Redis Server

It is an in-memory database that allows us to do fast lookups and retrieval. It uses RESP message format for serialisation and deserialisation.

Data Flow our project:

A diagram of a server

AI-generated content may be incorrect.

For speed, the Redis server is implemented using asyncio sockets which allows asynchronous processing of requests/responses and uses TCP connection as RESP is more suitable with TCP connections.

Input supports the following command:

1. String Commands
   1. SET <key> <value> - set the string value of a key with support for following options
      1. EX <seconds> - set the specified expire time, in seconds (a pos int)
      2. PX<milliseconds> - set the specified expire time, in milliseconds (a pos int)
      3. EAXT <timestamp-seconds> - set the specified unix time at which the key will expire, in milliseconds (a pos int)
      4. PXAT <timestamp-milliseconds> - set the specified unix time at which the key will expire, in milliseconds (a pos int)
   2. GET <key> - get the value of a key
   3. DEL <key> - delete a key
   4. EXISTS <key> - returns if key exists, can take multiple keys as arguments and returns the number of existing keys in the list.
   5. ECHO <message> - prints the message.
   6. PING – returns PONG.
2. Integer Commands
   1. INCR <key> - increment the integer value of the key by one
   2. DECR <key> - decrement the integer value of the key by one
3. List Commands
   1. LPUSH <key> <value> - insert the value at the head of the list.
   2. RPUSH <key> <value> - insert the value at the tail of the list.

Output from the Redis Server supports the following RESP style data types:

1. Simple Strings – first byte represented by +.
2. Errors – first byte of reply is -.
3. Integers – first byte of reply is :
4. Bulk Strings – first byte of reply is $
5. Arrays – first byte of reply is \*
6. Null Bulk strings - first byte of reply is $-1
7. Null Arrays - first byte of reply is \*-1

Persistence mechanisms we have used in the project:

1. AOF – we append all the write operations into a single file and replay this file on Redis server restart or crash.