

# Simulation study to use “REAL World Data” using RDB and Hadoop

Hiroshi Ohtsu<sup>1,2</sup>, Shiro Matsuya<sup>1</sup>

{ <sup>1</sup>Department of Data Science, <sup>2</sup>JCRAC Data Center },  
Center for Clinical Sciences,  
National Center for Global Health and Medicine (NCGM)

# Trademarks

---

- All product names, trademarks and registered trademarks are property of their respective owners. All company, product and service names used in this document are for identification purposes only.

# Contents

---

- Backgrounds
- Objectives
- Data used in this experiment
- A hypothetical research
- System configuration
- Measure the performance of SQL
- Exporting data to SAS/EG
- Limitations
- Conclusions
- Future Plan

# Contents

---

- Backgrounds
- Objectives
- Data used in this experiment
- A hypothetical research
- System configuration
- Measure the performance of SQL
- Exporting data to SAS/EG
- Limitations
- Conclusions
- Future Plan

# Past

薬剤疫学 Jpn J Pharmacoepidemiol, 10(1) July 2005 : 41

●資料

## 市販後安全性研究に関するガイドラインの提案

くすりの適正使用協議会 薬剤疫学部会 海外情報研究会\*

### Proporsal of Guidelines for Post-Marketing Safety Observational Studies

Pharmacoepidemiology Commit

\* Pharmacoepidemiology Commi  
4-2 Nihonbashi-Kodenmacho, C

日本統計学会誌  
第36巻, 第2号, 2007年3月  
205頁~217頁

## 降圧薬の使用成績調査データベース構築とその活用例

藤田利治<sup>1)</sup>, 真山武志<sup>2)</sup>

A database of anti-hypertensive drugs from Drug Use Investigations  
and its practical use example

Toshiharu Fujita and Takeshi Mayama



研究に参加したい方へ

参加を希望する方へ

参加するには（日程表）

参加に必要な資料

予約はこちらから



症例登録状況

地域別登録状況

施設毎登録状況（ログインが必要）

お問い合わせ



研究への参加を希望される方へ

# 2型糖尿病の方へ

**IoT※<sup>1</sup>を活用した糖尿病重症化予防  
に関する臨床研究に  
ご協力いただける方を募集しています！**

予約はこちらから



**募集期間：2018年1月～12月**

## 参加いただける方の主な条件

- 1 日常的にスマートフォンを使用している方  
[http://prism-j.umin.jp/material/available\\_phone.pdf](http://prism-j.umin.jp/material/available_phone.pdf) 掲載の  
端末を推奨
- 2 HbA1c※<sup>2</sup>が6.0%以上  
8.9%以下の2型糖尿病の方
- 3 20歳以上75歳未満の方

ご参加いただける方には  
「活動量計」「体重体組成計」  
「血圧計」を無償貸与しますので、  
2019年12月まで測定を  
お願いいたします



# Current situations

---

- GCP renovation
  - We should focus various clinical researches.
  - Data is bigger than classical clinical “Trials”.
- ICT (IoT) generates “more, more and more” data from our real life.
  - we should get useful information from this.

# FDA's paper (2016-2017)

## Real world evidence

---

*The NEW ENGLAND JOURNAL of MEDICINE*

### SOUNDING BOARD

## Real-World Evidence — What Is It and What Can It Tell Us?

Rachel E. Sherman, M.D., M.P.H., Steven A. Anderson, Ph.D., M.P.P.,  
Gerald J. Dal Pan, M.D., M.H.S., Gerry W. Gray, Ph.D., Thomas Gross, M.D., M.P.H.,  
Nina L. Hunter, Ph.D., Lisa LaVange, Ph.D., Danica Marinac-Dabic, M.D., Ph.D.,  
Peter W. Marks, M.D., Ph.D., Melissa A. Robb, B.S.N., M.S., Jeffrey Shuren, M.D., J.D.,  
Robert Temple, M.D., Janet Woodcock, M.D., Lilly Q. Yue, Ph.D., and Robert M. Califf, M.D.

The term “real-world evidence” is widely used by those who develop medical products or who study, deliver, or pay for health care, but its specific meaning is elusive. We believe it refers to information on health care that is derived from multiple sources outside typical clinical research

shortage of researchers with adequate methodologic savvy could result in poorly conceived study and analytic designs that generate incorrect or unreliable conclusions. Accordingly, if we are to realize the full promise of such evidence, we must be clear about what it is and how it can



# Real world evidence とは

Issue Brief

**NEHI** THE NETWORK FOR EXCELLENCE IN HEALTH INNOVATION

September 2015

**Real World Evidence: A New Era for Health Care Innovation**

**Real world evidence is not just "Big Data" – It's the integration of multiple sources of data**

Executive Summary

Real World Evidence (RWE), a potentially transformative force in U.S. health care

# Current situations

---

- We are just started to ride on the sea of big data.
- SAS software ( or recent BI tools)
  - From real world “data” to “EVIDENCE”.
  - But, we must work hard to look for evidence from “BIG data”.
    - Reduce cost
    - Reduce time
    - Reduce ...

# Motivation

---

- We are going to make a platform for analysis.
  - To use SAS BI tools ( or some BI tools ) wisely and to get better performance, we need to know current database techniques.
  - And, we are trying to establish a process to analyze some clinical research.

# Contents

---

- Backgrounds
- **Objectives**
- Data used in this experiment
- A hypothetical research
- System configuration
- Measure the performance of SQL
- Exporting data to SAS/EG
- Limitations
- Conclusions
- Future Plan

# Objectives

---

- To measure the performance of Apache Impala\* and MySQL on creating datasets for a hypothetical research and to know the outline of the performance.

\*Apache Impala is a modern open-source MPP SQL engine architected the ground up for the Hadoop data processing environment

from “Impala: A modern, Open-Source SQL Engine for Hadoop (Marcel Kornacker et al. 2015)”

# Contents

---

- Backgrounds
- Objectives
- **Data used in this experiment**
- A hypothetical research
- System configuration
- Measure the performance of SQL
- Exporting data to SAS/EG
- Limitations
- Conclusions
- Future Plan

# Data Used in this Experiment

---

- 50 million patients data generated by OSIM2, a data simulator, developed by the OMOP statistical methods group.
- The data has been used as a testbed for evaluating and comparing different estimation procedures for many purposes.
- Available from <ftp://ftp.ohdsi.org/osim2>

# Files of OSIM2

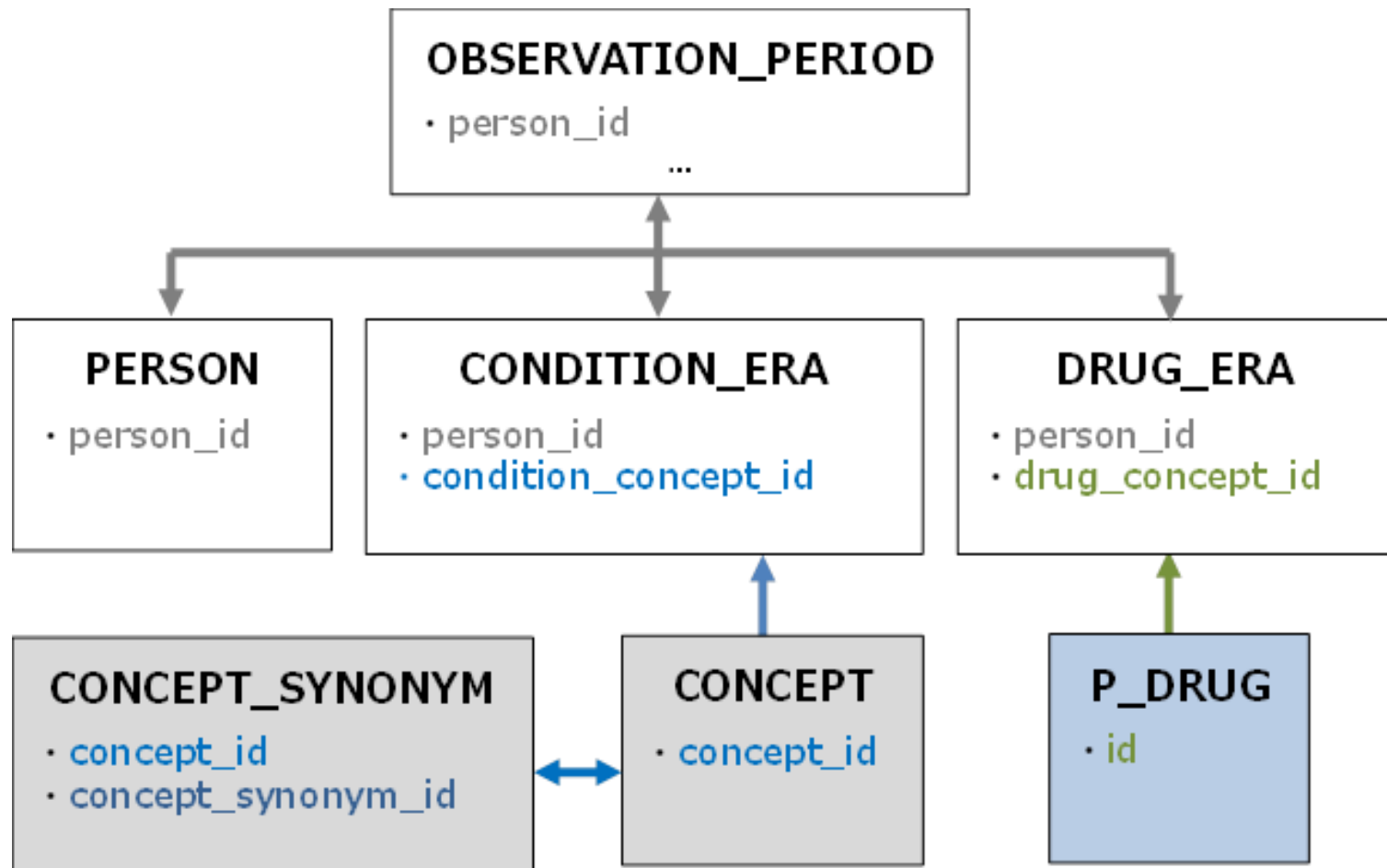
---

| File/Table Name    | Rows        | Size(GB) | Comments  |
|--------------------|-------------|----------|-----------|
| PERSON             | 50,000,000  | 2        |           |
| OBSERVATION_PERIOD | 50,000,000  | 3.2      |           |
| DRUG_ERA           | 746,519,618 | 48       |           |
| CONDITION_ERA      | 930,215,314 | 60       |           |
| CONCEPT            | 786,846     | 0.1      | SNOMED-CT |
| CONCEPT_SYNONYM    | 1,800,214   | 0.1      | SNOMED-CT |
| P_DRUG*            | 76,899      | 0.01     | RxNorm    |

\* P\_DRUG was made manually by presenters.



# Relation of Tables



# Contents

---

- Backgrounds
- Objectives
- Data used in this experiment
- **A hypothetical research**
- System configuration
- Measure the performance of SQL
- Exporting data to SAS/EG
- Limitations
- Conclusions
- Future Plan

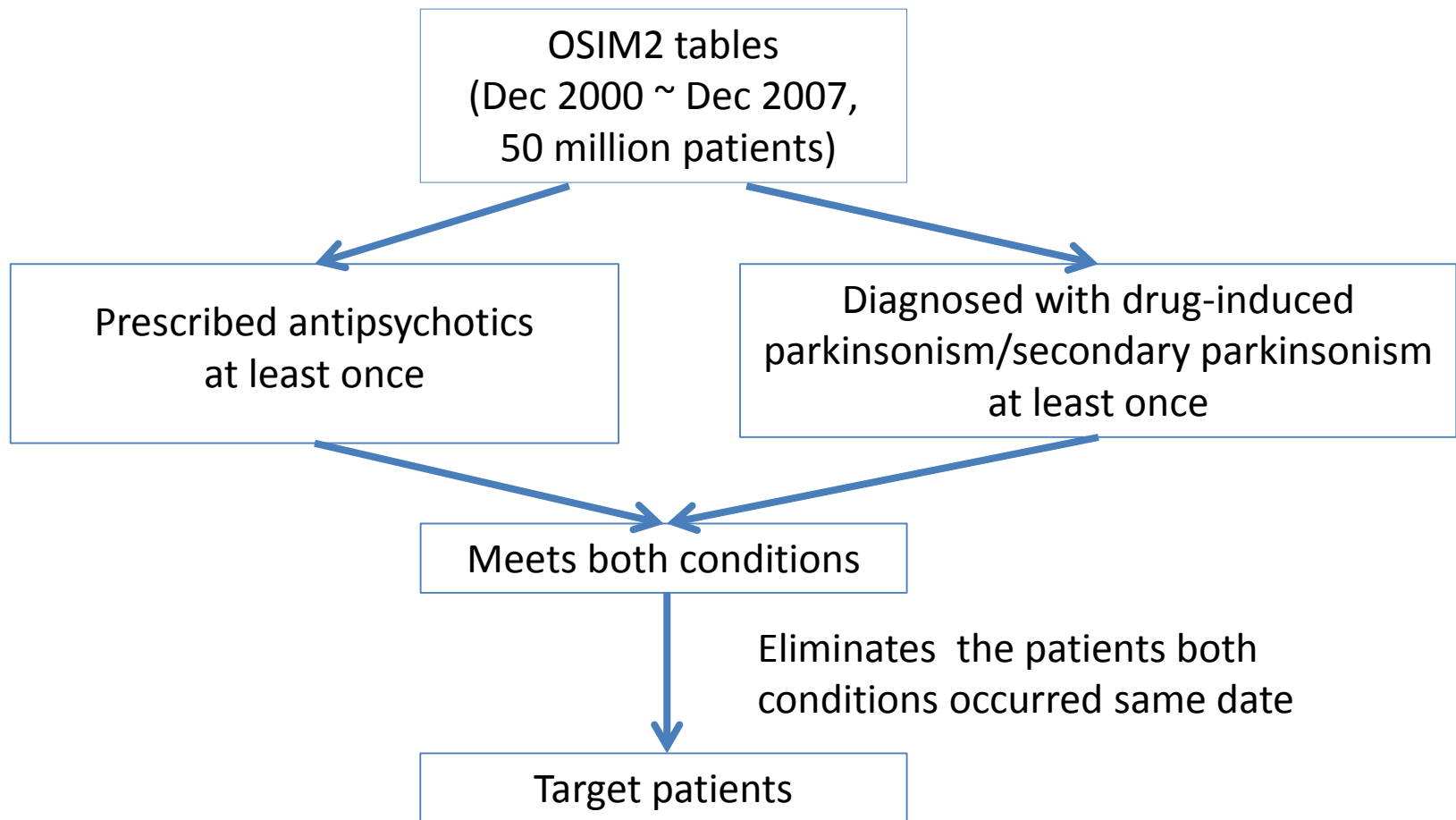
# A Hypothetical Research

---

- Extract the patients who were prescribed antipsychotics, then diagnosed with drug-induced parkinsonism or secondary parkinsonism.
- The dataset is used for analyzing with Sequence Symmetry Analysis (SSA) .
  - We are not going to talk much this part today.
- This hypothetical research is based on the report of Japan Pharmaceutical Manufacturers Association.

see  
[http://www.jpma.or.jp/medicine/shinyaku/tiken/allotment/pdf/real\\_world\\_data.pdf](http://www.jpma.or.jp/medicine/shinyaku/tiken/allotment/pdf/real_world_data.pdf) (in Japanese)

# Creating the Dataset for Analysis



# Antipsychotics of this Research

---

## ▪ Atypical (22)

risperidone, olanzapine, quetiapine, aripiprazole, perospirone, zotepine, blonanserin, amisulpride, amoxapine, asenapine, cariprazine, clozapine, iloperidone, lurasidone, melperone, nemonapride, paliperidone, remoxipride, sertindole, sultopride, trimipramine, ziprasidone

## ▪ Typical (28)

haloperidol, chlorpromazine, levomepromazine, fluphenazine, bromperidol, perphenazine, sulpiride, clocapramine, pimozide, tiapride, timiperone, prochlorperazine, propericiazine, spiperone, mosapramine, chlorprothixene, mesoridazine, periciazine, promazine, thioridazine, loxapine, molindone, thiothixene, droperidol, flupentixol, thioproperazine, trifluoperazine, zuclopenthixol

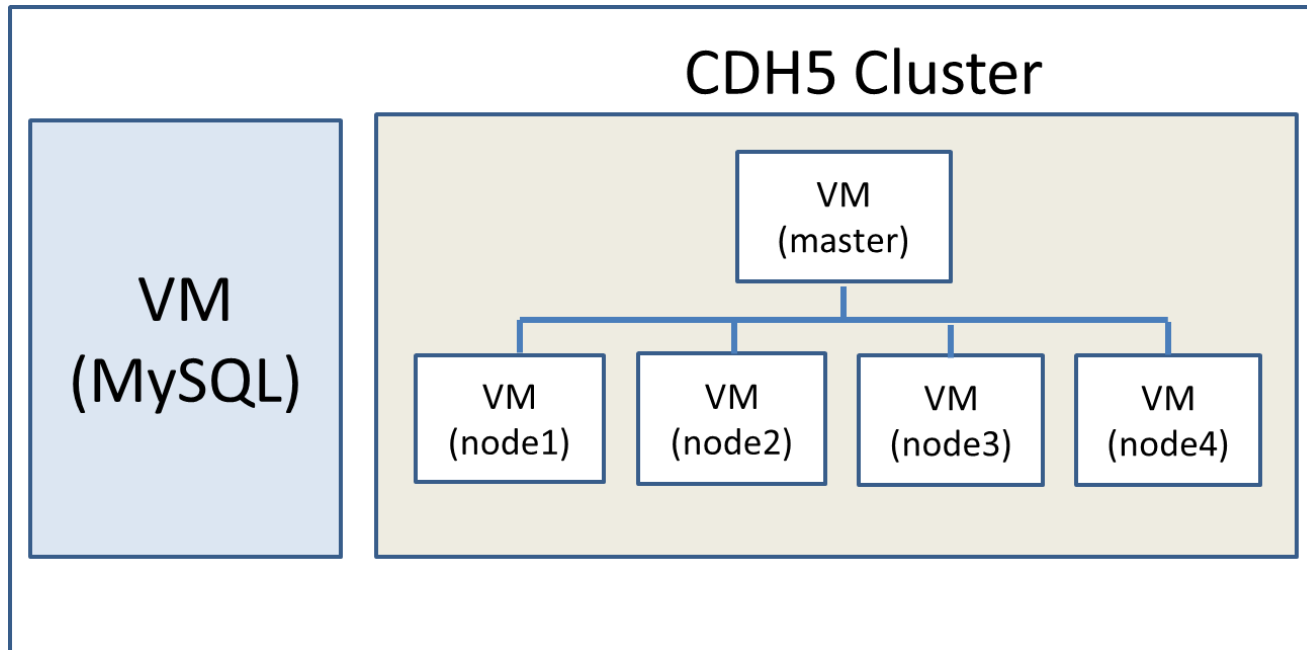
# Contents

---

- Backgrounds
- Objectives
- Data used in this experiment
- A hypothetical research
- **System configuration**
- Measure the performance of SQL
- Exporting data to SAS/EG
- Limitations
- Conclusions
- Future Plan

# System Configuration

## Host PC



CDH5 is Cloudera's 100% open source platform distribution, including Apache Hadoop and built specifically to meet enterprise demands.

# Hardware/Software

---

- Host PC
  - OS : Microsoft Windows 10 Pro (64bit)
  - CPU : Intel(R) Core(TM) i7-7820X CPU  
@ 3.60GHz, 3600 MHz, 8 Core, 16 Logical Processor
  - RAM : 128 GB (not ECC)
  - HDD : 2TB (for VM)
  - VM : VMware Workstation Pro 14 Pro
- MySQL 5.7.21 (On VM)
  - OS : CentOS6.9 (guest)
  - RAM : 64GB
  - CPU : 4 Core / 3 Logical (total 12 processors)
  - HDD : 500GB



# CDH5 on VM

---

- Manager Node \* 1

OS : CentOS6.9 (guest)  
RAM : 56GB  
CPU : 2 Core / 2 Logical (total 4 processors)  
HDD : 500GB

Actual consumption of physical memory that has been allocated to the virtual machine. The same is true of disk allocation.

- Data Node \* 4

OS : CentOS6.9 (guest)  
RAM : 20GB  
CPU : 1 Core / 2 Logical (total 2 processors)  
HDD : 500GB

- CDH5 provides Hadoop, HDFS, Hbase, Yarn, Sqoop, Hue, Hive, Impala, Spark, Oozie, etc.

# Data Load to MySQL and Hadoop

---

## DRUG\_ERA

| File Name | Rows        | Size(GB) |
|-----------|-------------|----------|
| DRUG_ERA  | 746,519,618 | 48       |

## Loading Time

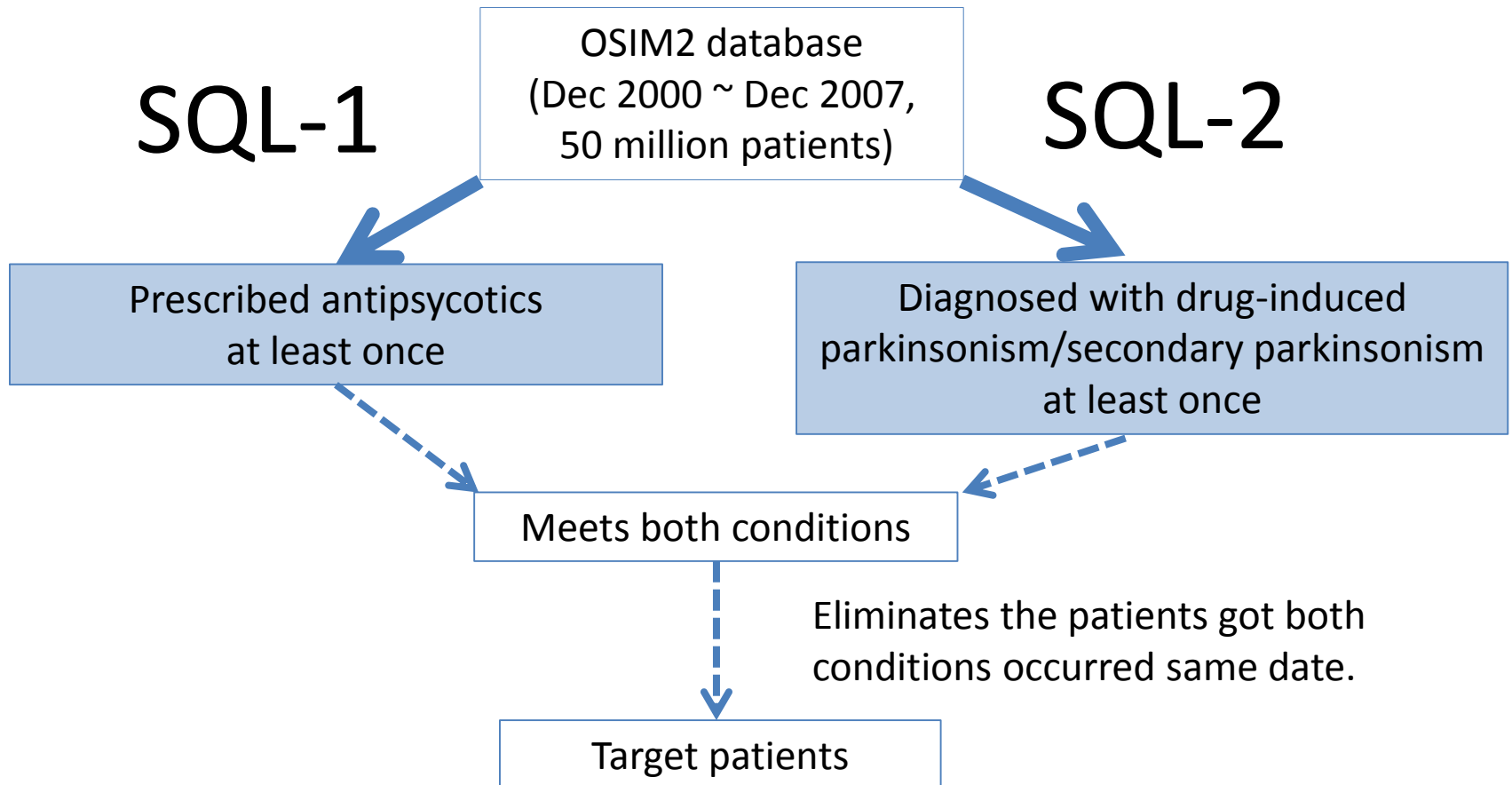
| DB     | Time                  | Comments   |
|--------|-----------------------|--|
| MySQL  | 120 min<br>(no index) | DRUG_ERA was created on <a href="#">CSV engine</a> and alter storage engine type <a href="#">from CSV to INNODB</a> . Then indexes were created on that table. <a href="#">Creating indexes took many hours!</a> |
| Hadoop | 90 min                | <a href="#">Apache sqoop</a> was used to <a href="#">load DRUG_ERA on MySQL to Hadoop</a> . Parquet and snappy were used for data format and compression method, respectively.                                   |

# Contents

---

- Backgrounds
- Objectives
- Data used in this experiment
- A hypothetical research
- System configuration
- **Measure the performance of SQL**
- Exporting data to SAS/EG
- Limitations
- Conclusions
- Future Plan

# Measure the Performance of SQL



# SQL-1

---

```
CREATE TABLE _drug AS  
    SELECT a.person_id, a.drug_concept_id,  
           drug_era_start_date, b.vocabulary  
    FROM drug_era a INNER JOIN p_drug b  
    ON a.drug_concept_id = b.id
```

| Table Name | Rows        | Original file Size |
|------------|-------------|--------------------|
| DRUG_ERA   | 746,519,618 | 48 GB              |
| P_DRUG     | 76,899      | 8 MB               |

# MySQL Result (SQL-1)

| No.  | time (hh:mm) |
|------|--------------|
| 1    | 2 : 16       |
| 2    | 2 : 28       |
| 3    | 2 : 03       |
| avg. | 2 : 16       |

## Measurement Conditions

- Measured 3 times and took the average.
- Each SQL was executed through *mysql* client.
- PC and VM were cold-started each time to avoid the influence of HDD cache and MySQL buffers.

```
CREATE TABLE _drug AS
  SELECT a.person_id, a.drug_concept_id, drug_era_start_date, b.vocabulary
  FROM drug_era a INNER JOIN p_drug b
  ON a.drug_concept_id=b.id ;
```

Query OK, 7260167 rows affected (2 hours 16 min 26.89 sec)

Records: 7260167 Duplicates: 0 Warnings: 0

← cumulative total number of patients

# MySQL Results(cont.)

---

- If PC and VM are NOT cold-started, we have better performance.

```
CREATE TABLE _drug_cont AS
```

```
  SELECT a.person_id, a.drug_concept_id, drug_era_start_date, b.vocabulary  
  FROM drug_era a INNER JOIN p_drug b  
  ON a.drug_concept_id=b.id;
```

```
Query OK, 7260167 rows affected (1 min 35.88 sec)
```

```
Records: 7260167 Duplicates: 0 Warnings: 0
```

- Possible reason of the better performance
  - HDD cash and MySQL buffer pools are effective

# SQL-1 Impala Result

---

| No.  | time( <a href="#">mm:ss</a> ) |
|------|-------------------------------|
| 1    | 2 : 41                        |
| 2    | 2 : 51                        |
| 3    | 1 : 26                        |
| avg. | 2 : 19                        |

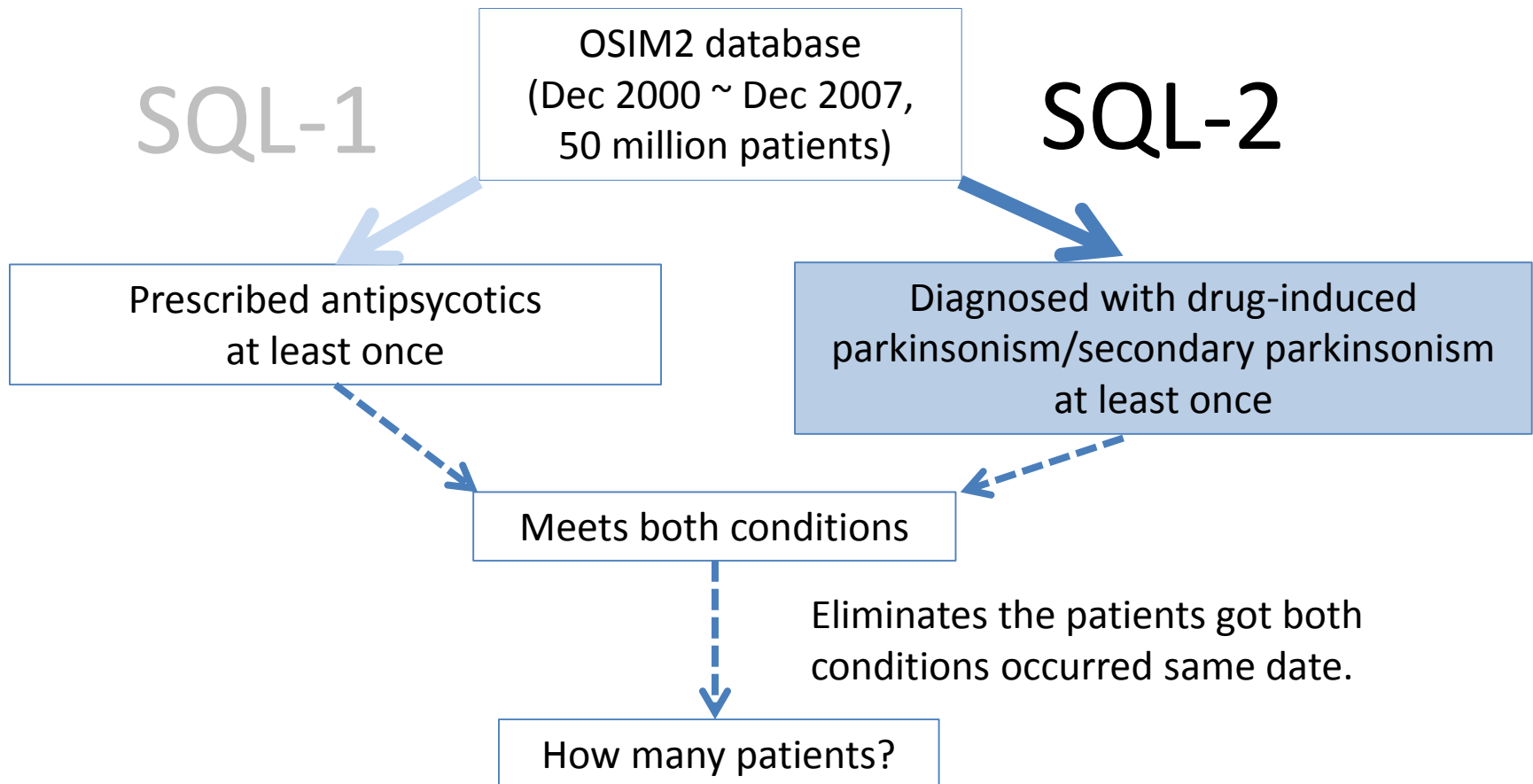
## Measurement Conditions

- Measured 3 times and took the average.
- Each SQL was executed through [HUE](#) workbench.
- PC and VMs were cold-started each time to avoid the influences of HDD cache and Hadoop cache.

- If PC and VMs are NOT cold-started, we also have better performance ([less than 1 min.](#))
- Possible reason for better performance
  - HDD cash and HDFS distributed caches are effective.



# Measure the Performance of SQL (reshown)



# SQL-2

```
CREATE TABLE p_cond AS
  SELECT concept_id, concept_name, vocabulary_id FROM concept
  WHERE (LOWER(CONCEPT_NAME) LIKE '%drug-induced parkinsonism%'
  OR LOWER(CONCEPT_NAME) LIKE '%secondary parkinsonism%');
```

## SQL-2:

```
CREATE TABLE _condition as
  SELECT a.person_id, condition_era_start_date
  FROM condition_era a
  INNER JOIN p_cond b
  ON a.condition_concept_id = b.concept_id
```

| Table Name    | Rows        | Original File Size |
|---------------|-------------|--------------------|
| CONDITION_ERA | 930,215,314 | 60 GB              |
| P_COND        | 13          | N/A                |

# Results of SQL-2

---

| No.  | Time(mm:ss) |        |
|------|-------------|--------|
|      | MySQL       | Impala |
| 1    | 5:52        | 2:41   |
| 2    | 3:06        | 4:02   |
| 3    | 4:37        | 3:02   |
| Avg. | 4:32        | 3:15   |

# Performance Comparison

| SQL   | Time(hh:mm:ss) |          |
|-------|----------------|----------|
|       | MySQL          | Impala   |
| SQL-1 | 02:16:00       | 00:02:19 |
| SQL-2 | 00:04:32       | 00:03:15 |

SQL-1:

$$(2*60 + 16)*60 / (2*60 + 19)*5 \approx 11.7$$

Divided by the number of VMs

Roughly speaking, **Impala is 11 times faster than MySQL!?**

- Of course the result can NOT be generalized.
  - **It's not really a fair comparison** because the number of VMs are different, resource allocations are different and ...

# MySQL Query Plan of SQL-1

EXPLAIN

```
SELECT a.person_id, a.drug_concept_id, drug_era_start_date, b.vocabulary  
FROM drug_era a INNER JOIN p_drug b ON a.drug_concept_id=b.id ;
```

\*\*\*\*\* 1. row \*\*\*\*\*

|               |          |                   |
|---------------|----------|-------------------|
| table         | : b      | # p_drug table    |
| type          | : ALL    | # full table scan |
| possible_keys | : idx_id | # available index |
| key           | : NULL   | # no index uses   |

\*\*\*\*\* 2. row \*\*\*\*\*

|       |                       |                                       |
|-------|-----------------------|---------------------------------------|
| table | : a                   | # drug_era table                      |
| type  | : ref                 | # use index (equivalent value search) |
| key   | : idx_drug_concept_id | # index used                          |
| ref   | : rwd2.b.id           |                                       |

# Join Operator

- Example: Relations R (A, B) and S (B, C)  
SELECT \* FROM R, S WHERE R.B=S.B
- Suppose
  - R is P\_DRUG and S is DRUG\_ERA
  - R.B=ID, S.B=DRUG\_CONCEPT\_ID

R

| A  | B |
|----|---|
| A1 | 0 |
| A2 | 1 |
| A3 | 2 |
| A4 | 1 |

S

| B | C  |
|---|----|
| 1 | C1 |
| 2 | C2 |
| 1 | C3 |
| 3 | C4 |
| 1 | C5 |



$R \bowtie S$

| A  | B | C  |
|----|---|----|
| A2 | 1 | C1 |
| A2 | 1 | C3 |
| A2 | 1 | C5 |
| A3 | 2 | C2 |
| A4 | 1 | C1 |
| A4 | 1 | C3 |
| A4 | 1 | C5 |

see

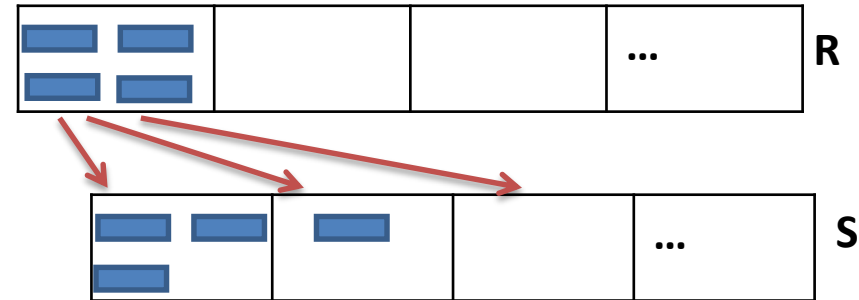
[https://www.informatik.hu-berlin.de/de/forschung/gebiete/wbi/teaching/archive/sose05/dbs2/slides/09\\_joins.pdf](https://www.informatik.hu-berlin.de/de/forschung/gebiete/wbi/teaching/archive/sose05/dbs2/slides/09_joins.pdf)

# Nested-loop Join

- Cost estimations

- Super-naïve

- $n(R)$ ,  $n(S)$  number of records in R and in S
    - $IO = n(R) * n(S)$



- Slight improvement

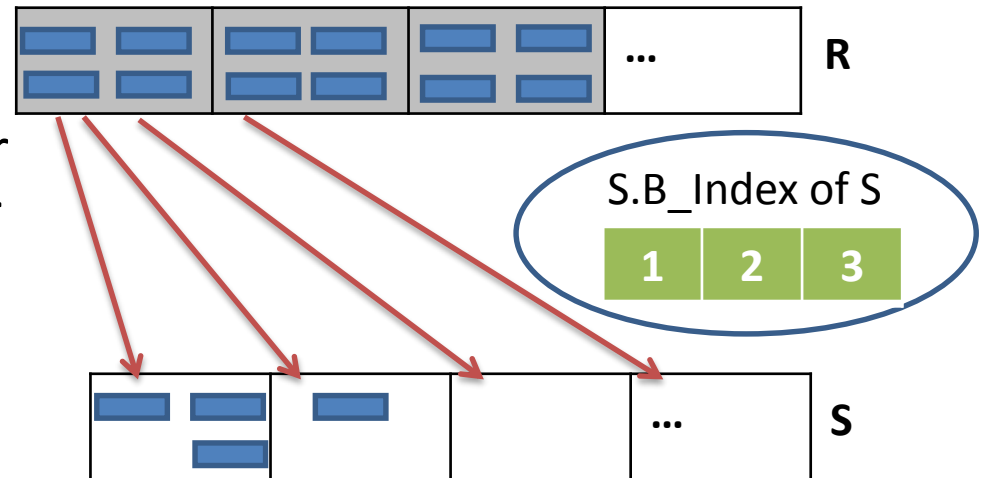
- R is outer relation, S is inner relation
    - Each block of outer relation is read once
    - Inner relation is read once for each block of outer relation
    - $B(R)$ ,  $b(S)$  number of blocks in R and in S
    - $IO = b(R) + b(R) * b(S)$
    - Use smaller relation as outer relation
      - For large relation, choice doesn't really matter

see

[https://www.informatik.hu-berlin.de/de/forschung/gebiete/wbi/teaching/archive/sose05/dbs2/slides/09\\_joins.pdf](https://www.informatik.hu-berlin.de/de/forschung/gebiete/wbi/teaching/archive/sose05/dbs2/slides/09_joins.pdf)

# Blocked Nested-loop Join(MySQL)

- Rule of thumb: Use all memory you can get
  - Use all memory the buffer manager allocates to your process (but this might be a difficult decision).
- Cost estimation
  - Outer relation is read once
  - Inner relation is read once for every chunk of R
  - There are  $\sim b(R)/m$  chunks
  - $IO = b(R) + b(R)*b(S)/m$
- Our case (Index Join)
  - Inner table is indexed, actually, one block-nested loop with index access
  - $b(R)*b(S)/m$  needs to be modified according to our assumptions, e.g. the number of different values of attribute B, B-tree search cost, etc.



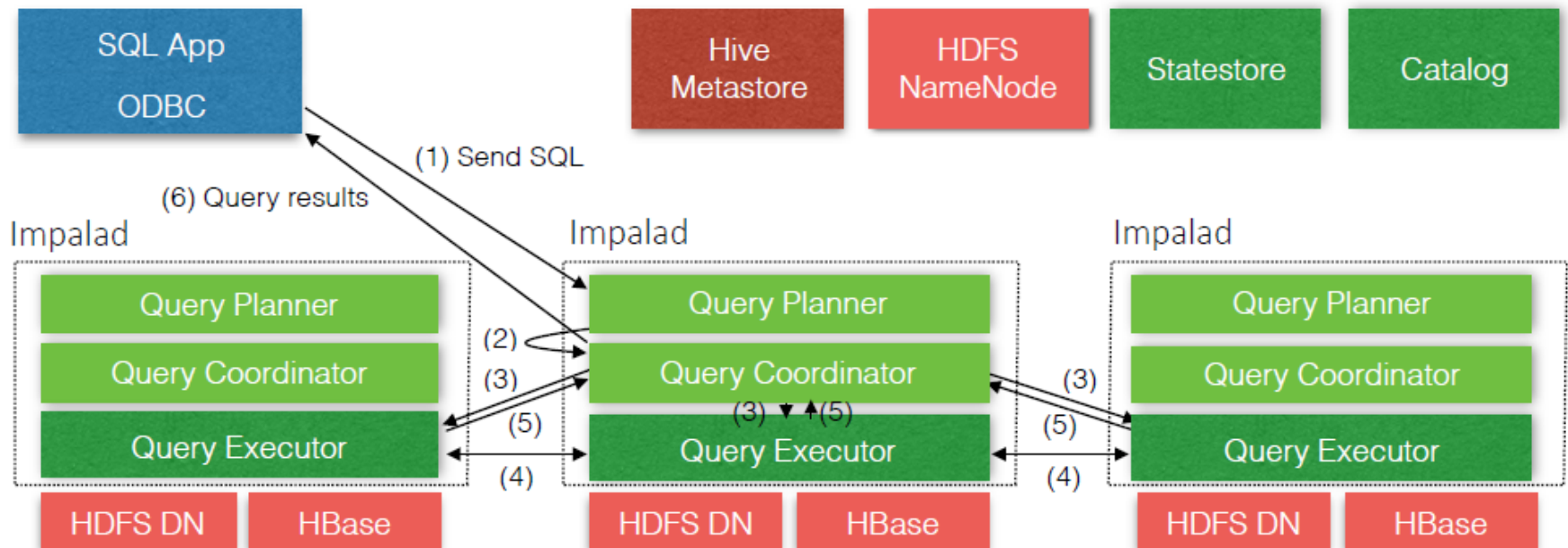


# Comments on MySQL Performance

---

- **MySQL optimizer is smart but not perfect.** You need to profile slow SQLs with EXPLAIN statement and optimizer hints then improve them.
- JOIN is potentially very expensive but required in all practical queries, so **we need understand the pros and cons of various JOINS.**
- HDD cache and MySQL buffer pools are usually effective.
- The query cache is deprecated as of MySQL 5.7.20, and is removed in MySQL 8.0.

# Architecture of Impala



from [http://cidrdb.org/cidr2015/Papers/CIDR15\\_Paper28.pdf](http://cidrdb.org/cidr2015/Papers/CIDR15_Paper28.pdf)

# Impala Query Plan

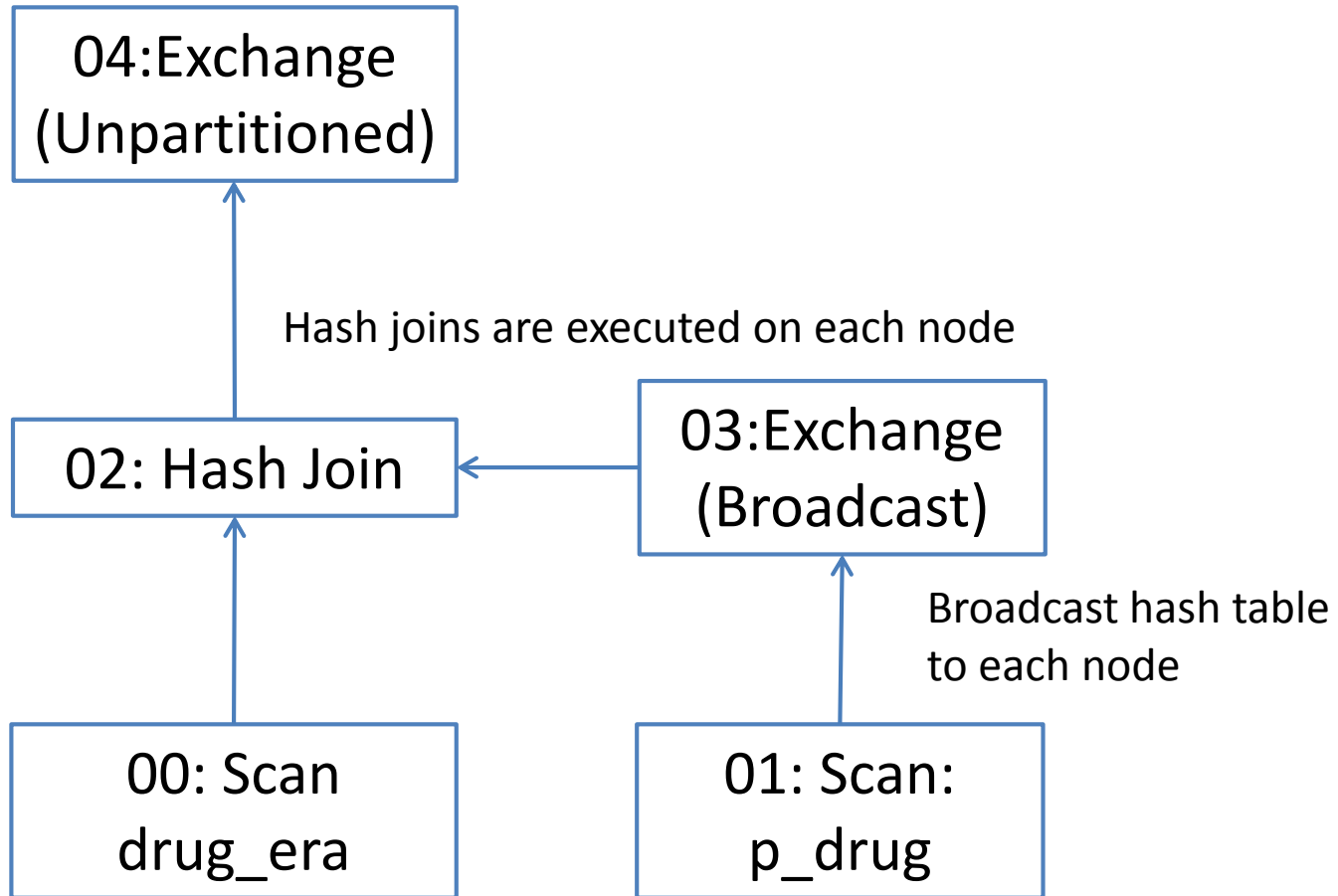
EXPLAIN

```
SELECT a.person_id, a.drug_concept_id, drug_era_start_date, b.vocabulary  
FROM drug_era a INNER JOIN p_drug b ON a.drug_concept_id=b.id ;
```

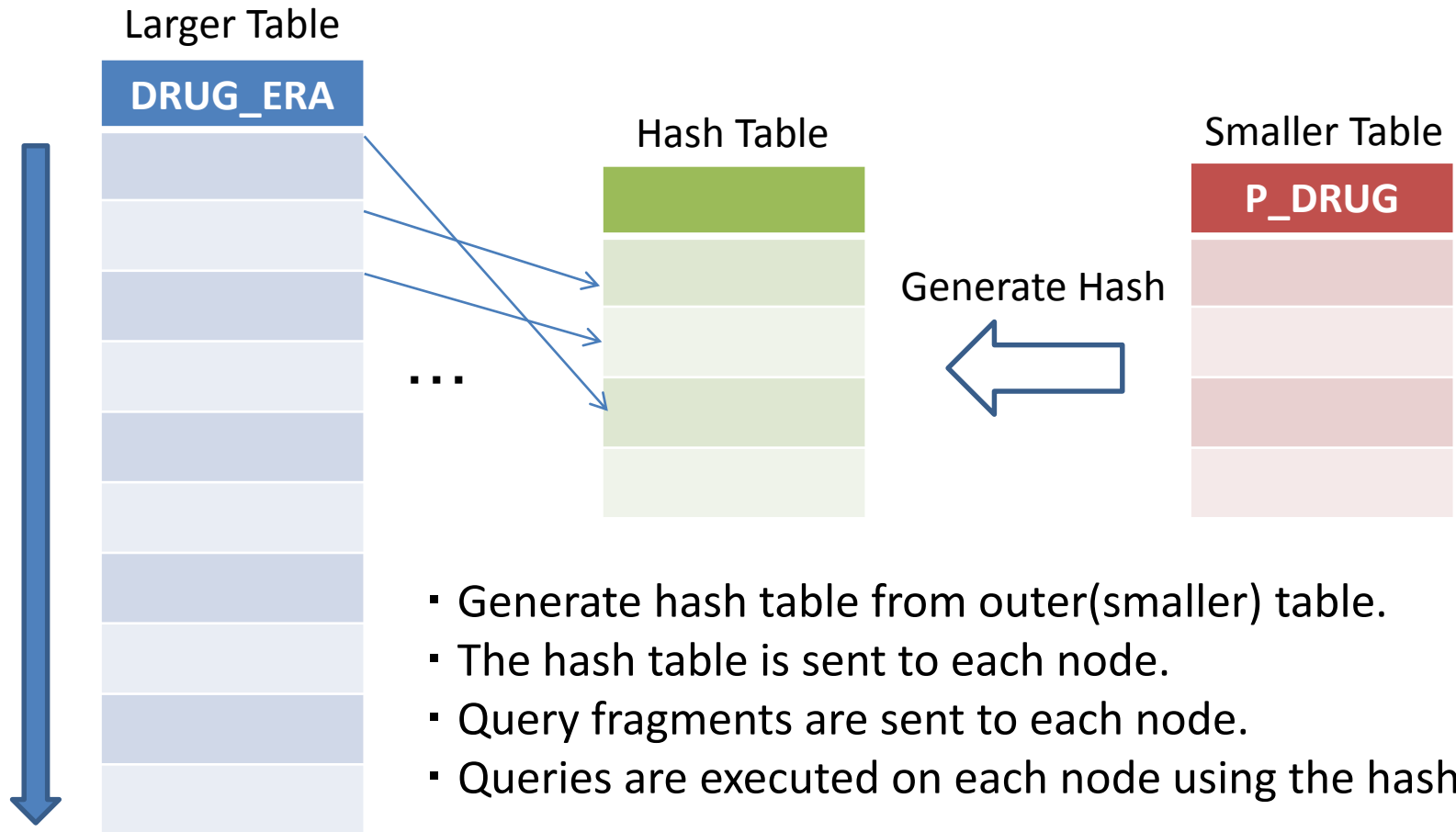
```
04:EXCHANGE [UNPARTITIONED]  
|  
"02:HASH JOIN [INNER JOIN, BROADCAST]"  
| hash predicates: a.drug_concept_id = b.id  
| runtime filters: RF000 <- b.id  
|  
|--03:EXCHANGE [BROADCAST]  
| |  
| 01:SCAN HDFS [default.p_drug2 b]  
|   partitions=1/1 files=1 size=1.63MB  
|  
00:SCAN HDFS [default.drug_era a]  
partitions=1/1 files=1 size=7.31GB  
runtime filters: RF000 -> a.drug_concept_id
```

# Impala Query Plan (cont.)

---



# Hash Join



see

[https://www.informatik.hu-berlin.de/de/forschung/gebiete/wbi/teaching/archive/sose05/dbs2/slides/09\\_joins.pdf](https://www.informatik.hu-berlin.de/de/forschung/gebiete/wbi/teaching/archive/sose05/dbs2/slides/09_joins.pdf)

# Why is Impala Fast?

---

- Massively parallel SQL engine
  - All impala daemons are *symmetric*.
  - Performance scalability is approximately proportional to the number of nodes (clusters).
- Runtime code generation using *LLVM*
  - LLVM is a modular and reusable compiler library and collection of related tools.
- I/O management
  - Using HDFS feature called *short-circuit local reads*.
- Hash Join
  - Not always fastest join algorithm. It depends on your objectives.

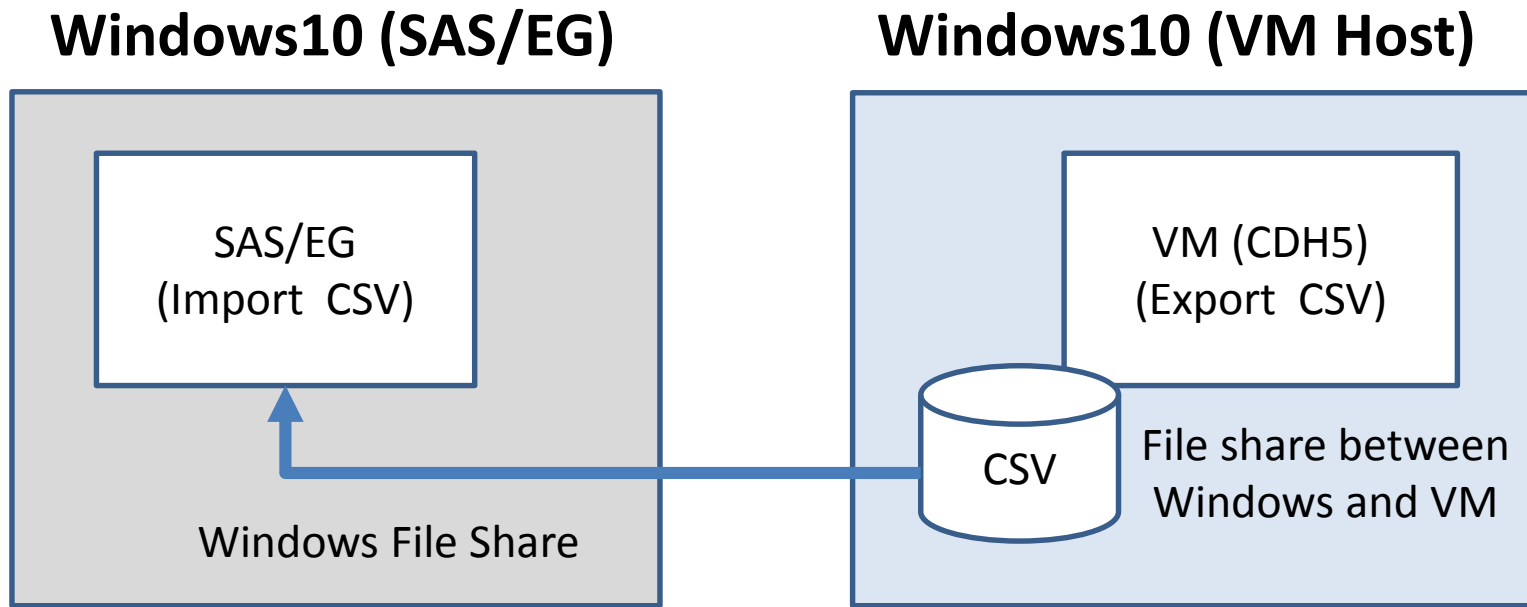
# Contents

---

- Backgrounds
- Objective
- Data used in this experiment
- A hypothetical research
- System configuration
- Measure the performance of SQL
- **Exporting data to SAS/EG**
- Limitations
- Conclusions
- Future Plan

# Exporting Data to SAS

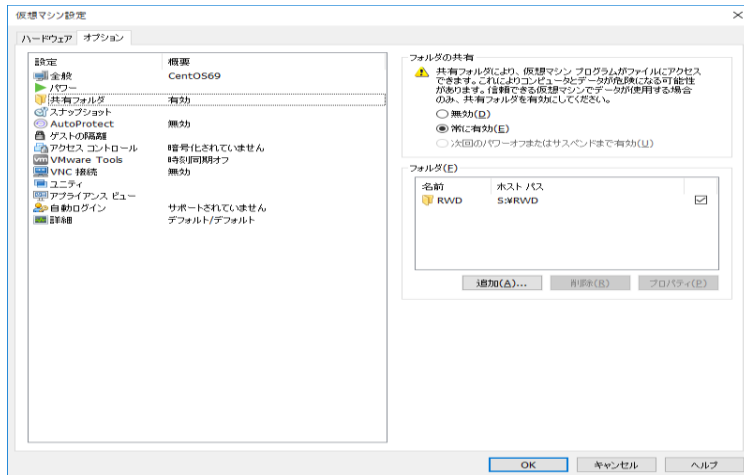
- Hue workbench supports exporting of SQL output in CSV/Excel format.
- *impala-shell* command is preferred for very large SQL output.
- Need SAS/ACCESS to {Hadoop, Impala}?





# Export Table in CSV Format

## Windows10 (VM Host)

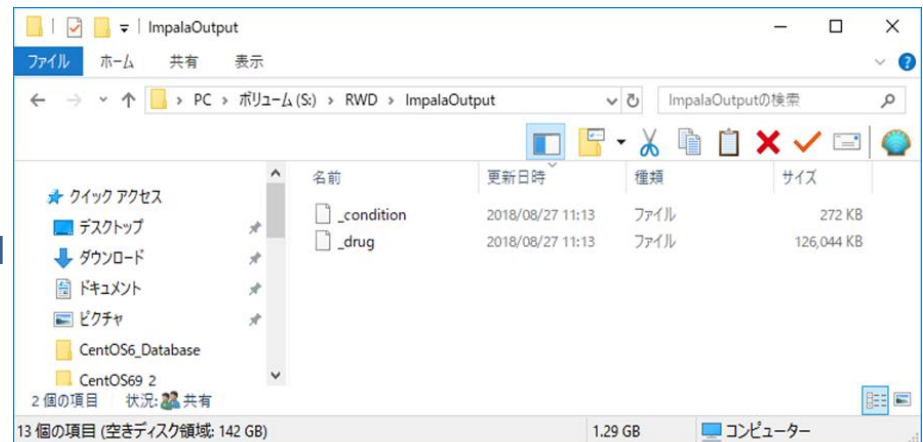


## Windows10 (VM Guest)

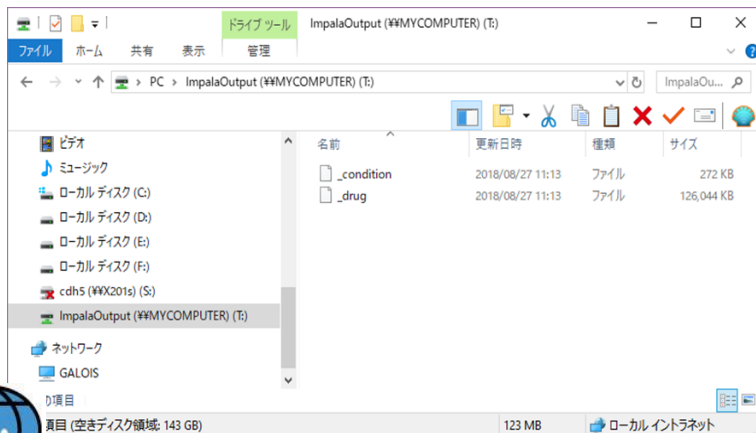
```
$ cd /mnt/hgfs/RWD/ImpalaOutput
```

```
$ impala-shell -i nd1.matsuya.org -B -o _drug --  
output_delimiter=';' -q "select * from _drug"
```

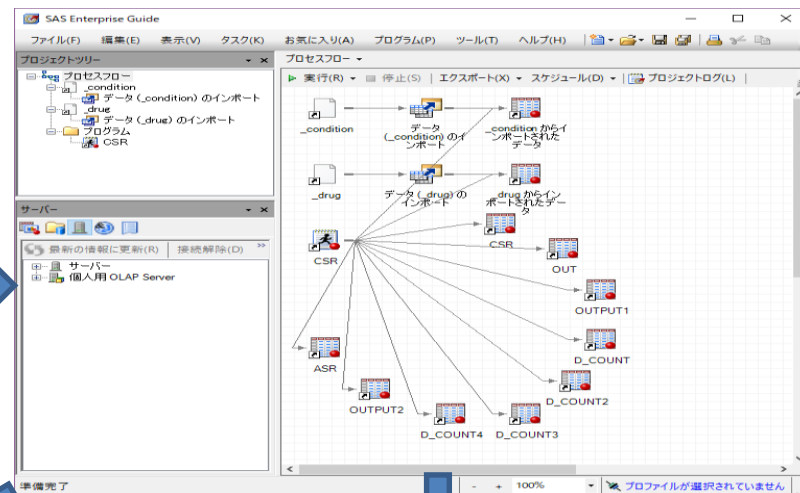
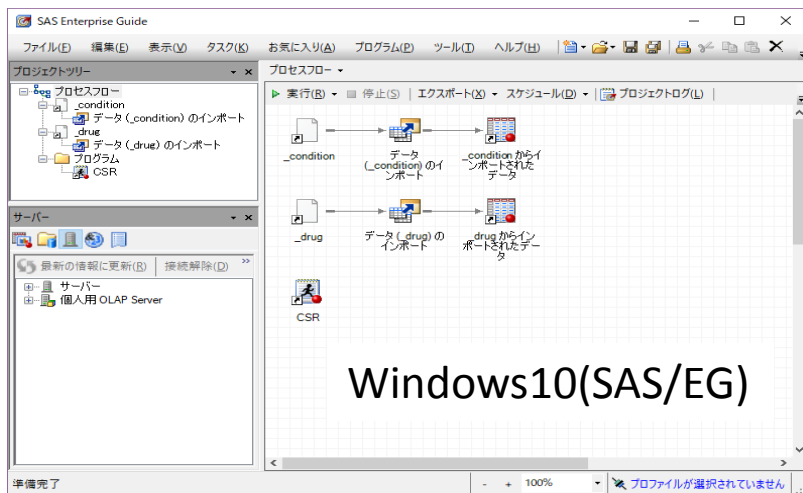
## Windows10 (VM Host)



## Windows10 (SAS/EG)



# Analysis using SAS/EG



SAS Enterprise Guide window showing a table of data. The Project Tree on the left includes a process flow with inputs for condition and drug data, leading to a CSR output. The main workspace shows a table with columns for CSR, NSR, ASR, ASR\_LCI, and ASR\_UCI.

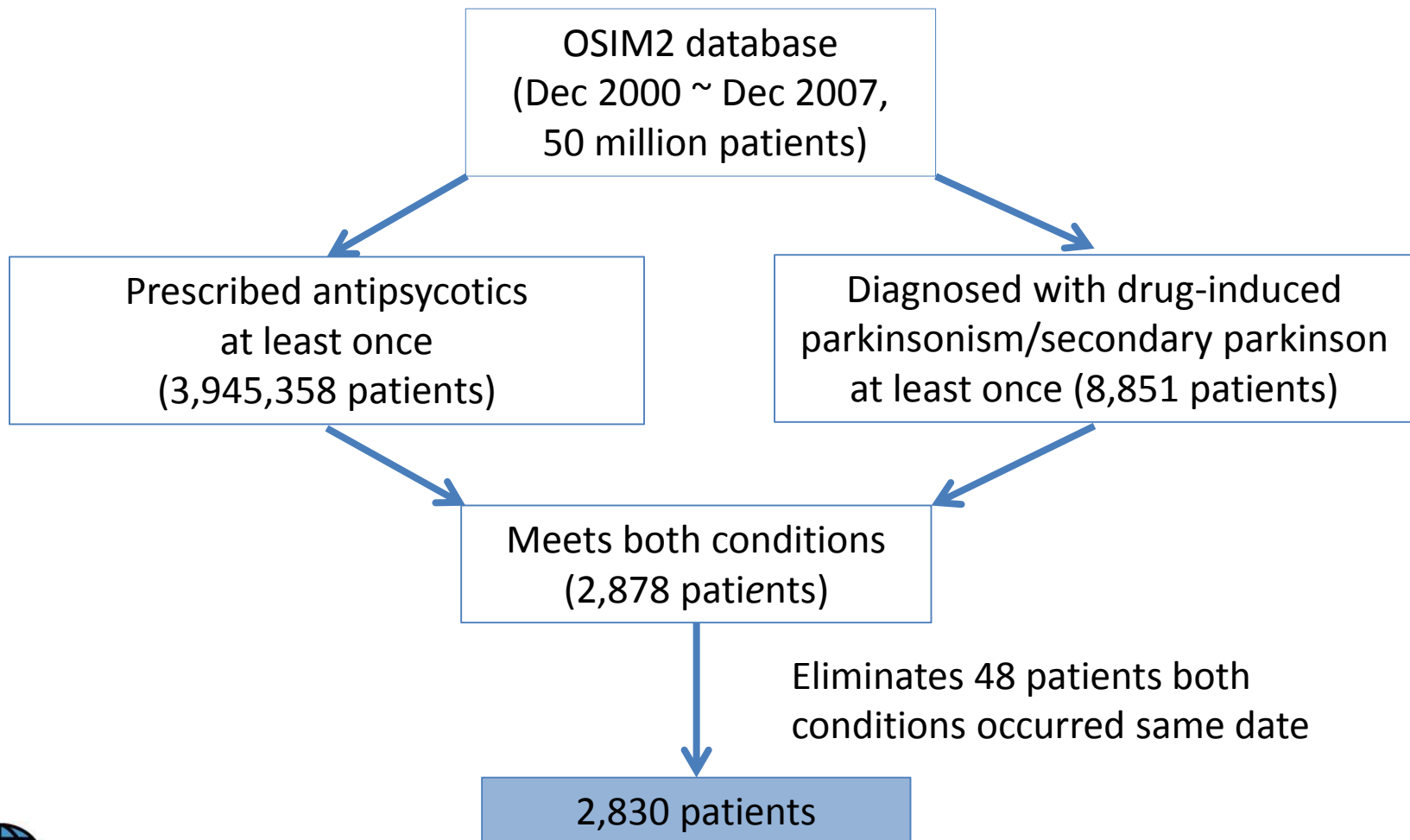
|   | CSR          | NSR          | ASR         | ASR_LCI      | ASR_UCI      |
|---|--------------|--------------|-------------|--------------|--------------|
| 1 | 1.0793534166 | 1.0774496988 | 1.001766874 | 0.9299191661 | 1.0792230677 |

SAS Enterprise Guide window showing a table of data. The Project Tree on the left includes a process flow with inputs for condition and drug data, leading to a CSR output. The main workspace shows a table with columns for FLG, COUNT, and PERCENT.

|   | FLG  | COUNT | PERCENT      |
|---|------|-------|--------------|
| 1 | C->D | 1361  | 47.289784573 |
| 2 | D->C | 1469  | 51.042390549 |
| 3 | UNK  | 48    | 1.6678248784 |

# Target Dataset for Hypothetical Research

---



# Contents

---

- Backgrounds
- Objective
- Data used in this experiment
- A hypothetical research
- System configuration
- Measure the performance of SQL
- Exporting data to SAS/EG
- **Limitations**
- Conclusions
- Future Plan

# Limitations

---

- Measuring performances were not executed in rigid way.
- MySQL, Impala, and Operating System were not optimized for this experiment.
- Multi-user setting was not considered.
- Security issues were not concerned.

# Contents

---

- Backgrounds
- Objective
- Data used in this experiment
- A hypothetical research
- System configuration
- Measure the performance of SQL
- Exporting data to SAS/EG
- Limitations
- **Conclusions**
- Future Plan

# Conclusions

---

- Apache Impala is significantly faster than MySQL on this experiments, especially join operation was performed on large tables.
- Hadoop ecosystem is high-maintenance.
  - Need a cluster management application like Cloudera Manager.
- Fast data loading is important to improve the turnaround time of ETL and analysis.

# Contents

---

- Backgrounds
- Objective
- Data used in this experiment
- A hypothetical research
- System configuration
- Measure the performance of SQL
- Exporting data to SAS/EG
- Limitations
- Conclusions
- **Future Plan**



# Standard Data Structures + $\alpha$



# Future...

---

- We hope to use it for future treatment strategies by looking at relationships of multiple diseases.
  - Cardio-Oncology/ Onco-Cardiology
  - Diabetes - ...
  - And more..

# BI tools

---

- Very powerful.
- However, in order to maximize performance, it is necessary to optimize database technologies as well as tools.

# 日本臨床試験学会第10回学術集会総会

- ホーム >
- ご挨拶 >
- 開催概要 >
- 組織委員 >
- 日程表 >
- プログラム >
- 事前参加登録 >
- 演題募集 >
- 参加者の皆様へ >
- 座長・演者の皆様へ >
- ポスター発表の皆様へ >
- 日本臨床薬理学会認定  
CRC制度について >
- 会場アクセス >
- 宿泊のご案内 >
- スポンサーに関するご案内 >
- リンク >

2028

2023

3926

2018

2019年

1月25<sup>(金)</sup>日・26<sup>(土)</sup>日

会期

会場 TFTホール (東京・有明)

会長 渡邊 裕司 (浜松医科大学 理事・副学長  
国立国際医療研究センター 臨床研究センター長)

2008

これまでの10年、

これからの10年、

## NEWS

- 2018.09.03 [スポンサーに関するご案内](#)を掲載しました。
- 2018.08.29 [組織委員](#)を掲載しました。
- 2018.08.08 ホームページを開設しました。

# **Thank you for your attention.**

This work was supported by The Grant for National Center for Global Health and Medicine (29A-1004).