

100 万点数据存盘和告警

性能测试报告

一、测试计算机指标

1、子站采用普通计算机:

型号	配置	数量	备注
工作站	<ul style="list-style-type: none"> ● 1.5GHz 双核处理器 ● 2GB 内存*1; ● 500GB硬盘*1 ● 以太网*1 ● 操作系统Windows7 ● 数据库MySQL 	2	工作站

2、主站采用高性能服务器:

型号	配置	数量	备注
服务器	<ul style="list-style-type: none"> ● 2.0GHz 四核处理器 ● 4GB 内存*1; ● 500GB硬盘*1 ● 以太网*2 ● 操作系统Windows7 ● 数据库MySQL 	1	服务器

二、存盘效率测试

1、1000 设备点存盘

The screenshot shows the QToad software interface. The main window displays a list of 1000 data points (var978 to var999) with columns for name, description, unit, IO address, and storage status. Below this, a 'hisdbview' window shows a data table with columns for sequence number, station, data number, time, value, data type, and offset.

序号	站名	数据编号	时间	数值	数据类型	偏移
1	0	0	2016-11-28 19:56:00	0.0	0	19
2	0	0	1,2016-11-28 19:56:00	0.0	0	19
3	0	0	2,2016-11-28 19:56:00	0.0	0	19

990	0	990 2016-11-28 19:56:00	0.0	0	53
991	0	990 2016-11-28 19:56:00	0.0	0	53
992	0	991 2016-11-28 19:56:00	0.0	0	53
993	0	992 2016-11-28 19:56:00	0.0	0	53
994	0	993 2016-11-28 19:56:00	0.0	0	53
995	0	994 2016-11-28 19:56:00	0.0	0	53
996	0	995 2016-11-28 19:56:00	0.0	0	53
997	0	996 2016-11-28 19:56:00	0.0	0	53
998	0	997 2016-11-28 19:56:00	0.0	0	53
999	0	998 2016-11-28 19:56:00	0.0	0	53
1000	0	999 2016-11-28 19:56:00	0.0	0	53

测试时间： 37 毫秒

之前测试的数据

- 2016-11-18 12:19:00 274 至 2016-11-18 12:19:00 337 共计耗时 63ms
- 2016-11-18 12:29:00 410 至 2016-11-18 12:29:00 458 共计耗时 48ms
- 2016-11-18 12:39:00 550 至 2016-11-18 12:39:00 615 共计耗时 65ms
- 2016-11-18 12:49:00 680 至 2016-11-18 12:49:00 727 共计耗时 47ms
- 2016-11-18 12:59:00 818 至 2016-11-18 12:59:00 867 共计耗时 49ms
- 2016-11-18 13:09:00 953 至 2016-11-18 13:09:00 999 共计耗时 46ms
- 2016-11-18 13:19:01 89 至 2016-11-18 13:19:01 135 共计耗时 46ms

2、10000 设备点存盘（单线程，普通机）

测试时间： 781ms

- 2016-11-18 14:40:00 256 至 2016-11-18 14:40:01 37 共计耗时 781ms
- 2016-11-18 14:50:00 393 至 2016-11-18 14:50:01 203 共计耗时 810ms
- 2016-11-18 15:00:00 534 至 2016-11-18 15:00:01 298 共计耗时 763ms

3、100 万设备点存盘（极限测试）：机制，采用单个事物处理机制，一次插入 20000 条数据，进行 50 次数据插入操作，此测试是完全占用 CPU 资源的前提下进行

////////////////////////////////////

```

hostDb.transaction();
QSqlQuery query(hostDb);
QDateTime dt = QDateTime::currentDateTime();
QString stime = dt.toString("yyyy-MM-dd hh:mm:ss");
QString dtime = dt.toString("yyyy-MM-dd hh:mm:ss.zzz");
QDebug()<< dtime + " start insert data";
QString insetStr = "insert into realsave (istation,id,time,value,kind,mesc) values ";
for(int i=1; i<=50; i++){

```

```

        //hostDb.transaction();
        QString sTemp;
        for(int j=1; j<=20000; j++){
            int nValue = i*j;
            sTemp +=
QString("%1,%2,%3','%4',%5,%6,").arg(0).arg(nValue).arg(stime).arg(nValue).arg(0).arg(0);
        }
        QString sssTemp=insetStr + sTemp.left(sTemp.length()-1);
        query.exec(sssTemp);
        //hostDb.commit();
    }
    hostDb.commit();
    dt = QDateTime::currentDateTime() ;
    dtime = dt.toString("yyyy-MM-dd hh:mm:ss.zzz");
    qDebug()<< dtime + " insert data over";
    //////////////////////////////////////

```

工作站性能机：从数据开始插入到插入完成，大概花费大概 30s 的时间

```

sql connect ok!

"2016-12-01 17:11:04.020 start insert data"
"2016-12-01 17:11:42.420 insert data over"

```

服务器性能机：从数据开始到插入完成大概花费 14s 的时间

```

sql connect ok!

"2016-12-01 16:48:52.565 start insert data"
"2016-12-01 16:49:06.433 insert data over"
OK!

```

理论的情况下，工作站性能机在处理 100 万点同时存盘（存储同一张表）的时候大概需要 30s，服务器性能机大概需要 15 秒左右的时间，但实际在使用的过程中这个时间只会更长

三、告警效率

1、 1000 设备点告警效率（普通机，单线程）

24	2016-12-01 18:20:43	var0	越限告警	描述0	越上限
23	2016-12-01 18:20:43	var9999	越限告警	描述9999	越上限
22	2016-12-01 18:20:38	var5000	越限告警	描述5000	越上限
21	2016-12-01 18:19:56	var5000	越限告警	描述5000	越上限
20	2016-12-01 18:19:49	var0	越限告警	描述0	越上限
19	2016-12-01 18:19:49	var9999	越限告警	描述9999	越上限
18	2016-12-01 18:18:11	var0	越限告警	描述0	越上限
17	2016-12-01 18:18:11	var9999	越限告警	描述9999	越上限
16	2016-12-01 18:18:06	var5000	越限告警	描述5000	越上限
15	2016-12-01 18:17:00	var5000	越限告警	描述5000	越上限
14	2016-12-01 18:16:55	var0	越限告警	描述0	越上限
13	2016-12-01 18:16:55	var9999	越限告警	描述9999	越上限
12	2016-12-01 18:16:28	var9999	越限告警	描述9999	越上限
11	2016-12-01 18:16:23	var5000	越限告警	描述5000	越上限
10	2016-12-01 18:16:18	var0	越限告警	描述0	越上限
9	2016-12-01 18:13:55	var0	越限告警	描述0	越上限
8	2016-12-01 18:13:55	var9999	越限告警	描述9999	越上限
7	2016-12-01 18:13:50	var5000	越限告警	描述5000	越上限
6	2016-12-01 18:12:16	var5000	越限告警	描述5000	越上限
5	2016-12-01 18:12:11	var0	越限告警	描述0	越上限
4	2016-12-01 18:12:11	var9999	越限告警	描述9999	越上限

优化处理机制，10000 点的数据报警最快能达到平均 5s 左右的时间

3、100 万点设备点告警效率（多线程、服务器）

根据服务器性能指标，可以采用多线程的处理机制，充分利用 CPU 资源，100 万点设备点告警效率大概最快 1ms，最慢可能会在 1 分钟左右