REDIS

- In memory “Non-relational” (key – value) database
- Persistent - saves data as a background task.
- Easy to install and use

Installation

- Easy to install
  Versions 2 is available uder yum or apt-get
  Version 3 is easy to download and install
- Install yourself:
  wget http::/download.redis.io/releases/redis-stable.tar.gz
  tar -xzf redis-stable.targ
  cd redis-stable
  make
  sudo make install

Running

- Can be set up as a daemon, or just run
  redis-server
  in a terminal.
- In another terminal, run
  redis-cli
  and start typing in commands.

Quitting

- Typing
  quit or
  exit
  will get you out of the client
- The only ways I have found to quit the server are:
  ^C or
  pkill redis-server

Some Commands

- The simplest structure associates a value (with the CLI this is a string, but it can be a binary blob) with a symbolic key.
- Key names are pretty arbitrary:
  1234 MyComplexKey-123 “A B C”
- Keys can also be separated into sections:
  MyCDS:Beethoven:23
  (in fact the colon here is just a convention)
  and there is a wild-card search for keys.

Examples

- Set a key:

  >SET MyCDS:Beethoven:1 “Symphony No 6”
  OK
  >SET MyCDS:Beethoven:2 “Symphony No 5”
  OK

- Fetch a key:

  >GET MyCDS:Beethoven:2
  “Symphony No 5”

Examples

- Search for keys:
  >KEYS *B*
  1) “A B C”
  2) “MyCDS:Beethoven:1”
  3) “MyCDS:Beethoven:2”
- Note: It returned keys, not values

HELP

- If you know a command, you can:
  > HELP keys

KEYS pattern

summary: Find all keys matching the given pattern

since: 1.0.0

group: generic

HELP

- There is also a full list of commands with help at
  http://redis.io/commands
- Links to documentation are at
  http://redis.io/documentation
- There is a useful short book you can download called “The Little REDIS Book”. The link is on the page.

More Complex Structures

- As well as simple string values, you can have
  - Lists with push and pop from either end.
  - Sets – no duplicates allowed – and sorted sets (with a value associated with each member; the sort is on the values) – you can select on value range and change/increment the values.
  - Hashes. Each key is associated with one or more tags each with a value. Perhaps the most useful structure.

Hash Tables

- HSET MyCDs:1 title “Symphony No 1”
  OK
- HMSET MyCDs:1 composer “Beethoven”
    conductor “Karajan”
    performer “Boston Symphony”
  OK
- HGET MyCDs:1 composer
  “Beethoven”

Hash Tables (cont)

- HSCAN MyCDs:1 0
  1) 0
  2) 1) “title”
     2) “Symphony No 1”
     3) “composer”
     4) “Beethoven”
     5) “conductor”
     6) “Karajan”
- The 0 tells the scan to start at the beginning.
- The 0 as the first part of the reply tells us we have it all; if is is any other number, use that as the start for the next HSCAN.

Programming

- LUA (a scripting language) can be used on the server.
- There are wrappers that allow programs to connect to the server:
  - HIREDIS is a wrapper for C/C++
  - There are wrappers for Java, Delphi, Bash, Scheme, Go, Python, Ruby, PHP and many more. Go to the “Clients” page on the web-site.
- You can also script the redis-cli

Safety

- By default, the server listens on Internet domain 127.0.0.1 port 6379.
- You can easily change this by editing the well documented redis.conf file and starting the server pointing to it. This can be a security risk.
- From version 3, you can turn off the internet-domain socket and listen only on a Unix-domain socket.