

个人资料



房燕良

[+ 加关注](#) [发私信](#)

访问: 125185次

积分: 1743

等级: [BLOG > 4](#)

排名: 第12304名

原创: 27篇 转载: 0篇

译文: 1篇 评论: 160条

引擎技术交流QQ群



游戏引擎能吃吗

(264656505)

文章搜索



常用网址

[Unity3D官网](#)
[Unreal Engine官网](#)
[游戏蛮牛](#)

文章分类

[Unity3D \(19\)](#)
[Unreal Engine \(2\)](#)
[自研引擎 \(2\)](#)
[其它 \(5\)](#)[CSDN专家精选, 微信开发学习路线大有看头!](#) [【博乐】点评美文, 得C币](#) [iOS开发前沿与Swift探秘](#) [Swift初级教程大汇总](#)

原 Unity3D脚本中的Awake()和Start()的本质区别

分类: [Unity3D](#)

2015-02-17 11:10 1482人阅读 评论(0) 收藏 举报

[unity3d](#) [脚本](#)

昨天被问到一个问题MonoBehaviour的Awake()和Start()的区别是啥?

当然Awake()会在Start()之前被调用, 这个地球人都知道了。如果只是顺序问题, 那当然就没太大必要搞两个函数了。仔细研究了一下API文档:

- Awake(): Awake is called when the script instance is being loaded.
- Start(): Start is called on the frame when a script is enabled just before any of the Update methods is called the first time.

OK, 从文档中我们看到他俩的区别是: **Awake()**是在脚本对象实例化时被调用的, 而**Start()**是在对象的第一帧时被调用的, 而且是在**Update()**之前。

为了更明确这点, 我们做一个小实验, 写一个脚本, 用来动态创建另外一个脚本对象:

```
[csharp]
01. using UnityEngine;
02. using System.Collections;
03.
04. public class TryObject : MonoBehaviour
05. {
06.
07.     // Use this for initialization
08.     void Start()
09.     {
10.         #if true
11.             GameObject dynaGO = new GameObject("DynamicGO");
12.             dynaGO.AddComponent<DynamicObject>();
13.         #else
14.             Object prefab = Resources.Load("DynamicGO");
15.             Object instance = GameObject.Instantiate(prefab);
16.         #endif
17.
18.     }
19.
20. }
```

另外一个脚本就写几个空函数, 用来打断点:

```
[csharp]
01. using UnityEngine;
```

文章存档

2015年08月 (1)

2015年07月 (3)

2015年04月 (1)

2015年03月 (1)

2015年02月 (1)

展开

阅读排行

《炉石传说》架构设计赏

(13888)

你不再需要TinyXML，推

(11247)

使用Unity3D的50个技巧

(10071)

分享一个小工具：Excel转

(7735)

2009年混合语言编程总结

(7620)

《炉石传说》架构设计赏

(7036)

Unity3D-RPG项目实战 (

(4863)

Unity 4.6的使用匿名dele

(4715)

《炉石传说》架构设计赏

(4589)

Unity3D-RPG项目实战 (

(4539)

最新评论

次世代关卡制作流程：使用Unre

风吹夏天: 大神回来了

分享一个小工具：Excel表快速转

valiant303: 解压看到一个

excel2json.exe的可执行文件，

但是没看到你提到的例子表格，

我怎么转换自己...

《炉石传说》架构设计赏析(2)：！

ZeroDeep: QQ群多少啊

分享一个小工具：Excel表快速转

胖小了个花: 感谢分享了哈，随便

试了下可以的。不过我要组织好

excel的格式。我想问下楼主，如

果我的excel里...

分享一个小工具：Excel表快速转

房燕良: @liyingwill123:已经上

传：

http://download.csdn.net/detail...

分享一个小工具：Excel表快速转

liyingwill123: 分享个编译的版本

吧，不可能为了用这个我还要装

个visual Studio啊

次世代关卡制作流程：使用Unre

王静娜: 学习了，给个赞

次世代关卡制作流程：使用Unre

jump fire: 好厉害的感觉。

(๑*۞*๑)

次世代关卡制作流程：使用Unre

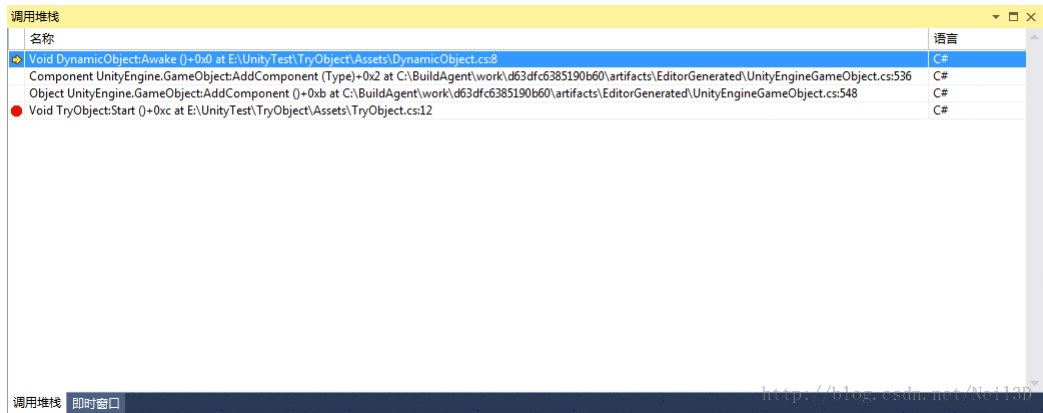
Json-Niu: 好高大上，学习了。

次世代关卡制作流程：使用Unre

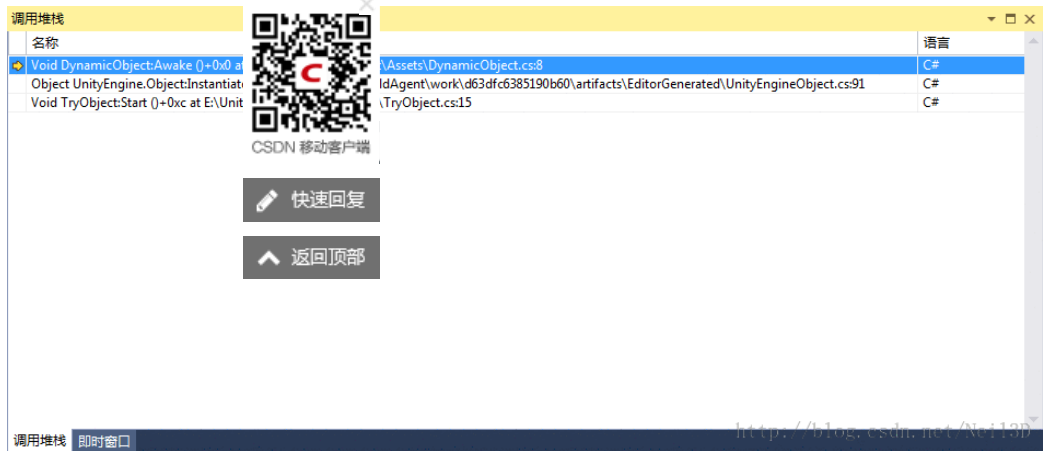
717606641: 学习了，感谢博主

```
02. using System.Collections;
03.
04. public class DynamicObject : MonoBehaviour
05. {
06.     void Awake()
07.     {
08.     }
09.
10.     // Use this for initialization
11.     void Start()
12.     {
13.
14.     }
15.
16.     // Update is called once per frame
17.     void Update()
18.     {
19.
20.     }
21. }
```

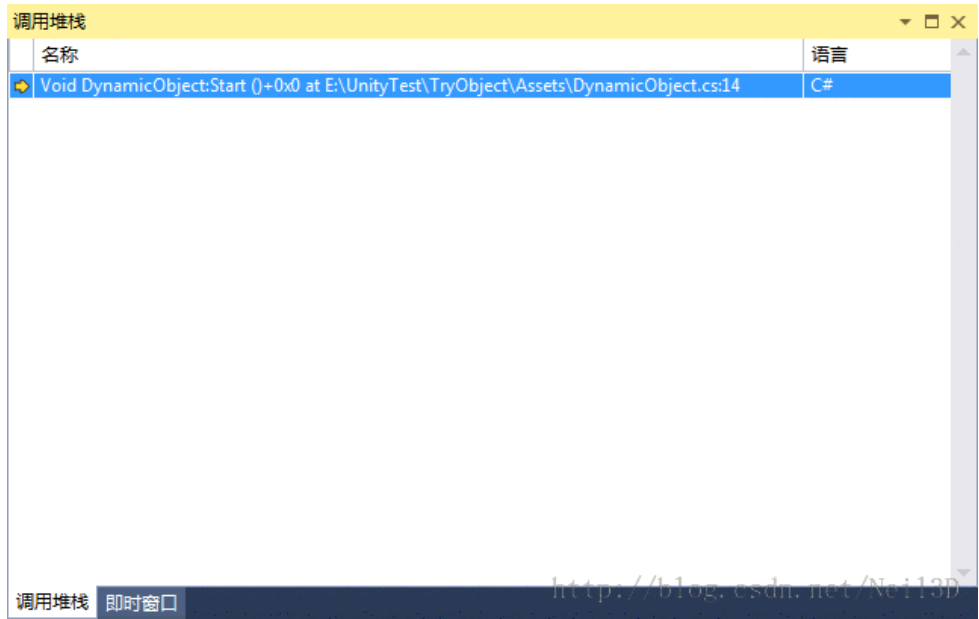
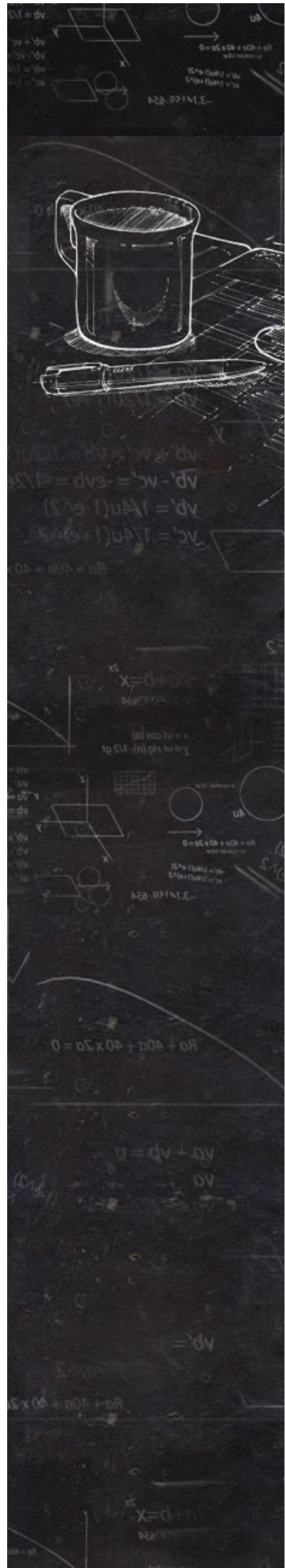
以下是使用AddComponent()方法时，DynamicObject:Awake()的调用堆栈：



下面是使用加载prefab的方式时，DvnamicObject:Awake()的调用堆栈：



以下是DynamicObject:Start()的调用堆栈：



这样的话，前面的结论就更明确了。在使用上，有几点值得注意：

- 脚本的一些成员， 如果想在创建之后的代码中立即使用，则必须写在**Awake()**里面；
- 当关卡加载时，脚本的**Awake**的次序是不能控制的；至于在关卡加载时，对象实例化和**Awake()**的调用关系，得看源码才知道了。

版权声明：本文为博主原创文章，未经博主允许不得转载。



- ▲ 上一篇
- 《炉石传说》架构设计赏析(7)：使用Google.ProtocolBuffers处理网络消息
- ▼ 下一篇
- 分享一个小工具：UnityRemoteLog

顶

2

踩

0

主题推荐

- unity3d
- 函数
- api
- 脚本
- initialization
- 对象
- 文档

猜你在找

- 基于Unity的游戏开发（下）
 - 深入浅出Unity3D——第一篇
 - bash shell脚本高级特性详解2
 - 实战进阶学习Unity3d游戏开发
 - 微信公众平台开发入门
- Unity3D脚本中Start和Awake的区别
 - unity3d Awake和 Start 区别
 - Unity3D中Awake和Start方法的区别
 - unity3d中脚本Awake和GetInstance的调用关系
 - unity3d 脚本参考-技术文档

准备好了么？跳 吧 ！

更多职位尽在 CSDN JOB

- unity3d高级开发工程师

我要跳槽

▪ 高级Unity3D开发工程师

我要跳槽
- 北京火谷网络科技有限公司

10-20K/月

珠海乐趣科技有限公司

13-26K/月



■ 高级UE4 / Unity3D游戏开发工程师

我要跳槽

■ Unity3D主程

我要跳槽

上海施嘉商务咨询有限公司

15-25K/月

上海云蟾网络科技有限公司

20-40K/月

Mobilize Your .NET

[xamarin.com](#)

Reuse your .NET and C# skills to code native iOS and Android apps.

查看评论

暂无评论

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

| 核心技术类目 | | | | | | | | | | | |
|------------|--------|-----------|---------|---------------|----------------|---------|------------|-----------|-----------|-----|--------|
| 全部主题 | Hadoop | AWS | 移动游戏 | Java | Android | iOS | Swift | 智能硬件 | Docker | | |
| OpenStack | VPN | Spark | ERP | IE10 | Eclipse | CRM | JavaScript | 数据库 | Ubuntu | NFC | |
| WAP | jQuery | BI | HTML5 | Spring | Apache | .NET | API | HTML | SDK | IIS | Fedora |
| LBS | Unity | Splashtop | UML | components | Windows Mobile | Rails | QEMU | KDE | Cassandra | | |
| CloudStack | FTC | coremail | OPhone | CouchBase | 云计算 | iOS6 | Rackspace | Web App | | | |
| SpringSide | Maemo | Compuware | 大数据 | aptech | Perl | Tornado | Ruby | Hibernate | ThinkPHP | | |
| HBase | Pure | Solr | Angular | Cloud Foundry | Redis | Scala | Django | Bootstrap | | | |

公司简介 | 招贤纳士 | 广告服务 | 银行汇款帐号 | 联系方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

网站客服 杂志客服 微博客服 webmaster@csdn.net 400-600-2320 | 北京创新乐知信息技术有限公司 版权所有 | 江苏乐知网络技术有限公司 提供商务支持
京 ICP 证 070598 号 | Copyright © 1999-2014, CSDN.NET, All Rights Reserved

