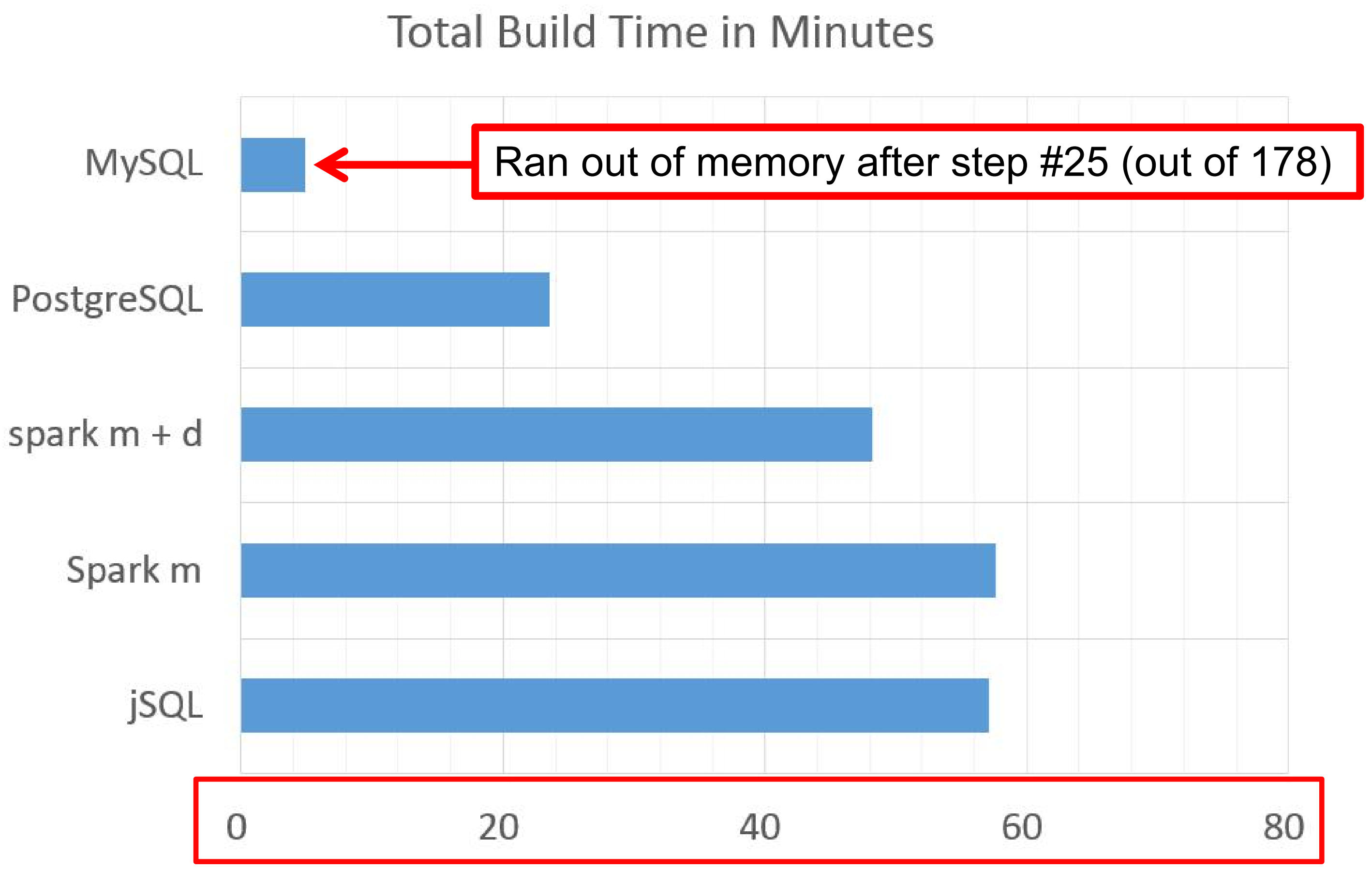
Total Space Cost



Ran out of memory after step #25 (out of 178)

The total cost of the `group` operators is 1.7GB

Total Build Time



Lessons Learned

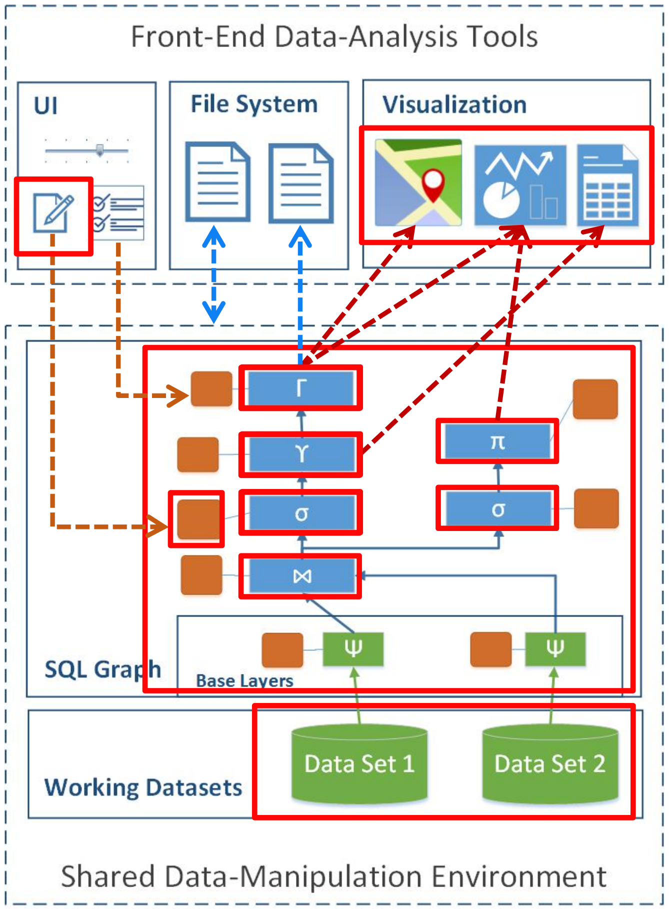
- jSQL$_e$ achived 92% reduction in space cost compared to PostgreSQL, while spending twice the build time.
- jSQL$_e$ achived 83% reduction in space cost compared to Spark, while spending more or less the same build time .
- MySQL? Just don’t use it.

Lessons Learned

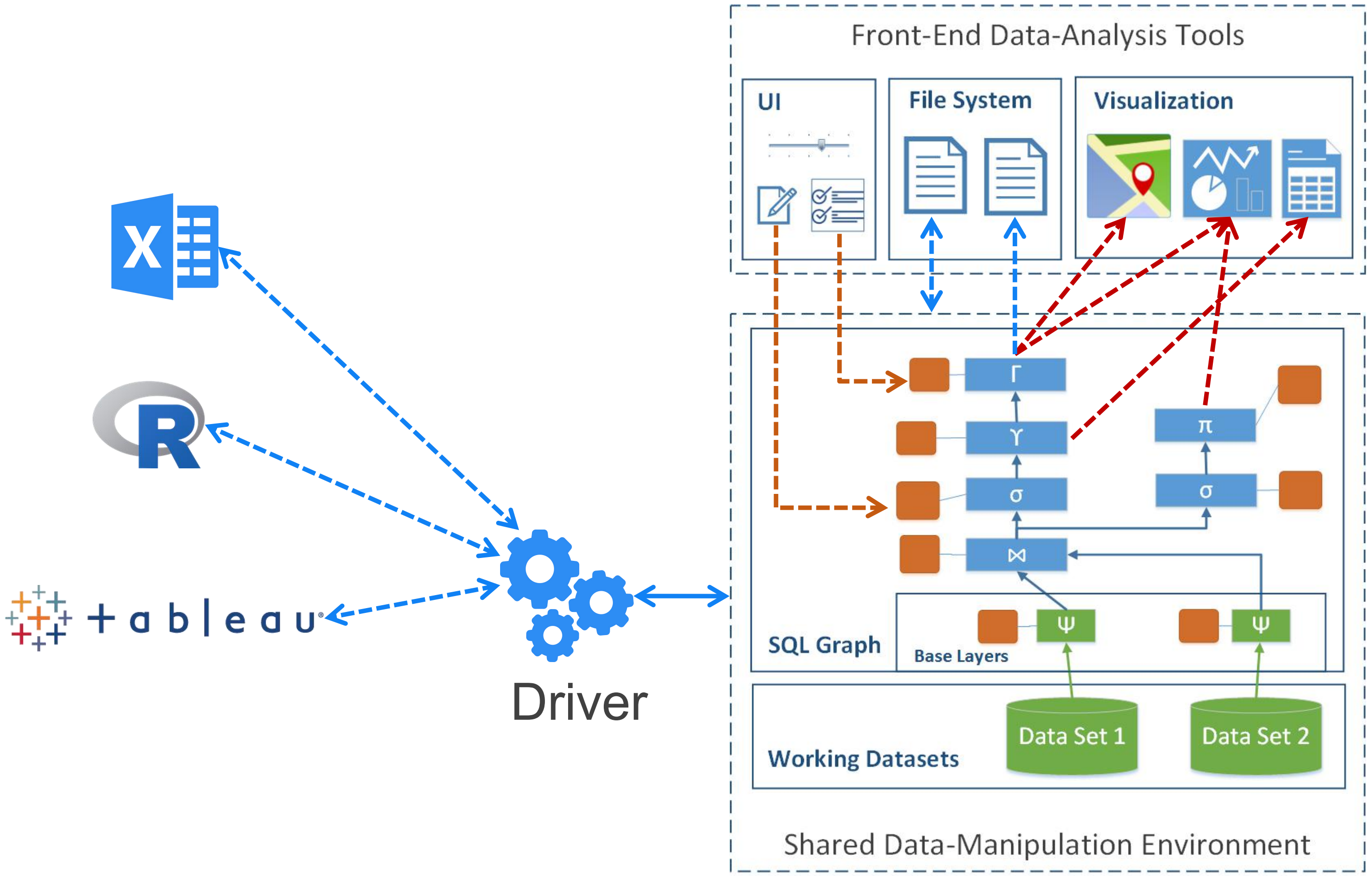
- Using current systems to do exploratory data analysis is extremely tedious.

Our Vision

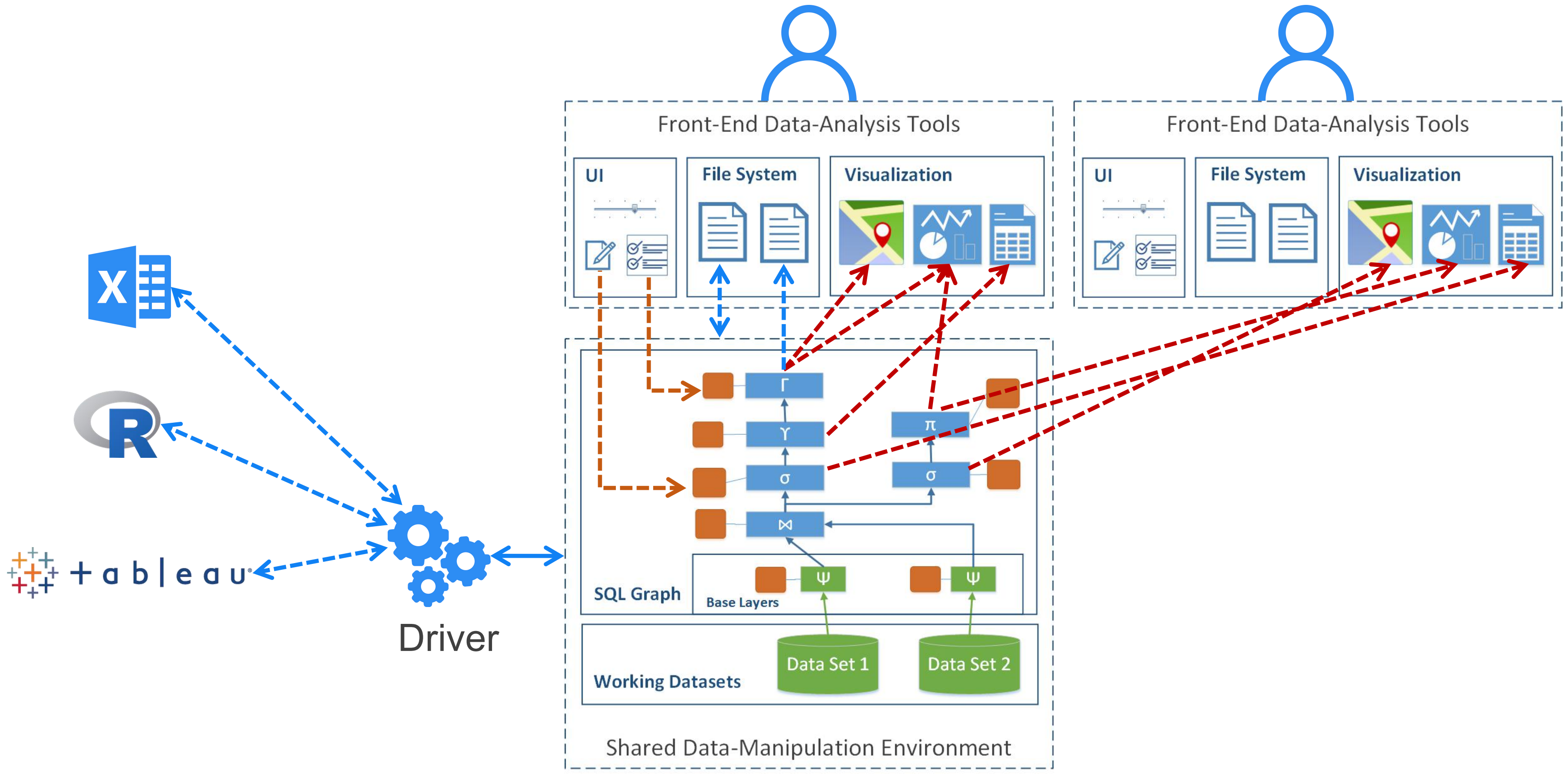
What Can We Do With jSQL$_e$?



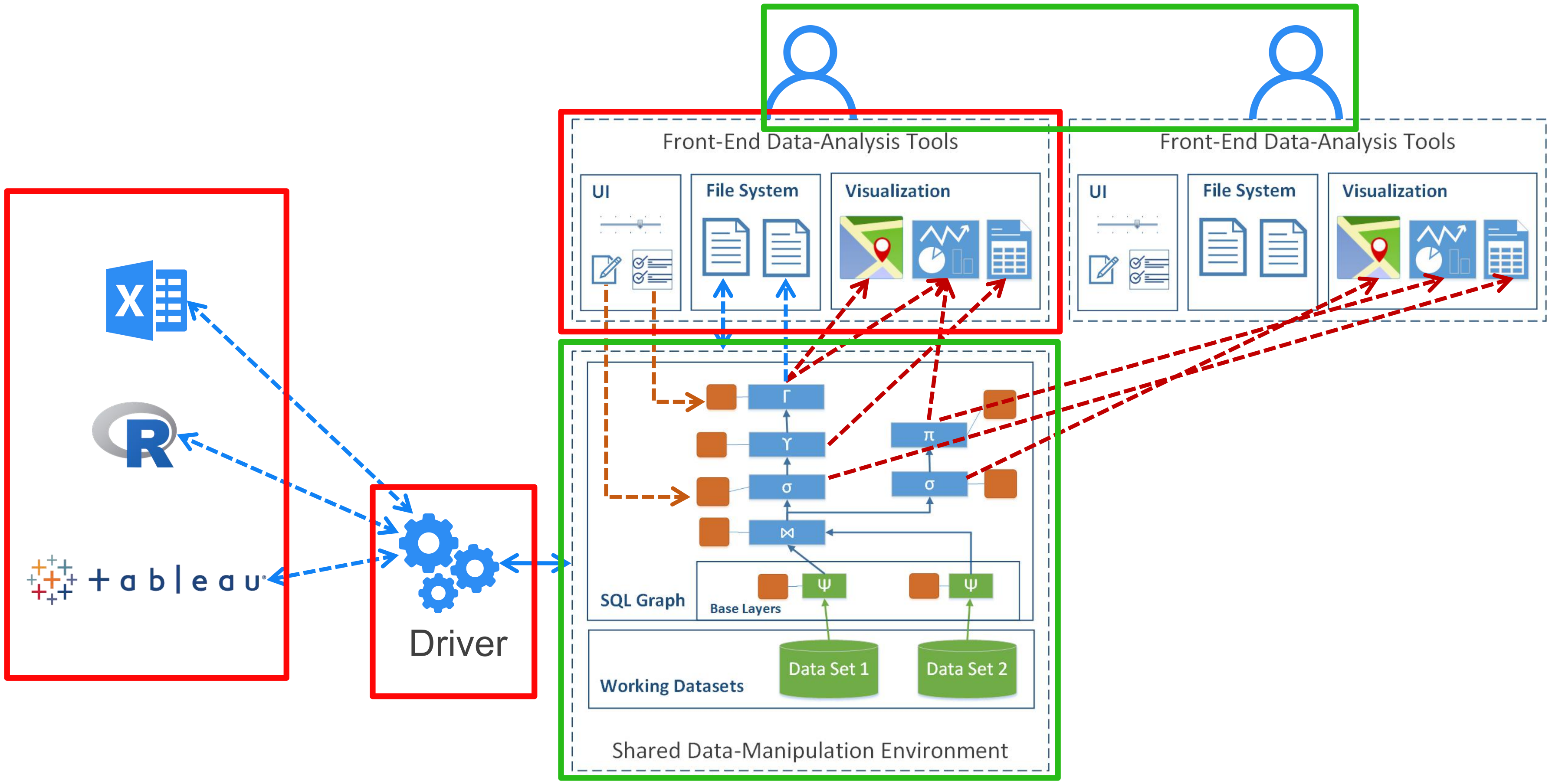
What Can We Do With jSQL$_e$?



What Can We Do With jSQL$_e$?



How Far Are We?



Thank You