

《Unity3D网络游戏实战（第2版）》



[《Unity3D网络游戏实战（第2版）》 下载链接1](#)

著者:罗培羽

出版者:机械工业出版社

出版时间:2019-1-1

装帧:平装

isbn:9787111612179

详解Socket编程，搭建稳健的网络框架；解决网游中常见的卡顿、频繁掉线等问题；探求适宜的实时同步算法。完整的多人对战游戏案例，揭秘登录注册、游戏大厅、战斗系统等模块的实现细节。

想要制作当今热门的网络游戏，特别是开发手机网络游戏，或者想要到游戏公司求职，都需要深入了解网络游戏的开发技术。本书分为三大部分，揭示网络游戏开发的细节。

第一部分“扎基础”（1-5章）

介绍TCP网络游戏开发的必备知识。包括TCP异步连接、多路复用技术；处理粘包分包的方法；设置正确的网络参数、完整的发送数据、安全的关闭连接等方法。第三章《实践出真知：大乱斗游戏》会介绍一款简单网络游戏开发的全过程。

第二部分 “搭框架” （6-7章）

介绍了商业级客户端网络框架的实现方法。这套框架具有较高的通用性，解决了网络游戏开发中常遇到的问题，且达到极致的性能要求，可以运用在多种游戏上。书中还实现了一套c#服务端框架，使用select多路复用，做到底层与逻辑分离，具有消息分发，事件处理等模块。

第三部分 “做游戏” （8-12章）

通过一个完整的实例，讲解网络游戏的设计思路。包括游戏实体的类设计、怎样组织代码、怎样实现游戏大厅（房间系统）、怎样实现实体的同步。这一部分会使用第二部分搭好的框架，一步步做出完整的游戏。

作者介绍:

罗培羽，正在创作好玩游戏的程序员。参与多个项目研发工作，拥有丰富的实战经验。

自2009年发布第一部视频教程《教你用vb制作rpg游戏》以来，先后发布和出版《教你制作3DRPG游戏》、《手把手教你用c#制作rpg游戏》、《Unity3D网络游戏实战（第1版）》等教材；目前关注手机游戏、AI技术等领域；还采访和记录一些普通开发者的心路历程。

目录: 前言

第1章 网络游戏的开端：Echo 1

1.1 藏在幕后的服务端 1

1.2 网络连接的端点：Socket 3

1.2.1 Socket 3

1.2.2 IP地址 3

1.2.3 端口 4

1.2.4 Socket通信的流程 6

1.2.5 TCP和UDP协议 7

1.3 开始网络编程：Echo 8

1.3.1 什么是Echo程序 8

1.3.2 编写客户端程序 8

1.3.3 客户端代码知识点 10

1.3.4 完成客户端 11

1.3.5 创建服务端程序 12

1.3.6 编写服务端程序 14

1.3.7 服务端知识点 15

1.3.8 测试Echo程序 15

1.4 更多API 16

1.5 公网和局域网 17

第2章 分身有术：异步和多路复用 19

2.1 什么样的代码是异步代码 19

2.2 异步客户端 20

2.2.1 异步Connect 21

2.2.2 Show Me The Code 22

2.2.3 异步Receive 23

2.2.4 异步Send	26
2.3 异步服务端	29
2.3.1 管理客户端	29
2.3.2 异步Accept	30
2.3.3 程序结构	31
2.3.4 代码展示	31
2.4 实践：做个聊天室	35
2.4.1 服务端	35
2.4.2 客户端	35
2.4.3 测试	36
2.5 状态检测Poll	36
2.5.1 什么是Poll	36
2.5.2 Poll客户端	37
2.5.3 Poll服务端	38
2.6 多路复用Select	41
2.6.1 什么是多路复用	41
2.6.2 Select服务端	42
2.6.3 Select客户端	44
第3章 实践出真知：大乱斗游戏	45
3.1 什么是大乱斗游戏	45
3.2 搭建场景	46
3.3 角色类Human	49
3.3.1 类结构设计	49
3.3.2 BaseHuman	49
3.3.3 角色预设	51
3.3.4 CtrlHuman	54
3.3.5 SyncHuman	57
3.4 如何使用网络模块	57
3.4.1 委托	57
3.4.2 通信协议	59
3.4.3 消息队列	60
3.4.4 NetManager类	60
3.4.5 测试网络模块	64
3.5 进入游戏：Enter协议	66
3.5.1 创建角色	67
3.5.2 接收Enter协议	70
3.5.3 测试Enter协议	70
3.6 服务端如何处理消息	72
3.6.1 反射机制	72
3.6.2 消息处理函数	73
3.6.3 事件处理	74
3.6.4 玩家数据	76
3.6.5 处理Enter协议	77
3.7 玩家列表：List协议	77
3.7.1 客户端处理	78
3.7.2 服务端处理	79
3.7.3 测试	79
3.8 移动同步：Move协议	80
3.8.1 客户端处理	80
3.8.2 服务端处理	81
3.8.3 测试	81
3.9 玩家离开：Leave协议	82
3.9.1 客户端处理	82
3.9.2 服务端处理	82

- 3.9.3 测试 82
- 3.10 攻击动作：Attack协议 83
 - 3.10.1 播放攻击动作 83
 - 3.10.2 客户端处理 87
 - 3.10.3 服务端处理 88
 - 3.10.4 测试 88
- 3.11 攻击伤害：Hit协议 89
 - 3.11.1 客户端处理 89
 - 3.11.2 服务端处理 91
- 3.12 角色死亡：Die协议 91
 - 3.12.1 客户端处理 91
 - 3.12.2 测试 92
- 第4章 正确收发数据流 94
 - 4.1 TCP数据流 94
 - 4.1.1 系统缓冲区 94
 - 4.1.2 粘包半包现象 96
 - 4.1.3 人工重现粘包现象 97
 - 4.2 解决粘包问题的方法 97
 - 4.2.1 长度信息法 97
 - 4.2.2 固定长度法 98
 - 4.2.3 结束符号法 98
 - 4.3 解决粘包的代码实现 99
 - 4.3.1 发送数据 99
 - 4.3.2 接收数据 99
 - 4.3.3 处理数据 101
 - 4.3.4 完整的示例 104
 - 4.3.5 测试程序 106
 - 4.4 大端小端问题 109
 - 4.4.1 为什么会有大端小端之分 110
 - 4.4.2 使用Reverse()兼容大小端编码 111
 - 4.4.3 手动还原数值 111
 - 4.5 完整发送数据 112
 - 4.5.1 不完整发送示例 113
 - 4.5.2 如何解决发送不完整问题 113
 - 4.5.3 ByteArray和Queue 117
 - 4.5.4 解决线程冲突 120
 - 4.5.5 为什么要使用队列 121
 - 4.6 高效的接收数据 122
 - 4.6.1 不足之处 122
 - 4.6.2 完整的ByteArray 123
 - 4.6.3 将ByteArray应用到异步程序 129
- 第5章 深入了解TCP，解决暗藏问题 133
 - 5.1 从TCP到铜线 133
 - 5.1.1 应用层 133
 - 5.1.2 传输层 134
 - 5.1.3 网络层 135
 - 5.1.4 网络接口 135
 - 5.2 数据传输流程 136
 - 5.2.1 TCP连接的建立 136
 - 5.2.2 TCP的数据传输 137
 - 5.2.3 TCP连接的终止 138
 - 5.3 常用TCP参数 138
 - 5.3.1 ReceiveBufferSize 138
 - 5.3.2 SendBufferSize 139

- 5.3.3 NoDelay 139
- 5.3.4 TTL 140
- 5.3.5 ReuseAddress 141
- 5.3.6 LingerState 142
- 5.4 Close的恰当时机 144
- 5.5 异常处理 146
- 5.6 心跳机制 147
- 第6章 通用客户端网络模块 148
 - 6.1 网络模块设计 148
 - 6.1.1 对外接口 148
 - 6.1.2 内部设计 149
 - 6.2 网络事件 150
 - 6.2.1 事件类型 151
 - 6.2.2 监听列表 151
 - 6.2.3 分发事件 152
 - 6.3 连接服务端 152
 - 6.3.1 Connect 152
 - 6.3.2 ConnectCallback 154
 - 6.3.3 测试程序 155
 - 6.4 关闭连接 156
 - 6.4.1 isClosing? 157
 - 6.4.2 Close 157
 - 6.4.3 测试 158
 - 6.5 Json协议 158
 - 6.5.1 为什么会有协议类 158
 - 6.5.2 使用JsonUtility 159
 - 6.5.3 协议格式 160
 - 6.5.4 协议文件 161
 - 6.5.5 协议体的编码解码 162
 - 6.5.6 协议名的编码解码 163
 - 6.6 发送数据 165
 - 6.6.1 Send 165
 - 6.6.2 SendCallback 166
 - 6.6.3 测试 167
 - 6.7 消息事件 168
 - 6.8 接收数据 170
 - 6.8.1 新的成员 171
 - 6.8.2 ConnectCallback 171
 - 6.8.3 ReceiveCallback 172
 - 6.8.4 OnReceiveData 172
 - 6.8.5 Update 174
 - 6.8.6 测试 175
 - 6.9 心跳机制 176
 - 6.9.1 PING和PONG协议 176
 - 6.9.2 成员变量 177
 - 6.9.3 发送PING协议 178
 - 6.9.4 监听PONG协议 178
 - 6.9.5 测试 179
 - 6.10 Protobuf协议 179
 - 6.10.1 什么是Protobuf 179
 - 6.10.2 编写proto文件 180
 - 6.10.3 生成协议类 181
 - 6.10.4 导入protobuf-net.dll 183
 - 6.10.5 编码解码 183

- 第7章 通用服务端框架 186
 - 7.1 服务端架构 187
 - 7.1.1 总体架构 187
 - 7.1.2 模块划分 187
 - 7.1.3 游戏流程 188
 - 7.2 Json编码解码 189
 - 7.2.1 添加协议文件 189
 - 7.2.2 引用System.web.Extensions 190
 - 7.2.3 修改MsgBase类 191
 - 7.2.4 测试 192
 - 7.3 网络模块 193
 - 7.3.1 整体结构 193
 - 7.3.2 ClientState 194
 - 7.3.3 开启监听和多路复用 194
 - 7.3.4 处理监听消息 196
 - 7.3.5 处理客户端消息 197
 - 7.3.6 关闭连接 198
 - 7.3.7 处理协议 199
 - 7.3.8 Timer 200
 - 7.3.9 发送协议 201
 - 7.3.10 测试 202
 - 7.4 心跳机制 204
 - 7.4.1 lastPingTime? 204
 - 7.4.2 时间戳 204
 - 7.4.3 回应MsgPing协议 205
 - 7.4.4 超时处理 205
 - 7.4.5 测试程序 206
 - 7.5 玩家的数据结构 206
 - 7.5.1 完整的ClientState 206
 - 7.5.2 PlayerData 208
 - 7.5.3 Player 208
 - 7.5.4 PlayerManager 209
 - 7.6 配置MySQL数据库 211
 - 7.6.1 安装并启动MySQL数据库 211
 - 7.6.2 安装Navicat for MySQL 213
 - 7.6.3 配置数据表 214
 - 7.6.4 安装connector 215
 - 7.6.5 MySQL基础知识 216
 - 7.7 数据库模块 217
 - 7.7.1 连接数据库 218
 - 7.7.2 防止SQL注入 220
 - 7.7.3 IsAccountExist 220
 - 7.7.4 Register 221
 - 7.7.5 CreatePlayer 223
 - 7.7.6 CheckPassword 224
 - 7.7.7 GetPlayerData 225
 - 7.7.8 UpdatePlayerData 226
 - 7.8 登录注册功能 226
 - 7.8.1 注册登录协议 227
 - 7.8.2 记事本协议 228
 - 7.8.3 注册功能 229
 - 7.8.4 登录功能 229
 - 7.8.5 退出功能 231
 - 7.8.6 获取文本功能 231

7.8.7 保存文本功能	231
7.8.8 客户端界面	232
7.8.9 客户端监听	233
7.8.10 客户端注册功能	235
7.8.11 客户端登录功能	235
7.8.12 客户端记事本功能	236
7.8.13 测试	236
第8章 完整大项目《坦克大战》	239
8.1 《坦克大战》游戏功能	239
8.1.1 登录注册	239
8.1.2 房间系统	240
8.1.3 战斗系统	241
8.2 坦克模型	242
8.2.1 导入模型	242
8.2.2 模型结构	243
8.3 资源管理器	245
8.3.1 设计构想	245
8.3.2 代码实现	245
8.3.3 测试	246
8.4 坦克类	246
8.4.1 设计构想	246
8.4.2 代码实现	246
8.4.3 测试	247
8.5 行走控制	248
8.5.1 速度参数	249
8.5.2 移动控制	249
8.5.3 测试	250
8.5.4 走在地形上	251
8.6 坦克爬坡	253
8.6.1 Unity的物理系统	253
8.6.2 添加物理组件	253
8.6.3 测试	254
8.7 相机跟随	255
8.7.1 功能需求	255
8.7.2 数学原理	255
8.7.3 编写代码	257
8.7.4 测试	258
8.8 旋转炮塔	260
8.8.1 炮塔元素	260
8.8.2 旋转控制	261
8.8.3 测试	262
8.9 发射炮弹	262
8.9.1 制作炮弹预设	262
8.9.2 制作爆炸效果	263
8.9.3 炮弹组件	264
8.9.4 坦克开炮	266
8.9.5 测试	268
8.10 摧毁敌人	269
8.10.1 坦克的生命值	269
8.10.2 焚烧特效	270
8.10.3 坦克被击中处理	271
8.10.4 炮弹的攻击处理	272
8.10.5 测试	272
第9章 UI界面模块	274

- 9.1 界面模块的设计 274
 - 9.1.1 简单的界面调用 274
 - 9.1.2 通用界面模块 275
- 9.2 场景结构 277
- 9.3 面板基类BasePanel 278
 - 9.3.1 设计要点 278
 - 9.3.2 代码实现 278
 - 9.3.3 知识点 279
- 9.4 界面管理器PanelManager 281
 - 9.4.1 层级管理 281
 - 9.4.2 打开面板 282
 - 9.4.3 关闭面板 284
- 9.5 登录面板LoginPanel 284
 - 9.5.1 导入资源 284
 - 9.5.2 UI组件 286
 - 9.5.3 制作面板预设 286
 - 9.5.4 登录面板类 289
 - 9.5.5 打开面板 289
 - 9.5.6 引用UI组件 290
 - 9.5.7 网络监听 291
 - 9.5.8 登录和注册按钮 293
 - 9.5.9 收到登录协议 293
- 9.6 注册面板RegisterPanel 294
 - 9.6.1 制作面板预设 294
 - 9.6.2 注册面板类 296
 - 9.6.3 按钮事件 298
 - 9.6.4 收到注册协议 299
- 9.7 提示面板TipPanel 300
 - 9.7.1 制作面板预设 300
 - 9.7.2 提示面板类 301
 - 9.7.3 测试面板 302
- 9.8 游戏入口GameMain 303
 - 9.8.1 设计要点 303
 - 9.8.2 代码实现 304
 - 9.8.3 缓存用户名 305
- 9.9 功能测试 306
 - 9.9.1 登录 306
 - 9.9.2 注册 307
 - 9.9.3 下线 309
- 第10章 游戏大厅和房间 310
 - 10.1 列表面板预设 311
 - 10.1.1 整体结构 311
 - 10.1.2 个人信息栏 312
 - 10.1.3 操作栏 312
 - 10.1.4 房间列表栏 313
 - 10.1.5 Scroll View 315
 - 10.1.6 列表项Room 316
 - 10.2 房间面板预设 318
 - 10.2.1 整体结构 318
 - 10.2.2 列表栏 319
 - 10.2.3 列表项Player 320
 - 10.2.4 控制栏 322
 - 10.3 协议设计 322
 - 10.3.1 查询战绩MsgGetAchieve协议 323

- 10.3.2 查询房间列表MsgGetRoom--List协议 323
- 10.3.3 创建房间MsgCreateRoom协议 324
- 10.3.4 进入房间MsgEnterRoom协议 324
- 10.3.5 查询房间信息MsgGetRoom-Info协议 324
- 10.3.6 退出房间MsgLeaveRoom协议 325
- 10.3.7 开始战斗MsgStartBattle协议 325
- 10.4 列表面板逻辑 326
 - 10.4.1 面板类 326
 - 10.4.2 获取部件 326
 - 10.4.3 网络监听 328
 - 10.4.4 刷新战绩 329
 - 10.4.5 刷新房间列表 329
 - 10.4.6 加入房间 330
 - 10.4.7 创建房间 331
 - 10.4.8 刷新按钮 332
- 10.5 房间面板逻辑 332
 - 10.5.1 面板类 332
 - 10.5.2 获取部件 333
 - 10.5.3 网络监听 334
 - 10.5.4 刷新玩家列表 334
 - 10.5.5 退出房间 336
 - 10.5.6 开始战斗 336
- 10.6 打开列表面板 337
- 10.7 服务端玩家数据 337
 - 10.7.1 存储数据 338
 - 10.7.2 临时数据 338
- 10.8 服务端房间类 339
 - 10.8.1 管理器和房间类的关系 339
 - 10.8.2 房间类的设计要点 340
 - 10.8.3 添加玩家 341
 - 10.8.4 选择阵营 343
 - 10.8.5 删除玩家 343
 - 10.8.6 选择新房主 345
 - 10.8.7 广播消息 345
 - 10.8.8 生成房间信息 345
- 10.9 服务端房间管理器 347
 - 10.9.1 数据结构 347
 - 10.9.2 获取房间 347
 - 10.9.3 添加房间 348
 - 10.9.4 删除房间 348
 - 10.9.5 生成列表信息 348
- 10.10 服务端消息处理 349
 - 10.10.1 查询战绩MsgGet-Achieve 350
 - 10.10.2 查询房间列表MsgGetRoom-List 350
 - 10.10.3 创建房间MsgCreate-Room 351
 - 10.10.4 进入房间MsgEnterRoom 351
 - 10.10.5 查询房间信息MsgGet-RoomInfo 352
 - 10.10.6 离开房间MsgLeave-Room 352
- 10.11 玩家事件处理 353
- 10.12 测试 354
- 第11章 战斗和胜负判定 358
 - 11.1 协议设计 358
 - 11.1.1 进入战斗MsgEnterBattle 359
 - 11.1.2 战斗结果MsgBattleResult 359

11.1.3 退出战斗MsgLeaveBattle 360

11.2 坦克 360

11.2.1 不同阵营的坦克预设 360

11.2.2 战斗模块 361

11.2.3 同步坦克SyncTank 362

11.2.4 坦克的属性 362

11.3 战斗管理器 363

11.3.1 设计要点 363

11.3.2 管理器类 363

11.3.3 坦克管理 364

11.3.4 重置战场 365

11.3.5 开始战斗 366

11.3.6 产生坦克 367

11.3.7 战斗结束 369

11.3.8 玩家离开 369

11.4 战斗结果面板 369

11.4.1 面板预设 369

11.4.2 面板逻辑 371

11.5 服务端开启战斗 373

11.5.1 能否开始战斗 373

11.5.2 定义出生点 373

11.5.3 坦克信息 376

11.5.4 开启战斗 377

11.5.5 消息处理 377

11.6 服务端胜负判断 378

11.6.1 是否死亡 379

11.6.2 胜负决断函数 379

11.6.3 定时器 380

11.6.4 Room::Update 380

11.7 服务端断线处理 381

11.8 测试 382

11.8.1 进入战场 382

11.8.2 离开战场 384

第12章 同步战斗信息 386

12.1 同步理论 387

12.1.1 同步的过程 387

12.1.2 同步的难题 387

12.2 状态同步 389

12.2.1 直接状态同步 389

12.2.2 跟随算法 390

12.2.3 预测算法 390

12.3 帧同步 391

12.3.1 指令同步 391

12.3.2 从Update说起 392

12.3.3 什么是同步帧 393

12.3.4 指令 394

12.3.5 指令的执行 394

12.4 协议设计 395

12.4.1 位置同步MsgSyncTank 396

12.4.2 开火MsgFire 396

12.4.3 击中MsgHit 397

12.5 发送同步信息 397

12.5.1 发送位置信息 397

12.5.2 发送开火信息 398

12.5.3 发送击中信息 399
12.6 处理同步信息 400
12.6.1 协议监听 400
12.6.2 OnMsgSyncTank 401
12.6.3 OnMsgFire 401
12.6.4 OnMsgHit 402
12.7 同步坦克SyncTank 402
12.7.1 预测算法的成员变量 402
12.7.2 移动到预测位置 403
12.7.3 初始化 404
12.7.4 更新预测位置 405
12.7.5 炮弹同步 406
12.8 服务端消息处理 406
12.8.1 位置同步MsgSyncTank 407
12.8.2 开火MsgFire 408
12.8.3 击中MsgHit 409
12.8.4 调试 410
12.9 完善细节 412
12.9.1 滚动的轮子和履带 412
12.9.2 灵活操作 413
12.9.3 准心 413
12.9.4 自动瞄准 414
12.9.5 界面和场景优化 414
12.9.6 战斗面板 415
12.9.7 击杀提示 416
12.10 结语 416
• • • • • ([收起](#))

[《Unity3D网络游戏实战（第2版）》 下载链接1](#)

标签

游戏开发

网络编程

游戏编程

编程

unity3D

游戏

game

Unity3D

评论

体验不好，资源也老。新版都取消自带了。

实例有价值，网游的基础开发讲得还算细致。
章节编排有的显得比较冗余，可以称得上啰嗦了；很多图片不知所云。

作者引导着编写了一个Socket网络库，并用其制作了一个坦克大战项目。市面上讲游戏服务器开发的书籍不多，这本书让我打了些基础。四星的原因是内容仍然太啰嗦，第九章的UI代码完全没有存在的必要，还有其他的UI逻辑、C#语法等等，都可以考虑让读者通过脚注或者读源码来学习，我希望这本书能更纯粹，能专注于一些基础教程不会提及的内容。

[《Unity3D网络游戏实战（第2版）》_下载链接1](#)

书评

[《Unity3D网络游戏实战（第2版）》_下载链接1](#)