

The following only discusses the parts that were evaluated as 'no' in the last feedback to show my subsequent works and system completion.

1. Is XML parsing implemented?

Unfortunately, this part has not yet been realized.

2. Is the Aggregation Server persistent?

Complete the file persistence in lines 35-60 of the file `PriorityQueueProcess.java`.

3. Is restore on failure implemented?

The modification in line 52 of `AggregationServer.java` ensures that the aggregation server can still read files normally after the failure and restart.

4. Is the heartbeat implemented on the Aggregation Server?

The main function of the `HeartBeatProcess` class in line 179 of `AggregationServer` is to process heartbeat packets sent from the content server. If the heartbeat packet cannot be received within 6 seconds (short time for testing), the file corresponding to the content server on the aggregation server will be deleted. Each content server has an independent `HeartBeatProcess` thread on the aggregation server side.

5. Is the heartbeat implemented on the Content Server?

The main function of the `HeartBeat` class in line 224 of the `ContentServer` file is to send heartbeat packets to the aggregation server regularly. Each content server will have an independent `HeartBeat` thread.

6. Is the system fault tolerant?

In addition to the above mentioned implementation of aggregation server recovery mechanism to avoid a single point of failure. The reconnection mechanism is also added to the `socketBuilder` method in line 129 of the `GETClient` file. Automatic reconnection will be performed twice after the first connection failure. The failure message is returned after three failed connections.

7. Is testing comprehensive?

The test contents include:

(1) normal flow test (get = put)

(2) multi client test (client1. get = client2. get)

(3) reconnection test (three reconnection attempts will be made after the first connection failure, and failure information will be returned after the last connection failure)

(4) aggregation server persistence test (the put content is persisted locally in the AS. As is unexpectedly closed, it will work normally after restart)

(5) shut down the content server, A local corresponding file will be deleted after 6 seconds

8. General feedback

The corresponding execution code and version information of the readme file have been improved. Removed useless code and improved comments.