Apache Tajo on Swift

Bringing SQL to the OpenStack World

Jihoon Son
Apache Tajo PMC member

Who am I

- Jihoon Son
  - Ph.D candidate (Computer Science & Engineering, 2010.3 ~)
  - Apache Tajo PMC and Committer (2014.5.1 ~)
  - Mentor of Google Summer of Code (2013)
- Contacts
  - Email: jihoonson AT apache.org
  - LinkedIn: https://www.linkedin.com/in/jihoonson

Outline

- OpenStack Swift
- Apache Tajo
- Tajo on Swift
- Demo
- Our Roadmap

OpenStack Swift

- Popular object storage
  - Images, videos, logs, ...
- Enterprises store objects on Swift to provide their services
  - Usually private clusters

SQL on Swift

- Data analysis is important to improve the quality of their services
  - SQL is one of the most powerful and popular query language
- Many enterprise data analysis tools relying on SQL
  - OLAP, visualization, data mining, …
- Need for using SQL on Swift

Apache Tajo

- Scalable, efficient, and fault-tolerant data warehouse system
  - Support SQL standards compliance
  - Efficient batch execution and interactive ad-hoc analysis
    - Low latency and high throughput
    - No use of MapReduce
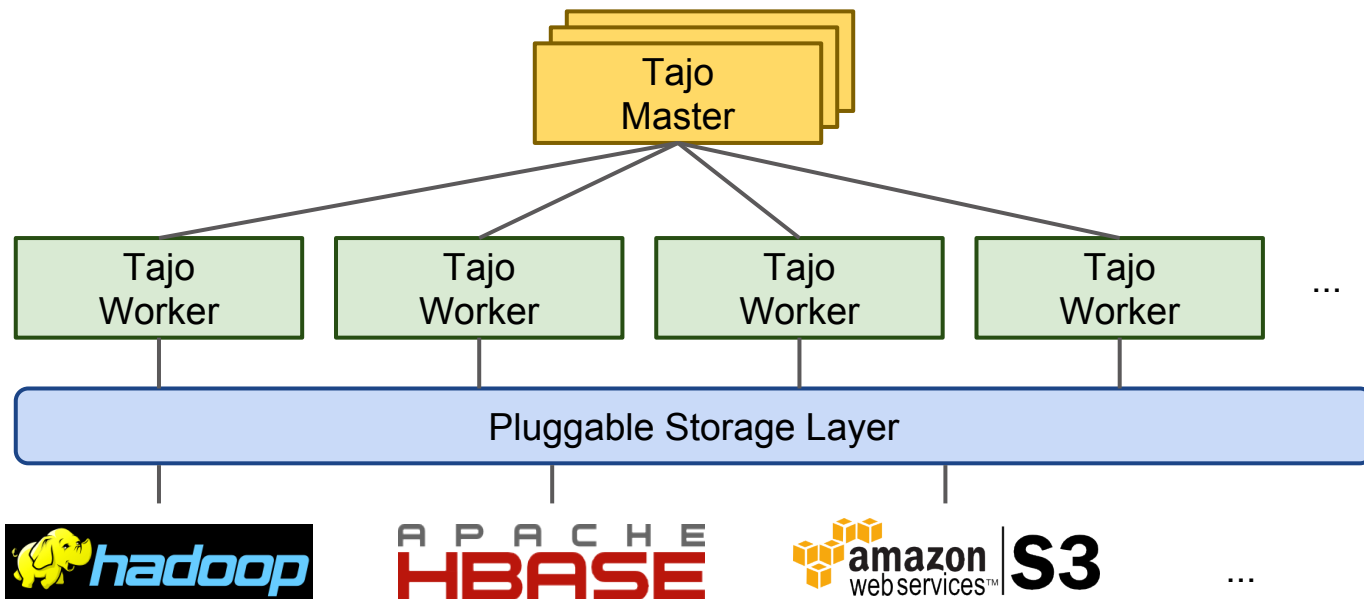 - No single point of failure

Apache Tajo

- Active open source project
  - 18 committers and 16 contributors
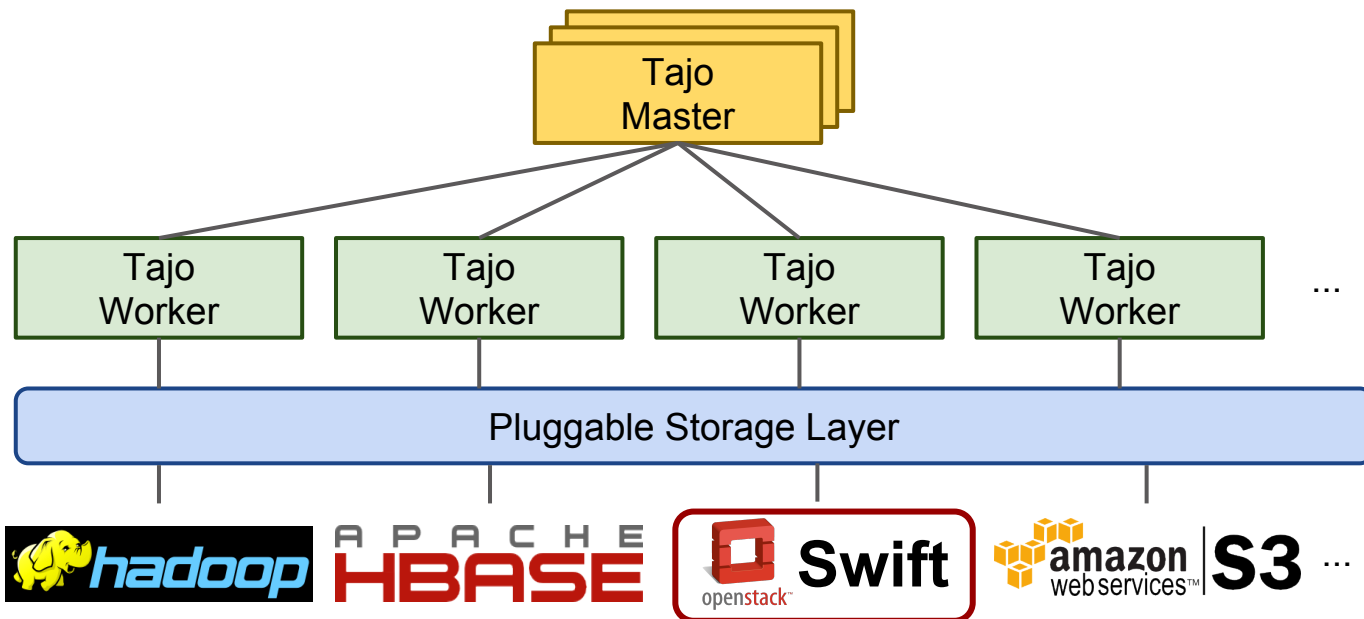 - Activity summary

Issues: 30 Day Summary
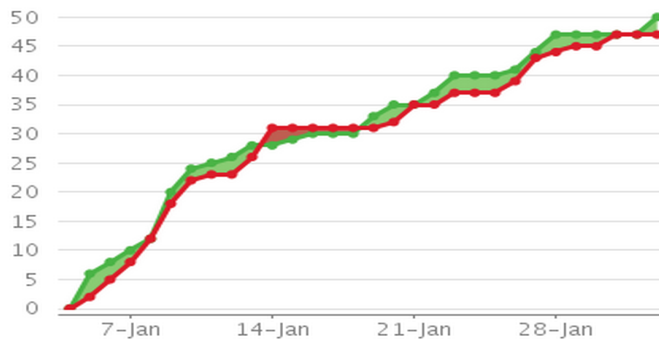
Issues: **47** created and **50** resolved
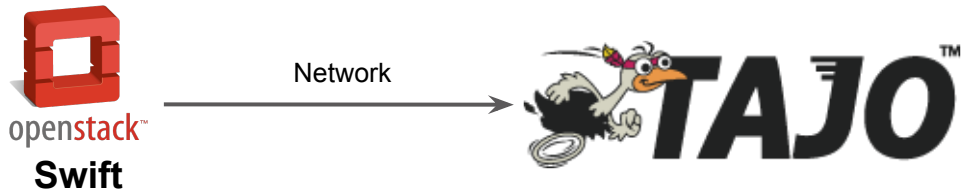
Apache Tajo

Tajo on Swift

Tajo on Swift

- No need to modify code of Tajo and Swift
  - Tajo can access Swift with the Hadoop-openstack library
    - But, doesn't need to install or run Hadoop
  - Just use it

Tajo on Swift

- Configuration highlights
  - Swift configuration
    - Need the keystone authentication for the Hadoop
    - No additional configurations
  - HDFS configuration
    - Different cloud providers support
      - Key name pattern

   `fs.swift.service.${provider}`

Tajo on Swift

● Configuration highl
  ○ Swift configuration
    ■ Need the keysto
    ■ No additional co
  ○ HDFS configuratio
    ■ Different cloud p
      ● Key name pat

        fs.swift.se

```xml
<property>
  <name>fs.swift.impl</name>
  <value>org.apache.hadoop.fs.swift.snative.SwiftNativeFileSystem</value>
</property>
<property>
  <name>fs.swift.blocksize</name>
  <value>131072</value>
</property>
<property>
  <name>fs.swift.service.tajo.auth.url</name>
  <value>http://192.168.0.1:5000/v2.0/tokens</value>
</property>
<property>
  <name>fs.swift.service.tajo.tenant</name>
  <value>demo</value>
</property>
<property>
  <name>fs.swift.service.tajo.username</name>
  <value>hadoop</value>
</property>
<property>
  <name>fs.swift.service.tajo.password</name>
  <value>{password}</value>
</property>
```
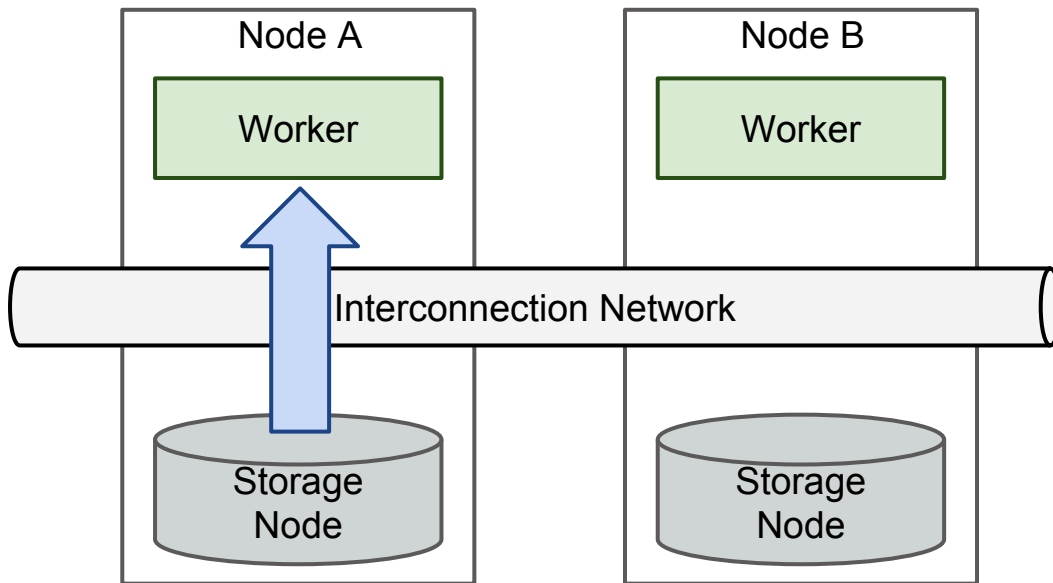
Tajo on Swift

- ## Data locality problem
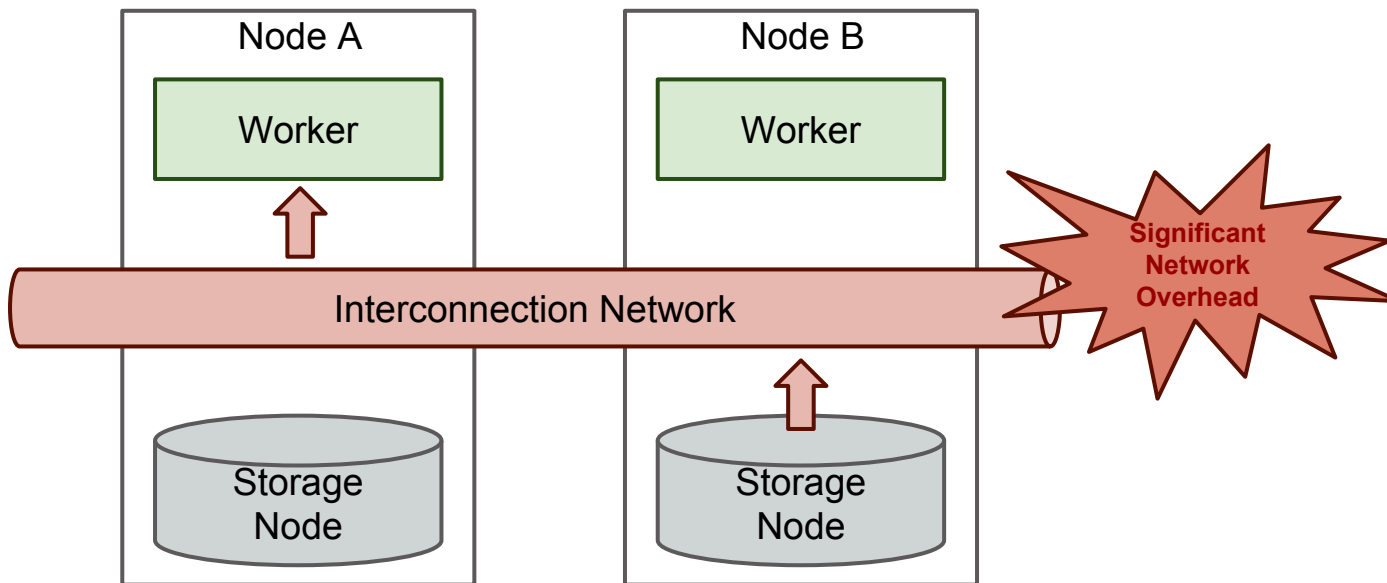
Tajo on Swift

● Data locality problem

Advanced Integration

- List endpoints middleware
    - Providing the location information of objects, accounts or containers
        - Tajo workers can directly access each object
    - Example

```
hadoop@t1 ~/tajo-0.10.0-SNAPSHOT $ curl -i -H "Accept: application/json" -H "Content-Type: application/json" -X GET http://192.168.0.1:8080/endpoints/v2/tajo/tpch/customer/customer.tbl.1
HTTP/1.1 200 OK
Content-Length: 278
Content-Type: application/json
X-Trans-Id: tx198d9332f1184d51881b1-0054cc763e
Date: Sat, 31 Jan 2015 06:29:19 GMT

{"headers": {"X-Backend-Storage-Policy-Index": "0"}, "endpoints": ["http://192.168.0.5:6000/sdb1/866/tajo/tpch/customer/customer.tbl.1", "http://192.168.0.6:6000/sdb1/866/tajo/tpch/customer/customer.tbl.1", "http://192.168.0.10:6000/sdb1/866/tajo/tpch/customer/customer.tbl.1"]}hadoop@t1 ~/tajo-0.10.0-SNAPSHOT $
```

Advanced Integration

- ## List endpoints middleware
  - ### Swift configuration

```
[pipeline:main]
pipeline = authtoken cache healthcheck keystoneauth list_endpoints proxy-logging proxy-server
[filter:list_endpoints]
use = egg:swift#list_endpoints
```

  - ### Hadoop configuration

```xml
<property>
  <name>fs.swift.service.tajo.auth.endpoint.prefix</name>
  <value>/endpoints/AUTH_</value>
</property>
<property>
  <name>fs.swift.service.tajo.location-aware</name>
  <value>true</value>
</property>
```
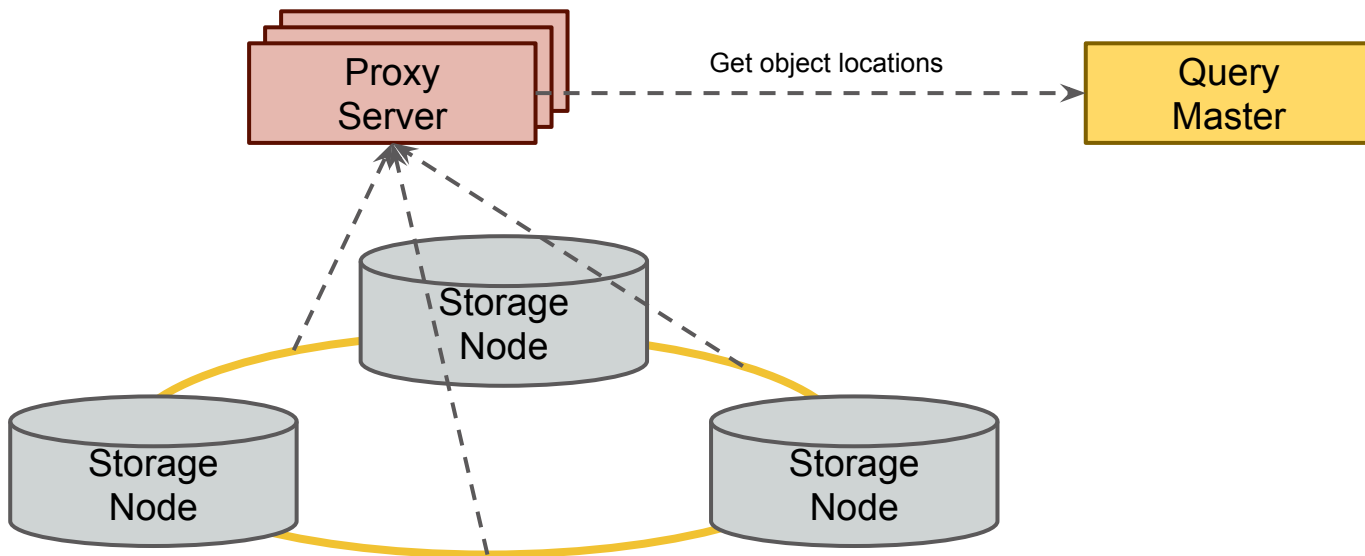
Advanced Integration

- Location-aware computing
  - Moving the processing close to the data
    - Avoiding the performance degradation due to the data transfer over the network
  - Important issue when Tajo and Swift share the same cluster
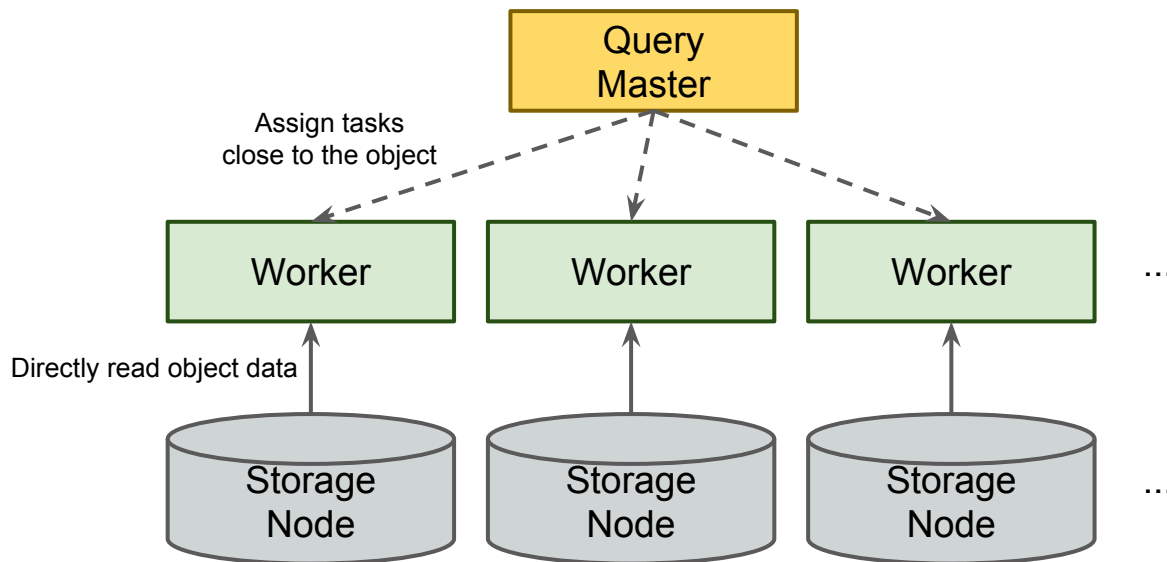
Location-aware Computing
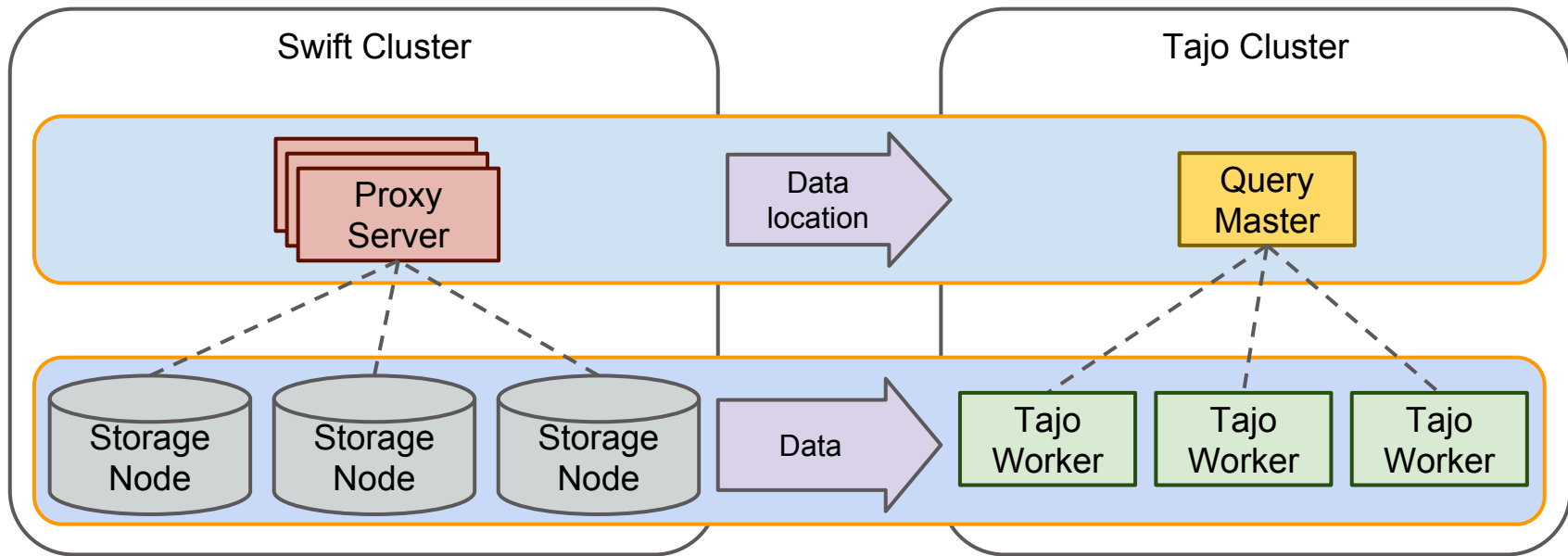
Location-aware Computing

1. Getting object locations from the ring

Location-aware Computing

2. Assigning tasks based on object locations

Query Master

Assign tasks close to the object

Worker   Worker   Worker   ...

Directly read object data

Storage Node   Storage Node   Storage Node   ...

Demo

Our Roadmap

- Storage layer specialized for Swift
- Block storage support
  - Cinder and Ceph
- Provisioning Tajo clusters
  - Sahara
  - Heat, TOSCA

Thanks!

http://tajo.apache.org/