

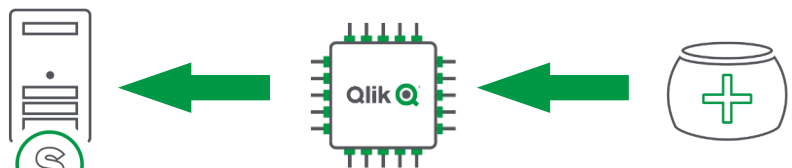
## What is QABDI ?

**Q**lik  
**A**ssociative  
**B**ig  
**D**ata  
**I**ndex

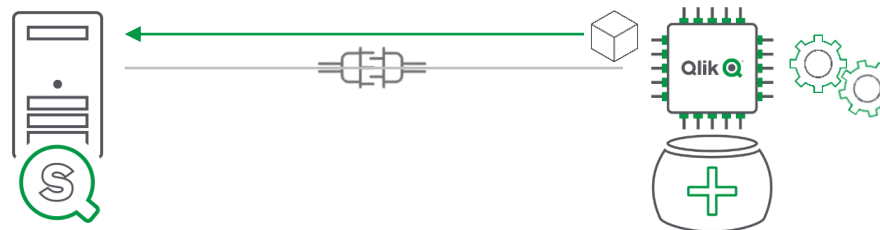
**QABDI is** a licensed product that pairs with Qlik Sense.

Using the Qlik Sense connector, QABDI offers quick, high performance and flexible associative experiences with large amounts of data.

**QABDI is deployed** using different technologies, on a flexible platform. Qlik Sense, QABDI and the BDI source can communicate on a combination of varied architecture:



**QABDI works by** connecting Qlik Sense to the ABDI index. QABDI builds up an index of a targeted big data source. Qlik Sense uses the QABDI connector to attach to the BDI back-end.



Using a select statement, prefixed with “QSL” as the extraction syntax, ABDI generates the requested dataset in JSON format. The specified data is then sent back to Qlik Sense for analysis.

```
QSL SET hDataHandle =
    {[data_model].1 <UniqueID={123}>};

[QStable]:
QSL SELECT
    [Field_01],
    [Field_02],
    [Field_03]
FROM [data_model].[table_name]
AT STATE hDataHandle;
```

## QABDI architecture

QABDI container run-time:



QABDI container orchestrator:



kubernetes

Package manager for K8s:

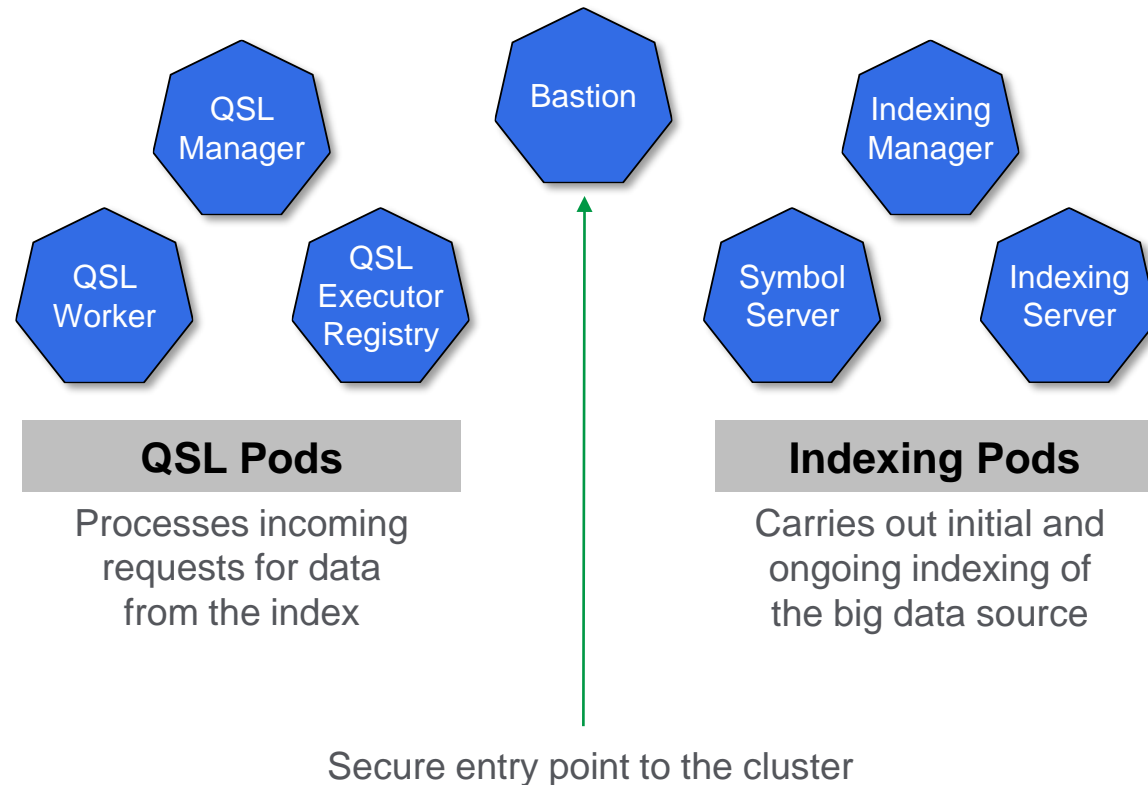


Qlik repository for binaries:

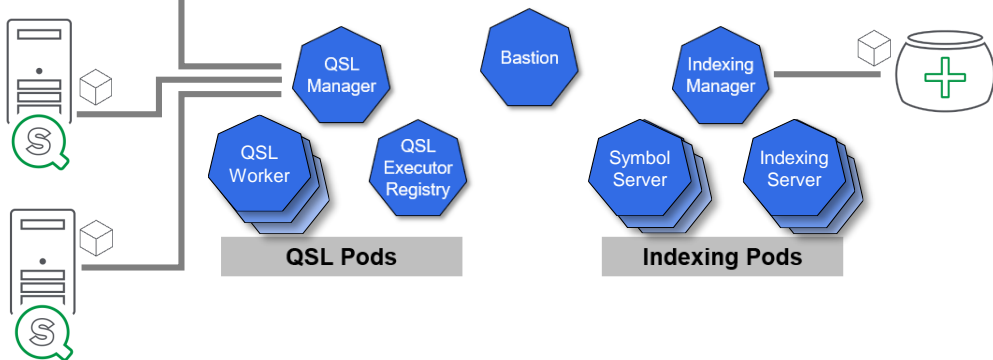


Jfrog Bintray

Kubernetes (K8s) pods are divided into two main categories:



*To help improve throughput, additional index and worker pods can be deployed. This containerized scaling helps mitigate increasing data volumes and requests from QS.*



## Big data source preparation

The **big data source** should be in an Apache Parquet format utilizing a three-level folder hierarchy structure:



+ dataset

**Root**  
**Table**  
**Directories**

**Dataset**  
**Folders**

**Dataset**  
**Files**

+ table1.table

+ table1\_set1.parquet

table1\_set1\_file1.parquet

table1\_set2\_file2.parquet

+ table1\_set2.parquet

+ table1\_set3.parquet

+ table2.table

+ table3.table

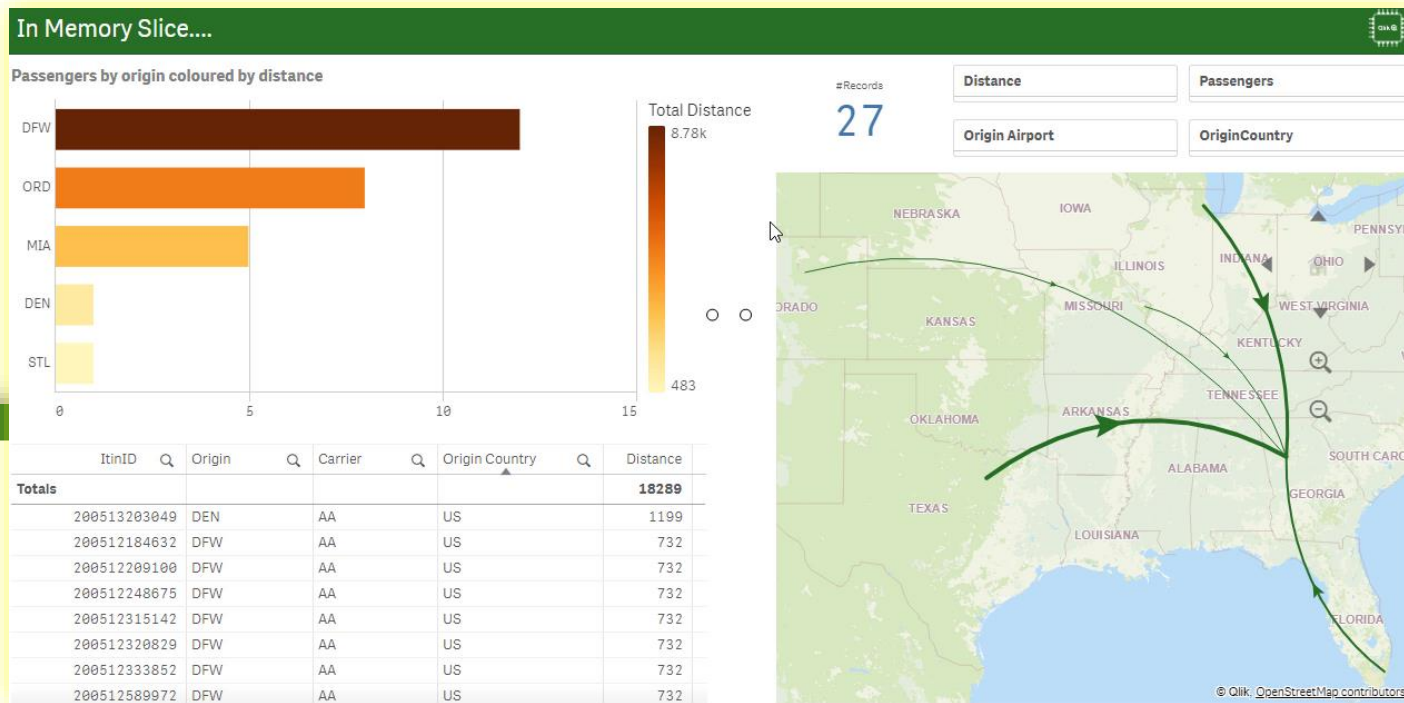
*Schema generation:*

*The source data is scanned and a metadata schema is created. This is subsequently used to ensure additional data added to the index, matches the structure.*

```
"data_set_name" : "alltripssmall",
"tables" :
[
  {
    "fields" :
    [
      {
        "col_no" : 0,
        "name" : "ItinID",
        "type" : "StringType"
      },
      ...
      {
        "col_no" : 35,
        "name" :
"CouponGeoType",
        "type" : "StringType"
      }
    ],
    "name" : "Flights",
    "num_indexlets" : 1
  },
  ...
]
```

## QABDI servicing ODAG (On-Demand App Generation)

- QS can operate an app data model in QABDI “Live Mode”. This becomes the summary app.
- Connected via App navigation links
- Shows filtered detail information in newly generated app:



**Flights**

**Flight Details**

Flight Details\_QSE-administrator  
6 Jun 2019, 12:25:06

**Origins and Destinations**

Origin: DEN, ORD, DFW, STL, MIA, MFR, LAS, SFO, LAX, MRY, LAS, SFO, LAX, MRY, ONT, PDX, LAX, MRY, ONT, PDX, PHX, ATL, RDU

Destination: ATL, MFR, LAS, SFO, LAX, MRY, ONT, PDX, ORD, PHX, RDU, VLD, BFL, BOI, BOS

**Generate new app**