Outlines for Final Report

Abstract

TODO - This is the last section to write

One of the major new features in Java 8 is the introduction of the stream functionality which allows developers processing data in a declarative way.

Introduction

TODO – This is also one of the last sections to write

Considerations

* User objects provide getter and setter accessor method.

Technologies

* Java 8
* Derby 10.13.1.1
* SQL Parser possibly (undetermined)

Streams API

Introduction about what Streams API offers in details, talk about also about parallelism. Explain all the operations that we use in our project, include an example for every operation.

.map()

.filter()

.flatmap()

.collect()

.concat()

.foreach() //not sure

Relational Algebra Operations

Explain what relational algebra operations are, state the basic operations and give an example for all of them.

Select

This is the most straight-forward mapping operation that we can do in Java using Streams. For simulating the WHERE clause in SQL, Streams provides the operation .filter().

The interesting thing with filter() and the Stream API in general is that the operation can apply at any place in the call chain, unlike the WHERE clause, which is limited to be placed right after the FROM clause.

Approach

Talk about the approach selected for the project (using Derby to get the execution plan, so we do not to spend time developing a sql parser into relation algebra operations)

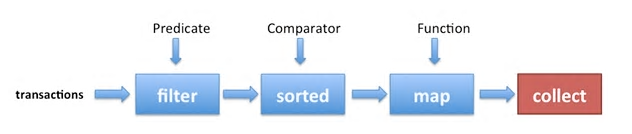
Decision Design

We are using Map<String, Object> as a data structure for simulating temporal tables. We were evaluation the idea of creating dynamically temporal classes in memory using ByteBuddy, but we found problems in some algebra operations such as Cartesian product.

We provide our own query execution plan and its correspondent parser, so we can change Derby for any other database engine.

Include classes’ diagram??

Talk about what is Stream and how it works. Explain all the operations from Stream that we use in the project for simulating SQL statements, include example and figures similar to this.



Include activity diagram

Our Solution

Explain our solution, explain the translation of all the relational algebra operations to streams operations, give an example for all of them and state the limitation of our solution.

Talk about the reason of using Supplier<Stream>

As we mentioned earlier, streams cannot be reused, to overcome this limitation we create a stream supplier to construct a new streams.

Projection:

* Limitations
  + Operations like concatenation are not supported (name || id).

Select:

* Limitations
  + For now, only supports and operations. Not supported or, group of expressions enclose by parenthesis, tuple predicate (i.e (name, id) = (‘Maria’, id)).

User Manual

Include a section showing how to use the project providing maybe some screenshots of queries examples and the result.

Conclusion