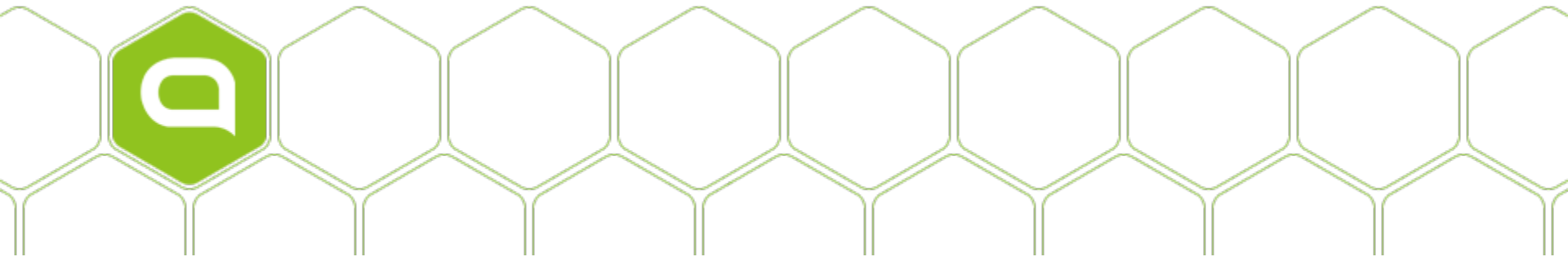


Introduction to Apache Tajo: Data Warehouse for Big Data

Jihoon Son / Gruter inc.



About Me



- Jihoon Son (@jihoonson)
 - Tajo project co-founder
 - Committer and PMC member of Apache Tajo
 - Research engineer at Gruter

Outline



- About Tajo
- Features of the Recent Release
- Demo
- Roadmap

What is Tajo?



- Tajo / tá:zo / 타조
 - An ostrich in Korean
 - The world's fastest two-legged animal

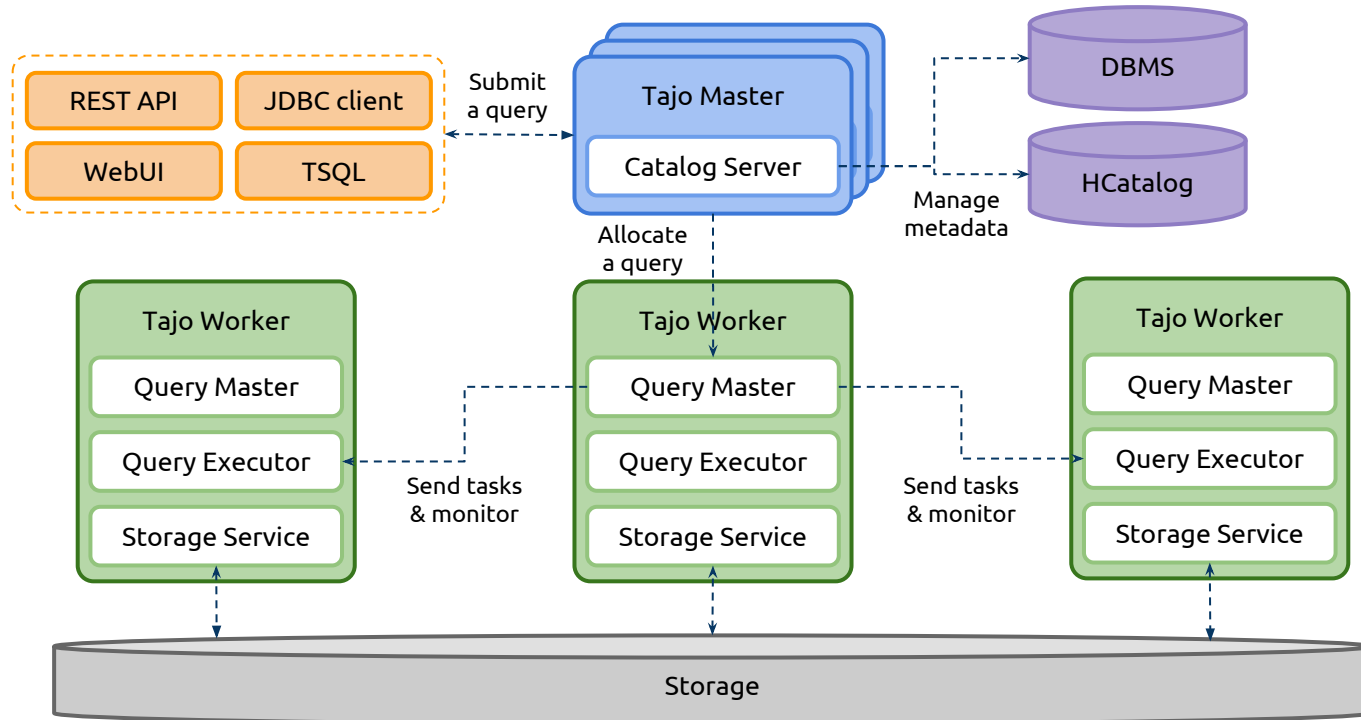


What is Tajo?



- Apache Top-level Project
 - Big **data warehouse** system
 - ANSI-SQL compliant
 - Mature SQL features
 - Various types of join, window functions
 - **Rapid query execution** with own distributed DAG engine
 - Low latency, and long running batch queries with a single system
 - Fault-tolerance
 - **Beyond SQL-on-Hadoop**
 - Support various types of storage

Architecture Overview



Who are Using Tajo?



- Use cases: replacement of commercial DW
 - 1st telco in South Korea
 - Replacement of long-running ETL workloads on several TB datasets
 - Lots of daily reports about user behavior
 - Ad-hoc analysis on TB datasets
 - Benefits
 - Simplified architecture for data analysis
 - An unified system for DW ETL, OLAP, and Hadoop ETL
 - Much less cost, more data analysis within same SLA
 - Saved license fee of commercial DW

Who are Using Tajo?



- Use cases: data discovery
 - Music streaming service (26 million users)
 - Analysis of purchase history for target marketing
 - Benefits
 - Interactive query on large datasets
 - Data analysis with familiar BI tools

Recent Release: 0.11



- Feature highlights
 - Query federation
 - JDBC-based storage support
 - Self-describing data formats support
 - Multi-query support
 - More stable and efficient join execution
 - Index support
 - Python UDF/UDAF support

Recent Release: 0.11



- Today's topic
 - Query federation
 - JDBC-based storage support
 - Self-describing data formats support
 - Multi-query support
 - More stable and efficient join execution
 - Index support
 - Python UDF/UDAF support

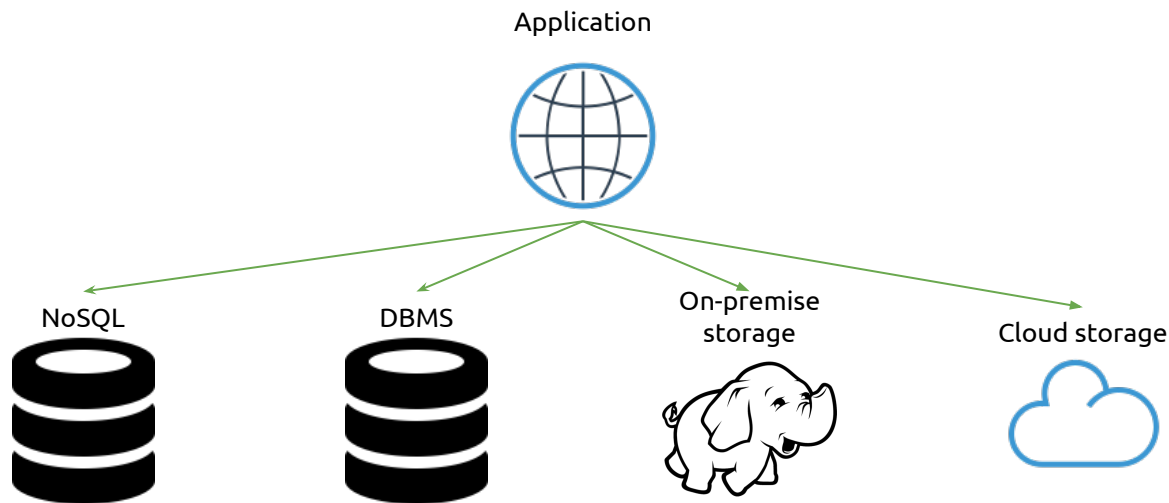
Query Federation with Tajo



Your Data



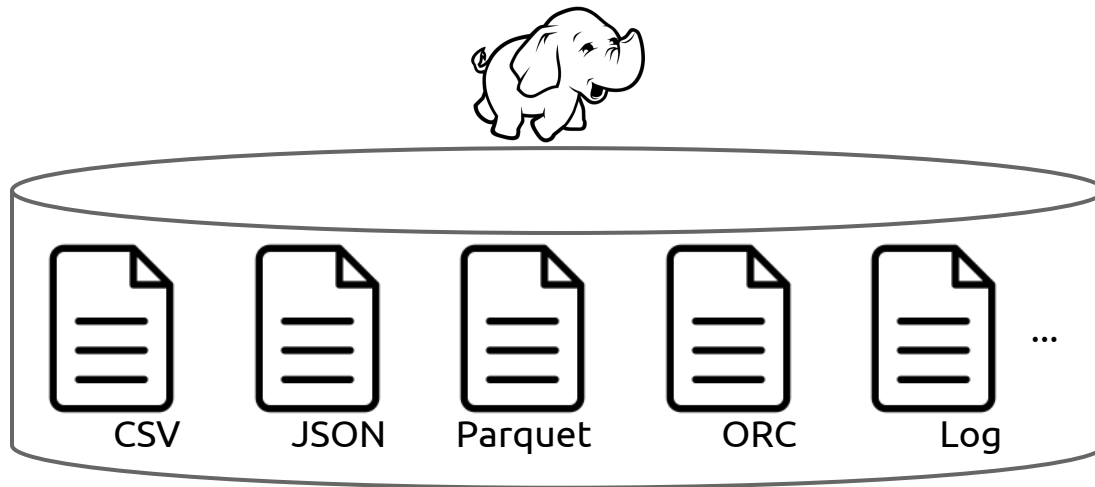
- Your data might be spread on multiple heterogeneous sites
 - Cloud, DBMS, Hadoop, NoSQL, ...



Your Data



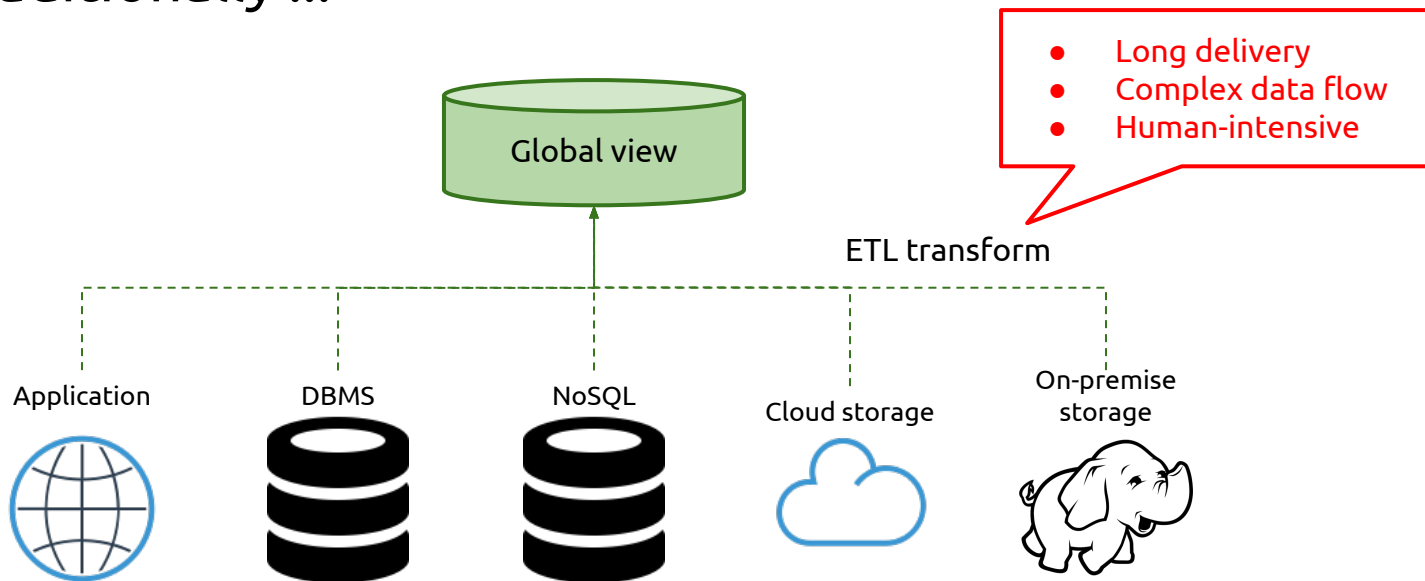
- Even in a single site, your data might be stored in different data formats



Your Data



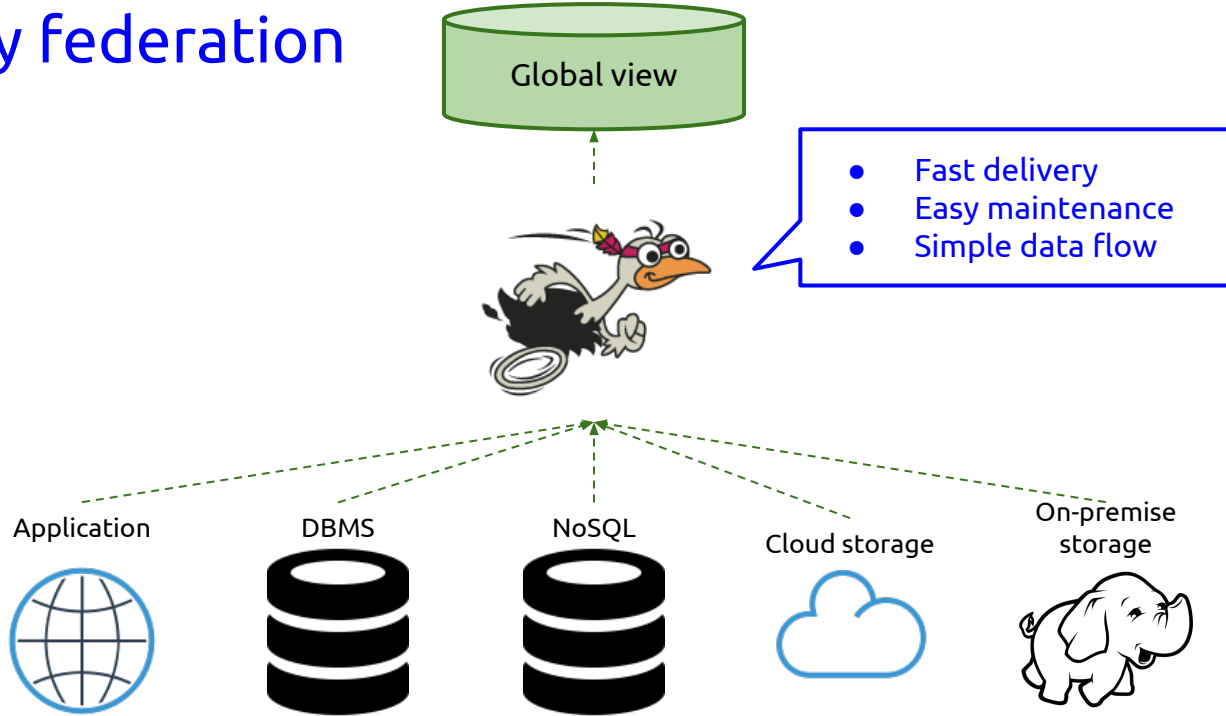
- How to analyze distributed data?
 - Traditionally ...



Your Data with Tajo



- Query federation



Storage and Data Format Support



Data
formats

{JSON}

Parquet



Sequence File

RCFile



Apache
orc™

Protocol Buffer

Storage
types



S3

APACHE
HBASE



elasticsearch.



Create Table



```
> CREATE EXTERNAL TABLE archive1 (id BIGINT, ...) USING text WITH  
( 'text.delimiter'='|' ) LOCATION 'hdfs://localhost:8020/archive1';
```

```
> CREATE EXTERNAL TABLE user (user_id BIGINT, ...) USING orc WITH  
( 'orc.compression.kind'='snappy' ) LOCATION 's3://user';
```

```
> CREATE EXTERNAL TABLE table1 (key TEXT, ...) USING hbase LOCATION  
'hbase:zk://localhost:2181/uptodate';
```

↑
Storage
URI

↑
Data
format

```
> ...
```

Create Table



```
> CREATE EXTERNAL TABLE archive1 (id BIGINT, ...) USING text WITH ('text.  
delimiter'='|','text.null'='\\N','compression.codec'='org.apache.hadoop.io.compress.  
SnappyCodec','timezone'='UTC+9','text.skip.headerlines'='2') LOCATION 'hdfs://localhost:  
8020/tajo/warehouse/archive1';
```

```
> CREATE EXTERNAL TABLE archive2 (id BIGINT, ...) USING text WITH ('text.  
delimiter'='|','text.null'='\\N','compression.codec'='org.apache.hadoop.io.compress.  
SnappyCodec','timezone'='UTC+9','text.skip.headerlines'='2') LOCATION 'hdfs://localhost:  
8020/tajo/warehouse/archive2';
```

```
> CREATE EXTERNAL TABLE archive3 (id BIGINT, ...) USING text WITH ('text.  
delimiter'='|','text.null'='\\N','compression.codec'='org.apache.hadoop.io.compress.  
SnappyCodec','timezone'='UTC+9','text.skip.headerlines'='2') LOCATION 'hdfs://localhost:  
8020/tajo/warehouse/archive3';
```

> ...

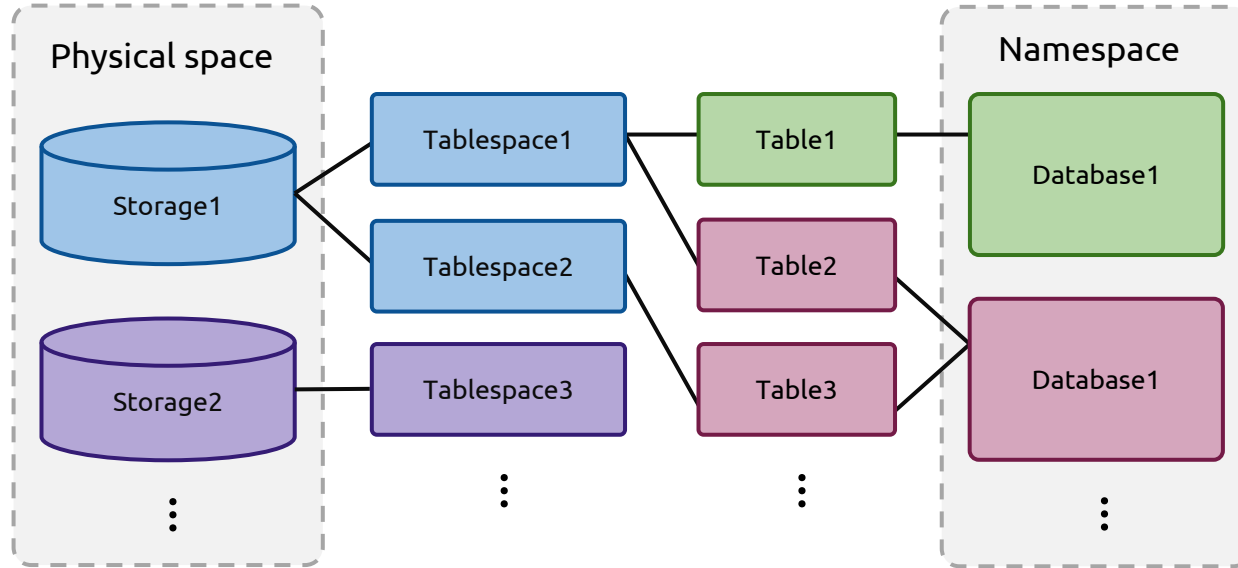
Too tedious!

Introduction to Tablespace



- Tablespace
 - Registered storage space
 - A tablespace is identified by an unique URI
 - Configurations and policies are shared by all tables in a tablespace
 - Storage type
 - Default data format and supported data formats
 - It allows users to reuse registered storage configurations and policies

Tablespaces, Databases, and Tables



Tablespace Configuration



```
{
  "spaces" : {
    "warehouse" : {
      "uri" : "hdfs://localhost:8020/tajo/warehouse",
      "configs" : [
        {'text.delimiter'='|'},
        {'text.null'='\\N'},
        {'compression.codec'='org.apache.hadoop.io.compress.SnappyCodec'},
        {'timezone'='UTC+9'},
        {'text.skip.headerlines'='2'}
      ]
    },
    "hbase1" : {
      "uri" : "hbase:zk://localhost:2181/table1"
    }
  }
}
```

Tablespace name

Tablespace URI

Create Table



> CREATE TABLE archive1 (id BIGINT, ...) TABLESPACE **warehouse**;

Tablespace
name

Data format is omitted. Default data format is **TEXT**.

```
"warehouse" : {  
  "uri" : "hdfs://localhost:8020/tajo/warehouse",  
  "configs" : [  
    {'text.delimiter'='|'},  
    {'text.null'='\\N'},  
    {'compression.codec'='org.apache.hadoop.io.compress.SnappyCodec'},  
    {'timezone'='UTC+9'},  
    {'text.skip.headerlines'='2'}  
  ]  
},
```

Create Table



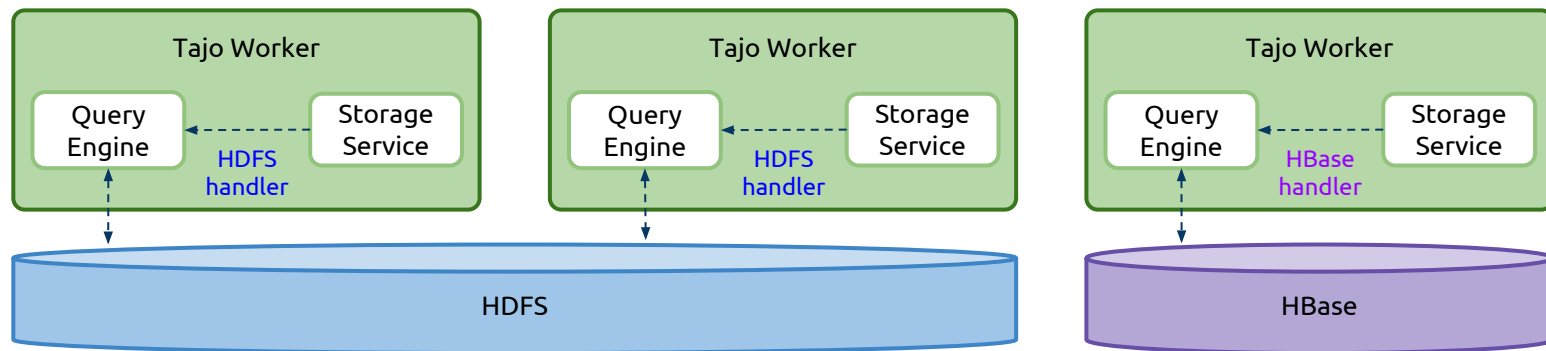
- > CREATE TABLE archive1 (id BIGINT, ...) TABLESPACE warehouse;
- > CREATE TABLE archive2 (id BIGINT, ...) TABLESPACE warehouse;
- > CREATE TABLE archive3 (id BIGINT, ...) TABLESPACE warehouse;
- > CREATE TABLE user (user_id BIGINT, ...) TABLESPACE aws USING orc
WITH ('orc.compression.kind'='snappy');
- > CREATE TABLE table1 (key TEXT, ...) TABLESPACE hbase1;
- > ...

Querying on Different Data Silos



- How does a worker access different data sources?
 - Storage service
 - Return a proper handler for underlying storage

> SELECT ... FROM hdfs_table, hbase_table, ...



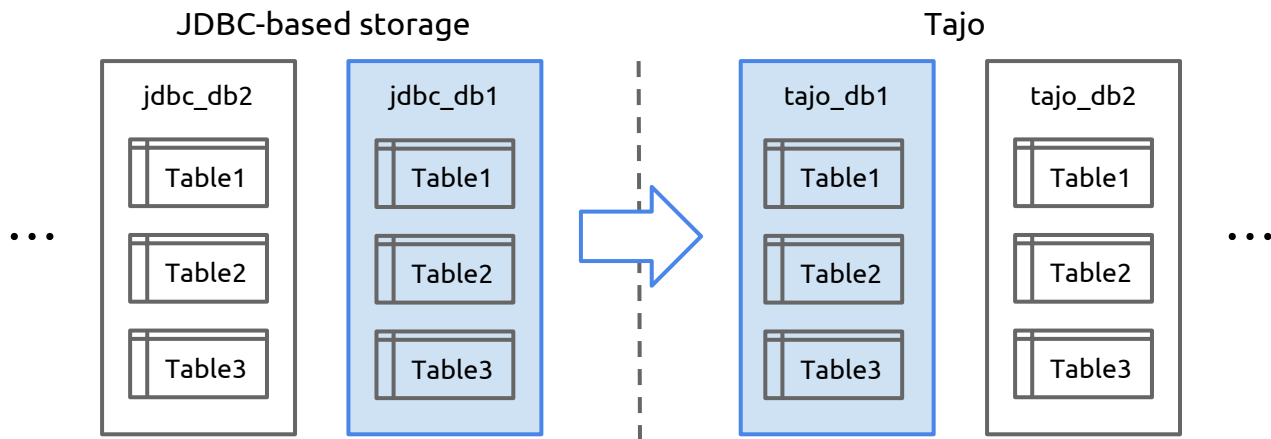
JDBC-based Storage Support



JDBC-based Storage



- Storage providing the JDBC interface
 - PostgreSQL, MySQL, MariaDB, ...
- Databases of JDBC-based storage are mapped to Tajo databases



Tablespace Configuration



```
{
  "spaces": {
    "pgsql_db1": {
      "uri": "jdbc:postgresql://hostname:port/db1"

      "configs": {
        "mapped_database": "tajo_db1"
        "connection_properties": {
          "user": "tajo",
          "password": "xxxx"
        }
      }
    }
  }
}
```

Tablespace name

PostgreSQL database name

Tajo database name

Return to Query Federation

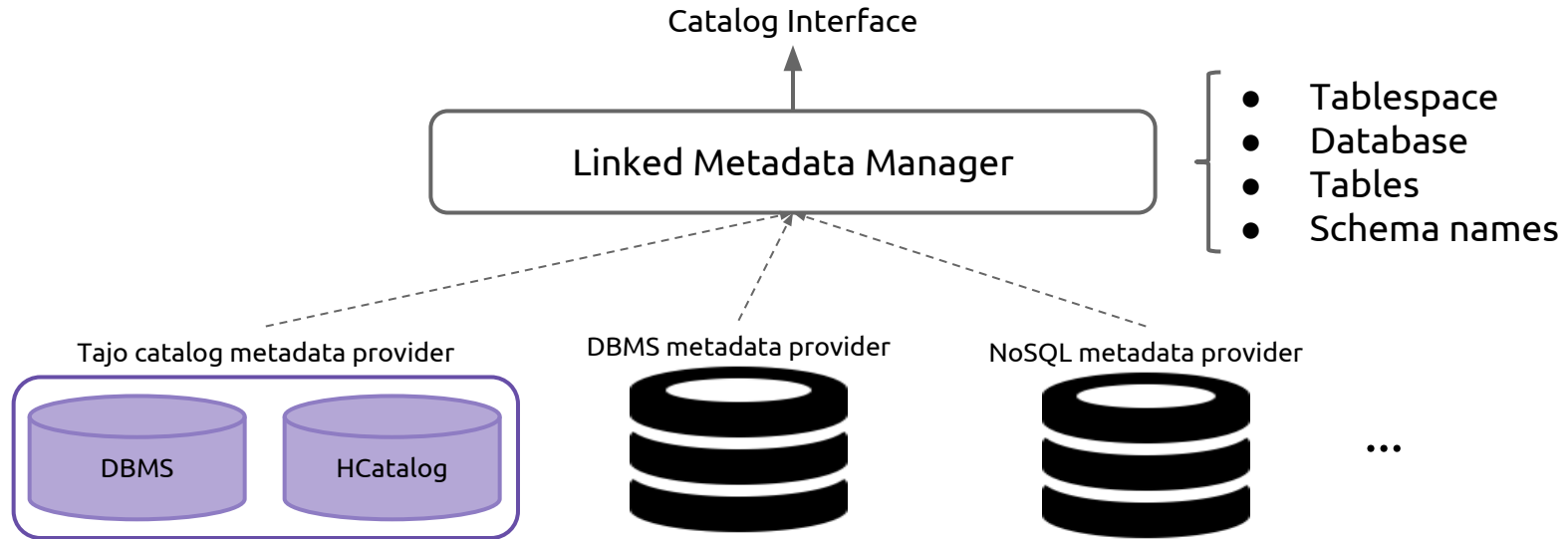


- How to correlate data on JDBC-based storage and others?
 - Need to have a global view of metadata across different storage types
 - Tajo also has its own metadata for its data
 - Each JDBC-based storage has own metadata for its data
 - Each NoSQL storage has metadata for its data
 - ...

Metadata Federation



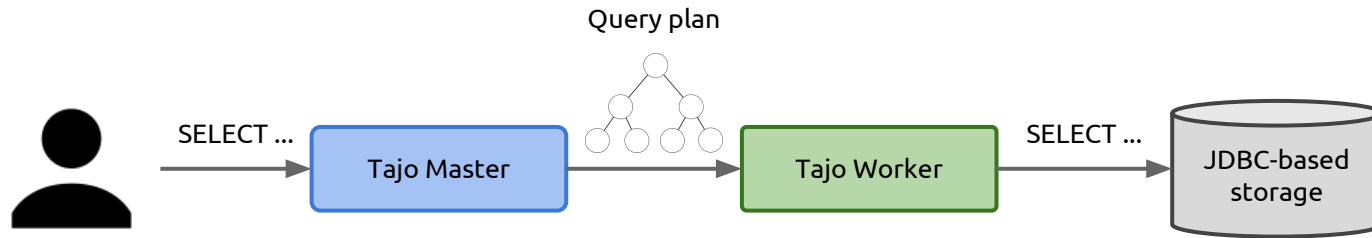
- Federating metadata of underlying storage



Querying on JDBC-based Storage



- A plan is converted into a SQL string
- Query generation
 - Diverse SQL syntax of different types of storage
 - Different SQL builder for each storage type



Operation Push Down

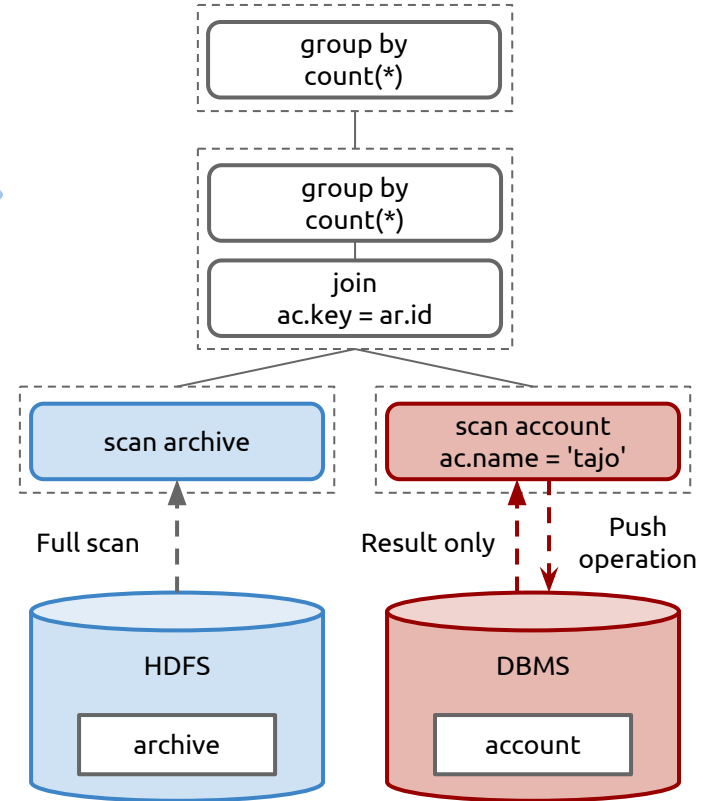
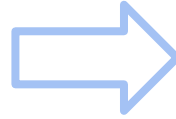


- Tajo can exploit the processing capability of underlying storage
 - DBMSs, MongoDB, HBase, ...
- Operations are pushed down into underlying storage
 - Leveraging the advanced features provided by underlying storage
 - Ex) DBMSs' query optimization, index, ...

Example 1



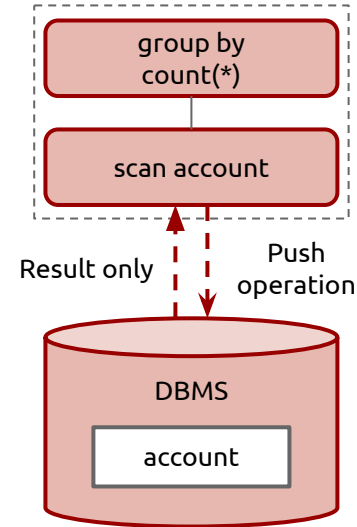
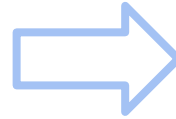
```
SELECT
  count(*)
FROM
  account ac, archive ar
WHERE
  ac.key = ar.id and
  ac.name = 'tajo'
```



Example 2



```
SELECT
  ac.name, count(*)
FROM
  account ac
GROUP BY
  ac.name
```



Self-describing Data Formats Support



Self-describing Data Formats



- Some data formats include schema information as well as data
 - JSON, ORC, Parquet, ...
- Tajo 0.11 natively supports self-describing data formats
 - Since they already have schema information, Tajo doesn't need to store it aside
 - Instead, Tajo can infer the schema at query execution time

Create Table with Nested Data Format



```
{ "title": "Hand of the King", "name": { "first_name": "Eddard", "last_name": "Stark" } }  
{ "title": "Assassin", "name": { "first_name": "Arya", "last_name": "Stark" } }  
{ "title": "Dancing Master", "name": { "first_name": "Syrio", "last_name": "Forel" } }
```

```
> CREATE EXTERNAL TABLE schemaful_table (  
  title TEXT,  
  name RECORD ( Nested type  
    first_name TEXT,  
    last_name TEXT  
  )  
) USING json LOCATION 'hdfs:///json_table';
```

How about This Data?



```
{"id":"2937257761","type":"ForkEvent","actor":{"id":"1088854","login":"CAOakleyl","gravatar_id":"","url":"https://api.github.com/users/CAOakleyl","avatar_url":"https://avatars.githubusercontent.com/u/1088854?","repo":{"id":"11909954","name":"skycocker/chromebrew","url":"https://api.github.com/repos/skycocker/chromebrew"},"payload":{"forkee":{"id":"38339291","name":"chromebrew","full_name":"CAOakleyl/chromebrew","owner":{"login":"CAOakleyl","id":"1088854","avatar_url":"https://avatars.githubusercontent.com/u/1088854?v=3","gravatar_id":"","url":"https://api.github.com/users/CAOakleyl","html_url":"https://github.com/CAOakleyl"},"followers_url":"https://api.github.com/users/CAOakleyl/followers","following_url":"https://api.github.com/users/CAOakleyl/following/other_user"},"gists_url":"https://api.github.com/users/CAOakleyl/gists/gist_id"},"starred_url":"https://api.github.com/users/CAOakleyl/starred/{owner}/{repo}"},"subscriptions_url":"https://api.github.com/users/CAOakleyl/subscriptions"},"organizations_url":"https://api.github.com/users/CAOakleyl/orgs"},"repos_url":"https://api.github.com/users/CAOakleyl/repos"},"events_url":"https://api.github.com/users/CAOakleyl/events/privacy"},"received_events_url":"https://api.github.com/users/CAOakleyl/received_events","type":"User","site_admin":false},"private":false,"html_url":"https://github.com/CAOakleyl/chromebrew","description":"Package manager for Chrome OS","fork":true,"url":"https://api.github.com/repos/CAOakleyl/chromebrew","forks_url":"https://api.github.com/repos/CAOakleyl/chromebrew/forks","keys_url":"https://api.github.com/repos/CAOakleyl/chromebrew/keys/key_id"},"collaborators_url":"https://api.github.com/repos/CAOakleyl/chromebrew/collaborators/collaborator"},"teams_url":"https://api.github.com/repos/CAOakleyl/chromebrew/teams","hooks_url":"https://api.github.com/repos/CAOakleyl/chromebrew/hooks","issue_events_url":"https://api.github.com/repos/CAOakleyl/chromebrew/issues/events/number"},"events_url":"https://api.github.com/repos/CAOakleyl/chromebrew/events","assignees_url":"https://api.github.com/repos/CAOakleyl/chromebrew/assignees/user"},"branches_url":"https://api.github.com/repos/CAOakleyl/chromebrew/branches/branch"},"tags_url":"https://api.github.com/repos/CAOakleyl/chromebrew/tags","blobs_url":"https://api.github.com/repos/CAOakleyl/chromebrew/git/blobs/sha"},"git_tags_url":"https://api.github.com/repos/CAOakleyl/chromebrew/git/tags/sha"},"git_refs_url":"https://api.github.com/repos/CAOakleyl/chromebrew/git/refs/sha"},"trees_url":"https://api.github.com/repos/CAOakleyl/chromebrew/git/trees/sha"},"statuses_url":"https://api.github.com/repos/CAOakleyl/chromebrew/statuses/sha"},"languages_url":"https://api.github.com/repos/CAOakleyl/chromebrew/languages","stargazers_url":"https://api.github.com/repos/CAOakleyl/chromebrew/stargazers"},"contributors_url":"https://api.github.com/repos/CAOakleyl/chromebrew/contributors"},"subscribers_url":"https://api.github.com/repos/CAOakleyl/chromebrew/subscribers"},"subscription_url":"https://api.github.com/repos/CAOakleyl/chromebrew/subscription"},"commits_url":"https://api.github.com/repos/CAOakleyl/chromebrew/commits/sha"},"git_commits_url":"https://api.github.com/repos/CAOakleyl/chromebrew/git/commits/sha"},"comments_url":"https://api.github.com/repos/CAOakleyl/chromebrew/comments/number"},"issue_comment_url":"https://api.github.com/repos/CAOakleyl/chromebrew/issues/comments/number"},"contents_url":"https://api.github.com/repos/CAOakleyl/chromebrew/contents/{+path"},"compare_url":"https://api.github.com/repos/CAOakleyl/chromebrew/compare/{base}...{head}"},"merges_url":"https://api.github.com/repos/CAOakleyl/chromebrew/merges","archive_url":"https://api.github.com/repos/CAOakleyl/chromebrew/{archive_format}/{ref}"},"downloads_url":"https://api.github.com/repos/CAOakleyl/chromebrew/downloads"},"issues_url":"https://api.github.com/repos/CAOakleyl/chromebrew/issues/number"},"pulls_url":"https://api.github.com/repos/CAOakleyl/chromebrew/pulls/number"},"milestones_url":"https://api.github.com/repos/CAOakleyl/chromebrew/milestones/number"},"notifications_url":"https://api.github.com/repos/CAOakleyl/chromebrew/notifications?since=all,participating"},"labels_url":"https://api.github.com/repos/CAOakleyl/chromebrew/labels/name"},"releases_url":"https://api.github.com/repos/CAOakleyl/chromebrew/releases/id"},"created_at":"2015-07-01T00:00:00Z","updated_at":"2015-06-28T10:11:09Z","pushed_at":"2015-06-09T07:46:57Z","git_url":"https://github.com/CAOakleyl/chromebrew.git","ssh_url":"git@github.com:CAOakleyl/chromebrew.git","clone_url":"https://github.com/CAOakleyl/chromebrew.git","svn_url":"https://github.com/CAOakleyl/chromebrew"},"homepage":"http://skycocker.github.io/chromebrew/","size":846,"stargazers_count":0,"watchers_count":0,"language":null,"has_issues":false,"has_downloads":true,"has_wiki":true,"has_pages":false,"forks_count":0,"mirror_url":null,"open_issues_count":0,"forks":0,"open_issues":0,"watchers":0,"default_branch":"master","public":true},"public":true,"created_at":"2015-07-01T00:00:01Z"}
```

...

Create Schemaless Table



```
> CREATE EXTERNAL TABLE schemaless_table (*) USING json LOCATION  
'hdfs:///json_table';
```

Allow any schema

That's all!

Schema-free Query Execution



```
> CREATE EXTERNAL TABLE schemaful_table (id BIGINT, name TEXT, ...)
  USING text LOCATION 'hdfs:///csv_table';

> CREATE EXTERNAL TABLE schemaless_table (*) USING json LOCATION
  'hdfs:///json_table';

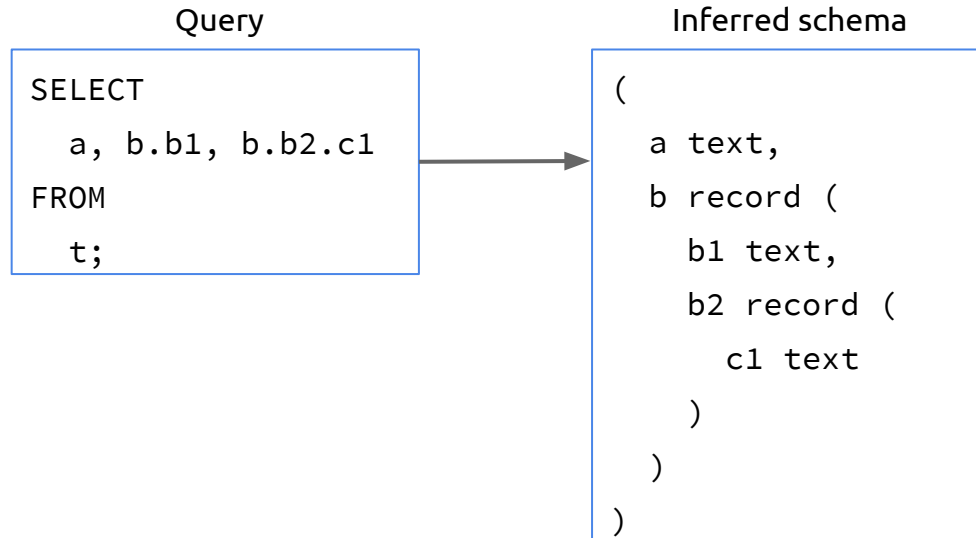
> SELECT name.first_name, name.last_name from schemaless_table;

> SELECT title, count(*) FROM schemaful_table, schemaless_table WHERE
  name = name.last_name GROUP BY title;
```

Schema Inference



- Table schema is inferred at query time
- Example



Demo



Demo with Command line



Roadmap



Roadmap



- 0.12
 - Improved Yarn integration
 - Authentication support
 - JavaScript stored procedure support
 - Scalar subquery support
 - Hive UDF support

Roadmap



- Next generation (beyond 0.12)
 - Exploiting modern hardware
 - Approximate query processing
 - Genetic query optimization
 - And more ...

```
tajo> select question from you;
```