

1.0 Initialize a ReducerFunctionalObjectSphinx object – creation in the constructor ReducingHandlerBuilder instance and building an instance of ReducingHandler class.

Initialize a \_pocoJSONReducingInputMessageConvertor object.

2.0 void accumulate(std::string& jsonReducingInputMessage, unsigned long long itemId) - takes as parameters a jsonReducingInputMessage string(see Reducer functional object message protocol.doc) and an itemId to which the input data belong to; invokes

\_pocoJSONReducingInputMessageConvertor.convertToReducingInputMessageFrom(2.1) method to convert the jsonReducingInputMessage string(json format) to ASM::reduce\_types::ReducingInputMessage format(native language format); invokes

reducingHandler->accumulateReducingData(2.2) method to accumulate data for the given itemId.

3.0 bool isCompleteTask(unsigned long long itemId, int nodesCount - takes itemId to check condition to finish processing data related to the itemId. Second parameter noesCount is second part of the checking condition; invokes method \_reducingHandler>getAccumulateReducingPartsNumberBy(3.1) to get number count of data part which were accumulated into the handler for the itemId.

4.0 reduce(unsigned long long itemId) – executes the reduce task for data related to the itemId; invokes \_reducingHandler->makeReducing(4.1) and \_reducingHandler->deleteReducingTaskBy(4.2)

5.0 unsigned int getTasksNumber() - return the number of unique itemIds which have data in the reduceHandler's structures, gets it information from \_reducingHandler->getTasksNumber() (5.1)

6.0 int getErrCode() - return an error code which was set by the last invoked method of a ReducerFunctionalObjectSphinx object

7.0 std::string getErrMsg() - return an error message which was set by the last invoked method of a ReducerFunctionalObjectSphinx object

8.0 int cleanupExpiredTasksByTTL(unsigned int maxRemoveTasks = 1, unsigned int ttl = 0) – removes elements from the global queue of running tasks number and inserts them in to the global queue of “terminated by TTL tasks”, removes no more that maxRemoveTasks elements which ttl value is less that input ttl parameters; returns number of removed elements, invokes \_reducingHandler->cleanupExpiredTasksByTTL(8.1)

9.0 unsigned int cleanupExpiredTasksByTTLQueue( unsigned int maxRemoveItems, unsigned int minTerminateTime ) - removes elements from the global queue of “terminated by TTL tasks”, removes no more that maxRemoveTasks elements which ttl value is less that input minTerminateTime parameters; returns number of removed elements; invokes \_reducingHandler->cleanupExpiredTasksByTTLQueue(9.1)

10.0 getExceededTTLTasks( unsigned int maxTaskNumber, std::vector<unsigned long long>& exceededTTLTasks, unsigned int ttl = 0 ) - adds to the vector(exceededTTLTasks) itemId which ttl value is less that input ttl parameters; adds to the vector no more that maxTaskNumber elements; invokes \_reducingHandler->getExceededTTLTasks(10.1)

11.0 int getRejectedMessages() - return number of rejected messages