

# RichFaces 4 Webinar Series

*Webinar #2:*

RichFaces 3 to RichFaces 4 —  
A Class in Moving Up

May 11, 2011

Max Katz

Charley Cowens

<http://exadel.com/web/portal/webinars>

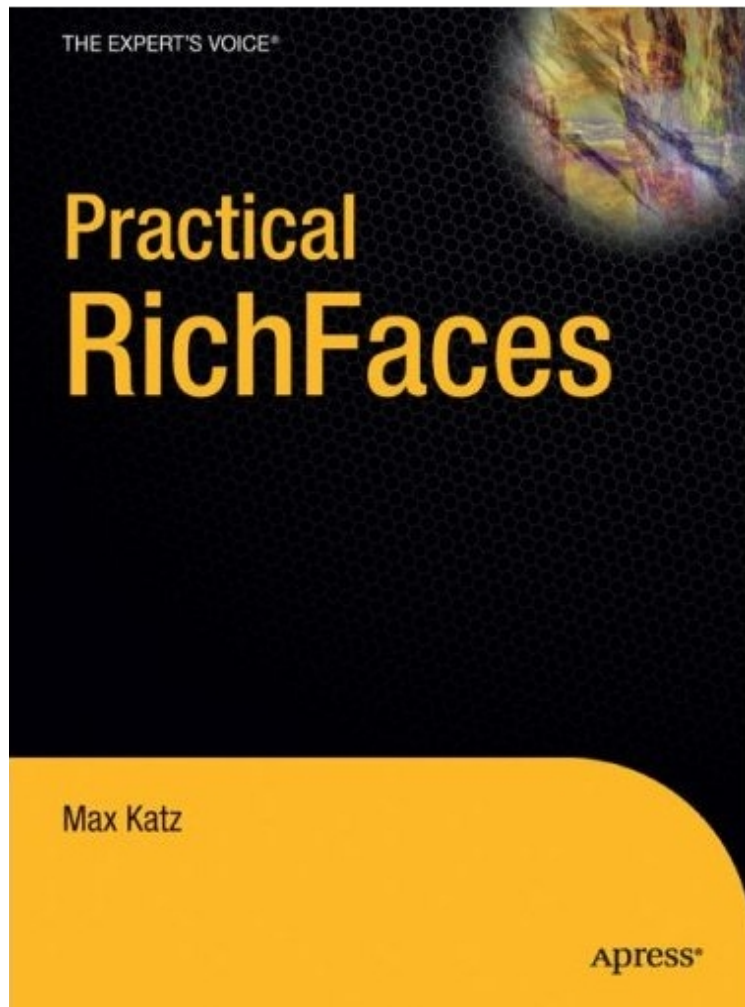
# RichFaces 4 Webinar Schedule

*June 15, 2011*

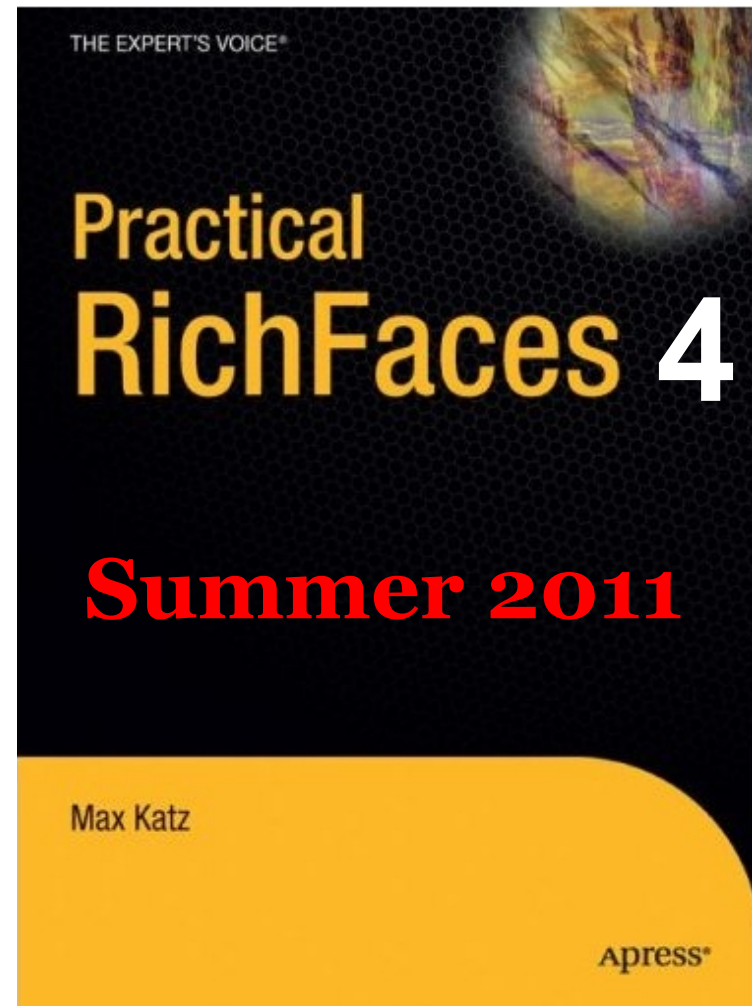
**RichFaces 4 – New and Advanced  
Features**

# Max Katz

- Senior Systems Engineer
- JSF, RichFaces, Java EE consulting, and training
- Manages [exadel.org](http://exadel.org) – Exadel's open source projects and community
- Community manager for [gotiggr.com](http://gotiggr.com) – tool for creating and sharing interactive web and mobile HTML prototypes



**Author of  
Practical RichFaces  
(Apress)**



**Lead-author of  
Practical RichFaces, 2/e  
(Apress, July 2011)**

Exadel is a global software engineering company.

- Founded in 1998, headquarters in San Francisco Bay Area
- 7 development offices in Europe
- 350+ employees

# What Exadel Does

## Services

- Professional services, rich enterprise application development, Eclipse development, custom rich component development, mobile development, training

## Products

- Open source with JBoss: RichFaces, JBoss Tools/JBoss Developer Studio; [exadel.org](http://exadel.org): Flamingo, Fiji, jsf4birt, JavaFX Plug-in for Eclipse; [gotiggr.com](http://gotiggr.com) – interactive Web and mobile HTML prototypes

# The Plan Is Simple

- 1) Review the New RichFaces 4
- 2) Moving up from RichFaces 3 to RichFaces 4



RichFaces 4 is a lightweight,  
open source framework for  
JSF 2



# RichFaces 4 – rich JSF framework

- UI components
  - a4j:\* tag library (core)
  - rich:\* tag library (UI)
  - Components' JavaScript API
- Skins
- Client-side validation (Bean Validation based)
- CDK – Component Development Kit

# RichFaces 4

100% built on top of JSF2,  
just extends functionality in  
JSF 2

# RichFaces 4

JavaScript is now entirely  
based on the popular jQuery  
library



# RichFaces 4

- All components are reviewed for consistency, usability
- Redesigned following semantic HTML principles
- Server-side and client-side performance optimization
- Strict code clean-up and review

# RichFaces 4

Zero-configuration.

Just drop RichFaces into the  
application.

# RichFaces 4

New client-side validation  
based on Bean Validation  
(JSR 303)

# RichFaces 4

New, and easy to use CDK  
(Component Development Kit),  
allows quickly to build your  
own custom rich components

# RichFaces 4

Run on: Tomcat 6/7, Resin,  
JBoss AS 6/7, GlassFish 3.x,  
WebLogic

(run on any server when JSF 2 application  
can be deployed)



# RichFaces 4

Run on: Google App Engine  
(GAE), Amazon EC2,  
CloudBees

# RichFaces 4

JSF implementations:  
Mojarra or Myfaces

# RichFaces 4

Any browser



# RichFaces 4

Tooling support through  
JBoss Tools, IntelliJ,  
NetBeans

# RichFaces versions

Version	JSF 1.1	JSF 1.2	JSF 2
RichFaces 3.1.x	●		
RichFaces 3.3.3*		●	●
RichFaces 4			●

\* Note: RichFaces 3.3.3 has basic JSF 2 support

# RichFaces 4

Let's quickly look at some RichFaces features in more detail...

# RichFaces 4 core – sending an Ajax request

a4j:ajax

a4j:commandButton

a4j:commandLink

a4j:jsFunction

a4j:poll

a4j:push

# <a4j:commandButton/Link> – built-in Ajax behavior

```
/* RichFaces button */  
<a4j:commandButton value="Save"  
    render="output"  
    action="#{bean.action}" />
```



# <a4j:jsFunction> – fire Ajax request from any JavaScript function, HTML event

```
<table>
  ...
  <td onmouseover="update('yellow')"/>
  ...
</table>

<h:form>
  <a4j:jsFunction name="update"
    action="#{bean.change}" render="...">
    <a4j:param value="param1" assignTo="#{bean.color}"/>
  </a4j:jsFunction>
</h:form>
```

# <a4j:poll> – periodically send an Ajax request

```
<a4j:poll interval="1000"  
          action="#{bean.count}"  
          render="output"  
          enabled="#{bean.pollEnabled}" />  
  
<h:panelGrid id="output">  
  ...  
</h:panelGrid>
```

## <a4j:push>

- Server-side events are pushed to client using Comet or WebSockets.
- Implemented using Atmosphere
- Provides excellent integration with EE containers, and advanced messaging services

```
<a4j:push address="topic@chat"  
    ondataavailable="alert(event.rf.data)" />
```

# RichFace 4 core – advanced rendering features

`<a4j:outputPanel>`

# <a4j:outputPanel> – auto rendered panel

```
<h:selectOneMenu value="#{bean.fruit}">  
  <a4j:ajax listener="#{bean.change}"/>  
</h:selectOneMenu>
```

```
<a4j:outputPanel ajaxRendered="true">  
  <h:panelGrid>  
    ...  
  </h:panelGrid>  
</a4j:outputPanel>
```

Rendered on every  
Ajax request



# RichFace 4 core – advanced execute features

<a4j:region>

# <a4j:region> – declaratively define execute region

```
<h:form>  
  <a4j:region>  
    <h:inputText />  
    <h:inputText />  
    <a4j:commandButton />  
  <a4j:region>  
</h:form>
```

# RichFace 4 core – more advanced features and tags

Client queue

a4j:status

a4j:param

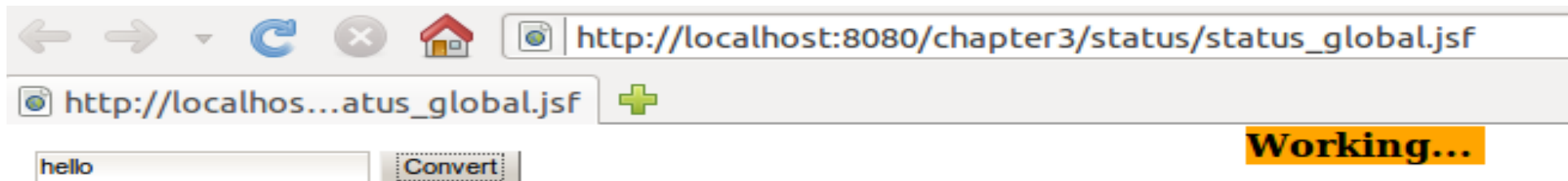
a4j:log



# RichFaces queue

- JSF 2 has very basic queue functionality
- RichFaces enhancements
  - Combining requests from the same or different components
  - Setting request delay
  - Ignoring “stale” responses

# <a4j:status> – Ajax request status



```
<a4j:status name="ajaxStatus">  
  <f:facet name="start">  
    <h:graphicImage value="ajaxStatus.jpg"/>  
  </f:facet>  
</a4j:status>
```

```
<h:form>  
  <a4j:commandButton status="ajaxStatus"/>  
</h:form>
```

## <a4j:param>

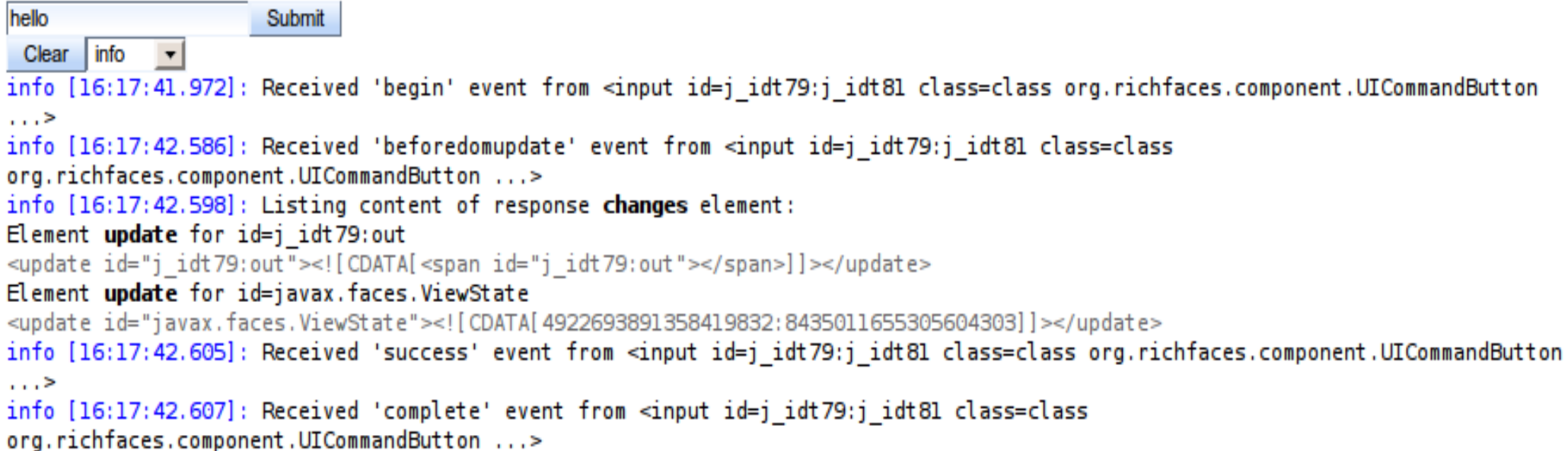
- Similar to <f:param> but simpler as it also assigns the value to a bean property automatically

```
<a4j:commandButton value="Save">  
  <a4j:param name="#{bean.product}" value="1009"/>  
</a4j:commandButton>
```

```
@ManagedBean  
public class Bean {  
    private String product;  
    // setter and getter  
}
```

# <a4j:log> – Ajax request information

- Levels:
  - debug, info, warn, error




```
hello Submit
Clear info
info [16:17:41.972]: Received 'begin' event from <input id=j_idt79:j_idt81 class=class org.richfaces.component.UICommandButton ...>
info [16:17:42.586]: Received 'beforedomupdate' event from <input id=j_idt79:j_idt81 class=class org.richfaces.component.UICommandButton ...>
info [16:17:42.598]: Listing content of response changes element:
Element update for id=j_idt79:out
<update id="j_idt79:out"><![CDATA[<span id="j_idt79:out"></span>]]></update>
Element update for id=javax.faces.ViewState
<update id="javax.faces.ViewState"><![CDATA[4922693891358419832:8435011655305604303]]></update>
info [16:17:42.605]: Received 'success' event from <input id=j_idt79:j_idt81 class=class org.richfaces.component.UICommandButton ...>
info [16:17:42.607]: Received 'complete' event from <input id=j_idt79:j_idt81 class=class org.richfaces.component.UICommandButton ...>
```

# RichFaces UI components

- Output, panels
- Input
- Menu
- Data iteration
- Tree
- Drag and drop
- Client side validation
- Miscellaneous

**Overview** JSF 2 and RichFaces 4

RichFaces is a component library for JSF and an advanced framework for easily integrating AJAX capabilities into business applications.



- 100+ AJAX enabled components in two libraries
- a4j: page centric AJAX controls
- rich: self contained, ready to use components
- Whole set of JSF benefits while working with AJAX
- Skinnability mechanism
- Component Development Kit (CDK)
- Dynamic resources handling
- Testing facilities for components, actions, listeners, and pages
- Broad cross-browser support
- Large and active community

Calendar for March, 2011. A time picker dialog is open over the 12th, showing 12:00 and OK/Cancel buttons.

Grouped list structure:

- Group 1
- Group 2
  - Item 2.1
  - Item 2.2
  - Item 2.3
  - Group 2.4
    - Item 2.5
- Group 3

	Expenses			subtotals
	Meals	Hotels	Transport	
San Jose				
25-Aug-97	\$37.74	\$112.00	\$45.00	
26-Aug-97	\$27.28	\$112.00	\$45.00	
	\$65.02	\$224.00	\$90.00	\$379.02
Seattle				
27-Aug-97	\$96.25	\$109.00	\$36.00	
28-Aug-97	\$35.00	\$109.00	\$36.00	
	\$131.25	\$218.00	\$72.00	\$421.25
<b>Totals</b>	<b>\$196.27</b>	<b>\$442.00</b>	<b>\$162.00</b>	<b>\$800.27</b>

**Person Info**

Name:

Email:

Range slider from 0 to 1000 with a value of 500.

Tree view structure:

- USA
  - United Kingdom
    - CBS Records
      - Bonnie Tylor - Hide your heart - 1988
    - Virgin redords
    - Polydor
    - CBS
    - Pickwick
    - Mucik Master
    - Island
    - A and M
    - Siren
    - Capitol
    - DECCA
    - London
  - Europe
    - Norway

Navigation: <<<< << < 1 2 3 4 5 > >> >>>>

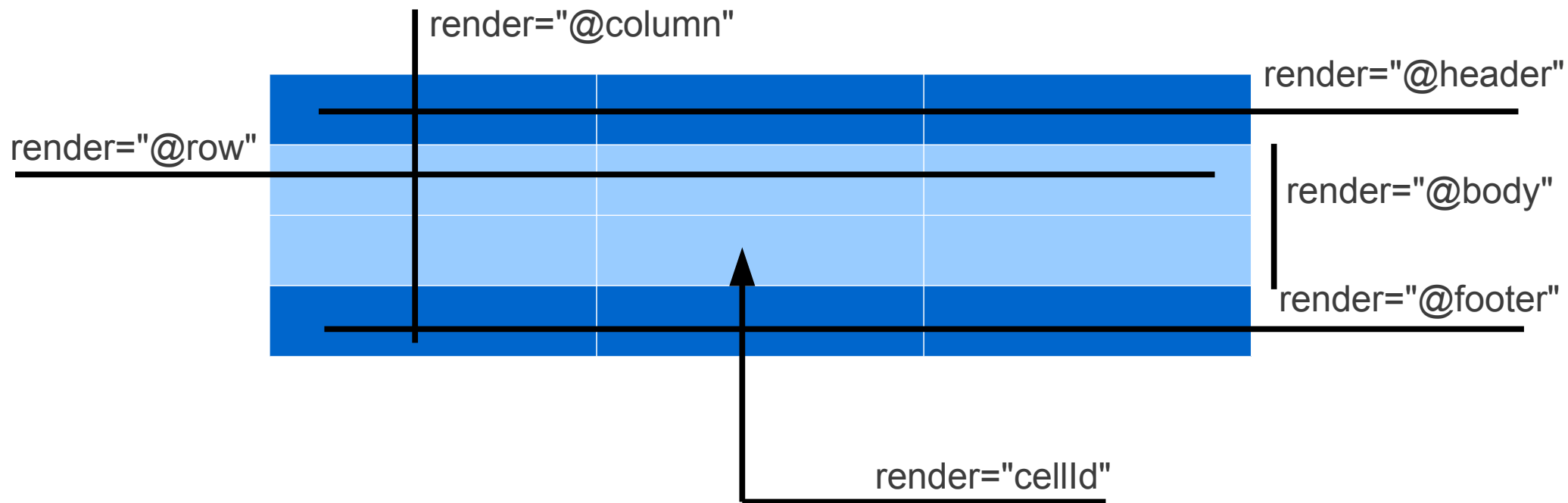
Vendor	Model	Price	Mileage	VIN
Chevrolet	Corvette	22868	67494.0	YCTWZHGYDYDJEPMKF
Chevrolet	Corvette	46135	42197.0	EYAVBBDZNMDZKXSZC
Chevrolet	Corvette	21787	77291.0	CAXBYKGOAPWMNTJSA
Chevrolet	Corvette	44008	30546.0	FCMRXFCKZTTQXOSWF
Chevrolet	Corvette	38483	64281.0	CLTQYMXAOEINWYXCX

Search dropdown menu showing results for 'nebraska':

- nebraska
- Nebraska
- Nevada
- New Hampshire
- New Jersey
- New Mexico
- New York

All RichFaces data iteration  
components support partial  
updates

# Partial table update



To render from outside the table:

`render="tableId@header"`

`render="tableId@body"`

`render="tableId@footer"`





# Client-side validation based on Bean Validation (JSR 303)

New in RichFaces 4

# RichFaces client validation

## <rich:validator>

```
public class Bean {  
    @Pattern(regexp="...")  
    private String email;  
}
```

```
<h:inputText id="email" value="#{bean.email}">  
    <rich:validator/>  
</h:inputText>  
<rich:message for="email"/>
```

# Many RichFaces components provide client-side JavaScript API

## `<rich:popupPanel>` JavaScript API

Method name	Description
<code>getTop()</code>	Return the top co-ordinate for the position of the pop-up panel.
<code>getLeft()</code>	Return the left co-ordinate for the position of the pop-up panel.
<code>moveTo(top,left)</code>	Move the pop-up panel to the co-ordinates specified with the top and left parameters.
<code>resize(width,height)</code>	Resize the pop-up panel to the size specified with the width and height parameters.
<code>show()</code>	Show the pop-up panel.
<code>hide()</code>	Hide the pop-up panel.

# Using `#rich:component()` to show/hide pop-up panel

```
<input type="button"
      onclick="#{rich:component('popup')}.show();"
      value="Open" />

<rich:popupPanel id="popup">
  <h:outputLink value="#"
    onclick="#{rich:component('popup')}.hide();
    return false;">
    <h:outputText value="Close" />
  </h:outputLink>
</rich:popupPanel>
```

# Skins

## Moscow

Moscow is the capital, the most populous city, and the most populous federal subject of Russia. The city is a major political, economic, cultural, religious, financial, educational, and transportation centre of Russia and the world, a global city. Moscow is the most populous city on the continent of Europe and the seventh largest city proper in the world, a megacity. The population of Moscow (as of 1 January 2010) is 10,562,099. [Read more](#)

## Moscow

Moscow is the capital, the most populous city, and the most populous federal subject of Russia. The city is a major political, economic, cultural, religious, financial, educational, and transportation centre of Russia and the world, a global city. Moscow is the most populous city on the continent of Europe and the seventh largest city proper in the world, a megacity. The population of Moscow (as of 1 January 2010) is 10,562,099. [Read more](#)

## Moscow

Moscow is the capital, the most populous city, and the most populous federal subject of Russia. The city is a major political, economic, cultural, religious, financial, educational, and transportation centre of Russia and the world, a global city. Moscow is the most populous city on the continent of Europe and the seventh largest city proper in the world, a megacity. The population of Moscow (as of 1 January 2010) is 10,562,099. [Read more](#)

## Moscow

Moscow is the capital, the most populous city, and the most populous federal subject of Russia. The city is a major political, economic, cultural, religious, financial, educational, and transportation centre of Russia and the world, a global city. Moscow is the most populous city on the continent of Europe and the seventh largest city proper in the world, a megacity. The population of Moscow (as of 1 January 2010) is 10,562,099. [Read more](#)

# Skins

- Lightweight extension on top of CSS
- Change look and feel of all rich component with a few minor changes
- Apply to standard JSF and HTML tags

# Ready-to-use skins

- classic
- wine
- blueSky
- ruby
- emeraldTown
- deepMarine
- plain
- japanCherry

```
<context-param>  
  <param-name>org.richfaces.skin</param-name>  
  <param-value>ruby</param-value>  
</context-param>
```



# Moving from RichFaces 3 to RichFaces 4...

# Things to know before upgrading

# JSF 1.x to JSF 2

- Migrating to RichFaces 4 will automatically mean you are also upgrading to JSF 2

# Yey! No more JSP!

- JSP no longer supported
- Must use Facelets with RichFaces 4

# Seam

- Seam 2 doesn't work with JSF 2
- Consider using Seam 3 with RichFaces 4

RichFaces filter configuration in web.xml file is no longer required

Just drop RichFaces Jars into the application

There are still optional context parameters (skins, etc)

a4j:\* tab library changes

<a4j:ajax> is 100% based on the standard <f:ajax> behavior:

RichFaces 3	RichFaces 4
<a4j:support>	<a4j:ajax>

But, the concepts stay the same:

- 1) How to send an Ajax request
- 2) Partial view processing
- 3) Partial view rendering



# Important core attribute changes

RichFaces 3	RichFaces 4
reRender	render
event=onclick	event=click
limitToList	limitRender
ajaxSingle	execute=@this
process	execute @all, @form, @this, @none, id's, EL, @region

RichFaces 3	RichFaces 4
<a4j:commandButton>, <a4j:commandLink>	<a4j:commandButton>, <a4j:commandLink>
<a4j:poll>	<a4j:poll>
<a4j:jsFunction>	<a4j:jsFunction>
<a4j:status>	<a4j:status>
<a4j:log>	<a4j:log>

RichFaces 3	RichFaces 4
<code>&lt;a4j:status&gt;</code>	<code>&lt;a4j:status&gt;</code>
<code>&lt;a4j:log&gt;</code>	<code>&lt;a4j:log&gt;</code>
<code>&lt;a4j:mediaOutput&gt;</code>	<code>&lt;a4j:mediaOutput&gt;</code>

## Other changes:

RichFaces 3	RichFaces 4
<a4j:actionParam>	<a4j:param>
<a4j:push>	<a4j:push> Component completely redesigned
<a4j:region>	<a4j:region>
N/A	<a4j:actionListener> Registers a listener on a component and allows simple method binding

No longer in RichFaces 4 as now part of JSF 2:

- `<a4j:loadScript>`
- `<a4j:loadStyle>`
- `<a4j:keepAlive>`

No longer in RichFaces 4:

- `<a4j:form>`
- `<a4j:ajaxListener>`

To redefine queue setting by individual components, a new `<a4j:attachQueue>` tag was introduced:

```
<a4j:queue />
...
<a4j:commandButton>
    <a4j:attachQueue requestDelay="1000"
        requestGroupId="ajaxGroup" />
</a4j:commandButton>
```

RichFaces 3	RichFaces 4
<code>&lt;a4j:queue&gt;</code>	<code>&lt;a4j:queue&gt;</code>
N/A	<code>&lt;a4j:attachQueue&gt;</code>

rich:\* tag library changes

# Rich inputs

RichFaces 3	RichFaces 4
<rich:calendar>	<rich:calendar>
<rich:inplaceInput>	<rich:inplaceInput>
<rich:inplaceSelect>	<rich:inplaceSelect>
<rich:inputNumberSlider>	<rich:inputNumberSlider>
<rich:inputNumerSpinner>	<rich:inputNumerSpinner>
<b>&lt;rich:suggestionBox&gt;</b>	<b>&lt;rich:autocomplete&gt;</b>
<b>&lt;rich:comboBox&gt;</b>	<b>Merged with &lt;rich:autocomplete&gt;</b>
N/A	<rich:select>



# Inputs that didn't make it...

RichFaces 3	RichFaces 4
<rich:colorPicker>	Not in Final, planned
<rich:editor>	Not in Final, planned

# Rich output/panels

RichFaces 3	RichFaces 4
<code>&lt;rich:panel&gt;</code>	<code>&lt;rich:panel&gt;</code>
<code>&lt;rich:modalPanel&gt;</code>	<code>&lt;rich:popupPanel&gt;</code>
<code>&lt;rich:panelBar&gt;</code>	<code>&lt;rich:accordion&gt;</code>
<code>&lt;rich:simpleTogglePanel&gt;</code>	<code>&lt;rich:collapsiblePanel&gt;</code>
<code>&lt;rich:tab&gt;</code> , <code>&lt;rich:tabPanel&gt;</code>	<code>&lt;rich:tab&gt;</code> , <code>&lt;rich:tabPanel&gt;</code>
<code>&lt;rich:togglePanel&gt;</code>	<code>&lt;rich:togglePanel&gt;</code>
<code>&lt;rich:toolBar&gt;</code>	<code>&lt;rich:toolbar&gt;</code>
<code>&lt;rich:progressBar&gt;</code>	<code>&lt;rich:progressBar&gt;</code>
<code>&lt;rich:toolTip&gt;</code>	<code>&lt;rich:tooltip&gt;</code>

# Outputs that didn't make it...

RichFaces 3	RichFaces 4
<code>&lt;rich:paint2D&gt;</code>	Possible with JSF 2
<code>&lt;rich:separator&gt;</code>	Use <code>&lt;hr/&gt;</code> with styling
<code>&lt;rich:spacer&gt;</code>	Doesn't satisfy semantic markup principles

# Rich selects/ordering

RichFaces 3	RichFaces 4
<code>&lt;rich:orderingList&gt;</code>	Not in Final, planned
<code>&lt;rich:listShuttle&gt;</code>	Not in Final, planned
<code>&lt;rich:pickList&gt;</code>	Not in Final, planned

# Rich menu

RichFaces 3	RichFaces 4
<code>&lt;rich:toolBar&gt;</code>	<code>&lt;rich:toolBar&gt;</code>
<code>&lt;rich:dropDownMenu&gt;</code>	<code>&lt;rich:toolBar&gt;</code>
<code>&lt;rich:panelMenu&gt;</code>	<code>&lt;rich:panelMenu&gt;</code>
<code>&lt;rich:contextMenu&gt;</code>	Not in Final, planned.

# Rich data iteration

RichFaces 3	RichFaces 4
<rich:dataTable>	<rich:dataTable>
<rich:dataGrid>	<rich:dataGrid>
<b>&lt;rich:datascroller&gt;</b>	<b>&lt;rich:dataScroller&gt;</b>
<a4j:repeat>	<a4j:repeat>
<rich:extendedDataTable> <rich:scrollableDataTable>	<rich:extendedDataTable>
<rich:dataList> <rich:dataOrderingList> <rich:dataDefintionList>	<rich:list>
N/A	<b>&lt;rich:collapsibleSubTable&gt;</b>

# Rich data iteration didn't make it

RichFaces 3	RichFaces 4
<rich:columns>	Not in Final
<rich:subTable>	Not in Final
<rich:dataFilterSlider>	Not in Final

# Validation

RichFaces 3	RichFaces 4
<code>&lt;rich:ajaxValidator&gt;</code>	Client validation: <code>&lt;a4j:validator&gt;</code>
<code>&lt;rich:beanValidator&gt;</code>	No longer needed. Support via JSF 2
<code>&lt;rich:graphValidator&gt;</code>	<code>&lt;rich:graphValidator&gt;</code>
<code>&lt;rich:message&gt;</code>	<code>&lt;rich:message&gt;</code>
<code>&lt;rich:messages&gt;</code>	<code>&lt;rich:messages&gt;</code>



Rich trees

Pretty much all version 3  
functionality is in RichFaces 4

Rich drag and drop

Pretty much all version 3  
functionality is in RichFaces 4

# Rich miscellaneous

RichFaces 3	RichFaces 4
<code>&lt;rich:componentControl&gt;</code>	<code>&lt;rich:componentControl&gt;</code>
<code>&lt;rich:jQuery&gt;</code>	<code>&lt;rich:jQuery&gt;</code>
N/A	<code>&lt;rich:hashParam&gt;</code> Groups parameters into hash parameter to be passed to components via JavaScript API.

# Rich miscellaneous that didn't make it

RichFaces 3	RichFaces 4
<code>&lt;rich:effect&gt;</code>	Not in Final
<code>&lt;rich:gmap&gt;</code>	Not in Final, planned
<code>&lt;rich:virtualEarth&gt;</code>	Not in Final, planned
<code>&lt;rich:insert&gt;</code>	Available as custom component. See RichFaces components showcase.
<code>&lt;rich:hotKey&gt;</code>	Not in Final. Use jQuery.
Layout components	Not in Final. Use YUI or HTML.

## A note on Not in Final

Component is not in 4.0.0 but many will be added in 4.x release

Get involved in RichFaces community, and let everyone know what you want!

Skins

Skins probably had the least  
changes

## Context parameter to set default application skin

RichFaces 3	RichFaces 4
org.richfaces.SKIN org.ajax4jsf.SKIN	org.richfaces.skin

## Skins in RichFaces 4

Migrated	Not migrated
<ul style="list-style-type: none"><li>•blueSky</li><li>•emeraldTown</li><li>•ruby</li><li>•classic</li><li>•japanCherry</li><li>•wine</li><li>•deepMarine</li><li>•NULL</li><li>•DEFAULT</li><li>•plain</li></ul>	<ul style="list-style-type: none"><li>•laguna</li><li>•darkX</li><li>•glassX</li></ul>



# Where can I try the new RichFaces 4?

<http://livedemo.exadel.com/richfaces4-demo>



Project Site | Project Wiki | RichFaces Community Space | [Download](#) | [Sources](#)

Select Skin: [wine](#) [ruby](#) [japanCherry](#) [emeraldTown](#) [deepMarine](#) [classic](#) [blueSky](#)

Ajax Action

Ajax Queue

Ajax Output/Containers

Validation new

Data Iteration new

Trees new

Output/Panels new

Menus

*Inputs and Selects*

rich:autocomplete

**rich:calendar**

rich:inputNumberSlider

**Simple Calendar**

Client Side Styling/Disablement

Data Model Support

The **rich:calendar** component allows you to select a date using a monthly calendar widget. It is possible to use the component in a **popup or inline mode**. In popup mode the calendar is initially rendered as an input for date with a button on the right side to trigger the popup. For inline mode, the monthly calendar is rendered on the page initially.

April, 2011							
	Sun	Mon	Tue	Wed	Thu	Fri	Sat
14	27	28	29	30	31	1	2
15	3	4	5	6	7	8	9
16	10	11	12	13	14	15	16
17	17	18	19	20	21	22	23
18	24	25	26	27	28	29	30
19	1	2	3	4	5	6	7
Today							

Disabled:

Popup Mode:

Apply Button:

Select Locale

US  DE  FR  RU

Select Date Pattern:

d/M/yy HH:mm ▼

# How can we help with RichFaces

- Web development with RichFaces
- Version 3 to 4 migration
- Performance tune-up
- Custom component development
- On-site training

Training	Days
JSF 1.2, 2	1-2
RichFaces 3, 4	1-2
JSF and RichFaces	2-3
RichFaces 3 to 4	1-2

That's it for RichFaces, there  
is one more thing I want to  
show you.

The screenshot displays the Tiggr mobile app development tool interface. The top menu bar includes 'Project', 'Screen', 'Edit', 'Align', 'Arrange', 'Help', and a 'Web Preview' dropdown. The project name is 'CheckOutThisApp' and the current screen is 'login'.

The main workspace shows a prototype of a mobile app screen titled 'CoolMobileApp'. It features a login form with the following elements:

- Email:** A text input field containing 'user@email.com'.
- Password:** A text input field containing '\*\*\*\*\*'.
- Sign In:** A button with a right-pointing arrow.
- About:** A button with a plus sign and the text 'About'.

On the left side, there is a 'Mobile' component palette with various UI elements: Button, Input, Textarea, Label, Link, Radio Button, Checkbox, a slider, an On/Off toggle, a Select dropdown, a list of items (1. Item, 2. Item, 3. Item), Image, and Grid. At the bottom of the palette are 'One', 'Two', and 'Three' tabs.

On the right side, the 'MobileScreen properties' panel is visible, showing settings for the current screen:

- Operations:** 'Save As Screen Template' button.
- Custom:** 'Screen type: Change...' button.
- Fixed width:**
- Has Header:**
- Has Footer:**
- Theme:** Gray (dropdown)
- Show screen borders:**
- Common:** Name: login (text field), Custom size: , Dimension: W 320 H 480 (text fields), Predefined screen size: 320x480 (dropdown).

At the bottom of the workspace, there is a chat window titled 'Chat :: Max Katz' with 'Messages' and 'Activities' checked and '1 user online'. The chat log shows:

- [00:46:24] Max Katz : User joined chat
- [00:46:30] Max Katz : Switch screen to tablets
- [00:46:30] Max Katz : Switch screen to phones
- [00:46:31] Max Katz : Switch screen to login

The chat input field contains '[00:47:32]' and has 'Send' and 'Clear' buttons. Below the chat window are 'Toggle Chat' and 'Show Annotations' buttons.

Create your first interactive prototype at <http://gotiggr.com>

Tiggr ([gotiggr.com](http://gotiggr.com)) is a web application for building and sharing interactive web and mobile HTML prototypes



max@exadel.com

Thank  
you!



@maxkatz



mkblog.exadel.com



gotiggr.com