Creating Intuitive & Interactive Dashboards with the ADF Data Visualization Components

Frank Houweling
UKOUG 2014
Agenda

• Why data visualization is important
• Examples where DVTs are used
• Graph demo: ADF Performance Monitor
• Basic steps creating a graph (ADF11g)
• Special features
  – colors, mouseover info, alerts, reference line, animation 3D, clicklistener, hide and show, stacked graphs, dual graphs, e.g.
• Advanced Graph Examples
  – Bubble, Spark, Treemap
• Other Tips & Challenges
• 12.1.3 DVT Components
Why data visualization is important

- Use of the power of visualization to present information
- Call to action to our end user instead of showing raw data - as we frequently do today
- Visualizations can be used to help end users focus on what is relevant: aggregate, exception, trend, comparison etc.
Glance, Scan, Commit

• Oracle Alta UI
• Oracle follows the “glance, scan, commit” design philosophy
  – start in an overview/dashboard mode with small information containers
  – These containers show only key information for the user to glance
  – If the user is interested in anything he can zoom in on the subject (Scan the information)
  – When the user needs to complete a task he navigates to the commit mode
Oracle Alta

• Work Better Application
Fusion Applications
Dashboard interactive graphs

- Dashboard of financial management application for local governments
WLS Enterprise Manager
• Database EM
Demo ADF Performance Monitor Dashboard
Demo ADF Performance Monitor
Errors Overview
– In this production ADF app, for a single ViewObject instance, more than **900,000** rows were loaded in the ADF app (Blue)!
Demo ADF Performance Monitor
Slow Activation of Application Module
Demo ADF Performance Monitor
Slow Passivation of ApplicationModule
Demo ADF Performance Monitor
Worst ApplicationModule activations & passivations overview

- Monitor aggregates of ApplicationModule activations and passivations
  - AVG, Occurrences, Total Exec Time, Min, Max
  - Zoom in into single executions
Demo ADF Performance Monitor
Worst ApplicationModule activations & passivations overview

![ADF Performance Monitor](image_url)
Demo ADF Performance Monitor
Occurrences overview
Agenda

• Why data visualization is important
• Examples where DVTs are used
• Graph demo: ADF Performance Monitor
• **Basic steps creating a graph (ADF11g)**
• Special features
  – colors, mouseover info, alerts, reference line, animation 3D, clicklistener, hide and show, stacked graphs, dual graphs, e.g.
• Advanced Graph Examples
  – Gauges, Bubble, Spark, Treemap
• Other Tips & Challenges
• 12.1.3 DVT Components
Basic Steps Creating a Graph - 1

- Create a ADFBC ViewObject with raw graph data
- Drag and drop the ViewObject from the DataControl to the page
• Select Category and Graph Type
• Select X and Y attributes
Basic Steps Creating a Graph - 4

- Generated code in page

```xml
<graph IterBinding="ActiveRequestsAMinuteIterator"
       id="ActiveRequestsAMinuteV07"
       xmlns="http://xmlns.oracle.com/adfm/dvt" type="LINE_VERT_ABS">
  <graphDataMap leafOnly="true">
    <series>
      <data>
        <item value="JvmHeap"/>
      </data>
    </series>
    <groups>
      <item value="RequestTime"/>
    </groups>
  </graphDataMap>
</graph>
```

- Generated code in PageDef

```xml
<graph IterBinding="ActiveRequestsAMinuteIterator"
       id="ActiveRequestsAMinuteV07"
       xmlns="http://xmlns.oracle.com/adfm/dvt" type="LINE_VERT_ABS">
  <graphDataMap leafOnly="true">
    <series>
      <data>
        <item value="JvmHeap"/>
      </data>
    </series>
    <groups>
      <item value="RequestTime"/>
    </groups>
  </graphDataMap>
</graph>
```
Basic Steps Creating a Graph - 5

- Result:
• Add attribute `dynamicResize="DYNAMIC_SIZE"` to `dvt:lineGraph`:
Basic Steps Creating a Graph - 8

- Colors are defined in `<dvt:seriesSet>`
  - Add a line color
  - Add a line width
  - Add a marker type
Basic Steps Creating a Graph - 9

- Position the legend

```xml
<dvt:lineGraph id="lineGraph1" shortDesc="JVM Heapspace" dynamicResize="DYNAMIC_SIZE"
    rendered="#{pageFlowScope.MainDashBoardBean.timeRange == '5 MINUTES'}"
    value="#{bindings.ActiveRequestsAMinuteV07.graphModel}"
    subType="LINE_VERT_ABS">
    <dvt:background />
    <dvt:specialEffects />
    </dvt:background>
    <dvt:graphPlotArea />
    <dvt:seriesSet>
        <dvt:series color="#695BA7" lineWidth="1" markerType="MT_MARKER" />
    </dvt:seriesSet>
    <dvt:01Axis />
    <dvt:y1Axis />
    <dvt:legendArea automaticPlacement="AP_NEVER" position="LAP BOTTOM" />
    <dvt:x1Axis />
</dvt:lineGraph>
```
Basic Steps Creating a Graph - 10

- In PageDefinition
  - Change Legend text
  - Change mouseover attribute labels

```
<graph IterBinding="ActiveRequestsAMinuteIterator"
     id="ActiveRequestsAMinuteVO7"
     xmlns="http://xmlns.oracle.com/adf/dvt" type="LINE_VERT_ABS">
    <graphDataMap leafOnly="true">
        <series>
            <data>
                <item value="JvmHeap" label="JVM Heap Usage"/>
            </data>
        </series>
        <groups>
            <item value="RequestTime" itemLabel="Request Time"/>
        </groups>
    </graphDataMap>
</graph>
```
Basic Steps Creating a Graph - 11

- Add Titles
• Show Time instead of Date on X-axis
Basic Steps Creating a Graph - 13

- 1. Hide and Show Behavior
- 2. Click on legend item
- 3. Graph is rescaled!
• Just add

```html
hideAndShowBehavior="withRescale"
```
Agenda slide

- Why data visualization is important
- Examples where DVTs are used
- Graph demo: ADF Performance Monitor
- Basic steps creating a graph (ADF11g)
- **Special features**
  - colors, mouseover info, alerts, reference line, animation 3D, clicklistener, hide and show, stacked graphs, dual graphs, e.g.
- Advanced Graph Examples
  - Gauges, Bubble, Spark, Treemap
- Other Tips & Challenges
- 12.1.3 DVT Components
• Combination Graph

```xml
<dvt:comboGraph id="comboGraph1" contentDelivery="immediate"
    shortDesc="JVM Heapspace" dynamicResize="DYNAMIC_SIZE"
    hideAndShowBehavior="withRescale"
    value="#{bindings.ActiveRequestsAMinute06.graphModel}" subtype="COMBINATION_VERT_ABS_2Y">
    <dvt:series color="#695BA7" lineWidth="1" markerType="MT_DEFAULT"/>
    <dvt:series color="#F05FA3" markerType="MT_BAR"/>
    <dvt:yTitle text="JVM Heap (MB)"/>
    <dvt:legendArea automaticPlacement="AP_NEVER" position="LAP_BOTTOM"/>
</dvt:comboGraph>
</graphDataMap>
```
Special Features – 2 (Alerts)

• Alerts
  – Warn, alert user for something
  – Show custom image
  – Add mouse over message
Special Features – 2 (Alerts)

```xml
<dvt:comboGraph id="lineGraph5" styleClass="AFStretchWidth"
  contentDelivery="immediate"
  rendered="#{pageFlowScope.MainDashBoardBean.timeRange=='WEEK' || page1 hideAndShowBehavior="withRescale"
  dynamicResize="DYNAMIC_SIZE"
  shortDesc="JVM Heap and Garbage Collection"
  value="#{bindings.JVMHeapADayVO.graphModel}"
  subType="COMBINATION_VERT_ABS_2Y">
  <dvt:alertSet alertMap="#{pageFlowScope.MainDashBoardBean.alertMapJVMHeapADay}"
    id="vgjklk"/>
    <dvt:background>
      <dvt:specialEffects/>
    </dvt:background>
    <dvt:graphPlotArea/>
    <dvt:seriesSet>
      <dvt:series color="#695BA7" lineWidth="1" markerType="MT_MARKER"/>
      <dvt:series color="#F05FA3" lineWidth="1" markerType="MT_BAR"/>
    </dvt:seriesSet>
    <dvt:y1Axis axisMinValue="0" axisMinAutoScaled="false"/>
    <dvt:y2Axis axisMinValue="0" axisMinAutoScaled="false"/>
    <dvt:legendArea automaticPlacement="AP_NEVER"
      position="LAP_BOTTOM"/>
    <dvt:y1Title text="JVM Heap (MB)"/>
    <dvt:y2Title text="Garbage Collection Time (sec)"/>
  </dvt:comboGraph>
```
public Map getAlertMap(VMHheapAkey) {

    int slowGBCollection = 15; //static threshold of 15 seconds for very slow JVM garbage collections
    DCIteratorBinding iteratorBinding = ADFModel.findIterator("JVMHeapAkeyV0Iterator");
    ViewObject viewObject = iteratorBinding.getViewObject();
    Map alertMap = new HashMap();
    if (viewObject != null) {

        //This construction with creating a new rowSetIteratoris needed to not skip the first row.
        RowSetIterator rowSetIterator = viewObject.createRowSetIterator(null);
        rowSetIterator.reset();
        int index = 0;

        while (rowSetIterator.hasNext()) {
            Row row = rowSetIterator.next();
            BigDecimal gbTime = (BigDecimal)row.getAttribute("GBTime");
            String label = (String)row.getAttribute("label");
            if (gbTime.doubleValue() > slowGBCollection) {
                Alert alert = new Alert();
                //X value of alert
                alert.setXValue(label);
                //Y value is plotted against the Y1 or Y2-axis.
                alert.setYValueAssignment(1);
                //Y value od alert
                alert.setYValue(gbTime.doubleValue() + 2);
                //set custom image
                alert.setImageSource("/images/warningred.png");
                //mouseover text in alert
                alert.setText("WARNING: very long running garbage collection: " + gbTime.doubleValue() + " sec. inTime: " + label);
                alertMap.put(new Integer(index), alert);
                index++;
                System.out.println("ALERT ADDED.. gbTime: " + gbTime);
            }
        }
    }

    return alertMap;
}
Special Features – 3
Reference Line

- Reference Line
  - JVM Garbage Collection is considered as very long running when > 20 Seconds
Special Features – 3 Reference Line

- You can make the reference line dynamic with EL expression

```xml
<dvt:comboGraph id="lineGraph5" contentDelivery="immediate"
    rendered="#{pageFlowScope.MainDashBoardBean.timeRange=='WEEK' || pageFlowScope.hideAndShowBehavior=='withRescale'
    dynamicResize='DYNAMIC_SIZE'
    shortDesc='JVM Heap and Garbage Collection'
    value="#{bindings.JVMHeapADayVO.graphModel}"
    subType='COMBINATION_VERT_ABS_2Y'>
    <dvt:referenceObjectSet>
        <dvt:referenceObject lineWidth='1'
            lineValue="#{pageFlowScope.MainDashBoardBean.longGarbageCollection}" association='Y2AXIS' color='"#ff4242"
            displayedInLegend='true'
            text='Reference Line: Long Running Garbage Collection'
            lineStyle='LS_DASH'/>
    </dvt:referenceObjectSet>
    <dvt:alertSet alertMap="#{pageFlowScope.MainDashBoardBean.alertMapJVMHeapADay}" id="vgjklk"/>
</dvt:comboGraph>
```
Special Features – 4
Custom Label on Axis

- Custom Label on X-axis
  - Transient attributes are useful
Special Features – 4
Custom Label on Axis

- Steps custom label on axis:
  - Create a Transient attribute
  - Generate the `ViewRowImpl`
  - Add your custom code

```xml
<graph IterBinding="SummaryRequestAnHourBarV0Iterator"
  id="SummaryRequestAnHourBarV0" xmlns="http://xmlns.oracle.com/adm/dvt" type="BAR_VERT_STACK">
  <graphDataMap leafOnly="true">
    <series>
      <data>
        <item value="SumNormal" label="Normal Requests"/>
        <item value="SumSlow" label="Slow Requests"/>
        <item value="SumVerySlow" label="Very Slow Requests"/>
      </data>
    </series>
    <groups>
      <item value="HourOfDay" label="HourOfDayLabel" itemLabel="Hour"/>
    </groups>
  </graphDataMap>
</graph>

/**
 * Return CUSTOM X-axis label for hour of day
 * Return label for every 2 hours
 */

public String getHourOfDayLabel() {
  //return (String) getAttributeInternal(HOUROFDAYLABEL);
  int value = getHourOfDay().intValue();
  int a = value%2;
  if(a==0)
    return ""+(value<10?"0":"")+getHourOfDay().intValue()+"h";
  return "";
}
Special Features – 4
Custom Label on Axis

- Set the VO transient attribute as value to the label attribute of the group item

```xml
<graph IterBinding="SummaryRequestAnHourBarVOIterator"
      id="SummaryRequestAnHourBarVO"
      xmlns="http://xmlns.oracle.com/adfm/dvt" type="BAR_VERT_STACK">
  <graphDataMap leafOnly="true">
    <series>
      <data>
        <item value="SumNormal" label="Normal Requests"/>
        <item value="SumSlow" label="Slow Requests"/>
        <item value="SumVerySlow" label="Very Slow Requests"/>
      </data>
    </series>
    <groups>
      <item value="HourOfDay" label="HourOfDayLabel" itemLabel="Hour"/>
    </groups>
  </graphDataMap>
</graph>
```
Special Features - 5

• Three D effect
• Be careful; does it add value to end-users?
• Animation
• Be careful; does it add value to end-users?
Special Features – 7
Make the graph interactive

- Click on graph and zoom in from Day to Hour Overview
Special Features – 7
Make the graph interactive

- Zoom in from Hour to Five Minute Overview
Special Features – 7
Make the graph interactive
Special Features – 7
Make the graph interactive

- Add a clickListener

```xml
<dvt:barGraph id="barGraph5" styleClass="AFStretchWidth"
  contentDelivery="immediate"
  rendered="#{pageFlowScope.MainDashBoardBean.timeRange=='HOUR'}"
  inlineStyle="width:300px; height:80.0px;"
  clickListener="#{pageFlowScope.MainDashBoardBean.onClickActiveSessionsrequestAMinute}"
  hideAndShowBehavior="withRescale"
  imageFormat="PNG" dynamicResize="DYNAMIC_SIZE"
  shortDesc="Response Time"
  value="#{bindings.SummaryRequestsFiveMinutesBarV0.graphModel}"
  subType="BAR_VERT_STACK">
</dvt:backgrounds>

```javacode```

```java
public void onClickActiveSessionsrequestAMinute(ClickEvent clickEvent) {

  ComponentHandle handle = clickEvent.getComponentHandle();
  ComponentInfo componentInfo = handle.getComponentInfo();
  timeRange = "5 MINUTES";

  if (componentInfo instanceof oracle.dss.util.DataComponentInfo) {
      DataComponentInfo dataComponentInfo = (DataComponentInfo)componentInfo;
      minute = dataComponentInfo.getColumn();
  }
  refresh();
}
```
• Why data visualization is important
• Examples where DVTs are used
• Graph demo: ADF Performance Monitor
• Basic steps creating a graph (ADF11g)
• Special features
  – colors, mouseover info, alerts, reference line, animation 3D, clicklistener, hide and show, stacked graphs, dual graphs, e.g.
• **Advanced Graph Examples**
  – Gauges, Bubble, Spark, Treemap
• Other Tips & Challenges
• 12.1.3 DVT Components
Advanced Graph Example 1: Bubble Graph

- Displays three dimensions of data; x, y and z value; z value is the size of the bubble
- Can facilitate the understanding of social, economical, and other relationships
Advanced Graph Example 1: Bubble Graph

- See complete example at AMIS blog at:
  - Life Expectancy by Country
  - Life expectancy (y-axis), income a year (x-axis) and the population (bubble size) is shown.
  - In steps of 10 years, for the last 50 years (1970, 1980, 1990, 20000 and 2010).
Advanced Graph Example 1: Bubble Graph

- ClickListener again to make it interactive and to navigate to detail graph
public void onClick(ClickEvent clickEvent) {
    ComponentHandle componentHandle = clickEvent.getComponentHandle();
    if (componentHandle instanceof DataComponentHandle) {
        DataComponentHandle handle = (DataComponentHandle) componentHandle;
        Attributes[] attr = (handle).getGroupAttributes();
        for (int i = 0; i < attr.length; i++) {
            Attributes a = attr[i];
            Object obj = a.getValue(a.ID_ATTRIBUTE);
            // Check if we have our ViewObject ID attribute
            if ("Country".equals(obj.toString())) {
                selectedCountry = (String)a.getValue(a.ID_VALUE);
                System.out.println("selectedCountry=\" + selectedCountry);
            }
        }
    }
    // Do something interesting with selectedCountry
    BindingContainer bindingContainer = BindingContext.getCurrent().getCurrentBindingsEntry();
    OperationBinding operationBinding = bindingContainer.getOperationBinding("ExecuteWithParamsSelect");
    operationBinding.getParamsMap().put("p_year", null);
    operationBinding.getParamsMap().put("p_country", selectedCountry);
    operationBinding.execute();
    renderDetailGraph = true;
    renderGraph = false;
    AdfFacesContext.getCurrentInstance().addPartialTarget(panelStretch);
}
Advanced Graph Example 2: Spark Graph

- See complete example at AMIS blog at:

<table>
<thead>
<tr>
<th>Ranking</th>
<th>RiderNumber</th>
<th>Name</th>
<th>Team</th>
<th>Gap</th>
<th>Overall Rank throughout Stages</th>
<th>Gap with Cadel Evans through Stages</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>141</td>
<td>EVANS Cadel</td>
<td>BMC RACING TEAM</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>11</td>
<td>SCHLECK Andy</td>
<td>TEAM LEOPARD-TREK</td>
<td>+ 01' 34''</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>18</td>
<td>SCHLECK Frank</td>
<td>TEAM LEOPARD-TREK</td>
<td>+ 02' 30''</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>181</td>
<td>VOECKLER Thomas</td>
<td>TEAM EUROP CAR</td>
<td>+ 03' 20''</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>CONTADOR Alberto</td>
<td>SAXO BANK SUNGARI</td>
<td>+ 03' 57''</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>21</td>
<td>SANCHEZ Samuel</td>
<td>EUSKALTEL - EUSKAD</td>
<td>+ 04' 55''</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>161</td>
<td>CUNEGO Damiano</td>
<td>LAMPRE - ISD</td>
<td>+ 06' 05''</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>91</td>
<td>BASSO Ivan</td>
<td>LIQUIGAS-CANNOND</td>
<td>+ 07' 23''</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>52</td>
<td>DANIELSON Tom</td>
<td>TEAM GARMIN - Cervin</td>
<td>+ 08' 15''</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>108</td>
<td>PERAUD Jean-Christel</td>
<td>AG2R LA MONDIALE</td>
<td>+ 10' 11''</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Advanced Graph Example 3: TreeMap

- Total execution time of worst ADF BC performing executions
  - The performance impact of frequent invoked executions can be much higher than executions that occur only a few times but are very slow on average
Advanced Graph Example 3: TreeMap

- Set the label (text) and value attribute (size on map)

```xml
<dvt:treemap id="t5" styleClass="AFStretchWidth"
    rendered=""{(pageFlowScope.DatabaseExecutionsBean.totalTimeGraph=='treemap')"
    contentDelivery="lazy"
    inlineStyle="height:500px;"
    value=""{(bindings.DatabaseBottlenecksWeightedV02.treeModel)"
    var="row">
    <af:switcher facetName=""{(row.hierTypeBinding.name)"
        id="sll">
        <f:facet name="DatabaseBottlenecksWeightedV020">
            <dvt:treemapNode value=""{(row.WeightedTime)"
                label=""{(row.NrlQuery)" id="tnl">
                <dvt:attributeGroups value=""{(row.NrlQuery)"
                    type="color" id="agl"/>
            </dvt:treemapNode>
        </f:facet>
    </af:switcher>
</dvt:treemap>
```
Agenda

• Why data visualization is important
• Examples where DVTs are used
• Graph demo: ADF Performance Monitor
• Basic steps creating a graph (ADF11g)
• Special features
  – colors, mouseover info, alerts, reference line, animation 3D, clicklistener, hide and show, stacked graphs, dual graphs, e.g.
• Advanced Graph Examples
  – Gauges, Bubble, Spark, Treemap
• Other Tips & Challenges
• 12.1.3 DVT Components
Colors

- Use a color palette to make look and feel of graphs consistent

**Color Palette**
This palette consists of 1 neutral and 7 key colors. Use key colors to draw the eye to important data or facts.

- Neutral
- Green
- Teal
- Blue
- Yellow
- Orange
- Red
- Violet
• Number Formatting a bit difficult
• Date/Time on X-axis a bit difficult
• Axis attributes `minorIncrement` and `majorIncrement` does not always seem to work

```xml
<dvt:x1Axis axisMinValue="0" axisMaxValue="23" minorIncrement="2"
    majorIncrement="2" axisMinAutoScaled="false"
    axisMaxAutoScaled="false"/>
```

• Area graph
  – A bit hard too click on data point
• Why data visualization is important
• Examples where DVTs are used
• Graph demo: ADF Performance Monitor
• Basic steps creating a graph (ADF11g)
• Special features
  – colors, mouseover info, alerts, reference line, animation 3D, clicklistener, hide and show, stacked graphs, dual graphs, e.g.
• Advanced Graph Examples
  – Gauges, Bubble, Spark, Treemap
• Other Tips & Challenges
• 12.1.3 DVT Components
ADF 12.1.3. Data Visualization Components

- Since ADF11g R1 (2008)
- Since ADF 12.1.3 (2014)
- Over 30+ (improved) chart types
- Mobile first design/touch support
- Improved Server Scalability
- Redesigned zoom and scroll
- Much Simpler Styling
New Gauges

- **Gauge types**
  - LED
  - status meter
  - Dial
  - new rating gauge
  - Supports input (Rating only)
New Designed Zoom and Scroll
Improved Time axis

- **Time Axis Nested Labels: Year and Month**
- **Time Axis with Irregular Intervals**
- **Mixed Frequency Time Axis**
Marquee zoom and selection
Time Line

- Visualize Events on an Interactive Timeline
- Support for time duration (yet MAF only, ADF will have it soon as well)
Thematic Map - New Features

- Image markers (option to set the orientation)
- Smart data zooming
- Drill down
- Support for custom base maps
- Hide area layer
- Marker locations can be updated
New Visualization: Diagram
New Visualization: Nbox (Yet MAF only)

- Visualizes and compares data across two dimensions
- Supports various options for color coding, marker shapes, and grouping
ADF Faces Component Demo

Graph and Chart: Migration Strategy

• Going forward, all the new features are introduced in DVT Charts
• Old Graph and Gauge tags deprecated in cases where the new tags are available
• Editing existing data-bound instances of deprecated tags is still supported
Challenges

• Data integrity (extremely max values or extremely low values)
• Sometimes better AVG values to get more insight
• Do not use too many data points
• Make graphs understandable; tell the end-user what is showed
  – clear legend
  – mouseover message
  – set X and Y title
• Keep it simple (!)
Learn More

- [http://technology.amis.nl/](http://technology.amis.nl/)