



# Delphi Web前端开发教程

## 基于TMS WEB Core框架



## 目录

Delphi WEB开发框架简介 .....	23
Delphi开发的部分应用软件App展示 .....	25
Delphi/Object Pascal的历史与现状 .....	40
Delphi的杀手锏是什么? .....	40
Wirth教授的Pascal时期 .....	41
Broland Turbo Pascal时期 .....	93
Delphi Object Pascal早期 .....	93
从CodeGear到Embarcadero的Object Pascal时期 .....	93
Delphi支持手机移动开发 .....	93
Delphi 10. x时期 .....	93
Delphi 11发布 .....	93
Delphi WEB开发三剑客IntraWEB、uniGUI、TMS WEB Core .....	93
WEB开发基础知识入门 .....	93
Delphi第三方WEB开发框架 .....	93
各个WEB框架特点比较 .....	93
Delphi业界领先的WEB网络框架 .....	93
重点技能和组件 .....	93
轻松部署分发 .....	93
范围和架构 .....	55
快速入门 .....	31
配置TMS WEB Core 设置 .....	61
配置 TMS WEB Core 项目设置 .....	63
自动化版本管理 .....	65
Lazarus安装TMS WEB Core .....	68
创建TMS WEB Core 网络WEB网络应用程序 .....	71
创建TMS WEB Core 渐进式WEB网页应用程序 .....	73
创建TMS WEB Core Electron 桌面应用程序 .....	76
创建TMS WEB Miletus 桌面应用程序 .....	78
调试 .....	81
Pascal转译JavaScript编译器Pas2JS .....	84
RTL运行时库 .....	85
预处理器 .....	86
命令行编译 .....	87
实用功能性函数 .....	91
页面设计 .....	93
绝对定位 .....	93
相对定位 .....	93
主题 .....	95
BiDiMode双向模式 .....	96
使用HTML模板 .....	101
JavaScript 和 CSS .....	101
使用HTML模板 .....	101
自动同步HTML模板 .....	115
实时预览 .....	117
数据库开发 .....	119

支持REST API的传统数据库解决方案 .....	120
支持REST API的云数据库解决方案 .....	121
支持其它编程语言的REST APIs .....	121
Application应用程序对象 .....	122
Forms表单窗体对象 .....	126
在运行期创建表单窗体 .....	127
在其它控件中托管表单窗体 .....	129
属性 .....	130
方法 .....	131
事件 .....	131
Form表单窗体 .....	133
Frames框架 .....	134
UI 控件类型 .....	135
封装HTML元素的UI控件 .....	135
使用HTML5 CANVAS元素自定义绘制控件 .....	135
TMS FNC 控件 .....	135
jQuery UI 控件 .....	135
标准组件 .....	136
可视化控件通用属性 .....	136
可视化控件通用事件 .....	137
TWebLabel标签 .....	138
说明 .....	138
HTML模板标签 .....	138
TWebLabel属性 .....	138
TWebLabel事件 .....	139
TWebButton按钮 .....	139
说明 .....	139
HTML模板标签 .....	140
TWebButton属性 .....	140
TWebButton事件 .....	140
TWebEdit编辑输入框 .....	140
说明 .....	140
HTML模板标签 .....	141
TWebEdit属性 .....	141
TWebEdit方法 .....	142
TWebEdit事件 .....	142
TWebEditAutocomplete自动完成编辑输入框 .....	142
说明 .....	142
HTML模板标签 .....	142
TWebEditAutocomplete属性 .....	143
TWebEditAutocomplete事件 .....	143
TWebSpinEdit箭头按钮数字编辑输入框 .....	144
说明 .....	144
HTML模板标签 .....	144
TWebSpinEdit属性 .....	144
TWebSpinEdit事件 .....	145
TWebMaskEdit掩码编辑输入框 .....	145

说明	145
TWebMaskEdit属性	146
TWebMaskEdit事件	146
TWebDateTimePicker日期时间选择器	146
说明	146
HTML模板标签	146
TWebDateTimePicker属性	147
TWebDateTimePicker事件	147
TWebListBox列表框	147
说明	148
HTML模板标签	148
TWebListBox属性	148
TWebListBox事件	148
TWebComboBox组合下拉框	149
说明	149
HTML模板标签	149
TWebComboBox属性	149
TWebComboBox事件	150
TWebLookupComboBox查询组合下拉框	150
说明	150
HTML模板标签	150
TWebLookupComboBox属性	150
TWebLookupComboBox事件	151
TWebColorPicker颜色选择器	151
说明	151
HTML模板标签	152
TWebColorPicker属性	152
TWebColorPicker事件	152
TWebCheckBox复选框	152
说明	153
TWebCheckBox属性	153
TWebCheckBox事件	153
TWebRadioButton单选按钮	153
说明	153
TWebRadioButton属性	153
TWebRadioButton事件	154
TWebMainMenu主菜单	154
说明	155
HTML模板标签	155
TWebMainMenu属性	155
TWebMainMenu事件	156
TWebPopupMenu弹出式菜单	156
说明	156
HTML模板标签	156
TWebPopupMenu属性	156
TWebPopupMenu方法	157
TWebPopupMenu事件	157

TWebMemo备忘录	157
说明	157
HTML模板标签	158
TWebMemo属性	158
TWebMemo方法	159
TWebMemo事件	158
TWebRadioGroup单选按钮群组	158
说明	158
TwebRadioGroup属性	159
TwebRadioGroup事件	159
TWebCheckGroup复选框群组	159
说明	159
TWebCheckGroup属性	160
TWebCheckGroup事件	160
TWebProgressBar进度条	160
说明	161
HTML模板标签	161
TWebProgressBar属性	161
TWebBadge徽章	161
说明	161
HTML模板标签	161
TWebBadge属性	162
TWebPaintBox绘图框	162
说明	162
TWebPaintBox属性	162
TWebPaintBox事件	162
TWebTrackBar轨迹条	163
说明	164
HTML模板标签	164
TWebTrackBar属性	164
TWebTrackBar事件	164
TWebScrollBar滚动框	164
说明	164
HTML模板标签	164
TWebScrollBar属性	165
TWebScrollBar事件	165
TWebSplitter分隔条	165
说明	165
TWebSplitter属性	166
TWebSplitter事件	166
TWebSignIn登录	166
说明	166
HTML模板标签	166
TWebSignIn属性	166
TWebSignIn方法	167
TWebSignIn事件	167
TWebPanel面板	167

说明	167
HTML模板标签	167
TWebPanel属性	168
TWebPanel事件	168
TWebHTMLContainer容器	168
说明	168
HTML 模板标签	168
TWebHTMLContainer属性	169
TWebHTMLContainer事件	169
TWebHTMLForm表单窗体	169
说明	169
HTML 模板标签	169
TWebHTMLForm属性	170
TWebHTMLDiv元素Div	171
说明	171
HTML 模板标签	171
TWebHTMLDiv属性	171
TWebHTMLSpan元素Span	170
说明	170
HTML 模板标签	171
TWebHTMLSpan属性	171
TWebHTMLAnchor元素锚	171
说明	171
HTML 模板标签	172
TWebHTMLAnchor属性	172
TWebConsoleLog控制台日志	172
说明	172
HTML 模板标签	172
TWebConsoleLog方法	172
TWebConsoleLog属性	172
TWebImageControl图像控件	173
说明	173
HTML 模板标签	173
TWebImageControl属性	173
TWebImageControl方法	173
TWebImageControl事件	174
TWebImageZoomControl图像放大控件	175
说明	175
HTML 模板标签	175
TWebImageControl属性	175
TWebImageControl事件	176
TWebLinkLabel链接标签	176
说明	176
HTML 模板标签	177
TWebLinkLabel属性	177
TWebLinkLabel事件	177
TWebRichEdit富文本编辑器	177

说明.....	177
HTML 模板标签.....	178
TWebRichEdit属性.....	178
TWebRichEdit方法.....	178
TWebRichEdit事件.....	178
TWebSyntaxMemo语法备忘录.....	179
加载一个文件.....	180
下载一个文件.....	181
TWebSyntaxMemo属性.....	181
TWebSyntaxMemo方法.....	182
TWebSyntaxMemo事件.....	183
TWebTabSet标签页.....	183
说明.....	183
HTML 模板标签.....	183
TWebTabSet属性.....	184
TWebTabSet方法.....	184
TWebTabSet事件.....	184
TWebPageControl多页控件.....	184
说明.....	184
HTML 模板标签.....	184
TWebPageControl属性.....	185
TWebPageControl方法.....	185
TwebPageControl事件.....	185
TWebTabsheet标签页.....	185
说明.....	185
TWebLoginPanel登录面板.....	185
说明.....	186
TWebLoginPanel属性.....	186
TWebLoginPanel事件.....	186
TWebRatingPanel评价面板.....	187
说明.....	187
TWebRatingPanel属性.....	187
TWebRatingPanel事件.....	187
TWebChatBox聊天框.....	188
说明.....	188
TWebChatbox属性.....	188
TWebChatbox事件.....	189
TWebChatbox方法.....	189
TMessage属性.....	189
TMessage事件.....	190
TWebSpeedButton快捷按钮.....	190
说明.....	190
HTML 模板标签.....	190
TWebSpeedButton属性.....	190
TWebSpeedButton事件.....	191
TWebPayPal贝宝支付.....	191
说明.....	191

HTML 模板标签	191
TWebPayPal属性	192
TWebPayPal事件	193
TWebToolBar工具栏	194
说明	194
TWebToolBar属性	194
TWebToolBar事件	194
TWebRichEditToolBar富文本编辑工具栏	195
说明	195
TWebRichEditToolBar属性	195
TWebRichEditToolBar事件	195
TWebGridPanel网格面板	195
说明	195
HTML 模板标签	196
TWebGridPanel属性	196
TWebGridPanel事件	197
TWebTreeView树形视图	197
说明	197
HTML 模板标签	197
TWebTreeView属性	197
TWebTreeView方法	198
TWebTreeView事件	198
示例代码	198
TWebAccordion手风琴面板	199
说明	199
HTML 模板标签	199
TWebAccordion属性	200
TWebAccordion事件	200
TAccordionSection属性	200
TWebResponsiveGridPanel响应式网格面板	200
说明	201
HTML 模板标签	201
TWebResponsiveGridPanel属性	201
TWebResponsiveGridPanel方法	201
TWebResponsiveGridPanel事件	201
TResponsiveLayoutItem属性	202
TWebMessageDlg消息对话框	202
说明	202
TWebMessageDlg属性	203
TWebMessageDlg方法	203
TWebMessageDlg事件	203
TWebWaitMessage等待消息框	204
说明	204
TWebWaitMessage属性	204
TWebWaitMessage方法	204
TWebWaitMessage事件	204
TWebFileUpload文件上传控件	205



说明.....	205
TWebFileUpload属性.....	205
HTML 模板标签.....	205
TWebFileUpload事件.....	205
TFile属性.....	206
TFile方法.....	206
TWebFilePicker文件选择器.....	208
说明.....	208
HTML 模板标签.....	208
TWebFilePicker属性.....	208
TWebFilePicker事件.....	208
示例代码.....	209
TWebShare分享.....	209
说明.....	209
TWebShare方法.....	209
TWebOpenDialog打开对话框.....	210
说明.....	210
TWebOpenDialog属性.....	210
TWebOpenDialog方法.....	210
TWebOpenDialog事件.....	211
TWebToast消息提示框.....	211
说明.....	211
TWebToast属性.....	211
TWebToast事件.....	211
TToastItem属性.....	212
TToastItem方法.....	212
TWebToggleButton开关按钮.....	212
说明.....	212
TWebToggleButton属性.....	212
TWebToggleButton事件.....	212
TWebBitBtn图像按钮.....	213
说明.....	213
HTML 模板标签.....	213
TWebBitBtn属性.....	213
TWebBitBtn事件.....	214
TWebGroupBox群组框.....	214
说明.....	214
HTML 模板标签.....	215
TWebGroupBox属性.....	215
TWebGroupBox事件.....	215
TWebStretchPanel弹性面板.....	215
说明.....	215
HTML 模板标签.....	216
TWebStretchPanel属性.....	217
TWebStretchPanel事件.....	217
TWebStringGrid字符串网格.....	217
说明.....	217

HTML 模板标签.....	217
TWebStringGrid属性.....	218
TWebStringGrid方法.....	219
TWebStringGrid事件.....	220
TWebListControl列表控件.....	221
说明.....	221
TListItem属性.....	221
TListItem方法.....	222
TWebListControl属性.....	222
TWebListControl事件.....	223
TWebTableControl表格控件.....	223
说明.....	223
HTML 模板标签.....	224
TWebTableControl属性.....	224
TWebTableControl方法.....	224
TWebTableControl事件.....	224
TWebResponsiveGrid响应式网格.....	225
说明.....	225
HTML 模板标签.....	226
TWebResponsiveGrid属性.....	226
TWebResponsiveGrid方法.....	226
TWebResponsiveGrid事件.....	227
TWebResponsiveGridItem属性.....	228
TWebLoginPanel登录面板.....	228
说明.....	228
TWebLoginPanel属性.....	228
TWebLoginPanel事件.....	229
TWebImageSlider图片轮播.....	229
说明.....	229
TWebImageSlider属性.....	230
TWebImageSlider公共属性.....	231
TWebImageSlider方法.....	231
TWebImageSlider事件.....	231
示例代码.....	231
TWebContinuousScroll持续滚动.....	231
说明.....	231
TWebContinuousScroll属性.....	232
TWebContinuousScroll方法.....	233
TWebContinuousScroll事件.....	233
TWebSignatureCapture签名截图.....	235
说明.....	235
TWebSignatureCapture属性.....	235
TWebSignatureCapture方法.....	235
TWebCalendar日历.....	236
TWebCalendar属性.....	236
TWebCalendar方法.....	237
TWebCalendar事件.....	237

TWebGoogleReCaptcha谷歌验证码	237
说明	237
TWebGoogleReCaptcha属性	237
TWebGoogleReCaptcha方法	237
TWebGoogleReCaptcha事件	237
TWebGoogleDrive谷歌网盘	238
说明	238
HTML 模板标签	238
TWebGoogleDrive属性	238
TWebGoogleMaps谷歌地图	238
说明	239
HTML 模板标签	239
TWebGoogleMaps属性	239
TWebGoogleMaps方法	239
TWebGoogleMaps事件	244
TWebGoogleChart谷歌图表	244
说明	244
TWebGoogleChart属性	245
TWebGoogleChart方法	247
TWebGoogleChart事件	247
示例	248
TWebSentry错误监控哨兵	250
设置步骤	250
首先调出项目设置	251
继续进行演示的其余部分	257
TWebSentry属性	266
TWebSentry方法	266
TWebBrowserControl浏览器控件	267
说明	267
HTML 模板标签	267
TWebBrowserControl属性	267
TWebMultimediaPlayer多媒体播放器	268
说明	268
HTML 模板标签	268
TWebMultimediaPlayer属性	268
TWebMediaCapture多媒体捕捉	269
说明	269
TWebMediaCapture属性	269
TWebMediaCapture方法	269
TWebMediaCapture事件	269
TWebYoutube油管视频网	270
说明	270
HTML 模板标签	270
TWebYoutube属性	270
TWebTwitterFeed推特新网网	271
说明	271
HTML 模板标签	271

TWebTwitterFeed属性	272
TWebCamera摄像头	272
说明	272
选择一个设备	272
自动启动摄像数据流	273
HTML 模板标签	273
TwebCamera属性	274
TWebCamera方法	274
TWebCamera事件	274
数据库感知组件	276
TWebDataSource数据源组件	276
说明	276
TWebDataSource属性	276
TWebClientDataSet客户端数据集组件	277
说明	277
TWebClientDataSet属性	277
TWebClientDataSet方法	277
TWebClientDataSet事件	278
TWebClientConnection客户端连接组件	278
说明	278
TWebClientConnection属性	278
TWebClientConnection事件	279
TWebDBLabel数据库标签	280
说明	280
TWebDBEdit数据库编辑输入框	280
说明	280
TWebDBEditAutoComplete数据库自动完成编辑输入框	280
说明	280
TWebDBCheckBox数据库复选框	280
说明	280
TWebDBSpinEdit数据库箭头按钮编辑输入框	281
说明	281
TWebDBMaskEdit数据库掩码编辑输入框	281
说明	281
TWebDBComboBox数据库组合下拉框	281
说明	281
TWebDBLookupComboBox数据库查询组合下拉框	281
说明	281
TWebDBMemo数据库备忘录	282
说明	282
TWebDBDateTimePicker数据库日期时间选择器	282
说明	282
TWebDBRadioGroup数据库单选按钮群组	282
说明	282
TWebDBLinkLabel数据库链接标签	282
说明	282
TWebDBImageControl数据库图像控件	282

说明.....	283
TWebDBTableControl数据库表格控件.....	283
说明.....	283
TWebDBResponsiveGrid数据库响应式网格.....	283
说明.....	283
TWebDBGrid数据库网格.....	285
说明.....	285
TWebDBNavigator数据库导航器.....	286
说明.....	286
非可视化组件和类.....	287
TWebTimer计时器.....	287
TWebClipboard剪贴板.....	287
TWebBluetooth蓝牙.....	287
TWebBluetooth类.....	287
TWebBluetoothDevice类.....	288
TWebBluetoothService类.....	289
TWebBluetoothCharacteristic类.....	289
TWebUSBHID人机交互设备.....	290
选择一个设备/接口.....	290
TWebUSBHID属性.....	291
TWebUSBHID方法.....	291
TWebUSBHID事件.....	291
TWebUSBSerial USB转串口.....	292
TWebUSBSerial属性.....	292
TWebUSBSerial方法.....	292
TWebUSBSerial事件.....	293
TWebElementActionList元素动作列表.....	293
说明.....	293
TWebElementActionList属性.....	294
TWebElementActionList事件.....	294
TWebElementAction属性.....	294
TWebElementAction方法.....	295
TWebElementAction事件.....	296
示例.....	296
TWebLocalStorage本地存储.....	296
TWebSessionStorage会话存储.....	297
TWebDeviceOrientation设备方向传感器.....	297
TWebSpeechSynthesis语音合成.....	298
TWebSpeechSynthesis方法.....	298
TWebSpeechSynthesis属性.....	298
TWebSpeechSynthesis事件.....	298
TWebURLValidator网址验证.....	299
TWebURLValidator属性.....	299
TWebURLValidator事件.....	299
TWebLocalTextFile本地文本文件.....	299
TWebLocalTextFile属性.....	300
TWebLocalTextFile方法.....	300

TWebLocalTextFile事件	300
TWebLocalBinaryFile本地二进制文件	300
TWebLocalBinaryFile属性	301
TWebLocalBinaryFile方法	301
TWebLocalBinaryFile事件	302
TWebLocalFolder本地文件夹	302
TWebLocalFolder属性	302
TWebLocalFolder方法	302
TWebLocalFolder事件	303
TWebGeoLocation地理位置信息	303
TWebGEOLocation方法	303
TWebGEOLocation事件	303
TWebSocketClient套接字客户端	304
TWebSocketClient属性	304
TWebSocketClient方法	305
TWebSocketClient事件	305
TWebHttpRequest网络请求	305
TWebHttpRequest属性	308
TWebHttpRequest方法	308
TWebHttpRequest事件	308
TWebCookies网络Cookie保存用户与后端服务器的交互状态	309
TWebClientConnector客户端连接	310
TWebAESEncryption加密AES算法	312
TWebAESEncryption属性	312
TWebAESEncryption方法	312
TWebAESEncryption事件	313
TWebRSAEncryption加密RSA算法	313
TWebRSAEncryption属性	313
TWebRSAEncryption方法	313
TWebRSAEncryption事件	314
TWebRSASignature签名RSA算法	315
TWebRSASignature属性	315
TWebRSASignature方法	315
TWebRSASignature事件	315
TWebHMACSignature签名HMAC算法	316
TWebHMACSignature属性	316
TWebHMACSignature方法	316
TWebHMACSignature事件	317
TWebPushNotifications推送消息	317
注册推送消息	317
多用户共享同一设备	317
TWebPushNotifications属性	318
TWebPushNotifications方法	318
TWebPushNotifications事件	319
TMS WEB Core 3D应用程序	320
创建3D图表应用程序	320
3D商业图表应用程序	322

3D柱状图演示	322
轴的术语	322
创建数据序列对象	322
演示中显示的其他功能	323
事件	324
3D三维数学图表应用程序	324
3D三维散点图演示	324
轴的术语	325
创建数据序列对象	325
演示中显示的其他功能	325
3D曲面图演示	325
如何工作	325
创建数据系列对象	326
演示中显示的其他功能	327
3D PaintBox 绘图框应用程序	327
3D PaintBox绘图框演示	327
轴的术语	327
添加对象的代码	327
直接调用Three.js API	328
其他功能的示例代码	328
事件	329
3D 模型应用程序	329
需要引用“ThreeJS Models (3d)” JS库	329
3D模型演示	329
加载OBJ/MTL模型的代码	330
加载GLTF模型的代码	330
模型的附加功能	330
TWebMyCloudDbClientDataset云数据存储服务客户端数据集组件	332
简介	332
使用TWebmyCloudDbClientDataset创建WEB应用程序	332
在myCloudData控制台设置你的myCloudData项目	332
创建一个TMS Web应用程序	332
设置TWebmyCloudDbClientDataset组件	332
指定组件属性	333
创建对象库中每个对象的字段或属性	333
在对象检查器Object Inspector中选择字段	333
在代码中创建字段	333
添加连接到DataSet数据集的数据组件	333
设置DataSource数据源和数据组件	334
设置DBGrid的列	334
设置一个新记录事件	334
运行Web应用程序	334
任务列表演示	334
本演示中的其他功能	334
故障排除	335
参考部分	335
TWebMyCloudDbClientDataset组件	335

TWebmyCloudDbClientDataSet属性 .....	335
TWebmyCloudDbClientDataSet方法 .....	336
TWebFirestoreClientDataset谷歌云Firestore客户端数据集组件 .....	337
简介 .....	337
使用TwebFirestoreClientDataset创建网络应用程序 .....	337
在Firebase控制台设置你的Firestore项目 .....	337
创建一个TMS的WEB应用程序 .....	338
设置TWebFirestoreClientDataset组件 .....	338
添加连接到DataSet数据集的数据组件 .....	339
运行网络应用程序 .....	340
任务列表演示 .....	340
疑难解答 .....	341
在Firestore过滤记录 .....	341
在Firestore层面上可用的过滤方法 .....	342
Firestore过滤演示 .....	345
基于代码处理的新异步方法 .....	346
登录认证的总结和替代方法 .....	348
用户的具体数据（多租户） .....	351
TWebFirestoreClientDataset参考资料 .....	353
TWebFirestoreClientDataset方法 .....	354
异步方法 .....	356
与签到有关的方法 .....	357
技巧、窍门、故障排除说明 .....	359
TWebRadServerClientDataset客户端数据集 .....	360
简介 .....	360
配置你的Embarcadero Rad服务器后端 .....	360
通过TWebRadServerClientDataset访问Rad Server .....	362
参考资料 .....	363
属性 .....	363
方法 .....	363
事件 .....	363
TWebDreamFactoryClientDataset开源DreamFactory客户端数据集 .....	364
简介 .....	364
配置DreamFactory后端 .....	364
创建SQLite服务'tasksdb' .....	364
创建模式表任务 .....	364
设置CORS .....	365
设置一个角色 "LoggedIn "来访问 "tasksdb "服务 .....	366
创建应用程序任务 .....	366
使用DreamFactory通过TWebDreamFactoryClientDataset .....	367
参考资料 .....	367
属性 .....	367
方法 .....	367
事件 .....	368
TWebFaunaDbClientDataSet云数据库FaunaDb客户端数据集 .....	369
简介 .....	369
配置FaunaDb后端服务 .....	369



通过TWebFaunaDBClientDataset使用FaunaDB .....	370
参考资料 .....	370
属性 .....	370
方法 .....	371
事件 .....	371
TWebSQLRestClientDataset客户端数据集, TWebSQLRestConnection服务连接 .....	372
简介 .....	372
配置SQLDBRESTBridge后端服务 .....	372
通过TWebSQLRestClientDataset使用SQLite .....	372
jQuery 组件 .....	375
TWebJQXButton按钮 .....	376
说明 .....	376
HTML 模板标签 .....	376
TWebJQXButton属性 .....	377
TWebJQXButton事件 .....	377
TWebJQXButtonGroup按钮群组 .....	377
说明 .....	377
HTML 模板标签 .....	377
TWebJQXButtonGroup属性 .....	378
TWebJQXButtonGroup事件 .....	378
TWebJQXCalendar日历 .....	378
说明 .....	378
HTML 模板标签 .....	378
TWebJQXCalendar属性 .....	379
TWebJQXCalendar事件 .....	379
TWebJQXColorPicker颜色选择器 .....	379
说明 .....	379
HTML 模板标签 .....	380
TWebJQXColorPicker属性 .....	380
TWebJQXColorPicker事件 .....	380
TWebJQXComboBox下拉组合框 .....	380
说明 .....	380
HTML 模板标签 .....	381
TWebJQXComboBox属性 .....	381
TWebJQXComboBox方法 .....	381
TWebJQXComboBox事件 .....	381
TWebJQXDateTimeInput日期时间输入框 .....	382
说明 .....	382
HTML 模板标签 .....	382
TWebJQXDateTimeInput属性 .....	382
TWebJQXDateTimeInput事件 .....	382
TWebJQXDropDownList下拉列表 .....	383
说明 .....	383
HTML 模板标签 .....	383
TWebJQXDropDownList属性 .....	383
TWebJQXDropDownList方法 .....	383
TWebJQXDropDownList事件 .....	383

TWebJQXGrid网格	384
说明	384
HTML 模板标签	385
TWebJQXGrid属性	385
TWebJQXGrid方法	386
TWebJQXGrid事件	386
TWebJQXKnob圆形仪表	387
说明	387
HTML 模板标签	387
TWebJQXKnob属性	387
TWebJQXKnob事件	388
TWebJQXMaskedInput掩码输入框	388
说明	388
HTML 模板标签	389
TWebJQXMaskedInput属性	389
TWebJQXMaskedInput事件	389
TWebJQXMenu菜单	390
说明	390
HTML 模板标签	390
TWebJQXMenu属性	390
TWebJQXMenu事件	390
TWebJQXNumberInput数字输入编辑框	391
说明	391
HTML 模板标签	391
TWebJQXNumberInput属性	391
TWebJQXNumberInput事件	391
TWebJQXProgressBar进度条	392
说明	392
HTML 模板标签	392
TWebJQXProgressBar属性	395
TWebJQXRangeSelector范围选择器	393
说明	393
HTML 模板标签	393
TWebJQXRangeSelector属性	393
TWebJQXRangeSelector事件	393
TWebJQXRating评价控件	394
说明	394
HTML 模板标签	394
TWebJQXRating属性	394
TWebJQXRating事件	394
TWebJQXResponsivePanel响应式面板	394
说明	394
TWebJQXResponsivePanel属性	395
TWebJQXResponsivePanel方法	395
TWebJQXResponsivePanel事件	395
TWebJQXSlider滑块择值	395
说明	396

HTML 模板标签.....	396
TWebJQXSlider属性.....	396
TWebJQXSlider事件.....	397
TWebJQXTabs标签页.....	397
说明.....	397
HTML 模板标签.....	397
TWebJQXTabs属性.....	397
TWebJQXTabs事件.....	398
TWebJQXTagCloud标签云.....	398
说明.....	398
HTML 模板标签.....	398
PropTWebJQXTagCloud属性.....	398
TWebJQXTagCloud事件.....	399
连接数据库.....	400
使用WebSockets.....	404
IndexedDB数据库.....	407
TMS WEB Core IndexedDB 库.....	407
TIndexedDbClientDataSet客户端数据集组件.....	407
TIndexedDb类.....	407
创建IndexedDB应用程序.....	407
创建一个TMS 网络web应用程序.....	407
添加连接到DataSet数据集的数据库感知组件.....	409
运行Web应用程序.....	410
管理IndexedDB数据库.....	410
任务列表演示.....	411
本演示中的其他功能.....	411
TWebIndexedDbClientDataSet客户端数据集组件.....	412
说明.....	412
TWebIndexedDbClientDataSet属性.....	412
TWebIndexedDbClientDataSet方法.....	413
TIndexedDb (高级用途).....	413
说明.....	414
属性.....	414
TIndexedDb方法和事件.....	414
Out-of-line key独立于对象store中的记录的【键】规范.....	417
设置索引.....	420
处理 IndexedDB 的异步行为.....	421
TMS WEB Electron桌面应用程序.....	423
创建TMS Web Electron桌面应用程序.....	423
构建应用程序.....	425
将你的应用程序迁移到更新的版本.....	426
替换the main.js文件.....	426
对话框回调.....	426
删除TElectronIPCCommunication.OnMessage中的Sender参数.....	427
删除TElectronIPCMain相关代码.....	427
访问开发工具.....	428
拖放.....	428

字体 .....	430
用本地数据库设置你的项目 .....	430
Electron组件 .....	432
TElectronOpenDialog打开对话框 .....	432
说明 .....	432
TElectronOpenDialog属性 .....	432
TElectronOpenDialog方法 .....	432
TElectronSaveDialog保存对话框 .....	433
说明 .....	433
TElectronSaveDialog属性 .....	433
TElectronSaveDialog方法 .....	433
TElectronMessageBox消息对话框 .....	434
说明 .....	434
TElectronMessageBox属性 .....	434
TElectronMessageBox方法 .....	435
TElectronErrorBox错误对话框 .....	435
说明 .....	435
TElectronErrorBox属性 .....	435
TElectronErrorBox方法 .....	435
TElectronMainMenu主菜单 .....	435
说明 .....	435
TElectronPopupMenu弹出式菜单 .....	435
说明 .....	435
TElectronPopupMenu方法 .....	436
TElectronBrowserWindow浏览器窗口 .....	437
说明 .....	437
TElectronBrowserWindow属性 .....	437
TElectronBrowserWindow方法 .....	437
TElectronBrowserWindow事件 .....	438
使用表单的多个窗口 .....	438
使用其他来源的多个窗口 .....	439
显示一个窗口 .....	439
TElectronTrayIcon托盘图标 .....	439
说明 .....	439
TElectronTrayIcon属性 .....	440
TElectronTrayIcon事件 .....	440
TElectronIPCCommunication电脑通信 .....	440
说明 .....	440
TElectronIPCCommunication属性 .....	440
TElectronIPCCommunication方法 .....	440
TElectronIPCCommunication事件 .....	440
发送消息给父窗口 .....	441
发送消息到一个通道 .....	441
接收消息 .....	441
TElectronMySQLClientDataSet数据库MySQL客户端数据集 .....	442
说明 .....	442
任务列表演示 .....	442

BLOB演示	443
TElectronMySQLClientDataSet属性	443
TElectronMySQLConnection数据库MySQL连接组件	443
说明	443
TElectronPostgreSQLClientDataSet数据库PostgreSQL客户端数据集组件	443
说明	443
任务列表演示	444
BLOB演示	445
TElectronPostgreSQLClientDataSet属性	445
TElectronPostgreSQLConnection数据库PostgreSQL连接组件	445
说明	445
TElectronFileWatcher文件监控	445
说明	445
TElectronFileWatcher属性	445
TElectronFileWatcher方法	446
TElectronGlobalShortcut键盘快捷键	446
说明	446
TElectronGlobalShortcut属性	446
TElectronGlobalShortcut事件	446
TElectronStringList字符串列表	446
TElectronStringList方法	446
TElectronBinaryDataStream二进制数据流	448
TElectronBinaryDataStream属性	448
TElectronBinaryDataStream方法	448
TElectronClipboard剪贴板	448
TElectronClipboard属性	448
TElectronClipboard方法	448
TElectronShell命令行解释器	449
TElectronShell方法	449
TElectronIPCRenderer电脑渲染器	449
TElectronIPCRenderer方法	449
TElectronDragAndDrop拖放	450
TElectronPath路径	450
TElectronPath方法	450
TElectronWindow窗口	450
TElectronWindow方法	451
其它可用方法	451
TMS WEB Miletus桌面应用程序	453
创建TMS Web Miletus桌面应用程序	453
调试和访问开发工具	455
部署	456
拖放	456
从PC桌面拖拽到Miletus应用程序中	456
从Miletus应用程序中拖拽到PC桌面	457
Miletus组件	458
TMiletusOpenDialog打开对话框	458
说明	458

TMiletusOpenDialog属性	458
TMiletusOpenDialog方法	458
TMiletusSaveDialog保存对话框	459
说明	459
TMiletusSaveDialog属性	459
TMiletusSaveDialog方法	460
TMiletusMessageBox消息框	460
说明	460
TMiletusMessageBox属性	460
TMiletusMessageBox方法	461
TMiletusErrorBox错误对话框	461
说明	461
TMiletusErrorBox属性	461
TMiletusErrorBox方法	461
TMiletusMainMenu主菜单	462
说明	462
更新TMiletusMainMenu主菜单	462
TMiletusPopupMenu弹出式菜单	462
说明	462
TMiletusPopupMenu方法	462
TMiletusPopupMenu事件	462
更新TMiletusPopupMenu弹出式菜单	463
TMiletusNotificationCenter通知中心	463
说明	463
TMiletusNotificationCenter方法	463
TMiletusWindow窗口	463
说明	463
TMiletusWindow属性	464
TMiletusWindow方法	464
TMiletusWindow事件	464
使用表单的多窗口	465
使用其他来源的多窗口	465
显示一个窗口	465
表单窗体之间通信	465
TMiletusTrayIcon托盘图标	465
说明	466
TMiletusTrayIcon属性	466
TMiletusTrayIcon事件	466
TMiletusClientDataSet客户端数据集	466
TMiletusClientDataSet属性	466
TMiletusAccessDBDriver数据库Access驱动	467
TMiletusMySQLDBDriver数据库MySQL驱动	467
TMiletusSQLiteDBDriver数据库SQLite驱动	467
TMiletusPostgreSQLDBDriver数据库PostgreSQL驱动	467
TMiletusMSSQLDBDriver数据库MSSQL驱动	468
TMiletusIBDBDriver数据库InterBase驱动	468
TMiletusFBDBDriver数据库FireBird驱动	468

---

TMiletusFileWatcher文件监控 .....	469
说明 .....	469
TMiletusFileWatcher属性 .....	469
TMiletusFileWatch事件 .....	469
TMiletusGlobalShortcuts全局快捷键 .....	469
说明 .....	469
TMiletusGlobalShortcut属性 .....	469
TMiletusGlobalShortcut事件 .....	469
TMiletusStringList字符串列表 .....	469
TMiletusStringList方法 .....	469
TMiletusBinaryDataStream二进制数据流 .....	470
TMiletusBinaryDataStream属性 .....	471
TMiletusBinaryDataStream方法 .....	471
TMiletusClipboard剪贴板 .....	471
TmiletusClipboard属性 .....	471
TmiletusClipboard方法 .....	471
TMiletusShell命令行解释器 .....	472
TMiletusShell方法 .....	472
TMiletusINIFile操作INI文件 .....	472
TMiletusINIFile方法 .....	472
TMiletusRegistry注册表 .....	474
TMiletusRegistry属性 .....	474
TMiletusRegistry方法 .....	474
其他可用的方法 .....	476
定制控件开发 .....	478

## 简介

Delphi是软件行业的传奇编程语言之一，该产品于1995年2月14日情人节在美国San Francisco（旧金山）正式发布，她是软件开发历史的基石。今天随着IT科技的飞速发展，各种新平台和框架不断涌现，各种新编程语言也在不断推出，知名的TIOBE编程语言排行榜总计有多达数百种编程语言，让初学者眼花缭乱，但是Delphi因其作为跨平台本机原生开发工具的可靠性和有效性而在IT界发展趋势来来去去的潮流中站稳了脚跟，在编程语言的世界中拥有自己的一席之地。

从Delphi 1.0发布至今的27年中，对于整整一代软件开发者来说，Delphi给全世界数以百万计的程序员提供了编程工具和赖以养家糊口的工作技能，他（她）们使用Delphi构建了数以百万计的应用软件，以及被数以百万计的用户使用的应用。有史以来使用最多的Delphi应用程序可能是Windows桌面版Skype或者是中国著名的网络游戏《热血传奇》，许多人依靠Delphi获得了广受赞誉的名声甚至实现了个人财务自由。同时Delphi也被用来开发价值数百万美元的企业软件和在金融市场上管理千百万美元的应用软件。现在Delphi至今仍被用来维护这些应用软件，并在工业、金融、健康、旅游和体育领域（包括其他领域）继续创建新的应用软件.....包括为用于生日派对的移动应用软件，或者用来管理运营一个小型机场的软件系统。

2022年最新发布的Delphi 11.1可以让开发人员以5倍速度只写一套代码，即可为Windows, Android, iOS, macOS, 和Linux分别开发编译构建原生本机应用软件，甚至通过TMS WEB Core框架也能用于开发树莓派Raspberry Pi OS应用。



图 Delphi 11 Alexandria

更重要的是，随着时间的推移，Delphi不仅没有被历史抛弃，它还随着当代软件



解决方案日益复杂和客户开发需求的日益增长而不断发展和扩展其功能、特性和库。自Delphi 1发布以来的27年里，Delphi一直与快速发展的IT行业保持着高度、紧密的联系，今天Delphi 11更提供了一个独特且极具吸引力的主张，它结合了多平台本机功能、低代码、高生产力开发以及与其他编程语言的集成（通过C++ Builder使用C++，通过 Python4Delphi 使用Python）。

今天，从北美大陆到欧洲、从非洲草原到遥远的东方大国，众多世界领先的公司包括财富100强公司以及医疗机构、国防部门、地质勘察、船舶制造、金融证券、核工业、航天工业等关键业务依赖使用 Delphi 开发的应用软件，如：索尼、福特、ABB、西门子、软银、3M、佳能、摩托罗拉、诺基亚、丰田、Dell、松下、现代、UPS、美国航空航天局等。

本教程作者从1998年开始使用Delphi，从事软件开发25年，见证了Delphi从辉煌到低谷，又从低谷重新崛起的历史，笔者在这期间也曾使用过JavaScript、PHP、Java、C#、QT等编程语言和框架，但是Delphi以其类英语的优雅编程语法、可视化快速开发的高效率、丰富的第三方组件、原生跨平台开发的能力，在众多开发工具中独树一帜！成为笔者最喜欢的编程语言！

无论你是一名JavaScript前端程序员或者是一名Java后端程序员，更或者是一名C#开发者，笔者强烈推荐你学习、试用一下Delphi，毕竟艺多不压身！相信你也一定会和我一样喜欢上Delphi！因为Delphi可以让你一个人抵上5个Java程序员的开发效率，节省大量的编码时间、配置时间、排错时间，让你拥有更多的时间陪伴家人。无论是WEB前端、还是服务器后端，无论是桌面软件还是移动App应用开发，你一个人就可以用Delphi一门语言全部搞定！为什么不立即下载 RAD Studio Delphi 的免费试用版并在本机应用程序中试用它的强大功能和速度呢？！



图 Delphi 11

## Delphi开发的部分应用软件App展示



图 Delphi开发的MMORPG《热血传奇》

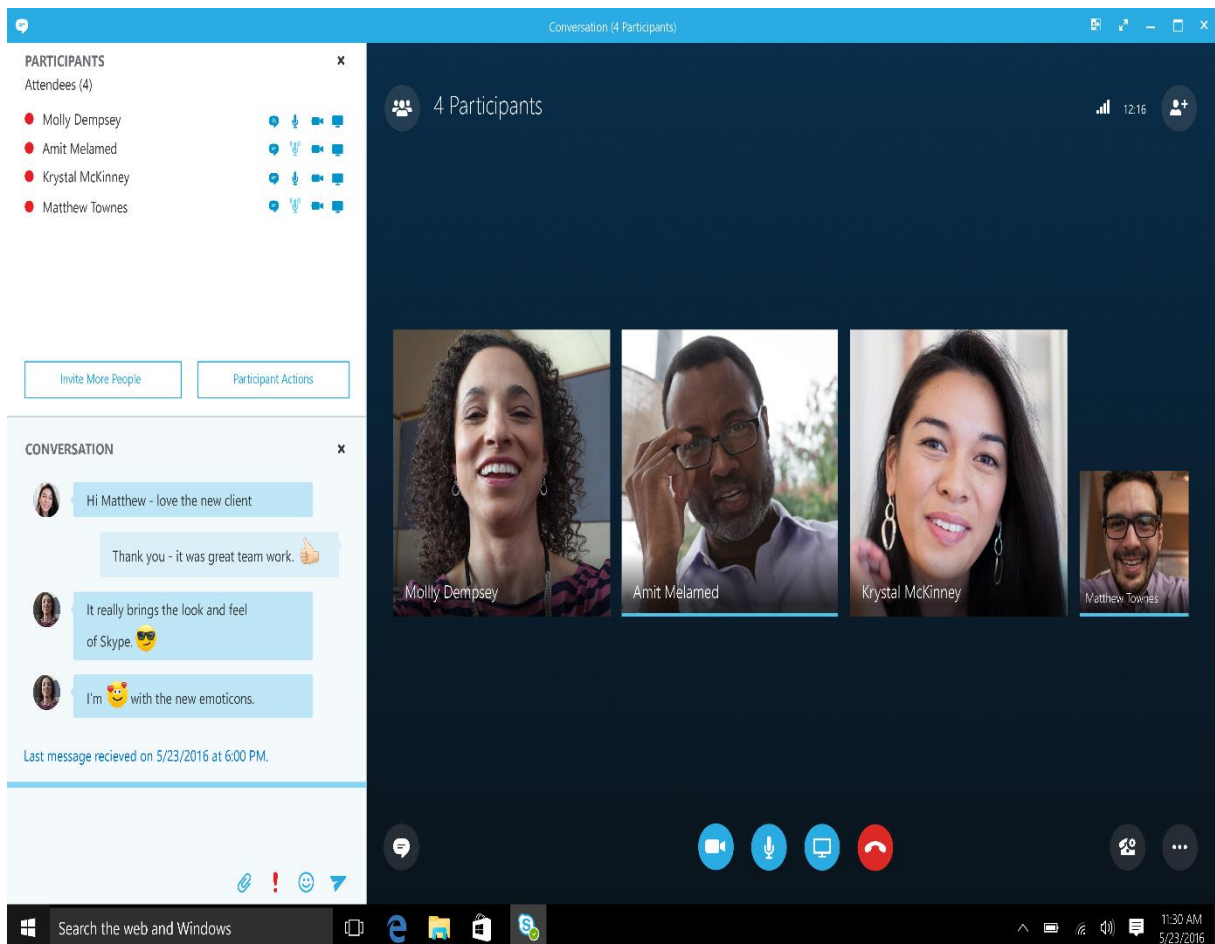


图 Delphi开发的《Skype》桌面版



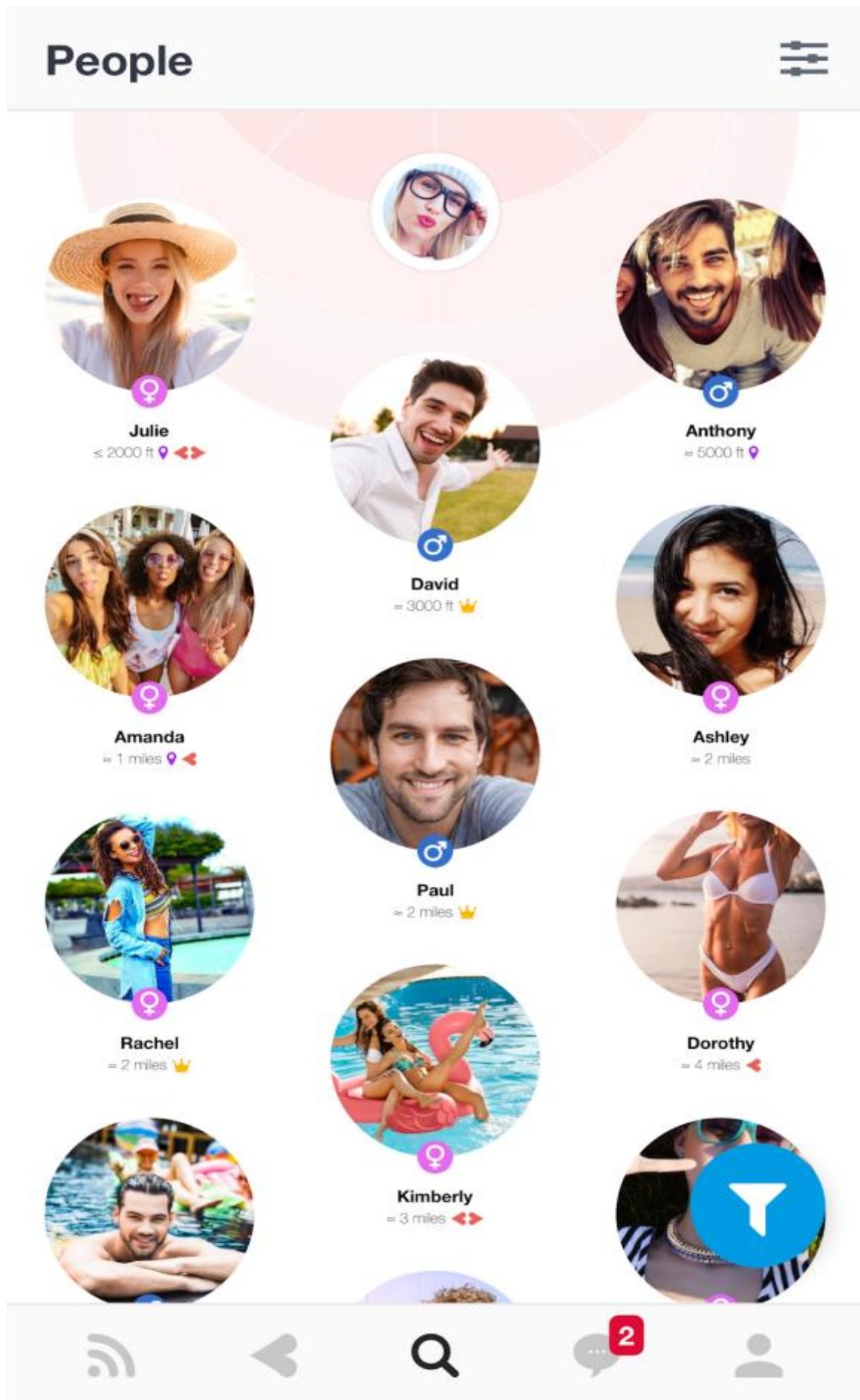


图 Delphi开发的手机社交App《KisKis》



图 Delphi开发的iOS手机游戏App《Idle Western》

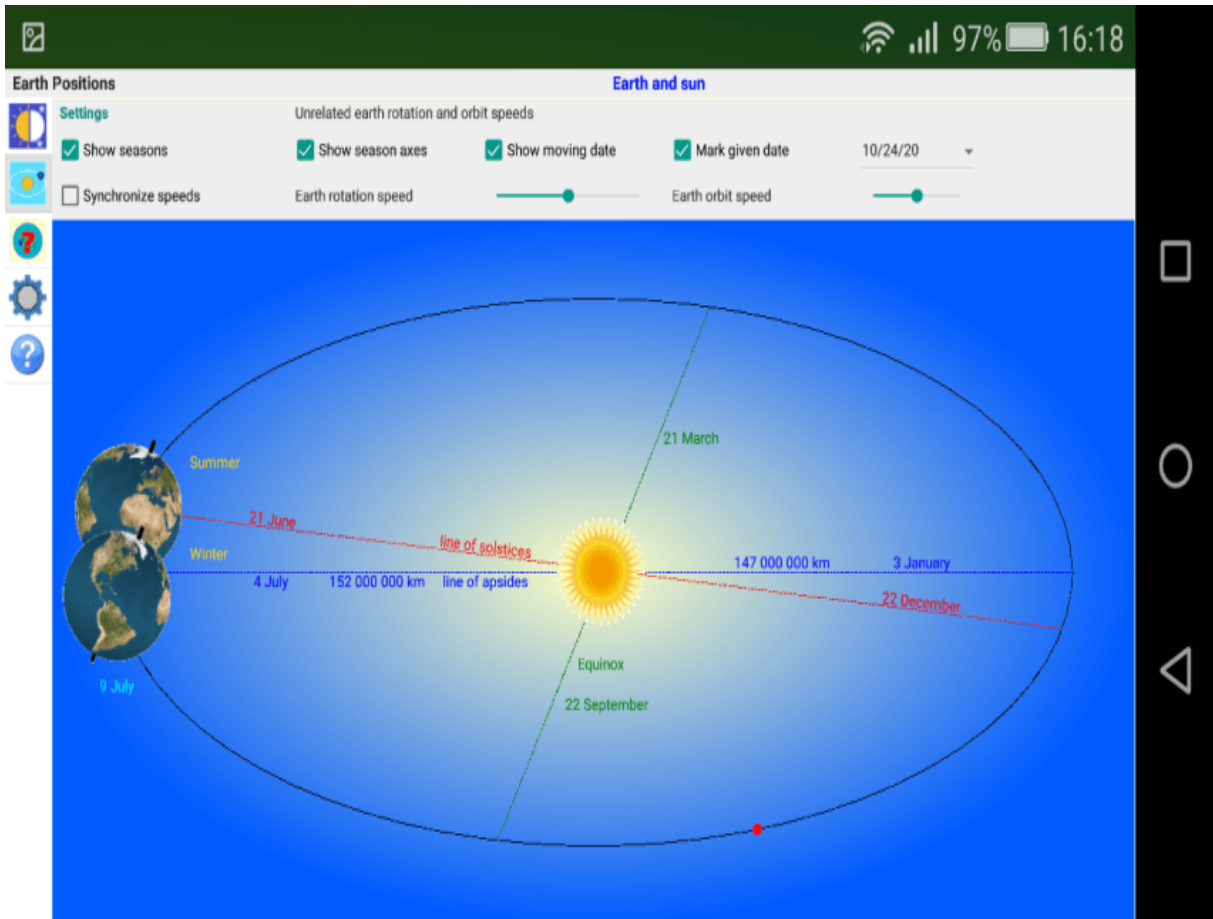


图 Delphi开发的太阳系教学手机App《Earth Positions》

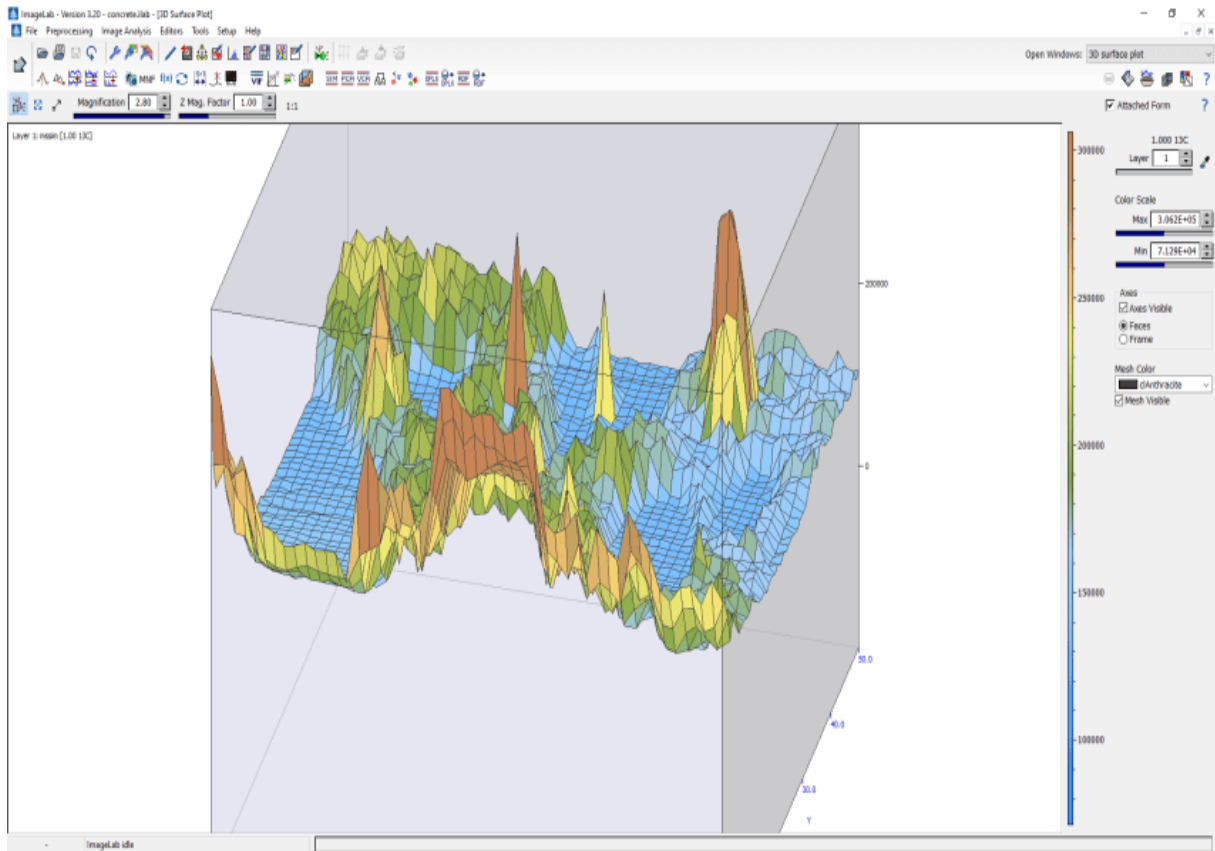


图 Delphi开发的原生多传感器成像工具，用于处理和分析高光谱图像软件《ImageLab》

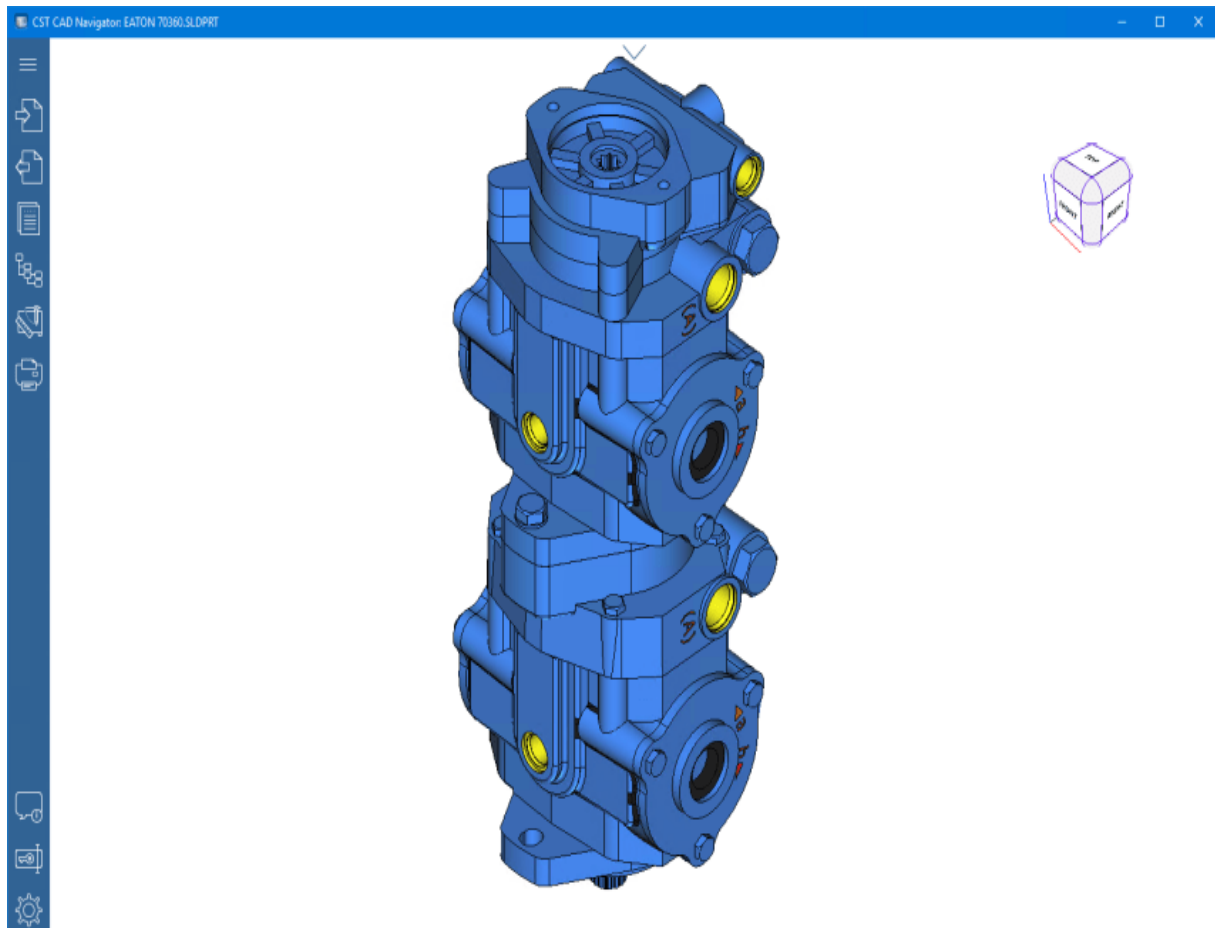


图 Delphi开发的跨平台CAD看图软件《CST CAD Navigator》





图 Delphi开发的即时战略游戏软件《Knights Province》



图 Delphi开发的猎鹰9火箭模拟仿真软件《Falcon 9 – First Stage Simulator》

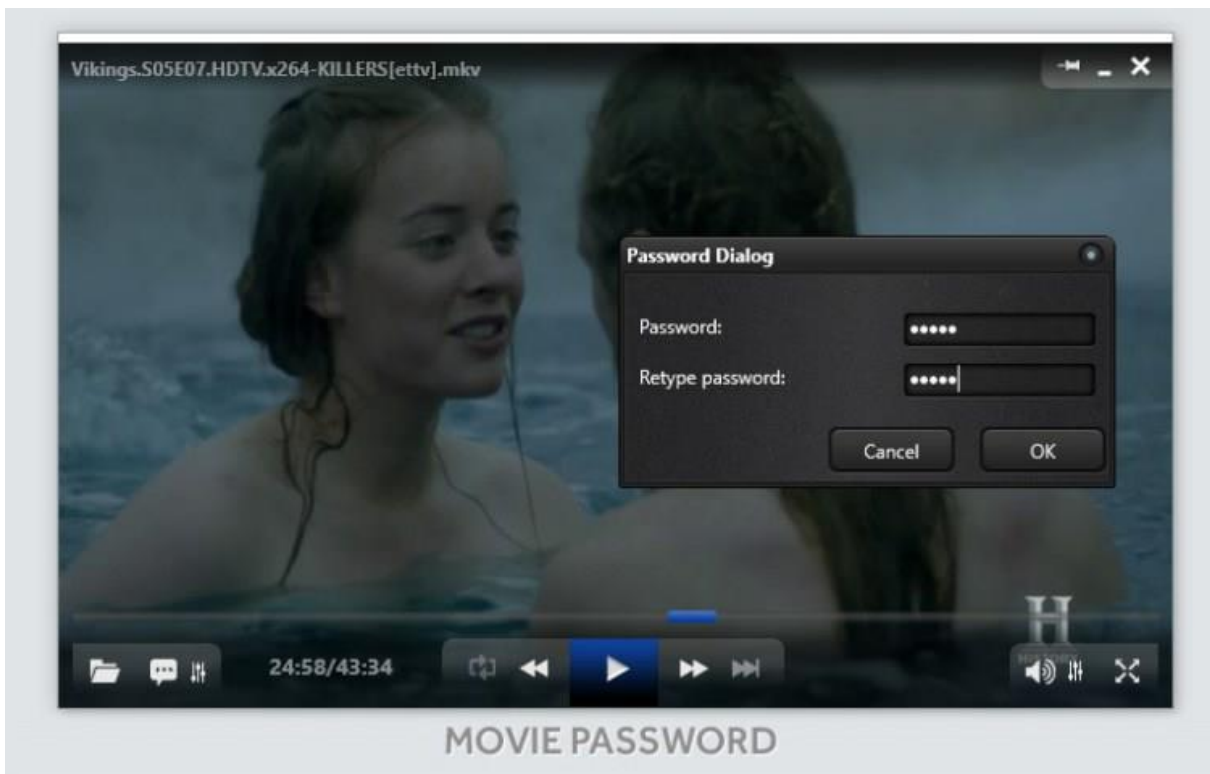


图 Delphi开发的视频播放器软件《ALLPlayer》

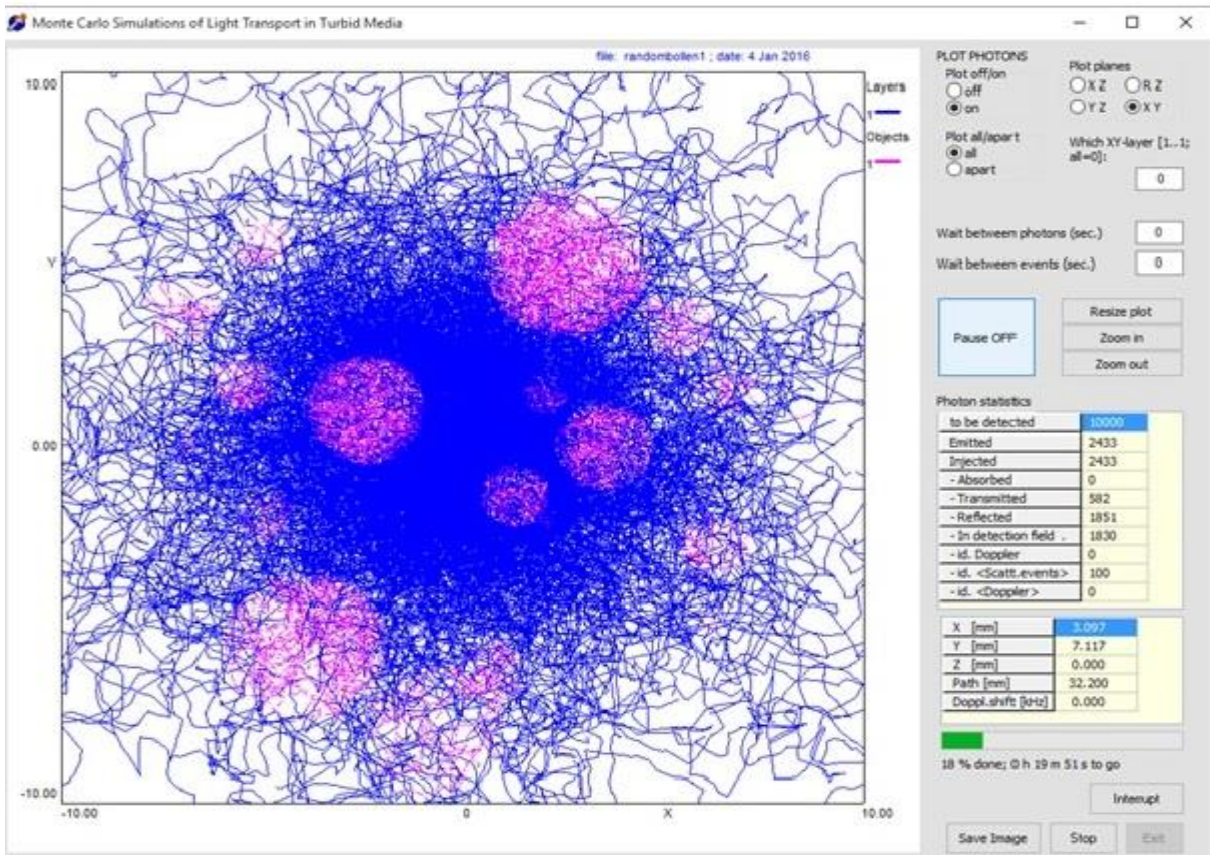


图 Delphi开发的独特的生物医学和临床应用《www.demul.net》



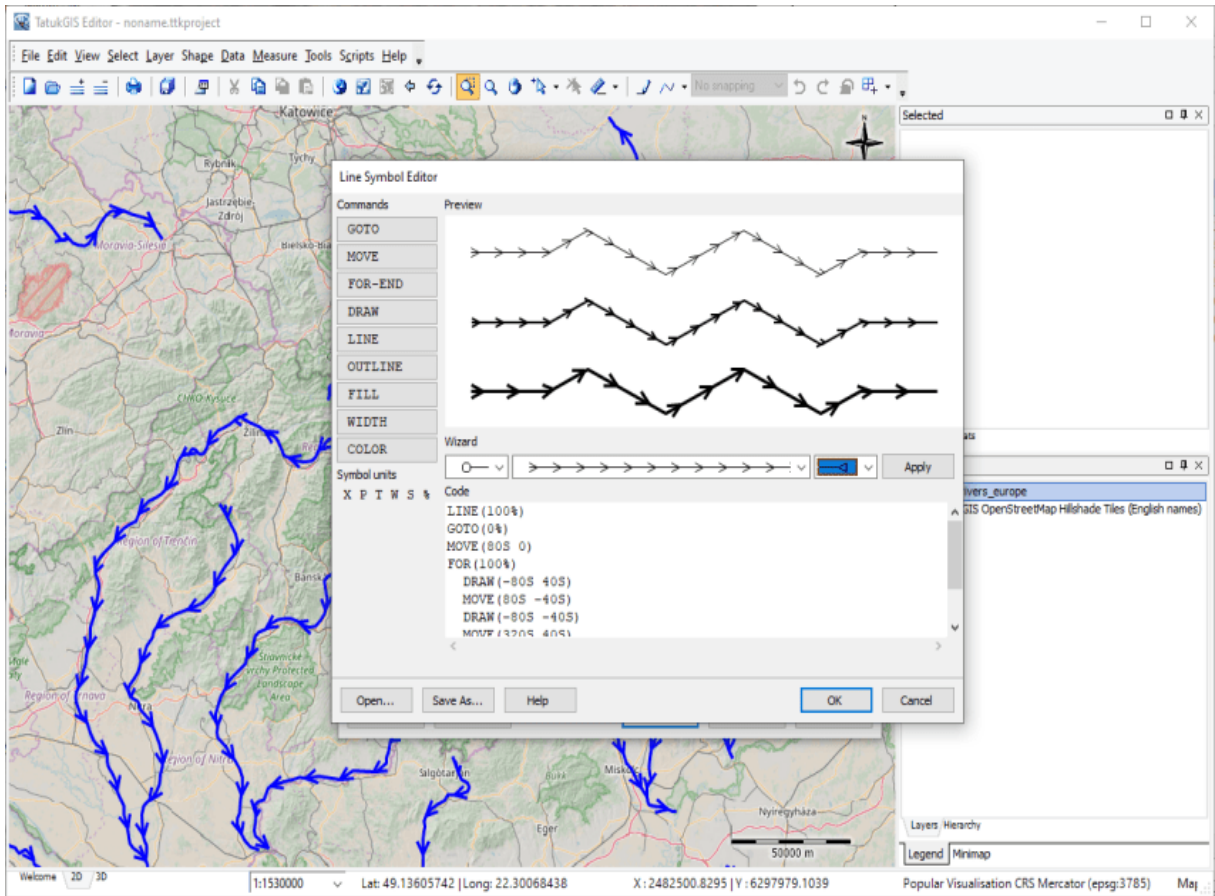


图 Delphi开发的强大的地理信息系统GIS软件包《TatukGIS Editor》

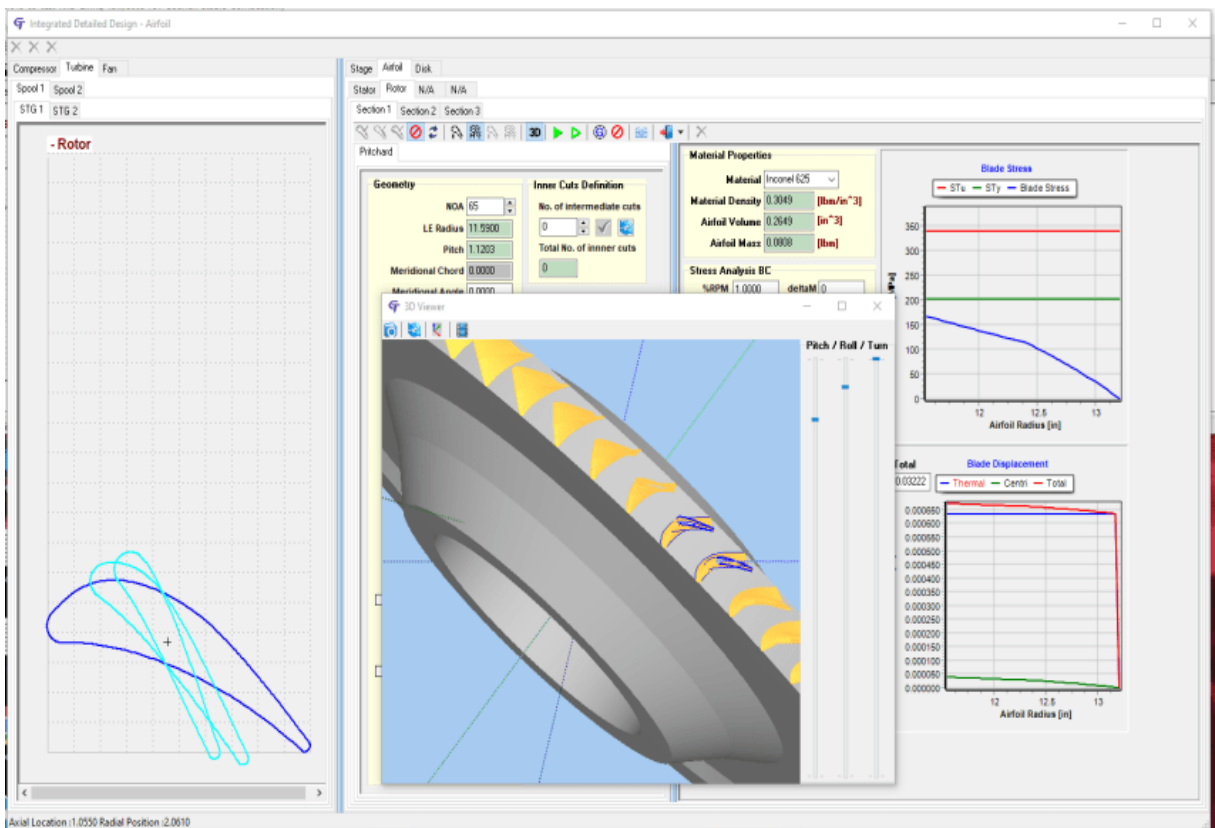


图 Delphi开发的燃气轮机多学科集成设计软件《MDIDS-GT》



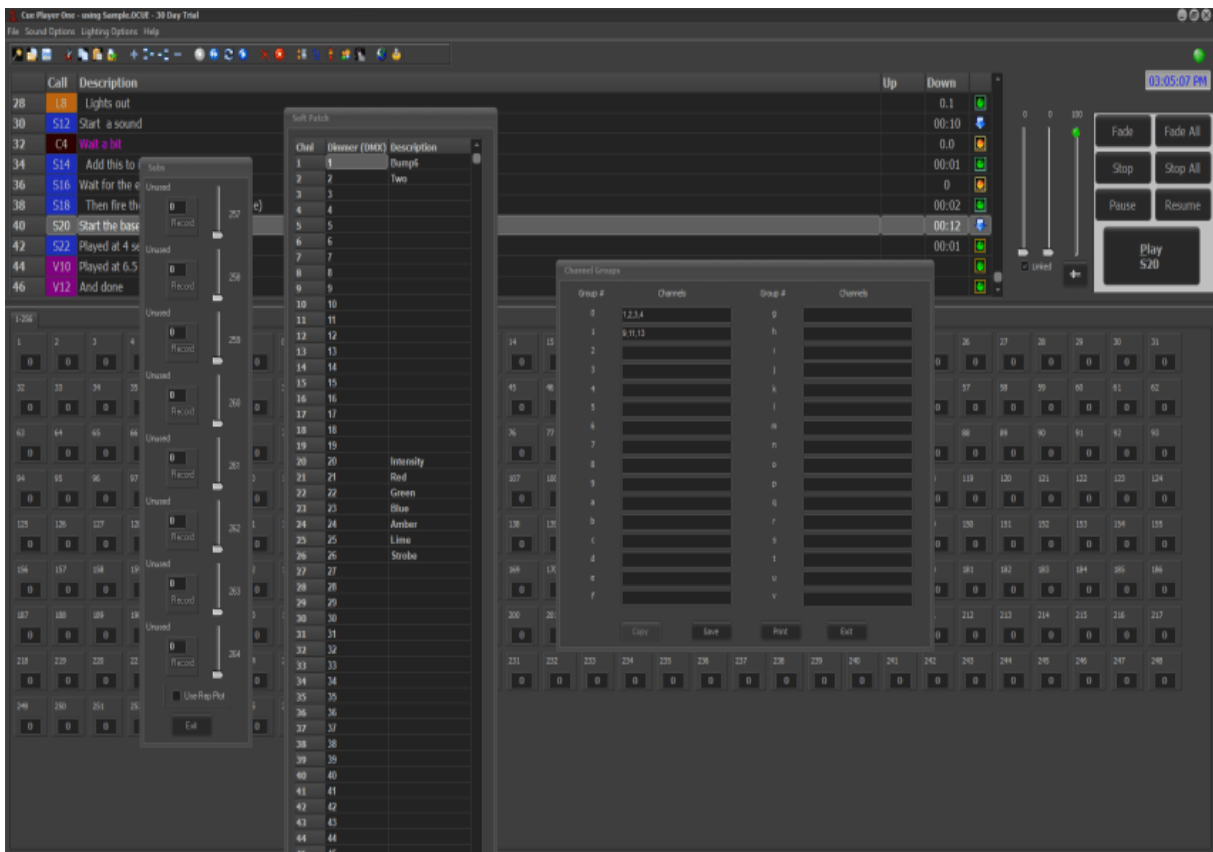


图 Delphi开发的用于剧院控制声音、视频、灯光和配件的专业软件《Cue Player One》

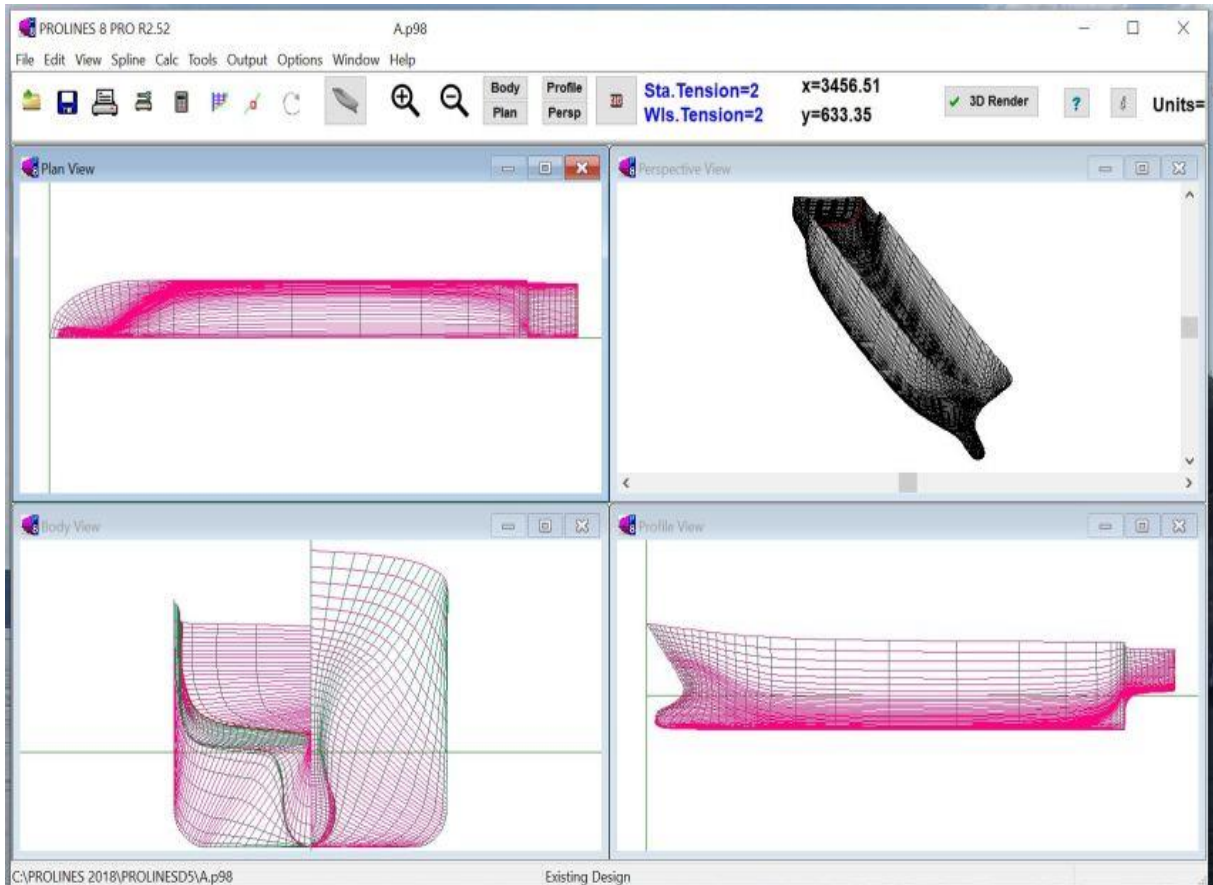


图 Delphi开发的船舶CAD设计软件《PROLINES 8》



图 Delphi开发的在线销售的连锁超市开发的商城应用App《Meu Venturão》



图 Delphi开发的卡牌游戏软件《Astral Heroes》

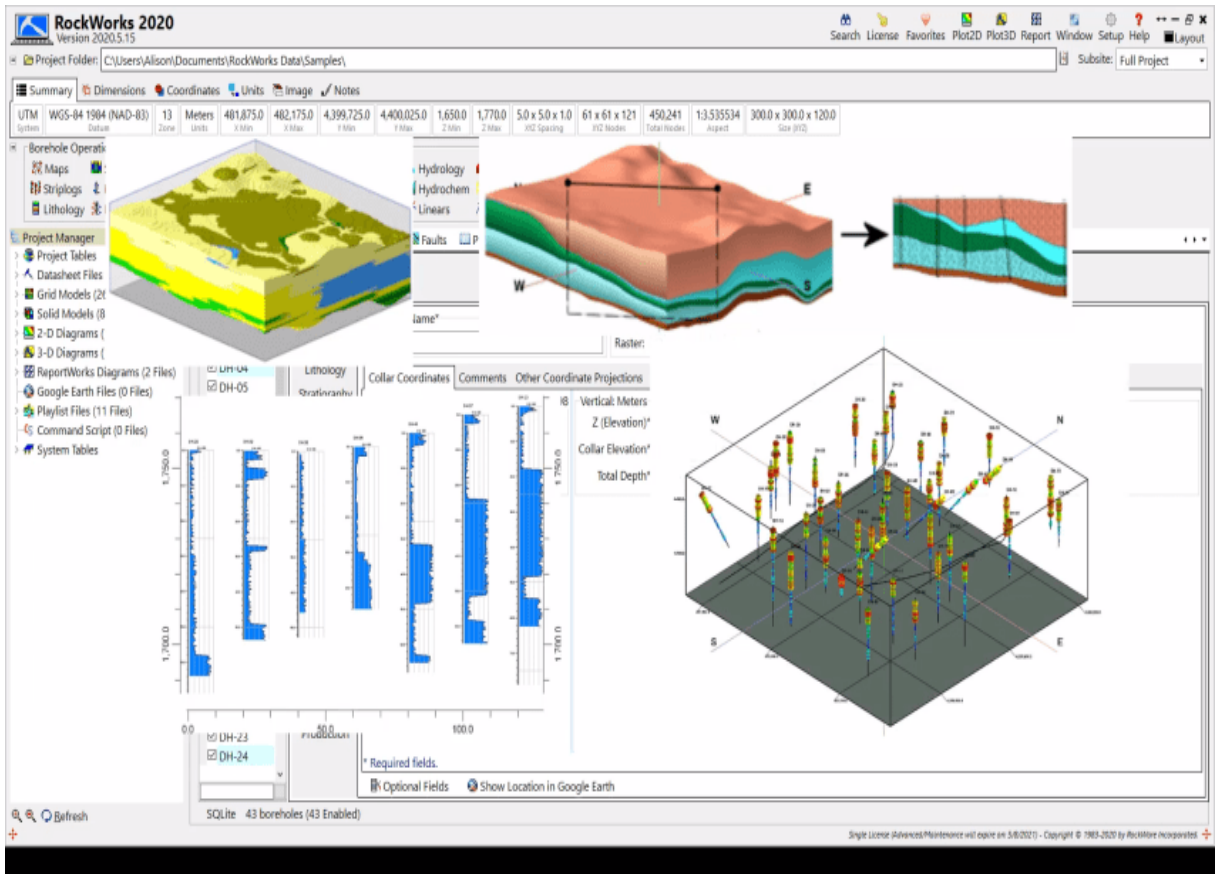


图 Delphi开发的地表和地下数据可视化的强大石油和采矿软件《RockWorks》



图 Delphi开发的生存探险3D游戏《StoneQuest》



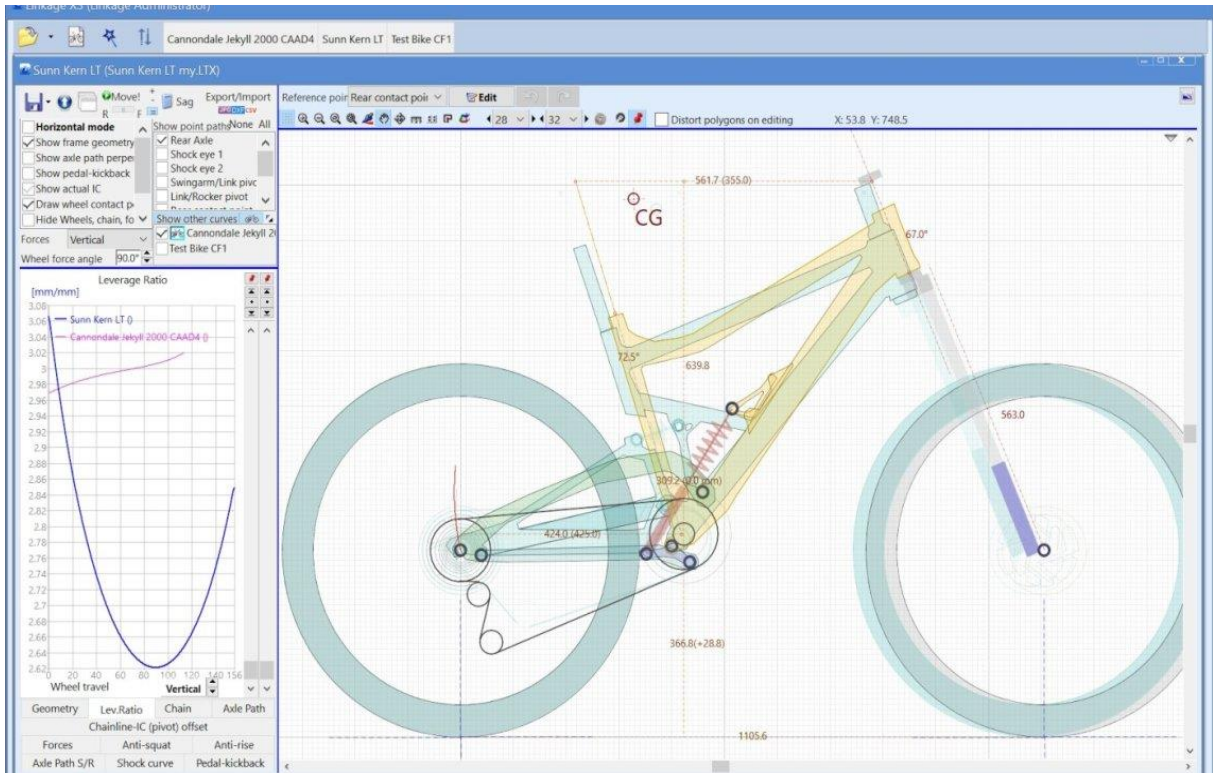


图 Delphi开发的山地自行车CAD设计软件《Linkage》

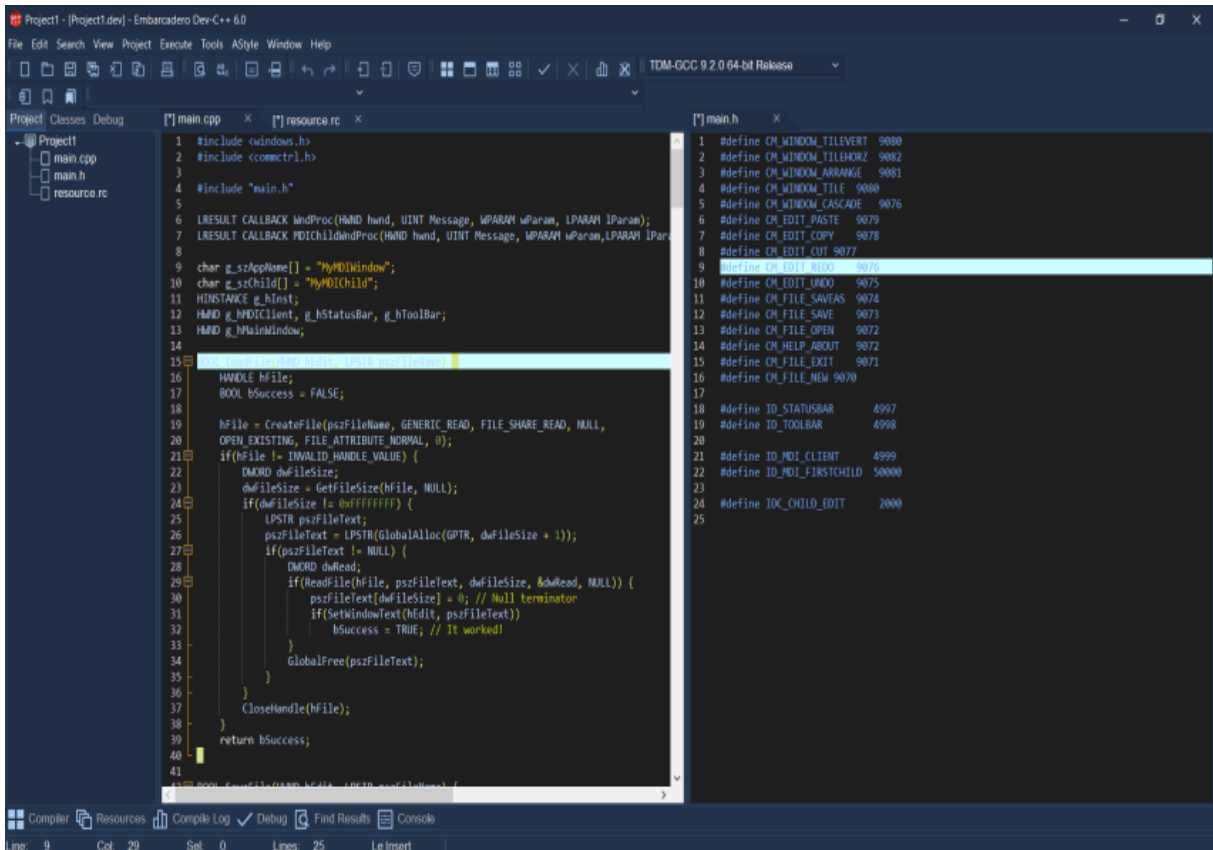


图 Delphi开发的开源C++编程软件《Dev C++》

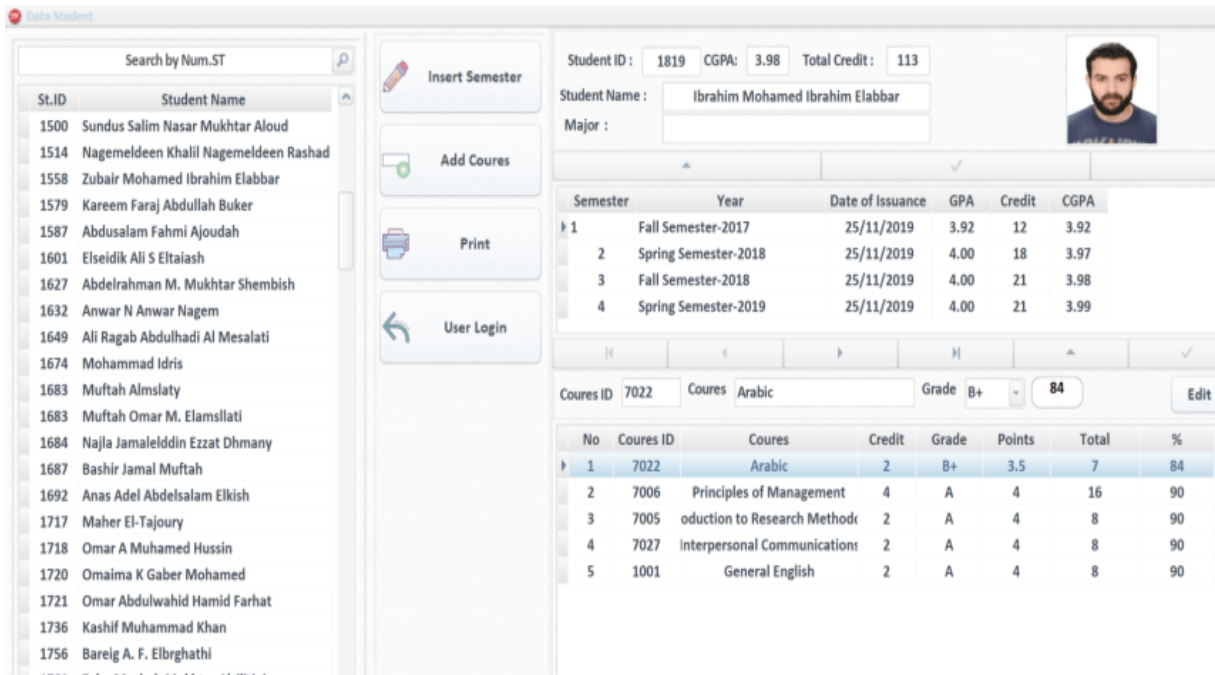


图 Delphi开发的学生注册、学习和考试系统软件



图 Delphi开发的角色扮演RPG游戏《Firecast 8》

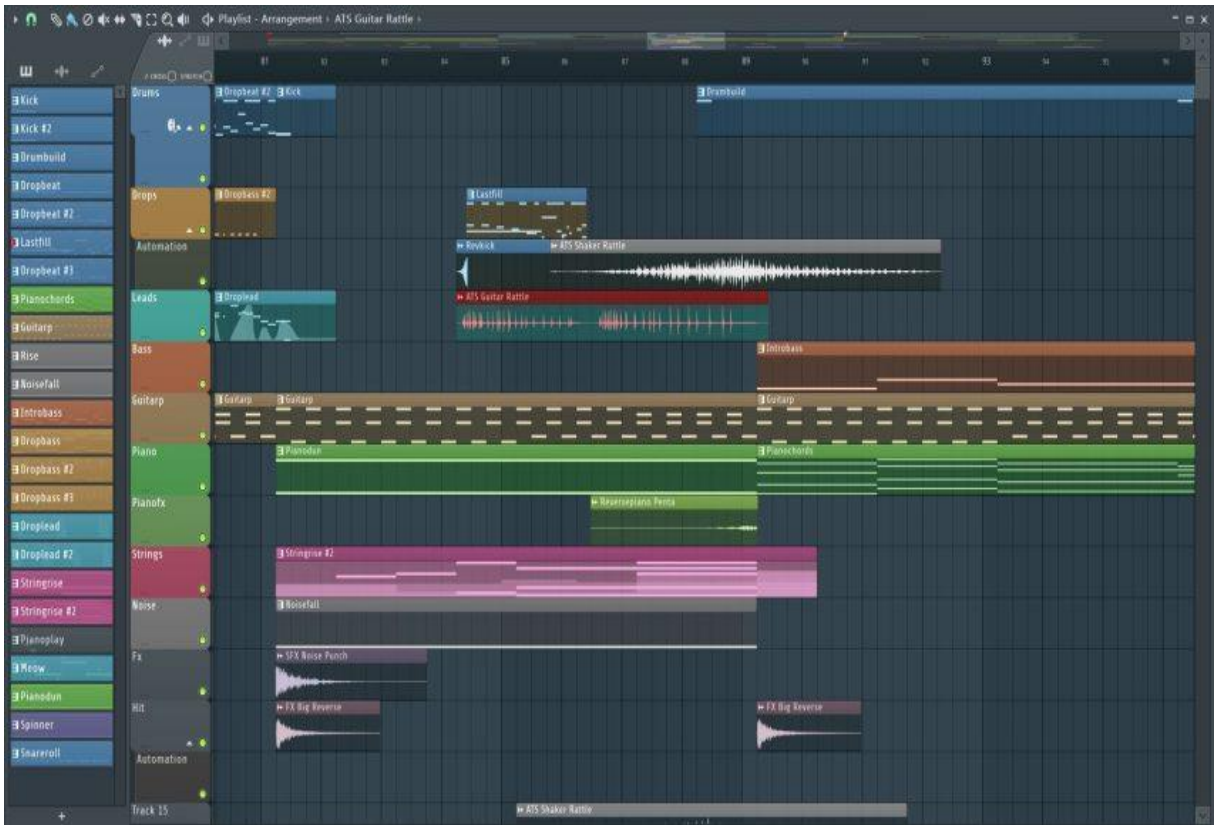


图 Delphi开发的数字音频工作站软件《FL Studio》



图 Delphi开发的即时战略游戏《Rise of Legions》





图 Delphi 开发的手机点餐App 《Wagamama》

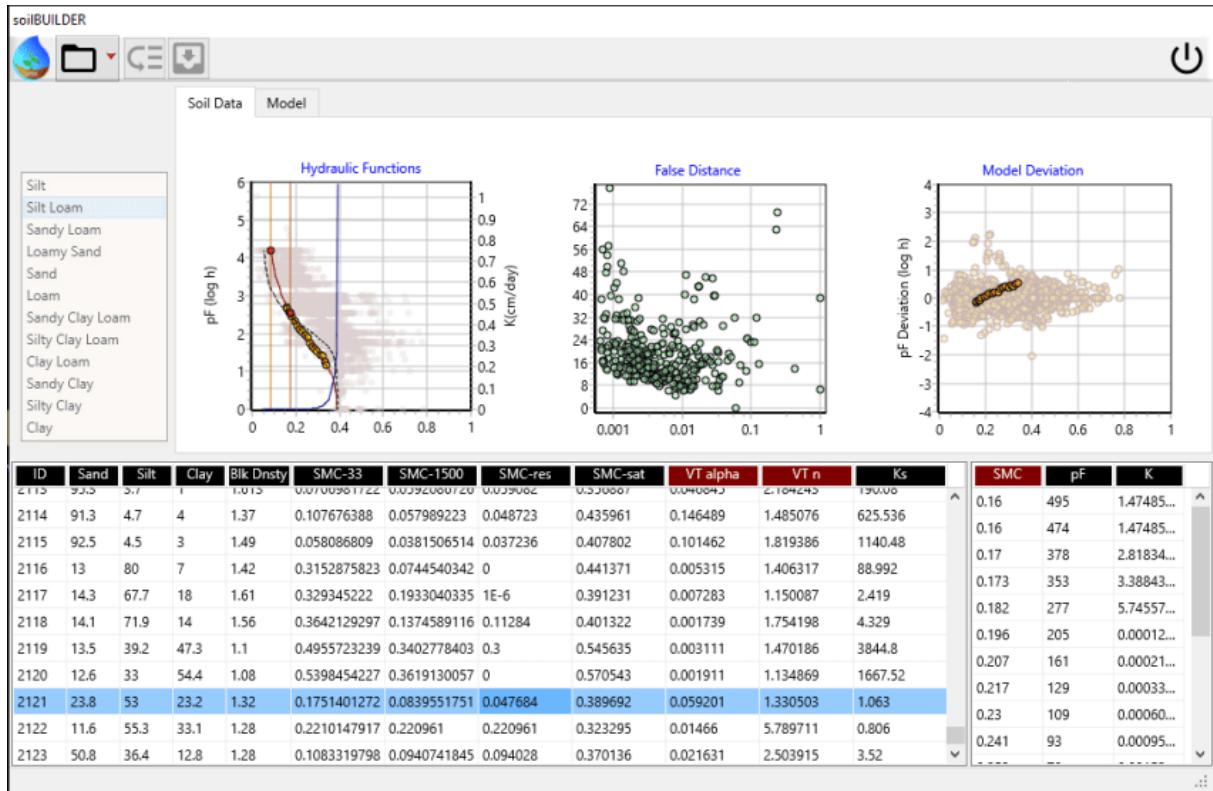


图 Delphi开发的土壤水力预测软件《oilBUILDER》

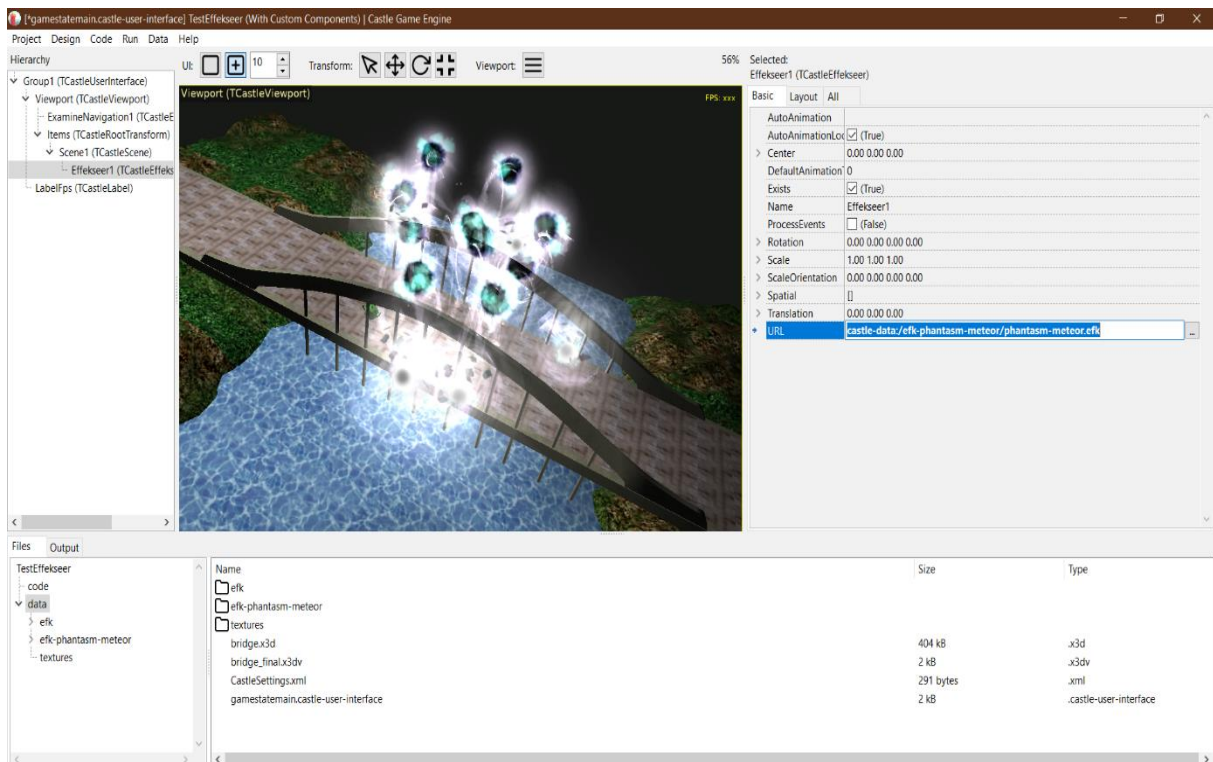


图 Delphi/Free Pascal开发的开源3D 游戏引擎《Castle Game Engine》

多年以来我一直告诉遇到的每一个程序员，Delphi是为所有平台、桌面、移动、服务端或前端Web开发应用程序的最好编程语言。Delphi不但可以为桌面电脑PC（如Windows、Linux或macOS）开发精美强大的应用程序软件，也同样适用于iOS和Android



等移动设备App开发。您可以使用Delphi创建各种类型的Web网络应用程序 - 不仅仅是简单的公司网站，而是包含现代Web上所有功能的强大应用。

Delphi真的很容易学习——编程语言的语法可以在几个小时内学会。Delphi 的 Object Pascal 语言是由经典的编程语言Pascal演变而来的，Pascal长期以来一直是大学和教育培训中非常流行的语言，用于教授计算机专业的大学生软件开发的基本概念，了解编程的基础知识，是编程语言的重要组成部分。

## Delphi/Object Pascal的历史与现状

从技术上讲，Delphi 是一个由 Object Pascal 编程语言支持并为其提供开发环境的IDE，可以实现高生产力和快速应用程序开发的工具。

Delphi 的第一个版本的想法最初是从 Borland 的 Turbo Pascal 开始出现的。然而，当时决定添加到 Pascal 语言中的面向对象扩展并不理想，因此开发团队决定重新开始并从其它编程语言汲取先进的开发理念（包括来自Apple 的 Object Pascal）。

## Delphi的杀手锏是什么？

最终，Delphi 的拖放式组件可视化开发迅速证明了这种方法能够为用户促进快速的可视化应用程序开发。现在，从那些盛大的Delphi 1.0发布会开始已经过去了二十七年多，Delphi已经有了数百个内置的可视化和非可视化组件，开发人员可以快速设计一个主界面响应式 UI 布局，然后轻松定制OS操作系统平台和设备特定的视图，而无需重新设计整体用户界面。这种将功能封装到可重用组件中的方式允许实际编写很少量的代码就能构建功能强大而稳定的软件。开发人员可以——而且仍然可以——用很少的代码或不用写代码来完全或大部分地生成完整的应用程序。从字面上理解，Delphi是最早的低代码开发平台之一。

Delphi/Object Pascal是一种为当今越来越多的计算设备——从智能手机和平板电脑到台式机和服务器——而构建的编程语言开发工具。尽管在中国大陆地区，由于某些复杂市场因素导致Delphi/Object Pascal不像Java那样流行，也无法像90年代那样受到更多中国用户的青睐，但是作为一种全栈万能编程语言，Delphi/Object Pascal在中国台湾省、东南亚、非洲、欧美广大地区仍然拥有广泛的忠实的用户群体，其中一些客户开发了很多令人印象深刻的软件产品，例如：著名的MMORPG游戏《热血传奇》客户端和服务端都是用Delphi开发的，被广泛使用的社交软件Skype也是由Delphi开发的，以及中国著名的文字处理软件WPS早期版本，知名的游戏引擎Unity3D的Pascal开源版Castle Game Engine以及GitHub上数量众多的Delphi/Object Pascal开源项目，这从侧面证明了Delphi/Object Pascal的巨大能力。

Delphi/Object Pascal并不是凭空出现的。它在基于坚实的计算机基础上被精心

设计，进而成为现代程序员的首选开发工具之一。它在快速可视化编程和生成程序的编译速度、语法的清晰度、可读性和表达能力之间提供了一个理想的、近乎完美的平衡。

Delphi/Object Pascal所建立的基础是古老的Pascal系列编程语言。就像谷歌的Go，或苹果的Objective-C，都是植根于C的语言一样，Object Pascal也是植根于传统经典的Pascal语言。

接下来的简介包括了Delphi/Object Pascal语言家族的简史和围绕Pascal、Turbo Pascal、Delphi的Pascal和Object Pascal的各个开发工具。虽然学习Delphi前端WEB开发并不是真的需要读这个，但肯定值得了解Delphi/Object Pascal语言的演变和它今天的发展状况。

我们今天在Embarcadero开发工具中使用的Object Pascal编程语言是在1995年2月14日情人节Borland公司在美国旧金山正式发布Delphi 1时发明的，当时它是一种全新的可视化编程开发环境。第一种面向对象Pascal语言是从已经在Turbo Pascal产品中使用的语言扩展而来。Borland公司并没有发明Pascal，它只是帮助它变得非常流行，并扩展了它的基础语法，以克服它与C/C++语言相比的一些局限。

下面的章节涵盖了该语言的历史，从Wirth教授最初发明的Pascal到最近基于LLVM的Delphi的Object Pascal编译器，以及用于ARM芯片和移动设备开发，Delphi承诺未来也将支持WebAssembly开发。这将有助于Delphi成为WEB开发的有力竞争者。

## Wirth教授的Pascal时期

Pascal语言最初是由瑞士苏黎世理工学院的教授Niklaus Wirth在1971年设计的。最完整的Wirth传记可以在<http://www.cs.inf.ethz.ch/~wirth>中找到。

Pascal被设计为Algol语言的简化版本，设计的初衷是主要用于教育目的。Algol本身创建于1960年。当Pascal被发明的时候，许多编程语言已经存在，但只有少数几种被广泛使用，例如：FORTRAN、Assembler、COBOL和BASIC。新语言的关键思想是结构化编程，通过数据类型、变量声明和结构化程序控制的强大概念来管理。

这种语言也被设计成一种教学工具，也就是说，使用最佳实践来教授大学生编程。不用说，Wirth教授的Pascal的核心原则对所有编程语言的历史都产生了巨大的影响，远远超出了那些仍然基于Pascal语法的语言。至于教学语言，学校和大学往往遵循其他标准（如工作要求或工具供应商针对大学教师提供的免费教育版），而不是看哪种语言更能帮助学习编程的关键概念。

## Turbo Pascal时期

Borland公司举世闻名的Pascal编译器，称为Turbo Pascal，于1983年推出，开发依据是《Pascal：用户手册和报告》，作者是Jensen和Wirth。Turbo Pascal编译器是有史以来最畅销的编译器系列之一，由于它兼顾了简单、强大和价格之间的平衡，使得这种语言在PC平台上特别受欢迎。原作者是安德斯-海尔斯伯格（Anders Hejlsberg），他后来又开发了非常受欢迎的编程语言C#和TypeScript。

Turbo Pascal引入了集成开发环境（IDE），在这里你可以编辑代码，运行编译器，查看错误，并跳回到包含这些错误的行。现在听起来很简单，但以前你必须退出编辑器，回到DOS；运行命令行编译器，写下错误行，打开编辑器，然后找到错误行。此外，Borland公司以49美元的价格出售Turbo Pascal，而微软的Pascal编译器售价为几百美元。Turbo Pascal多年来的成功促使微软最终放弃了其Visual Studio中的Pascal编译器产品线。实际上，你可以从Embarcadero开发者网络的博物馆部分下载一份Borland的Turbo Pascal的原始版本：<http://edn.embarcadero.com/museum>，值得一提的是著名的图像处理软件PhotoShop 1.0版就是使用Turbo Pascal开发的。包括著名的苹果电脑最初的操作系统也是使用Pascal开发的，直到苹果公司为了乔布斯回归苹果收购了Objective-C而放弃了Object Pascal，这使得Borland公司几乎垄断了Pascal这种编程语言。



图 Turbo Pascal 7.0

## Delphi Object Pascal早期



微信扫码关注“泰山老父”公众号，获取《Delphi Web前端开发教程》全本版

经过9个版本的Turbo和Borland Pascal编译器，Borland逐渐将该语言扩展到面向对象的编程（OOP）领域，Borland在1995年发布了Delphi 1.0，将Pascal变成了一种可视化的编程语言，并引入了可能是有史以来最好的Windows库：Visual Component Library或称VCL框架。Delphi以多种方式扩展了Pascal语言，包括许多面向对象的扩展，这与其他风格的Object Pascal不同，包括Borland Pascal with Objects编译器（Turbo Pascal的最后化身）中的那些。

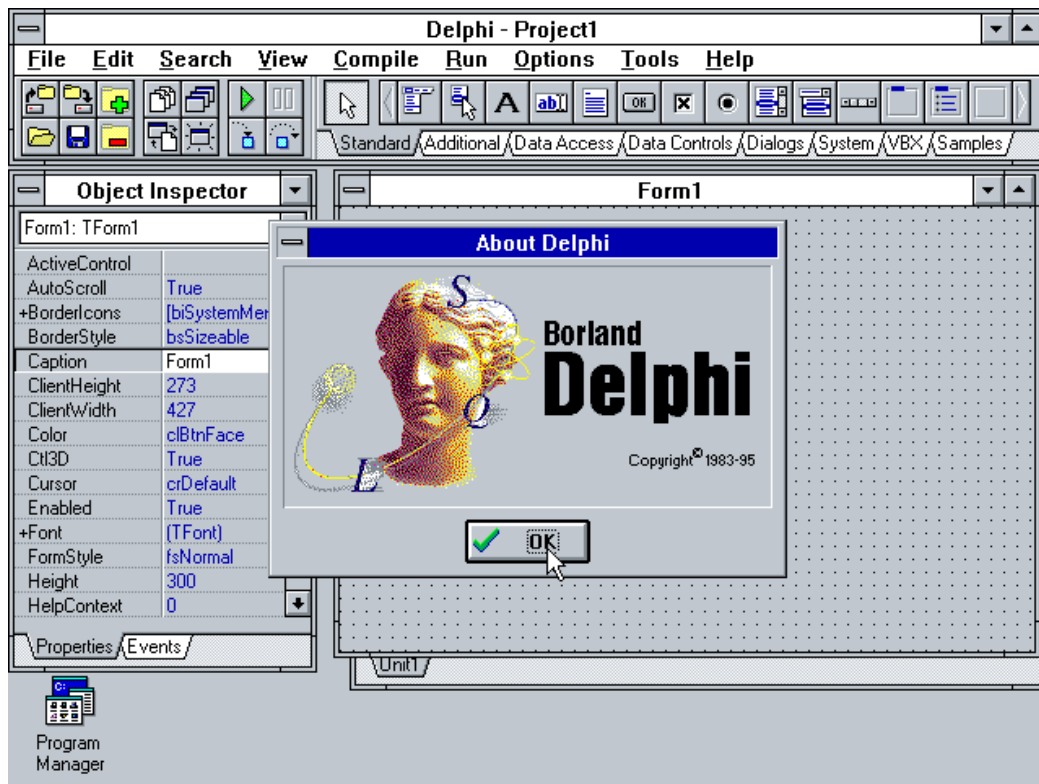


图 Delphi 1.0

1995年对于编程语言来说确实是一个特殊的年份，因为它见证了Delphi的Object Pascal、Java、JavaScript和PHP的首次亮相。这些是一些至今仍在使用的最流行的编程语言。事实上，其他大多数流行的语言（C、C++、Python、Objective-C和COBOL）都更古老，而唯一较新的流行语言是C#。

Borland将Pascal编译器带到了32位世界，实际上是重新设计了它，以提供一个与C++编译器通用的代码生成器。这将许多以前只在C/C++编译器中才有的优化功能带到了Pascal语言中。

在Delphi 3中，Borland在语言中加入了接口的概念，使类的表现力和它们的关系有了飞跃性的发展。随着Delphi 7.0的发布，Borland公司正式开始将Object Pascal语言称为Delphi语言，但当时该语言并没有真正的改变。当时Borland还创建了Kylix，一个适用于Linux的Delphi版本，后来又为Delphi 8所包含的微软.NET框架创建了一个Delphi编译器。这两个项目后来都被放弃了，但2003年底发布的Delphi 8标志着对话

言的一系列非常广泛的变化以支持.NET，这些变化后来被Win32 Delphi编译器和所有其他后续编译器所采用。

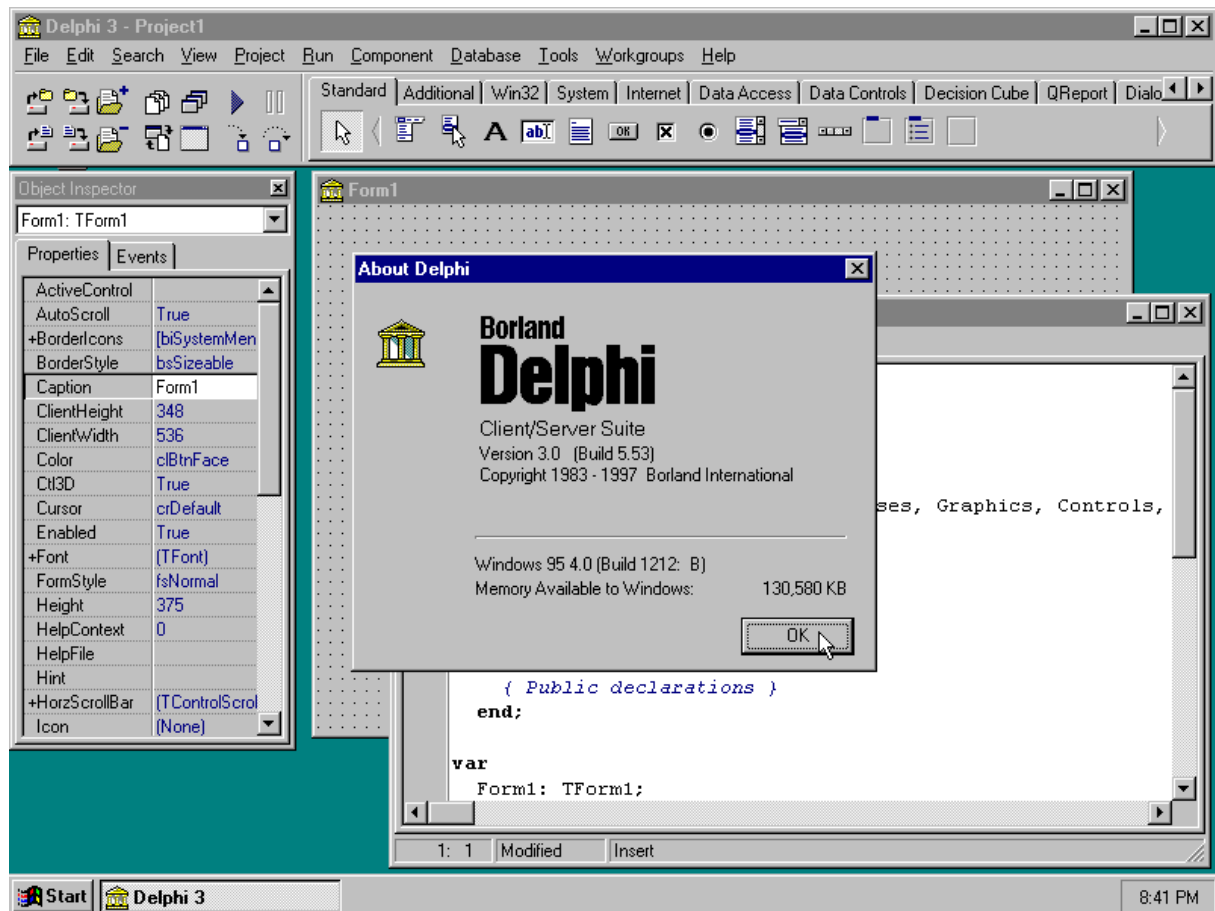


图 Delphi 3

## 从CodeGear到Embarcadero的Object Pascal时期

由于Borland公司对其在开发工具方面的投资前景缺乏信心，后来的版本如Delphi 2007，是由母公司的子公司CodeGear负责开发的。这个子公司（或业务部门）后来被出售给了Embarcadero科技公司。在Delphi 2007版本之后，公司重新专注于发展和扩展Object Pascal语言，并增加了期待已久的功能，如Unicode支持（在Delphi 2009中）、泛型、匿名方法或闭包、扩展的运行时类型信息或映射，以及许多其他重要的语言功能。同时，除了Win32编译器，该公司还推出了Win64编译器作为Delphi XE2的一部分，以及macOS编译器，在Borland和Kylix的失败尝试之后，Delphi重新回到了跨平台战略，即他们的Delphi for Linux版本。然而，这一次的想法是拥有一个单一的Windows开发环境并交叉编译到其他平台，后来的事实证明了这一策略是正确的。对Mac的支持只是该公司多设备战略的开始，它拥抱了桌面和移动平台，如iOS和Android。这一战略是通过采用新的跨平台GUI框架（称为FireMonkey）而实现的。

## 支持移动开发领域

以前所有的Delphi版本都是以Intel x86 CPU为目标，而向移动领域的转变则引入了第一个针对ARM芯片的Object Pascal编译器。这一转变导致了编译器和相关工具的整体重新架构，即“编译器工具链”，其基础是开源的LLVM编译器架构。

与Delphi XE4一起发布的iOS的ARM编译器是第一个基于LLVM的Object Pascal编译器，但也是第一个引入一些新功能的编译器，如自动引用计数（ARC），后来又被开发团队从该语言中删除。接下来，在同一年（2013年），Delphi XE5增加了对Android平台的支持，有了第二个基于LLVM的ARM编译器。总而言之，Delphi XE5为Object Pascal语言提供了6个编译器（用于Win32、Win64、macOS、Mac上的iOS模拟器、iOS ARM和Android ARM支持）。所有这些编译器都是今天的Delphi的一部分，并且大体上支持通用的语言定义。

在2014年的前几个月，Embarcadero发布了一个新的开发工具，基于相同的核心移动技术，并称为AppMethod。2014年4月，该公司还发布了XE6版本的Delphi，而2014年9月，AppMethod和Delphi XE7第三次发布，随后，在2015年春季，Delphi XE8发布，其中包括第一个ARM 64位编译器，针对iOS。

## Delphi 10. x时期

在Delphi 10 Seattle之后，随着公司被Idera公司收购，Embarcadero创建了10. x系列版本。Delphi 10.1 柏林、Delphi 10.2 东京、Delphi 10.3 里约和Delphi 10.4 悉尼。在此期间，Embarcadero增加了对新目标平台和操作系统的支持，即Linux 64位、Android 64位和macOS 64位。该公司还重新关注Windows VCL框架，增加了对Windows 10操作系统的具体支持。在10. x系列的过程中，Embarcadero继续拓展Object Pascal语言，引入了内联变量声明和自定义管理记录等功能，以及许多其他较小的改进。

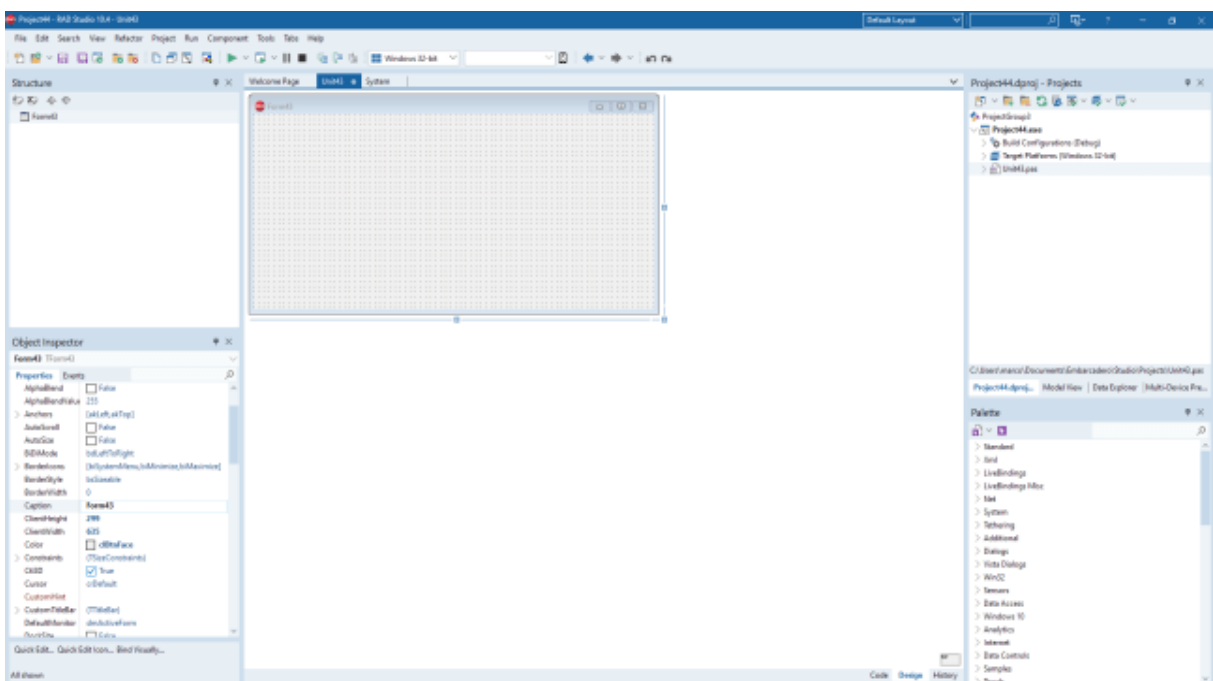


图 Delphi 10.4.2



## Delphi 11的发布

随着微软Windows 11的发布和苹果公司放弃了10.x macOS操作系统的长系列版本，Embarcadero决定不再采用10.x系列，而是采用从11 Alexandria开始的主要版本的顺序编号，从而与此时的Windows 11版本号相匹配。这样做是为了强调产品与Windows操作系统的紧密联系，即使在接受了多设备跨平台开发作为现代Delphi的核心原则之后。在Delphi 11中值得注意的是对macOS ARM-64位平台的支持，为苹果M1 CPU生成原生代码，并首次推出了支持高DPI的IDE。

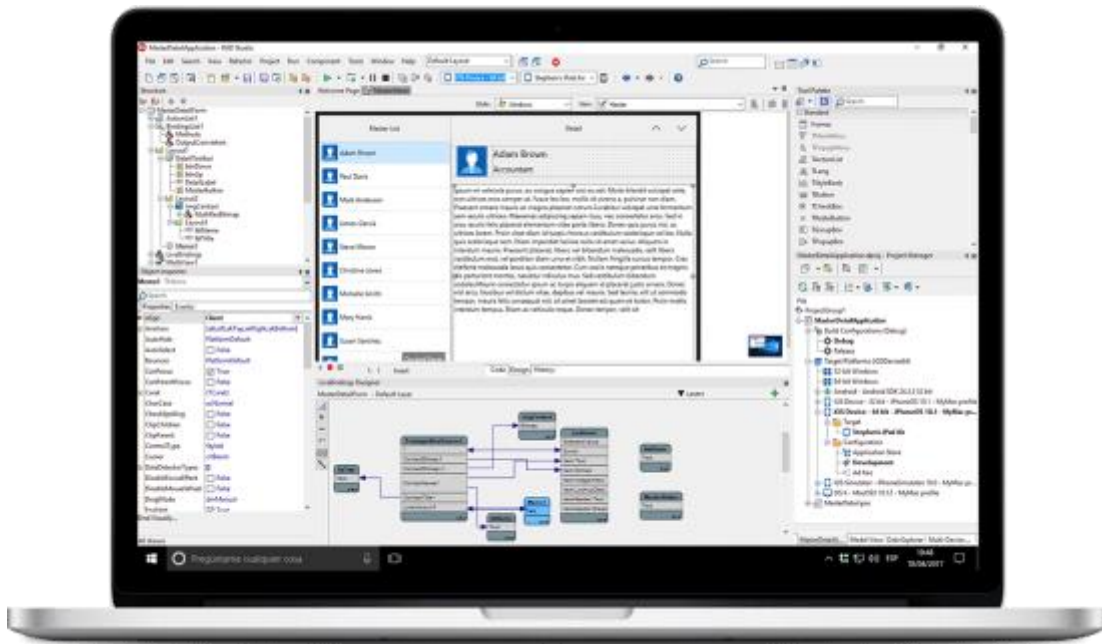


图 Delphi 11 IDE

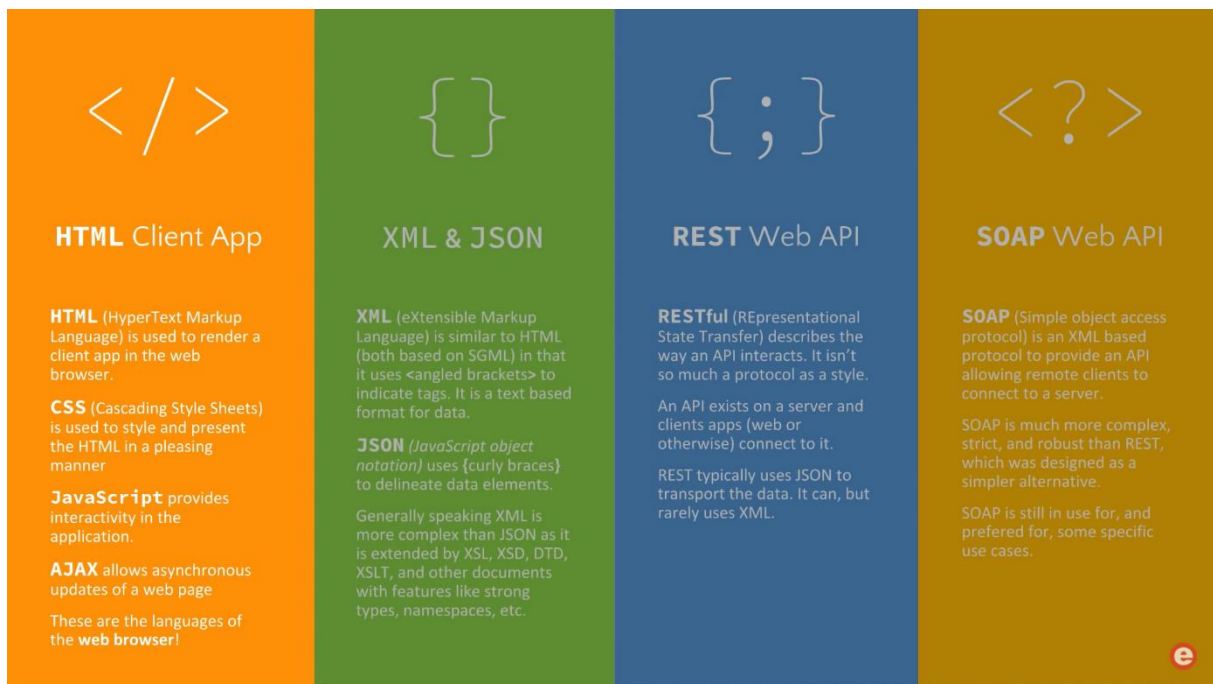
VCL 和 FireMonkey (FMX) 是成熟的 GUI 库。VCL 专注于原生 Windows 开发，在您的Windows电脑中至少有一个应用软件是用 Delphi 和 Visual Component Library 构建的。例如，KMPlayer、WinRAR、MySQL管理工具、AIMP、BurnAware、Dev-C++、EarMaster、FL Studio、Nero Burning Rom、Ultra ISO 以及更多应用软件。而 FireMonkey 为 Windows、Linux、macOS 甚至 Android 带来了强大灵活的 GUI 框架。通过Embarcadero 新发布的免费 Python 模块可以将 Delphi 的 GUI 框架的强大功能和灵活性带入 Python。

## Delphi WEB开发三剑客IntraWEB、uniGUI、TMS WEB Core

长期以来，Delphi支持者一直将Delphi的市场占有率下滑归罪于开发团队对Web开发的重视不够，也因为这个原因，Delphi第三方组件厂商为了弥补Delphi官方在WEB开发上的不足，推出了众多WEB开发解决方案，其中最流行的三大WEB开发框架，分别是：塞浦路斯AtozedSoftware公司开发的IntraWEB、土耳其飞漫软件公司开发的uniGUI、比利时TMS软件公司开发的TMS WEB Core，当然还有其它一些开源Web框架，笔者不再

详述，但在下面的幻灯片中会有提及。

## WEB开发基础知识入门



### HTML 客户端应用（这些是网络浏览器的语言！）

**HTML**（超文本标记语言HyperText Markup Language）是一种用于创建网页的标准标记语言。HTML用来建立自己的WEB站点，HTML运行在浏览器上，由浏览器来解析用于在网络浏览器中呈现客户端应用程序。

**CSS**(Cascading Style Sheets) 用于渲染HTML元素标签的样式，被用来以一种舒适的方式对HTML进行样式和展示。

**JavaScript** 在应用程序中提供互动性。

**AJAX** 允许网页的异步更新。

### XML & JSON

**XML**（可扩展标记语言）与HTML（两者都基于SGML）相似，它使用<角括号>来表示标签。它是一种基于文本的数据格式。

**JSON**（JavaScript对象符号）使用{大括号}来划分数据元素。

一般来说，XML比JSON更复杂，因为它被XSL、XSD、DTD、XSLT和其他具有强类型、命名空间等特征的文档所扩展，目前主流的开发方案都是使用JSON。



## REST Web API

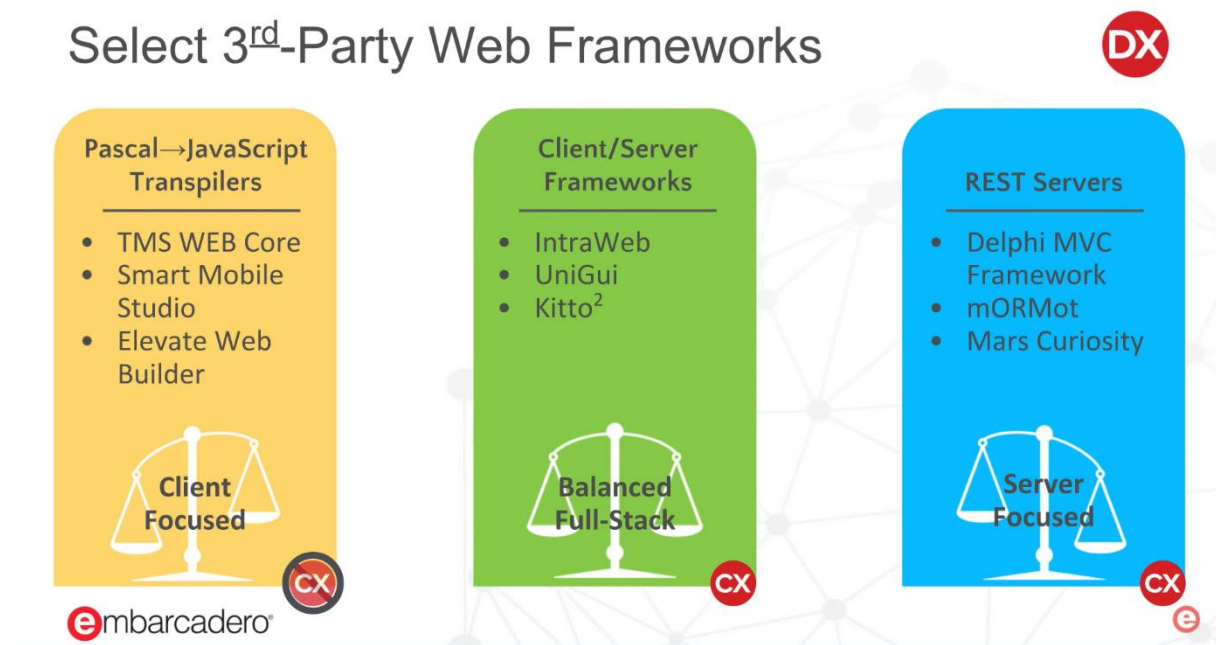
RESTful (REpresentational State Transfer) 描述了一个API的交互方式。它与其说是一种协议，不如说是一种风格。一个API存在于服务器上，客户端应用程序（网络Web或其他）连接并调用该 WEB API。REST通常使用JSON来传输数据，一般很少使用XML。

## SOAP Web API

SOAP（简单对象访问协议）是一个基于XML的协议，提供一个允许远程客户连接到服务器的API。

SOAP比REST更复杂、更严格、更稳健，REST是作为一个更简单的替代方案而设计的。SOAP仍然在使用，而且在一些特定的使用情况下更受开发者欢迎。

## Delphi第三方WEB开发框架



### 1、Pascal → JavaScript 转译框架

Pascal到JavaScript的转译器框架特点：

- 用Object Pascal（基于Delphi的语言）编写，并将其翻译/编译成JavaScript；
- 通常基于pas2js或DWScript；

[delphitools.info/dwscript](http://delphitools.info/dwscript) (没有“发布/订阅”模型)

[wiki.freepascal.org/pas2is](http://wiki.freepascal.org/pas2is) (配置比较复杂)

- 包括大型支持库;
- 创建一个纯粹的网络前端解决方案, 使用HTML、JavaScript和CSS或者其他JavaScript框架;
- 与许多流行的服务器解决方案兼容;
- 不需要开发人员了解JavaScript知识;

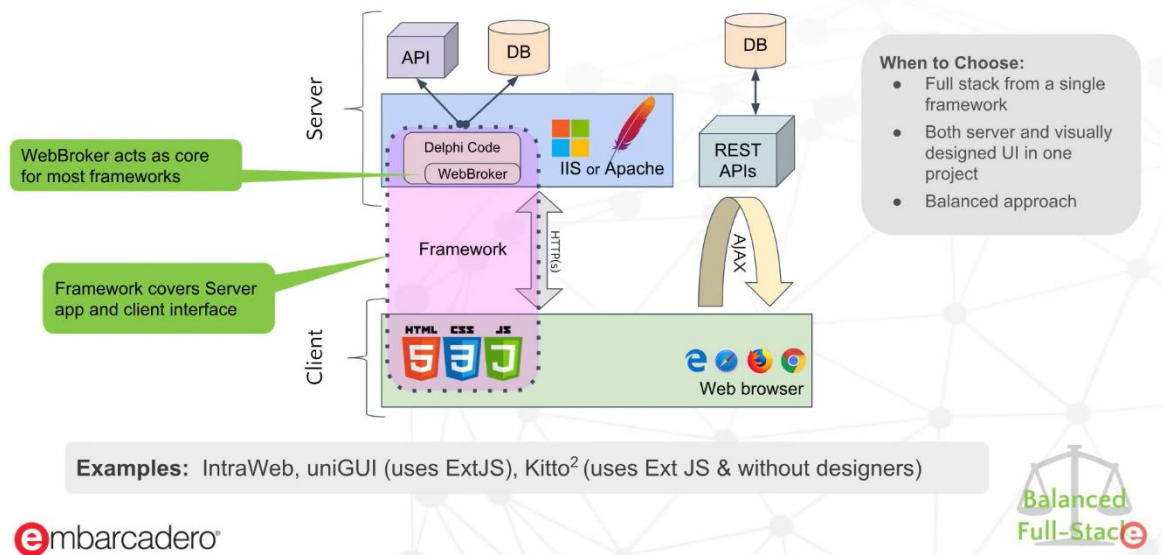
主要框架厂商: TMS WEB Core、Smart Mobile Studio、Elevate Web Builder、西安碧树西风DeWeb(delphibbs.com)、浙江星五WebPascal等。

## 2、Client/Server Frameworks 客户端(前端)/服务端 全栈框架

客户/服务器框架特点:

- 来自单一框架的全栈开发;
- 在一个项目中既可以开发服务器端, 又有前端用户界面视觉设计器;
- 性能均衡;

### Client/Server Frameworks



主要框架厂商: IntraWeb、uniGui、Kitto2 (使用Ext JS, 没有界面设计器)。

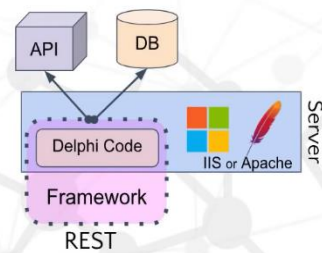
## 3、REST Servers服务端(后端)框架

REST服务端特点:

- 为远程资源提供一个REST API接口。也可以为其他网络内容提供服务；
- 包括在Delphi Enterprise & Architect企业版和架构师版中的RAD服务器、DataSnap、WebBroker；
- 开源框架：Delphi MVC Framework、MARS-Curiosity、Synopsis mORMot ORM/SOA/MVC Framework；

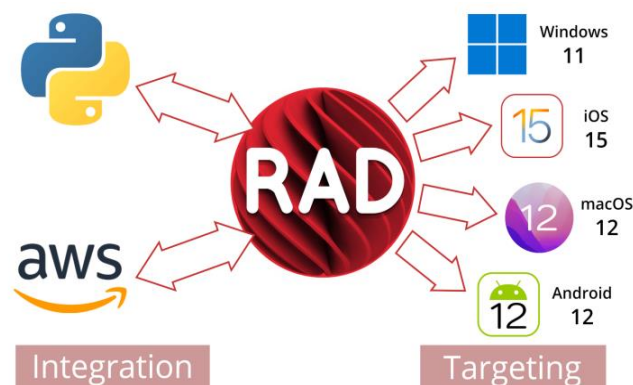
## REST Servers

- Provides a REST API interface to remote resources
  - Can also serve other web content
- Included with Delphi Enterprise & Architect
  - RAD Server
  - DataSnap
  - WebBroker
- Open source projects:
  - Delphi MVC Framework
  - MARS-Curiosity Delphi REST Library
  - Synopsis mORMot ORM/SOA/MVC Framework



embarcadero®

- 国产Delphi中间件厂商（排名不分先后）：成都QQ600585开源ZNet (ZServer4D)、DIOCP、杭州快驴、全能中间件、咏南中间件、ABFrameWork、西安高勇、火神中间件等（这里只列出笔者知道的国产中间件，各家的价格和性能请自查。）。。



善于利用这些开源或者商业中间件能节省开发团队几个月的时间和金钱成本，帮助公司快速赢得客户的好评，这些Delphi中间件的大多数性能超越了go、Node.JS等产品。使用Delphi点击几下鼠标就能快速可视化设计开发项目的前端和后端，有了Delphi和VCL/FMX框架，跨平台地实现全流程的实时可视化快速软件开发就成为了可能。软件行业的初学者们，远见者们——未来的大门已经敞开了。

## 各个框架特点比较

Framework	Need Client JavaScript	RAD IDE Integration	Visual UI Designer	License & Availability	Includes Client GUI	Includes Server	Server Platforms		
							Windows	Linux	Other
WebBroker	N/A	Yes	No	Included	No	Yes	Windows	Linux	Android*
DataSnap	N/A	Yes	No	Included	No	Yes	Windows	Linux	
Ethea's Kitto2	No	Both	No	Open Source	Yes	Combined	Windows	Linux	
Mars-Curiosity	N/A	Yes	No	Open Source	No	Yes	Windows	Linux	
MVC Framework	N/A	Yes	No	Open Source	No	Yes	Windows	Linux	
RAD Server	N/A	Yes	No	Included	No	Yes	Windows	Linux	
AtoZed IntraWeb	Optional	Yes	Yes	Free v14 (or buy v15 & v17)	Combined	Combined	Windows	Linux	(v15+ ), Apache, IIS
Elevate Web Builder	No	Standalone	Yes	Commercial	Yes	Compatible	Windows		
FMSoft UniGui	Optional	Yes	Yes	Commercial	Combined	Combined	Windows		
Synopse mORMot	N/A	Yes	No	Open Source	No	Yes	Windows		
Sencha Ext JS	Yes	Standalone	Yes*	Included*	Yes	Compatible			Browser
TMS WebCore	No	Yes	Yes	Commercial with Free CLC	Yes	Compatible			Browser
Smart Mobile Studio	No	Standalone	Yes	Commercial with Free CLC	Yes	Compatible			Any

- ✓ WebBroker Android isn't officially supported (limited testing), and other WebBroker based frameworks may work there too
- ✓ IntraWeb added Linux support in v15 & v17, while v14 is Windows only
- ✓ mORMot Linux is via FPC for now
- ✓ Elevate Web Builder support for the Elevate Web Server is Windows only, but could work with other servers on other platforms

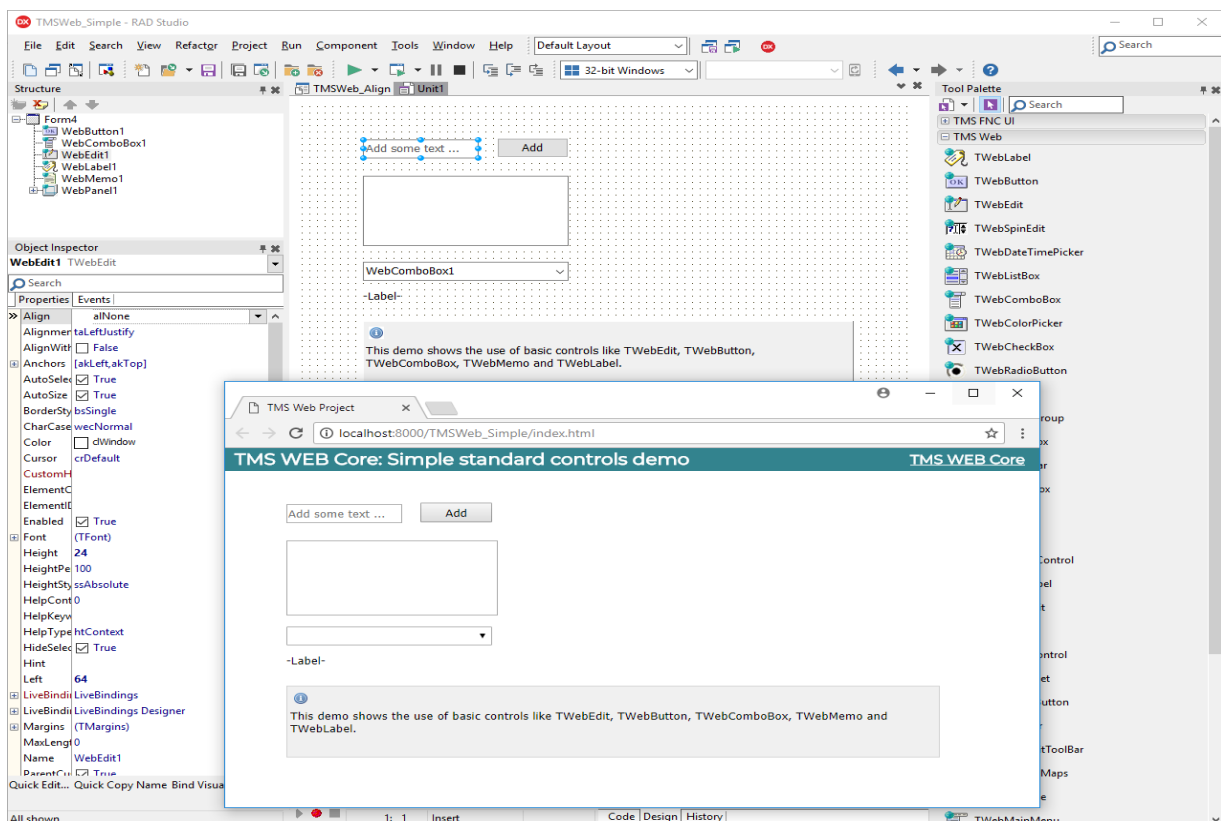


## Server Platforms

Windows support typically includes IIS or Apache, and possibly Windows Service or Standalone server.

Linux uses Apache.

根据笔者对这三大框架的开发经验，IntraWEB适合于WEB前、后端的开发，其自带的网络服务器非常强大、稳定，笔者使用Cesium框架开发的WEB GIS地理信息系统前端不需要Apache Tomcat或Nginx即可稳定运行；uniGUI是对JavaScript库Sencha ExtJS的封装，它带有两套VCL组件包，分别支持桌面端和手机端Web页面，更适合于开发网络版企业管理软件、ERP软件、财务进销存软件等，比如将Delphi开发的数据库管理软



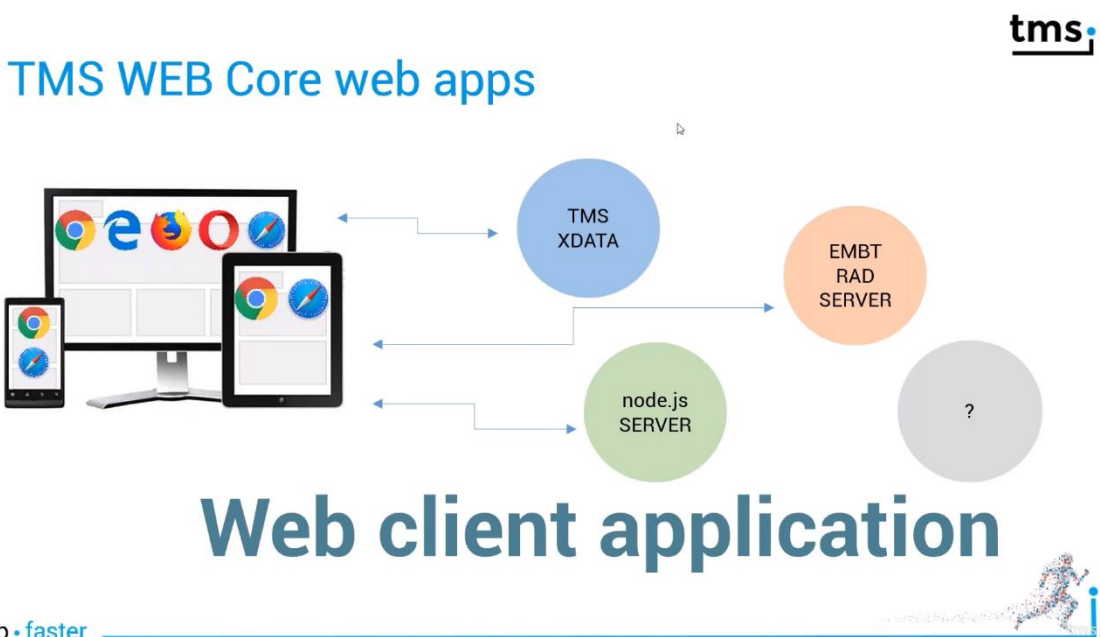
件会更容易的转换成uniGUI网络版；TMS WEB Core适合开发传统的企业网站、前端、以及IOS、Android、树莓派、Linux、PC端等应用，理论上可以封装包括Sencha ExtJS在内的任何第三方JS库（比uniGUI的封装方案更好），其主要特点如下：

## 业界领先的网络框架

- 现代 SPA Web 单页应用程序模型，与流行的 JavaScript 框架和 CSS 样式兼容，具有大型可重用数据库访问组件包。支持 TMS XData、RAD Server 和其他编程语言服务端以及各种国产中间件；
- 从 Delphi/Object Pascal 生成基于纯 HTML5/CSS3/Javascript 的 Web 应用；
- 用于常见 UI 界面控件和访问浏览器功能的标准组件框架；
- 通过浏览器直接调试 Pascal 代码；
- 由经过多年开发的可靠且经过验证的 Delphi Pascal 到 Javascript 开源编译器 pas2js 提供后台支持；
- 利用 Electron 或 Miletus 框架构建跨平台桌面应用程序；
- PWA 支持：开始构建开箱即用的渐进式 Web 应用程序；

## 重用技能和组件

- 集成在 Delphi IDE 中的基于组件的 RAD 快速可视化开发；
- 一个真正具有革命性和创新性的 TMS FNC 组件框架，现在也支持 Web 开发，允许创建可在 VCL、FMX、LCL 和 WEB 上使用的 UI 控件；
- 开放以使用其他现有的 JavaScript 框架和库；
- 开放使用 HTML/CSS 进行设计；
- 开放使用其他 jQuery 控件甚至其他 Javascript 框架；
- 为 jqWidgets 库中的 jQuery 框架提供 Pascal 类包装器；
- 轻松连接到 REST 云服务，包括连接到数据库的 TMS Xdata 和各种开源的 Delphi REST 框架，例如：mORMot 等；





## 轻松部署分发

- 应用程序仅由 HTML 和 JavaScript 文件组成，可以轻松部署在任何轻型或重型网络服务器上；
- 使用任何现有的负载均衡软件或技术以获得最高性能；
- 包含小巧方便的调试网络服务器，用于快速 RAD 开发；

综上所述，学习使用 Delphi & TMS Web Core，您可以：

- 长期支持和持续升级的VCL开发框架，让客户对项目更有自信的长远规划，强大的功能使公司能够轻松构建自定义工具或扩展组件，在客户选择的平台上进行开发，保护客户的源代码不被利用，并确信客户的应用程序可以维护数十年；
- 创建最安全的本机原生应用软件，与VCL框架一样的长期稳定性和向后兼容性。通过反编译复杂度保护项目源代码来测试支持、可访问性和知识产权安全；
- 涵盖各种开发需求的数百种组件，有效帮助软件快速开发上市，灵活的FMX跨平台框架允许企业瞄准广泛的用户，为任何领域构建软件，并毫无障碍的访问所有硬件和操作系统功能；
- 通过强大的所见即所得Delphi IDE进行可视化开发，借助数百个组件和强大的可视化设计形式，开发人员可以在几秒钟内制作原型并构建 GUI 应用程序。此外，编辑器允许您根据需要使用代码更改 UI 元素；
- 帮助开发团队生产力提高5倍，换句话说1个Delphi程序员抵得上5个Java程序员的开发效率，这是对开发人员完成典型开发任务所需的工作量和代码的度量。生产力可以通过两种不同的方式实现——由于Delphi强大的本机组件和开发库以及 IDE 工具（如代码完成和可视化设计）降低了开发人员编码难度；
- Delphi是真正的全栈万能编程语言，具有更强灵活性和功能性，Delphi可伸缩性、数据库访问、对设备特定功能的访问以及目标平台，远超其他开发工具；
- 仅使用一个代码库为 5 种不同的操作系统Windows, Android, iOS, macOS, 和Linux开发应用软件，同时使用Delphi开发构建本机应用程序，有效解决一些安全问题。例如，Electron 和 C#、Java应用程序很容易被反编译。从 UI 元素到产品的核心业务逻辑，而使用Delphi构建的应用程序除了UI之外很难进行逆向工程，Delphi反编译后的代码是汇编代码，一般人很难读懂；
- 开发团队只需熟悉一种Delphi编程语言技能，同时开发后端、前端、桌面、移动应用、Web应用。
- 更优雅、更适合阅读的源代码，当开发人员用 Delphi 编程语言编写代码时，就像

用英语写一首诗一样。此外，所有命令和关键字都很直观，广泛的现代语法糖和久经考验的强大“指针”丝毫不会限制程序员的开发能力；

- 作为物联网时代的开发工具，通过Delphi自带的本机物联网传感器组件，可以直接访问硬件，例如：GPS、摄像头、麦克风、蓝牙设备、MQTT、串口、Modbus通信、PLC设备，并且执行速度更快，为树莓派、嵌入式Linux开发强大的物联网应用。

TMS Web Core 允许您使用Delphi作为“后台”编程语言来创建完全交互式的网页。当您点击编译按钮而不是创建传统的 Windows 可执行文件时，您将获得带有 CSS 的纯 HTML 页面，只需要准备好复制到普通的常规 Web 服务器发布即可——除此之外没有其他任何要求。

Web 应用程序的可编程功能以 JavaScript 文件的形式提供。您不必编写甚至学习 JavaScript——您只需编写 Delphi 代码，编译器就会为您将该代码转换为 JavaScript。

TMS Web Core 真正突出了 Delphi 语言在编写程序时是多么简单而强大。“强类型”语言是秘诀的一部分，而 RAD 中的“R”真正表明了“快速”的意思。使用 Delphi可以频繁且重复地生成健壮、高性能且功能丰富的应用程序，而且花费的时间和工作量极少。TMS Web Core 将这种可视化快速应用开发的功能扩展到使用 Delphi 创建交互式网站。

我甚至无法告诉你 TMS Web Core 对 Delphi 程序员来说是如何改变游戏规则。Delphi早期也有一些值得信赖的替代品，例如古老的IntraWeb、uniGUI等，但 TMS Web Core 对我来说感觉更“正确”，真正的体验是创建一个常规的 Delphi 应用程序，它恰好作为一组 HTML 页面弹出。没有混乱，没有大惊小怪，并且很容易在您选择的任何托管服务上安装到任何服务器主机上，无论是Windows或者Linux独立服务器还是虚拟主机。

Delphi是最简单、开发速度最快的编程语言开发工具，用于在 Windows、MacOS、Linux、iOS 和 Android 操作系统上开发简单或专业的应用程序。初学者可以通过其广泛的示例、教程、帮助文件和对代码的LSP支持轻松学习。Delphi带有用于高性能原生 Windows应用程序开发的屡获殊荣的VCL框架和用于跨平台UI的强大的FireMonkey (FMX) 框架。官方免费提供一个Delphi社区版供学生、初学者和初创公司使用；

Delphi王者归来，重装上阵！Delphi拥有27年不断进化发展的强大生命力，数量庞大的第三方组件，活跃的全球开发者生态链，乐于助人的Delphi程序员社区，积极进取的研发团队，所有这一切都会极大的帮助Delphi开发者在移动互联网和物联网时代游刃有余！未来已来，欢迎加入Delphi开发者社区！

**(试读结束，全文共486页，如需全文版本，请微信搜索关注“泰山老父”公众号获取)**

微信扫码关注“泰山老父”公众号，获取《Delphi Web前端开发教程》全本版



### 郑重声明

本文档为企业内部培训资料，免费赠阅，仅供个人学习和研究使用，涉及的软件工具及文档资料均来自互联网，版权归相应公司所有，如需商用请购买正版，如有侵犯您的商标权、著作权和/或其他合法权利的，请联系我公司，我司法务专员将依法处理，本公司不承担任何法律责任。

济南战石电子商务有限公司