# SSDB - 高性能的支持丰富数据结构的 NoSQL 数据库, 替代 Redis

http://ssdb.io/zh\_cn/

**特性**

* 替代 Redis 数据库, Redis 的 100 倍容量
* LevelDB 网络支持, 使用 C/C++ 开发
* Redis API 兼容, 支持 Redis 客户端
* 适合存储集合数据, 如 list, hash, zset...
* 客户端 API 支持的语言包括: [C++](http://ssdb.io/docs/cpp/), [PHP](http://ssdb.io/docs/zh_cn/php/), Python, [Java](http://ssdb.io/docs/java/), Go
* 持久化的队列服务
* **主从复制, 负载均衡**

[SSDB vs Redis](http://ssdb.io/docs/zh_cn/redis-to-ssdb.html)

**PHP 使用**

<?php

require\_once('SSDB.php');

$ssdb = new SimpleSSDB('127.0.0.1', 8888);

$resp = $ssdb->set('key', '123');

$resp = $ssdb->get('key');

echo $resp; // output: 123

[PHP API 文档](http://ssdb.io/docs/zh_cn/php/)

**安装**

wget --no-check-certificate https://github.com/ideawu/ssdb/archive/master.zip

unzip master

cd ssdb-master

make

# optional, install ssdb in /usr/local/ssdb

sudo make install

**启动**

# start master

./ssdb-server ssdb.conf

# or start as daemon

./ssdb-server -d ssdb.conf

**性能**

性能数据使用 ssdb-bench(SSDB) 和 redis-benchmark(Redis) 来获取.

ConcurrencyQPSSSDB vs Redisssdb-setssdb-getRedis-setRedis-get10203040500200004000060000Highcharts.com

**许可协议**

SSDB 采用 [New BSD License](http://opensource.org/licenses/BSD-3-Clause) 许可协议, 一个非常宽松灵活的协议.

# ideawu/ssdb: SSDB - A fast NoSQL database, an alternative to Redis

https://github.com/ideawu/ssdb

# SSDB - A fast NoSQL database for storing big list of data

SSDB is a high performace key-value(key-string, key-zset, key-hashmap) NoSQL database, **an alternative to Redis**.

SSDB is stable, production-ready and is widely used by many Internet companies including QIHU 360.

Features

* LevelDB client-server support, written in C/C++
* Designed to store collection data
* Persistent key-value, key-zset, key-map('hashmap') storage
* Redis clients are supported
* Client API supports including C++, PHP, Python, Cpy, Java, nodejs, Ruby, Go([see all](http://ssdb.io/docs/clients.html))
* Persistent queue service
* **Replication(master-slave), load balance**
* GUI administration tool([phpssdbadmin](https://github.com/ssdb/phpssdbadmin))
* Built-in CLI nagios self-checks

PHP client API example

<?php

require\_once('SSDB.php');

$ssdb = new SimpleSSDB('127.0.0.1', 8888);

$resp = $ssdb->set('key', '123');

$resp = $ssdb->get('key');

echo $resp; // output: 123

[More...](http://ssdb.io/docs/php/)

Who's using SSDB?

[](https://camo.githubusercontent.com/c71f02fe298a8ce4141c72d262f36a7370e74b0e/687474703a2f2f737364622e696f2f696d672f737364622d75736572732f616c6c2d737364622d75736572732e6a7067)

[And more...](http://ssdb.io/docs/users.html)

Documentation

* [View online](http://ssdb.io/docs/)
* [Contribute to SSDB documentation project](https://github.com/ideawu/ssdb-docs)

Compile and Install

$ wget --no-check-certificate https://github.com/ideawu/ssdb/archive/master.zip

$ unzip master

$ cd ssdb-master

$ make

$ #optional, install ssdb in /usr/local/ssdb

$ sudo make install

# start master

$ ./ssdb-server ssdb.conf

# or start as daemon

$ ./ssdb-server -d ssdb.conf

# ssdb command line

$ ./tools/ssdb-cli -p 8888

# stop ssdb-server

$ ./ssdb-server ssdb.conf -s stop

# for older version

$ kill `cat ./var/ssdb.pid`

See [Compile and Install wiki](http://ssdb.io/docs/install.html)

Performance

Typical performance

Total 1000 requests.

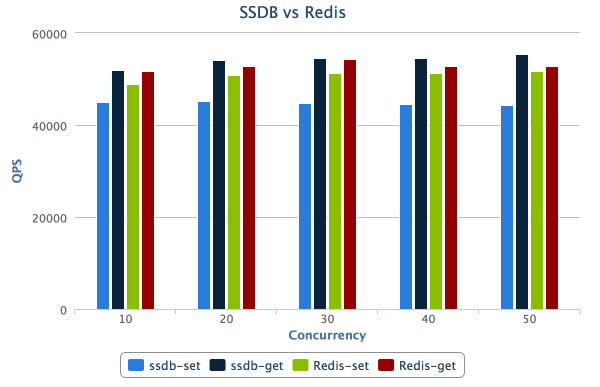
writeseq : 0.546 ms/op 178.7 MB/s

writerand : 0.519 ms/op 188.1 MB/s

readseq : 0.304 ms/op 321.6 MB/s

readrand : 0.310 ms/op 315.0 MB/s

SSDB vs Redis

[](https://camo.githubusercontent.com/2f61d9b6acbfdb713c88791b474a99232bba1355/687474703a2f2f737364622e696f2f737364622d76732d72656469732e706e673f676974687562)

[View full SSDB vs Redis benchmark charts...](http://ssdb.io/)

Concurrency benchmark

========== set ==========

qps: 44251, time: 0.226 s

========== get ==========

qps: 55541, time: 0.180 s

========== del ==========

qps: 46080, time: 0.217 s

========== hset ==========

qps: 42338, time: 0.236 s

========== hget ==========

qps: 55601, time: 0.180 s

========== hdel ==========

qps: 46529, time: 0.215 s

========== zset ==========

qps: 37381, time: 0.268 s

========== zget ==========

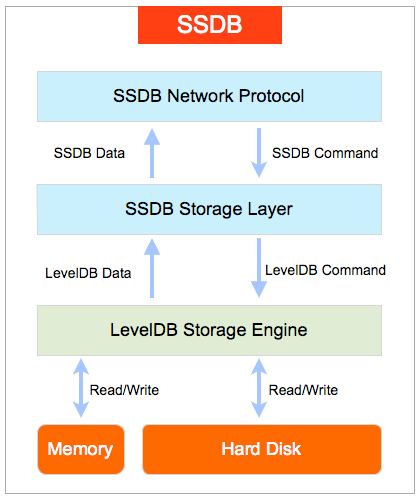
qps: 41455, time: 0.241 s

========== zdel ==========

qps: 38792, time: 0.258 s

Run on a 2013 MacBook Pro 13 inch with Retina display.

Architecture

[](https://camo.githubusercontent.com/6f3243b32deae6f762859f3ee91eeeb36f291518/687474703a2f2f737364622e696f2f737364622e706e67)

Windows executable

Download ssdb-server.exe from here: <https://github.com/ideawu/ssdb-bin>

SSDB library for iOS

make ios

# ls ios/

include/ libleveldb-ios.a libsnappy-ios.a libssdb-ios.a libutil-ios.a

Drag the static libraies files into your iOS project. Then add ios/include to your iOS project's **Header Search Paths**, which is set in **Build Settings**.

Links

* [Author's homepage](http://www.ideawu.com/blog/)
* [Cpy Scripting Language](https://github.com/ideawu/cpy)
* [Google LevelDB](https://code.google.com/p/leveldb/)
* [Lua ssdb client driver for the ngx\_lua](https://github.com/LazyZhu/lua-resty-ssdb)
* [Yet another ssdb client for Python](https://github.com/ifduyue/pyssdb)
* [SSDB 中文文档](http://www.ideawu.net/blog/category/ssdb)

Changes made to LevelDB

See [Changes-Made-to-LevelDB wiki](https://github.com/ideawu/ssdb/wiki/Changes-Made-to-LevelDB)

LICENSE

SSDB is licensed under [New BSD License](http://opensource.org/licenses/BSD-3-Clause), a very flexible license to use.

Authors

@ideawu

Thanks

* 刘建辉, [liujianhui@gongchang.com](mailto:liujianhui@gongchang.com)
* wendal(陈镇铖), [wendal1985@gmail.com](mailto:wendal1985@gmail.com), [http://wendal.net](http://wendal.net/)