Design Doc

Major functions:

1. TransactionManager
2. DataManager (Site)
3. Main

Function descriptions:

1. TransactionManager:
   TransactionManager is the core function to perform operations.
   Major methods:
   - parse: parse the input file or standard output into the required format and invoke a transaction manager to execute all the events. Write the execution results to an output file.
   - get_command: invoke a transaction manager according to the command type to execute the event
   - command: begin(), end(), read(), write(), recover(), dump(), abort(), commit(), fail() and recover() are included and will call the corresponding method in DataManager
   - deadlock_detect: use dfs to detect if there is a cycle in the lock graph
   - error_handler: exception handler to handle the invalid input error

   Variable:
   1. transaction_table
   2. timestamp
   3. operation_queue
   4. data_manager_list

   Internal classes:
   1. Parser
   2. Transaction
   3. Operation

2. DataManager (Site):
   Each site represents one data manager. DataManager stores the information about each site. TransactionManager calls DataManager to perform operations on each variable at each site.
   Major methods:
   - command: implementation of begin(), end(), read(), write(), recover(), dump(), abort(), commit(), fail() and recover() are in DM.

- generate_graph: generate a graph for each site. An edge will be added if there is wait-to relationship between two transactions.

Variable:

1. site_id
2. variable_table
3. is_working
4. visited_transaction

Internal classes:

1. LockItem (superclass of ReadLockItem and WriteLockItem)
2. LockManager(each variable has its own LM)
3. CommitValue
4. Variable

Division of work:

Ziheng Cao has done the deadlock detection part in both TM and DM. Yucong Liu has done command, parse, error handler in both TM and Dm.