**关键字：**

**$emit : 向上传**

**$broadcast : 向下传**

**$on : 接受**

**语法：**

**$scope.$emit(key , value);**

**$scope.$broadcast(key , value);**

**$scope.$on(key,function(event , value){**

**//key、value对应$emit/$broadcast的key、value**

**});**

AngularJS中的作用域有一个非常有层次和嵌套分明的结构。其中它们都有一个主要的$rootScope(也就说对应的Angular应用或者ng-app)，然后其他所有的作用域部分都是继承自这个$rootScope的，或者说都是嵌套在主作用域下面的。很多时候，你会发现这些作用域不会共享变量或者说都不会从另一个原型继承什么。

那么在这种情况下，如何在作用域之间通信呢？其中一个选择就是在应用程序作用域之中创建一个单例服务，然后通过这个服务处理所有子作用域的通信。

在AngularJS中还有另外一个选择：通过作用域中的事件处理通信。但是这种方法有一些限制；例如，你并不能广泛的将事件传播到所有监控的作用域中。你必须选择是否与父级作用域或者子作用域通信。

$on、$emit和$broadcast使得event、data在controller之间的传递变的简单。

* $emit只能向parent controller传递event与data
* $broadcast只能向child controller传递event与data
* $on用于接收event与data

| **事件属性** | **目的** |
| --- | --- |
| **event.targetScope** | **发出或者传播原始事件的作用域** |
| event.currentScope | 目前正在处理的事件的作用域 |
| event.name | 事件名称 |
| event.stopPropagation() | 一个防止事件进一步传播(冒泡/捕获)的函数(这只适用于使用`$emit`发出的事件) |
| event.preventDefault() | 这个方法实际上不会做什么事，但是会设置`defaultPrevented`为true。直到事件监听器的实现者采取行动之前它才会检查`defaultPrevented`的值。 |
| event.defaultPrevented | 如果调用了`preventDefault`则为true |

完整例子：

<!DOCTYPE **html**>  
<**html lang="en"**>  
<**head**>  
 <**meta charset="UTF-8"**>  
 <**title**>$emit $broadcast</**title**>  
 <**script src="angular.js"**></**script**>  
</**head**>  
<**body**>  
 <**div ng-controller="parent"**>  
 father: {{***username***}}  
 <**div ng-controller="me"**>  
 me: {{***username***}}  
 <**button type="button" ng-click="emit()"**>向parent传</**button**>  
 <**button type="button" ng-click="broadcast()"**>向child传</**button**>  
 <**div ng-controller="child"**>  
 child: {{***username***}}  
 </**div**>  
 </**div**>  
 </**div**>  
<**script**>  
 **var *app***=angular.*module*(**"myApp"**,[]);  
  
 ***app***.**controller**(**"parent"**,[**"$scope"**,**function**($scope){  
 *//reserve from parent* $scope.$on(**"toParent"**,**function**(event,data){  
 $scope.**username**=data;  
 });  
 }]);  
  
 ***app***.**controller**(**"me"**,[**"$scope"**,**function**($scope){  
 $scope.**username**=**"tongwl"**;  
  
 *//to parent* $scope.emit=**function**(){  
 $scope.$emit(**"toParent"**,$scope.**username**);  
 }  
  
 *//to child* $scope.broadcast=**function**(){  
 $scope.$broadcast(**"toChild"**,**"changedName"**);  
 }  
  
 }]);  
  
 ***app***.**controller**(**"child"**,[**"$scope"**,**function**($scope){  
 $scope.$on(**"toChild"**,**function**(event,data){  
 $scope.**username**=data;  
 });  
 }]);  
  
  
 angular.**element**(***document***).ready(**function**(){  
 angular.*bootstrap*(***document***,[**"myApp"**]);  
 });  
  
</**script**>  
</**body**>  
</**html**>

Tongwl