### **创建小部件扩展**

通过部件库（Widget Factory）创建小部件是通过向 $.widget() 传递小部件名称和一个原型对象来完成的。下面的实例是在 "custom" 命名空间中创建一个 "superDialog" 小部件。

$.widget( "custom.superDialog", {} );

为了支持扩展，$.widget() 可选性地接受作为父部件使用的小部件的构造函数。当指定一个父部件时，把它作为第二个参数进行传递，放在小部件名称后面，在小部件原型对象前面。

$.widget( "custom.superDialog", $.ui.dialog, {} );

小部件的原型对象是传递给 $.widget() 的最后一个参数。到目前为止，我们的实例使用的是一个空的对象。现在让我们给这个对象添加一个方法：

$.widget( "custom.superDialog", $.ui.dialog, {

red: function() {

this.element.css( "color", "red" );

}});

// Create a new <div>, convert it into a superDialog, and call the red() method.

$( "<div>I am red</div>" )

.superDialog()

.superDialog( "red" );

### **扩展已有的方法**

有时候，您需要调整或添加已有部件方法的行为。您可以把方法名称指定为原型对象上需要重载的方法名称。下面的实例重载了 dialog（对话框）的 [open() 方法](http://www.runoob.com/jqueryui/api-dialog.html" \l "method-open)。由于对话框默认是打开的，当运行这段代码时，"open" 将会被记录。

$.widget( "custom.superDialog", $.ui.dialog, {

open: function() {

console.log( "open" );

}});

// Create a new <div>, and convert it into a superDialog.

$( "<div>" ).superDialog();

### **使用 \_super() 和 \_superApply() 来访问父部件**

[\_super()](http://www.runoob.com/jqueryui/api-jQuery-widget.html" \l "method-_super) 和 [\_superApply()](http://www.runoob.com/jqueryui/api-jQuery-widget.html" \l "method-_superApply) 在父部件中调用了同样的方法。请看下面的实例。就像上一个实例，这个实例也重载了 open() 方法来记录 "open"。然而，这次运行 \_super() 是调用了 dialog（对话框）的 open()，并打开对话框。

$.widget( "custom.superDialog", $.ui.dialog, {

open: function() {

console.log( "open" );

// Invoke the parent widget's open().

return this.\_super();

}});

$( "<div>" ).superDialog();

### **重定义小部件**

jQuery UI 1.9 添加了重定义小部件的功能。因此，可以不用创建一个新的小部件，我们只需要传递 $.widget() 这样一个已有的小部件名称和构造函数即可。下面的实例在 open() 中添加了相同的记录，但不是通过创建一个新的小部件来完成的。

$.widget( "ui.dialog", $.ui.dialog, {

open: function() {

console.log( "open" );

return this.\_super();

}});

$( "<div>" ).dialog();

### **小部件（Widgets）和多态性（Polymorphism）**

当在小部件扩展及它们的插件之间进行交互时候，有一点值得注意，父部件的插件不能用来调用子部件元素上的方法。下面的实例演示了这一点。

$.widget( "custom.superDialog", $.ui.dialog, {} );

var dialog = $( "<div>" ).superDialog();

// This works.

dialog.superDialog( "close" );

// This doesn't.

dialog.dialog( "close" );