1. 常用命令

就DB来说，Redis成绩已经很惊人了，且不说[memcachedb](http://baike.baidu.com/view/3400999.htm)和tokyocabinet之流，就说原版的memcached，速度似乎也只能达到这个级别。Redis根本是使用内存[存储](http://baike.baidu.com/view/87682.htm)，持久化的关键是这三条指令：SAVE BGSAVE LASTSAVE …

**连接操作相关的命令**

* quit：关闭连接（connection）
* auth：简单密码认证

**持久化**

* save：将数据同步保存到磁盘
* bgsave：将数据异步保存到磁盘
* lastsave：返回上次成功将数据保存到磁盘的Unix时戳
* shundown：将数据同步保存到磁盘，然后关闭服务

**远程服务控制**

* info：提供服务器的信息和统计
* monitor：实时转储收到的请求
* slaveof：改变复制策略设置
* config：在运行时配置Redis服务器

**对value操作的命令**

* exists(key)：确认一个key是否存在
* del(key)：删除一个key
* type(key)：返回值的类型
* keys(pattern)：返回满足给定pattern的所有key
* randomkey：随机返回key空间的一个
* keyrename(oldname, newname)：重命名key
* dbsize：返回当前数据库中key的数目
* expire：设定一个key的活动时间（s）
* ttl：获得一个key的活动时间
* select(index)：按索引查询
* move(key, dbindex)：移动当前数据库中的key到dbindex数据库
* flushdb：删除当前选择数据库中的所有key
* flushall：删除所有数据库中的所有key

**对String操作的命令**

* set(key, value)：给数据库中名称为key的string赋予值value
* get(key)：返回数据库中名称为key的string的value
* getset(key, value)：给名称为key的string赋予上一次的value
* mget(key1, key2,…, key N)：返回库中多个string的value
* setnx(key, value)：添加string，名称为key，值为value
* setex(key, time, value)：向库中添加string，设定过期时间time
* mset(key N, value N)：批量设置多个string的值
* msetnx(key N, value N)：如果所有名称为key i的string都不存在
* incr(key)：名称为key的string增1操作
* incrby(key, integer)：名称为key的string增加integer
* decr(key)：名称为key的string减1操作
* decrby(key, integer)：名称为key的string减少integer
* append(key, value)：名称为key的string的值附加value
* substr(key, start, end)：返回名称为key的string的value的子串

**对List操作的命令**

* rpush(key, value)：在名称为key的list尾添加一个值为value的元素
* lpush(key, value)：在名称为key的list头添加一个值为value的 元素
* llen(key)：返回名称为key的list的长度
* lrange(key, start, end)：返回名称为key的list中start至end之间的元素
* ltrim(key, start, end)：截取名称为key的list
* lindex(key, index)：返回名称为key的list中index位置的元素
* lset(key, index, value)：给名称为key的list中index位置的元素赋值
* lrem(key, count, value)：删除count个key的list中值为value的元素
* lpop(key)：返回并删除名称为key的list中的首元素
* rpop(key)：返回并删除名称为key的list中的尾元素
* blpop(key1, key2,… key N, timeout)：lpop命令的block版本。
* brpop(key1, key2,… key N, timeout)：rpop的block版本。
* rpoplpush(srckey, dstkey)：返回并删除名称为srckey的list的尾元素，并将该元素添加到名称为dstkey的list的头部

**对Set操作的命令**

* sadd(key, member)：向名称为key的set中添加元素member
* srem(key, member) ：删除名称为key的set中的元素member
* spop(key) ：随机返回并删除名称为key的set中一个元素
* smove(srckey, dstkey, member) ：移到集合元素
* scard(key) ：返回名称为key的set的基数
* sismember(key, member) ：member是否是名称为key的set的元素
* sinter(key1, key2,…key N) ：求交集
* sinterstore(dstkey, (keys)) ：求交集并将交集保存到dstkey的集合
* sunion(key1, (keys)) ：求并集
* sunionstore(dstkey, (keys)) ：求并集并将并集保存到dstkey的集合
* sdiff(key1, (keys)) ：求差集
* sdiffstore(dstkey, (keys)) ：求差集并将差集保存到dstkey的集合
* smembers(key) ：返回名称为key的set的所有元素
* srandmember(key) ：随机返回名称为key的set的一个元素

**对Hash操作的命令**

* hset(key, field, value)：向名称为key的hash中添加元素field
* hget(key, field)：返回名称为key的hash中field对应的value
* hmget(key, (fields))：返回名称为key的hash中field i对应的value
* hmset(key, (fields))：向名称为key的hash中添加元素field
* hincrby(key, field, integer)：将名称为key的hash中field的value增加integer
* hexists(key, field)：名称为key的hash中是否存在键为field的域
* hdel(key, field)：删除名称为key的hash中键为field的域
* hlen(key)：返回名称为key的hash中元素个数
* hkeys(key)：返回名称为key的hash中所有键
* hvals(key)：返回名称为key的hash中所有键对应的value
* hgetall(key)：返回名称为key的hash中所有的键（field）及其对应的value