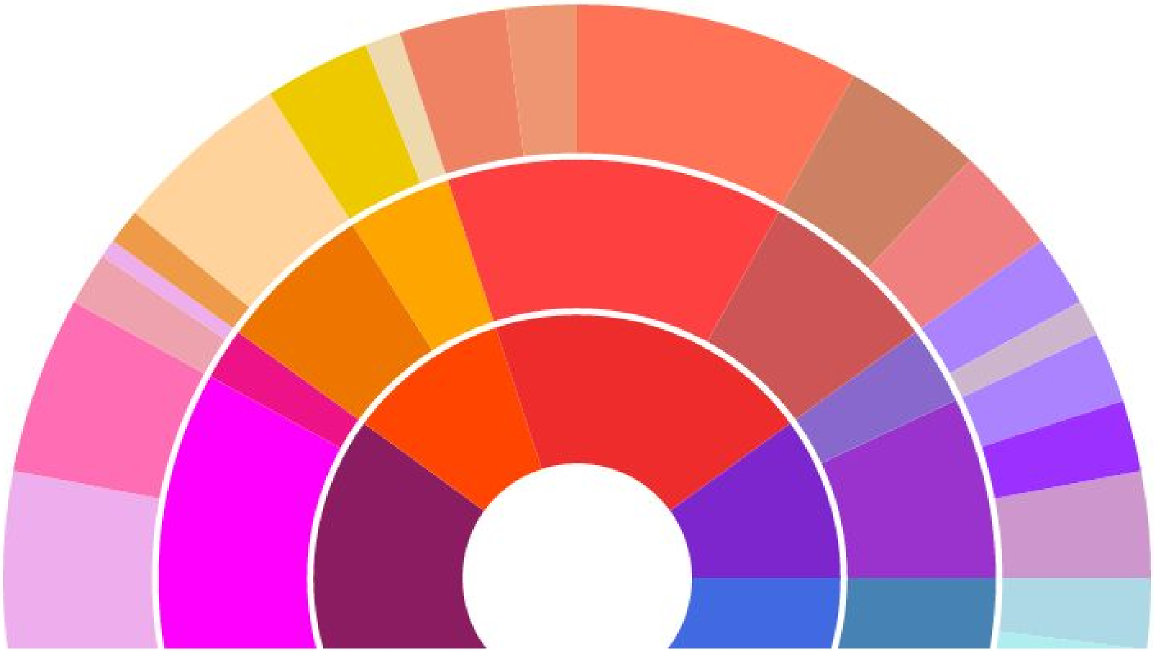
HTML5 Charting: Multi-Series Pie Charts

As you already know, Pie Charts are a great way to convey information to users and quickly analyze the “biggest|smallest” share of total data under review.

The popularity of traditional Pie Charts cannot be understated nor can its one major limitation – the inability to display multiple series. This limitation, can only be addressed via rings and with our last major release (v15.2), the DevExtreme HTML5 Pie Chart includes full support for multiple series (rings) within a single chart.

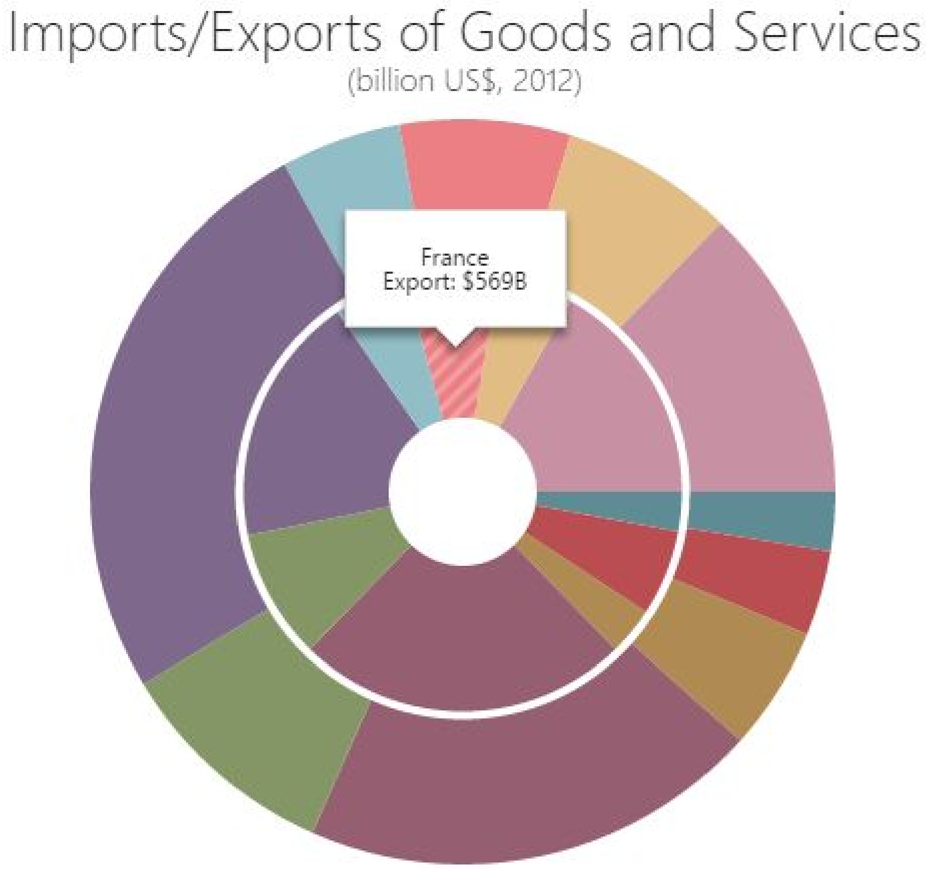


A Ring to Rule Them All

Because a Pie Chart is circular and can only display a single series at any point in time, the only practical way to display multiple series is to use rings within the pie itself.

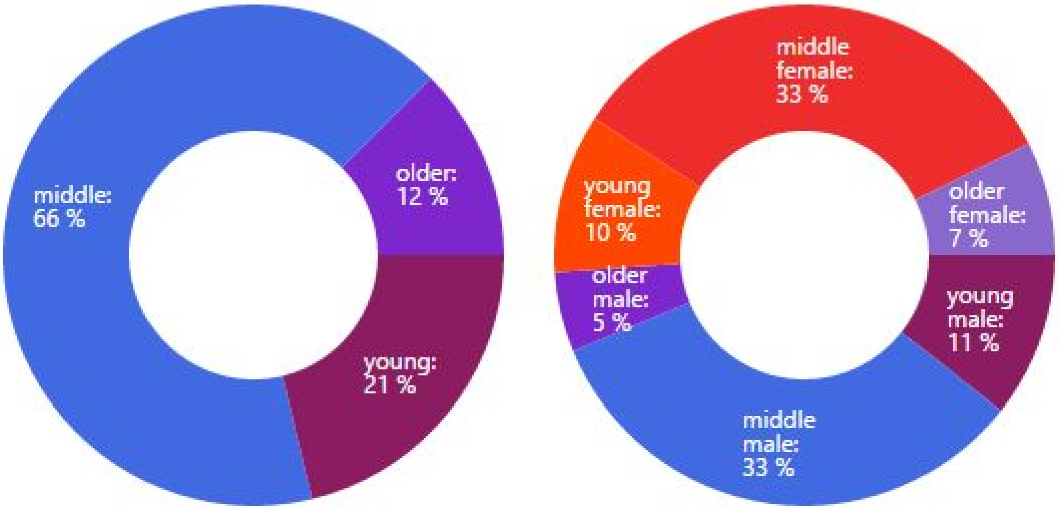
Before we describe the value of Pie Chart rings in the context of multiple data series, let’s take a quick look at its use in a standard Pie Chart.

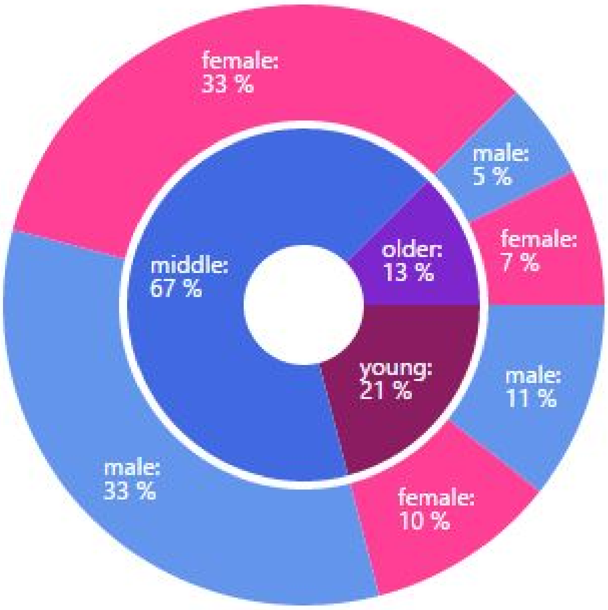
For utmost clarity, the chart below displays US imports and exports per country on both the pie’s inner and outer rings. By hovering over a section, tooltips provide additional information.



Multi-Series Pie Chart

As you build your application, you will invariably encounter issues with available screen real estate and use of multiple “single ring” Pie Charts may be impossible to reconcile. When limited in this manner, the display of multiple rings within a single Pie Chart offers more options to you and your UX team.

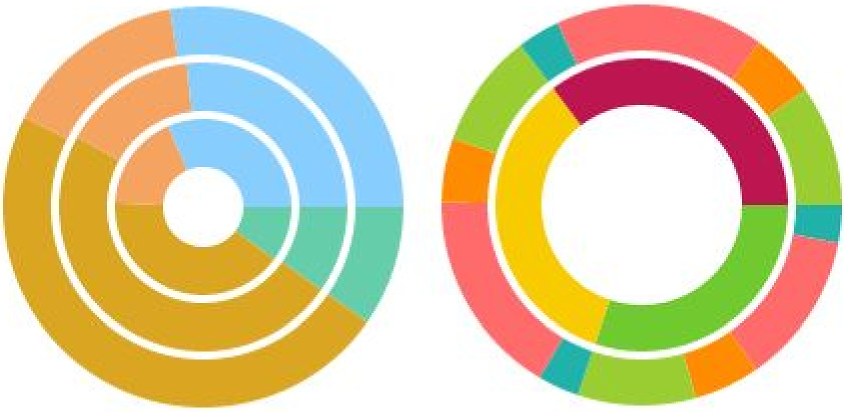




As you can see in the screenshots above, a multi-ring Pie Chart can easily convey the same information to end-users while minimizing use of screen space.

DevExtreme’s HTML5 Pie Chart can present data in two unique ways when using multiple rings:

- Each ring can represent an independent series - useful when presenting equivalent data.
- A ring detail segment is displayed on another ring. This chart type is often called a “sunburst” and is very helpful to those who must present composite data.



How it Works

To get started and introduce multi-ring Pie Charts to your web or mobile app, simple configure the appropriate DevExtreme Pie Chart options as described below.

In this example (product export volume), we'll first create a widget inside an HTML container, specify its size and bind it to data.

```
$("#container").dxPieChart({
        size: {
                width: 350,
                height: 400
        },
        dataSource: [{
        arg: "fish",
       2001: 200,
       2002: 220,
       2003: 225
   }, {
        arg: "meat",
       2001: 800,
       2002: 1090,
       2003: 1100
   }, {
        arg: "fruits",
       2001: 350,
       2002: 340,
       2003: 345
   }, {
        arg: "vegetables",
       2001: 620,
       2002: 605,
       2003: 638
   }]
});
```

At this stage, the widget is empty because the series has yet to be specified. Since our data contains information about product export volume over a three year period, we'll display each year within a separate ring (specify a separate series for each year).

```
$("#container").dxPieChart({
        //…,
        series: [{
                name: "2001",
            valueField: "2001"
         }, {
                name: "2002",
            valueField: "2002"
         }, {
                name: "2003",
            valueField: "2003"
         }]
});
```

By default, the inner series is displayed as a circle. To display it as a ring, we need to set the Pie Chart to "doughnut" and specify its inner radius.

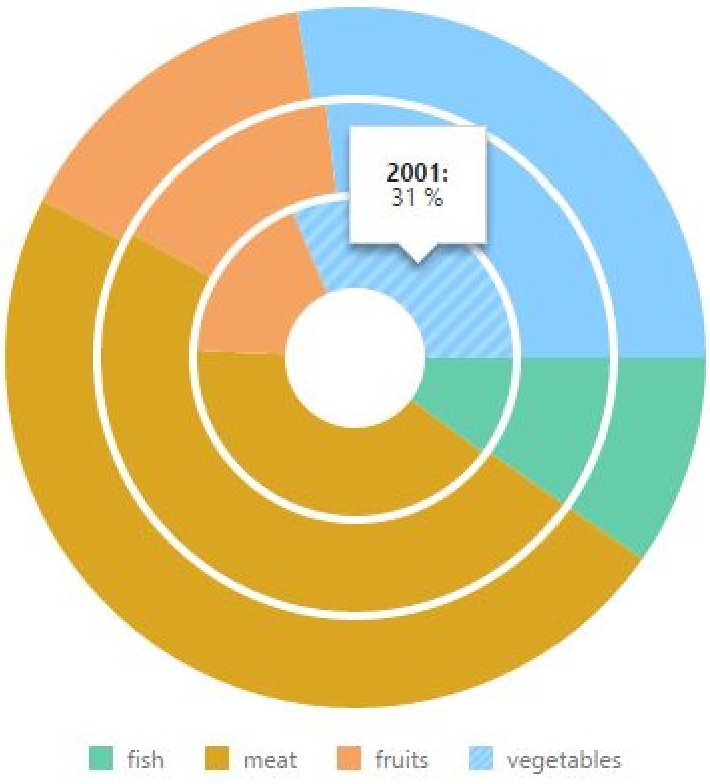
```
$("#container").dxPieChart({
        //…,
        type: "doughnut",
        innerRadius: 0.2
});
```

To boost data clarity, we'll define a unique color for each point in the series using the Chart's **palette** option. Simultaneously, we'll specify position settings for text and other chart elements.

```
$("#container").dxPieChart({
        //…,
        palette: ["#66CDAA", "#DAA520", "#F4A460", "#87CEFF"],
        legend: {
                horizontalAlignment: "center",
      verticalAlignment: "bottom",
      itemTextPosition: "right"
    }
});
```

To effectively engage the end-user when using our app, we’ll add an interactive tooltip that’s displayed whenever a pie segment is highlighted. This is a simple matter of activating the Pie Chart’s tooltip and specifying the text to use. In this example, we’ll display the series name and a percentage.

```
$("#container").dxPieChart({
    //…,
    tooltip: {
        enabled: true,
      customizeTooltip: function (arg) {
            return {
            text: "<b>" + arg.seriesName + ":</b>\n" + arg.percentText
        }
      }
    }
});
```



In this example, we'll show you the second example, let's begin by first creating the widget, specifying its size, and then binding the given data. As the current example does not require the legend, we'll make it invisible.

```
$("#container").dxPieChart({
        dataSource: [{
                arg: "2001",
                val: 1970
        }, {
                arg: "2001-fish",
           product:  200
        }, {
                arg: "2001-meat",
           product: 800
        }, {
                arg: "2001-fruits",
           product: 350
        }, {
                arg: "2001-vegetables",
           product: 620
        }, {
                arg: "2002",
           val: 2255
        }, {
                arg: "2002-fish",
           product: 220
        }, {
                arg: "2002-meat",
           product: 1090
        }, {
                arg: "2002-fruits",
           product: 340
        }, {
                arg: "2002-vegetables",
           product: 605
        }, {
                arg: "2003",
           val: 2308
        }, {
                arg: "2003-fish",
           product: 225
        }, {
                arg: "2003-meat",
           product: 1100
        }, {
                arg: "2003-fruits",
           product: 345
        }, {
                arg: "2003-vegetables",
           product: 638
        }],
        size: {
```

```
                width: 350,
            height: 400
        },
        legend: {
                visible: false
        }
});
```

Much like our first example, our chart is empty because our series has yet to be specified. The data in this example contains information about product export volume for the last three years and the total volume for each year. The inner ring will display total year volume and the outer ring will detail each sector of the inner ring - displaying annual volume by products. We’ll start by configuring the series array:

```
$("#container").dxPieChart({
        //…,
        series: [{
                //let's customize it later
        }, {
                valueField: "product"
        }]
});
```

To make the chart more informative, we’ll enable and customize labels for the inner ring.

```
$("#container").dxPieChart({
        //…,
        series: [{
                label: {
                visible: true,
            position: "inside",
            backgroundColor: "none",
            customizeText: function (arg) {
                        return "<b>" + arg.argument + "</b>:\n" + arg.percentText;
            }
        }
        }, {
                valueField: “product”
        }]
});
```

Next, we’ll set chart type to “doughnut”.

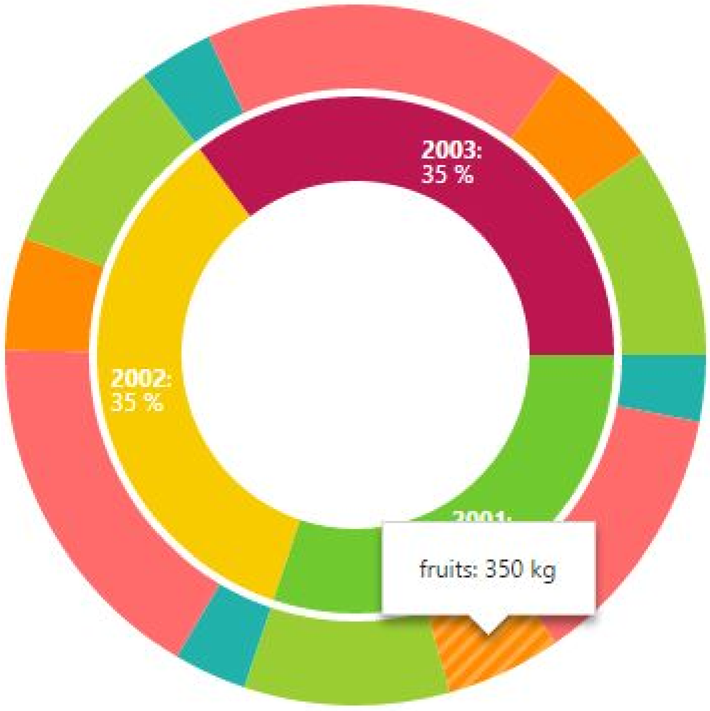
```
$("#container").dxPieChart({
        //…,
        type: "doughnut"
});
```

Though we can specify colors for inner ring points using the **palette** option, outer ring point colors must be specified using the **customizePoint** option.

```
$("#container").dxPieChart({
        //…,
        palette: "bright",
    customizePoint: function(arg) {
        var product = arg.argument.split("-")[1];
        if (product === "fish") {
                return { color: "#20B2AA" };
        } else if (product === "meat") {
                return { color: "#FF6A6A" };
        } else if (product === "fruits") {
                return { color: "#FF8C00" };
        } else if (product === "vegetables") {
                return { color: "#9ACD32" };
        }
    }
});
```

For more granular control over information available to end-users, we’ll customize tooltip settings so that the inner ring displays year and percentage and the outer ring displays product and percentage.

```
$("#container").dxPieChart({
        //…,
        tooltip: {
        enabled: true,
        customizeTooltip: function (arg) {
                var product = arg.argument.split("-")[1];
            if (product !== undefined) {
                return {
                        text: product + ": " + arg.valueText + " kg"
                }
            } else {
                return {
                                text: arg.argumentText + ": " + arg.valueText + " kg"
                }
            }
        }
    }
});
```



There you have it – a few simple steps are all you’ll need to follow in order to create high-impact multi-series Pie Charts with DevExtreme.

We’d love to hear what you think about our HTML5 Pie Chart and use of rings. Feel free to contact us with feedback or questions – we are here to help.