QueryArrow: Semantically Unified Query and Update of Heterogeneous Data Stores

Hao Xu
xuhao@renci.org

Ben Keller Antoine de Torcy Jason Coposky

University of North Carolina at Chapel Hill

June 14, 2017

Acknowledgement

This research is paritally supported by

- The National Science Foundation under Grant Number OCI 0940841 "DataNet Federation Consortium" (Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation) and
- iRODS Consortium

Resources

- QueryArrow: `https://github.com/xu-hao/QueryArrow` fs branch
- QueryArrow Database Plugin: `http://github.com/xu-hao/temporary-irods-mod` qa3 branch

Motivting Applications

Big Metadata Challenge:

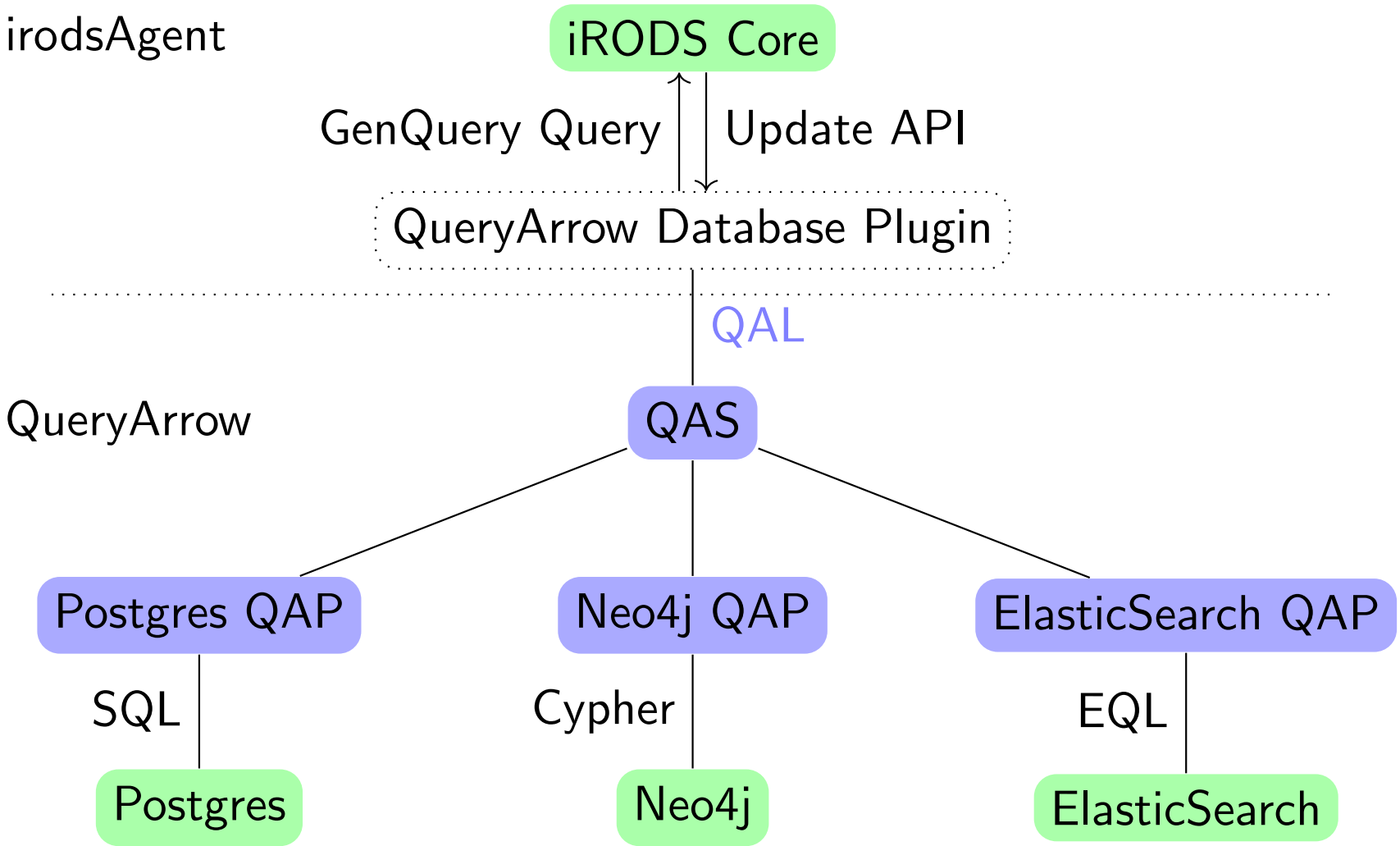
- Aggregation: integrating metadata from multiple data stores
- Policies: for example, metadata access control
- Unify query and update
- Semantically specification
- Heterogeneity: NoSQL, Indexing
- Improve Testing, Modularity, Robustness

QueryArrow Elements

QueryArrow is a semantically unified query and update system for heterogeneous metadata store

- QueryArrow Service (QAS): Register Databases and Execution of QAL
- QueryArrow Language (QAL): Configuration, QL/DML
- QueryArrow Plugins (QAP): Mappings between QAL and Databases
- iRODS QueryArrow Database Plugin

QueryArrow Architecture

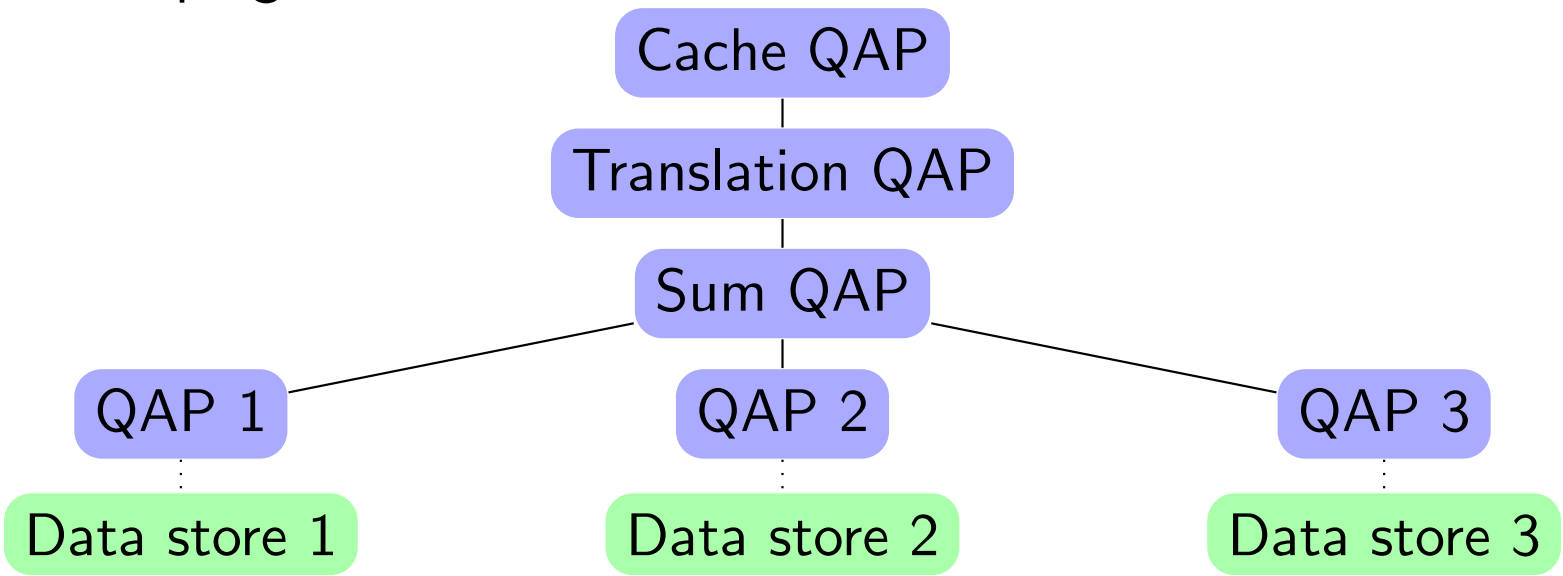


QAPs

Name	Description
Sum QAP	aggregation
Translation QAP	policy support
Cache QAP	caching
Mutable Map QAP	in-memory mutable map
Immutable Map QAP	in-memory immutable map
ElasticSearch QAP	interfacing with ElasticSearch
Neo4j QAP	interfacing with Neo4j
PostgreSQL QAP	interfacing with Postgres
SQLite3 QAP	interfacing with SQLite3
CockroachDB QAP	interfacing with CockroachDB
FileSystem QAP	file system

QAP Composition

Higher-order plugin



QAL Syntax

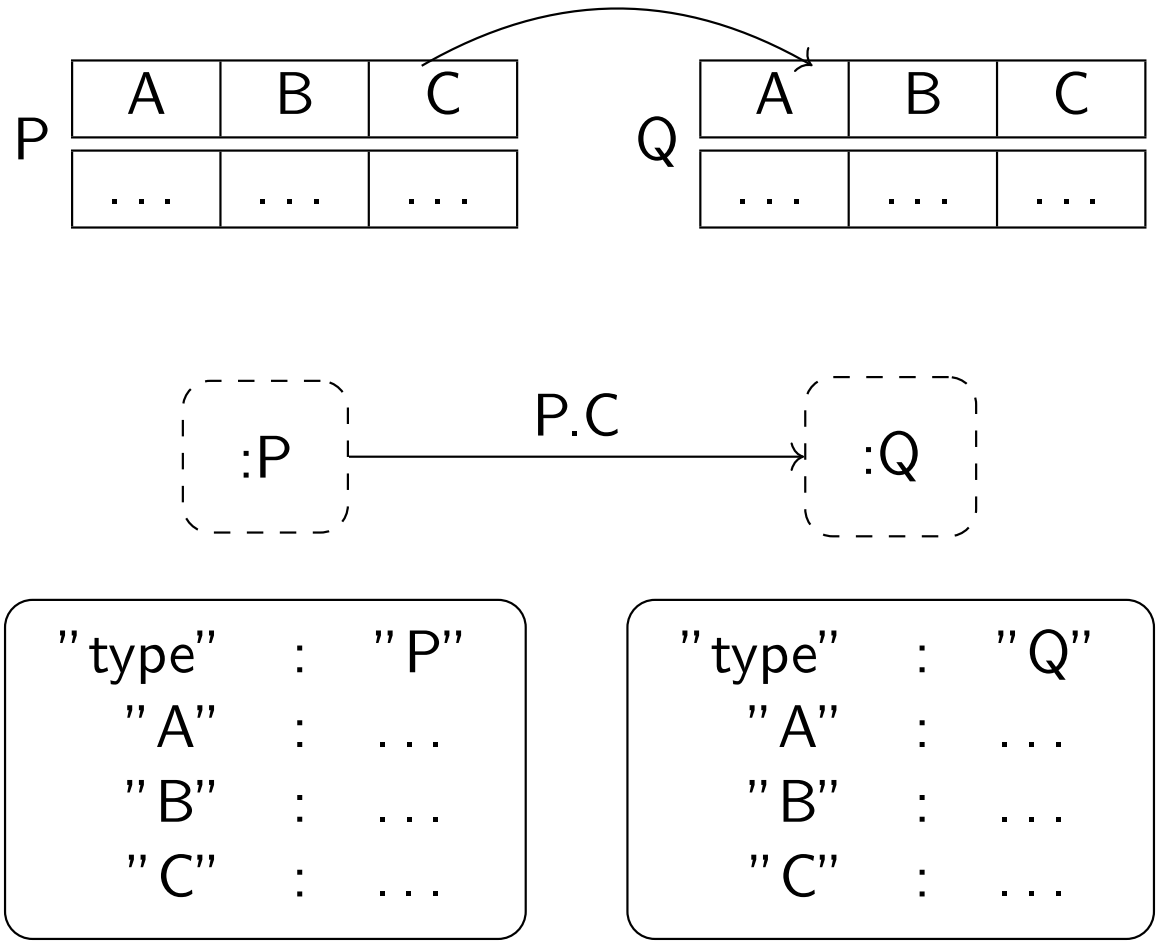
$$
\begin{array}{rcll}
t & ::= & p \mid v & \textit{terms} \\
a & ::= & P(t_1, \ldots, t_n) & \textit{atom} \\
c & ::= & a \mid \textit{insert } a \mid \textit{delete } a \mid g\ c \mid b & \\
 & & \mid \mathbf{1} \mid \mathbf{0} \mid c \oplus c \mid c \otimes c & \textit{command} \\
\hline
pt & ::= & \textit{input? output? key? ty} & \textit{parameter type} \\
R & ::= & \textit{rewrite } a\ c \mid \textit{rewrite insert } a\ c & \\
 & & \mid \textit{rewrite delete } a\ c & \\
 & & \mid \textit{predicate } P(pt_1, \ldots, pt_n) & \\
 & & \mid \textit{import} \ldots & \\
 & & \mid \textit{export} \ldots & \textit{configuration}
\end{array}
$$

QAL Semantics

"store-and-heap" Monad

`https://github.com/xu-hao/CertifiedQueryArrow`

QAL Translation



QueryArrow Service

- Local
- Unix domain socket
- TCP
- HTTP

QueryArrow Database Plugin

- Plugs into iRODS, requires no changes to iRODS core except for bug fixes
- Talks to QueryArrow Service
- Implements 84 db plugin functions
- Backward compatibility with GenQuery
- Tested on 4.2.x:
  - 4.2.0: PostgreSQL QAP passed python tests suite except 2 tests using ssl plugin on Ubuntu 16.04/Centos 7
  - 4.2.1: PostgreSQL QAP passed python tests suite except 2 tests using ssl plugin on Ubuntu 16.04/Centos 7 is running

Install Script

Run iRODS install script through QueryArrow

Demo

- Only Postgresql
- Combined Postgresql and Neo4j