ezMQ - Data transmission based on message queue among the microservices

Description

This messaging scheme, ezMQ, provides single unified data streaming interface among the EdgeX microservices regardless of underlying Message Queue protocol. It allows the microservices to exchange small size data packets (actually, we need to define the meaning of 'small size') or events with high frequency. So the microservices will be able to stream data inside edge device(in-proc, localhost) or with the other edge devices over a high speed LAN (Wired/Wireless). For the simultaneous data transmission, multiple microservices can publish to or subscribe from an endpoint which could be an <IP, port> combination or named topics. The microservices can initiate or stop data flow anytime without worrying about underlying operations. The microservices will be able to set standard filters over streaming data and the ezMQ will take care of applying them in highly optimized manner. The ezMQ will be used as data streaming backbone between different instances of the microservices . And the ezMQ provides high speed data serialization /deserialization for effective data transmission. And the ezMQ will also support dynamic topic based data delivery and creation and removal of filters.

Highlighted Features

- Share streaming data either directly between micro services or on topics shared across an edge cluster.
- Provide API's (ezMQ client library) to stream reliable or regular data.
- Provide API's (ezMQ client library) to abstract highly efficient and standardized serialization and deserialization of data to be shared between microservices regardless of language or platform used to develop microservices.
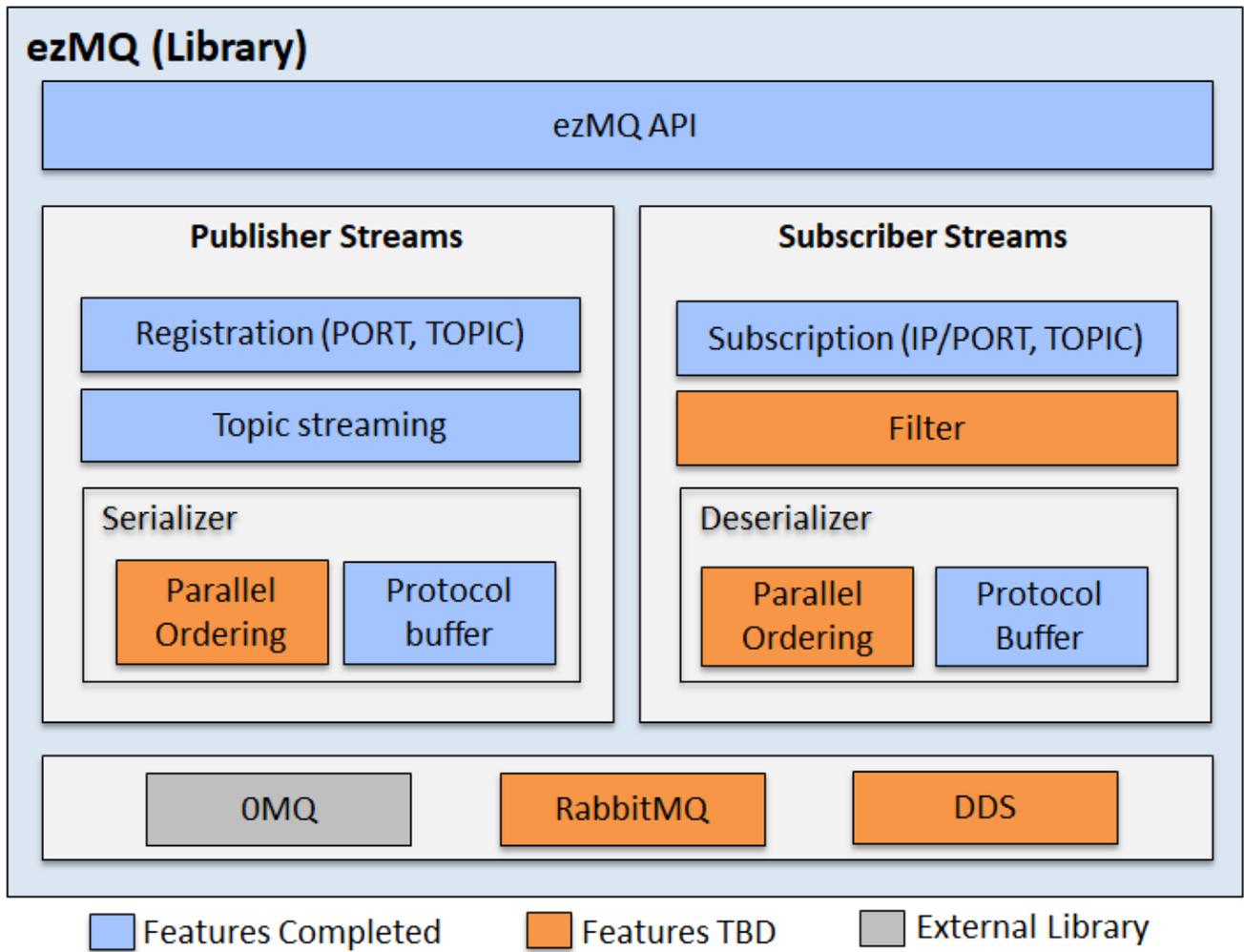
Functional Component (ezMQ Library)

- Deployed Location: Inside micro services.
- Role: Interface for all data streaming requirements, serialization / de-serialization of data, direct streaming to other microservices.

Design

The attached documents describe the high level design consideration.

- Data transmission scheme based on message queue

Code repository

Current status

ezMQ library has been provided in C, C++, GoLang, Java, Node.js, and Python.