1、xxxactivity调用系统的所有接口原计划在baseActivity里面

*//save data to sqlite***public boolean** setData(MsgObject msgObject,**int** operation) **throws** IOException {  
 **return true**;  
}  
*// get data from sqlite***public** List getData(**int** mType){  
 **return null**;  
}  
*//get key value***public** String getKValue(String Key){  
 **return null**;  
}  
*//save key value***public boolean** setKValue(String key, String Value){  
 **return true**;  
}  
*//get file path&name***public** String getFile(**int** FType){  
 **return null**;  
}  
*//get file path***public** String getPath(**int** FType){  
 **return null**;  
}  
*//record file path&name***public boolean** setFile(**int** FType, String file){  
 **return true**;  
}  
  
*//send command to Service***public boolean** setMsg(MsgObject msgObject){  
 Log.*i*(**"test"**,**"setMsg"**);  
 **return mService**.execMsg(msgObject);  
}  
*//get message from service***public** MsgObject getMsg(**int** MType){  
 **return mService**.pickMsg(MType);  
}

界面收到消息时，在如下函数中处理消息。

**public abstract void** onHandleMsg(**int** MsgType);

2、界面点击发送消息到网络的处理过程：

界面将所有消息封装到一个msgObject对象中,调用

setMsg() [baseActivity] -->

execMsg() [smartService] -->

sendProc() [Protocol]-->

protocolSend() [smartProtocol] -->

createPkt(), sendProc() [xxxProtocol] proSendProc()[smartProtocol] -->

SendPkt() [smartService] -->

sendPacket() [NetHandler] -->

msocket.send() [NetHandler]

3、从网络接收一个报文的处理过程：

Receiver( ) [NetHandler] -->

mSocket.receive() [NetHandler] -->

callBack.NetNotify(pkt) [smartService] -->

handler.sendMessage(msg) [smartService] -->

handleMessage(Message msg) [smartService] -->

recvProc(pkt) [Protocol] -->

protocolRecv(pkt)[smartProtocol] -->

proRecvProc(pkt)[smartProtocol] recvProc(pkt) [xxxProtocol] -->

setRet() addData() [smartService] -->

pushMsg() [smartService] -->

sendMsgToActivity() [smartService] -->

sendBroadcast() -->

onReceive() [baseActivity] -->

onHandleMsg( ) [xxxActivity] -->

getData( ) getMsg() [baseActivity] -->

此外，还有状态变化处理，定时器处理，重发处理，文件传输处理等在service中

4、smartService还需要实现的接口：

**public void** addData( MsgObject msgObject)

**public void** modData(**int** sort, **int** no, String path)  
**public void** modData(**int** sort, **int** id, **byte** state)

**public void** delData(**int** type, **int** no)

5、smartService已经实现的接口：

**execMsg(MsgObject msgObject) //界面通知服务**

**NetNotify(PacketMsg pkt) //接收线程通知服务**

**pickMsg(int type) //界面获取信息**

**setRet(...) //通知界面**

**6、APP主要类大致关系图**

XXXActivity

SmartService

XXXActivity

XXXActivity

BaseActivity

sqlite

Application

NetHandler

Protocol

smartProtocol

xxxProtocol

xxxProtocol