



# Using Jenkins

as a Self-Service Data Portal for Apache  
Hadoop+Hive Data Warehouse

by

Jack Jacinto

Manager, Software Engineering

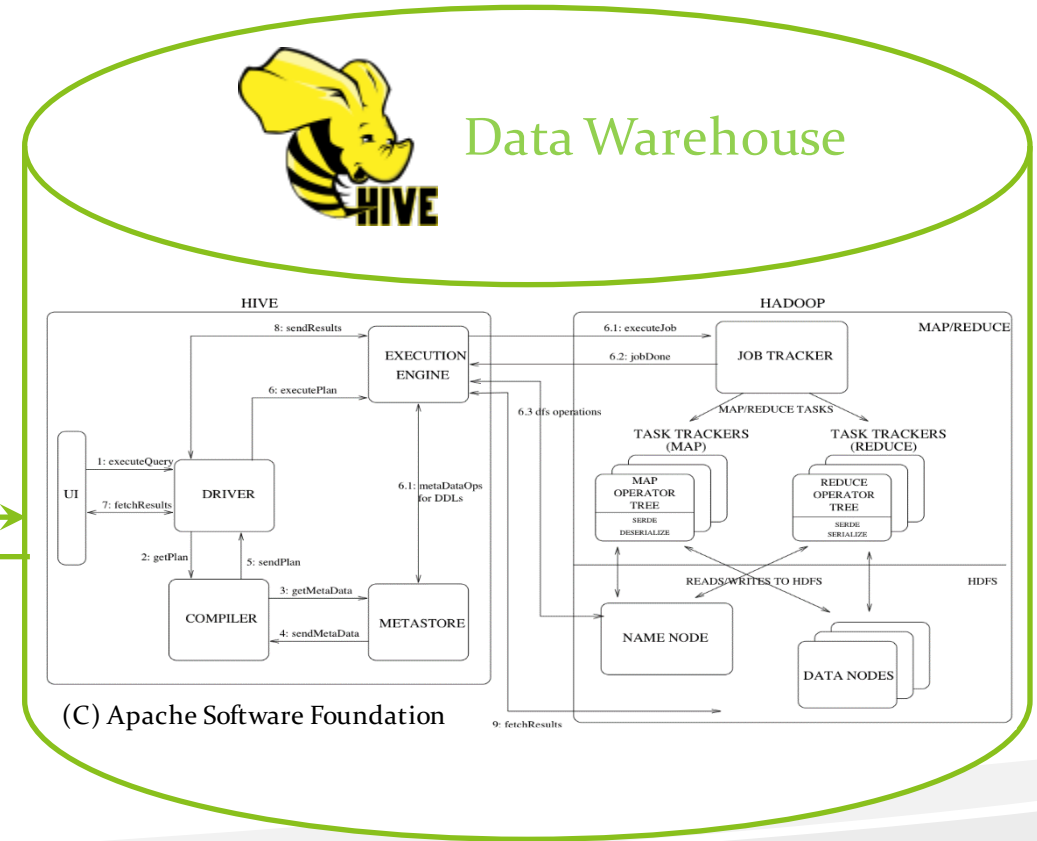
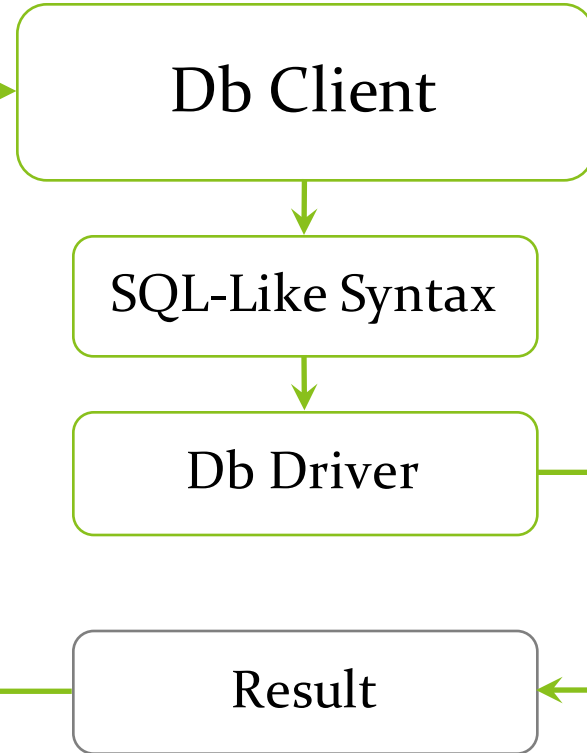
<https://www.linkedin.com/in/acjacinto>



# Objectives

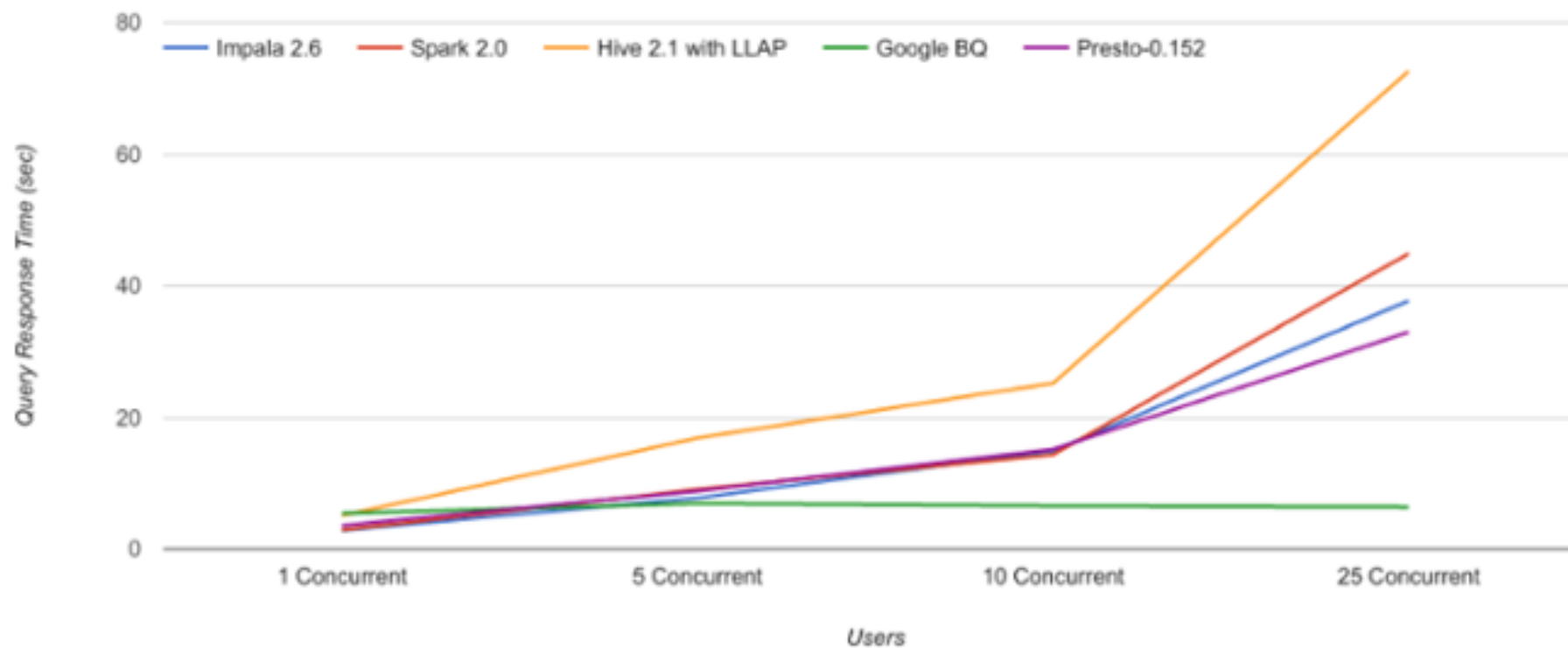
- Describe the thought process in finding an open source – based Self-Service Data Portal solution for Apache Hadoop+Hive data warehouse
- Provide configuration tips on how to repurpose Jenkins

# User Interaction with Apache Hadoop+Hive



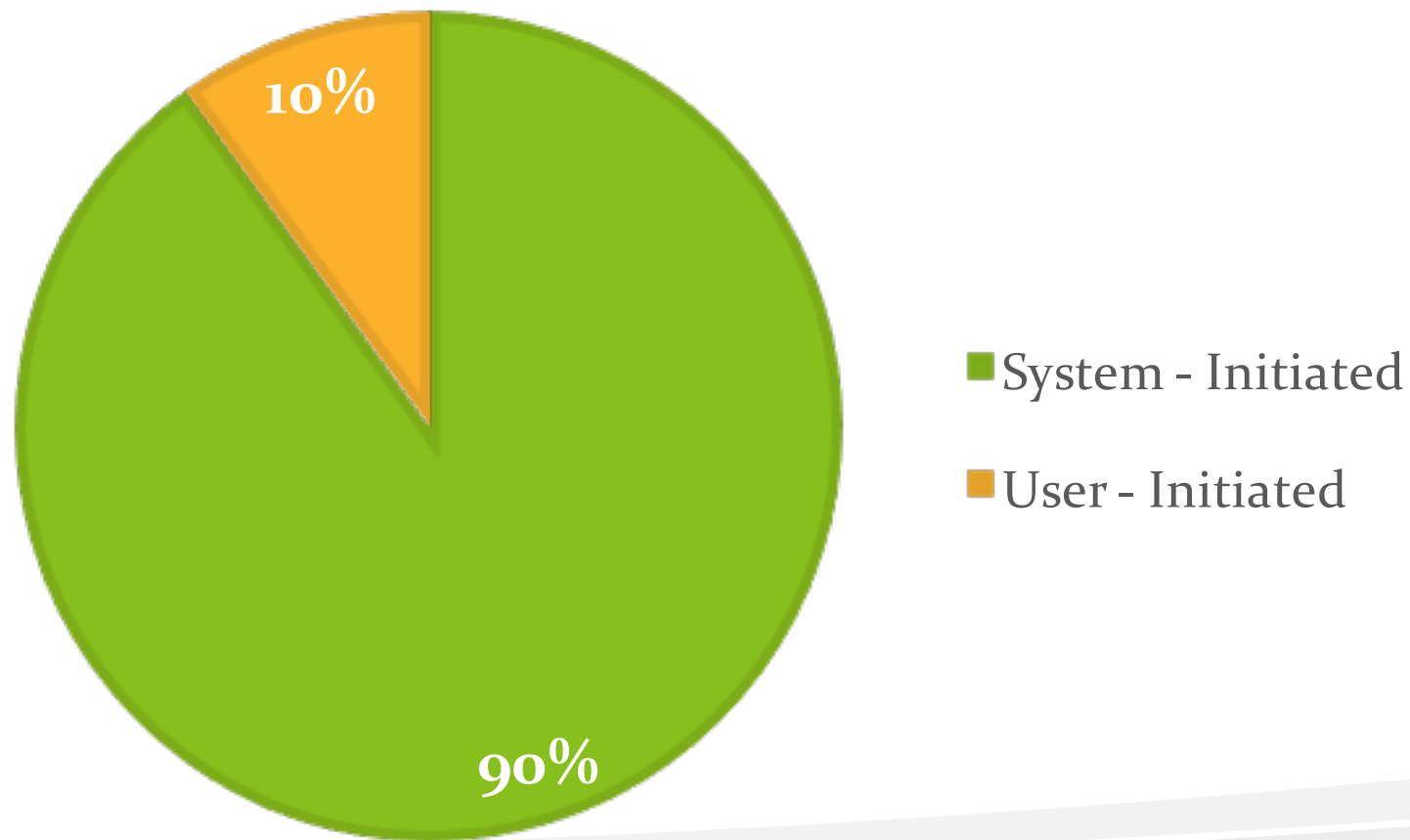


# Apache Hive Query Response Time



(c) atscale.com

# Sample Usage Distribution of an Apache Hive Instance



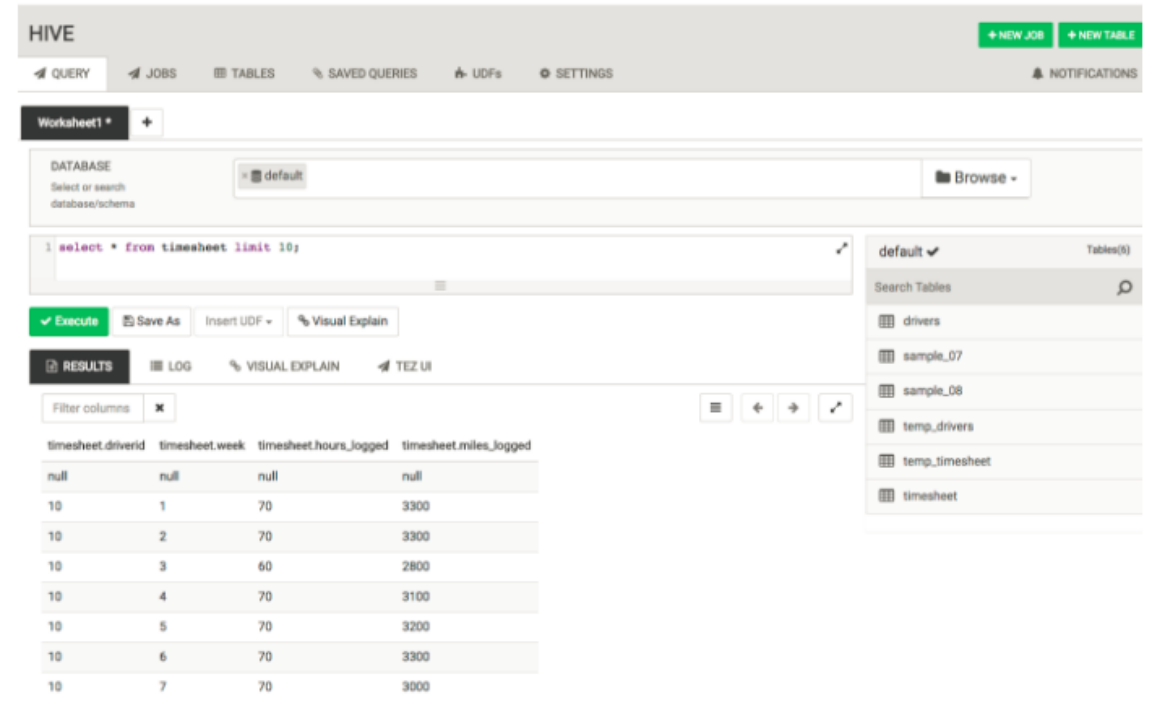


## What the users want...

- Store more data
- Autonomy in retrieving the data
- Retrieving the data in a fast manner

# Different ways to retrieve the data from Apache Hive, and their respective pros and cons

- via CLI
- via Desktop Client
- via Web Client
- ...
- via Ticketing System ?





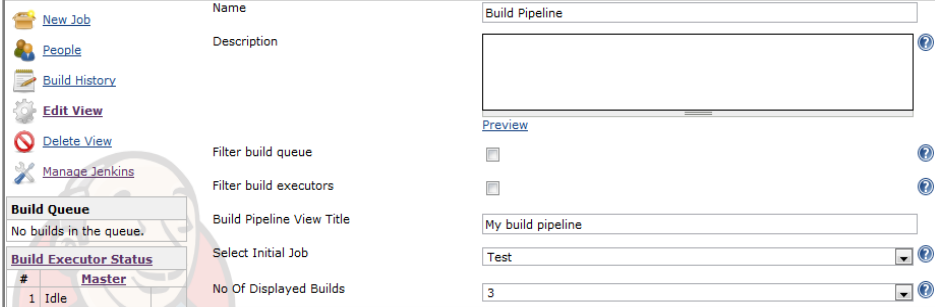
# There must be a better way that supports ...

- User-Friendly Web-based Forms
- Input Validation
- Asynchronous Submissions
- Request Queueing
- Request Throttling
- User Restriction
- Performance Tracking
- Email Notification
- Archiving
- Ease of Integration and Maintenance

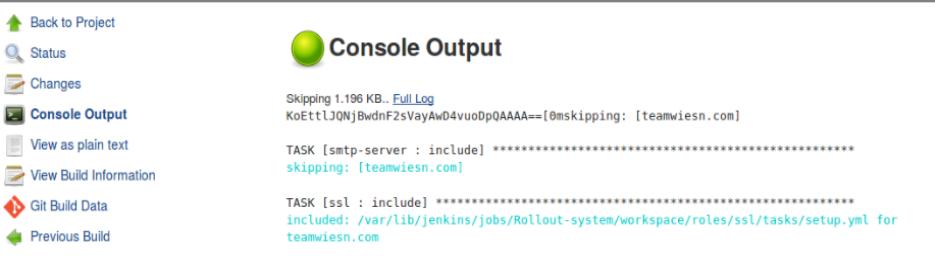


# What is Jenkins ?

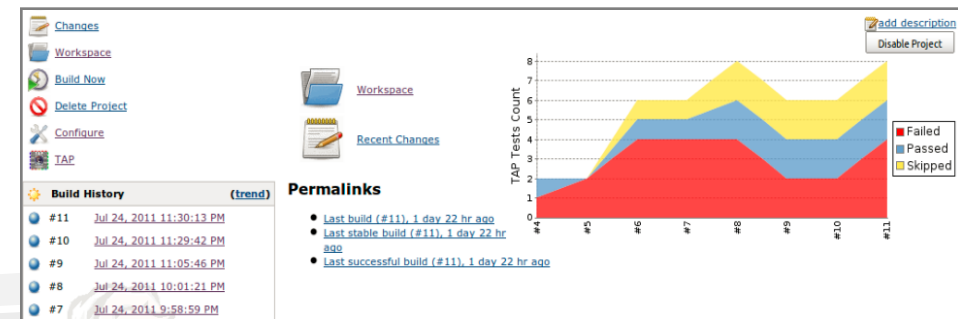
- The leading open source automation server
- Primarily intended for automating software builds, testing and deployment
- Visit <https://jenkins.io>



The image shows the 'New Job' configuration page in Jenkins. On the left, there is a sidebar with links: 'New Job', 'People', 'Build History', 'Edit View', 'Delete View', and 'Manage Jenkins'. Below these are sections for 'Build Queue' (showing 'No builds in the queue') and 'Build Executor Status' (showing '1 Idle Master'). The main area is for configuring a new job named 'Build Pipeline'. It includes a 'Description' field, a 'Preview' link, checkboxes for 'Filter build queue' and 'Filter build executors', a 'Build Pipeline View Title' field (set to 'My build pipeline'), a 'Select Initial Job' dropdown (set to 'Test'), and a 'No Of Displayed Builds' dropdown (set to '3').



The image shows the 'Console Output' view in Jenkins. On the left, there is a sidebar with links: 'Back to Project', 'Status', 'Changes', 'Console Output', 'View as plain text', 'View Build Information', 'Git Build Data', and 'Previous Build'. The main area displays the console output for a build. It starts with 'Skipping 1.196 KB.. Full Log' and 'KoEttLJONjBwdnF2sVayAw04vu0p0AAAA==[0mskipping: [teamwiesn.com]]'. Below this, there are two task blocks: 'TASK [smtp-server : include] \*\*\*\*\*' and 'TASK [ssl : include] \*\*\*\*\*'. The output for the 'ssl' task shows 'included: /var/lib/jenkins/jobs/Rollout-system/workspace/roles/ssl/tasks/setup.yml for teamwiesn.com'.





## Jenkins supports ...

- 👍 User-Friendly Web-based Forms
- 👍 Input Validation
- 👍 Asynchronous Submissions
- 👍 Request Queueing
- 👍 Request Throttling
- 👍 User Restriction
- 👍 Performance Tracking
- 👍 Email Notification
- 👍 Archiving
- 👍 Ease of Integration and Maintenance

# How can Jenkins be integrated with Apache Hive ?



 Jenkins

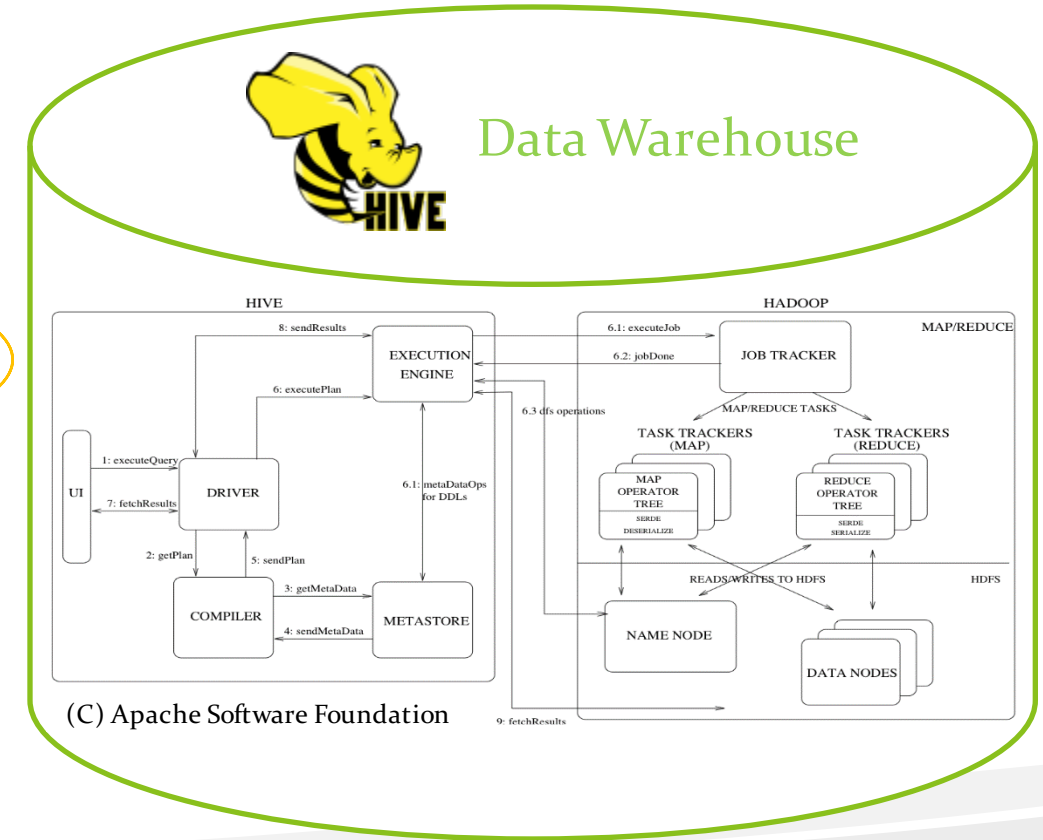
Java-based Client  
(Executable)

Generate Report  
(CSV)

Notify User

Download Report

Report (CSV)





## Code snippet of the Java-based client

```
// export
this.exportCSV(this.executeHQL(hql), file);

// output
Utils.throwMessage("Download the report at " + url + file);
```



## Code snippet of the Java-based client

```
protected ResultSet executeHQL(String query, Map<String, String> configs) {  
    try {  
        Configuration conf = new Configuration();  
        conf.set("hadoop.security.authentication", "Kerberos");  
        UserGroupInformation.setConfiguration(conf);  
        UserGroupInformation.loginUserFromKeytab(DB_USERNAME, DB_PASSWORD);  
  
        HiveJDBCClient client = new HiveJDBCClient(DB_URL + getHiveConfigs(conf));  
        Statement stmt = client.createStatement();  
        return stmt.executeQuery(query);  
    } catch (Exception e) {  
        System.err.println("Unexpected error");  
        e.printStackTrace();  
    }  
  
    return null;  
}
```



## Code snippet of the Java-based client

```
protected void exportCSV(ResultSet resultSet, String output) {
    FileWriter fileWriter = null;
    CSVPrinter csvFilePrinter = null;
    fileWriter = new FileWriter(output);
    csvFilePrinter = new CSVPrinter(fileWriter, CSVFormat.DEFA

    ResultSetMetaData metadata = resultSet.getMetaData();
    int count = metadata.getColumnCount();

    // Header
    Object[] header = new Object[count];
    for (int i = 0; i < count; i++) {
        header[i] = metadata.getColumnName(i + 1);
    }
    csvFilePrinter.printRecord(header);

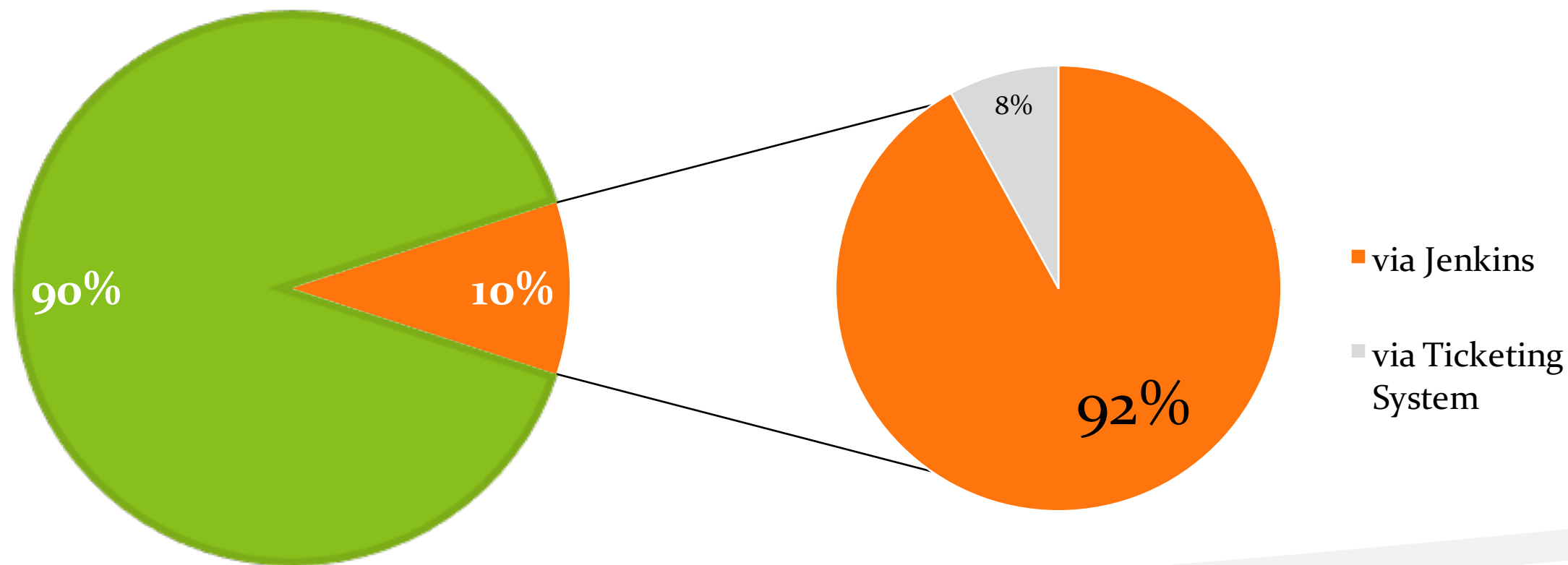
    // Rows
    while (resultSet.next()) {
        Object[] row = new Object[count];
        for (int i = 0; i < count; i++) {
            row[i] = resultSet.getObject(i + 1);
        }
        csvFilePrinter.printRecord(row);
    }
}
```



# Demo

\_(to be uploaded....)

# What Happened After Using Jenkins as a Data Portal?



Current Usage Distribution of the Apache Hive Instance





## What else happened?

- The team received an executive recognition for Automation early this year, with just 3 months of operations.
- The use of Jenkins for Self-Service Data Portal is being expanded
- The solution pattern is now being reused for different databases and data warehouses



# Q&A, and Thank You!

Special Thanks to Marie Grace Jacinto and Chuong Phan

Contact Info: <https://www.linkedin.com/in/acjacinto>