```
function (scope,el,attrs,ctrl) {
 $(document).ready(function(){
  resize_window();
  totalQuota = ctrl.data.result.Total;
  //drawQuotaChart();
            drawQuotaChart(totalQuota);
 });

 $(window).resize(function(){
  resize_window();
 });


 function resize_window(){
  var panel_height = ($(document).height()- 415) + 'px';
  $(el).find('.panel-body').css('height',panel_height);
 }

 //function drawQuotaChart(){
 function drawQuotaChart(totalQuota) {
  var gaugeOptions = {

   accessibility: {
    enabled: true
   },

   chart: {
    type: 'solidgauge',
    description: 'Charts showing quota consumption for VM, vCPU,Storage Volume and Network
resources'
   },

   title: 'Quota Consumption Chart',

   pane: {
    center: ['50%', '80%'],
    size: '100%',
    startAngle: -90,
    endAngle: 90,
    background: {
     backgroundColor: (Highcharts.theme && Highcharts.theme.background2) ll '#EEE',
     innerRadius: '65%',
     outerRadius: '100%',
     shape: 'arc'
    }
   },

   tooltip: {
    enabled: false
   },

   // the value axis
   yAxis: {
```

```
    stops: [
     [0.1, '#55BF3B'], // green
     [0.5, '#DDDF0D'], // yellow
     [0.9, '#DF5353'] // red
    ],
    lineWidth: 0,
    tickPixelInterval: 400,
    minorTickInterval: null,
    tickAmount: 2,

    title: {
     y: 60
    },
    labels: {
     y: 15,
     distance: -13
    }
   },

   plotOptions: {
    solidgauge: {
     innerRadius: '65%',
     dataLabels: {
      y: 10,
      borderWidth: 0,
      useHTML: true
     }
    }
   }
  };

  // The gauge
 /*
            $('#container1').highcharts(Highcharts.merge(gaugeOptions, {
   yAxis: {
    labels: {
     formatter: function(){
      return this.value;
     }
    },
    min: 0,
    max: parseInt(totalQuota[1].User_Limit),
    tickPositioner:  function() {
     return [0,parseInt(totalQuota[1].User_Limit)];
    },
    title: {
     text: totalQuota[1].Type
    }
   },

   credits: {
    enabled: false
   },
```

```
 series: [{
  name: totalQuota[1].Type,
  data: [parseInt(totalQuota[1].User_Used)],
  dataLabels: {
   format: '<div style="text-align:center"><span style="font-size:14px;color:' +
   ((Highcharts.theme && Highcharts.theme.contrastTextColor) || 'black') + '">{y}</span><br/>' +
   '<span style="font-size:12px;color:silver">Used</span></div>'
  }
 }]
}));

$('#container2').highcharts(Highcharts.merge(gaugeOptions, {
 yAxis: {
  labels: {
   formatter: function(){
    return this.value;
   }
  },
  min: 0,
  max: parseInt(totalQuota[0].User_Limit) ,
  tickPositioner:  function() {
   return [0,parseInt(totalQuota[0].User_Limit)];
  },
  title: {
   text: totalQuota[0].Type
  }
 },

 credits: {
  enabled: false
 },

 series: [{
  name: totalQuota[0].Type,
  data: [parseInt(totalQuota[0].User_Used)],
  dataLabels: {
   format: '<div style="text-align:center"><span style="font-size:14px;color:' +
   ((Highcharts.theme && Highcharts.theme.contrastTextColor) || 'black') + '">{y}</span><br/>' +
   '<span style="font-size:12px;color:silver">Used</span></div>'
  }
 }]
}));

$('#container3').highcharts(Highcharts.merge(gaugeOptions, {
 yAxis: {
  labels: {
   formatter: function(){
    return this.value;
   }
  },
  min: 0,
  max: parseFloat(totalQuota[3].User_Limit),
```

```
 tickPositioner:  function() {
  return [0,parseFloat(totalQuota[3].User_Limit)];
 },
 title: {
  text: totalQuota[3].Type
 }
},

credits: {
 enabled: false
},

series: [{
 name: totalQuota[3].Type,
 data: [parseFloat(totalQuota[3].User_Used)],
 dataLabels: {
  format: '<div style="text-align:center"><span style="font-size:14px;color:' +
  ((Highcharts.theme && Highcharts.theme.contrastTextColor) || 'black') + '">{y}</span><br/>' +
  '<span style="font-size:12px;color:silver">GB Used</span></div>'
 }
}]
}));

$('#container4').highcharts(Highcharts.merge(gaugeOptions, {
yAxis: {
 labels: {
  formatter: function(){
   return this.value;
  }
 },
 min: 0,
 max: parseInt(totalQuota[2].User_Limit),
 tickPositioner:  function() {
  return [0,parseInt(totalQuota[2].User_Limit)];
 },
 title: {
  text: totalQuota[2].Type
 }
},

credits: {
 enabled: false
},

series: [{
 name: totalQuota[2].Type,
 data: [parseInt(totalQuota[2].User_Used)],
 dataLabels: {
  format: '<div style="text-align:center"><span style="font-size:14px;color:' +
  ((Highcharts.theme && Highcharts.theme.contrastTextColor) || 'black') + '">{y}</span><br/>' +
  '<span style="font-size:12px;color:silver">Used</span></div>'
 }
}]
```

```
}));

$('#container5').highcharts(Highcharts.merge(gaugeOptions, {
 yAxis: {
  labels: {
   formatter: function(){
    return this.value;
   }
  },
  min: 0,
  max: parseInt(totalQuota[4].User_Limit),
  tickPositioner:  function() {
   return [0,parseInt(totalQuota[4].User_Limit)];
  },
  title: {
   text: totalQuota[4].Type
  }
 },

 credits: {
  enabled: false
 },

 series: [{
  name: totalQuota[4].Type,
  data: [parseInt(totalQuota[4].User_Used)],
  dataLabels: {
   format: '<div style="text-align:center"><span style="font-size:14px;color:' +
   ((Highcharts.theme && Highcharts.theme.contrastTextColor) || 'black') + '">{y}</span><br/>' +
   '<span style="font-size:12px;color:silver">Used</span></div>'
  }
 }]
}));
          */

for (var i in totalQuota) {
          //for (var i=0; i < 6; i++) {
           $('#container'+i).highcharts(Highcharts.merge(gaugeOptions, {
 yAxis: {
  labels: {
   formatter: function(){
    return this.value;
   }
  },
  min: 0,
  max: parseInt(totalQuota[i].User_Limit),
  tickPositioner:  function() {
   return [0,parseInt(totalQuota[i].User_Limit)];
  },
  title: {
   text: totalQuota[i].Type
  }
 },
```

```
   credits: {
    enabled: false
   },

   series: [{
    name: totalQuota[i].Type,
    data: [parseInt(totalQuota[i].User_Used)],
    dataLabels: {
     format: '<div style="text-align:center"><span style="font-size:14px;color:' +
     ((Highcharts.theme && Highcharts.theme.contrastTextColor) || 'black') + '">{y}</span><br/>' +
     '<span style="font-size:12px;color:silver">Used</span></div>'
    }
   }]
  }));
            }

 }
}
```