

本周工作

屠晓涵

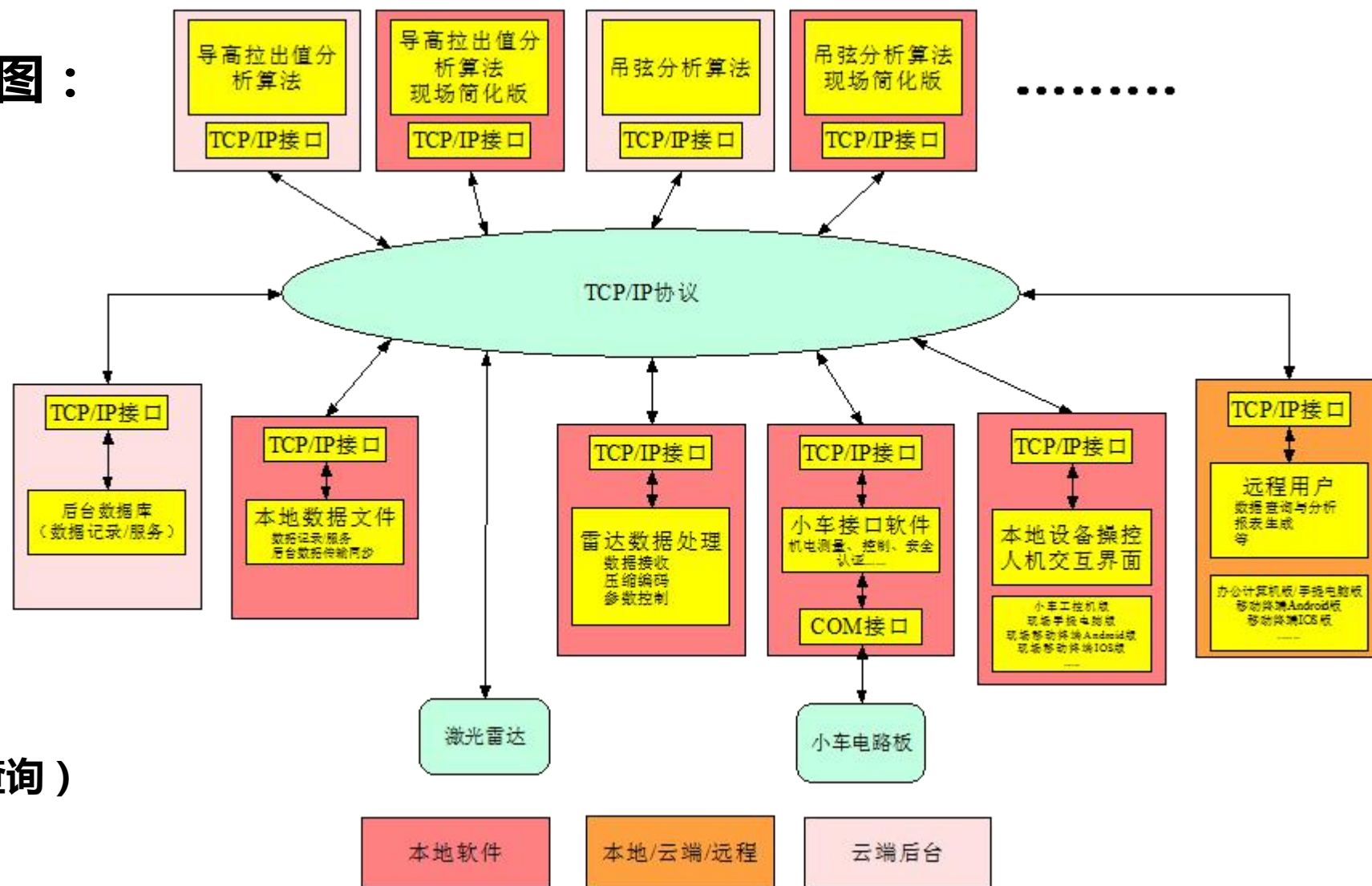
高铁机器人改进版本(V2.0)总结

本地（小车）如右图：

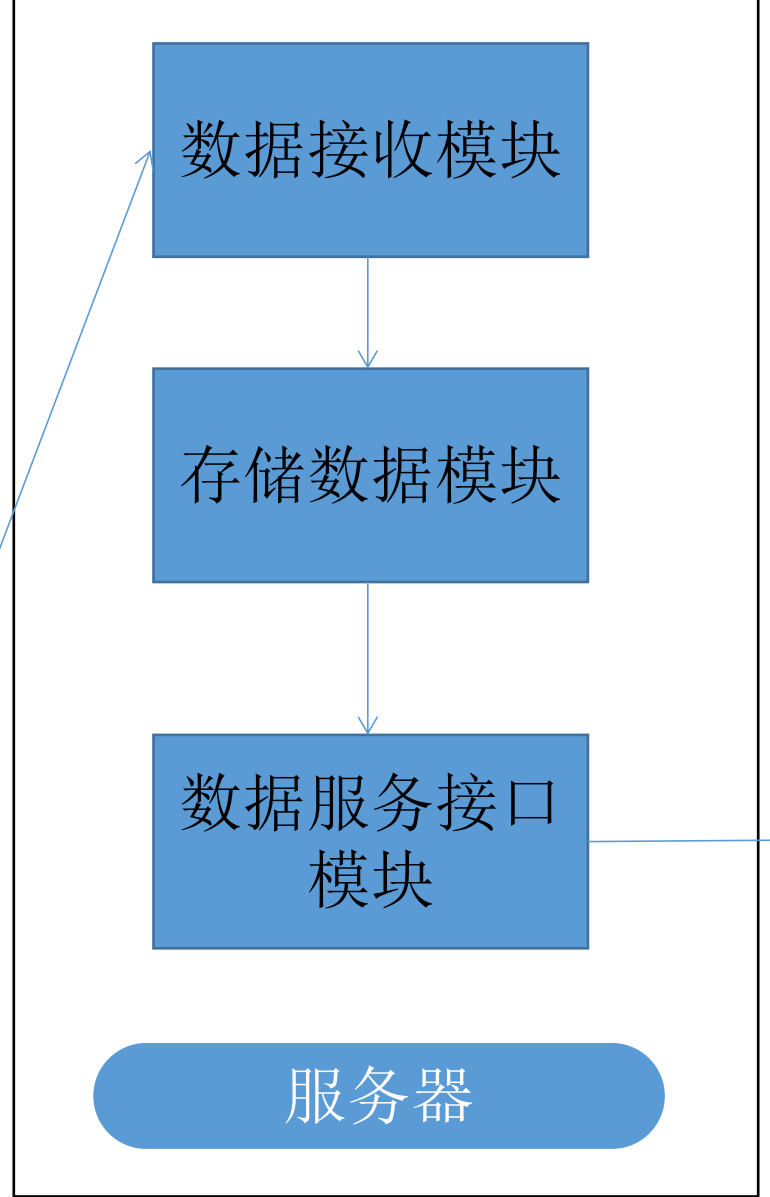
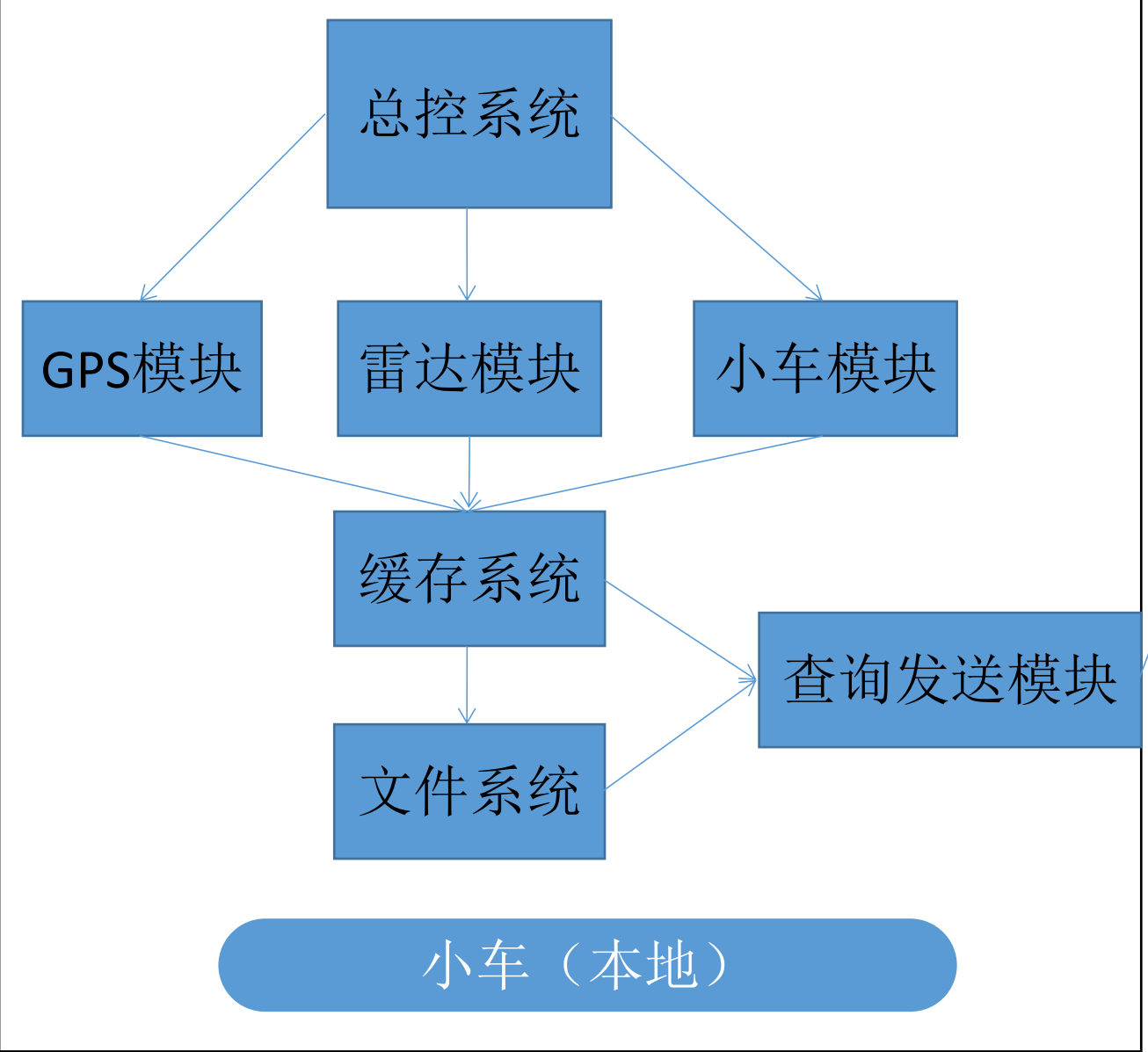
- 总控系统
- GPS模块
- 雷达模块
- 缓存系统
- 文件系统
- 查询发送模块

服务器：

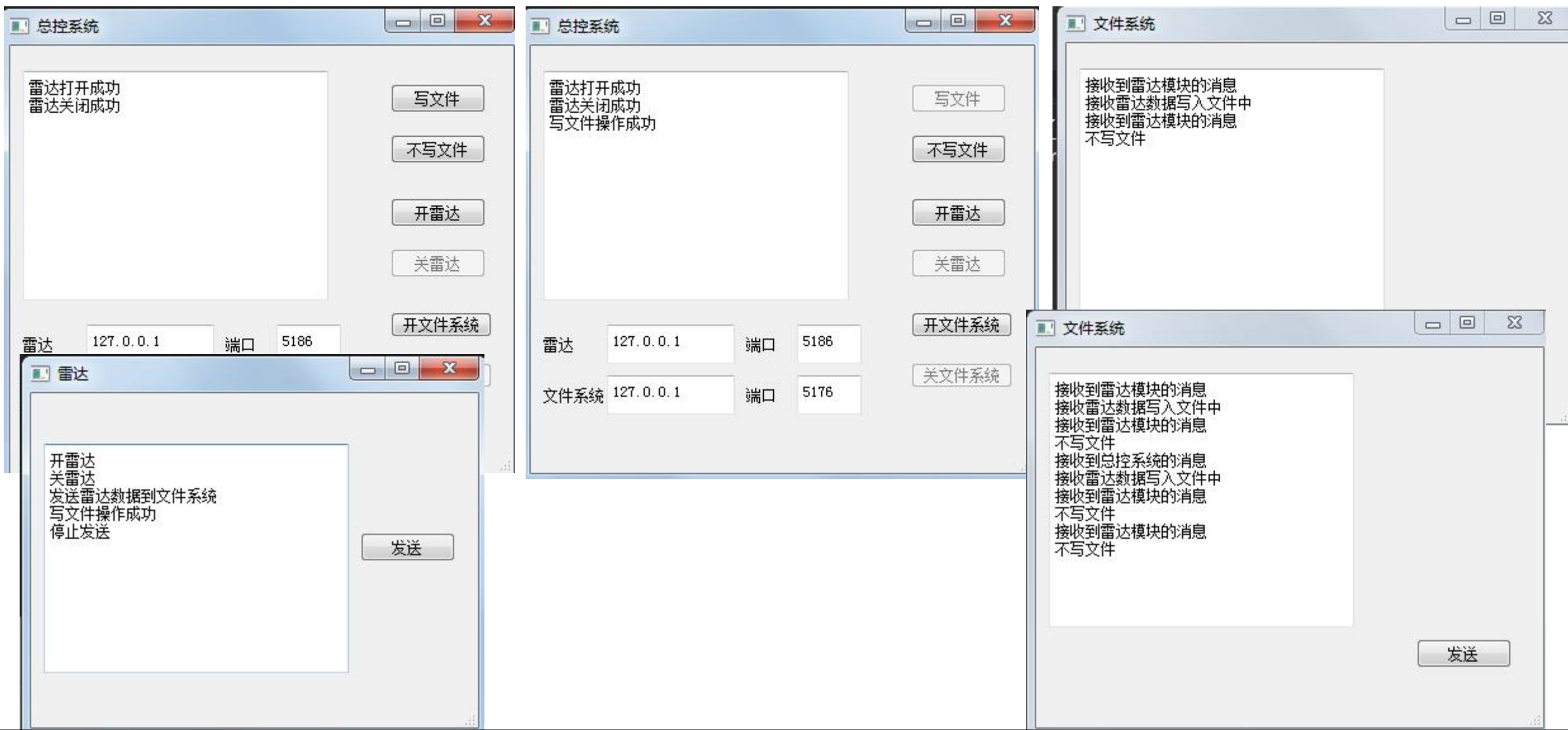
- 数据接收模块
- 存储数据模块
- 数据服务接口模块（查询）



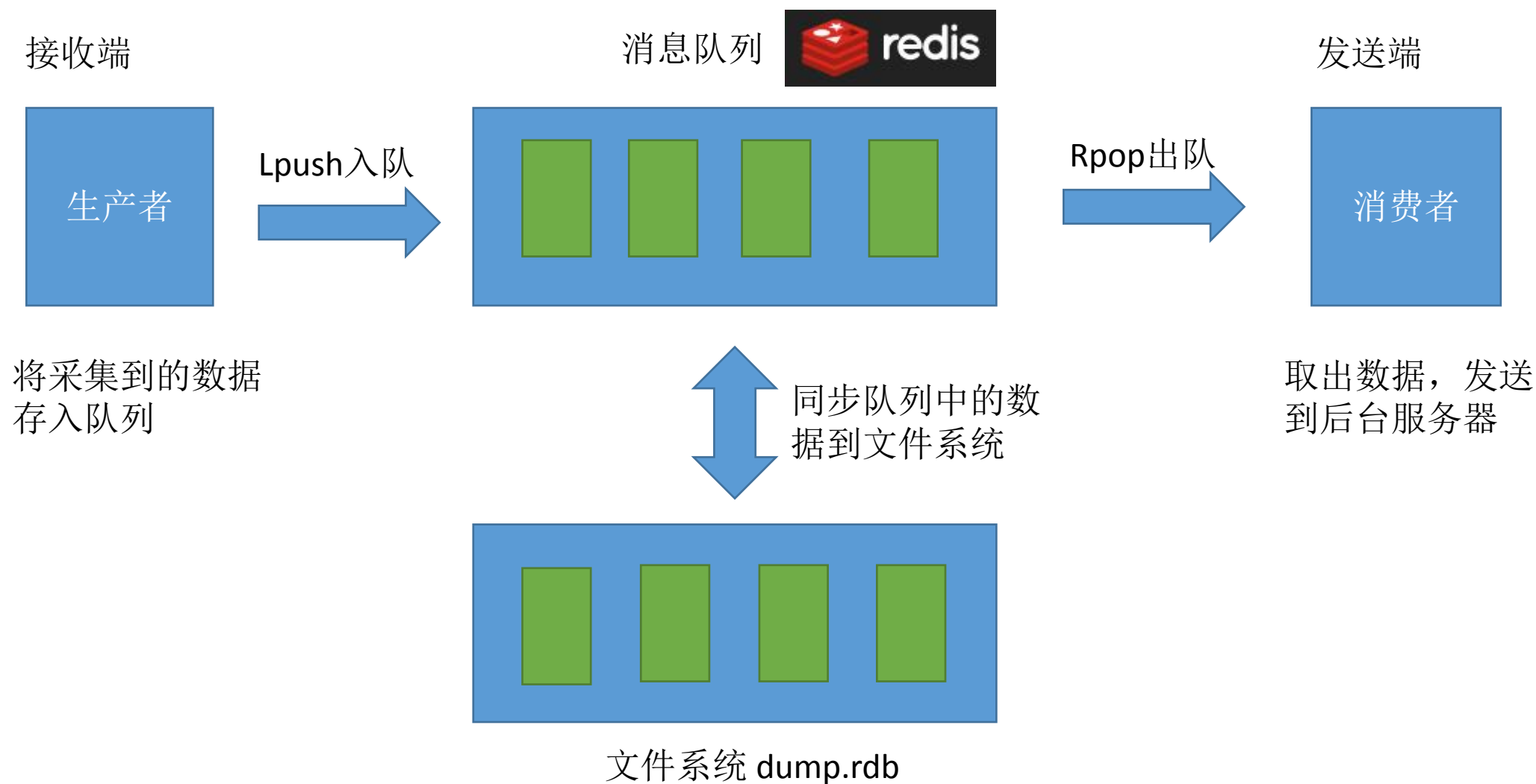
总体设计框架



高铁机器人V2.0实现效果



缓存系统设计过程——生产者消费者模型



测试效果

```
F:\redis\redis64-2.6.12.1\redis-server.exe  
t config. In order to specify a config file use F:\redis\redis64-2.6  
server.exe /path/to/redis.conf  
  
Redis 2.6.12 <00000000/0> 64  
Running in stand alone mode  
Port: 6379  
PID: 8576  
  
http://redis.io  
  
[8576] 19 Oct 14:11:05.067 # Server started, Redis version 2.6.12  
[8576] 19 Oct 14:11:05.083 * DB loaded from disk: 0.016 seconds  
[8576] 19 Oct 14:11:05.083 * The server is now ready to accept connections on po  
rt 6379
```

```
F:\redis\redis64-2.6.12.1\redis-cli.exe  
redis 127.0.0.1:6379> config get *  
1) "dbfilename"  
2) "dump.rdb"  
3) "requirepass"  
4) ""  
5) "masterauth"  
6) ""  
7) "bind"  
8) ""  
9) "unixsocket"  
10) ""  
11) "logfile"  
12) ""  
13) "pidfile"  
14) "/var/run/redis.pid"  
15) "maxmemory"  
16) "0"  
17) "maxmemory-samples"  
18) "3"  
19) "timeout"  
20) "0"  
21) "tcp-keepalive"  
22) "0"  
23) "auto-aof-rewrite-percentage"  
24) "100"
```

测试效果

```
19
20 void qRedisTest::startTest()
21 {
22     redis = new qRedis("localhost",6379);
23     connect(redis, SIGNAL(returnData(qRedis::Reply)), this, SLOT(slotMessage(qRedis::Reply)));
24
25     qRedis::Reply reply;
26
27     if (!redis->openConnection())
28     {
29         qDebug() << "Could not connect to server...";
30         exit(0);
31     }
32
33     qDebug() << "Connected to server...";
34
35     //qDebug() << "AUTH:" << redis->auth("redisZzZ");
36     for(int i=0; i<1000000;i++){
37         QString values = "\\id:5411113430101\\", "\\ filepath\\":\\"/home/bxd/0000.jpg\\", "\\ externsio
38         values.append(i);
39         qDebug()<<"LPUSH1" << redis->|push("bxd",values);
40         qDebug()<<"llen:" << redis->llen("bxd");
41     }
42     // QString s=redis->rpop("bxd");
```

应用程序输出

```
qt-redis-qt-redis
LPUSH1 1676
llen: 1676
LPUSH1 1677
llen: 1677
LPUSH1 1678
llen: 1678
LPUSH1 1679
llen: 1679
LPUSH1 1680
llen: 1680
LPUSH1 1681
llen: 1681
LPUSH1 1682
llen: 1682
LPUSH1 1683
llen: 1683
LPUSH1 1684
```

设置内存为2M时，存储到达上限，不能继续插入数据

1. 配置key置换策略，替换部分key
2. 控制入队和出队频率，避免达到设置的maxmemory

测试效果——地理空间

- 时间、位置的获取
- 127.0.0.1:6379> GEOADD cities 116.404269 39.91582 "beijing"
121.478799 31.235456 "shanghai"
- (integer) 2
- 127.0.0.1:6379> ZRANGE cities 0 -1 WITHSCORES
- 1) "shanghai"
- 2) "4054803475356102"
- 3) "beijing"
- 4) "4069885555377153"

测试效果--地理空间

- 时间、位置的获取
- 127.0.0.1:6379> GEODIST cities beijing shanghai km
- "1068.5677"
- 127.0.0.1:6379> GEOPOS cities beijing shanghai
- 1) "116.40426903963088989"
- 2) "39.91581928642635546"
- 1) "121.47879928350448608"
- 2) "31.23545629441388627"

总结

- 高铁机器人沿着特定轨道行走，重点是数据测量准确：
 - 所有传感器数据缓存到本地数据库，采用生产者消费者模型
 - 测量结果实时反映在机器人屏幕上
 - 服务器永久地存储数据，并提供服务
- 银行服务机器人自主运动，重点是让机器人知道以下两点：
 - 我在哪？——定位
 - 周围环境是什么样的？——建图
 - 我要去哪？——自主导航