# Redis预研报告

## Redis概述

Redis是一个开源的使用ANSI C语言编写、支持网络、基于内存可持久化的日志型、Key-Value数据库，并提供多种语言的API,redis一个key-value的存储系统，广泛应用于缓存方向。

目前，许多web系统都使用Redis作为缓存和数据库。因为Redis的使用简单，操作快速，性能高。在网络访问量大的系统中，Redis具有智能缓存特性，可以有效减少磁盘IO,提升系统性能。

## Redis特点

1. 高性能：Redis的可扩展性和性能非常出众，可以在秒级读写数据，极大地提升了系统吞吐量；
2. 丰富的数据类型：Redis内置了丰富的数据类型，包括String、List、Hash、Set、Sorted Set等。
3. 内存管理：Redis提供了自己专属的内存管理方式，可以控制内存使用程度；
4. 过期数据管理：Redis实现了对数据的自动过期管理；
5. 丰富的API:Redis还提供了丰富的API，用于数据的灵活管理，提升Web应用的可扩展性。
6. 支持数据持久化，包括RDB、AOF两种持久化机制。

## Redis应用场景

1. 实时数据处理

物联网中的传感器会产生大量的实时数据，传统的数据库机制无法对数据进行实时处理和管理。Redis的高速读写能力和内存存储机制，非常适合处理物联网中的实时数据，如温度、湿度、信号等传感器数据实时存储、移动设备位置定时上传等。

1. 设备状态管理

在物联网设备管理中，设备状态信息的记录和查询是非常重要的工作。Redis提供了可靠的数据持久化方式，可以把设备状态数据实时存储到内存中，并将这些数据定期保存到硬盘中，最大程度地保证状态数据的可靠性和稳定性。

1. 设备控制

在物联网中，设备控制通常是通过远程指令进行的，为了实现快速、准确的指令发送和控制，Redis的列表可以被用于创建任务队列，每个任务指令在队列中轮流执行，实现对设备的远程控制。

1. 实时监控和报警

基于Redis的发布/订阅模式，物联网设备可以通过订阅相应主题实现数据实时监控和报警。例如，在温度超出阈值范围时，就可以实时发布温度检测主题，通知相关设备和人员进行处理。

## Redis数据存储方式

Redis数据库的存储方式采用了一种内存和磁盘混合存储的方式，具体如下：

1. 内存存储

Redis的内存存储是基于key-value存储机制的，数据存储在内存中，保证了其高效的读写性能。

1. 磁盘存储

为了避免内存不足的情况下，数据丢失的问题，Redis采取了RDB（Redis Database）和AOF（Append Only File）两种不同的磁盘存储方式。

– RDB: 是一种快照的存储方式，Redis会将当前内存中的数据保存为一个快照文件(.rdb)，该文件包含了整个Redis数据库的全部数据。

– AOF: 是一种追加日志的存储方式，Redis会将所有的写操作追加到磁盘上的日志文件中，该文件包含了这个 Redis实例的所有写操作。

## Redis本地读写速度测试

测试方案

环境： Windows本地；

存储数据类型：key-value，其中key作为通道名，value作为文档字符串(JSON\XML)；

参数：num\_tables(通道数量), 线程池ThreadPool，num\_records（插入数据数量），线程池线程数量ThreadNum, frequency（每一事务提交执行语句的数量，也即采样频率）；

测试任务：开启事务，使用批量插入，同时向1024通道插入数据1000条数据，即1024个通道都要同时插入1000条数据，采样频率1KHZ；

测试指标：每一秒能存（读取）多少数据点，耗时，数据库文件（快照二进制）大小。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 指标  数据量 | 写入总共用时(s) | 每秒能存储数据 | 每个通道每秒存储数据 | 生成数据文件大小  (rdb文件) |
| 1000 | 1.44s | 711111 | 694 | 3.13MB |
| 10000 | 113.435 | 90271 | 88 | 30.4MB |
| 100000 | 2032s | 50393 | 49 | 303.6MB |

|  |  |  |  |
| --- | --- | --- | --- |
| 指标  数据量 | 读取总共用时 | 每秒能读取数据 | 每个通道每秒读取数据 |
| 1000 | 0.05s | 20480000 | 20000 |
| 10000 | 0.15s | 66493506 | 64395 |
| 100000 | 1.21s | 84628099 | 82644 |

数据点间隔跳读测试(多线程)：

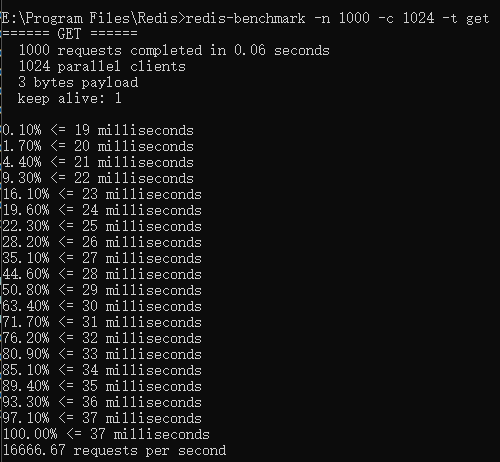
|  |  |
| --- | --- |
| 指标  数据量 | 读取总共用时 |
| 1000 | 0.06s |
| 10000 | 0.28s |
| 100000 | 1.26 s |

评估测试结果：数据量，每个通道达到10000及以上，写入速度大幅度降低，原因有以下几点：

1. Redis默认采用单线程处理写操作，造成每个通道数据量级别大的时候，等待写入的时长增加；
2. Redis采用内存数据库，增加了数据存储容量，造成CPU使用占用过多问题，限制了系统读写性能；
3. Redis 配置开启了持久化（如RDB快照或AOF日志），那么写入操作可能会被频繁的持久化操作所阻塞，从而影响写入速度。持久化操作会涉及磁盘IO，这可能会成为写入性能的瓶颈；

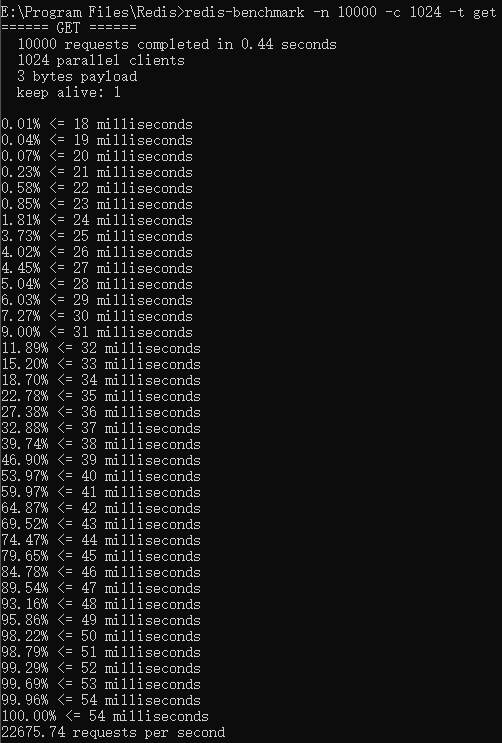
## Redis-benchmark压力测试工具测试

测试方案：模拟1024个客户端，get并发发送1000个请求，写入1000条数据



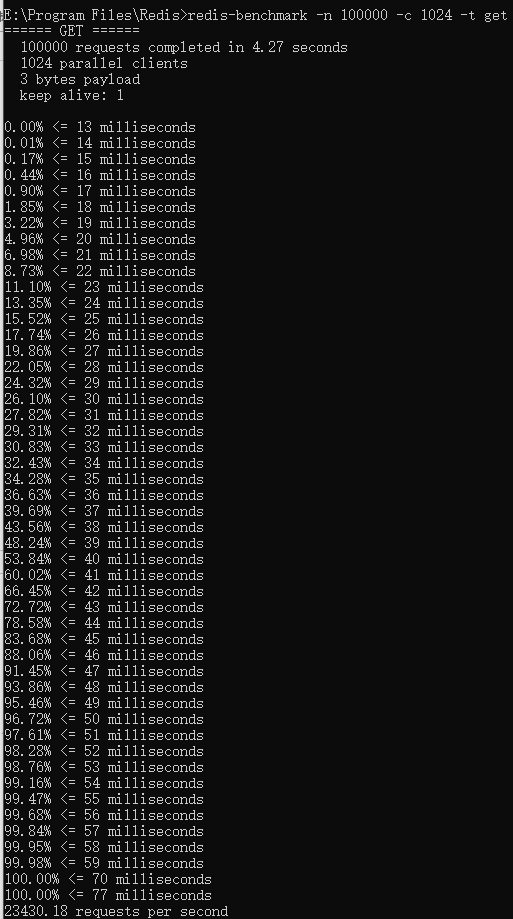
测试结果：0.06s写入了1000条数据，1024个客户端并发，每次只写入3个字节，平均吞吐速度每秒16666.67个请求。

测试方案：模拟1024个客户端，get并发发送10000个请求，写入10000条数据



测试结果：0.44s写入了10000条数据，1024个客户端并发，每次只写入3个字节，平均吞吐速度每秒22675.74个请求。

测试方案：模拟1024个客户端，get并发发送100000个请求，写入100000条数据



测试结果：4.27s写入了100000条数据，1024个客户端并发，每次只写入3个字节，平均吞吐速度每秒23430.18个请求。

## Redis DB数据库结构损坏恢复测试方法

Redis 的数据存储方式基于内存，并且支持持久化，通常有两种持久化方式：RDB（Redis Database Dump）和AOF（Append-Only File）。如果数据结构损坏，恢复的难易程度取决于损坏的程度、选择的持久化方式以及备份策略。

**RDB 持久化方式**： RDB 快照是 Redis 将内存中的数据定期或手动快照到磁盘上的一种方式。如果数据损坏，你可以尝试使用最近的 RDB 快照来进行恢复。恢复过程中，你需要停止 Redis 服务，将备份的 RDB 文件复制回 Redis 数据目录，然后重新启动 Redis 服务。

但是，RDB 快照可能会导致数据丢失，因为最后一次快照之后的数据更改都会丢失。此外，如果 RDB 文件本身损坏，恢复可能会更加困难。

**AOF 持久化方式：** AOF 持久化方式记录了 Redis 执行的每个写操作，以追加的方式写入日志文件。在损坏情况下，你可以尝试修复损坏的 AOF 文件。Redis 提供了一个 aof-check 指令，可以用来检查和修复损坏的 AOF 文件。恢复过程涉及停止 Redis、使用 aof-check 进行修复，然后重新启动 Redis。

类似 RDB，AOF 持久化也可能会导致部分数据丢失，因为损坏的 AOF 文件中的一部分操作可能无法恢复。