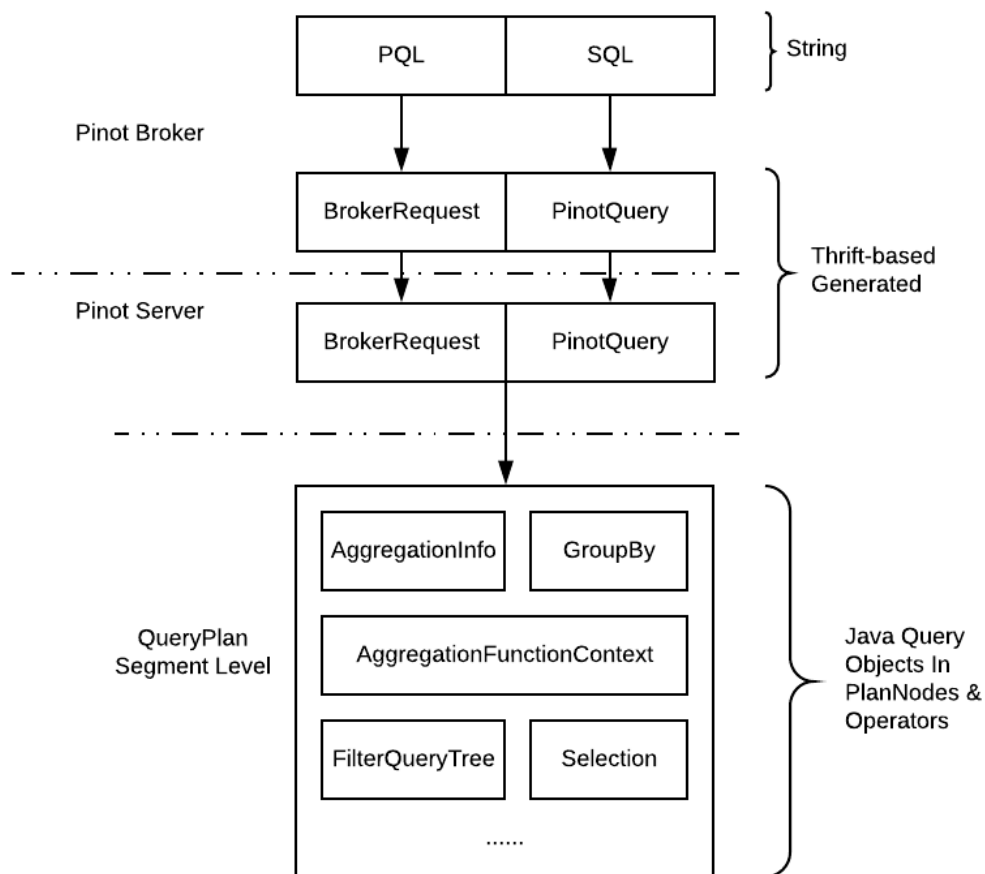


Pinot SQL Support Migration Plan

The goal of this plan is to replace the usage of BrokerRequest, which is parsed from PQL with PinotQuery, which is parsed from Calcite SQL.

Current Query Execution Flow

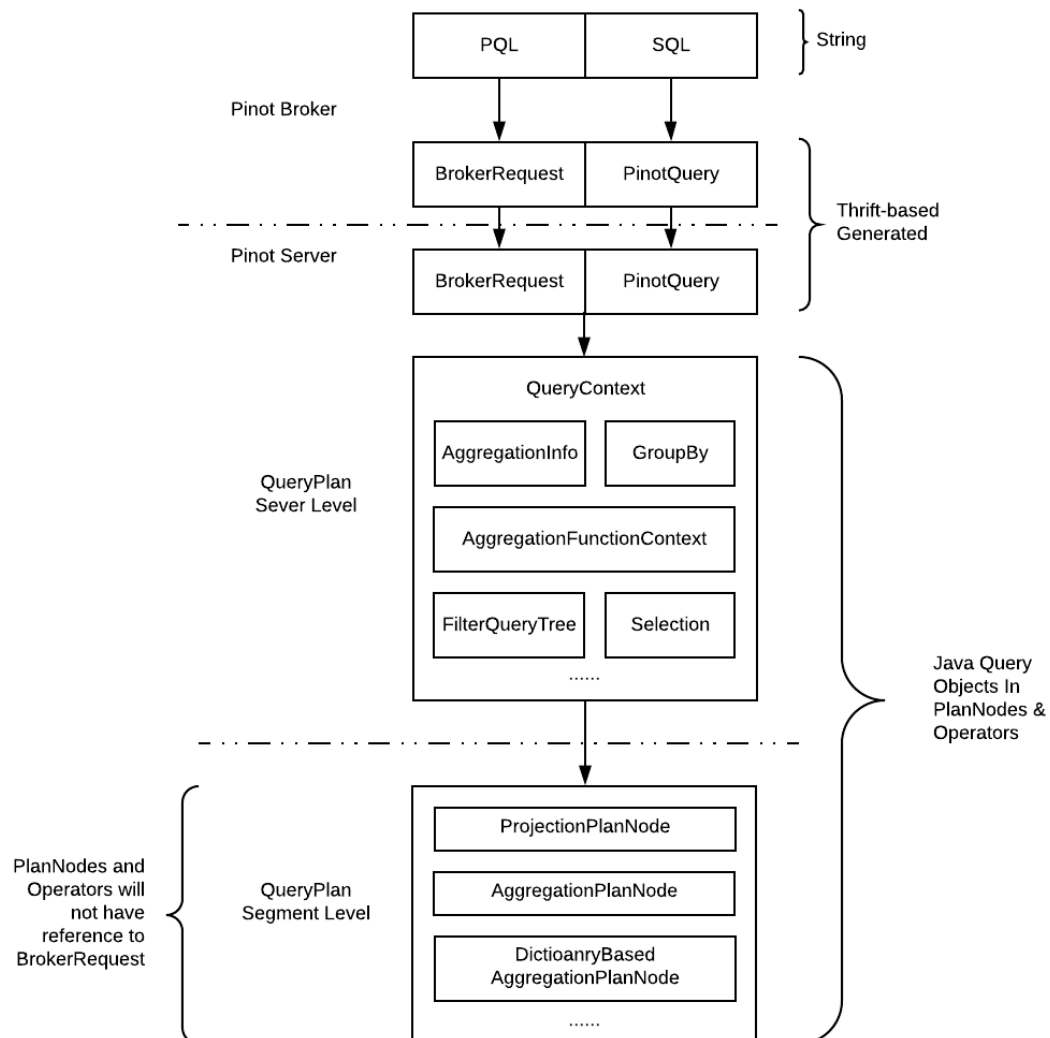


Originally we parse PQL to BrokerRequest at Pinot Broker layer and send it to Pinot Server to generate QueryPlan. Recently we also added a Thrift based class PinotQuery to hold structure parsed from Calcite SQL.

Based on the query type(Aggregation/GroupBy/Selection), detailed PlanNodes and Operators are generated at segment level, which means that we deeply integrated BrokerRequest into PlanNodes and Operators...

1. BrokerRequest is widely spread into multiple PlanNodes, which the run() method will be called to return corresponding Operators at segment level.
 - a. org.apache.pinot.core.plan.AggregationGroupByOrderByPlanNode
 - b. org.apache.pinot.core.plan.AggregationGroupByPlanNode
 - c. org.apache.pinot.core.plan.AggregationPlanNode
 - d. org.apache.pinot.core.plan.CombinePlanNode
 - e. org.apache.pinot.core.plan.DictionaryBasedAggregationPlanNode
 - f. org.apache.pinot.core.plan.DocIdSetPlanNode
 - g. org.apache.pinot.core.plan.FilterPlanNode
 - h. org.apache.pinot.core.plan.MetadataBasedAggregationPlanNode
 - i. org.apache.pinot.core.plan.SelectionPlanNode
 - j. org.apache.pinot.core.plan.TransformPlanNode
2. BrokerRequest is also spread into several CombineOperators
 - a. org.apache.pinot.core.operator.CombineOperator
 - b. org.apache.pinot.core.operator.CombineGroupByOperator
 - c. org.apache.pinot.core.operator.CombineGroupByOrderByOperator

Proposed New Query Execution Flow



The new proposed flow introduces a new class `QueryContext` which holds the information parsed from `BrokerRequest`, e.g. `AggregationInfo`, `GroupBy`, etc. This parse only occurs once, then this `QueryContext` will be passed down to ALL `PlanNodes` as well as `CombineOperators`.

Those objects in `QueryContext` should be common across all the segments. We should provide a static method to parse `BrokerRequest` into `QueryContext`, so that both Pinot Broker and Pinot Server will be able to use it to convert query objects.

Pros:

- `QueryContext` is a processed object which is more efficient compared to `BrokerRequest` to be used in each `PlanNode/Operator`

- QueryContext is an internal implementation, so we don't have backward compatibility issue.
- Performance improvement by not parsing BrokerRequest for every segment.

Overall Execution Plan:

1. Define and create QueryContext class which summarizes parsed information from BrokerRequest.
2. Build tooling to generate QueryContext from BrokerRequest
3. Replace the usage of BrokerRequest inside PlanNodes and Operators with QueryContext
4. Build tooling to generate QueryContext from PinotQuery
5. Validation on query results