QVD Utils

EVL QVD Utils enables reading/writing QVD files without using Qlik Sense or QlikView. Also provides metadata from QVD/QVX header. Useful for converting various sources directly to/from a QVD file, and to prevent overloading a Qlik server by processing many, or large, QVD files.

EVL Microservices are built on top of the core EVL software and retain its flexibility, robustness, high productivity, and ability to read data from various sources; including JSON and Excel files, databases—Oracle, Teradata, PostgeSQL, etc.—and streaming data like Kafka.

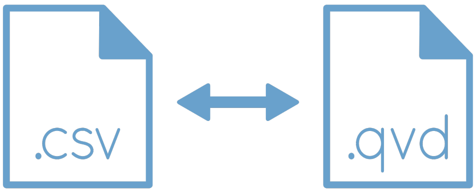
QVD Utils used standalone

Command Line Utilities:

`qvd-header` – retrieve information from QVD header: number of records, field names as JSON, etc.

`csv2qvd` – convert CSV to QVD

`qvd2csv` – convert QVD to CSV

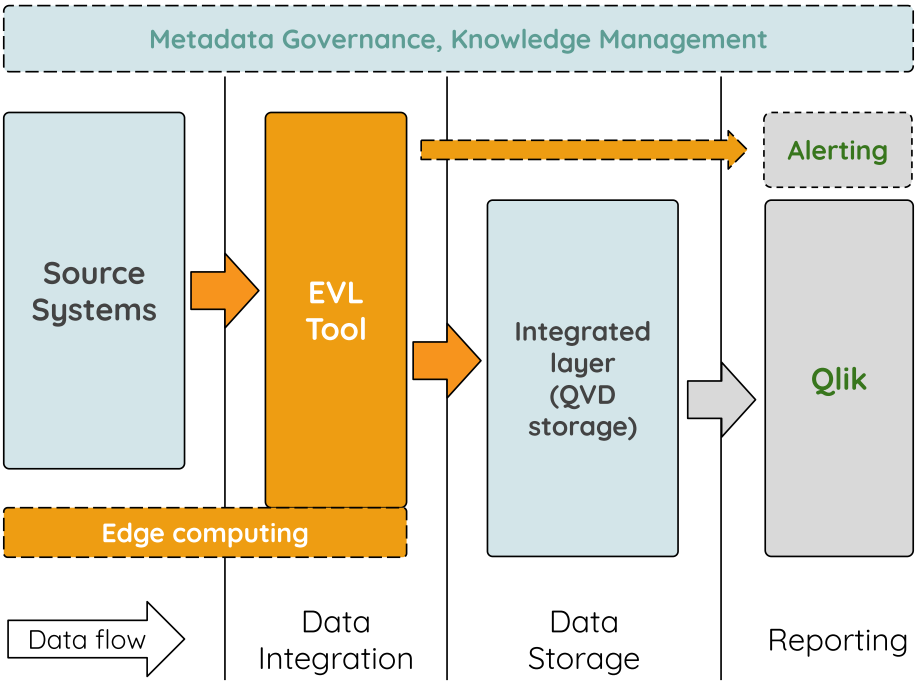


Advantages:

- High read/write performance
- Ideal for automation and batch processing
- Save Qlik resources by offloading processes
- Improve data load times by integrating load processes into standard ETL

QVD Utils in combination with other Microservices

- In join with other EVL Microservice(s) (like EVL Staging) can be used for example in QVD-based architecture:



CSV to QVD Conversion Example

Example input CSV file with header, and semicolon separated fields:

ID	Name	Code	Price	Created
1	item_11	11	15,85	2016-05-04
2	item_12	12	21,25	2019-11-11
3	item_21	21	12,99	2019-10-05

Run:

```
$ csv2qvd table.csv > table.qvd
```

csv2qvd will use the field names present in the header, and data types derived from the values, to produce an output file named "table.qvd".

There are several options to change the default behaviour. You can specify your field names when there is no header in the CSV for example:

```
$ csv2qvd --no-header --header="ID,Name,Code,Price,Created" table.csv >table.qvd
```

QVD to CSV Conversion Example

A CSV file can be created from a QVD file, with the above mentioned structure, by running:

```
$ qvd2csv table.qvd > table.csv
```

qvd2csv will use the field names present in the QVD header to create an output file named "table.csv" with header, and colon separators.

There are several options to change the default behaviour. For example specify different field separators, add Windows end-of-lines, and specify format for dates:

```
$ qvd2csv --dos-eol --separator=";" --date="%d/%m/%Y" table.qvd > table.csv
```

Get metadata from QVD

Get number of records:

```
$ qvd-header --no-of-records table.qvd
```

Get the list of fields with their data types in JSON format:

```
$ qvd-header --output=json --fields table.qvd
```

When the estimated data types are not expected ones, it can be manually changed via declarations in the EVD file (EVL data definition file). See https://docs.evltool.com/evl-qvd-utils for details.