## Backbone.js event bind 新功能 listenTo and stopListening

简介[Backbone.js Events](http://translate.googleusercontent.com/translate_c?depth=1&hl=zh-CN&prev=/search%3Fq%3Dbackbone%2B%2Blistento%26newwindow%3D1%26safe%3Dstrict&rurl=translate.google.com.hk&sl=zh-TW&u=http://backbonejs.org/&usg=ALkJrhhe-bEGE8ngKJqO3mJRP3toasoz7g" \l "Events" \t "_blank)如何使用[on](http://translate.googleusercontent.com/translate_c?depth=1&hl=zh-CN&prev=/search%3Fq%3Dbackbone%2B%2Blistento%26newwindow%3D1%26safe%3Dstrict&rurl=translate.google.com.hk&sl=zh-TW&u=http://backbonejs.org/&usg=ALkJrhhe-bEGE8ngKJqO3mJRP3toasoz7g" \l "Events-on" \t "_blank)跟[off](http://translate.googleusercontent.com/translate_c?depth=1&hl=zh-CN&prev=/search%3Fq%3Dbackbone%2B%2Blistento%26newwindow%3D1%26safe%3Dstrict&rurl=translate.google.com.hk&sl=zh-TW&u=http://backbonejs.org/&usg=ALkJrhhe-bEGE8ngKJqO3mJRP3toasoz7g" \l "Events-off" \t "_blank)来处理事件，在升级Backbone到0.9.9过程中，其中一项重大新功能就是Backbone [listenTo](http://translate.googleusercontent.com/translate_c?depth=1&hl=zh-CN&prev=/search%3Fq%3Dbackbone%2B%2Blistento%26newwindow%3D1%26safe%3Dstrict&rurl=translate.google.com.hk&sl=zh-TW&u=http://backbonejs.org/&usg=ALkJrhhe-bEGE8ngKJqO3mJRP3toasoz7g" \l "Events-listenTo" \t "_blank)and [stopListening](http://translate.googleusercontent.com/translate_c?depth=1&hl=zh-CN&prev=/search%3Fq%3Dbackbone%2B%2Blistento%26newwindow%3D1%26safe%3Dstrict&rurl=translate.google.com.hk&sl=zh-TW&u=http://backbonejs.org/&usg=ALkJrhhe-bEGE8ngKJqO3mJRP3toasoz7g" \l "Events-stopListening" \t "_blank) ，大家来看看Change logs:

Added listenTo and stopListening to Events. They can be used as inversion-of-control flavors of on and off, for convenient unbinding of all events an object is currently listening to. view.remove() automatically calls view.stopListening().

如果大家想更了解listenTo and stopListening可以直接看国外这篇[Managing Events As Relationships, Not Just References](http://translate.googleusercontent.com/translate_c?depth=1&hl=zh-CN&prev=/search%3Fq%3Dbackbone%2B%2Blistento%26newwindow%3D1%26safe%3Dstrict&rurl=translate.google.com.hk&sl=zh-TW&u=http://lostechies.com/derickbailey/2013/02/06/managing-events-as-relationships-not-just-references/&usg=ALkJrhgsNcytvpX_4ntPMadHL3B0xu-QBQ" \t "_blank) ，或者是看看[此commit log](https://github.com/documentcloud/backbone/commit/1191640d84b24fde145eebd10dbd49e2335db506" \t "_blank) ，里面详细介绍为什么Backbone会增加此功能，其实最主要的重点是以前在Backbone View 写法都是像底下这样

initialize : function ( ) {   
if ( this . model ) {   
 this . model . on ( "change" , this . render , this ) ;   
 }   
} ,   
render : function ( ) {   
}

上面我们可以看到当model 有任何改变时，就会trigger render function，这是最简单的写法，当我们移除View 的时候，此事件就会跟着消失，但是如果是Reference 到其他Object 物件方法，如底下

myObject. on ( "some:event" , anotherObject. someHandler ) ;

当anotherObject物件被移除时，这时myObject还是绑定事件在anotherObject上面，这就会出现[zombie objects in JavaScript apps](http://translate.googleusercontent.com/translate_c?depth=1&hl=zh-CN&prev=/search%3Fq%3Dbackbone%2B%2Blistento%26newwindow%3D1%26safe%3Dstrict&rurl=translate.google.com.hk&sl=zh-TW&u=http://blog.wu-boy.com/2013/02/backbone-js-event-bind-listento-and-stoplistening/lostechies.com/derickbailey/2011/09/15/zombies-run-managing-page-transitions-in-backbone-apps/&usg=ALkJrhgqhV4McPXqhch2DOvfg3g_6u_4mQ" \t "_blank) ，当然这是有解法的，我们只需要在移除anotherObject之前呼叫底下程式码:

myObject. off ( "some:event" , anotherObject. someHandler ) ;

透过上面解法，我们可以将Backbone.js View 写法改成底下:

Backbone. View . extend ( {   
initialize : function ( ) {   
 this . model . on ( "change:foo" , this . doStuff , this ) ;   
} ,   
doStuff : function ( foo ) {   
*// do stuff in response to "foo" changing*   
} ,   
remove : function ( ) {   
this . model . off ( "change:foo" , this . doStuff , this ) ;   
*// call the base type's method, since we are overriding it*   
Backbone. View . prototype . remove . call ( this ) ;   
}   
} )

将[Backbone.js View Remove](http://translate.googleusercontent.com/translate_c?depth=1&hl=zh-CN&prev=/search%3Fq%3Dbackbone%2B%2Blistento%26newwindow%3D1%26safe%3Dstrict&rurl=translate.google.com.hk&sl=zh-TW&u=http://backbonejs.org/&usg=ALkJrhhe-bEGE8ngKJqO3mJRP3toasoz7g" \l "View-remove" \t "_blank)改写如下

remove : function ( ) {   
this . model . off ( "change:foo" , this . doStuff , this ) ;   
*// call the base type's method, since we are overriding it*   
Backbone. View . prototype . remove . call ( this ) ;   
}

这样就可以解决僵尸事件，但是如果有十个事件，那在remove 里面就会写的很杂，所以在0.9.9 版本出现listenTo 和stopListening 来解决此问题。 程式码改成如下

Backbone. View . extend ( {

initialize : function ( ) {   
this . listenTo ( this . model , "change:foo" , this . doStuff ) ;   
} ,   
  
doStuff : function ( foo ) {   
*// do stuff in response to "foo" changing*   
}   
  
*// we don't need this. the default `remove` method calls `stopListening` for us*   
*// remove: function(){*   
*// this.stopListening();*   
*// ...*   
*//}*   
} )

注意四大要点

1. 在Backbone.View 里面改用listenTo 写法   
2. 将this.model 传入listenTo function   
3. 不用传入最后一个参数this 当作context 变数   
4.不需要再override Backbone.View的remove function了，因为在程式码里面就会自动呼叫this.stopListening()

所以以后不要在写this.model.on了，请全面改写成this.listenTo(this.model, ..)写法。