通信方法固定，使用协议类型和所传数据来区分功能。

3D报关与报检Unity与winform通讯协议：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 协议编号 | 协议说明 | 发送类型 | 发送端数据  （MessageContent） | 示例 |
| 101 | 点击左列表时，调用，开始任务3D部分 | Winform->unity | GameData中的TaskType枚举 | NetModel nm = new NetModel();  nm.ID = 101;  nm.MessageContent = new ProtoObject(TaskType.进口报检\_出证放行);  SocketManager.SendMsg(nm); |
| 102 | 打开单据，unity流程中调用单据展示 | Unity->Winform | GameData中的InvoiceType枚举 | NetModel nm = new NetModel();  nm.ID = 102;  nm.MessageContent = new ProtoObject(ManagerSystemType.进口报检\_委托报检系统);  SocketManager.SendMsg(nm); |
| 103 |  |  |  |  |
| 104 |  |  |  |  |
| 105 |  |  |  |  |
| 106 |  |  |  |  |
| 104 | 显示委托材料 | Unity->Winform |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| 124 | 显示委托材料 | Unity->Winform |  |  |
| 125 | 审核报检材料 | Unity->Winform |  |  |
| 126 |  |  |  |  |

示例：

1. 发送消息：（102为打开单据消息ID，MessageContent则为单据类型）

NetModel nm = new NetModel();

nm.ID = 102;

nm.MessageContent = new ProtoObject(ManagerSystemType.进口报检\_委托报检系统);

SocketManager.SendMsg(nm);

1. 接收消息：（提前注册号消息ID对应的事件即可）

SocketManager.RegisterMsgHandle(305, () =>

{

Debuge.LogError("unity端自定义的101消息方法体！");

});