QlikView

QlikView 11 Extension Examples:

A Web Developer's Guide to 5 Extension Concepts and Code

Brian Munz - Last Updated 10/20/2011

QlikView

Contents

QlikView

Introduction

Extensions are a powerful part of QlikView that was introduced in version 10. They allow us to extend the capabilities of QlikView by opening its data and associative analysis to web technologies. The innovative uses of extensions have greatly increased since they were introduced, so some new examples were created in order to show the range and capabilities of extensions. Overall, the purpose of these 5 extensions was to show a varying degree of difficulty, usefulness, and creativity in the hopes of inspiring new and innovative uses for extensions. The 5 extensions described in this document can be viewed in the Extensions Examples document which installs with QlikView 11.

Assumptions

It is assumed that the reader of this document is a web developer who is knowledgeable in JavaScript, HTML, and general web technologies. This is a more technical document than simply showcasing the end product. Rather, this document is meant as a look "under the hood" of the extensions.

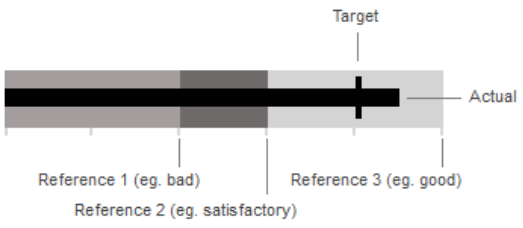
The code of the Script.js files are in the Appendix of this document. Also, the extensions are installed with QlikView 11 (typically in the C:\Program Files\QlikView\Examples\Extensions directory), so they can be viewed and used there. The code is commented heavily, so it is assumed that the reader of this document will look through this code and use the comments as a general guide. This paper will feature and discuss technical concepts that were utilized.

Also, a general knowledge of extensions is assumed as far as adding an extension to a document, where to find them in the file structure, etc.

Bullet Chart: No Frills, Hand-Coded JavaScript, CSS, and HTML

Overview

The bullet chart is a chart that was first introduced by Stephen Few several years ago.



Prior to this extension it was possible to achieve this kind of chart, however setting up these charts in standard QlikView was very time-consuming and laborious and involved layer several bar charts on top of each other. Using an extension, creating a bullet chart could be simplified and also perhaps ease the page load of creating all of those bar charts when simple HTML divs would do the job.

This extension was coded using standard JavaScript, HTML, and CSS. No images or 3rd party technologies were necessary.

The overall idea behind the coding for the bullet chart was to simply use divs of varying heights and widths to represent the 5 bars needed to create the chart. These various divs can be overlaid on top of each other using absolute positioning and z-index values in CSS. In order to determine their different sizes, the code simply uses the Max Value bar value and sizes the rest of the bars in relation to that one.

Conflicting IDs

We are creating a series of divs in the extension and appending other divs to them to create this extension. So, initially a container div is created to hold all of the bars inside it. This works fine until you decide to add more than one bullet chart to a page. If you had two bullet charts, they'd both contain divs which have the same ID, so when you try to append new elements to the div, it wouldn't know which ID to target. This is a problem for many extensions, so it is important to keep in mind while building extensions.

In order to get around this, a random number is created and added that to the container div ID:

```
var r=Math.floor(Math.random()*10001);
var divName = "bulletContainer" + r;
this.Element.innerHTML = "<div id='" + divName + "'></div>";
```

This way, each bullet chart in a document will have a unique ID and they will not compete with each other.

Color Formatter

Something else to point out is the use of the colorFormatter function:

```
colorFormatter = function(c, type){
      var cc = c.toLowerCase();
      if((cc.length === 0) && (colorErr != 1)){
            colorErr.push(type);
      }else if((cc.indexOf("rgb") === -1) && (cc.indexOf("#") === -1)){
            if(cc.length < 6){
                  var addIt = "#";
                  for(var i=cc.length; i < 6;i++){
                        addIt += "0";
                  }
                  cc = addIt + cc
```

```
            }else{
                  cc = "#" + cc;
            }
            return cc;
      }else{
            return cc;
      }
}
```

This function is used in several extensions to allow for the entry of colors in both Hex format as well as RGB format, since RGB is more familiar to QlikView developers. This simply further aids in making the extension as user friendly as possible.

Horizontal and Vertical Support

Bullet charts are used in both a horizontal and vertical format. To accommodate this, the code was modified in several places to change the placement of the divs from the left to the bottom, for example. While one way to add this flexibility would be to have a check box in the properties, an easier solution was implemented in the code. All of the information that is needed is already present in the actual sizing of the extension itself, so in the code the type of chart is determined this way:

```
if(thisHeight > thisWidth){
          orient = "v";
          ticScale = thisHeight;
      }else{
          orient = "h";
          ticScale = thisWidth;
      }
```

Quite simple, if the width of the extension is greater than the height, it should be displayed as a horizontal bullet chart. Otherwise, the chart should be displayed vertically. This makes the extension more flexible and provides one less setting for the user to make.

Error Handling

With several of these extensions, error handling has been added to provide some guidance if the extension is not working. In this extension, an array is created called `colorErr`. If there are any items in this array at the end of the function, an error message is displayed accordingly.

Calendar Chart: Help QlikView Help Your Extension

Overview

Date based data is very commonly used in QlikView, however, there was no real way to create a useful calendar display of events. The aim of this extension was to provide an easy visual guide so a person

could view a month's schedule at a glance. Also, the ability to view a week's events vs. a month's events was provided.

Date Data Formatting

While displaying a calendar of events may sound simple, it becomes clear pretty fast that it is more complicated. The largest problem is that most campaigns and longer events are entered into data as a start date and an end date. This would be okay for creating a calendar if the start and end date were within the same month. If not, however, due to the associative nature of QlikView, if a person chose a month, a campaign with a start date earlier than that month would not come through to the extension in the result set. QlikView is only interested in data associated with that month, and since the range of that campaign starts prior, that data will no longer be associated.

There were several ways that were considered in solving this. One option was to manipulate the data coming into the extension so that there was a row for each day of the month and each event of that day. The problem here of course is that if you had 8 events (or more!) on the same day, that would be 8 rows of data. So, if you had a 30 day month with 8 events per day, that would be 240 rows of data. This could slow down the extension especially if there were several year-long campaigns that could cause many events to exist for each day.

In order to cut the number of rows greatly, this extension works by having an entry for each event per week. So, in the properties, there is an event week start date and end date. While greatly reducing the returned data, the code then simply iterates through the week start and end dates to display the data. For example, with the sDate and eDate being the start date and end date for that week, we use this code to iterate through that range and make an event entry in the calendar for each day:

```
while (sDate.valueOf() < eDate.valueOf() + 86400000) {
```

By making some modifications and adding some tables in QlikView itself, the extension could circumvent a limitation of using simple date ranges and ease the load.

3rd Party JavaScript Functions

Working with dates and calendars can be somewhat laborious, especially if the extension needs to be flexible for any date range past, present, or future. Thus, it's feasible to think that a ewb developer might decide to use an open source set of javascript functions to do the heavy lifting when it comes to calculating and displaying the calendar dates.

At the top of this extension, an older set of javascript functions have been added to aid in this way. Developers should proceed with caution when using 3rd party code, however, not everyone has time to build an entire calendar generator, so it's often necessary in achieving the extension. In this case, these functions create the shell into which the events are placed as the data is traversed.

Flyouts and Tooltips

Having tooltips and flyouts within an extension can be very useful for displaying more information about a given item. In this extension, a flyout is used to either display details about the specific events in week view or the full list of events in month view. Given the constraints of the height and width of a desktop application, not many events can be displayed on each day in month view. The solution to this was to display the full list of events in a flyout when the day is clicked. In the code, all of the events past row 5 would not be visible due to a CSS overflow of hidden on the calendar days. This way, the information is there, however, it is not displayed and does not disrupt the layout. Once this is set up, all that needs to be done when a day is clicked, is fill the flyout with that full amount of data by copying the contents of the day's <ul> into the flyout like so:

```
$('.calTable td').click(function(){
      if (($(this).html() != " ") && ($(this).find('ul').length > 0)) {
            $('#fo').hide();
            $('#fo #content').html('<ul>' + $(this).find('ul').html() +
'</ul>');
```

Similarly, for the week view, the information about each event is placed into a div with a display:none in order to hide it initially. When the event is clicked, the flyout is filled with that hidden data.

Week View vs. Month View

Similar to the bullet chart, rather than adding another property to set whether the extension should display in week view or month view, the code was simply modified to detect if a given week range was chosen exclusively. If so, this was detected and the "view" variable would be set to "m" rather than "w."

Geographical Heat Map: Heating Up Your Visualizations

Overview

A geographical heat map can be a very powerful visualization tool for comparative analysis. Marking certain regions of a map as "hot" or "cold" or simply one category vs. another (Republican vs. Democrat), can be a useful way to quickly extract valuable information. This extension uses the SVG image format and an 3rd party open source JavaScript library to achieve this chart.

Creating An SVG Image (Sort Of)

Creating and working with scalable images on the web can be very difficult especially if you want to allow user interaction. Fortunately, there are SVG (Scalable Vector Graphics). These images are essentially XML files that contain paths which draw out an image, making it very scalable and data friendly.

Unfortunately, many browsers don't support SVG yet and it is important to accommodate as wide an audience as possible when developing.

Thankfully, there is Raphael. Raphael, (raphaeljs.com) is an open source javascript library that is very good at using javascript to draw images using the same types of paths SVG does. In fact, if the browser supports SVG, the image will be rendered as SVG, but if not, Raphael will draw the image itself.

This amazing library is fairly easy to use and set up, but once glance at the code and you can see that setting up an SVG to be drawn can be time consuming. Once the SVG image is created with all of the states as separate paths (this is important), each states path much me manually enterered into Raphael and be given an ID and attributes.

For example, in setting up Hawaii to an existing Raphael object, the following code is needed:

```
us.HI = R.path("M224.08,523.1l1.883-
3.469l2.196...etc...").attr(attr).scale(scaleNum,scaleNum, 0, 0);
```

The path is the data extracted from the SVG file (copied and pasted). WHile this is laborious, the benefit is that us.HI is now an object in Raphael. It can have click events attached to it, its attributes can be modified (such as color), and so on. Thus, once all 50 states are entered, there are 50 objects that can be manipulated by us.

Hot and Cold

Given that there are now state objects in Raphael to manipulate, the state's percentage value needs to be displayed in terms of the hot and cold colors. For example, if the cold color (0%) is white and the hot color (100%) is red and Hawaii's percentage is 50%, Hawaii should display as a color halfway between red and white.

The way this is achieved is to have another SVG which lies beneath the individual state SVGs. In this case it is an outline of the entire United States. The cold color is applied to this entire region in order to represent a value of 0%. Then, all that needs to be done is simply set all of the states to the hot color, and adjust the opacity of the state to their associated percentage value. For example:

```
st.attr("fill-opacity", per);
st.animate({fill: hot}, 800);
```

The chosen state has its opacity attribute in Raphael set to the percentage variable and its fill color set to the hot color.

So in the case of Hawaii mentioned above, the US outline would be set to white, Hawaii's color would be red, and the opacity of that state would be set to 50%, achieving the desired effect.

QlikView

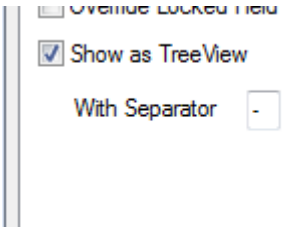
Org Chart: Creating a Hierarchy

Overview

An organizational chart, and hierarchical data in general can be very complex. All of the elements after the very first one have dependencies and the potential for parent elements and child elements. Given this fact, a hierarchy can also be viewed as nothing more than a series of embedded HTML lists. So, the challenge for this extension was two fold: get the hierarchical data into a <ul>, and visually represent the hierarchy.

Manipulating the Data

In QlikView, a hierarchical tree view can be easily achieved by loading the data in a delineated fashion and choosing "show as TreeView" in the listbox settings:



When this is done, the data will be displayed in a traditional treeview, giving the user that ability to drill down through the various levels.

The only problem from the viewpoint of extensions, is that this data comes in looking similar to this

Charles Johnson-Donald Williams-Daniel Fertig

This indicates that Daniel Fertig's boss is Donald William's boss whose boss is Charles Johnson. Traversing through this data might not be too difficult if it came into the extension in a very specific and organized way. Data comes into an extension, however, in a format similar to a straight table or a SQL query result set. So the entry for Charles Johnson-Donald Williams-Daniel Fertig could be the first entry we read. Thus, the hierarchical structure needs to be able to be built starting from any point in the hierarchy, the result of which is a JSON object containing all of the proper dependencies.

The code in the extensions and appendix should help explain the nuts and bolts of the function, but overall, the hyphen delineated hierarchy row is split into an array:

```
var chain = row[0].text.split("-");
```

This array contains all of the names in that given row. Each of those names is then checked against the existing hierarchy that has been generated so far by looping through the hierarchy over and over until the proper child is placed with the proper parent.

Another consideration with using this kind of hierarchy is in displaying data about the different people. In tree view, if you select a person who is 3 levels down, QlikView sees this as a selection of that person and anyone below them, thus due to the associative nature of QlikView, that is the only data available. Luckily, due to the hyphenated string we have, we're able to at least see who the person's bosses are even without getting information on them. So, as we loop through these hyphenated strings, many people in the tree begin as having no data, however once they are the current person whose data is represented in the row, that information is added to their entry in the JSON structure.

Traversing the JSON

Now that the JSON is created, traversing through it to create HTML lists is not very difficult thanks to recursion. In the code snippet below, you can see that as a new object is reached in JSON, the code checks whether or not that element has any children or not.

```
if (this.children.length > 0) {
      if (this.path == "dis") {
            returnThis += '<li><span class="dis">' + this.name + '</span>';
            returnThis += "<ul>";
            $.each(this.children, function(index, value){
                  returnThis += traverse(this);
            });
            returnThis += "</ul>";
      }else {
            returnThis += '<li><span class="parent">' + this.name + '<span
class="hiddenInfo"><h3>' + this.name + '</h3><p class="d1">' + this.d1 +
'</p><p class="d2">' + this.d2 + '</p></span><a class="moreInfo">more
information</a></span>';
...
```

So, essentially, if it's found that the element has children, recursively call the same function to run it on those children. This way, the function will crawl its way through all the data, calling itself when needed. Also, it is determined whether that element is disabled or not ("dis"), and if it is not, the person's data is printed within the li, to be used later by the flyout.

CSS and Image Trickery

One of the harder parts in developing this extension was avoiding any sort of actual canvas or javascript drawing to connect the elements. Without going into exhaustive detail, much of the connections between the different boxes (which are images) is done by applying background images. As the ULs are created with their subsequent list items (which also may contain their own ULs and LIs, the background for these ULs get wider and wider. If this list grows wider it means that there are several siblings and their connections can be represented (as they are in the extension) as a horizontal line running above the elements. It might be helpful now to take a look at the images inside the folder installed with the extension.

The simple way to achieve this line was to apply a background image to the ULs which is simply a horizontal rule that repeats across the width of the UL. The seemingly then puts a border on top of the different ULs. Then all that needs to be done to connect those boxes to that rule is to add a line to the top of the box which meets the rule above it. Of course, there are exceptions and issues taht need to be accounted for, and their solutions can be seen in the code, but overall the hardest part of connecting the boxes is achieved this way.

Pie Chart with Whiskers: Quick and Easy

Overview

Sometimes an entirely new and unique chart isn't needed. Rather, maybe there's a slight addition that is needed to a basic chart that QlikView cannot provide. In this vein, the Pie Chart with whiskers is a simple pie chart with the pie values displayed near the slices with the slight modification that there are lines connecting the text to the pie piece. This extension is an example of heavy use of 3rd party libraries to achieve a quick chart.

Extensions in Extensions

JQPlot is a javascript charting library which allows developers to easily create data visualizations by simply including their libraries and passing in the proper data. Unfortunately, JQPlot did not offer the desired whisker visualization that was needed, however, an extension was developed to JQPlot which extends the pie chart library to allow it. Thus, in the extension:

```
Qva.LoadScript("/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensions/QlikView/Examples/piechartlabel/jquery.jqplot.min.js", function() {  //JQPlot pie charting library

      Qva.LoadScript("/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensions/QlikView/Examples/piechartlabel/jqplot.pieRenderer.lineLabels.js", pie_label_done); //Custom pie chart extension which adds whiskers
```

The jqplot library is included then the extension library is included soon after. Once the libraries are loaded and ready, it is as simple as getting the pie chart data and feeding it into a JSON object that the chart needs:

```
for (var rowIx = 0; rowIx < _this.Data.Rows.length; rowIx++) {
      line1.push([_this.Data.Rows[rowIx][0].text,
parseFloat(_this.Data.Rows[rowIx][1].text)]);
}
```

The chart then displays, as easy as...pie.

Conclusion

Hopefully the 5 extensions described in this paper will aid developers in thinking creatively about extensions. An important conclusion to draw is that almost anything that can be done in JavaScript can be done in an extension...and more. The power of QlikView and its interaction with extensions can really make them a powerful and useful tool.

Appendix A - Bullet Chart

Script.js

```
var colorErr = [];

//Function to place tic marks on bullet chart if the user has chosen this
option in the properties
placeTics = function(vh, vhnum, tnum, c){
// vh - determines if chart is vertical or horizontal
// vhnum - total width or height of chart
// tnum - number of tics to use
// c - the actual chart div
    var ticLong = "5px";
    var ticShort = "2px";

      //calculate the amount of space between the tics
      var unit = vhnum/(tnum - 1);

    var cSpan = 0;
    for(var i=0;i<tnum;i++){
        var newTic = document.createElement("div");
        newTic.className = "tic";

        newTic.style.backgroundColor = "#d8d8d8";

            //If it's not the first tic...
            if(cSpan != 0){
            if(vh == "h"){
                        newTic.style.left  = (cSpan-2) + 'px';
            }else{
                        newTic.style.top  = (cSpan-2) + 'px';
            }
            //if it is the first tic, forget about adjusting the placement by
2 pixels to accomodate the size of the tic
        }else{
            if(vh == "h"){
                        newTic.style.left  = cSpan + 'px';
            }else{
                        newTic.style.top  = cSpan + 'px';
            }
        }
```

```
        if(vh == "h"){
                  newTic.style.width = ticShort;
                  newTic.style.height = ticLong;
                  newTic.style.top  =  (c.offsetHeight - 5) + 'px';
            }else{
                  newTic.style.width = ticLong;
                  newTic.style.height = ticShort;
                  newTic.style.left  =  (c.offsetWidth - 5) + 'px';
            }
        c.appendChild(newTic);
        cSpan = cSpan + unit;
    }
}

//function to accomodate hex color entry as well as RGB
colorFormatter = function(c, type){
      var cc = c.toLowerCase();
      if((cc.length === 0) && (colorErr != 1)){
            colorErr.push(type);
      }else if((cc.indexOf("rgb") === -1) && (cc.indexOf("#") === -1)){
            if(cc.length < 6){
                  var addIt = "#";
                  for(var i=cc.length; i < 6;i++){
                        addIt += "0";
                  }
                  cc = addIt + cc
            }else{
                  cc = "#" + cc;
            }
            return cc;
      }else{
            return cc;
      }
}
Qva.AddExtension('QlikView/Examples/bullet', function() {
    colorErr = [];

      //Load CSS

Qva.LoadCSS("/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensions/QlikVi
ew/Examples/bullet/bullet.css");

      //Create random number to avoid ID conflicts with multiple extension
instances
      var r=Math.floor(Math.random()*10001);
      var divName = "bulletContainer" + r;
    this.Element.innerHTML = "<div id='" + divName + "'></div>";

      var tot = this.Data.Rows[0][0].text; //Max range bar
      var ticFlag = this.Layout.Text4.text;     //detect if user has chosen
to show tic marks
      var maxColor = colorFormatter(this.Layout.Text0.text.toString(), "Max
Range Bar color");  //color of max range bar
```

```
	//Determine the size of the first bar in relation to the max range.
	//This technique will be used for the remaining value bars
	var m1 = this.Data.Rows[0][2].text/tot;
	var m1Color = "";
	if((m1 != 0) && !(isNaN(m1))){ //if there is a value for this bar, set
the color
		m1Color = colorFormatter(this.Layout.Text2.text.toString(),
"Medium Range Bar color");
	}
	var m2 = this.Data.Rows[0][1].text/tot;
	var m2Color = "";
	if((m2 != 0) && !(isNaN(m2))){
		m2Color = colorFormatter(this.Layout.Text1.text.toString(),
"Short Range Bar color");
	}
	var per = this.Data.Rows[0][3].text/tot;
	var perColor = colorFormatter(this.Layout.Text3.text.toString(), "Fill
Bar Color");
	var bench = this.Data.Rows[0][4].text/tot;
	var orient = "";
	var ticScale = "";
	//get height and width
	var thisHeight = this.GetHeight();
	var thisWidth = this.GetWidth();
	//use height and width to determine if the chart should be displayed
horizonally or vertically
	if(thisHeight > thisWidth){
	    orient = "v";
	    ticScale = thisHeight;
	}else{
	    orient = "h";
	    ticScale = thisWidth;
	}
	var bul = document.getElementById(divName);
	//set bullet div height and weight to the size of the extension
	bul.style.height = this.GetHeight() + "px";
	bul.style.width = this.GetWidth() + "px";
	// If tics are chosen to be shown, adjust the size to accomodate the
tics
	if((ticFlag == 1) && (orient == "h")){
		var scaleH = bul.offsetHeight - 5;
	}else{
		var scaleH = bul.offsetHeight;
	}
	if((ticFlag == 1) && (orient == "v")){
		var scaleW = bul.offsetWidth - 5;
	}else{
		var scaleW = bul.offsetWidth;
	}
	//begin creating all of the bars based on the calulated values above.
	var scaleBar = document.createElement("div");
	scaleBar.style.height = scaleH + 'px';
```

```
	scaleBar.style.width = scaleW + 'px';
	scaleBar.style.backgroundColor = maxColor;
	// lowest z-index since it is the bottom bar.  Other bars will have
higher values.
	scaleBar.style.zIndex = "1";
	scaleBar.className = 'largeBar';
	bul.appendChild(scaleBar);
	var m1Bar = document.createElement("div");
	if(orient == "v"){
		m1Bar.style.width = scaleW + 'px';
		m1Bar.style.height = (scaleH * m1) + 'px';
		m1Bar.style.bottom = 0 + 'px';
	}else{
		m1Bar.style.height = scaleH + 'px';
		m1Bar.style.width = (scaleW * m1) + 'px';
	}
	m1Bar.style.backgroundColor = m1Color;
	m1Bar.style.zIndex = "2";
	m1Bar.className = 'largeBar';
	bul.appendChild(m1Bar);
	var m2Bar = document.createElement("div");
	if(orient == "v"){
		m2Bar.style.width = scaleW + 'px';
		m2Bar.style.height = (scaleH * m2) + 'px';
		m2Bar.style.bottom = 0 + 'px';
	}else{
		m2Bar.style.height = scaleH + 'px';
		m2Bar.style.width = (scaleW * m2) + 'px';
	}
	m2Bar.style.backgroundColor = m2Color;
	m2Bar.style.zIndex = "3";
	m2Bar.className = 'largeBar';
	bul.appendChild(m2Bar);
	var perBar = document.createElement("div");
	if(orient == "v"){
		perBar.style.width = (scaleW/3) + 'px';
		perBar.style.height = (scaleH * per) + 'px';
		perBar.style.bottom = 0 + 'px';
	} else{
		perBar.style.height = (scaleH/3) + 'px';
		perBar.style.width = (scaleW * per) + 'px';
	}
	perBar.style.backgroundColor = perColor;
	perBar.style.zIndex = "4";
	perBar.className = 'perBar';
	bul.appendChild(perBar);
	if(orient == "v"){
		perBar.style.left = (scaleW/3) + 'px';
	}else{
		perBar.style.top = (scaleH/3) + 'px';
	}
	var benBar = document.createElement("div");
	if(orient == "v"){
```

```
			benBar.style.width = (scaleW - (scaleW/4)) + 'px';
			benBar.style.height = (scaleH/60) + 'px';
		}else{
			benBar.style.height = (scaleH - (scaleH/4)) + 'px';
			benBar.style.width = (scaleW/60) + 'px';
		}
		benBar.style.backgroundColor = perColor;
		benBar.style.zIndex = "5";
		benBar.className = 'benBar';
		bul.appendChild(benBar);
		if(orient == "v"){
			benBar.style.left = ((scaleW/4)/2) + 'px';
			benBar.style.bottom = (scaleH * bench) + 'px';
		}else{
			benBar.style.top = ((scaleH/4)/2) + 'px';
			benBar.style.left = (scaleW * bench) + 'px';
		}
		/*
		number of tic marks include the tic for zero.  So if you want tics for
1-5, the value in placeTics
		should be 6 in order to include the 0 tic
		*/
		if(ticFlag == 1){
			placeTics(orient, ticScale, 6, bul);
		}
		// If there is an error to display due to an empty field, display that
error.
		if(colorErr.length > 0){
			if(colorErr.length === 1){
				this.Element.innerHTML = "Please enter a hex or rgb color
value in the " + colorErr[0] + " box";
			}else{
				this.Element.innerHTML = "Please enter a hex or rgb color
value in the ";
				for(var i=0;i < colorErr.length; i++){
					if(i != 0){
						this.Element.innerHTML += ", ";
					}
					if((i + 1) === colorErr.length){
						this.Element.innerHTML += "and ";
					}
					this.Element.innerHTML += colorErr[i];
				}
				this.Element.innerHTML += " boxes";
			}
			this.Element.style.font = "11px arial,sans-serif";
			this.Element.style.color = "red";
		}
});
```

bullet.css

```
#bulletContainer{
	display:block;position:relative;
```

```
}
.largeBar, .perBar, .benBar{position:absolute;display:block;}
.tic{display:block;position:absolute;z-index:7;}
```

Appendix B - Calendar

Script.js

```
<!--

//
// Originally written by Keith Jenci
// www.mredkj.com
//
// June 08, 2002 - version 1.0.0
// November 14, 2002 - version 1.1.0
//

// Constants - v.1.1.0 - these were taken out of CalendarFormatter and made
global.

var NO_DAY_CALENDAR = 99; // set to something other than a valid day (1-31).
For example, 99
var NO_NAV_MONTH = -1;
var MONTHS_CALENDAR = new Array("January", "February", "March", "April",
"May", "June", "July", "August", "September", "October", "November",
"December");
var lowMonth = new Array("Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul",
"Aug", "Sep", "Oct", "Nov", "Dec");
var   DAYS_1_CALENDAR = new Array("S", "M", "T", "W", "T", "F", "S");
var DAYS_2_CALENDAR = new Array("Su", "Mo", "Tu", "We", "Th", "Fr", "Sa");
var DAYS_3_CALENDAR = new Array("Sun", "Mon", "Tue", "Wed", "Thu", "Fri",
"Sat");
var DAYS_FULL_CALENDAR = new Array("Sunday", "Monday", "Tuesday",
"Wednesday", "Thursday", "Friday", "Saturday");

/*
 * Use this to construct a CalendarDisplay object.
 * There are other functions in this JavaScript file
 * that act as instance methods for CalendarDisplay.
 *
 * Example:
 * var calendarEx = new CalendarDisplay();
 *
 * Also refer to example in setNavigationOn
 *
 * v.1.1.0 - modified - Added link and open functionality.
 */
function CalendarDisplay()
{
      // Day Format constants. The actual number values aren't important.
      // setDayFormat uses these.
```

```
	this.ONE_LETTER = 1;
	this.TWO_LETTER = 2;
	this.THREE_LETTER = 3;
	this.FULL_LETTER = 4;

	// The name is used when constructing links for navigation
	this.name = "";

	// general settings and methods
	this.dayFormat = this.ONE_LETTER;
	this.setDayFormat = setDayFormatCD;
	this.createYear = createYearCD;
	this.createMonth = createMonthCD;
	this.createMonthInternal = createMonthInternalCD;
	this.formatter;
	this.monthValidated = "";
	this.dayValidated = "";
	this.yearValidated = "";
	this.errorThrown = false;

	// settings and methods for links
	this.linkIsOn = false;
	this.linkDayHandlerName = ""; // This is set by implementing logic when
calling setLinkOn
	this.setLinkOn = setLinkOnCD;
	this.setLinkOff = setLinkOffCD;

	// settings and methods for open window
	this.openIsOn = false;
	this.setOpenOn = setOpenOnCD;
	this.setOpenOff = setOpenOffCD;
	this.autoCloseIsOn = false;
	this.setAutoCloseOn = setAutoCloseOnCD;
	this.setAutoCloseOff = setAutoCloseOffCD;
	this.windowWidth = 190; // This is adjusted in setDayFormat
	this.windowHeight = 235;
	this.windowWidthYear = 690; // This is adjusted in setDayFormat
	this.windowHeightYear = 600;

	// navigation settings and methods
	this.navigateMonth = navigateMonthCD;
	this.navigateYear = navigateYearCD;
	this.navigationIsOn = false;
	this.setNavigationOn = setNavigationOnCD;
	this.setNavigationOff = setNavigationOffCD;
}

/*
 * Set the day name format.
 * Once a CalendarDisplay object is created, can use the variables:
 * ONE_LETTER, TWO_LETTER, THREE_LETTER, FULL_LETTER
 *
 * Example:
```

```
 * var calendarEx = new CalendarDisplay();
 * calendarEx.setDayFormat(calendarEx.TWO_LETTER);
 *
 * v.1.1.0 - modified - Validate specifically for each format instead of
using a range.
 *
 */
function setDayFormatCD(dayF)
{
	this.dayFormat = dayF;
	if (dayF == this.ONE_LETTER || dayF == this.TWO_LETTER)
	{
		this.windowWidth = 190;
		this.windowWidthYear = 690;
	}
	else if (dayF == this.THREE_LETTER)
	{
		this.windowWidth = 280;
		this.windowWidthYear = 750;
	}
	else if (dayF == this.FULL_LETTER)
	{
		this.windowWidth = 430;

		// not recommended to use full letter with the year calendar
		this.windowWidthYear = 800;
	}
	else
	{
		// Could validate here, but let the formatter display #ERR
	}

}

/*
 * Create and return the HTML for all the months in a calendar year,
 * given the month, day, and year.
 * Month is zero based, which means January is 0, February is 1, etc.
 *
 * The year should be valid, but month and day are not required,
 * so they can be set to any value out of range.
 * It's recommended to use "" or NO_DAY_CALENDAR.
 * If a valid month and day are passed in, then that day will be highlighted.
 *
 * If setOpenOn() has been called, then the year is also opened in a separate
window.
 *
 * Example (for December 21, 2002):
 * var calendarEx = new CalendarDisplay();
 * document.writeln(calendarEx.createYear(11, 21, 2002));
 *
 * v.1.1.0 - modified - Added open window logic.
 */
```

```
function createYearCD(cdMonth, cdDay, cdYear)
{
	this.formatter = new CalendarFormatter();

	var r;
	var c;
	var incr = 0;

	this.formatter.setupCalendarYear();

	for (r=0; r<3; r++)
	{
		this.formatter.startRow();

		for (c=0; c<4; c++)
		{
			this.formatter.startColumn();
			var dd = (cdMonth == incr) ? cdDay : NO_DAY_CALENDAR;
			this.createMonthInternal(incr, dd, cdYear);
			this.formatter.endColumn();
			incr++;
		}

		this.formatter.endRow();
	}

	this.formatter.concludeCalendarYear();

	if (this.openIsOn)
	{
		// setup navigation
		var prefixFormatter = new CalendarFormatter();
		if (this.navigationIsOn && !this.errorThrown)
		{
			var aYear = this.yearValidated;
			prefixFormatter.addNavigationYear(aYear, this.name);
		}

		var settings = "scrollbars=yes,menubar=yes,resizable=yes,width="
+ this.windowWidthYear + ",height=" + this.windowHeightYear;
		var calendarWindowYear = window.open("","CalendarWindowYear",
settings);
		calendarWindowYear.document.open();

	calendarWindowYear.document.writeln("<HTML><HEAD><TITLE>Calendar</TITLE
></HEAD>");
		calendarWindowYear.document.writeln("<BODY
onLoad='self.focus()'>");
		calendarWindowYear.document.writeln(prefixFormatter.results);
		calendarWindowYear.document.writeln(this.formatter.results);
		calendarWindowYear.document.writeln("</BODY></HTML>");
		calendarWindowYear.document.close();
	}
```

```
	return this.formatter.results;
}

/*
 * Create a month and return the HTML for a month,
 * given a month, day, and year.
 * Month is zero based, which means January is 0, February is 1, etc.
 *
 * The year and month should be valid, but day is not required,
 * so it can be set to any value out of range.
 * It's recommended to use "" or NO_DAY_CALENDAR.
 * If a day is passed in, then that day will be highlighted.
 *
 * If setOpenOn() has been called, then the month is also opened in a
separate window.
 *
 * Example (for December 21, 2002):
 * var calendarEx = new CalendarDisplay();
 * document.writeln(calendarEx.createMonth(11, 21, 2002));
 *
 * v.1.1.0 - modified - Added open window logic.
 */
function createMonthCD(cdMonth, cdDay, cdYear)
{
	// this formatter is used in createMonthInternal
	this.formatter = new CalendarFormatter();

	// createMonthInternal does validation, so call it first
	var	calendarWindowInternal = this.createMonthInternal(cdMonth, cdDay,
cdYear);

	if (this.openIsOn)
	{
		// setup navigation
		var prefixFormatter = new CalendarFormatter();
		if (this.navigationIsOn && !this.errorThrown)
		{
			var aMonth = this.monthValidated;
			var aYear = this.yearValidated;
			prefixFormatter.addNavigation(aMonth, aYear, this.name);
		}

		// open window
		var settings = "scrollbars=yes,menubar=yes,resizable=yes,width="
+ this.windowWidth + ",height=" + this.windowHeight;
		var calendarWindow = window.open("","CalendarWindowMonth",
settings);
		calendarWindow.document.open();

	calendarWindow.document.writeln("<HTML><HEAD><TITLE>Calendar</TITLE></H
EAD>");
		calendarWindow.document.writeln("<BODY onLoad='self.focus()'>");
```

```
		calendarWindow.document.writeln(prefixFormatter.results);
			calendarWindow.document.writeln(calendarWindowInternal);
			calendarWindow.document.writeln("</BODY></HTML>");
			calendarWindow.document.close();
		}

		return calendarWindowInternal;
}

/*
 * This function is a middle man to call createMonth for the navigation in
createMonth.
 *
 * Original purpose: If the navigation buttons are recoded to be href links,
 * then will need this so the return value doesn't attempt to load an invalid
page.
 * The href navigation links were slow in Netscape 4.
 *
 * Now this method is used so the buttons don't have to include a dummy day.
 *
 * v.1.1.0 - new
 */
function navigateMonthCD(cdMonth, cdYear)
{
		var cdDay = NO_DAY_CALENDAR;
		this.createMonth(cdMonth, cdDay, cdYear);
}

function navigateYearCD(cdYear)
{
		var cdMonth = NO_NAV_MONTH;
		var cdDay = NO_DAY_CALENDAR;
		this.createYear(cdMonth, cdDay, cdYear);
}

/*
 * Create a month and return it.
 * Used internally by createMonth and createYear.
 *
 * v.1.1.0 - modified - Using addFormattedLinkDay now instead of
addFormattedDay.
 */
function createMonthInternalCD(cdMonth, cdDay, cdYear)
{
		if (isNaN(cdDay) || cdDay == "")
		{
			cdDay = NO_DAY_CALENDAR;
		}

		this.dayValidated = cdDay;
		this.formatter.dayChosen = cdDay;

		if (isNaN(cdMonth) || cdMonth < 0 || cdMonth >= MONTHS_CALENDAR.length)
```

```
	{
		this.errorThrown = true;
		this.monthValidated = 0; // monthValidated is set to 0 to limit
JavaScript errors
		return "Not a valid month.";
	}
	this.errorThrown = false;
	this.monthValidated = cdMonth;

	if (isNaN(cdYear) || cdYear == "")
	{
		this.errorThrown = true;
		this.yearValidated = 0; // yearValidated is set to 0 to limit
JavaScript errors
		return "Not a valid year.";
	}
	this.errorThrown = false;
	this.yearValidated = cdYear;

	var startingDay = 1;
	var aDate = new Date(cdYear, cdMonth, startingDay);

	var DAY_ROW = 0;
	var SPACE_BEFORE = 1;
	var DAYS_BETWEEN = 2;
	var SPACE_AFTER = 3;
	var phase = SPACE_BEFORE;

	this.formatter.setupCalendarMonth(cdMonth, cdYear);

	var rows;
	for (rows = 0; rows < 6; rows++)
	{
		this.formatter.startRow();
		var cols;
		for (cols = 0; cols < 7; cols++)
		{
			if (phase == DAY_ROW)
			{
				if (cols >= 6) phase = SPACE_BEFORE;
			}
			else if (phase == SPACE_BEFORE)
			{
				var tempDay = aDate.getDay();
				if (cols == tempDay)
				{
					var autoCloseReallyOn = (this.autoCloseIsOn &&
this.openIsOn);
					this.formatter.addFormattedLinkDay(1,
this.linkIsOn, this.linkDayHandlerName, this.openIsOn, autoCloseReallyOn,
this.objectName, aDate);
					aDate.setDate(2);
					phase = DAYS_BETWEEN;
```

```
                        }
                        else
                        {
                              this.formatter.addFormattedDayBlank();
                        }
                  }
                  else if (phase == DAYS_BETWEEN)
                  {
                        // After using setDate, calling getMonth will
                        // verify we're still in the current month
                        if (aDate.getMonth() == cdMonth)
                        {
                              var dayOfMonth = aDate.getDate();
                              var autoCloseReallyOn = (this.autoCloseIsOn &&
this.openIsOn);
                              this.formatter.addFormattedLinkDay(dayOfMonth,
this.linkIsOn, this.linkDayHandlerName, this.openIsOn, autoCloseReallyOn,
this.objectName, aDate);
                              aDate.setDate(dayOfMonth + 1);
                        }
                        else
                        {
                              this.formatter.addFormattedDayBlank();
                              phase = SPACE_AFTER;
                        }
                  }
                  else if (phase == SPACE_AFTER)
                  {
                        this.formatter.addFormattedDayBlank();
                  }
            } // end for cols

            this.formatter.endRow();

      } // end for rows

      this.formatter.concludeCalendarMonth();

      return this.formatter.results;

}

/*
 * Set a switch on so days in the calendar will show up as links.
 *
 * The handlerName should be the name of a method that is available
 * from any link that has javascript: in it.
 *
 * It should expect the arguments in the order: month, day, year.
 * Month is the zero based month (from 0 to 11).
 * Day is the number day (from 1 to 31)
 * Year is the full year (example 2002).
 *
```

```
 * Right now, it is not set up to call a method within CalendarDisplay,
 * so CalendarDisplay does not know which date link was clicked.
 * That information must be maintained in the implementing logic.
 *
 * v.1.1.0 - new
 */
function setLinkOnCD(handlerName)
{
	this.linkIsOn = true;
	this.linkDayHandlerName = handlerName;
}

/*
 * Set a switch off so days in the calendar will not show up as links.
 *
 * v.1.1.0 - new
 */
function setLinkOffCD()
{
	this.linkIsOn = false;
}

/*
 * Turn navigation on for the month or year calendar.
 *
 * For a month calendar, there will be buttons like << < > >>
 * For a year calendar, there will buttons like << >>
 * which are:
 * << previous year
 * <  previous month
 * >  next month
 * >> next year
 *
 * To include navigation buttons, provide the name of the object that was
instantiated.
 *
 * Make sure the calendar object is global by constructing it without a var
if it is inside a function.
 * If the object is declared outside a function, then it will be global
either way.
 * It needs to be global so the navigation button's onclick event will be
able to reference the object.
 *
 * Example:
 * calendarWithNavEx = new CalendarDisplay();
 * calendarWithNavEx.setNavigationOn("calendarWithNavEx");
 *
 * v.1.1.0 - new
 */
function setNavigationOnCD(objectName)
{
	this.navigationIsOn = true;
	this.name = objectName;
```

```
	if (this.name == null) this.name = "";
}

/*
 * Turn navigation off.
 *
 * v.1.1.0 - new
 */
function setNavigationOffCD()
{
	this.navigationIsOn = false;
}

/*
 * Set a switch on so the calendar will open in a new window.
 * It will still return the calendar result.
 *
 * v.1.1.0 - new
 */
function setOpenOnCD()
{
	this.openIsOn = true;
}

/*
 * Set a switch off so the calendar will not open in a new window.
 * It will just return the calendar result.
 *
 * v.1.1.0 - new
 */
function setOpenOffCD()
{
	this.openIsOn = false;
}

/*
 * Set that the opened window should be closed when a date is selected.
 *
 * v.1.1.0 - new
 */
function setAutoCloseOnCD()
{
	this.autoCloseIsOn = true;
}

/*
 * Set that the opened window should not be closed when a date is selected.
 *
 * v.1.1.0 - new
 */
function setAutoCloseOffCD()
{
	this.autoCloseIsOn = false;
```

```
}

/*
 * CalendarDisplay uses this to construct a CalendarFormatter object
 * which it uses to form the HTML for the calendar.
 *
 * There are other functions in this JavaScript file
 * that act as instance methods for CalendarFormatter.
 */
function CalendarFormatter()
{
      this.results = "";

      this.dayChosen = NO_DAY_CALENDAR;
      this.setupCalendarYear = setupCalendarYearCF;
      this.setupCalendarMonth = setupCalendarMonthCF;
      this.concludeCalendarYear = concludeCalendarYearCF;
      this.concludeCalendarMonth = concludeCalendarMonthCF;
      this.addFormattedDay = addFormattedDayCF;
      this.addFormattedLinkDay = addFormattedLinkDayCF;
      this.addFormattedNonDay = addFormattedNonDayCF;
      this.addFormattedDayName = addFormattedDayNameCF;
      this.addFormattedDayBlank = addFormattedDayBlankCF;
      this.startRow = startRowCF;
      this.endRow = endRowCF;
      this.startColumn = startColumnCF;
      this.endColumn = endColumnCF;

      this.addNavigation = addNavigationCF;
      this.addNavigationYear = addNavigationYearCF;
      this.getPreviousYearLink = getPreviousYearLinkCF;
      this.getPreviousMonthLink = getPreviousMonthLinkCF;
      this.getNextMonthLink = getNextMonthLinkCF;
      this.getNextYearLink = getNextYearLinkCF;

}

function setupCalendarYearCF(month, year)
{
      this.results = this.results + '<table class="calTable" border="1"
cellpadding="5" cellspacing="0">';
}

function setupCalendarMonthCF(month, year)
{
      this.results = this.results + '<table class="calTable">';
}

function concludeCalendarYearCF()
{
      this.results = this.results + "</table>";
}
```

```
function concludeCalendarMonthCF()
{
	this.results = this.results + "</table>";
}

/*
 * Add a formatted day to the results with no link.
 *
 * This function isn't used internally in CalendarDisplay in v.1.1.0.
 * The code has been moved to addFormattedLinkDay.
 *
 * v.1.1.0 - modified
 */
function addFormattedDayCF(num)
{
	var isLinkOn = false;
	var objectName = ""; // ignored when link is off
	var month; // ignored when link is off
	var day; // ignored when link is off
	var year; // ignored when link is off
	this.addFormattedLinkDay(num, isLinkOn);
}

/*
 * Add a formatted day to the results with a link, if isLinkOn is true.
 * The way the link is constructed, the function specified by handlerName is
referenced
 * with the zero based month, day, and full year.
 * Keep in mind the handlerName needs to be referenced globally.
 *
 * Also, it will refer to the opener if isWindowOpen is true,
 * which means the calendar is in an opened window.
 *
 * v.1.1.0 - added
 */
function addFormattedLinkDayCF(num, isLinkOn, handlerName, isWindowOpen,
autoCloseIsOn, objectName, aDate)
{
	var linkStart = "";
	var linkEnd = "";
	if (isLinkOn)
	{
		var aYear = aDate.getFullYear();
		var aMonth = aDate.getMonth();
		var aDay = aDate.getDate();
		linkStart = "<a href=\"javascript:"
			+ ((isWindowOpen) ? "window.opener." : "")
			+ handlerName + "("
			+ aMonth + ","
			+ aDay + ","
			+ aYear + ");"
			+ ((autoCloseIsOn) ? "window.close();" : "")
			+ "\">";
```

```
			linkEnd = "</a>";
		}
		var d = new Date();
		if ((num == this.dayChosen) && (d.getMonth() === aDate.getMonth()))
		{
			this.results = this.results + '<td align="center" class="today"
id="day' + num + '"><b>' + linkStart + num + ' '  +
DAYS_3_CALENDAR[aDate.getDay()] + linkEnd + '</b></td>';
		}
		else
		{
			this.results = this.results + '<td align="center" id="day' + num
+ '">' +  linkStart + num + ' '  + DAYS_3_CALENDAR[aDate.getDay()] + linkEnd
+ '</td>';
		}
}

function addFormattedNonDayCF(contents)
{
		this.results = this.results + '<td align="center">' + contents +
'</td>';
}

function addFormattedDayNameCF(cal, num)
{
		if (cal.dayFormat == cal.ONE_LETTER)
		{
			this.addFormattedNonDay(DAYS_1_CALENDAR[num]);
		}
		else if (cal.dayFormat == cal.TWO_LETTER)
		{
			this.addFormattedNonDay(DAYS_2_CALENDAR[num]);
		}
		else if (cal.dayFormat == cal.THREE_LETTER)
		{
			this.addFormattedNonDay(DAYS_3_CALENDAR[num]);
		}
		else if (cal.dayFormat == cal.FULL_LETTER)
		{
			this.addFormattedNonDay(DAYS_FULL_CALENDAR[num]);
		}
		else
		{
			this.addFormattedNonDay("#ERR");
		}
}

function addFormattedDayBlankCF()
{
		this.addFormattedNonDay(" ");
}

function startRowCF()
```

```
{
	this.results = this.results + "<tr>";
}

function endRowCF()
{
	this.results = this.results + "</tr>";
}

function startColumnCF()
{
	this.results = this.results + "<td>";
}

function endColumnCF()
{
	this.results = this.results + "</td>";
}

function addNavigationCF(aMonth, aYear, aName)
{
	this.results = this.results
		+ "<p><form name=\"navigationCalendar\">"
		+ this.getPreviousYearLink(aMonth, aYear, aName)
		+ " " + this.getPreviousMonthLink(aMonth, aYear, aName)
		+ " " + this.getNextMonthLink(aMonth, aYear, aName)
		+ " " + this.getNextYearLink(aMonth, aYear, aName)
		+ "</form></p>";
}

function addNavigationYearCF(aYear, aName)
{
	var aMonth = NO_NAV_MONTH;
	this.results = this.results
		+ "<p><form name=\"navigationCalendar\">"
		+ this.getPreviousYearLink(aMonth, aYear, aName)
		+ " " + this.getNextYearLink(aMonth, aYear, aName)
		+ "</form></p>";
}

function getPreviousYearLinkCF(aMonth, aYear, aName)
{
	var previousYear = aYear;
	previousYear--;
	var previousYearLink = "<input type=\"button\" value=\"<<\"
onclick=\"javascript:window.opener.";
	if (aMonth == NO_NAV_MONTH)
	{
		previousYearLink += aName + ".navigateYear(";
	}
	else
	{
		previousYearLink += aName + ".navigateMonth(" + aMonth + ",";
```

```
	}
	previousYearLink += previousYear + ")\" />";

	return previousYearLink;
}

function getPreviousMonthLinkCF(aMonth, aYear, aName)
{
	var previousMonth = aMonth;
	var previousYear = aYear;
	previousMonth--;
	if (previousMonth < 0)
	{
		previousMonth = 11;
		previousYear--;
	}
	var previousMonthLink = "<input type=\"button\" value=\"< \"
onclick=\"javascript:window.opener."
		+ aName + ".navigateMonth("
		+ previousMonth + ","
		+ previousYear + ")\" />";

	return previousMonthLink;
}

function getNextMonthLinkCF(aMonth, aYear, aName)
{
	var nextMonth = aMonth;
	var nextYear = aYear;

	// note: use nextMonth++ instead of nextMouth = nextMonth + 1,
	// because the second one treats nextMonth as a string.
	nextMonth++;
	if (nextMonth > 11)
	{
		nextMonth = 0;
		nextYear++;
	}
	var nextMonthLink = "<input type=\"button\" value=\" >\"
onclick=\"javascript:window.opener."
		+ aName + ".navigateMonth("
		+ nextMonth + ","
		+ nextYear + ")\" />";

	return nextMonthLink;
}

function getNextYearLinkCF(aMonth, aYear, aName)
{
	var nextYear = aYear;
	nextYear++;
```

```
	var nextYearLink = "<input type=\"button\" value=\">>\"
onclick=\"javascript:window.opener."
	if (aMonth == NO_NAV_MONTH)
	{
		nextYearLink += aName + ".navigateYear(";
	}
	else
	{
		nextYearLink += aName + ".navigateMonth(" + aMonth + ",";
	}
	nextYearLink += nextYear + ")\" />";

	return nextYearLink;
}
function checkArray(arr, str){
    var returnIt = -1;
    for(var t=0;t<arr.length;t++){
        if(arr[t].indexOf(str) > -1){
            returnIt = t;
        }
    }
    return returnIt;
}
//-->
// Create an array of colors to use for the different event types.
var colorArray = new Array('#3399cc', '#cc6666', '#7ba550', 'ffcc66',
'#d17314', '#b974fd', '#993300', '#99cccc', '#669933', 'blue',
'DarkOliveGreen','red','DarkMagenta', 'SaddleBrown', 'DarkOrange',
'DarkCyan', 'DarkRed','DarkSlateGrey', 'Green', 'MediumVioletRed',
'MidnightBlue', 'HotPink','Olive','PaleVioletRed','Orange','DarkSeaGreen');
Qva.LoadScript('/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensions/Qli
kView/Examples/calendar/jquery.js', function () { //Load jquery
	Qva.AddExtension('QlikView/Examples/calendar', function() {
	//Load CSS

Qva.LoadCSS("/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensions/QlikVi
ew/Examples/calendar/main.css");
	this.Element.innerHTML = "";
	var view = "w";  //view variable will tell the code if it should
display in week mode of month mode
	  var campType = new Array();  //Array to capture the different
campaign types
	var dayArr = new Array();
	this.Element.className += " calFrame";
	var calData = [];  //JSON object where the calendar data will be
stored
	var thisMonth = "none";
	var thisYear = "none";
	  var thisCamp = "none";
	  var testWeek = "none";
	var thisWeek = [];
	var errorMsg = "";
```

```
            //Loop through the data and create and store the JSON that will
be used to build the extension
        for (var rowIx = 0; rowIx < this.Data.Rows.length; rowIx++) {
                    //Check to make sure a month has been chosen
                    if(thisMonth == "none"){
                          thisMonth = this.Data.Rows[rowIx][3].text;
                    }else if(thisMonth != this.Data.Rows[rowIx][3].text){
                          this.Element.innerHTML += "Please Choose a Month";
                          errorMsg = "true";
                          break;
                    }
                    //Check that a year has been chosen
                    if(thisYear == "none"){
                          thisYear = this.Data.Rows[rowIx][4].text;
                    }else if(thisYear != this.Data.Rows[rowIx][4].text){
                          this.Element.innerHTML += "Please Choose a Year";
                          errorMsg = "true";
                          break;
                    }
                    //See if a week has been selected
                    if(testWeek == "none"){
                          testWeek = this.Data.Rows[rowIx][7].text;
                    }else if(testWeek != this.Data.Rows[rowIx][7].text){
                          view = "m";
                    }
                    //append data to JSON object

      calData.push({"eventWeekStart":this.Data.Rows[rowIx][5].text,
"eventWeekEnd":this.Data.Rows[rowIx][6].text,
"campaignSubType":this.Data.Rows[rowIx][1].text,
"campaignName":this.Data.Rows[rowIx][0].text,
"description":this.Data.Rows[rowIx][2].text});
        }
            thisMonth = $.inArray(thisMonth, lowMonth) + 1;
                  //MONTH VIEW
            if ((view == "m") && (errorMsg != "true")) {
                          var nowDate = new Date();

                          var calendar = new CalendarDisplay();
                          calendar.setDayFormat(calendar.THREE_LETTER);
                          // Use calendar library to creat the month
                          this.Element.innerHTML +=
calendar.createMonth((thisMonth - 1), nowDate.getDate(), thisYear);
                          this.Element.innerHTML += "<div
id='eventList'><h2>Types</h2><div id='listContent'></div></div>";
                          // loop through JSON
                          for (var q = 0; q < calData.length; q++) {
                                var thisData = calData[q];
                                var sSplit =
thisData.eventWeekStart.split('/');
                                var eSplit = thisData.eventWeekEnd.split('/');
                                //convert date in the object to javascript date
object
```

```
                                var sDate = new Date(sSplit[2], (sSplit[0] -
1), sSplit[1]);
                                var eDate = new Date(eSplit[2], (eSplit[0] -
1), eSplit[1]);

                                //Check if this row's campaign name is already
in the list of campaign types.  If not, add it.
                                var arrLoc = checkArray(campType,
thisData.campaignSubType);
                                if (arrLoc == -1) {
                                      arrLoc = campType.length;
                                      campType.push(thisData.campaignSubType);
                                }

                                while (sDate.valueOf() < eDate.valueOf() +
86400000) {
                                      if ((sDate.getMonth() + 1) == thisMonth)
{
                                            //select the element representing
the current day in the loop
                                            var thisDay =
document.getElementById("day" + sDate.getDate());
                                            var checkUL =
document.getElementById("day" + sDate.getDate() + "List");
                                            // Check if a list of items for
that day already exists.  If not, add a list
                                            if (checkUL == null) {
                                                  var newUL =
document.createElement("ul");
                                                  newUL.setAttribute('id',
'day' + sDate.getDate() + 'List');
                                                  newUL.setAttribute('class',
'day' + sDate.getDate() + 'List');
                                                  thisDay.appendChild(newUL);
                                                  checkUL = newUL;
                                            }
                                            // Add a new LI for the event that
day
                                            var newLI =
document.createElement("li");
                                            newLI.style.color =
colorArray[arrLoc];
                                            newLI.innerHTML =
thisData.campaignName;
                                            checkUL.appendChild(newLI);
                                      }

                                      //move on to the next day
                                      sDate.setTime(sDate.valueOf() +
86400000);

                                }
```

```
                        }
                        // Fill list of campaigns using the campType array
                        var eL = document.getElementById("listContent");
                        var fullHTML = "";
                        if (campType.length > 0) {
                              fullHTML += "<ul>";
                              for (var z = 0; z < campType.length; z++) {
                                    fullHTML += "<li style='color:" +
colorArray[z] + "'>" + campType[z] + "</li>";
                              }
                              fullHTML += "</ul>";
                              eL.innerHTML += fullHTML;
                        }
                        // Create fly out
                        this.Element.innerHTML += '<div id="fo"><img
src="/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensions/QlikView/Examp
les/calendar/x.png" class="xImg"/><div id="f_top"></div><div id="f_mid"><div
id="f_arrow"></div><div id="content"></div></div><div id="f_btm"></div>';
                        $('.xImg').click(function(){
                              $('#fo').hide();
                        });
                        $('.calTable td').click(function(){
                              if (($(this).html() != " ") &&
($(this).find('ul').length > 0)) {
                                    $('#fo').hide();
                                    $('#fo #content').html('<ul>' +
$(this).find('ul').html() + '</ul>');
                                    $('#fo').css('left',
($(this).position().left + 90) + 'px');
                                    $('#fo').css('top',
($(this).position().top - 10) + 'px');
                                    $('#fo').show();
                              }
                        });
                  // WEEK VIEW
                  }else if((view == "w") && (errorMsg != "true")){
                  var weekT = $('<table />').addClass('weekView');
                        var sSplit = testWeek.split('/');
                        var sDate = new Date(sSplit[2], (sSplit[0] - 1),
sSplit[1]);
                        var eDate = new Date();
                        eDate.setTime(sDate.valueOf() + (86400000 * 6));

               while (sDate.valueOf() < eDate.valueOf() + 86400000) {
                              // create the weekday containers
                     var weekday = $('<td />').addClass('day' +
sDate.getDate()).text(sDate.getDate() + ' ' +
DAYS_3_CALENDAR[sDate.getDay()]);
                     weekT.append(weekday);
                              sDate.setTime(sDate.valueOf() + 86400000);
                  }
                  $('.calFrame').append(weekT);
```

```
                        for (var q = 0; q < calData.length; q++) {
                              var thisData = calData[q];
                              var sSplit =
thisData.eventWeekStart.split('/');
                              var eSplit = thisData.eventWeekEnd.split('/');
                              var sDate = new Date(sSplit[2], (sSplit[0] -
1), sSplit[1]);
                              var eDate = new Date(eSplit[2], (eSplit[0] -
1), eSplit[1]);

                              var arrLoc = checkArray(campType,
thisData.campaignSubType);
                              if (arrLoc == -1) {
                                    arrLoc = campType.length;
                                    campType.push(thisData.campaignSubType);
                              }
                              //Loop through the JSON
                              while (sDate.valueOf() < eDate.valueOf() +
86400000) {
                                    if ((sDate.getMonth() + 1) == thisMonth)
{
                                          var thisDay = $(".day" +
sDate.getDate());
                                          //var checkUL = $("#day" +
sDate.getDate() + "List");
                                          //this.Element.innerHTML +=
checkUL;
                                          if ($("#day" + sDate.getDate() +
"List").length == 0){
                                                var newUL = $('<ul
/>').attr('id', 'day' + sDate.getDate() + 'List').addClass('day' +
sDate.getDate() + 'List');
                                                thisDay.append(newUL);

                                          }
                                          var checkUL = $('#day' +
sDate.getDate() + 'List');
                                          var liFullHTML = "<a
class='campNameLink'>" + thisData.campaignName + "</a><div
class='moreInfo'><h4>" + thisData.campaignName + "</h4><p><strong>Event
Type:</strong> " + thisData.campaignSubType + "<br />";
                                          if (thisData.description != "") {
                                                liFullHTML += "<strong>Event
Description: </strong>" + thisData.description + "<br />";
                                          }
                                          liFullHTML += "</p></div>";
                                          var newLI = $('<li
/>').css('color',colorArray[arrLoc]).html(liFullHTML);
                                          checkUL.append(newLI);
                                    }

                                    sDate.setTime(sDate.valueOf() +
86400000);
```

```
                        }
                    }
                    //attach flyout
                    this.Element.innerHTML += '<div id="fo"><img
src="/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensions/QlikView/Examp
les/calendar/x.png" class="xImg"/><div id="f_top"></div><div id="f_mid"><div
id="f_arrow"></div><div id="content"></div></div><div id="f_btm"></div>';
                $('.calFrame').append("<div id='eventList'><h2>Types</h2><div
id='listContent'></div></div>");
                    var eL = $("#listContent");
                    var fullHTML = "";
                    if (campType.length > 0) {
                        fullHTML += "<ul>";
                        for (var z = 0; z < campType.length; z++) {
                            fullHTML += "<li style='color:" +
colorArray[z] + "'>" + campType[z] + "</li>";
                        }
                        fullHTML += "</ul>";
                        eL.append(fullHTML);
                    }
                    //Attach event to close flyout when the X is clicked
                    $('.xImg').click(function(){
                        $('#fo').hide();
                    });
                    //Attach click event to load the flyout when an event
is clicked
                    $('.campNameLink').click(function(){
                        $('#fo').hide();
                        //replace the old html with the new data
                        $('#fo
#content').html($(this).parent().find('.moreInfo').html());
                        $('#fo #content h4').css('color',
$(this).parent().css("color"));
                        $('#fo').css('left', ($(this).position().left +
65) + 'px');
                        $('#fo').css('top', ($(this).position().top -
33) + 'px');
                        $('#fo').show();
                    });
                }
    });
});
```

Main.css

```
.calFrame{color:#999;padding:10px 0px 10px 20px;font-size:12px;font-
family:helvetica, Myriad Pro, arial, sans-serif;overflow:auto;}
```

```
.calFrame td{height:73px;width:79px;border:1px solid #ccc;text-
align:left;vertical-align:top;padding:3px 0px 0px 5px;font-
weight:bold;cursor:pointer;}
.calFrame td.today{background-color:#f0f7fb;}
.calFrame h1{font-size:14px;font-weight:bold;color:#333;}
.calFrame table{float:left;}
.calFrame #eventList{float:left;border:1px solid #ccc;width:151px;border-
left:0px;margin-top:25px;height:415px;padding-left:4px;overflow:auto;}
.calFrame #eventList h2{font-size:12px; font-weight:bold;color:#333;padding-
left:3px;}
.calTable, .weekView{margin-top:15px;}
.calTable ul, .calTable ol, .weekView ul, .weekView ol  {padding:0px;padding-
left:14px;margin:0px;width:61px;overflow:hidden;height:50px;}
.weekView ul, .weekView ol  {height:434px;}
#eventList ul{padding:0px;padding-left:18px;margin:0px;overflow:hidden;}
.calFrame li{padding:0px !important;}
.calTable td li, .weekView td li{font-size:11px; font-weight:normal;list-
style:disc;height:12px;width:700px;}
#fo{width:222px;position:absolute;z-index:9999;display:none;}
#f_top{width:222px;height:8px;display:block;background:url(/QvAjaxZfc/QvsView
Client.aspx?public=only&name=Extensions/QlikView/Examples/calendar/f_top.png)
transparent no-repeat left top;}
#f_mid{width:222px;display:block;background:url(/QvAjaxZfc/QvsViewClient.aspx
?public=only&name=Extensions/QlikView/Examples/calendar/f_mid.png)
transparent repeat-y left top;position:relative;}
#f_btm{width:222px;height:9px;display:block;background:url(/QvAjaxZfc/QvsView
Client.aspx?public=only&name=Extensions/QlikView/Examples/calendar/f_btm.png)
transparent no-repeat left top;}
#f_arrow{width:11px;height:26px;display:block;background:url(/QvAjaxZfc/QvsVi
ewClient.aspx?public=only&name=Extensions/QlikView/Examples/calendar/f_arrow.
png) transparent no-repeat left top;position:absolute;top:20px;left:-9px;}
#fo #content{height:150px;overflow:auto;width:215px;}
#fo ul, #fo #content p{margin:0px;padding:0px 10px 0px
20px;width:164px;overflow:hidden;}
#fo #content h4{margin:0px;padding:0px 10px 5px 20px;}
#fo #content p{width:164px;color:#333;font-size:11px;}
#fo ul li{width:170px;}
.weekView td{width:79px;height:434px;}
#fo h4{font-size:11px;}
.weekView .campName{display:none;}
.weekView .moreInfo{display:none;clear:both;}
.xImg{position:absolute;top:10px;right:10px;cursor:pointer;z-index:9999;}
```

Appendix C - Geographic Heat Map

Script.js

```
function geoHeat_Example_Init()
{

    // Use QlikView's method of loading other files needed by an extension.
These files should be added to your extension .zip file (.qar)
```

```
	Qva.LoadScript("/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensio
ns/QlikView/Examples/geoHeat/raphael.js", geoHeat_Example_Done);

    //If more than one script is needed you can nest the calls to get them
loaded in the correct order

//Qva.LoadScript("/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensions/Q
likView/Examples/geoHeat/file1.js", function() {

//Qva.LoadScript("/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensions/Q
likView/Examples/geoHeat/file2.js", FC_Ext_Example_Done);
    //});

}
//Function to allow for RGB color entry as well as Hex (#)
function colorFormatter(c, type){
	var cc = c.toLowerCase();
	if(cc.length === 0){
		alert("Please enter a hex or rgb color value in the " + type + "
box");
	}else if((cc.indexOf("rgb") === -1) && (cc.indexOf("#") === -1)){
		if(cc.length < 6){
			var addIt = "#";
			for(var i=cc.length; i < 6;i++){
				addIt += "0";
			}
			cc = addIt + cc
		}else{
			cc = "#" + cc;
		}
		return cc;
	}else{
		return cc;
	}
}

function geoHeat_Example_Done()
{
	// Add extension creates the extension and add the function defined in
the second parameter as the OnUpdate fuction
	// for the extension in question. This function is where the rendering
code should go.
	Qva.AddExtension("QlikView/Examples/geoHeat",
	  function ()
	  {

	    var _this = this;
		//Get the various colors for the hot, cold, and disabled sections
of the map
		colorFormatter(this.Layout.Text0.text.toString(), "hot color");
		var hot = colorFormatter(this.Layout.Text0.text.toString(), "hot
color");
```

```
            var not = colorFormatter(this.Layout.Text1.text.toString(), "cold
color");
            var dis = colorFormatter(this.Layout.Text2.text.toString(), "no
data color");
            // Create random number to use in naming the div so multiple
instances of the extension will live together peacefully
          var r=Math.floor(Math.random()*10001);
            /*
                  Create container div and canvas and canvasBg divs for the
map.  The canvasBg div is meant to hold the map image which outlines the
entire
                  regional area.  This image is set to the "cold" color and
sites behind the regular canvas element which holds the individual states
             */
          _this.Element.innerHTML = '<div id="canvasBg' + r + '"
style="width:' + _this.GetWidth() + 'px;height:' + _this.GetHeight() +
'px;left: 0; position: absolute;top: 0;z-index:3;"></div><div id="canvas' + r
+ '" style="width:' + _this.GetWidth() + 'px;height:' + _this.GetHeight() +
'px;left: 0; position: absolute;top: 0;z-index:1000;"></div>';
          _this.Element.style.overflow = "hidden";
            // Initiate Raphael to recognize the canvas elements
          var R = Raphael("canvas" + r, _this.GetWidth(),_this.GetHeight());
          var R2 = Raphael("canvasBg" + r,
_this.GetWidth(),_this.GetHeight());
            /*
                  Create attributes for the two canvases.  The first attr is
set to the disabled color since each state will start that way.
                  Then the state data will be looped through and each hot
state will be colored accordingly
            */
        var attr = {
            "fill": dis,
            "stroke": "#ffffff",
            "stroke-width": 1,
            "stroke-linejoin": "round"
        }
          var attrOut = {
            "fill": not,
            "stroke": "#ffffff",
                  "fill-opacity": 1,
            "stroke-width": 1,
            "stroke-linejoin": "round"
        }
            // get the image's height and width ratio
            var imgRatio = 909/593;
            var scaleNum = 0;

            // if the window's height and width ratio is greater than this
number, it is too wide and the height should be used to scale the image
            // if the window's height and width ratio is less than this
number, it is too wide and the width should be used to scale the image
            if((_this.GetWidth()/_this.GetHeight()) > imgRatio){
                  scaleNum = _this.GetHeight()/593;
```

```
		}else{
			scaleNum = _this.GetWidth()/909;
		}
		// Use the SVG data to manually enter in the paths into Raphael
	var outline =
R2.path("M37.023,519.956l0.156,2.682l3.296,3.469l2.982,0.156l1.569-3.625l-
0.784-0.947l-3.609-0.945L37.023,519.956z M46.282,348.1l0.157-1.262l-1.255-
1.498l-3.217,0.08l0.628,1.418l0.548,1.184 L46.282,348.1z
M126.158,540.454l0.313,2.207l2.354-1.42l-0.941-1.734L126.158,540.454z
M205.248,575.928l-5.021-2.051l-4.865-1.734l-1.412-3.625l-2.197-2.682l-2.667-
3.783l-3.295-5.834l-3.766-1.734l-1.098-2.365l-1.883-1.104l-0.628-2.838l-
2.981-2.049l-1.883,1.418l-1.098,0.158l-1.412,1.576l-0.314,2.996l-
0.628,0.156l-3.295,1.893l-0.471-2.521l-6.748-6.623l-0.157-2.838h-2.511l-
1.412,1.104l-2.982-0.156l-1.569-0.947l0.313-83.248l-0.001-0.008l-4.237-2.84l-
4.708-1.891l-1.883,0.787l-2.04,0.631l-5.021-2.521l-1.412,0.945l-6.12-4.1l-
6.905-0.631l-1.883-1.418l-0.314-3.625l-3.766-0.316l-1.57,1.576l-0.627-
1.104l0.471-1.105l-1.883-2.051l-3.453-2.365h-1.098l-2.667,2.996l-
4.394,0.158l-0.941-0.789l-1.099,0.158l-3.61,2.049l-2.667,0.631l-1.57-0.943l-
3.609,2.363l-1.099,2.996l-3.61,3.471l-4.08,0.154l-1.727-0.631h-3.138l-
2.825,1.895l2.354,4.11l2.825,4.572l0.784,3.781l-
0.471,1.578l0.471,1.105l3.295,1.895l-0.941,0.789l0.941,3.154l2.197,4.256l-
1.726,0.945l-4.865-1.732l-2.667-1.578l0.471-1.418l1.412-2.051l-1.412-1.104l-
2.511-0.314l-3.767,0.631l0.157,1.576h-0.627l-0.941-0.949h-2.04l-6.59,0.473l-
0.471,0.787l1.883,3.627l2.04,1.734l-0.627,2.365l-
0.471,3.783l3.452,3.627l4.865,0.158l3.453,1.732l1.883,0.473l3.139-
0.787l2.511-0.787l1.099,1.891c0,0-0.313,0.473-0.941,0.947c-0.628,0.473-
0.942,3.311-0.942,3.311l-0.785,4.256l-2.196,1.262h-1.57l-0.941-0.631l-1.57-
0.631l-3.139,2.365l-2.354,0.318l-0.313-2.365l-2.511-1.105l-2.667,1.262l-
1.883,3.625l-2.511,1.578l-4.394,3.15l-0.941,4.889l1.098,2.682l1.255-
0.471l0.785,0.471l-1.098,1.104l-0.157,2.049l-
2.04,0.789v0.947l2.04,3.471l0.627,3.783l1.412,1.26h5.964l0.784-
1.42l0.941,0.947v4.258l0.628,1.734l-1.255,0.16l-1.413,0.945l0.627,1.893l-
0.471,2.521l-1.413,1.42l2.511,0.314l2.04-1.104l3.923-
0.158l0.314,1.893l1.099,0.473l1.255-1.262l1.412,5.676l1.413,0.16l0.156-
2.838l0.941-0.945l1.255,0.314l0.627,2.365l3.295-0.629l1.57-1.264l-
0.314,2.523l-1.57,1.42l-2.196,4.414l-1.883,3.625l-3.923,1.418l-1.255,3.625h-
0.784l-0.628-0.471l-1.255,0.314l-3.295,0.631l-2.982,0.945l-2.196,2.684l-
0.941,0.314l-1.57-1.262l-2.667-0.158l-1.727,0.789l-2.825,0.631l-2.04,2.049l-
4.237,1.578l-6.121-0.631l-2.982,1.262v1.891l1.57,1.104l0.941-0.945l2.197-
0.631l2.511,1.105l2.04,0.316l1.569-0.789l2.04-1.26h3.139l2.825-
1.893l3.295,1.26l2.825-0.947l3.452-1.26l-0.314,2.365h0.942l0.785-2.207l3.138-
0.158l2.667,0.947l0.471-1.105l1.413-1.418l-1.727-1.26l5.179-2.207l5.178-
1.105l2.04-1.104l7.533-4.11l6.277-4.104l2.667-3.15l-0.784-2.051l-2.198-
0.943l0.314-1.576l2.982-2.525l4.081-1.418l1.098-2.207l1.727-2.521l2.354-
3.312l4.865-2.682l3.609,0.316l1.255,2.523h-2.667l-0.942-1.262l-4.237,2.209l-
0.157,0.787v1.732l-2.197,3.785l-0.941,2.363l2.197,0.631v1.26l-
3.61,0.631l0.471,2.049l4.552,0.158l3.295-2.049l1.727-2.049l2.982-
2.049l2.667,1.734l1.255-1.104l1.255,0.787l0.471,1.893l1.255,0.318l3.766-
4.262l-0.627-1.574l-3.139,2.682l1.413-4.258l-0.941-0.316l-1.099,2.836l-0.471-
4.412l3.452-1.42l3.453-0.314l2.667,5.676l2.04,0.314l6.277,3.785l0.627-
0.473h5.336l5.493,2.365l3.61,1.104l2.825-1.104l1.726-1.578v1.578l-
1.255,3.15l3.923,2.525l4.708,2.521l2.511,2.836l3.295,2.365l2.511-
0.629l0.471,0.316l-
```

2.04,1.26v2.207l1.57,1.105l1.57,2.836l0.627,3.625l2.04,0.473l3.453,5.834l1.41
2,1.26v-1.104l-0.314-5.52l0.784,0.475l0.628,3.625l0.942,3.625l1.098-
0.475l0.785-3.311h0.784v4.887l1.413,1.42l-
1.727,1.895c0,0,0.628,1.104,1.255,1.418c0.628,0.314,0.942-0.945,0.942-
0.945l0.941,2.207c0,0,1.569,2.68,2.04,3.311c0.471,0.631,1.255,0.945,1.255,0.9
45l0.785-0.787l-1.255-1.734l-0.471-1.576l2.04,1.258l1.883,1.896l1.098-0.473l-
0.156-3.627l-1.412-2.68l0.784-
0.789l1.727,1.104l2.667,1.42l1.413,2.996l1.57,0.787l1.726-2.207l0.941-3.311l-
0.471-2.996L205.248,575.928z
M65.27,373.168c0,0,0,1.738,0,2.133s0.706,1.809,0.706,1.809l1.255,1.184l1.413-
0.232l-1.177-1.895l-1.727-3.074L65.27,373.168z M224.08,523.11l0.001-0.002h-
0.001V523.11z M49.42,348.336l4.394,0.236l0.863-0.631l-2.982-1.182l-3.452-
2.051l-1.177,0.631l0.784,1.42L49.42,348.336z M98.54,552.75l-2.667,1.893l-
4.551,3.469l-4.709,2.838l-1.256,1.104l1.099,5.676l2.825,0.789l4.865-
2.838l4.237-2.523l-1.57-2.365l0.472-2.365l-2.04,1.262l-2.826-0.789l1.57-
1.104l1.884,0.789l3.766-1.734l0.471-1.42l-2.355-0.787L98.54,552.75z
M69.665,367.887l1.49,0.553l0.549-1.416l-0.941-1.738l-2.589-1.971l-
1.021,0.162v1.178l1.727,2.287L69.665,367.887z M81.433,33.319l0.706-
0.63l0.314-1.419l1.647,0.552c0,0-1.412,2.208-
1.412,2.522c0,0.316,0.864,0.237,0.864,0.237l1.334-0.395l0.706-1.971l-0.628-
0.788l1.49-1.656l-0.784-1.498h-2.276l-1.49,1.576l-0.471-1.34l-
1.961,0.157l0.471,2.444L81.433,33.319z M76.882,574.194l1.412,1.734l0.943-
1.104l-0.785-1.893L76.882,574.194z M646.686,130.442l3.138,0.157l1.256-
1.183c0,0-0.08-1.418-0.394-1.576c-0.313-0.157-1.571-1.813-1.571-1.813l-
2.117,0.236l-1.57,0.158l-0.313,1.104l0.946,0.473 L646.686,130.442z
M16.622,573.247l0.314,1.578l-1.883,1.891l-1.883-0.473l-0.471-1.576l-2.511-
0.316l-3.295,0.947l-3.295,2.207l1.096,0.631l1.57-0.316l4.237-
1.418l3.61,1.104l2.667,0.473l4.238-1.104l1.098-1.734l-2.825-
2.523L16.622,573.247z M565.953,96.887l3.219-1.895l4.994-5.017l-0.443-0.447l-
2.219,0.557l-5.215,3.791l-2.107,0.781l-
1.775,2.007l1.551,1.115L565.953,96.887z M53.971,576.084h2.354l0.784-1.891l-
2.04-0.316L53.971,576.084z M19.132,510.338l-1.097-1.576l-1.099-2.523h-
0.941v1.893l2.04,2.996L19.132,510.338z M53.344,572.143l-
2.198,0.947l1.57,1.891l2.04-1.576L53.344,572.143z M36.865,486.688l-1.412-
1.418l-1.099-2.365l-
1.726,2.365l0.156,2.521h2.355l1.726,1.42l1.726,3.941l2.826,0.314l0.157-
0.945l-2.039-5.203L36.865,486.688z M864.562,184.404l-
2.22,2.564l1.111,0.892l2.105-1.003l2.33-1.003l-1.775-1.45H864.562z
M873.992,183.848l-0.443,1.672l-2.107,0.668l-0.443,0.557l3.773-0.111l1.221-
0.446l-1.002-2.23L873.992,183.848z M814.598,322.006l-
0.707,0.236l0.629,1.812l1.412,2.287l0.627,5.992l-0.156,1.971l-1.492,0.631l-
3.059,2.521l-2.512,2.443l1.02,0.158l1.098-1.182l2.746-2.365l3.297-
1.498l0.158-3.311l-0.393-4.336L814.598,322.006z M875.171,119.957l-
1.492,1.498l1.728,2.759l1.648-0.867l-0.55-2.365L875.171,119.957z
M895.885,94.103L894,92.054l-2.669,0.16l0.471,1.104l-0.158,0.473h-0.784l-
2.824-2.207l-0.158-4.73l-0.785-1.892h-5.807l-8.788-29.168l-1.884-0.946l-
5.808-2.365l-1.412-0.158l-1.727,1.735l-3.922,2.837v0.945l-
0.785,0.788L854.292,58l-1.255-2.207v-1.104l-1.255-0.157h-1.255l-2.04,4.099l-
2.824,8.672l-1.728,4.73l0.159,4.73l0.157,1.419l-0.785,2.681l-
0.787,0.945v5.833l1.888,2.365l-1.414,3.941l-2.511,4.572l-0.786,5.361v2.522l-
1.468-0.399l-2.371,0.336l0.072,0.224l-0.778-0.004l-0.628,1.576l-0.629-0.474l-
0.939-0.945l-1.412,1.892l-0.929,5.35l-0.132,0l-26.244,6.875l0.004,0.023l-
7.07,1.945l-4.396,1.418l-3.138-0.157l-5.336,1.261l-2.991,1.215l-2.346,1.78l-

2.037,2.365l-2.668,3.625l-2.826,4.415l-1.412,2.837l-0.627,0.788l-
5.649,5.361l0.159,2.522l0.785,1.104l1.726,0.788h1.727v1.419l-
1.098,2.049l0.312,1.418l1.414,2.05l-0.16,2.207l-1.883,1.104h-2.039l-
1.571,1.892l-1.726,3.153l-2.039,1.734l-5.021,0.473l-2.513,1.104l-
2.039,1.261l-1.571-0.157l-1.884-1.261l-6.118,0.157l-3.14,0.473l-3.924,1.262l-
4.236,1.418l-2.825,1.735l1.883,5.991l1.727,0.157l1.256,2.996l0.156,0.946l-
1.098,0.945l-2.041,2.68l-0.472,1.576l-1.883,1.735l-1.727,1.104l-0.941,1.576l-
1.252,1.104l-2.6,1.762l0.009,0.083l-5.09,5.247l-1.255,0.473l-4.059,2.918l-
0.028-0.199l-6.906,4.063l-3.767,2.207l-3.297,3.626l-3.924,3.784l-
3.137,0.788l-2.824,0.473l-5.336,2.523l-2.039,0.157l-3.297-2.996l-
5.021,0.631l-2.512-1.419l-2.194-1.251l-1.096-0.952l0.157-1.419l2.04-
3.626l1.94-1.695l-0.222-4.929l1.551-1.558l1.06-0.334l0.217-3.468l1.49-
2.957l1.02,0.591l0.16,0.63l0.783,0.158l1.883-0.947l-0.315-9.301l-3.139-
8.042l-2.197-8.829l-2.354-3.153l-2.512-1.734l-1.568,1.104l-3.768,1.735l-
1.884,4.888l-2.668,3.626l-1.098,0.631l-1.412-0.631c0,0-2.512-1.418-2.352-
2.049c0.155-0.63,0.471-4.888,0.471-4.888l3.295-1.261l0.787-3.311l0.625-
2.522l2.354-1.576l-0.312-9.776l-1.57-2.208l-1.256-0.788l-0.783-2.05l0.783-
0.788l1.572,0.316l0.156-1.577l-2.354-2.208l-1.254-2.523h-2.51l-4.396-1.418l-
5.336-3.311h-2.667l-0.627,0.631l-0.941-0.472l-2.981-2.208l-2.824,1.734l-
2.824,2.207l0.312,3.469l0.943,0.316l2.036,0.473l0.474,0.788l-2.51,0.788l-
2.515,0.315l-1.412,1.734l-0.312,2.049l0.312,1.576l0.314,5.361l-3.451,2.049l-
0.627-0.157V153.7l1.253-2.365l0.629-2.365l-0.785-0.788l-1.884,0.788l-
0.939,4.099l-2.668,1.104l-1.727,1.892l-0.157,0.946l0.628,0.788l-0.628,2.523l-
2.196,0.473v1.104l0.785,2.365l-1.101,5.991l-
1.569,3.942l0.627,4.572l0.472,1.104l-0.785,2.365l-0.312,0.788l-
0.313,2.681l3.452,5.833l2.824,6.307l1.412,4.73l-0.786,4.572l-0.94,5.833l-
2.353,5.045l-0.314,2.68l-2.992,3.034l-0.706,0.735l-0.462,0.015l-2.744,1.577l-
2.039,1.892l-1.1,0.315l-1.412-0.945l-2.926-0.852l-1.156-0.726l-0.785-2.523l-
1.256-3.626l-1.567-1.734l-1.412-2.523l-0.228-5.353l-0.065,0.009l0.136-3.958l-
1.571-4.415l-0.627-5.991l-1.1-2.365l0.94-2.996l0.785-2.838l1.412-2.522l-
0.627-3.311l-0.629-3.469l0.475-1.735l1.883-2.365l0.158-2.68l-0.787-
1.261l0.63-2.523l0.471-3.153l2.668-5.519l2.824-6.623l0.155-2.208l-0.312-
0.946l-0.784,0.474l-4.078,6.149l-2.668,3.941l-1.883,1.735l-0.785,2.208l-
1.412,0.788l-1.103,1.892l-1.412-0.316l-0.158-1.734l1.256-2.365l2.037-
4.572l1.728-1.576l0.985-2.06l0.087,0.061l3.64-7.934l0.942-3.941l1.883-
4.257l0.783-0.157l1.101,1.576h0.627l4.395-
2.365l1.412,1.576l0.473,0.157l1.255-1.104l1.098-2.996l2.354-0.788l6.748-
0.631l1.886-2.521l5.022-0.158l5.647,1.262h1.726l3.139-
1.419l2.197,0.157l2.039-0.63l3.609,0.473l0.787,0.316l1.254-0.316l-1.254-
0.945l-1.257-0.631l-3.139-2.996v-6.78l-1.412-0.472l-1.098,1.104l-
5.965,1.577l-1.884,0.473l-2.823-0.788l-0.472-0.316v-5.519l-1.411-0.157l-
2.513,1.261l-4.395,1.892l-6.437,0.316l-3.295,1.104l-3.922,3.469l-
1.572,0.946h-1.098l-1.256,0.788l-1.57-0.473l-1.568-1.262l-1.411,0.946l-
3.767,0.157l-2.668-2.68l-1.414-2.996l-1.412-1.104l-3.138-0.946h-2.196l-1.255-
1.262l-3.453,2.838l-0.94,1.104l-0.785-0.472l0.314-2.522l2.354-3.153l0.471-
2.365l2.197-0.788l1.412-2.996l3.609-0.945l0.313-0.947l-1.098-1.104l-
4.553,0.473l-4.236,2.365l-2.197,2.208l-1.254,1.735l-1.727,0.788l-
1.886,2.838l-0.157,1.261l-4.236,2.049l-2.354,1.892l-5.807,0.947l-
0.629,0.63v0.947l-3.451,2.207l-2.667,0.788l-1.797,1.009l0.064,0.082l-
0.465,1.117l-1.098-0.158l-0.629-1.104l-2.668-0.788l-1.099,0.157l-
1.727,0.946l-0.939-0.63l0.627-1.892l1.886-2.996l1.098-1.104l-1.883-1.419l-
2.041,0.788l-2.824,1.892l-7.221,3.153l-2.822,0.63l-2.824-0.473l-0.538-
0.483l0.263-0.252l-0.668-0.53l3.295-2.996l1.256-0.157l4.393-4.888l1.728-

0.788l2.196-3.784l2.354-3.469l2.981-2.522l4.637-1.958l8.975-3.961l3.825-
1.792l0.593-2.223l-4.311,0.362l-0.69,1.058h-0.629l-1.727-2.996l-8.632,0.315l-
0.94,0.788h-0.939l-0.473-1.261l-0.785-1.734l-2.51,0.473l-3.139,3.153l-
1.568,0.788h-2.982l-2.512-0.946v-2.05l-1.254-0.157l-0.473,0.472l-2.513-
1.261l-0.471-2.837l-1.412,0.473l-0.471,0.945l-2.354-0.473l-5.18-2.365l-3.764-
2.522h-2.824l-1.256-0.946l-2.197,0.631l-1.098,1.104l-0.314,1.261h-4.707v-
2.049l-6.121-0.315l-0.313-1.419h-4.709l-1.57-1.576l-1.412-5.991l-0.783-
5.361l-1.886-0.788l-2.196-0.473l-0.628,0.158l-0.312,8.041l-29.252-0.034l-
0.128-0.046l-28.524-0.55l-18.048-0.631l-25.735-1.261l-24.133-1.821l-0.084-
0.016l-31.021-3.208l-28.404-3.467l-28.403-3.941l-31.385-5.203l-17.89-3.311l-
31.765-6.762l-13.499-2.583l0.03-0.118l-15.652-3.624l-22.284-5.519l-19.459-
5.519l-8.318-1.892l-9.416-2.68l-4.237-1.418l-1.099,0.787l-
0.314,0.788l2.354,4.73l1.099,2.522l-1.726,3.469v2.208l0.627,1.419l-
0.784,1.733l0.471,3.153l0.942,1.262l-0.158,1.261l-1.412,0.157l-0.628-1.892l-
1.098-2.365L85.2,40.654l0.314-2.049l2.197-0.474l-0.313-1.734l-0.314-1.104l-
2.04,1.261l-1.256,1.104v2.365l-2.197,0.158l-3.295-0.947l-2.825-1.418l-2.981-
0.631l-4.394-2.049l-3.14-1.891l-2.667-2.522l-2.354-2.838l-2.04-0.473l-
2.196,9.144l1.727,3.153v7.726l-0.628,2.837l1.416,6.939l2.667,2.68l-
4.237,0.63l-0.157,3.625l2.511,1.104l-1.57,3.941l-2.667,0.316l-
0.313,2.837l2.196,2.837l1.727-0.945l1.863,1.459h-0.003L54.6,77.704l-
2.197,6.307l-3.138,9.775l-3.139,6.307l-4.865,13.717l-6.277,13.244l-
7.846,12.298l-1.883,2.837l-0.784,8.356l-1.255,5.833l2.161,4.377l-
1.535,4.607l-0.157,7.253l-5.021,11.509l-2.982,2.522l-0.313,1.104l-
1.727,0.788l-1.412,4.099l-0.785,3.153l2.667,4.099l1.57,4.1l1.099,3.469l-
0.314,6.307l-1.726,3l-0.628,5.676l-
0.941,3.626l1.727,3.784l2.667,4.415l2.196,4.73l1.255,3.941l-0.313,3.153l-
0.314,0.472v2.049l5.493,6.149l-0.471,2.365l-0.628,2.207l-
0.627,1.892l0.157,8.042l2.04,3.626l1.883,2.522l2.667,0.473l0.941,2.68l-
1.099,3.469l-2.04,1.576h-1.098l-
0.784,3.784l0.471,2.838l3.138,4.257l1.569,5.203l1.413,4.571l1.255,2.995l3.295
,5.676l1.413,2.523l0.471,2.838l1.57,0.945v2.365l-0.784,1.895l-1.727,6.934l-
0.471,1.896l2.354,2.68l4.08,0.475l4.394,1.734l3.766,2.049h2.825l2.825,2.994l2
.511,4.73l1.099,2.207l3.766,2.049l4.708,0.789l1.412,2.047l0.628,3.154l-
1.413,0.631l0.314,0.945l3.139,0.789l2.667,0.154l2.825,4.572l3.766,4.1l0.785,2
.207l2.511,4.102l0.313,3.152v9.146l0.471,1.734l9.73,1.418l19.145,2.68l14.121,
1.58l0.353,0.105l-2.55,2.105l-
0.314,1.418l0.471,0.947l18.36,10.404l11.77,7.414l14.281,8.355l16.32,9.777l11.
926,2.363l25.95,3.418v-0.002l15.007,1.939l1.255-
9.773l16.164,2.521l13.182,1.734l-0.022-
0.133l1.761,0.443l2.511,2.838l1.412,4.256l4.552,2.207l1.255,3.154l6.905,7.725
l1.255,1.578l4.865,2.049l1.098,2.049l1.569,0.947l0.471,2.68l3.139,6.307v7.883
l2.197,4.572l7.218,7.564l5.021,2.051l1.727,1.891v0.631l3.766,2.207l1.883,0.63
1l1.727,1.105l2.511,0.947l2.354-2.363l4.237-5.99l0.942-3.625l2.197-
3.154l3.453-1.418l4.394-
1.734l2.982,2.207l7.218,0.631l6.592,1.104l2.511,2.049v1.105l2.511,2.996l5.806
,5.203l0.158,1.416l1.727,1.895l0.784,4.1l5.178,11.982l-
0.158,1.893l4.08,2.521l3.453,6.465l3.295,4.26l3.14,1.258l1.569,2.209l-
1.256,4.256l0.628,0.949l1.255,0.627l-0.312,3.314l-
0.628,0.629l0.628,2.207l3.138,1.893l1.254,6.307l2.04,3.783l7.376,3.311l5.021,
1.107l4.079,2.996l3.139,0.627l1.256-0.469l5.335,1.104l5.492,3.783l2.984-
1.893l0.939-1.418l-1.727-2.684l-0.94-5.988l-1.727-6.779l-0.784-2.365l0.784-
4.414l1.099-3.785l1.255-4.414l1.412-5.357l-2.198-1.895l0.941-1.893l3.924-
0.316l3.766-5.359l3.297-0.631l5.648-3.469l1.727-1.42l6.119-3.311l5.493-

2.365l5.177-3.152l2.668-2.051l5.491-5.201l1.256-0.789l2.038-1.418l2.514-
1.893l0.938-1.893l9.572-4.414l6.277-1.703l-0.002-0.02l4.39-1.43l7.846-
0.316l10.045,3.471l6.275,1.104l3.609-1.42l3.14,1.105l3.138,0.947l0.784-
2.051l-3.136-1.105l-2.513,0.475l-2.668-1.576c0,0,0.157-1.264,0.785-
1.418c0.627-0.158,2.982-0.947,2.982-0.947l1.727,1.418l1.727-
0.947l3.138,0.631l1.412,2.365l0.313,2.207l4.396,0.316l1.726,1.732l-
0.784,1.576l-1.255,0.789l1.568,1.576l8.16,3.471l3.453-1.262l0.94-2.365l2.51-
0.631l1.727-1.416l1.255,0.945l0.785,2.836l-2.197,0.789l0.627,0.631l3.297-
1.26l2.195-3.312l0.787-0.473l-2.039-0.316l0.783-1.576l-0.156-1.418l2.04-
0.473l1.098-1.262l0.626,0.789c0,0-
0.155,2.996,0.629,2.996c0.783,0,4.08,0.631,4.08,0.631l3.922,1.891l0.942,1.42h
2.824l1.098,0.945l2.198-2.994v-1.418h-1.255l-3.295-2.684l-5.648-0.787l-3.139-
2.205l1.098-2.684l2.197,0.318l0.157-0.631l-1.728-0.945v-0.475h3.14l1.726-
2.994l-1.255-1.893l-0.312-2.684l-1.411,0.158l-1.886,2.049l-0.627,2.521l-
2.981-0.629l-0.941-1.736l1.728-1.891l1.962-1.734l0.695-0.58l0.033,0.074l1.07-
0.76l2.51-4.1l1.727,0.789l6.592-1.893l2.039,0.314l1.412,0.787h5.022l1.129-
1.225l0.125-0.189h2.67l0.626,0.16l1.256-2.684l1.412-
4.258l2.197,0.631l2.984,5.834v0.943l-2.668,1.891l2.668,0.316l6.123-
2.412l3.287-0.582l5.809-2.047l5.807-0.475l4.236-
0.633l7.377,1.734l7.846,3.785l1.566,1.418l2.826,1.104l1.57,1.895l0.312,2.68l3
.138-1.256h3.767l3.453-1.896l3.608-3.469l2.98,0.158l0.473-1.104l-0.787-
0.947l0.16-1.893l3.922-
0.789h2.512l2.824,1.422l4.078,1.418l2.354,3.623l2.668,0.943l1.1,3.314l3.295,1
.578l1.568,2.521l1.887,0.631l5.021,1.258l1.256,2.996l2.979,3.625v9.303l-
1.412,4.572l0.314,2.68l1.252,4.73l1.727,3.941l0.785-0.475l1.412-4.414l-2.51-
0.945l-0.314-0.629l1.57-0.633l4.395,0.949l0.158,1.576l-3.139,5.357l-
2.039,2.365l3.453,3.625l2.51,2.996l2.824,5.203l2.826,3.783l2.039,4.889l1.727,
0.314l1.568-2.049l1.727,1.105l2.512,3.939l0.627,3.473l2.982,4.258l0.783-
1.262l3.768,0.312l3.451,2.207l3.297,5.049l0.783,3.312l0.314,2.834l1.098,0.945
l1.256,0.473l2.353-0.945l1.413-1.574l3.767-0.158l2.979-1.418l2.668-3.154l-
0.468-1.896l-0.312-2.363l0.627-1.893l-0.314-1.893l2.354-1.262l0.312-3.312l-
0.626-1.732l-0.472-11.666l-1.254-7.412l-4.396-8.041l-3.449-5.676l-2.513-
5.203l-2.824-2.838l-2.823-7.254l0.783-1.262l1.101-1.262l-1.572-2.834l-3.922-
3.625l-4.709-5.361l-3.609-6.146l-5.178-9.146l-3.621-9.502l-2.029-8l-0.19-
0.012l-0.755-6.137l2.199-9.775l1.411-4.1l-0.473-2.521l3.924-6.779l-0.582-
1.578l-1.616,0.863l1.575-0.863l2.038-1.256l4.865-5.359l-0.94-3.154l2.824-
0.156l3.452-3.312l1.57-0.787l2.197-3.311l2.666-2.682l2.039-3.473l2.354-
0.631l1.1-2.682l1.568-0.787l0.473-6.148l2.513-5.988l5.244-5.307l5.578-
2.109l4.552-0.475l0.471-2.363l1.885-6.307l3.295-4.572l6.277-5.203l5.021-
2.365l2.666-0.631l1.103,0.473h1.252l2.982-4.73l1.885-3.469l-1.257,0.475l-
2.196,2.207l-0.627-1.578h-4.079l1.883-6.146l-0.783-1.264h-1.885v-0.943l-
0.312-1.264l1.568,1.264l1.412,0.156l2.354,0.314l3.609-1.576l1.256-
2.838l0.627-2.047l2.51-1.26l0.314-4.1l-0.785-0.631l2.354-0.158l-0.627-2.207l-
2.354-2.365l-3.451-6.307l-1.659-4.582l-0.034-0.07l-3.955-6.854l-1.101,0.157l-
3.139,0.632l-2.51,0.471l-0.314-0.471l2.197-1.42l-0.159-2.207l-1.253-1.577l-
1.412-1.104v-1.104l-1.885-1.576v-0.945l3.137,1.104l0.16-1.734l-1.256-2.049l-
0.156-2.208l-0.942-0.945l0.475-4.415l-1.256-1.261l-3.139-0.631l-2.513-2.208l-
5.021-0.315l-1.83-1.517l-0.005-0.003l4.186-
0.061l2.982,0.947l2.041,0.314l0.938-2.049l-1.412-2.049v-1.734l-2.354-2.05l-
2.039-5.361l1.255-5.203l-0.157-2.05l-1.255-1.261c0,0,1.414-1.576,1.414-
2.207s0.471-2.05,0.471-2.05l1.884-1.261l1.885-1.576l0.471,0.946l-
1.412,1.576l-1.256,3.626l0.314,1.104l1.727,0.315l0.471,5.361l-
2.039,0.945l0.314,3.469l0.471-0.157l1.099-1.892l1.571,1.734l-1.571,1.262l-

0.312,3.311l2.511,3.311l3.766,0.474l1.57-
0.788l3.139,5.045l1.727,0.474v3.468l-2.196,4.73l-
0.472,6.78l1.412,3.311l1.412,0.157l1.883-4.099l0.785-3.469l0.158-6.937l2.981-
4.73l2.039-6.78v-5.517h-0.009l-0.02,0.004l-2.178-7.099l-0.94,0.473l-3.449-
2.365l-1.727-4.572l-1.884-3.469l-2.198-0.945l-2.039-3.469l0.783-2.048l0.771-
2.008l-
0.582,2.502l1.062,2.189l3.139,2.837l4.709,2.208l3.924,0.63l0.156,1.42l-
0.783,0.945l0.312,2.68h0.785l2.039-2.365l0.785-4.73l2.666-3.941l2.984-
6.307l1.098-5.36l-0.627-1.104l-0.153-9.147l-1.569-3.311l-1.101,0.788l-
2.666,0.316l-0.473-0.473l1.098-0.946l2.039-1.892l0.062-1.067l0.198-
0.084l1.469,0.52l3.924-1.104l5.648-1.892l2.512-0.945l6.901-5.202l3.767-
2.837l3.297-3.469l-4.08-1.576l-1.254,1.418l-2.824,2.681l-7.688,3.784l-2.197-
0.157l-1.571-0.631l-1.099,0.631l-2.197,2.522l-1.412,1.261l-1.254,0.316l-
0.312-1.261l1.883-1.735l0.202-1.424l0.009,0.01l4.811-3.317l3.451-3.153l1.885-
2.049l0.783,0.631l2.668-1.42l5.021-1.104l8.159-3.626l-0.032-0.146l2.392-
0.8l2.353-1.892l1.098-1.734l1.256,0.158l2.826-1.499l-0.003-0.005l2.514-
1.336l-0.472-2.207l0.784-1.418l2.825-1.42l0.783,2.996l-0.471,1.735l-
2.354,1.418v0.947l1.884-1.419l3.768-4.415l3.765-1.892l4.08-1.418l-0.313-
2.365l-0.939-2.837l-1.885-2.365l-1.727-0.788l-2.037,0.157l-
0.474,0.473l0.941,1.261l1.412-0.788l2.039,1.576l0.784,2.681l-1.727,1.733l-
2.197,0.947l-3.451-0.473l-3.764-5.833l-2.199-2.522h-1.727l-1.1,0.788l-1.884-
2.523l0.314-1.418l2.354-5.045l-2.807-4.304l0.002-0.044l0.295-1.493l1.057-
3.21l-0.03-0.011l1.48-7.183l0.156-4.257l1.414-1.261l-0.941-3.153l2.982-
3.469l1.57,1.419l1.256-0.474l5.178-2.996l0.312-2.837l2.825-0.474l1.57-2.522l-
0.158-1.576l-0.313-2.996l0.787-1.261l-0.314-1.104l-1.256-1.418l2.041-
1.261l1.727,3.153l1.412-0.316l0.314,1.261v1.577l1.883,0.157l0.471-2.68l0.474-
0.631l-1.572-1.42l0.94-1.418l2.354-1.419l0.94-1.418l1.726-
0.158c0,0,0.313,2.049,0.942,2.049c0.626,0,1.254,0,1.254,0l4.71-5.361l3.296-
2.996l1.883-0.63l2.041-4.572v-1.892L895.885,94.103z M795.926,273.918l-
1.807,0.474l1.098,2.444l0.156,1.734l1.1,1.813c0,0,0.865-0.867,0.865-1.183c0-
0.316-0.707-2.996-0.707-2.996L795.926,273.918z M881.525,113.494l-
1.332,2.207l1.725,1.813c0,0,1.254,0.079,1.254-0.236s0.234-1.971,0.234-
1.971l0.865-0.788l-0.785-1.735L881.525,113.494z M770.892,558.27l-
1.413,4.492l-3.138,5.125l-4.237,4.1l-3.294,1.891l-
2.434,1.498l1.178,1.025l2.668-2.049l5.175-4.1l3.612-3.785l2.433-6.463l0.94-
1.656l0.157-3.311l-0.707,0.473 L770.892,558.27z M445.979,529.1l-3.373,5.283l-
3.217,6.699l-1.492,8.041l0.548,6.859l2.668,7.016l0.548,6.938l0.869-0.236l-
0.158-1.418l-0.471-5.047l-2.511-7.018l-0.865-5.988l0.313-5.051l1.962-
6.463l4.236-6.387l4.63-6.463l-0.628-0.238L445.979,529.1z
M288.577,541.713v1.262l1.883,3.785l2.982-1.105l0.314-1.576l-1.569-
2.049L288.577,541.713z M751.825,576.006l-1.255-0.473l-2.277,1.576l-
1.412,2.287l-1.255,0.236l-2.354,0.631l0.863,0.867l2.277-1.184l4.08-
1.418l3.374-1.184l-0.393-1.023 L751.825,576.006z M292.972,539.823l2.511-
1.42l0.314-1.262l-5.964-0.631l-0.627,0.475l-4.552-0.947l-
1.255,1.893l5.336,1.576 L292.972,539.823z M238.833,514.122l-3.923,1.734l-
0.943,3.627l5.964,2.523l2.039-0.316l1.57-3.783l-0.628-3.311L238.833,514.122z
M272.099,528.155l-2.039-3.469l-5.65,2.838l-0.627,1.734l3.61,5.361l2.354-
0.316l1.098-0.473l1.412,1.262l3.609-0.158l0.942-1.418l-2.826-
1.734L272.099,528.155z M225.963,519.641l-1.882,3.467h2.353l2.04-2.996l-0.314-
0.789L225.963,519.641z M328.595,565.206l-1.57-2.363l-3.295-2.051l-7.376-
3.469l-3.295-1.576l-1.57,1.262l0.315,2.049l1.569,1.734l-0.471,0.945l-
2.04,2.996l-2.196,0.947l0.156,2.521l2.04,1.734l1.57,5.518l-
0.628,2.365l0.314,4.416l3.452,2.521l1.727,0.158l2.197-2.838l2.354-

```
3.154l2.982-1.734l1.412,0.631l4.079-1.418l3.766-3.154l0.314-1.26l-3.923-
2.523L328.595,565.206z M310.547,545.186l-4.237-2.68l-4.238-1.107l-4.55-
2.365l-2.197,2.838l4.708,6.469l2.354,1.418l4.708-0.945l3.452-
1.734V545.186z").attr(attrOut).scale(scaleNum,scaleNum, 0, 0);
            var us = {};
            us.HI = R.path("M224.08,523.111l1.883-3.469l2.196-
0.318l0.314,0.789l-2.04,2.996h-2.354V523.11z
M233.967,519.483l5.964,2.523l2.04-0.316l1.57-3.783l-0.628-3.311l-4.08-0.475l-
3.923,1.734L233.967,519.483L233.967,519.483z
M263.783,529.258l3.61,5.361l2.354-0.316l1.098-0.473l1.412,1.262l3.609-
0.158l0.941-1.418l-2.825-1.734l-1.883-3.627l-2.04-3.469l-
5.649,2.838L263.783,529.258L263.783,529.258z M283.399,537.93l1.255-
1.893l4.552,0.947l0.627-0.475l5.964,0.631l-0.314,1.262l-2.511,1.42l-4.237-
0.316L283.399,537.93z M288.577,542.975l1.883,3.785l2.982-1.105l0.314-1.576l-
1.57-2.049l-3.609-0.316V542.975L288.577,542.975z M295.325,541.872l2.197-
2.838l4.55,2.365l4.237,1.109l4.237,2.68v1.895l-3.452,1.734l-4.708,0.945l-
2.354-1.418L295.325,541.872L295.325,541.872z M311.489,557.008l1.569-
1.262l3.295,1.576l7.376,3.469l3.295,2.051l1.57,2.363l1.883,4.258l3.923,2.523l
-0.313,1.26l-3.766,3.154l-4.08,1.418l-1.412-0.631l-2.982,1.734l-2.354,3.154l-
2.197,2.838l-1.727-0.158l-3.453-2.521l-0.313-4.416l0.628-2.365l-1.57-5.518l-
2.04-1.734l-0.156-2.521l2.196-0.947l2.04-2.996l0.471-0.945l-1.569-
1.734L311.489,557.008L311.489,557.008z").attr(attr).scale(scaleNum,scaleNum,
0, 0);
            us.AK = R.path("M151.266,459.096l-
0.313,83.248l1.569,0.947l2.982,0.156l1.412-
1.104h2.511l0.157,2.838l6.748,6.623l0.471,2.521l3.295-1.891l0.628-
0.158l0.314-2.996l1.412-1.576l1.098-0.158l1.883-
1.418l2.981,2.049l0.628,2.838l1.883,1.104l1.098,2.365l3.766,1.734l3.295,5.834
l2.667,3.783l2.197,2.682l1.412,3.625l4.865,1.734l5.021,2.051l0.941,4.256l0.47
1,2.996l-0.941,3.311l-1.726,2.207l-1.57-0.787l-1.413-2.996l-2.667-1.42l-
1.727-1.104l-0.784,0.789l1.412,2.68l0.156,3.627l-1.098,0.473l-1.883-1.896l-
2.04-1.258l0.471,1.576l1.255,1.734l-0.785,0.787c0,0-0.784-0.314-1.255-0.945c-
0.471-0.631-2.04-3.311-2.04-3.311l-0.941-2.207c0,0-0.314,1.258-0.942,0.945c-
0.627-0.314-1.255-1.418-1.255-1.418l1.727-1.893l-1.413-1.42v-4.887h-0.784l-
0.785,3.311l-1.098,0.473l-0.942-3.625l-0.628-3.625l-0.784-
0.473l0.314,5.518v1.104l-1.412-1.258l-3.453-5.836l-2.04-0.471l-0.627-3.625l-
1.57-2.834l-1.57-1.107v-2.205l2.04-1.26l-0.471-0.316l-2.511,0.627l-3.295-
2.365l-2.511-2.834l-4.708-2.521l-3.923-2.527l1.255-3.152v-1.578l-
1.726,1.578l-2.825,1.107l-3.61-1.107l-5.493-2.365h-5.336l-0.627,0.473l-6.277-
3.785l-2.04-0.314l-2.667-5.676l-3.453,0.314l-3.452,1.42l0.471,4.41l1.099-
2.834l0.941,0.316l-1.413,4.26l3.139-2.684l0.627,1.576l-3.766,4.262l-1.255-
0.318l-0.471-1.893l-1.255-0.787l-1.255,1.104l-2.667-1.734l-2.982,2.049l-
1.727,2.047l-3.295,2.049l-4.552-0.16l-0.471-2.047l3.61-0.631v-1.258l-2.197-
0.631l0.941-2.363l2.197-3.785v-1.734l0.157-0.787l4.237-
2.209l0.942,1.262h2.667l-1.255-2.527l-3.609-0.316l-4.865,2.682l-2.354,3.316l-
1.727,2.521l-1.098,2.207l-4.081,1.414l-2.982,2.529l-
0.314,1.576l2.198,0.945l0.784,2.047l-2.667,3.152l-6.277,4.104l-7.533,4.1l-
2.04,1.104l-5.178,1.105l-5.179,2.207l1.727,1.26l-1.413,1.418l-0.471,1.105l-
2.667-0.947l-3.138,0.158l-0.785,2.207h-0.942l0.314-2.365l-3.452,1.26l-
2.825,0.947l-3.295-1.26l-2.825,1.893h-3.139l-2.04,1.26l-1.569,0.789l-2.04-
0.316l-2.511-1.107l-2.197,0.631l-0.941,0.945l-1.57-1.104v-1.893l2.982-
1.26l6.121,0.631l4.237-1.578l2.04-2.049l2.825-0.631l1.727-
0.789l2.667,0.16l1.57,1.26l0.941-0.314l2.196-2.684l2.982-0.945l3.295-
```

```
0.631l1.255-0.314l0.628,0.469h0.784l1.255-3.623l3.923-1.418l1.883-
3.625l2.196-4.412l1.57-1.42l0.314-2.523l-1.57,1.26l-3.295,0.627l-0.627-
2.365l-1.255-0.314l-0.941,0.945l-0.156,2.838l-1.413-0.16l-1.412-5.676l-
1.255,1.262l-1.099-0.473l-0.314-1.893l-3.923,0.158l-2.04,1.104l-2.511-
0.314l1.413-1.42l0.471-2.521l-0.627-1.895l1.413-0.947l1.255-0.16l-0.628-
1.734v-4.258l-0.941-0.947l-0.784,1.418h-5.964l-1.412-1.26l-0.627-3.785l-2.04-
3.469v-0.949l2.04-0.789l0.157-2.047l1.098-1.105l-0.785-0.471l-1.255,0.471l-
1.098-2.684l0.941-4.887l4.394-3.152l2.511-1.58l1.883-3.623l2.667-
1.26l2.511,1.104l0.313,2.365l2.354-0.316l3.139-
2.365l1.57,0.631l0.941,0.631h1.57l2.196-1.262l0.785-4.256c0,0,0.314-
2.838,0.942-3.312c0.628-0.473,0.941-0.947,0.941-0.947l-1.099-1.893l-
2.511,0.787l-3.139,0.789l-1.883-0.471l-3.453-1.736l-4.865-0.158l-3.452-
3.625l0.471-3.783l0.627-2.365l-2.04-1.734l-1.883-3.625l0.471-0.787l6.59-
0.473h2.04l0.941,0.949h0.627l-0.157-1.576l3.767-
0.631l2.511,0.314l1.412,1.105l-1.412,2.049l-
0.471,1.418l2.667,1.578l4.865,1.732l1.726-0.945l-2.197-4.256l-0.941-
3.154l0.941-0.789l-3.295-1.895l-0.471-1.105l0.471-1.578l-0.784-3.781l-2.825-
4.572l-2.354-4.1l2.825-1.895h3.138l1.727,0.629l4.08-0.156l3.61-3.469l1.099-
2.996l3.609-2.365l1.57,0.945l2.667-0.631l3.61-2.047l1.099-
0.158l0.941,0.789l4.394-0.16l2.667-2.994h1.098l3.453,2.365l1.883,2.049l-
0.471,1.105l0.627,1.104l1.57-
1.576l3.766,0.316l0.314,3.625l1.883,1.418l6.905,0.631l6.12,4.1l1.412-
0.945l5.021,2.52l2.04-0.631l1.883-
0.787l4.708,1.891l4.237,2.84L151.266,459.096L151.266,459.096z
M39.535,487.319l2.04,5.203l-0.158,0.945l-2.825-0.314l-1.727-3.941l-1.726-
1.42h-2.354l-0.156-2.52l1.726-2.365l1.099,2.365l1.412,1.418L39.535,487.319z
M37.023,519.956l3.61,0.789l3.608,0.945l0.785,0.947l-1.57,3.625l-2.982-0.156l-
3.295-3.469L37.023,519.956z M16.936,506.239l1.099,2.523l1.098,1.576l-
1.098,0.789l-2.04-2.996v-1.893H16.936z M3.598,577.504l3.295-2.207l3.295-
0.947l2.511,0.316l0.471,1.576l1.883,0.473l1.883-1.891l-0.314-1.578l2.667-
0.631l2.825,2.523l-1.099,1.734l-4.237,1.104l-2.667-0.473l-3.609-1.104l-
4.237,1.418l-1.57,0.316L3.598,577.504z M51.146,573.09l1.57,1.891l2.04-1.576l-
1.412-1.262L51.146,573.09z M53.971,576.084l1.099-2.207l2.04,0.316l-
0.785,1.891H53.971L53.971,576.084z M76.882,574.194l1.412,1.734l0.942-1.104l-
0.785-1.893L76.882,574.194z M85.357,562.053l1.099,5.676l2.825,0.789l4.865-
2.838l4.237-2.523l-1.57-2.365l0.471-2.365l-2.04,1.262l-2.825-0.789l1.57-
1.104l1.883,0.789l3.766-1.734l0.471-1.42l-2.354-0.787l0.785-1.893l-
2.667,1.893l-4.551,3.469l-4.708,2.838L85.357,562.053z M126.471,542.661l2.354-
1.42l-0.941-1.734l-
1.727,0.947L126.471,542.661z").attr(attr).scale(scaleNum,scaleNum, 0, 0);
            us.FL =
R.path("M731.267,450.268l2.03,8l3.62,9.502l5.178,9.145l3.609,6.148l4.709,5.36
1l3.922,3.625l1.572,2.834l-1.1,1.262l-
0.784,1.262l2.824,7.252l2.822,2.834l2.515,5.203l3.447,5.676l4.397,8.041l1.253
,7.414l0.474,11.664l0.624,1.736l-0.313,3.311l-2.351,1.26l0.314,1.895l-
0.627,1.893l0.312,2.363l0.468,1.898L767.516,563l-2.98,1.418l-3.766,0.158l-
1.414,1.576l-2.35,0.945l-1.256-0.473l-1.098-0.945l-0.314-2.834l-0.783-3.314l-
3.297-5.047l-3.451-2.207l-3.768-0.314l-0.783,1.258l-2.982-4.258l-0.627-
3.473l-2.512-3.934l-1.727-1.107l-1.568,2.049l-1.727-0.318l-2.039-4.889l-
2.826-3.783l-2.824-5.199l-2.51-2.996l-3.453-3.625l2.039-2.365l3.139-5.357l-
0.16-1.576l-4.395-0.949l-1.572,0.631l0.314,0.629l2.51,0.947l-1.412,4.414l-
0.785,0.469l-1.727-3.941l-1.252-4.73l-0.314-2.68l1.412-4.572v-9.303l-2.982-
3.623l-1.254-2.996L706,485.426l-1.889-0.631l-1.568-2.521l-3.295-1.578l-1.101-
```

```
3.316l-2.667-0.945l-2.354-3.625l-4.078-1.416l-2.824-1.42h-2.512l-
3.922,0.789l-0.16,1.891l0.787,0.949l-0.473,1.104l-2.98-0.158l-3.609,3.469l-
3.453,1.896h-3.766l-3.138,1.256l-0.313-2.68l-1.57-1.895l-2.826-1.104l-1.566-
1.418l-7.848-3.785l-7.375-1.734l-4.236,0.631l-5.809,0.473l-5.809,2.047l-
3.375,0.598l-0.229-7.852l-2.512-1.893l-1.727-1.734l0.314-2.996l9.886-
1.26l24.795-2.838l6.589-0.625l5.963-
0.16l2.512,3.781l1.412,1.418l7.689,0.158l10.504-0.631l20.882-1.258l5.286-
0.656l4.443,0.027l0.156,2.838l2.514,0.787l0.311-4.26l-1.568-4.411l1.098-
1.576l5.65,0.789L731.267,450.268L731.267,450.268z M743.272,580.264l2.354-
0.631l1.255-0.236l1.412-2.287l2.278-
1.576l1.253,0.473l1.648,0.316l0.393,1.023l-3.374,1.184l-4.079,1.418l-
2.278,1.184L743.272,580.264z M756.376,575.375l1.178,1.025l2.668-2.049l5.176-
4.11l3.613-3.785l2.433-6.463l0.938-1.656l0.159-3.311l-0.707,0.473l-0.94,2.76l-
1.412,4.492l-3.139,5.125l-4.236,4.11l-
3.295,1.891L756.376,575.375z").attr(attr).scale(scaleNum,scaleNum, 0, 0);
            us.SC = R.path("M736.76,419.366l-1.727,0.945l-2.51-1.258l-0.629-
2.051l-1.254-3.469l-2.197-2.049l-2.512-0.631l-1.568-4.73l-2.668-5.834l-4.08-
1.893l-2.041-1.893l-1.254-2.523l-2.037-1.896l-2.201-1.258l-2.197-2.838l-2.98-
2.207l-4.398-1.734l-0.471-1.42l-2.353-2.836l-0.472-1.418l-3.295-5.047l-
3.301,0.158l-3.922-2.365l-1.253-1.26l-0.312-1.734l0.785-1.893l2.197-0.953l-
0.312-2.047l5.961-2.525l8.789-4.412l7.061-0.787l16.008-
0.475l2.195,1.896l1.572,3.154l4.236-0.475l12.24-
1.418l2.824,0.787l12.241,7.414l9.812,7.92l-5.262,5.324l-2.512,5.988l-
0.475,6.15l-1.568,0.787l-1.1,2.68l-2.354,0.631l-2.039,3.473l-2.667,2.682l-
2.197,3.311l-1.572,0.787l-3.452,3.316l-2.824,0.156l0.94,3.154l-
4.865,5.361L736.76,419.366L736.76,419.366z").attr(attr).scale(scaleNum,scaleN
um, 0, 0);
            us.GA = R.path("M667.242,365.76l-4.713,0.787l-8.16,1.104l-
8.316,0.867v2.129l0.156,2.051l0.629,3.311l3.295,7.725l2.354,9.617l1.412,5.992
l1.57,4.73l1.412,6.775l2.039,6.148l2.512,3.311l0.471,3.312l1.887,0.787l0.156,
2.051l-1.725,4.729l-0.473,3.154l-0.156,1.893l1.57,4.26l0.312,5.199l-
0.783,2.365l0.623,0.789l1.412,0.787l0.627,3.311l2.512,3.785l1.412,1.418l7.689
,0.158l10.504-0.631l20.882-1.258l5.286-
0.656l4.443,0.025l0.156,2.84l2.516,0.787l0.309-4.26l-1.568-4.412l1.098-
1.576l5.65,0.789l4.832,0.309l-0.758-6.143l2.2-9.775l1.412-4.11l-0.474-
2.52l3.924-6.779l-0.582-1.578l-1.772,0.945l-2.51-1.258l-0.628-2.051l-1.256-
3.469l-2.197-2.049l-2.512-0.631l-1.569-4.73l-2.667-5.834l-4.07-1.893l-2.041-
1.893l-1.254-2.523l-2.037-1.896l-2.201-1.258l-2.197-2.838l-2.98-2.207l-4.398-
1.734l-0.471-1.42l-2.353-2.836l-0.472-1.418l-3.295-5.047l-3.301,0.158l-3.922-
2.365l-1.253-1.262l-0.314-1.734l0.785-1.891l2.197-0.949l-0.158-2.227l-
1.725,0.492l-5.65,0.947l-
6.748,0.787L667.242,365.76L667.242,365.76z").attr(attr).scale(scaleNum,scaleN
um, 0, 0);
            us.AL = R.path("M605.097,471.87l-1.568-14.822l-2.668-18.289l0.16-
13.717l0.785-30.271l-0.16-16.24l0.16-6.26l7.529-0.361l26.99-2.53l19.874-
0.863l-
0.143,2.129l0.156,2.049l0.629,3.312l3.295,7.725l2.354,9.617l1.412,5.988l1.57,
4.73l1.412,6.779l2.039,6.148l2.512,3.311l0.471,3.312l1.887,0.787l0.156,2.051l
-1.721,4.729l-0.475,3.154l-0.158,1.893l1.57,4.256l0.312,5.203l-
0.783,2.365l0.623,0.789l1.412,0.787l0.789,3.469h-6.125l-6.588,0.631l-
24.795,2.838l-9.887,1.26l-0.312,2.996l1.725,1.734l2.512,1.893l0.564,7.74l-
6.371,2.51l-2.668-0.318l2.668-1.891v-0.938l-2.984-5.834l-2.197-0.631l-
```

```
1.412,4.258l-1.256,2.684l-0.624-
0.16H605.097L605.097,471.87z").attr(attr).scale(scaleNum,scaleNum, 0, 0);
            us.NC =
R.path("M805.559,307.729l1.66,4.582l3.449,6.307l2.354,2.365l0.627,2.207l-
2.354,0.158l0.785,0.631l-0.314,4.1l-2.51,1.26l-0.627,2.047l-1.254,2.838l-
3.609,1.576l-2.354-0.314L800,335.323l-1.568-
1.258l0.311,1.258v0.945h1.887l0.783,1.262l-
1.883,6.148h4.079l0.627,1.578l2.197-2.207l1.256-0.475l-1.886,3.473l-
2.983,4.73h-1.253l-1.103-0.473l-2.665,0.631l-5.021,2.365l-6.277,5.203l-
3.295,4.568l-1.887,6.307l-0.471,2.363l-4.551,0.475l-5.723,2.164l-9.654-
7.996l-12.239-7.412l-2.824-0.787l-12.237,1.418l-4.236,0.475l-1.572-3.154l-
2.195-1.896l-16.008,0.473l-7.061,0.789l-8.789,4.412l-5.961,2.523l-
1.57,0.316l-5.65,0.947l-6.748,0.787l-6.589,0.475l0.311-4.73l1.727-1.42l2.668-
0.631l0.625-3.623l4.08-2.684l3.767-1.418l4.079-3.469l4.236-2.049l0.629-
2.994l3.764-3.785l0.625-
0.156c0,0,0,1.104,0.787,1.104c0.786,0,1.884,0.314,1.884,0.314l2.198-
3.469l2.038-0.631l2.198,0.318l1.571-3.473l2.824-2.523l0.473-2.047v-
3.865l4.395,0.709l6.926-1.258l15.357-1.896l16.635-2.525l19.998-3.68l18.605-
3.73l11.143-2.838L805.559,307.729L805.559,307.729z M809.34,339.901l2.512-
2.443l3.057-2.52l1.494-0.631l0.156-1.971l-0.627-5.992l-1.412-2.287l-0.629-
1.812l0.707-0.236l2.668,5.361l0.393,4.336l-0.158,3.311l-3.299,1.498l-
2.746,2.365l-
1.098,1.182L809.34,339.901z").attr(attr).scale(scaleNum,scaleNum, 0, 0);
            us.TN = R.path("M674.461,329.338l-50.375,4.889l-15.297,1.734l-
4.486,0.502l-3.76-0.027v3.783l-8.16,0.471l-6.748,0.631l-10.77,0.053l-
0.258,5.686l-2.076,6.119l-0.965,2.941l-1.309,4.273l-0.314,2.523l-
3.924,2.207l1.412,3.473l-0.941,4.256l-1.49,1.656l7.924-0.08l23.383-
1.891l5.18-0.162l7.846-0.469l26.99-2.525l9.874-0.787l8.174-0.951l8.16-
1.104l4.712-0.787l0.31-4.73l1.727-1.42l2.668-0.631l0.624-3.623l4.08-
2.684l3.768-1.412l4.079-3.473l4.236-2.045l0.629-2.996l3.763-3.785l0.626-
0.156c0,0,0,1.104,0.786,1.104c0.788,0,1.886,0.316,1.886,0.316l2.197-
3.471l2.039-0.631l2.197,0.318l1.57-3.469l2.824-2.525l0.475-2.047l0.172-
3.836l-2.211,0.051l-2.354,1.893l-7.687,0.158l-
11.643,1.854L674.461,329.338L674.461,329.338z").attr(attr).scale(scaleNum,sca
leNum, 0, 0);
            us.RI = R.path("M846.293,192.013l-0.469-4.101l-0.787-4.257l-
1.645-5.755l5.566-1.499l1.572,1.104l3.295,4.257l2.824,4.337l-2.826,1.499l-
1.255-0.158l-1.099,1.734l-
2.35,1.892L846.293,192.013L846.293,192.013z").attr(attr).scale(scaleNum,scale
Num, 0, 0);
            us.CT = R.path("M846.763,192.013l-0.939-4.101l-0.783-4.257l-1.57-
5.833l-5.021,1.104l-21.186,4.652l0.629,3.232l1.412,7.096v7.883l-
1.104,2.207l1.783,2.058l4.811-3.317l3.451-3.153l1.885-
2.049l0.783,0.631l2.668-1.42l5.021-
1.104L846.763,192.013L846.763,192.013z").attr(attr).scale(scaleNum,scaleNum,
0, 0);
            us.MA = R.path("M871.441,186.188l2.105-0.669l0.443-
1.672l0.998,0.111l1.004,2.23l-1.221,0.446l-
3.773,0.111L871.441,186.188L871.441,186.188z M862.343,186.968l2.219-
2.564h1.553l1.775,1.45l-2.33,1.003l-
2.104,1.003L862.343,186.968L862.343,186.968z M828.562,165.524l16.949-
4.1l2.195-0.63l2.037-3.153l3.627-1.623l2.809,4.304l-2.354,5.045l-
0.314,1.418l1.883,2.523l1.1-
```

```
0.788h1.727l2.201,2.522l3.762,5.833l3.451,0.473l2.197-0.947l1.727-1.733l-
0.785-2.681l-2.039-1.576l-1.412,0.788l-0.941-1.261l0.475-0.473l2.035-
0.157l1.727,0.788l1.887,2.365l0.938,2.837l0.314,2.365l-4.08,1.418l-
3.764,1.892l-3.768,4.415l-1.883,1.419v-0.947l2.354-1.418l0.471-1.735l-0.783-
2.996l-2.826,1.42l-0.785,1.418l0.473,2.207l-2.668,1.418l-2.666-4.415l-3.297-
4.257l-1.571-1.104l-5.57,1.498l-4.943,1.024l-21.183,4.652l-0.94-5.597l0.629-
10.327l5.021-0.867L828.562,165.524").attr(attr).scale(scaleNum,scaleNum, 0,
0);
            us.ME = R.path("M893.999,92.052l1.887,2.049l2.197,3.626v1.892l-
2.041,4.572l-1.883,0.63l-3.296,2.996l-4.712,5.361c0,0-0.627,0-1.253,0c-
0.629,0-0.943-2.049-0.943-2.049l-1.725,0.158l-0.939,1.418l-2.354,1.419l-
0.941,1.418l1.572,1.42l-0.475,0.631l-0.471,2.68l-1.883-0.157v-1.577l-0.314-
1.261l-1.412,0.316l-1.727-3.153l-2.041,1.261l1.256,1.418l0.314,1.104l-
0.789,1.261l0.314,2.996l0.16,1.576l-1.572,2.522l-2.826,0.474l-0.312,2.837l-
5.176,2.996l-1.256,0.474l-1.57-1.419l-2.984,3.469l0.941,3.153l-1.414,1.261l-
0.157,4.257l-1.534,7.433l-2.385-1.127l-0.477-2.996l-3.762-1.104l-0.314-2.68l-
7.061-22.862l-4.627-14.41l2.371-0.336l1.465,0.399v-2.522l0.789-5.361l2.51-
4.572l1.414-3.941l-1.889-2.365v-5.833l0.789-0.945l0.785-2.681l-0.157-1.419l-
0.16-4.73l1.727-4.73l2.824-8.672l2.04-
4.099h1.254l1.255,0.157v1.104l1.255,2.207l2.668,0.63l0.785-0.788v-
0.945l3.922-2.837l1.727-
1.735l1.412,0.158l5.807,2.365l1.886,0.946l8.786,29.168h5.809l0.785,1.892l0.15
9,4.73l2.824,2.207h0.786l0.158-0.473l-0.471-
1.104L893.999,92.052L893.999,92.052z M873.679,121.455l1.492-
1.499l1.334,1.025l0.55,2.365l-1.648,0.867L873.679,121.455L873.679,121.455z
M880.193,115.701l1.723,1.813c0,0,1.255,0.079,1.255-0.236s0.232-1.971,0.232-
1.971l0.866-0.788l-0.785-1.735l-
1.961,0.709L880.193,115.701z").attr(attr).scale(scaleNum,scaleNum, 0, 0);
            us.NH = R.path("M853.376,155.98l0.297-1.493l1.057-3.21l-2.467-
0.891l-0.477-2.996l-3.762-1.104l-0.314-2.68l-7.061-22.862l-4.468-14.183l-
0.867-0.005l-0.627,1.576l-0.629-0.474l-0.938-0.945l-1.412,1.892l-
0.931,5.35l0.303,5.527l1.883,2.68v3.941l-3.609,3.941l-
2.512,1.104v1.104l1.1,1.734v8.356l-0.785,8.988l-0.159,4.73l0.941,1.261l-
0.158,4.415l-0.474,1.735l1.414,1.971l16.795-4.021l2.195-0.63l2.04-
3.153L853.376,155.98L853.376,155.98z").attr(attr).scale(scaleNum,scaleNum, 0,
0);
            us.VT = R.path("M817.106,167.888l-0.785-5.519l-2.98-10.721l-
0.629-0.316l-2.824-1.261l0.785-2.837l-0.785-2.05l-2.512-4.415l0.943-3.784l-
0.789-5.045l-2.35-6.307l-0.783-4.801l26.361-
7.023l0.314,5.675l1.883,2.68v3.941l-3.609,3.941l-
2.512,1.104v1.104l1.1,1.734v8.356l-0.785,8.987l-0.16,4.73l0.941,1.261l-
0.157,4.415l-0.474,1.735l1.414,1.971l-
6.748,1.34L817.106,167.888L817.106,167.888z").attr(attr).scale(scaleNum,scale
Num, 0, 0);
            us.NY = R.path("M801.728,202.26l-1.1-0.946l-2.51-0.158l-2.197-
1.892l-2.354-5.203l-2.916-0.908l-2.264-2.087l-18.045,3.941l-41.74,8.514l-
8.635,1.419l-0.717-6.751l2.6-1.762l1.252-1.104l0.941-1.576l1.727-1.104l1.883-
1.735l0.471-1.576l2.041-2.68l1.098-0.945l-0.156-0.946l-1.256-2.996l-1.727-
0.157l-1.883-5.991l2.826-1.735l4.236-1.418l3.924-1.262l3.139-0.473l6.119-
0.157l1.883,1.261l1.572,0.157l2.039-1.261l2.512-1.104l5.021-0.473l2.039-
1.734l1.726-3.153l1.571-1.892h2.039l1.883-1.104l0.16-2.207l-1.415-2.05l-
0.312-1.418l1.098-2.049v-1.419h-1.727l-1.725-0.788l-0.785-1.104l-0.16-
2.522l5.65-5.361l0.627-0.788l1.412-2.837l2.826-4.415l2.668-3.625l2.035-
```

```
2.365l2.348-1.78l2.989-1.215l5.338-1.261l3.136,0.157l4.396-1.418l7.344-
2.02l0.506,4.856l2.354,6.307l0.783,5.045l-
0.941,3.784l2.512,4.415l0.789,2.049l-
0.789,2.837l2.824,1.261l0.629,0.316l2.984,10.721l0.471,5.045l-
0.471,10.564l0.785,5.361l0.785,3.468l1.412,7.096v7.883l-
1.1,2.207l1.783,1.944l-0.217,1.526l-1.883,1.735l0.312,1.261l1.252-
0.316l1.412-1.261l2.197-2.522l1.099-0.631l1.572,0.631l2.197,0.157l7.689-
3.784l2.824-2.681l1.252-1.418l4.08,1.576l-3.299,3.469l-3.764,2.837l-
6.901,5.202l-2.512,0.945l-5.648,1.892l-3.924,1.104l-1.467-0.52l-0.567-
3.266l0.472-2.68l-0.16-2.05l-2.51-1.104l-4.395-0.946l-3.766-
1.104L801.728,202.26L801.728,202.26z").attr(attr).scale(scaleNum,scaleNum, 0,
0);
            us.NJ = R.path("M801.728,202.26l-2.041,2.365v2.996l-1.883,2.996l-
0.157,1.576l1.253,1.261l-0.16,2.365l-
2.198,1.104l0.786,2.68l0.16,1.104l2.667,0.315l0.94,2.523l3.451,2.365l2.351,1.
577v0.788l-3.136,2.996l-1.572,2.208l-1.412,2.68l-2.197,1.261l-1.178,0.709l-
0.234,1.183l-
0.592,2.542l1.061,2.189l3.139,2.837l4.71,2.208l3.925,0.63l0.156,1.42l-
0.783,0.945l0.312,2.68h0.785l2.039-2.365l0.785-4.73l2.666-3.941l2.984-
6.307l1.098-5.36l-0.627-1.104l-0.151-9.147l-1.572-3.311l-1.1,0.788l-
2.666,0.316l-0.475-0.473l1.098-0.946l2.039-1.892l0.062-1.067l-0.372-
3.349l0.473-2.68l-0.16-2.05l-2.51-1.104l-4.395-0.946l-3.766-
1.104L801.728,202.26L801.728,202.26z").attr(attr).scale(scaleNum,scaleNum, 0,
0);
            us.PA = R.path("M797.491,238.05l1.098-0.631l2.197-1.261l1.412-
2.68l1.572-2.208l3.139-2.996v-0.788l-2.354-1.576l-3.451-2.365l-0.939-2.523l-
2.668-0.315l-0.157-1.104l-0.788-2.68l12.2-1.104l0.158-2.365l-1.255-
1.261l0.156-1.576l1.884-2.996v-2.996l2.277-2.365l-1.334-0.947l-2.51-0.157l-
2.197-1.892l-2.357-5.203l-2.913-0.908l-2.265-2.087l-18.044,3.941l-
41.744,8.515l-8.632,1.419l-0.479-6.909l-5.326,5.49l-1.253,0.473l-
4.08,2.935l2.826,18.665l1.642,10.485l3.468,18.786l4.494-0.733l11.594-
1.465l36.819-7.476l14.441-2.753l8.057-1.583l1.252-1.229l2.039-
1.576h2.03v0.003H797.491L797.491,238.05z").attr(attr).scale(scaleNum,scaleNum
, 0, 0);
            us.DE = R.path("M796.391,241.362l0.787-2.05l0.234-1.261h-1.961l-
2.035,1.576l-
1.415,1.419l1.415,4.099l2.195,5.519l2.037,9.46l1.572,6.149l4.865-0.157l5.959-
1.183l-2.199-7.173l-0.939,0.473l-3.449-2.365l-1.727-4.572l-1.883-3.469l-
2.201-0.945l-2.037-
3.469L796.391,241.362L796.391,241.362z").attr(attr).scale(scaleNum,scaleNum,
0, 0);
            us.MD = R.path("M810.042,264.854l-5.957,1.261l-4.865,0.157l-
1.569-6.149l-2.04-9.46l-2.195-5.519l-1.256-4.289l-8.057,1.582l-14.441,2.754l-
36.354,7.364l1.098,4.888l0.941,5.519l0.314-0.316l2.035-2.365l2.197-
2.996l2.354-0.157l1.412-1.42l1.727-2.522l1.255,0.63l2.823-0.315l2.515-
2.05l1.947-1.417l1.788-
0.472l1.601,1.102l2.824,1.419l1.883,1.734l1.178,1.498l4.004,1.656v2.838l5.336
,1.261l1.883,1.261l0.941-1.892l2.197,1.576l-1.412,3.153l-0.314,2.681l-
1.725,2.522v2.05l0.627,1.734l4.916,1.322l4.186-
0.061l2.982,0.947l2.041,0.314l0.938-2.049l-1.412-2.049v-1.734l-2.352-2.05l-
2.039-5.361l1.254-5.203l-0.157-2.05l-1.255-1.261c0,0,1.415-1.576,1.415-
2.207c0-0.631,0.471-2.05,0.471-2.05l1.883-1.261l1.887-1.576l0.471,0.946l-
1.412,1.576l-1.256,3.626l0.314,1.104l1.727,0.315l0.472,5.361l-
```

```
2.038,0.945l0.314,3.469l0.471-0.157l1.098-1.892l1.572,1.734l-1.572,1.262l-
0.312,3.311l2.51,3.311l3.766,0.474l1.571-
0.788l3.138,5.045l1.727,0.474v3.468l-2.197,4.73l-
0.471,6.78l1.412,3.311l1.412,0.157l1.883-4.099l0.785-3.469l0.16-6.937l2.982-
4.73l2.039-6.78v-5.517L810.042,264.854L810.042,264.854z
M794.119,274.392l1.098,2.444l0.156,1.734l1.1,1.813c0,0,0.866-0.867,0.866-
1.183s-0.707-2.996-0.707-2.996l-0.705-
2.287L794.119,274.392z").attr(attr).scale(scaleNum,scaleNum, 0, 0);
            us.WV = R.path("M733.324,252.621l1.08,4.822l0.941,5.519l0.314-
0.316l2.035-2.365l2.197-2.996l2.354-0.157l1.412-1.42l1.727-
2.522l1.255,0.63l2.823-0.315l2.514-2.05l1.948-1.417l1.788-
0.472l1.601,1.102l2.824,1.419l1.883,1.734l1.334,1.262l-1.02,4.888l-5.494-
2.996l-4.393-1.734l-0.159,5.203l-0.472,2.049l-1.568,2.68l-0.627,1.576l-
2.984,2.365l-0.473,2.207l-3.295,0.316l-0.314,2.996l-1.098,5.361h-2.512l-
1.256-0.788l-1.568-2.68l-1.727,0.157l-0.314,4.257l-2.039,6.464l-
4.863,10.563l0.787,1.259l-0.16,2.683l-2.035,1.894l-1.412-0.317l-3.139,2.365l-
2.512-0.945l-1.727,4.572c0,0-3.607,0.787-4.236,0.945c-0.627,0.158-2.354-
1.262-2.354-1.262l-2.35,2.207l-2.513,0.631l-2.824-0.787l-1.255-1.262l-2.127-
2.949l-3.055-1.938l-2.512-2.685l-2.824-3.624l-0.623-2.208l-2.515-1.418l-
0.786-1.576l-0.23-5.124l2.116-0.079l1.886-0.788l0.156-2.68l1.572-1.419l0.157-
4.888l0.941-3.784l1.253-0.63l1.256,1.104l0.471,1.734l1.727-0.945l0.475-
1.576l-1.099-1.735v-2.365l0.938-1.261l2.197-3.311l1.256-
1.419l2.039,0.473l2.197-1.576l2.982-3.311l2.195-3.784l0.314-5.519l0.475-
4.888v-4.572l-1.101-2.996l0.94-1.419l1.247-1.261l3.388,19.338l4.498-
0.733L733.324,252.621L733.324,252.621z").attr(attr).scale(scaleNum,scaleNum,
0, 0);
            us.VA =
R.path("M699.568,306.477l1.883,3.469l1.256,1.262l2.824,0.787l2.512-
0.629l2.35-2.209c0,0,1.727,1.42,2.354,1.262s4.236-0.945,4.236-0.945l1.727-
4.572l2.512,0.945l3.139-2.365l1.412,0.319l2.035-1.896l0.16-2.68l-0.785-
1.262l4.868-10.563l2.039-6.464l0.314-4.257l1.727-
0.157l1.568,2.68l1.256,0.788h2.51l1.1-5.361l0.314-2.996l3.295-0.316l0.475-
2.207l2.982-2.365l0.627-1.576l1.568-2.68l0.474-2.05l0.159-
5.203l4.395,1.735l5.492,2.996l0.783-
4.966l4.08,1.972v2.837l5.336,1.261l1.883,1.261l0.941-1.892l2.197,1.577l-
1.412,3.153l-0.314,2.681l-
1.725,2.523v2.049l0.627,1.734l4.916,1.322l1.83,1.517l5.021,0.315l2.512,2.208l
3.139,0.631l1.256,1.261l-0.477,4.415l0.943,0.945l0.157,2.208l1.255,2.049l-
0.159,1.734l-3.136-
1.104v0.945l1.885,1.576v1.104l1.412,1.104l1.253,1.576l0.159,2.207l-
2.197,1.42l0.314,0.469l2.51-0.469l3.139-0.631l1.1-0.158l3.955,6.854l-
4.744,1.66l-11.141,2.834l-18.605,3.73l-19.998,3.684l-16.635,2.523l-
15.357,1.893l-6.926,1.26l-4.381-0.602l-2.055-0.029l-2.354,1.893l-
7.686,0.162l-11.643,1.854l-9.707,0.91l2.676-1.35l5.493-3.311l3.764-2.047v-
2.051l1.725-1.734l4.396-5.203l4.079-
3.469L699.568,306.477L699.568,306.477z").attr(attr).scale(scaleNum,scaleNum,
0, 0);
            us.KY = R.path("M699.457,307.202l-3.027,3.059l-4.08,3.469l-
4.395,5.203l-1.725,1.734v2.051l-3.764,2.049l-5.493,3.311l-2.535,1.371l-
50.35,4.779l-15.297,1.734l-4.486,0.5l-3.76-0.027v3.783l-8.16,0.473l-
6.748,0.631l-10.121,0.201l0.973-1.213l2.115-1.719l1.998-1.113l0.223-
3.123l0.887-1.783l-1.559-2.477l0.777-1.859l2.197-1.734l2.039-
0.631l2.668,1.262l3.451,1.262l1.098-0.318l0.16-2.207l-1.256-2.365l0.314-
```

```
2.205l1.885-1.418l2.512-0.631l1.57-0.631l-0.785-1.734l-0.627-1.893l1.098-
0.787l1.02-3.232l2.904-1.652l5.648-0.949l3.453-
0.469l1.412,1.89l1.727,0.789l1.727-3.154l2.824-
1.418l1.883,1.576l0.785,1.105l2.039-0.473l-0.156-3.311l2.822-1.578l1.102-
0.787l1.098,1.576h4.551l0.785-2.047l-0.314-2.207l2.824-3.469l4.551-
3.784l0.471-4.415l2.668-0.316l3.764-1.734l2.668-1.892l-0.313-1.899l-1.412-
1.42l0.546-2.128l4.002-0.236l2.354-
0.788l2.824,1.576l1.571,4.257l5.647,0.315l1.726,1.734l2.039,0.158l2.355-
1.419l2.98,0.473l1.256,1.419l2.667-2.523l1.727-
1.261h1.569l0.628,2.68l1.725,0.945l3.453,2.05l0.156,5.36l0.785,1.576l2.516,1
.419l0.623,2.207l2.824,3.623l2.512,2.684L699.457,307.202z").attr(attr).scale(
scaleNum,scaleNum, 0, 0);
            us.OH = R.path("M708.828,206.711l-6.906,4.063l-3.764,2.207l-
3.299,3.626l-3.924,3.784l-3.135,0.788l-2.824,0.473l-5.336,2.523l-
2.039,0.157l-3.299-2.996l-5.021,0.631l-2.512-1.419l-2.311-1.318l-
4.753,0.687l-9.887,1.576l-
7.531,1.183l1.253,14.268l1.724,13.402l2.515,22.861l0.546,5.597l4.002-
0.236l2.354-
0.788l2.824,1.576l1.571,4.257l5.646,0.315l1.727,1.734l2.039,0.158l2.354-
1.419l2.981,0.473l1.256,1.419l2.667-2.523l1.727-
1.261h1.569l0.626,2.68l1.727,0.945l3.373,2.287l2.117-0.079l1.888-
0.788l0.156-2.68l1.572-1.418l0.157-4.888l0.941-3.784l1.255-
0.631l1.255,1.104l0.471,1.734l1.727-0.945l0.472-1.577l-1.098-1.733v-
2.365l0.94-1.262l2.197-3.311l1.256-1.419l2.036,0.474l2.2-1.576l2.98-
3.311l2.197-3.784l0.314-5.519l0.472-4.888v-4.572l-1.098-2.996l0.94-
1.418l1.334-0.71l-1.805-
10.485L708.828,206.711L708.828,206.711z").attr(attr).scale(scaleNum,scaleNum,
0, 0);
            us.MI = R.path("M562.406,96.664l1.775-2.007l2.105-0.78l5.215-
3.791l2.219-0.557l0.441,0.446l-4.994,5.017l-3.219,1.895l-
1.996,0.892L562.406,96.664L562.406,96.664z
M646.059,127.998l0.627,2.444l3.135,0.157l1.256-1.183c0,0-0.08-1.418-0.393-
1.576c-0.314-0.157-1.572-1.813-1.572-1.813l-2.117,0.236l-1.57,0.158l-
0.314,1.104L646.059,127.998z M675.243,189.49l-3.139-8.042l-2.197-8.829l-
2.354-3.153l-2.512-1.734l-1.568,1.104l-3.768,1.735l-1.883,4.888l-
2.668,3.626l-1.098,0.631l-1.412-0.631c0,0-2.512-1.418-2.35-2.049c0.156-
0.63,0.471-4.888,0.471-4.888l3.295-1.261l0.789-3.311l0.623-2.522l2.354-
1.576l-0.312-9.776l-1.57-2.208l-1.256-0.788l-0.783-2.05l0.783-
0.788l1.572,0.316l0.156-1.577l-2.354-2.208l-1.252-2.523h-2.51l-4.397-1.418l-
5.336-3.311h-2.667l-0.627,0.631l-0.941-0.472l-2.982-2.208l-2.824,1.734l-
2.824,2.207l0.312,3.469l0.943,0.316l2.035,0.473l0.475,0.788l-2.511,0.788l-
2.515,0.315l-1.412,1.734l-0.312,2.049l0.312,1.576l0.314,5.361l-3.45,2.049l-
0.626-0.157V153.7l1.252-2.365l0.629-2.365l-0.785-0.788l-1.885,0.788l-
0.939,4.099l-2.668,1.104l-1.726,1.892l-0.158,0.946l0.627,0.788l-0.627,2.523l-
2.197,0.473v1.104l0.785,2.365l-1.1,5.991l-
1.57,3.942l0.625,4.572l0.473,1.104l-0.785,2.365l-0.312,0.788l-
0.312,2.68l13.451,5.833l2.824,6.307l1.412,4.73l-0.787,4.572l-0.939,5.833l-
2.352,5.045l-0.314,2.68l-2.992,3.034l-0.721,0.751l4.5-0.158l21.345-2.207l5.96-
0.63l0.314,1.183l7.533-1.183l9.885-1.576l5.065-0.449l-1.296-1.126l0.159-
1.419l2.038-3.626l1.941-1.695l-0.225-4.929l1.551-1.558l1.059-0.334l0.217-
3.468l1.49-2.957l1.02,0.591l0.16,0.63l0.783,0.158l1.883-0.947L675.243,189.49z
M547.591,125.541l1.797-1.009l2.667-0.788l3.451-2.207v-0.947l0.629-0.63l5.807-
0.947l2.354-1.892l4.236-2.049l0.158-1.261l1.887-2.838l1.727-0.788l1.254-
```

```
1.735l2.197-2.208l4.236-2.365l4.553-0.473l1.098,1.104l-0.312,0.947l-
3.611,0.945l-1.412,2.996l-2.197,0.788l-0.471,2.365l-2.354,3.153l-
0.314,2.522l0.787,0.472l0.939-1.104l3.453-
2.838l1.255,1.262h2.198l3.136,0.946l1.412,1.104l1.414,2.996l2.668,2.68l3.766-
0.157l1.411-0.946l1.568,1.262l1.572,0.473l1.255-0.788h1.098l1.572-
0.946l3.922-3.469l3.295-1.104l6.436-0.316l4.396-1.892l2.512-
1.261l1.412,0.157v5.519l0.471,0.316l2.824,0.788l1.883-0.473l5.965-
1.577l1.098-1.104l1.412,0.472v6.78l3.139,2.996l1.256,0.631l1.254,0.945l-
1.254,0.316l-0.788-0.316l-3.61-0.473l-2.037,0.63l-2.199-0.157l-3.138,1.419h-
1.725l-5.647-1.262l-5.024,0.158l-1.886,2.522l-6.748,0.631l-2.354,0.788l-
1.098,2.996l-1.252,1.104l-0.474-0.157l-1.412-1.576l-4.396,2.365h-0.627l-1.1-
1.576l-0.783,0.157l-1.883,4.257l-0.943,3.941l-3.639,7.934l-1.582-1.12l-1.332-
1.337l-1.552-10.034l-3.551-1.115l-1.332-2.229l-12.203-2.676l-2.442-1.115l-
7.993-2.23l-7.988-
1.115L547.591,125.541L547.591,125.541z").attr(attr).scale(scaleNum,scaleNum,
0, 0);
            us.WY = R.path("M342.631,157.013l-11.175-0.944l-31.148-3.214l-
15.757-2.007l-27.519-4.013l-19.307-2.899l-1.378,10.9l-3.726,23.662l-
5.104,29.655l-1.485,10.256l-
1.621,11.595l6.333,0.905l16.393,2.23l8.573,1.147l19.967,2.407l36.174,4.014l23
.747,2.007l4.217-43.256l1.554-
24.75L342.631,157.013L342.631,157.013z").attr(attr).scale(scaleNum,scaleNum,
0, 0);
            us.MT = R.path("M344.824,136.354l0.784-11.345l2.193-24.332l1.332-
14.716l1.223-13.879l-31.021-3.208l-28.404-3.467l-28.403-3.941l-31.385-5.203l-
17.89-3.311l-31.766-6.762l-4.348,20.82l3.33,7.358l-
1.332,4.459l1.776,4.459l3.107,1.338l3.55,10.034l3.55,3.567l0.444,1.115l3.329,
1.115l0.444,2.007l-6.88,17.169v2.453l2.44,3.122h0.888l4.661-2.899l0.666-
1.115l1.554,0.669l-
0.222,5.128l2.664,12.264l2.885,2.453l0.888,0.669l1.775,2.229l-
0.443,3.345l0.666,3.345l1.11,0.892l2.22-2.23h2.663l3.107,1.561l2.442-
0.892h3.994l3.55,1.561l2.664-0.446l0.444-2.898l2.884-
0.669l1.332,1.338l0.443,3.122l2.506,2.427l1.489-
11.123l19.307,2.899l27.519,4.014l15.757,2.006l31.148,3.214l11.136,1.174l1.644
-
15.516L344.824,136.354L344.824,136.354z").attr(attr).scale(scaleNum,scaleNum,
0, 0);
            us.ID = R.path("M134.357,189.816l4.609-18.009l4.217-17.28l1.332-
4.125l2.441-5.797l-1.22-2.229l-2.441,0.111l-0.776-1.003l0.444-1.115l0.333-
3.01l4.328-5.351l1.775-0.446l1.11-1.115l0.555-3.122l0.888-0.669l3.773-
5.685l3.772-4.237l0.222-3.679l-3.33-2.564l-1.275-4.292l0.389-9.421l3.55-
16.054l4.328-20.291l3.662-13.156l0.739-3.709l13.706,2.623l-
4.348,20.82l3.33,7.358l-
1.332,4.459l1.776,4.459l3.107,1.338l3.55,10.034l3.55,3.567l0.444,1.115l3.33,1
.115l0.444,2.006l-6.88,17.169v2.453l2.44,3.122h0.888l4.661-2.899l0.666-
1.115l1.554,0.669l-
0.222,5.128l2.663,12.263l2.885,2.453l0.888,0.669l1.776,2.229l-
0.444,3.345l0.666,3.345l1.11,0.892l2.22-2.23h2.663l3.107,1.561l2.442-
0.892h3.994l3.55,1.561l2.664-0.446l0.444-2.899l2.884-
0.668l1.332,1.337l0.444,3.122l2.552,2.006l-3.662,23.858l-4.994,29.544l-4.661-
0.78l-8.101-1.338l-9.987-1.784l-11.651-2.007l-12.317-2.285l-7.768-1.951l-
8.989-1.784l-9.432-
```

```
1.895L134.357,189.816L134.357,189.816z").attr(attr).scale(scaleNum,scaleNum,
0, 0);
            us.WA =
R.path("M88.652,22.838l4.237,1.418l9.416,2.68l8.318,1.892l19.459,5.519l22.284
,5.519l15.652,3.624l-0.975,3.788l-3.662,13.156l-4.328,20.29l-3.55,16.054l-
0.186,9.461l-13.574-3.329l-14.647-3.456l-14.98,0.112l-0.444-1.338l-
5.326,2.006l-4.328-0.557l-2.331-1.561l-1.22,0.669l-4.55-0.223l-1.664-1.337l-
5.104-2.007l-0.777,0.111l-4.217-1.449l-1.886,1.784l-5.992-0.334l-5.77-
4.013l0.666-0.781l0.222-7.581l-2.22-3.791l-3.994-0.558l-0.666-2.453l-2.285-
0.455l-1.863-1.46l-1.727,0.945l-2.196-2.837l0.313-2.837l2.667-0.316l1.57-
3.941l-2.511-1.104l0.157-3.625l4.237-0.63l-2.667-2.68l-1.416-6.939l0.628-
2.837v-7.726l-1.727-3.153l2.196-
9.144l2.04,0.473l2.354,2.838l2.667,2.522l3.14,1.891l4.394,2.049l2.981,0.631l2
.825,1.418l3.295,0.947l2.197-0.158v-2.365l1.256-1.104l2.04-
1.261l0.314,1.104l0.313,1.734l-
2.197,0.474L85.2,40.654l1.727,1.418l1.098,2.365l0.628,1.892l1.412-
0.157l0.158-1.261l-0.942-1.262l-0.471-3.153l0.784-1.733l-0.627-1.419v-
2.208l1.726-3.469l-1.099-2.522l-2.354-4.73l0.314-
0.788L88.652,22.838L88.652,22.838z M79.472,28.668l1.961-
0.157l0.471,1.34l1.49-1.576h2.276l0.784,1.498l-1.49,1.656l0.628,0.788l-
0.706,1.971l-1.334,0.395c0,0-0.864,0.079-0.864-0.237c0-0.315,1.412-
2.523,1.412-2.523l-1.647-0.552l-0.314,1.419l-0.706,0.63l-1.49-
2.207L79.472,28.668L79.472,28.668z").attr(attr).scale(scaleNum,scaleNum, 0,
0);
            us.TX = R.path("M344.417,341.768l22.026,1.059l30.182,1.115l-
1.332,23.189l-0.444,17.393l0.222,1.561l4.221,3.568l1.995,1.113l0.666-
0.23l0.666-2.006l1.327,1.783h1.998v-1.338l2.663,1.338l-
0.44,3.791l3.995,0.223l2.44,1.115l3.995,0.668l2.441,1.785l2.219-
2.008l3.329,0.67l2.441,3.344h0.892v2.23l2.216,0.668l2.219-
2.23l1.779,0.67h2.441l0.888,2.453l4.66,1.783l1.328-0.668l1.779-
4.014h1.109l1.113,2.006l3.995,0.67l3.549,1.338l2.885,0.891l1.775-0.891l0.666-
2.453h4.218l1.996,0.896l2.663-2.012h1.113l0.665,1.562h3.995l1.554-
2.006l1.775,0.441l1.996,2.453l3.107,1.783l3.107,0.895l2.663,1.227l2.221,1.896
l2.881-
1.34l2.666,1.115l0.651,10.828l0.018,9.463l0.665,9.143l0.663,3.791l2.444,4.014
l0.888,4.904l4.217,5.352l0.223,3.123l0.666,0.668l-0.666,8.25l-
2.881,4.906l1.553,2.006l-0.666,2.453l-0.666,7.137l-1.33,3.119l0.273,3.498l-
6.277,1.703l-9.572,4.414l-0.938,1.893l-2.513,1.895l-2.038,1.416l-
1.256,0.789l-5.492,5.201l-2.666,2.049l-5.176,3.152l-5.493,2.365l-6.12,3.311l-
1.726,1.42l-5.649,3.471l-3.296,0.629l-3.765,5.359l-3.925,0.318l-
0.939,1.893l2.198,1.896l-1.412,5.354l-1.254,4.414l-1.1,3.785l-
0.784,4.414l0.784,2.365l1.727,6.779l0.94,5.988l1.727,2.684l-0.94,1.416l-
2.983,1.895l-5.493-3.783l-5.335-1.104l-1.255,0.469l-3.139-0.627l-4.079-
2.996l-5.021-1.105l-7.376-3.312l-2.039-3.781l-1.254-6.307l-3.138-1.895l-
0.628-2.207l0.628-0.627l0.312-3.316l-1.254-0.625l-0.627-0.951l1.254-4.256l-
1.57-2.209l-3.141-1.26l-3.295-4.26l-3.453-6.465l-4.08-2.521l0.158-1.895l-
5.179-11.982l-0.784-4.1l-1.727-1.896l-0.158-1.416l-5.806-5.201l-2.511-2.996v-
1.104l-2.511-2.049l-6.591-1.105l-7.218-0.631l-2.981-2.205l-4.394,1.734l-
3.453,1.416l-2.196,3.154l-0.942,3.625l-4.237,5.99l-2.354,2.365l-2.511-0.949l-
1.727-1.105l-1.883-0.629l-3.767-2.207v-0.631l-1.726-1.891l-5.022-2.051l-
7.218-7.564l-2.198-4.572v-7.883l-3.138-6.307l-0.471-2.682l-1.57-0.947l-1.098-
2.047l-4.865-2.049l-1.255-1.578l-6.905-7.725l-1.255-3.154l-4.551-2.207l-
1.412-4.256l-2.511-2.838l-1.883-0.475l-0.631-
```

```
4.561l7.768,0.668l28.186,2.676l28.185,1.561l2.219-23.184l3.773-54.18l1.554-
18.283l1.332,0.027 M441.66,569.936l-0.548-6.938l-2.668-7.016l-0.548-
6.859l1.492-8.041l3.217-6.699l3.373-5.283l3.06-3.475l0.628,0.238l-
4.63,6.463l-4.236,6.387l-1.962,6.463l-
0.313,5.051l0.865,5.988l2.511,7.02l0.471,5.047l0.16,1.418L441.66,569.936z").a
ttr(attr).scale(scaleNum,scaleNum, 0, 0);
            us.CA = R.path("M131.179,394.456l4.015-0.789l1.442-2.119l0.554-
2.676l-3.55-0.557l-0.444-0.781l0.444-1.672l0.11-5.688l1.998-0.67l2.774-
2.676l0.555-4.793l1.442-3.568l1.886-1.895l3.329-1.672l1.554-1.338l0.111-
2.23l-0.999-0.557l-0.776-1.004l-1.22-5.688l-2.664-4.795l0.562-3.186l-2.449-
3.17l-14.093-22.184l-18.863-28.319l-22.082-33l-12.019-18.123l1.588-
6.405l6.769-25.307l7.878-30.658l-13.093-3.567l-13.093-3.345l-12.206-4.014l-
7.323-2.006l-11.096-2.899l-6.844-2.352l-1.535,4.607l-0.157,7.253l-
5.021,11.509l-2.982,2.522l-0.313,1.104l-1.727,0.788l-1.412,4.099l-
0.785,3.153l2.667,4.099l1.57,4.1l1.099,3.469l-0.314,6.307l-1.726,3l-
0.628,5.676l-0.941,3.626l1.727,3.784l2.667,4.415l2.196,4.73l1.255,3.941l-
0.313,3.153l-0.314,0.472v2.049l5.493,6.149l-0.471,2.365l-0.628,2.207l-
0.627,1.892l0.157,8.042l2.04,3.626l1.883,2.522l2.667,0.473l0.941,2.68l-
1.099,3.469l-2.04,1.576h-1.098l-
0.784,3.784l0.471,2.838l3.138,4.257l1.569,5.203l1.413,4.571l1.255,2.996l3.295
,5.676l1.413,2.523l0.471,2.838l1.57,0.945v2.365l-0.784,1.895l-1.727,6.934l-
0.471,1.895l2.354,2.68l4.08,0.475l4.394,1.734l3.766,2.049h2.825l2.825,2.994l2
.511,4.73l1.099,2.207l3.766,2.049l4.708,0.789l1.412,2.047l0.628,3.154l-
1.413,0.631l0.314,0.945l3.139,0.789l2.667,0.156l2.825,4.572l3.766,4.1l0.785,2
.207l2.511,4.1l0.313,3.154v9.145l0.471,1.734l9.73,1.418l19.145,2.68L131.179,3
94.456L131.179,394.456z M45.184,345.34l1.256,1.498l-0.158,1.262l-3.139-
0.078l-0.548-1.184l-0.628-1.418L45.184,345.34L45.184,345.34z
M47.066,345.34l1.177-0.631l3.452,2.051l2.982,1.182l-0.863,0.631l-4.394-
0.236l-1.57-1.576L47.066,345.34z
M67.153,364.655l1.727,2.287l0.785,0.945l1.49,0.553l0.549-1.412l-0.941-1.742l-
2.589-1.971l-1.021,0.166V364.655z M65.74,373.09l1.727,3.074l1.177,1.896l-
1.413,0.232l-1.255-1.184c0,0-0.706-1.414-0.706-1.809s0-2.133,0-
2.133L65.74,373.09L65.74,373.09z").attr(attr).scale(scaleNum,scaleNum, 0, 0);
            us.AZ = R.path("M131.532,394.557l-2.55,2.105l-
0.314,1.418l0.471,0.947l18.36,10.406l11.77,7.416l14.281,8.354l16.32,9.779l11.
926,2.365l25.95,3.418l1.989-12.98l3.643-26.256l6.761-51.889l4.133-30.043l-
24.967-3.746l-26.409-4.459l-32.45-6.162l-2.836,17.645l-0.444,0.441l-
1.664,2.564l-2.44-0.111l-1.22-2.676l-2.664-0.334l-0.888-1.115h-0.888l-
0.888,0.562l-1.887,1.002l-0.11,6.801l-0.222,1.672l-0.555,12.264l-
1.442,2.119l-0.554,3.232l2.664,4.795l1.22,5.688l0.776,1.004l0.999,0.557l-
0.111,2.23l-1.554,1.338l-3.329,1.672l-1.886,1.895l-1.442,3.568l-0.555,4.793l-
2.774,2.676l-1.998,0.67l-0.11,5.686l-0.444,1.672l0.444,0.781l3.55,0.557l-
0.554,2.68l-
1.442,2.117L131.532,394.557L131.532,394.557z").attr(attr).scale(scaleNum,scal
eNum, 0, 0);
            us.NV =
R.path("M134.357,189.816l20.366,4.399l9.432,1.895l8.989,1.784l7.823,2.062l-
1.165,5.407l-3.44,17.057l-3.662,19.956l-1.886,8.697l-2.108,13.267l-
3.218,16.165l-3.107,14.827l-1.91,10.241l-2.861,17.297l-0.444,0.443l-
1.664,2.564l-2.44-0.111l-1.22-2.676l-2.664-0.334l-0.888-1.115H147.41l-
0.888,0.559l-1.887,1.004l-0.11,6.801l-0.222,1.672l-0.555,12.264l-
1.439,2.143l-2.445-3.145l-14.093-22.182l-18.863-28.32l-22.082-33l-12.019-
```

```
18.123l1.588-6.405l6.769-25.307l7.656-
30.573l32.624,7.941l13.316,2.898").attr(attr).scale(scaleNum,scaleNum, 0, 0);
            us.UT = R.path("M244.16,319.862l-24.855-4.014l-26.41-4.459l-
32.524-6.029l2.009-10.025l3.107-14.827l3.218-16.165l2.108-13.268l1.887-
8.696l3.662-19.956l3.44-17.057l1.082-
5.435l12.345,2.202l11.651,2.007l9.987,1.784l8.101,1.338l4.661,0.78l-
1.597,10.368l-1.621,11.595l6.333,0.905l16.394,2.23l8.904,1.147l-
3.002,21.582l-3.107,21.851l-3.643,27.294l-
1.461,10.835L244.16,319.862L244.16,319.862z").attr(attr).scale(scaleNum,scale
Num, 0, 0);
            us.CO = R.path("M362.448,331.456l3.995-64.215l1.554-21.851l-
32.402-2.675l-23.746-2.007l-36.174-4.014l-20.085-2.453l-2.552,21.628l-
3.107,21.851l-3.643,27.295l-1.461,10.834l-
0.554,3.791l33.4,4.014l35.856,4.322l31.803,2.148l6.464,0.67l11.096,0.557").at
tr(attr).scale(scaleNum,scaleNum, 0, 0);
            us.NM = R.path("M273.354,437.182l-0.787-
4.719l7.767,0.668l28.186,2.676l28.185,1.562l2.22-23.189l3.772-54.182l1.554-
18.283l1.332,0.027l-0.014-11.428l-32.039-2.344l-35.856-4.32l-33.456-4.012l-
4.078,29.996l-6.761,51.889l-3.643,26.256l-1.989,12.98l15.007,1.939l1.255-
9.773l16.164,2.521L273.354,437.182L273.354,437.182z").attr(attr).scale(scaleN
um,scaleNum, 0, 0);
            us.OR = R.path("M134.639,189.422l4.328-17.615l4.217-17.28l11.332-
4.125l2.441-5.797l-1.22-2.229l-2.441,0.111l-0.776-1.003l0.444-1.115l0.333-
3.01l4.328-5.35l11.775-0.446l1.11-1.115l0.555-3.122l0.888-0.669l3.773-
5.685l3.772-4.237l0.222-3.679l-3.33-2.564l-1.017-4.085l-13.63-3.496l-14.647-
3.456l-14.98,0.111l-0.444-1.338l-5.326,2.007l-4.328-0.557l-2.331-1.561l-
1.22,0.669l-4.55-0.223l-1.664-1.338l-5.104-2.006l-0.777,0.111l-4.217-1.449l-
1.886,1.784l-5.992-0.334l-5.77-4.014l0.666-0.78l0.222-7.581l-2.219-3.791l-
3.994-0.558l-0.666-2.453l-2.285-0.455L54.6,77.704l-2.197,6.307l-3.138,9.775l-
3.139,6.307l-4.865,13.717l-6.277,13.244l-7.846,12.298l-1.883,2.837l-
0.784,8.356l-
1.255,5.833l2.161,4.379l6.845,2.353l11.096,2.899l7.324,2.006l12.206,4.014l13.
093,3.345l13.093,3.792 M134.357,189.816l-45.546-
10.788l32.623,7.941l13.316,2.898").attr(attr).scale(scaleNum,scaleNum, 0, 0);
            us.ND = R.path("M456.102,141.929l-0.666-8.25l-1.775-7.135l-1.775-
13.156l-0.443-9.588l-1.775-3.344l-1.554-4.906V85.516l0.666-3.791l-1.898-
5.363l-28.525-0.55l-18.047-0.631l-25.736-1.261l-24.215-1.837l-1.223,13.878l-
1.332,14.716l-2.193,24.332l-
0.627,11.843l54.529,3.515L456.102,141.929L456.102,141.929z").attr(attr).scale
(scaleNum,scaleNum, 0, 0);
            us.SD = R.path("M457.55,215.575l-0.926-1.993l-1.632-2.755l1.775-
4.237l1.327-5.574l-2.663-2.007l-0.442-2.675l0.89-2.453h1.775l0.441-6.689l-
0.222-29.655l-0.441-2.899l-3.997-3.344l-1.11-1.784v-1.561l1.997-1.561l1.327-
1.338l0.336-3.122l-56.48-1.561l-54.529-3.79l-0.745,5.149l-1.565,15.476l-
1.306,17.503l-
1.554,25.084l15.091,1.003l19.53,1.115l17.31,1.115l23.082,1.115l10.435-
0.446l1.995,2.23l4.66,2.898l1.113,0.892l4.216-1.338l3.773-0.446l2.663-
0.223l1.775,1.338l4.886,1.561l2.882,1.561l0.443,1.561l0.888,2.007h1.775L457.5
5,215.575L457.55,215.575z").attr(attr).scale(scaleNum,scaleNum, 0, 0);
            us.NE = R.path("M468.974,257.206l1.331,2.453l-
0.222,2.229l2.441,3.79l3.106,4.014h-5.992l-42.828-0.446l-39.281-1.338l-
21.197-0.78l1.664-21.74l-32.402-2.676l4.217-
42.922l15.091,1.003l19.53,1.115l17.31,1.115l23.081,1.115l10.435-
```

```
0.446l1.995,2.23l4.66,2.898l1.113,0.892l4.216-1.338l3.773-0.446l2.663-
0.223l1.777,1.338l4.884,1.561l2.882,1.561l0.443,1.561l0.888,2.006h1.775l1.554
-
0.111l1.113,5.24l2.663,7.358l0.888,4.682l2.22,3.568l0.666,5.128l1.553,4.013l0
.223,6.466l1.883,5.365").attr(attr).scale(scaleNum,scaleNum, 0, 0);
            us.IA =
R.path("M548.812,213.057l0.052,1.784l2.221,0.669l0.887,1.115l0.445,1.784l3.77
3,3.345l0.662,2.229l-0.662,3.345l-1.779,3.567l-0.666,2.453l-2.214,1.784l-
1.779,0.669l-5.104,1.337l-0.664,1.784l-0.666,2.007l0.666,1.338l1.775,1.561l-
0.223,4.013l-1.775,1.561l-0.666,1.561v2.675l-1.775,0.446l-1.553,1.115l-
0.227,1.338l0.227,2.007l-1.67,1.728l-3.217-3.29l-1.109-2.229l-7.547,0.669l-
9.543,0.446l-24.633,0.892l-13.095,0.223l-9.099,0.223l-1.057,0.118l-1.606-
5.246l-0.223-6.466l-1.553-4.014l-0.666-5.128l-2.22-3.567l-0.888-4.683l-2.665-
7.358l-1.111-5.24l-1.332-2.118l-1.552-2.676l1.778-4.237l1.327-5.574l-2.663-
2.007l-0.442-2.675l0.89-2.453h1.665h11.207l48.159-0.669l17.536-0.669l3.994-
0.112l0.666,3.233l2.218,1.561l0.224,1.338l-
1.999,3.345l0.227,3.122l2.439,3.79l2.441,1.115l2.885,0.446L548.812,213.057L5
48.812,213.057z").attr(attr).scale(scaleNum,scaleNum, 0, 0);
            us.MS = R.path("M604.971,472.063l-1.129,1.225h-5.025l-1.412-
0.787l-2.037-0.316l-6.591,1.893l-1.727-0.789l-2.51,4.11l-1.07,0.76l-1.092-
2.428l-1.109-3.791l-3.334-3.121l1.114-7.357l-0.666-0.893l-1.775,0.223l-
7.994,0.67l-23.523,0.668l-0.441-1.561l0.666-7.805l3.33-6.021l5.103-8.918l-
0.887-2.008h1.112l0.666-3.121l-2.221-1.783l0.223-1.785l-1.998-4.459l-0.276-
5.213l1.328-2.59l-0.389-4.236l-1.328-2.898l1.328-1.338l-1.328-2.006l0.441-
1.785l0.887-6.02l2.885-2.676l-0.666-2.006l3.553-5.129l2.662-0.893v-2.453l-
0.666-1.338l2.666-5.127l2.662-1.115l0.105-3.328l8.42-0.076l23.383-
1.891l5.328-0.223l0.008,6.213l0.159,16.24l-0.785,30.271l-
0.16,13.717l2.668,18.289L604.971,472.063L604.971,472.063z").attr(attr).scale(
scaleNum,scaleNum, 0, 0);
            us.IN = R.path("M598.351,311.049l-0.157-3.783l0.471-4.412l2.197-
2.836l1.727-3.786l2.51-4.11-0.471-5.676l-1.725-2.68l-0.314-3.153l0.785-
5.361l-0.471-6.78l-1.256-15.609l-1.255-14.979l-0.941-
11.431l2.984,0.868l1.412,0.945l1.1-0.315l2.039-1.892l2.742-1.577l4.945-
0.158l21.344-2.207l5.961-
0.63l0.16,1.104l1.41,14.346l1.727,13.402l2.512,22.861l0.471,5.519l-
0.471,2.208l1.412,1.419l0.312,1.892l-2.668,1.892l-3.768,1.735l-2.665,0.316l-
0.472,4.415l-4.55,3.784l-2.824,3.468l0.313,2.208l-0.788,2.047h-4.548l-1.101-
1.576l-1.1,0.789l-2.824,1.576l0.157,3.311l-2.039,0.473l-0.785-1.104l-1.883-
1.576l-2.824,1.418l-1.727,3.154l-1.727-0.789l-1.412-1.891l-3.453,0.469l-
5.648,0.945L598.351,311.049L598.351,311.049z").attr(attr).scale(scaleNum,scal
eNum, 0, 0);
            us.IL = R.path("M598.193,310.891v-3.625l0.471-4.412l2.197-
2.836l1.727-3.785l2.51-4.11-0.471-5.676l-1.725-2.68l-0.314-3.153l0.785-
5.361l-0.471-6.78l-1.256-15.609l-1.255-14.979l-0.785-11.352l-1.256-0.788l-
0.785-2.523l-1.256-3.626l-1.568-1.734l-1.412-2.523l-0.227-5.353l-9.617,1.28l-
26.404,1.672l-8.434-
0.418l0.221,2.314l2.221,0.668l0.887,1.115l0.443,1.784l3.773,3.345l0.662,2.229
l-0.662,3.345l-1.779,3.567l-0.666,2.453l-2.216,1.782l-1.779,0.669l-
5.104,1.337l-0.664,1.784l-0.666,2.006l0.666,1.338l1.775,1.561l-0.223,4.014l-
1.775,1.561l-0.666,1.561v2.676l-1.775,0.446l-1.553,1.115l-
0.227,1.338l0.227,2.007l-1.67,1.283l-
0.998,2.731l0.443,3.567l2.218,7.135l7.104,7.358l5.323,3.567l-
0.223,4.237l0.889,1.338l6.213,0.446l2.665,1.338l-0.663,3.567l-2.219,5.797l-
```

```
0.666,3.12l2.219,3.791l6.215,5.129l4.434,0.668l1.999,4.906l1.993,3.121l-
0.888,2.898l1.553,4.014l1.778,2.008l2.878-0.33l0.557-2l2.197-1.734l2.04-
0.631l2.667,1.262l3.453,1.262l1.098-0.318l0.159-2.207l-1.255-2.365l0.314-
2.205l1.883-1.418l2.51-0.631l1.572-0.631l-0.786-1.734l-0.626-1.893l1.098-
0.787L598.193,310.891L598.193,310.891z").attr(attr).scale(scaleNum,scaleNum,
0, 0);
            us.MN = R.path("M455.88,141.929l-0.444-8.25l-1.775-7.135l-1.775-
13.156l-0.443-9.588l-1.775-3.344l-1.554-4.906V85.516l0.666-3.791l-1.771-
5.317l29.254,0.034l0.31-8.041l0.628-
0.158l2.196,0.473l1.888,0.788l0.783,5.361l1.412,5.99l1.57,1.576h4.711l0.314,
1.419l6.121,0.315v2.049h4.707l0.314-1.261l1.098-1.104l2.197-
0.631l1.256,0.946h2.824l3.763,2.522l5.18,2.365l2.354,0.473l0.471-0.945l1.412-
0.473l0.471,2.837l2.512,1.261l0.475-
0.472l1.254,0.157v2.05l2.512,0.946h2.984l1.568-0.788l3.139-3.153l2.51-
0.473l0.786,1.734l0.474,1.261h0.938l0.94-0.788l8.632-
0.315l1.726,2.996h0.629l0.69-1.058l4.311-0.362l-0.593,2.223l-3.825,1.792l-
8.976,3.961l-4.637,1.958l-2.982,2.522l-2.354,3.469l-2.197,3.784l-
1.727,0.788l-4.393,4.888l-1.256,0.157l-3.295,2.996l0.668,0.53l-2.736,2.64l-
0.223,2.676v8.25l-1.111,1.561l-5.104,3.791l-
2.222,5.797l0.445,0.223l2.441,2.007l0.664,3.122l-
1.778,3.122v3.79l0.444,6.466l2.885,2.898h3.328l1.775,3.122l3.332,0.446l3.77,5
.575l6.881,4.013l1.998,2.676l0.889,7.358h-3.994L516.9,193.21l-48.157,0.669h-
11.208l0.551-6.689l-0.222-29.655l-0.442-2.899l-3.996-3.344l-1.109-1.784v-
1.561l1.995-1.561l1.331-
1.338L455.88,141.929L455.88,141.929z").attr(attr).scale(scaleNum,scaleNum, 0,
0);
            us.WI = R.path("M593.031,210.159l0.141-4.114l-1.571-4.415l-0.627-
5.991l-1.1-2.365l0.941-2.996l0.785-2.838l1.412-2.522l-0.627-3.311l-0.629-
3.469l0.477-1.735l1.883-2.365l0.158-2.68l-0.789-1.261l0.629-2.523l0.472-
3.153l2.668-5.519l2.824-6.623l0.156-2.208l-0.312-0.946l-0.786,0.474l-
4.078,6.149l-2.668,3.941l-1.883,1.735l-0.785,2.208l-1.412,0.788l-
1.104,1.892l-1.412-0.316l-0.16-1.734l1.256-2.365l2.035-4.572l1.727-
1.576l1.07-2.235l-1.58-0.883l-1.332-1.337l-1.553-10.034l-3.549-1.115l-1.332-
2.229l-12.205-2.676l-2.441-1.115l-7.99-2.229l-7.988-1.115l-4.049-5.271l-
0.512,1.23l-1.098-0.158l-0.629-1.104l-2.668-0.788l-1.098,0.157l-1.727,0.946l-
0.938-0.63l0.625-1.892l1.887-2.996l1.098-1.104l-1.883-1.419l-2.041,0.788l-
2.824,1.892l-7.223,3.153l-2.82,0.63l-2.824-0.473l-0.953-0.856l-2.059,2.765l-
0.223,2.676v8.25l-1.109,1.561l-5.105,3.792l-
2.221,5.797l0.445,0.223l2.441,2.007l0.664,3.122l-
1.775,3.122v3.79l0.441,6.466l2.885,2.898h3.328l1.775,3.122l3.334,0.446l3.768,
5.575l6.881,4.013l1.999,2.676l0.889,7.246l0.666,3.233l2.218,1.561l0.224,1.338
l-
1.999,3.345l0.227,3.122l2.441,3.791l2.439,1.115l2.887,0.446l1.188,2.453h8.909
l25.852-
1.561L593.031,210.159L593.031,210.159z").attr(attr).scale(scaleNum,scaleNum,
0, 0);
            us.MO = R.path("M537.993,260.104l-3.107-3.122l-1.109-2.229l-
7.547,0.669l-9.543,0.446l-24.633,0.892l-13.095,0.223l-7.656,0.111l-
2.219,0.112l1.22,2.453l-
0.222,2.229l2.441,3.79l2.996,4.014l2.993,2.676l2.219,0.223l1.332,0.892v2.899l
-1.775,1.561l-0.445,2.23l1.998,3.344l2.441,2.899l2.441,1.784l1.332,11.371l-
0.666,34.449l0.221,4.57l0.445,6.689l22.631-0.445l22.637-0.67l20.197-
0.891l10.652-0.445l1.775,2.898l-0.445,2.229l-3.104,2.676l-
```

```
0.663,2.898l5.988,0.439l4.881-0.67l1.999-6.242l-0.139-5.492l2.803-1.42l1.33-
1.561l1.998-1.113l0.223-3.123l0.887-1.783l-1.332-2.369l-2.881,0.363l-1.779-
2.008l-1.553-4.014l0.887-2.898l-1.993-3.121l-1.999-4.906l-4.434-0.668l-6.215-
5.129l-2.219-3.791l0.666-3.121l2.219-5.794l0.666-3.567l-2.667-1.338l-6.213-
0.446l-0.889-1.338l0.224-4.237l-5.323-3.567l-7.102-7.358l-2.221-7.135l-0.443-
3.568L537.993,260.104L537.993,260.104z").attr(attr).scale(scaleNum,scaleNum,
0, 0);
            us.AR = R.path("M572.456,353.061l-4.725,0.916l-5.989-0.447l0.666-
2.898l3.104-2.676l0.443-2.229l-1.775-2.895l-10.654,0.441l-20.193,0.896l-
22.635,0.67l-
22.637,0.445l1.555,6.686v8.027l1.33,10.703l0.223,36.902l2.221,1.895l2.881-
1.338l2.666,1.115l0.639,10.953l21.555-0.027l18.643-0.893l10.484,0.135l1.439-
2.59l-0.389-4.236l-1.328-2.895l1.328-1.338l-1.328-2.006l0.441-1.785l0.887-
6.021l2.885-2.676l-0.666-2.006l3.553-5.129l2.662-0.895v-2.453l-0.666-
1.336l2.666-5.129l2.662-1.115l-0.287-3.561l2.38-1.104l0.941-4.57l-1.412-
3.471l3.924-2.207l0.312-2.523l1.311-
4.273L572.456,353.061L572.456,353.061z").attr(attr).scale(scaleNum,scaleNum,
0, 0);
            us.OK = R.path("M362.17,331.233l-10.375-0.445l-6.242-
0.475l0.249,0.195l-0.527,11.266l21.166,1.059l30.182,1.115l-1.332,23.184l-
0.444,17.393l0.222,1.561l4.217,3.568l1.997,1.113l0.666-0.225l0.666-
2.006l1.327,1.783h1.998v-1.338l2.663,1.338l-
0.44,3.791l3.995,0.223l2.441,1.115l3.994,0.668l2.441,1.785l2.219-
2.008l3.329,0.67l2.441,3.344h0.892v2.23l2.216,0.668l2.219-
2.23l1.779,0.67h2.441l0.888,2.453l4.66,1.783l1.328-0.668l1.779-
4.014h1.109l1.113,2.006l3.995,0.67l3.549,1.338l2.885,0.891l1.775-0.891l0.666-
2.453h4.217l1.997,0.893l2.663-2.008h1.113l0.665,1.562h3.995l1.554-
2.006l1.775,0.443l1.996,2.453l3.107,1.783l3.104,0.895l2.666,1.561l-0.221-
37.236l-1.332-10.703v-8.027l-1.553-6.686l-0.445-6.689l-0.221-4.348l-
12.873,0.781l-45.053-0.447l-43.72-
2.006L362.17,331.233L362.17,331.233z").attr(attr).scale(scaleNum,scaleNum, 0,
0);
            us.KS = R.path("M487.395,334.354l-12.875,0.67l-45.051-0.447l-
43.721-2.006l-23.441-1.227l4.022-
64.219l21.194,0.78l39.278,1.338l42.831,0.446h5.881l2.993,2.676l2.22,0.223l1.3
32,0.892v2.898l-1.775,1.561l-
0.445,2.23l1.997,3.344l2.442,2.899l2.441,1.784l1.33,11.371L487.395,334.354L48
7.395,334.354z").attr(attr).scale(scaleNum,scaleNum, 0, 0);
            us.LA = R.path("M583.596,477.94l-1.217-2.221l-1.109-3.791l-3.334-
3.121l1.114-7.357l-0.666-0.893l-1.775,0.223l-7.994,0.67l-23.523,0.668l-0.441-
1.561l0.666-7.805l3.33-6.021l5.103-8.918l-0.887-2.008h1.112l0.666-3.121l-
2.221-1.783l0.223-1.785l-1.998-4.459l-0.334-5.352h-10.539l-18.643,0.893l-
21.555,0.027l0.023,9.338l0.666,9.143l0.666,3.791l2.441,4.014l0.888,4.904l4.22
,5.35l0.223,3.123l0.666,0.668l-0.666,8.25l-2.883,4.906l1.552,2.006l-
0.664,2.453l-0.665,7.135l-1.33,3.121l0.12,3.529l4.548-1.482l7.847-
0.316l10.046,3.471l6.274,1.104l3.611-1.42l3.139,1.109l3.137,0.947l0.785-
2.051l-3.136-1.104l-2.512,0.473l-2.668-1.576c0,0,0.159-1.262,0.786-
1.418c0.626-0.158,2.983-0.947,2.983-0.947l1.727,1.418l1.726-
0.951l3.137,0.631l1.412,2.365l0.313,2.207l4.396,0.316l1.726,1.738l-
0.785,1.574l-1.255,0.791l1.568,1.574l8.16,3.471l3.453-1.262l0.941-2.365l2.51-
0.629l1.726-1.416l1.253,0.945l0.786,2.836l-2.198,0.789l0.627,0.629l3.297-
1.258l2.197-3.312l0.787-0.473l-2.039-0.316l0.783-1.576l-0.156-1.418l2.04-
0.473l1.098-1.26l0.624,0.789c0,0-
```

```
0.156,2.994,0.629,2.994c0.783,0,4.08,0.631,4.08,0.631l3.922,1.893l0.94,1.418h
2.824l1.099,0.947l2.199-2.994v-1.418h-1.256l-3.295-2.682l-5.648-0.787l-3.139-
2.205l1.098-2.684l2.197,0.318l0.157-0.631l-1.727-0.945v-0.475h3.139l1.726-
2.992l-1.255-1.895l-0.312-2.682l-1.412,0.158l-1.885,2.049l-0.627,2.521l-
2.981-0.629l-0.941-1.736l1.727-1.893l1.962-
1.734L583.596,477.94L583.596,477.94z").attr(attr).scale(scaleNum,scaleNum, 0,
0);

            // Loop through all of the states and add a click event which
chooses that state in QlikView
            for (var state in us) {
                  (function (st, state) {
                        st[0].style.cursor = "pointer";
                        st[0].onclick = function () {
                              _this.Data.SearchColumn(0, state);
                        }
                  })(us[state], state);
            }
            var checker = "";
            // Loop through data
            for (var rowIx = 0; rowIx < _this.Data.Rows.length; rowIx++) {
                  var row = _this.Data.Rows [rowIx];
                  var tState = "";
                  var per = 0;
                  var breakIt = "0";
                  tState = row[0].text;
                  //get percentage value
                  per = row[1].text/100;
                  //Loop through the states until you get the one currently
identified in the data
                  for (var state in us) {
                        (function (st, state) {
                              checker = state + ' ' + tState;
                              // If it's the state we want, set the opacity
to the state's percentage.  This way the cold color will show through the
proper amount.
                              if(tState == state){
                                    st.attr("fill-opacity", per);
                                    st.animate({fill: hot}, 800);
                                    breakIt = "1";
                              }
                        })(us[state], state);
                        if(breakIt == "1"){
                              break;
                        }
                  }
            }
    }, false);
}

// Call the initialization function
geoHeat_Example_Init();
```

Appendix D - Org Chart

Script.js

```
Qva.LoadScript('/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensions/Qli
kView/Examples/org/jquery.js', function () {

Qva.LoadScript('/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensions/Qli
kView/Examples/org/json.js', function () {
            traverse = function (json) {
            var returnThis = "";
            if (json.length > 1) {
                $.each(json, function (index, value) {
                                // Determine if the person has any people under
him/her and create the HTML.  If so, traverse through those using recursion
                    if (this.children.length > 0) {
                                        if (this.path == "dis") {
                                              returnThis += '<li><span
class="dis">' + this.name + '</span>';
                                              returnThis += "<ul>";
                                              $.each(this.children,
function(index, value){
                                                    returnThis += traverse(this);
                                              });
                                              returnThis += "</ul>";
                                        }
                                        else {
                                              returnThis += '<li><span
class="parent">' + this.name + '<span class="hiddenInfo"><h3>' + this.name +
'</h3><p class="d1">' + this.d1 + '</p><p class="d2">' + this.d2 +
'</p></span><a class="moreInfo">more information</a></span>';
                                              returnThis += "<ul>";
                                              $.each(this.children,
function(index, value){
                                                    returnThis += traverse(this);
                                              });
                                              returnThis += "</ul>";
                                        }
                    } else {
                                        if(this.path == "dis"){
                                              returnThis += '<li><span
class="dis">' + this.name + '</span>';
                                        }else{
                                    returnThis += '<li><span class="name">' +
this.name + '<span class="hiddenInfo"><h3>' + this.name + '</h3><p
class="d1">' + this.d1 + '</p><p class="d2">' + this.d2 + '</p></span><a
class="moreInfo">more information</a></span>';
                                        }
                    }
                    returnThis += '</li>';
                });
            } else {
                  //If there's only one item in the JSON.
```

```
                var current = "";
                if (json.name) {
                    current = json;
                } else {
                    current = json[0];
                }
                if (current.children.length > 0) {
                            if(current.path == "dis"){
                                  returnThis += "<li><span class='dis'>" +
current.name + "</span>";
                            returnThis += "<ul>";
                            $.each(current.children, function (index, value) {
                                returnThis += traverse(this);
                            });
                            returnThis += "</ul>";
                                }else{
                                  returnThis += "<li><span class='parent'>"
+ current.name + "<span class='hiddenInfo'><h3>" + current.name + "</h3><p
class='d1'>" + current.d1 + "</p><p class='d2'>" + current.d2 +
"</p></span><a class='moreInfo'>more information</a></span>";
                            returnThis += "<ul>";
                            $.each(current.children, function (index, value) {
                                returnThis += traverse(this);
                            });
                            returnThis += "</ul>";
                                }

                } else {
                            if(current.path == "dis"){
                                  returnThis += "<li><span class='dis'>" +
current.name + "</span>";
                            }else{
                                  returnThis += "<li><span class='name'>" +
current.name + "<span class='hiddenInfo'><h3>" + current.name + "</h3><p
class='d1'>" + current.d1 + "</p><p class='d2'>" + current.d2 +
"</p></span><a class='moreInfo'>more information</a></span>";
                            }

                }
                returnThis += '</li>';
            }
            return returnThis;
        }
        Qva.AddExtension('QlikView/Examples/org', function () {
                  var _this = this;
                  // Add CSS

Qva.LoadCSS("/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensions/QlikVi
ew/Examples/org/main.css");
            //Add class to the extension container so we can target it
                  if (this.Element.className.indexOf("orgFrame") < 0) {
                this.Element.className += " orgFrame";
            }
```

```
            $('.orgFrame').empty();
            var jsonHTML = "";
            var output = [];
                  // Loop through data and create proper JSON hierarchy.
            for (var rowIx = 0; rowIx < this.Data.Rows.length; rowIx++) {
                var row = this.Data.Rows[rowIx];
                        //Split the hyphen delineated hierarchy string into
an array
                var chain = row[0].text.split("-");
                        var cleanArray = new Array();
                        //create an array based on the hierarchical data for
this row
                        for (var j = 0; j < chain.length; j++) {
                              if ((row[3].text.indexOf(chain[j]) == 0) &&
(chain[j] != row[3].text)) {
                                    cleanArray.push(row[3].text);
                                  }else if ((row[3].text.indexOf(chain[j]) > -1)
&& (row[3].text.indexOf(chain[j]) != 0)) {
                                  }else{
                                    cleanArray.push(chain[j]);
                                  }
                        }
                        chain = cleanArray;
                var currentNode = output;
                        // traverse through the JSON to see if these new
people are already represented yet
                        // if not, add them in the proper place
                for (var j = 0; j < chain.length; j++) {
                    var wantedNode = chain[j];
                    var lastNode = currentNode;
                    for (var k = 0; k < currentNode.length; k++) {
                        if (currentNode[k].name == wantedNode) {
                                          if(((j + 1) == chain.length) &&
(currentNode[k].d1 == "dis")){
                                                //person has been wrongfully
dissed.  Undis them.
                                                //Also, add the proper
information for undissed peopl
                                                currentNode[k].path =
row[0].text;
                                                currentNode[k].d1 =
row[1].text;
                                                currentNode[k].d2 =
row[2].text;
                                          }
                                          currentNode =
currentNode[k].children;

                                break;
                        }
                    }
                    // If we couldn't find an item in this list of children
                    // that has the right name, create one:
```

```
                    if (lastNode == currentNode) {
                        if (wantedNode == chain[chain.length - 1]) {
                                        var tPath = row[0].text;
                            var desc1 = row[1].text;
                            var desc2 = row[2].text;
                        } else {
                            var tPath = 'dis';
                            var desc1 = 'dis';
                            var desc2 = 'dis';
                        }
                        var newNode = currentNode[k] = {
                            name: wantedNode,
                            path: tPath,
                            d1: desc1,
                            d2: desc2,
                            children: []
                        };
                        currentNode = newNode.children;
                    }
                }
            }
                  jsonHTML += "<ul class='top'>";
                  //Use the traverse function to create the HTML
            jsonHTML += traverse(output);
            jsonHTML += "</ul>";
            this.Element.innerHTML = jsonHTML;
                  //Create flyout
            var flyoutHTML = "<div id='org_fly'><a class='closeIt'>Close Fly
Out</a><div class='org_fly_top'></div><div class='org_fly_mid'>";
            flyoutHTML+= "</div><div class='org_fly_btm'></div><div
id='fly_arrow'></div></div>";
            this.Element.innerHTML += flyoutHTML;
                  // Add click event to select the chosen person in qlikview.
                  $('span.name').click(function(event){
                        if (!($(event.target).hasClass('moreInfo'))) {
                              _this.Data.SearchColumn(0, "*" +
$(this).find('.hiddenInfo h3').text() + "*");
                        }
                  });
                  $('span.parent').click(function(event){
                        if (!($(event.target).hasClass('moreInfo'))) {
                              _this.Data.SearchColumn(0, "*" +
$(this).find('.hiddenInfo h3').text() + "*");
                        }
                  });
                  // Add click event to display flyout and append selected
person's information inside of it
            $('.moreInfo').click(function(){
                        var tFly = $('#org_fly');
                        tFly.hide();
                        $('.org_fly_mid').empty();
                        tFly.detach();
                        $(this).parent().parent().append(tFly);
```

```
                //var pos = $(this).offset();
                        //var mpos = $('.orgFrame').offset();

        $('.org_fly_mid').html($(this).parent().find('.hiddenInfo').html());
                tFly.css('top', -20 + 'px').css('left',
(($(this).parent().parent().width()/2) + 70) + 'px').show();
            });
                    //In order to have the chart display properly using the
background images, the first and last people will need to be detected
            $.each($('.orgFrame ul'), function(){
                if($(this).children("li").length > 1){
                    $(this).children("li").first().addClass("first");
                    $(this).children("li").last().addClass("last");
                }else if (!($(this).hasClass("top"))) {
                        $(this).children("li").first().addClass("orphan");
                        $(this).css('background-image', 'none');
                    }else {
                        $(this).children("li").first().addClass("tiptop");
                    }
            });
                    // For the first and last items, a blank div will need to
be created to cover the unneeded part of the background image
                    $.each($('li .first'), function(){
                        var coverDiv = $('<div />').addClass('cover');
                        coverDiv.css('left','0px');
                        coverDiv.css('width','50%');
                        coverDiv.css('padding-right','2px');

                        $(this).prepend(coverDiv);
                    });
                    $.each($('li .last'), function(){
                        var coverDiv = $('<div />').addClass('cover');
                        coverDiv.css('right','-3px');
                        coverDiv.css('width','50%');
                        //coverDiv.css('padding-left','2px');
                        $(this).prepend(coverDiv);
                    });
                    $('.closeIt').click(function(){
                        $('#org_fly').hide();
                    });
                    //Set the focus of the frame to be the top of the org
chart.
                    window.setTimeout(function() {
                        $(".orgFrame").scrollLeft(($(".tiptop").width()/2) -
($('.orgFrame').width()/2));
                    }, 1000);

        });
    });
});
```

main.css

```
.orgFrame{position:relative;width:100% !important;overflow:scroll;white-
space:nowrap;text-align:center;}
.orgFrame ul{clear:both;list-style:none;margin:0px auto;padding:0px;}
.orgFrame ul span{display:inline-block;width:132px;height:45px;text-
align:center;position:relative;}
.orgFrame ul
span.parent{background:url(/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Ext
ensions/QlikView/Examples/org/org_box_parent.png) white no-repeat top
left;cursor:pointer;}
.orgFrame ul
span.dis{background:url(/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extens
ions/QlikView/Examples/org/org_box_dis.png) white no-repeat top
left;color:#b2b2b2;}
.orgFrame ul
span.name{background:url(/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Exten
sions/QlikView/Examples/org/org_box.png) white no-repeat top
left;cursor:pointer;}
.orgFrame
a.moreInfo{overflow:hidden;height:9px;width:9px;position:absolute;background:
url(/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensions/QlikView/Exampl
es/org/more_info.png) white no-repeat top left;top:10px;left:117px;text-
indent:-9999px;cursor:pointer;}
.orgFrame li{display:inline-block;margin:0px auto;vertical-align:top;text-
align:center;font-size:11px; font-family:helvetica, arial, sans-serif;line-
height:26px;color:#333;padding-top:10px;position:relative;}
.orgFrame ul li ul
li{background:url(/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensions/Q
likView/Examples/org/connect.png) transparent no-repeat center
top;position:relative;}
.orgFrame ul li ul
li.orphan{background:url(/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Exten
sions/QlikView/Examples/org/connect_orphan.png) transparent no-repeat center
top;}
.orgFrame ul li
ul{background:url(/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensions/Q
likView/Examples/org/top_rule.png) transparent repeat-x center top;}
.orgFrame #org_fly{position:absolute;top:0px;left:0px;width:330px;z-
index:300000;display:none;font-size:12px; font-family:helvetica,arial,sans-
serif;text-align:left;}
.orgFrame #org_fly
.org_fly_top{width:330px;height:7px;overflow:hidden;background:url(/QvAjaxZfc
/QvsViewClient.aspx?public=only&name=Extensions/QlikView/Examples/org/org_fly
_top.png) transparent no-repeat top left;}
.orgFrame #org_fly
.org_fly_mid{width:330px;overflow:hidden;background:url(/QvAjaxZfc/QvsViewCli
ent.aspx?public=only&name=Extensions/QlikView/Examples/org/org_fly_mid.png)
transparent repeat-y top left;position:relative;}
.org_fly_mid img{padding:0px;margin:0px 12px 0px 12px;}
#org_fly h3,#org_fly p{font-size:12px;margin:0px;padding:0px 10px 5px;}
.org_fly_mid p{padding:0px;margin:5px 0px 0px;text-
align:left;color:#333;font-size:12px;font-family:helvetica, arial, sans-
serif;}
```

```
.orgFrame #org_fly
.org_fly_btm{width:330px;height:11px;overflow:hidden;background:url(/QvAjaxZf
c/QvsViewClient.aspx?public=only&name=Extensions/QlikView/Examples/org/org_fl
y_btm.png) transparent no-repeat top left;}
.orgFrame #org_fly
#fly_arrow{width:11px;height:27px;overflow:hidden;background:url(/QvAjaxZfc/Q
vsViewClient.aspx?public=only&name=Extensions/QlikView/Examples/org/org_fly_a
rrow.png) transparent no-repeat top left;position:absolute;left:-
9px;top:32px;}
.closeIt{position:absolute;right:10px;top:7px;display:block;height:9px;
width:9px;overflow:hidden;text-indent:-
9999px;background:url(/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensio
ns/QlikView/Examples/org/close.png) transparent no-repeat top left;z-
index:9999;cursor:pointer;}
.cover{display:block;position:absolute;width:68px;height:15px;background:whit
e;top:0px;}
.hiddenInfo{display:none !important;}
.rowNum{display:none !important;}
.tiptop{margin-top:40px !important;}
```

Appendix E - Pie Chart with Whiskers

Script.js

```
function pie_label_init(){
	Qva.LoadScript("/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensio
ns/QlikView/Examples/piechartlabel/excanvas.min.js", function() {  // FOR IE
Canvas support

	Qva.LoadScript("/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensio
ns/QlikView/Examples/piechartlabel/jquery.js", function() {

	Qva.LoadScript("/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensio
ns/QlikView/Examples/piechartlabel/jquery.jqplot.min.js", function() {
	//JQPlot pie charting library

	Qva.LoadScript("/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensio
ns/QlikView/Examples/piechartlabel/jqplot.pieRenderer.lineLabels.js",
pie_label_done); //Custom pie chart extension which adds whiskers
			});
		});
	});
}
function pie_label_done(){
	//Set up options for jqPlot pie chart
	var optionsObj = {
		//Create an array of colors to use for the pie chart
		seriesColors: ['#3399cc', '#cc6666', '#7ba550', '#ffcc66',
'#d17314', '#b974fd', '#993300', '#99cccc', '#669933', '#898989', '#eda1a1',
```

```
'#c6e2a9', '#d4b881', '#137d77', '#d7c2ec', '#ff5500', '#15dfdf', '#93a77e',
'#cb5090', '#bfbfbf'],
            grid: {
                  borderWidth: 0,
                  shadow: false
            },
            seriesDefaults: {
                  renderer: $.jqplot.PieRenderer,
                  //set the pie chart to have whiskers (or line labels) and
assign a color to them
                  rendererOptions: { lineLabels: true, lineLabelsLineColor:
'#777'},
                  shadow: false
            }
      }
      $(document).ready(function() {
      Qva.AddExtension('QlikView/Examples/piechartlabel', function() {

      Qva.LoadCSS("/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensions/
QlikView/Examples/piechartlabel/style.css");
            // Add jqplot specific CSS

      Qva.LoadCSS("/QvAjaxZfc/QvsViewClient.aspx?public=only&name=Extensions/
QlikView/Examples/piechartlabel/jquery.jqplot.min.css");
            var _this = this;
            // create random number to use in the div id to avoid conflicts
with multiple charts on the same page
            var r=Math.floor(Math.random()*101);
            var rDiv = "div" + r;
            var rC = "pieLabel" + r;
            if($('.pieLabel').length < 1){
                  _this.Element.className += " pL " + rC;
                  $('.pieLabel').text('what is up');
            }
            $('.' + rC).empty();
            line1 = [];
            //Loop through the data and create JSON that is used by jqPlot to
create the pie chart.
            for (var rowIx = 0; rowIx < _this.Data.Rows.length; rowIx++) {
                  line1.push([_this.Data.Rows[rowIx][0].text,
parseFloat(_this.Data.Rows[rowIx][1].text)]);
            }
            $('<div
/>').attr('id',rDiv).height(_this.GetHeight()).width(_this.GetWidth()).append
To('.' + rC);
            chart = $.jqplot(rDiv, [line1], optionsObj);
            $('.' + rC + ' *').unbind();  //for some reason an error was
appearing in IE on mouseover.  I've unbinded all events for the chart to
suppress it.
      });
      });
}
pie_label_init();
```

style.css

```
.pL{font-size:10px;}
```