# Cocos Creator组件化代码开发解析

对于引擎而言，它使用的是一个一个的“功能组件”，那个组件能完成什么样的功能，就把它这个组件实例化，实例化后，加到节点上（角色），帮助角色完成对应的功能；

（1）节点：组件实例的载体（人）

（2）组件类：衣服

（3）组件实例：一件衣服

（4）组件实例化：添加组件到节点上

pt:每一个节点：节点+节点属性 组件实例 + 实例的属性 添加到组件的按钮；

系统为我们分类好了各种可用的组件类：–>系统开发好的；

根据类型 -->new组件类的实例 -->节点上；

Step1：

新建一个组件类，–>编辑器 -->组件类 -->识别到 -->用户脚本分类；

自动生成得一个 继承自cc.Component的类，自动找到它；

Step2：

new组件类的实例 -->节点上；

（1）直观方式：“添加到组件” 按钮 -->“添加到组件实例到节点”；

（2）代码：addComponent来添加我们的组件实例到节点上；

好处：我们的组件类开发完成以后，所有的节点都可以使用；

Step3:

节点上有了这些组件实例了，引擎运行的时候，会固定时期调用函数入口，有了这些入口，你就可以插入你的代码了；

固定时期 固定入口

开始运行之前： 引擎会调用组件实例.start； -->表.函数里面this指向表

游戏每次刷新： 引擎会调用组件实例.update(dt)；

dt：非常重要，当前update距离上次update过去的时间间隔

问题1：

引擎怎么找到组件实例：装载场景 -->节点 -->组件实例；

问题2：

什么叫每次刷新

游戏引擎，每隔一段时间刷新一下画面，时间间隔非常短

|------|------|—dt—|

dt：动画的所有的基准，100速度移动，从上一次update -->更新多少，100\*dt

console.log：打印，可以输出任何变量的值和数据

浏览器看打印，F2，chrome/firefox;

this.当前组件实例

this.node:当前组件实例所在节点；

Step4：

组件实例，可以定义属性绑定到编辑器–>可视化的修改组件实例的属性；

简单的数据绑定：名字:值

复杂的数据类型：名字：{type：cc.Node,default:null}

cc:所有cocos的对象类的开头

如果\_开头的属性，不会绑定到编辑器，但是，还是可以做为属性来访问到。