

Actions Tech. Note 13

Draft 1.0

Triggering Slave Groups

Freeway 2.0 introduced the idea of slave groups to facilitate the interaction of actions on pages and across frame-sets. The concept of a slave group is quite simple. The slave group represents a set of entities that are able to receive messages - and different messages (or triggers) are sent to members of this group.

The standard Freeway actions and those actions that are written by Softpress have ten slave groups:

- Aqua
- Cadmium
- Chartreuse
- Cobalt
- Indigo
- Rose
- Slate
- Turquoise
- Ultramarine
- Zinc

These groups are arbitrary names that were chosen as being easily identifiable to the user and because the names were relatively pleasant sounding. It is quite possible to write actions that have other or different slave groups if you need to.

An action can message the slave group, with a message pair that consists of a trigger number and a message.

Standard Actions currently identify target numbers from 1 to 20. There are just two message numbers 1 and 0. The target numbers typically correspond to something the images in a "Slave Image" action. The messages typically loosely have the following meanings:

1 Trigger Enter e.g. MouseOver, Click etc.

0 Trigger Leave e.g. Mouse Out etc.

To send a message to members of the slave group there are two JavaScript functions that appear in all the Softpress authored actions. You can write your own if you need or want to but you are probably best off treating these as a single functional unit and copy and pasting them. The code is as follows:

```
function FWCallHit(func,targNum,msg)
{
    if(func)
        for (var i in func)
            func[i](targNum,msg);
}

function FW_Hit(frameset,chain,targNum,msg)
{

```

```

if (frameset && frameset.length)
  for (var i=0 ; i <frameset.length ; i++)
  {
    FW_Hit(frameset[i].frames,chain,targNum,msg);
    FWCallHit(top["FT_"+chain],targNum,msg);
    FWCallHit(frameset[i].window["FT_"+chain],targNum,msg);
  }
else
  FWCallHit(window["FT_"+chain],targNum,msg);
}

```

To use the code you should call `FW_Hit`. This function has four arguments:

frameset - the top-most frameset that you want to trigger slave actions in. Typical arguments are: `top.frames`, `parent.frames`, `this.frames`
chain - the chain that you want to trigger
targetNum - the target number (typically 1..20)
msg - the message (typically 0 or 1)

Example 1

The code

```
FW_Hit(top.frames, 'Indigo',10,1)
```

Will send a message 1 to target 10 to all actions in slave group `Indigo` to all pages within the current window (going through all framesets contained in that window).

Example 2

The code

```
FW_Hit(this.frames, 'Chartreuse',5,1)
```

Will send a message 1 to target 5 to all actions in slave group `Chartreuse` to all actions on the current page

Example 3

The following action will create a link and let you set up an arbitrary call to the function `FW_Hit` when you do a mouseover. It is intended as an example and as a template for you to cut and paste when making your own actions.

```

<action name="Trigger Test">

<action-popup name="frameset">
  <value name="top.frames">
  <value name="parent.frames">
  <value name="this.frames">
</action-popup>

<action-popup name="chain">
  <value name="Aqua" />
  <value name="Cadmium" />

```

```

<value name="Chartreuse" />
<value name="Cobalt" />
<value name="Indigo" default/>
<value name="Rose" />
<value name="Slate" />
<value name="Turquoise" />
<value name="Ultramarine" />
<value name="Zinc" />
</action-popup>

<action-popup name="targetNum">
  <value name="1" /> <value name="2" /> <value name="3" /> <value name="4" /> <value
name="5" />
  <value name="6" /> <value name="7" /> <value name="8" /> <value name="9" /> <value
name="10" /> <value name="11" />
  <value name="12" /> <value name="13" /> <value name="14" /> <value name="15" />
<value name="16" /> <value name="17" />
  <value name="18" /> <value name="19" /> <value name="20" />
</action-popup>

<action-popup name="msg">
  <value name="0" />
  <value name="1" default />
</action-popup>

<action-markup custom name="FW_Hit">
function FWCallHit(func,targNum,msg)
{
  if(func)
    for (var i in func)
      func[i](targNum,msg);
}
function FW_Hit(frameset,chain,targNum,msg)
{
  if (frameset && frameset.length)
    for (var i=0 ; i <frameset.length ; i++)
    {
      FW_Hit(frameset[i].frames,chain,targNum,msg);
      FWCallHit(top["FT_"+chain],targNum,msg);
      FWCallHit(frameset[i].window["FT_"+chain],targNum,msg);
    }
  else
    FWCallHit(window["FT_"+chain],targNum,msg);
}
</action-markup>

<action-javascript>

// This appends a piece of JavaScript stored in /action-markup/ to a specific tag
function AppendJavaScript(tag, markup)
{
  // Append a piece of markup if it is not already defined
  if (tag && !fwPage[markup])
  {
    var javascript = tag.fwAddJavaScript();
    javascript.fwAddRawOpt(fwMarkups[markup]);
    fwPage[markup] = true;
  }
}

function fwBeforeEndBody()
{
  // find the head tag

```

```

var headTag = fwDocument.fwTags.fwFind("head");

// find the container
var containerTag = fwDocument.fwTags.fwFind("",fwItem);

if (headTag && containerTag)
{
    // add the trigger method "FW_Hit"
    AppendJavaScript(headTag, "FW_Hit");

    // make the "mouseover" string.
    var mouseOverStr = "FW_Hit("
        +fwParameters["frameset"] +", "           // frameset
        +" '"+fwParameters["chain"]+"', "         // chain
        +fwParameters["targetNum"]+"",
        +fwParameters["msg"]+" )";

    // add a /a/
    a = containerTag.fwAdd("a",true);

    // set the href of "#"
    a.href=' "#" ';

    // set the mouseover
    a.fwAddJToTag("onmouseover", mouseOverStr);

    // add the text
    a.fwAddRaw(mouseOverStr);
}
}

</action-javascript>
</action>

```