



JSFTemplating

<https://jsftemplating.dev.java.net>

Ken Paulsen

Senior Staff Engineer
Sun Microsystems, Inc.



Agenda

- Welcome!
- What is JSFTemplating?
- Creating Pages
- Additional Features
- Developer Tips
- Questions?

What is JSFTemplating?

- JSF Page Framework?
- JSF Renderer Framework?

Both!

Oh... and it's Open Source too.

<https://jsftemplating.dev.java.net>

Quotes

"This is going really fast and I may just drop my current JSF/Facelets implementation on the floor for JSFTemplating."

"...developed a GUI in a couple hours that would had took a day in JSF... (interesting.)"

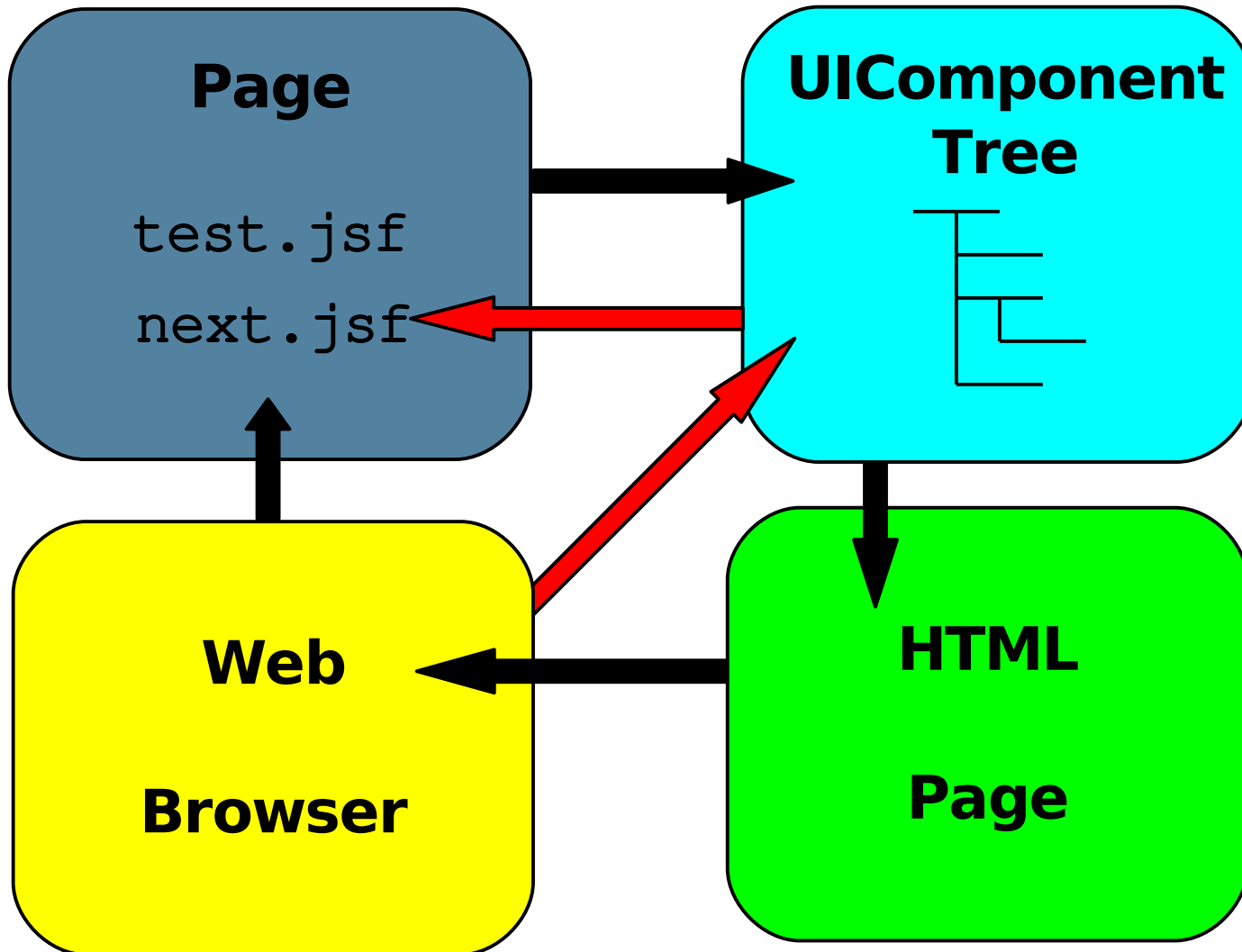
Key Features

- Multiple format support
- #include
- pageSession
- Events / Handlers
- Factories
- Inter-operates w/ JSP
- Better Performance
- Instant changes
- AJAX Support

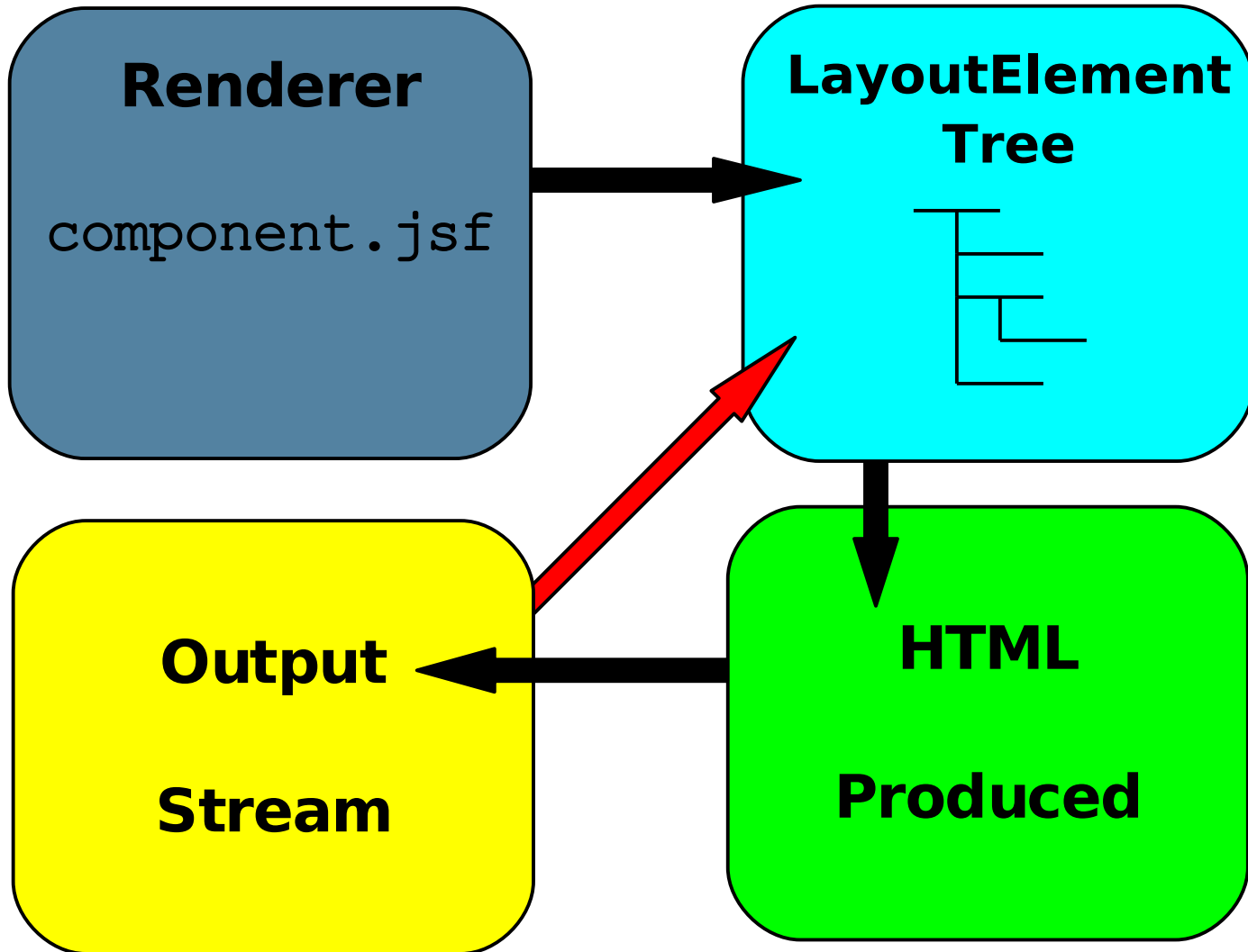
High Level Overview

- Template-based description of Renderer or UIComponent tree
- Template becomes "LayoutElement" tree (meta-data)
- LayoutViewHandler creates UIComponent tree
- TemplateRenderer walks tree

High Level Overview – Page



High Level Overview – Renderer



How to create JSFTemplating pages

- 1) Setup standard JSF environment
- 2) Put `jsftemplating.jar` in `WEB-INF/lib`
- 3) Put `.jsf` files in docroot (or classpath)
- 4) Try out your new `.jsf` pages!

Hello World!

```
"Hello World!"
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
"HelloWorld.jsf" 1 line
```

DEMO

Hello World!

Another Example Page

```

<sun:page>
  <sun:html>
    <sun:head id="head" />
    <sun:body>
      <sun:form id="form">
        "<p>&nbsp;</p>"
        <!if $attribute{in} & ! (#{in} = abc) >
          "<b>You typed: #{in}, try typing: abc</b>"
        </if>
        <!if !#{in}>
          "<b>Type something and click 'Go!'"</b>"
        </if>
        <!if #{in}=abc>
          "<b>Congratulations! You did it!"</b>"
        </if>
        "<br /><br />"
        <sun:textField id="in" value="#{requestScope.in}" />
        "<br /><br />"
        <sun:hyperlink text="Go!" />
      </sun:form>
    </sun:body>
  </sun:html>
</sun:page>

```

DEMO

Example #2

Creating Pages

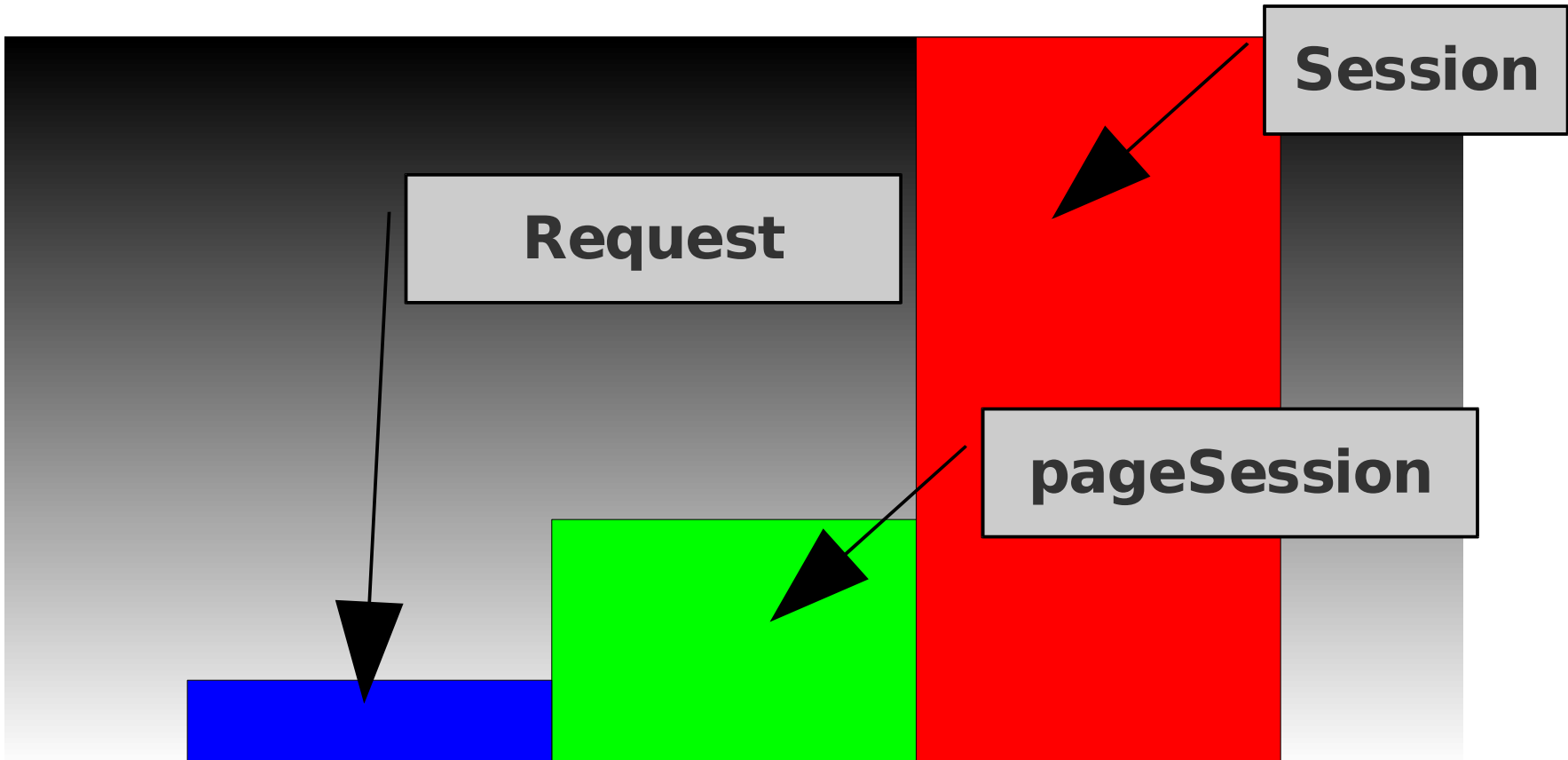
- No compile time
- Instant changes (Debug Mode)
- No JSP problems
- Events and handlers are available
 - > (hint: “info.jsf” to see a list)
- #include files

DEMO

#include

Page Session

Scope Lifespan



DEMO

pageSession

Events

- initPage
- beforeCreate / afterCreate
- beforeEncode / encode / afterEncode
- command
- decode
- + more

Handler example (defining)

```
@Handler(id="getUserResponse",
    input={@HandlerInput(name="userInput", type=String.class)},
    output={@HandlerOutput(name="response", type=String.class)})

public static void calculateResponse(HandlerContext context) {
    // Get the input
    String in = (String) context.getInputValue("userInput");

    // Business logic
    String resp = "Type something and click 'Go!'";
    if (in != null) {
        if (in.equals("abc")) {
            resp = "Congratulations, you did it!";
        } else {
            resp = "You typed " + in + ", try typing 'abc'.";
        }
    }

    // Set the response
    context.setOutputValue("response", resp);
}
```

Handler example (using)

```
<!beforeEncode
    getUserResponse (usrInput="{in}" response=>$pageSession{resp});
    println(value="We responded with: #{resp}");
/>
```

```
<!command
    getUserResponse (usrInput="{in}" response=>$pageSession{resp});
    println(value="We responded with: #{resp}");
/>
```

DEMO

events / handlers

Quote

“One thing I absolutely believe is a waste of space is how SJSC2 has create four .java files per page.”

Handlers

- Chainable
- Reusable
- Mappable I/O
 - > EL Enabled
 - > Automatic type conversion
 - > Required field checking

Reduces the amount of Code

What Handlers?

Q: How can I find out what handlers are available?

A: **info.jsf**

DEMO

[info.jsf](#)

ComponentFactory's

- Alternate approach to “binding”
- Parameterizable
- Allows you to “extend” components w/o extending
 - > DropDown
 - > Table
 - > Tree

ComponentFactory Example

```
@UIComponentFactory("sun:button")
```

```
public class ButtonFactory extends ComponentFactoryBase {  
    public UIComponent create(FacesContext ctx, LayoutComponent  
        descriptor, UIComponent parent) {  
        UIComponent comp = ctx.getApplication().  
            createComponent("com.sun.webui.jsf.Button");  
        if (descriptor.getOption("primary") == null) {  
            comp.setValueBinding("primary",  
                context.getApplication().  
                    createValueBinding("#{true}"));  
        }  
        if (parent != null) {  
            addChild(ctx, descriptor, parent, comp);  
        }  
        setOptions(ctx, descriptor, comp);  
        return comp;  
    }  
}
```

Can I use JSF JSP's with JSFTemplating?

Yes

(but you may not want to)

DEMO

Speed Test

Developer Tips

- Set DEBUG to true:

```
<context-param>  
  <param-name>  
    com.sun.jsftemplating.DEBUG  
  </param-name>  
  <param-value>true</param-value>  
</context-param>
```

- Set your Log level:

```
<property name="com.sun.jsftemplating"  
value="FINER" />
```

Questions?

JSFTemplating

<https://jsftemplating.dev.java.net>

Ken Paulsen

ken.paulsen@sun.com