

第16讲 | 热点剖析（三）：试试看，你也可以编写一款HTML5小游戏！

2018-06-30 蔡能

从0开始学游戏开发

[进入课程 >](#)



讲述：蔡能

时长 07:51 大小 3.64M



前两周，我分析了[HTML5 的核心竞争力](#)以及[HTML5 游戏的引擎](#)。选择好了 HTML5 引擎，我们就可以开始制作游戏了。

对于编写 HTML5 小游戏，你或许会很有兴趣，因为 HTML5 现在已然是一个潮流，而且利用引擎编写 HTML5 游戏已经变得相当方便。

如何选择一款引擎？

我选择的是比较容易上手的 lufylengend 引擎。为什么要选择这款引擎呢？因为它只需要下载一个压缩包，并且不需要特别繁琐的说明和设置就能直接上手，用作我们的教学示例是最合适的。

如果使用白鹭引擎或者 Cocos-2d 引擎这些比较有名的引擎，可能会有这些问题。

1. 这些引擎从工具到套件都非常成熟，你直接下载一个引擎启动器或者组件管理器，就可以一应俱全地在启动器里面下载，配置到默认路径。但是，这些工具拥有纷繁复杂的界面，你连上手都要费一段时间，更别说短时间内熟练使用并制作出一款游戏。
2. 这些引擎需要引入的库或者使用的方式极为系统，所以你需要系统地引入库文件，才可以使用。事实上我要做的示例，并不需要很多复杂的东西，你只需要跟我从头到尾走一遍，就能明白编写 HTML5 游戏是怎么回事。
3. 这些引擎需要别的工具支持，比如 node.js。作为新手来说，光配置 node.js 就是一项比较麻烦的工作。所以我选择了 lufylengend 引擎这一个比较“单纯的”引擎来作为教学引擎。

几个简单的说明

你可以从这个地址下载最新版本：

<https://github.com/lufylegend/lufylegend.js/archive/lufylegend-1.10.1.zip>。下载下来的安装包大约有 30M 大，解压缩后有 36M 左右。解压缩后，我们会看到一系列的 js 文件。

我先对这些文件做一些说明，并且对最基础的编程流程做一个简单的梳理。

压缩包内包含 lufylegend- 版本号.js 和 lufylegend- 版本号.min.js 这两个**完整版本**，还有 lufylegend- 版本号.simple.js 和 lufylegend- 版本号.simple.min.js 这两个**缩减版本**，其中带 min 字样的是去除了回车和空格的压缩版本。如果你使用 JavaScript 编程，缩减版本对于你来说再熟悉不过的了。

其中，simple 缩减版本与完整版本的区别在于，它将 LBox2d、LQuadTree、LTransitionManager、LoadingSample1、LoadingSample2、LoadingSample3、LoadingSample4、LoadingSample5 等几个类与 HTML5 引擎的常用部分分离，缩减了引擎的体积。如果需要用到被分离的部分功能的话，可以手动进行加载。

随后，可以在 HTML 代码中将 legend 引擎的库件引入。调用 LInit 函数，初始化库件。然后开始游戏代码的编写。剩下，你只需要有一款合适的 IDE 就可以开始编程了。

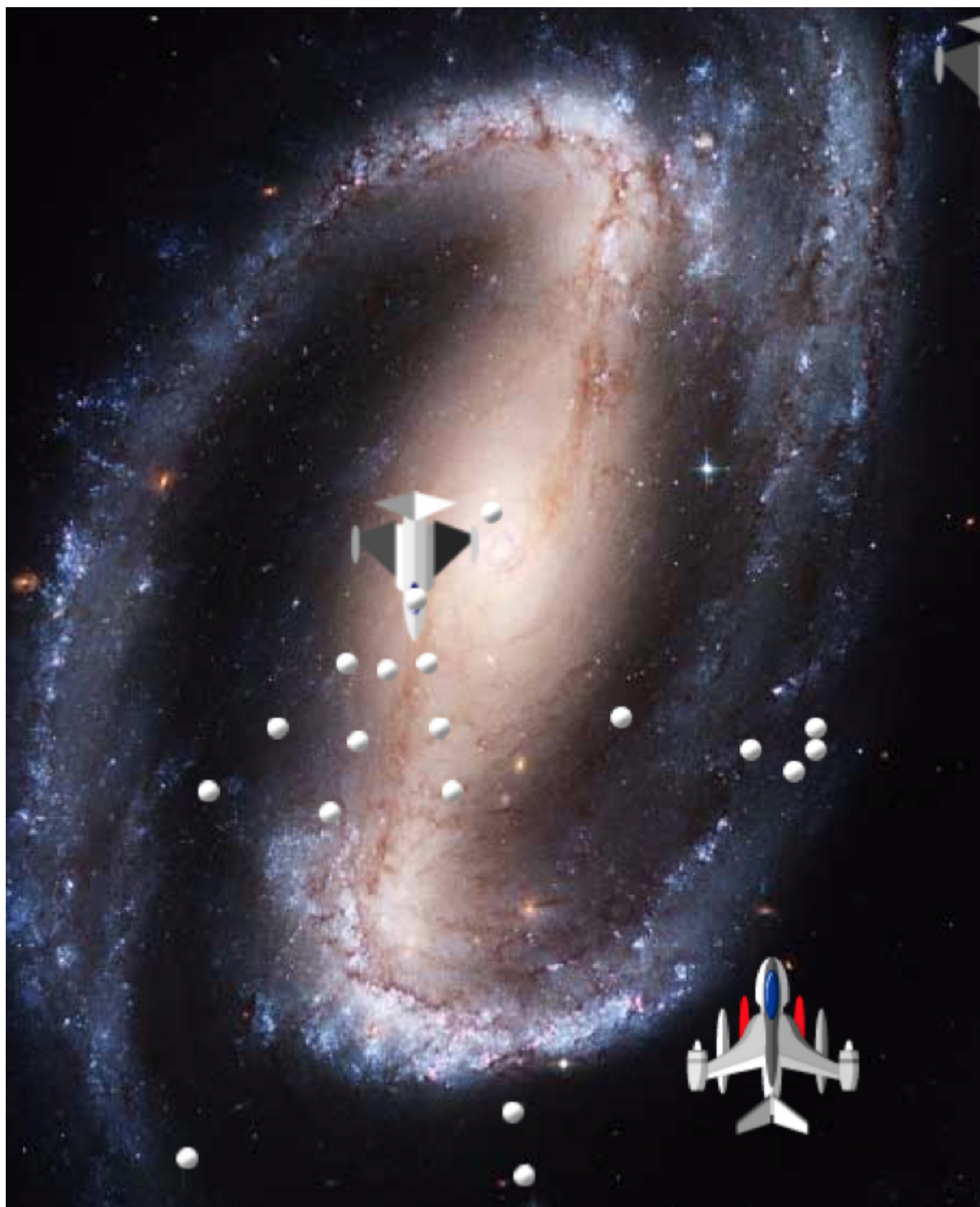
对制作 HTML5 游戏来说，你首先要做的是，将游戏场景在浏览器中呈现出来。把 Windows 的窗体，从系统客户端程序搬到浏览器上，呈现的窗体从可执行文件变成了浏览

器。从这个角度讲，** 浏览器担负了应用层的工作，浏览器本身担负了解释程序，并且渲染的过程，** 所以，从理论上讲，**相同的游戏类型和游戏内容，HTML5 游戏的渲染速度是比不上客户端游戏的渲染速度的。**

一起动手制作吧！

很凑巧的是，lufylengend 引擎也拥有一款打飞机 demo。我们只需要在 lufylegend 引擎目录的 examples/demo 下，找到 barrage 目录。

这一款打飞机的游戏，打开后是这个样子。



这和我们之前讲述的内容非常类似。那我就借解释一下这款游戏的代码，来教你制作。

在目录下，有一个 `index.html`，这就是游戏的开始页面。在这个页面下，我们先来看一下部分代码。

```
1 <script type="text/javascript" src="../../load_lufylegend.js"></script>
2 <script type="text/javascript">
3   if(LGlobal.canTouch){
4     LGlobal.stageScale = LStageScaleMode.EXACT_FIT;
5     LSystem.screen(LStage.FULL_SCREEN);
6   }
7 </script>
8 <script type="text/javascript" src="../js/Main.js"></script>
```

如果你熟悉 web 编程，对于这些代码肯定非常熟悉。在开始的地方，我们看到载入的 JavaScript 代码是 load_lufylegend.js，这个 js 文件包含在打飞机游戏的上一层目录，内部就只有一行代码。

 复制代码

```
1 document.write('<script type="text/javascript" src="../../../lufylegend-1.10.1.min.js">');
```

我们看到，在这个 js 文件内，包含了 lufylegend 的原始引擎文件。至于为什么要这么做，为什么要使用两个文件包含引擎，是因为这样可以保持游戏代码的干净。如果你要修改引擎的包含路径，不需要修改游戏本体文件，只需要修改 load_lufylegend.js 包含路径即可。


而 LGlobal.canTouch 这段话的意思是，如果是移动版本的话，设置缩放模式为适应屏幕，并且为全屏，代码是：

 复制代码

```
1 LSystem.screen(LStage.FULL_SCREEN)
```


最后，导入游戏脚本 Main.js 文件。

在 Main.js 里面，我们看到，它还包含了三个别的 js 文件，代码是这样。

 复制代码


```
1 imgData.push({type:"js",path:"../js/Global.js"});
2   imgData.push({type:"js",path:"../js/Bullet.js"});
3   imgData.push({type:"js",path:"../js/Plain.js"});
```

它包含了一个共有类 Global.js、子弹类 Bullet.js 以及飞机类 Plain.js。之后的代码是这样的。

 复制代码


```
1 loadingLayer = new LoadingSample1();
2 addChild/loadingLayer);
```

其中 LoadingSample1 是“载入进度条”类，我们可以在下面的代码看到载入的实现。

 复制代码


```
1 LLoadManage.load(
2 imgData,
3 function(progress){
4 loadingLayer.setProgress(progress);
5 },
6 function(result){
7 imglist = result;
8 removeChild/loadingLayer);
9 loadingLayer = null;
10 gameInit();
11 }
12 );
```

在载入结束后，就开始 gameInit 函数的调用，也就是游戏初始化。

 复制代码


```
1 function gameInit(event){
2 // 游戏底层实例化
3 backLayer = new LSprite();
4 addChild/backLayer);
5 ctrlLayer = new LSprite();
6 addChild(ctrlLayer);
```

LSprite 是引擎的基本显示列表构造，里面包含了显示内容的列表节点，addChild 就是将显示内容添加到节点列表。

 复制代码


```
1 // 添加游戏背景
2 bitmapdata = new LBitmapData(imglist["back"]);
3 bitmap = new LBitmap(bitmapdata);
4 backLayer.addChild(bitmap);
```

这几段代码就是将背景图片也添加到显示节点列表。

 复制代码

```
1 // 得分显示
2 pointText = new LTextField();
3 pointText.color = "#ffffff";
4 pointText.size = 20;
5 pointText.text = point;
6 backLayer.addChild(pointText)
```


这是一个得分的显示，所以需要新建一个文本类，并设置颜色和大小，并将之放到显示节点的列表。

 复制代码

```
1 // 加入玩家
2 player = new Plain("player",0,200,600,[5]);
3 backLayer.addChild(player);
```

我们需要新建一个玩家类。新建玩家，其实就是新建一个飞机类型，所以我们在这里看到一个 plain 类的创建。


这个创建函数的实现原型是这样的。

 复制代码

```
1 function Plain(name,belong,x,y,bullets)
```

你可能会觉得奇怪，Plain 是什么意思，在它的 demo 里面，Plain 是飞机的意思，然而可能是作者的一时疏忽或者是英文“捉急”，所以就把 Plane 写成了 Plain。以下所有和飞机相关的代码都是 Plain，虽然并不影响代码的运行，但是出于严谨考虑，我在这里更正一下，Plain 等于 Plane。

第一个参数是名字，第二个参数是飞机所属，表明是属于敌人还是玩家，随后两个参数 (x,y) 是飞机在 2D 画布上所显示的位置，最后一个 bullets 是子弹的数组。


 复制代码

```
1 // 添加帧事件，开始游戏循环
2 backLayer.addEventListener(LEvent.ENTER_FRAME,onframe);
3 // 添加控制事件
4 backLayer.addEventListener(LMouseEvent.MOUSE_DOWN,ondown);
5 backLayer.addEventListener(LMouseEvent.MOUSE_UP,onup);
6 }
```

在初始化的最后，我们需要添加鼠标事件，将鼠标事件 LMouseEvent 的鼠标上下操作事件都放入到事件侦听函数内：addEventListener。

看到这里，你不禁会问，按照我们所教学的，游戏应该会有一个大循环来完成游戏的显示、操作等内容。那这个循环在哪里呢？

事实上这个循环，就在上面这串代码中。

 复制代码

```
1 backLayer.addEventListener(LEvent.ENTER_FRAME,onframe);
```

其中，LEvent 为事件类。**和传统客户端游戏不同，在 HTML5 游戏引擎中，循环采用了事件代码来完成，只要你在侦听器函数中注册了事件，都会一帧一帧不停地调度这个事件，以此达到循环的效果。**

在这里，注册的侦听事件函数就是 onframe 函数。查看它的源代码你可以看到 onframe 函数的实现细节，我就不在这里进行阐述了。

小结

今天，我们使用 lufylegend 引擎剖析了 HTML5 游戏的编写方式。我来总结一下。

HTML5 游戏的启动页是一个 HTML 文件。这个文件包含引擎的 js 文件。在其他别的引擎中，也是使用类似的方式来加载引擎。包含了引擎，就可以针对引擎所封装的接口进行游戏的开发。

HTML5 游戏的循环方式和传统游戏的循环方式不同。由于 HTML5 游戏引擎绝大部分是使用 JavaScript 脚本编写的，而 JS 本身就是以事件驱动的方式来工作的，所以**使用事件驱动是 HTML5 游戏引擎的一个很大特点**。我们在事件侦听函数中注册一个事件函数，在这个函数内编写“游戏循环”内的代码，就能起到传统游戏循环的作用。

在 HTML5 游戏中，我们几乎不需要关心图片的刷新或者清空操作。这些操作比之传统客户端游戏更为简洁，我们只需要关心游戏的图片呈现以及操作即可。

简单来说，****HTML5 游戏就是一种使用事件驱动模式并渲染在网页上的一种技术，省却了传统游戏的底层操控。****你在制作的时候，可以把更多的关注点放在游戏逻辑和可玩性上。

最后，留一个小问题给你。

在类似这样的 HTML5 打飞机游戏中，碰撞检测是怎么做的呢？请结合我以前的讲过的内容给出一个答案。

欢迎留言说出你的看法。我在下一节的挑战中等你！

从0开始学游戏开发

你的游戏开发入门第一课

蔡能

原网易游戏引擎架构师
资深游戏底层技术专家



新版升级：点击「👤 请朋友读」，10位好友免费读，邀请订阅更有**现金**奖励。

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 第15讲 | 如何设置淡入淡出和碰撞检测？

下一篇 第17讲 | 如何制作游戏资源包和保存机制？

精选留言 (3)

写留言



slark

2018-07-30

1

如果想学H5引擎，去搜下phaser，里面Demo很丰富
展开



III

2018-06-30

1

作为一名web前端，对游戏开发有兴趣，对这一篇教程很熟悉，做过一些很简单的h5游戏，但感觉缺少了游戏方面知识，所以来跟着老师学习这个课程...

作者回复: 谢谢，h5目前不是专栏主讲内容，所以会有些侧重不同。



赵鹏

2019-01-29



感觉lufylegend非常偏向很小的游戏，跟flash的效果一样。

展开 ▾